

INTERNATIONAL TRADE CENTRE



UNITED NATIONS
CONFERENCE ON
TRADE AND DEVELOPMENT

WORLD TRADE
ORGANIZATION



New features of the WinISIS system

Guidebook for users of CDS/ISIS for DOS

prepared by

Zbigniew M. Nowicki

Expert in Information Technologies

Geneva, February 2001

Preface

This guidebook is part of the International Trade Centre UNCTAD/WTO (ITC) efforts to assist partner organizations in developing countries in managing efficiently their information services. It presents new features of the CDS/ISIS for Windows (WinISIS) system, which is developed by UNESCO. It covers innovations introduced in consecutive versions of the system till the version 1.4. This version was still in final beta tests when this book was prepared.

This guidebook is addressed to experienced users of CDS/ISIS for DOS and therefore good knowledge of all functions of this system and experience in its applications are initial requirements. The book will provide readers with information on new features available in the Windows version of the system and with examples of their applications. The new features considerably increase user-friendliness of the system. However, many of innovations of the formatting language are quite advanced and therefore these features can be fully utilized only bases designed by the users who are conversant with this language.

For further information please contact the

Trade Information Section
Division of Product and Market Development
International Trade Centre UNCTAD/WTO
Palais des Nations
1211 Geneva 10
Switzerland
Fax (4122) 7334439
E-mail: itcreg@intracen.org
Url: <http://www.intracen.org>

***CDS/ISIS Micro and mini version is an information retrieval software
produced and distributed by UNESCO***

Table of contents

1. Introduction	1
2. General overview	2
2.1. Database processing	2
2.2. Data entry and searching	3
2.3. Formatting language	3
2.4. User friendliness	4
2.5. CDS/ISIS dynamic link libraries	5
3. Data entry	6
3.1. Data validation	6
3.2. Other functions of the validation file	7
3.2.1. Pick-up list management	7
3.2.2. Automatic change of the data entry worksheet	9
3.2.3. Automatic entry of data	10
3.3. Exemplary validation file	10
4. Retrieval	12
4.1. Query formulation	12
4.1.1. Expert search interface	12
4.1.2. Guided search interface	13
4.2. Displaying search results	14
4.2.1. Standard mode	14
4.2.2. Tabular mode	15
4.2.2.1. Changing the layout of the tabular display window	16
4.2.2.2. Explanation of query formulation rules	17
4.2.3. Highlighting of retrieved terms	18
4.3. Using the dictionary window	18
4.3.1. Customization of the dictionary display	19
4.3.2. Hiding the prefixes	20
4.4. Using the special versions of WinISIS	21
5. Sorting and printing	22
6. Database definition	24
6.1. Creation of a data entry worksheet	24
6.2. Creation of a format	25
6.3. Creation of a FST table	26
7. Formatting language	27
7.1. Output of selected field occurrences	27
7.2. New functions of the formatting language	27
7.2.1. Numeric functions	27
7.2.2. String functions	28
7.3. Use of external formats	29
7.4. Referencing other databases	30
7.5. Working with format variables	30
7.6. Using fonts	31
7.6.1. Fonts and colors definition	31
7.6.2. Character formatting	32
7.7. Paragraph formatting	33
7.8. Outputting pictures	34
7.9. Hypertext links in formats	35
7.9.1. Navigating among records	36
7.9.2. Changing the current display format	36
7.9.3. Printing	36
7.9.4. Executing external programs	37

7.9.5. Opening another database	37
7.9.6. Executing system actions	38
7.9.7. Working with windows	38
7.9.7.1. Text windows	38
7.9.7.2. Loading format files	39
7.9.7.3. View windows	40
7.9.7.4. Identifying windows	41
7.9.8. Polymorphic formats	42
7.10. Interaction with the user	42
7.10.1. Prompting for the user's input	42
7.10.2. Using dialog windows	43
8. System management	45
8.1. System files	45
8.2. System parameters	47
8.2.1. Access paths	48
8.2.2. Database management	49
8.2.3. Access restrictions	49
8.2.4. Data entry	50
8.2.5. Searching and displaying search results	51
8.2.6. Printing	52
8.2.7. Layout of main window and database window	52
8.2.8. Font definitions	53
8.2.9. Language dependent parameters	53
8.2.10. Parameter file redirection	54
8.2.11. Miscellaneous parameters	54
8.3. Access restrictions	55
8.3.1. Restricted access to system functions	55
8.3.2. Database access restrictions	55
8.3.3. Restricted access to records or fields	55
8.3.4. Use of view mode	57
8.4. Character conversions	57
8.5. System personalization	58
Annex. The <i>Multilateral Trading System</i> database	60
1. MTS database	60
1.1. Record structure	60
1.2. Data entry worksheets	62
1.3. Validation file	62
1.4. Dictionary	63
1.5. Display and print formats	64
1.5.1. MTS format	64
1.5.2. SHORT format	64
1.5.3. FULL format	66
1.5.4. DES format	69
1.5.5. DESCR format	69
1.5.6. D1 format	71
1.5.7. D2 format	72
1.5.8. PUB format	72
1.5.9. PUBL format	73
1.5.10. TITLE format	73
1.5.11. PRINT format	73
1.5.12. EXPLAIN format	75
1.6. The MTS.EXP file	75
2. TITT database	76
2.1. Display formats	77
2.1. Print formats	80
3. LANG database	80
4. PUB database	81
4.1. Display and print formats	81

1. Introduction

WinISIS is a Windows version of the CDS/ISIS system (Computerized Information Service / Integrated Scientific Information System) which was originated at ILO the early 60's and since 70's is maintained and developed by UNESCO. In 1985 the first microcomputer version of CDS/ISIS was launched and since then it was disseminated to over 20 000 institutions in various countries. The first Window version was distributed for testing in May 1995 and the first WinISIS version officially realized was version 1.31 launched in November 1998. UNESCO distributed with this version two documents:

- *CDS/ISIS for Windows Reference manual (version 1.31),*
- *WinISIS – Supplement to the Reference Manual (versions 1.0 to 1.3).*
Title of the reference manual is a bit misleading – in fact it is a slightly improved version of the system documentation for version 1.0 that was released in November 1997 and this is why this manual was accompanied by the supplementary documentation. The similar documentation for version 1.4 is entitled:
- *WinISIS 1.4 Changes and News.*

The chapter 2 of this guidebook presents a general overview of the most important new features of the WinISIS system. The next four chapters provide detailed descriptions of the new features available during data entry, retrieval, sorting and printing and database definition. The seventh chapter is devoted to the new features of the formatting language. It discusses among others the hypertext commands that enable creation of interactive display formats. The last chapter is devoted to management of the WinISIS system. It describes among others the numerous new system parameters of WinISIS and methods of the system personalization.

All examples of application of the new WinISIS features have been taken from the MTS database developed by ITC. This database stores bibliographic descriptions of various types of publications related to the Multilateral Trading System. The MTS database contains references to various auxiliary databases maintained in ITC, such as the TITT database that stores the *Thesaurus of International Trade Terms*. The annex presents in detail structure of the MTS database and of all the auxiliary databases. This fully functional application of WinISIS can be applied as an example by the readers who will design new WinISIS databases for own purposes or adjust for WinISIS structures of existing databases that have been designed using CDS/ISIS for DOS.

2. General overview

This chapter gives an overview of the most important innovations introduced in the Windows version of the CDS/ISIS system. The most obvious innovation is the graphical user interface typical for all Windows applications. Other new features have been presented briefly below in the separate subchapters devoted to database processing, data entry and searching, formatting language and user-friendliness of the package. The last subchapter describes the ISISDLL and WISIS.DLL dynamic link libraries that can be used to prepare very powerful Windows applications processing CDS/ISIS databases.

2.1. Database processing

WinISIS uses the same database structure as CDS/ISIS for DOS. Databases created by DOS versions of the CDS/ISIS system do not require any changes to be processed by the Windows version of this system. Moreover, if display and print formats as well as FST tables do not use new features of the formatting language of WinISIS, a database created by the Windows version of the CDS/ISIS system can be processed by its DOS version.

New database processing features of WinISIS include:

- many databases can be opened simultaneously, in particular the same database can be opened several times; each of these databases is displayed in a separate window; if the same database was opened in various windows each of them can display another record (or the same record shown according to another display format);
- easy access (from the pull-down menu) to databases used recently; users can be forced to select a database from the database list only (all users can have access to the same list or each of them can apply his/hers personalized database list);
- maximum record size has been increased almost 4 times (30 KB in the Windows version as compared with 8 KB in the DOS versions) what greatly simplifies processing of full-text databases (however, a CDS/ISIS database containing records bigger than 8 KB cannot be longer processed in the DOS environment);
- global record processing operations (add a field, delete a field and replace text in a field contents) are available from the system utilities menu.

Starting from version 1.1 of WinISIS the database definition wizard greatly simplifies creation of new databases and modification of existing ones. In particular, for new databases:

- the print format assistant automatically creates a default format according to the selected format template,
- the dictionary assistant automatically creates a very simple inverted file FST.

These two assistants use information on database fields stored in the field definition table. In consequence, a simple database can be quickly created by a novice having no prior knowledge of the CDS/ISIS system. The database definition wizard facilitates also creation and maintenance of complex databases by the advanced users of the package (see the chapter *Database definition* for detailed description of this wizard).

Starting from the version 1.4 UNESCO is going to distribute special versions of WinISIS system that will maintain databases with non-standard length of inverted file terms. Such a version may be useful for searching CDS/ISIS databases for which standard length of CDS/ISIS dictionary terms is not sufficient. However, the user who applies a special version for creation of a database should be aware that this database can be processed neither by normal versions of

WinISIS nor by CDS/ISIS for DOS (see the subchapter *Using special versions of WinISIS* in the chapter *Retrieval*).

2.2. Data entry and searching

The following new features are available during data entry:

- by dragging them and dropping onto the edit box, data from some other windows (e.g. inverted file terms from the dictionary window or file names from the file manager window) can be easily entered into the record being currently edited;
- record contents can be validated during data entry; various conditions to be met by particular fields and the whole record are specified in the database validation file and the record being edited can be saved only if its contents fulfil these conditions.

In version 1.4 of WinISIS the database validation file may contain additional specifications of:

- names of data entry worksheets applied automatically for editing various categories of records;
- values that can be entered into the specified field (the so called pick-up list);
- values assigned automatically to the specified fields at the very beginning or at very end of the record modification.

These features are discussed in the chapter *Data entry* in detail.

The following new features are available during searching:

- apart from the standard search interface (called expert search, because it allows to formulation of sophisticated queries, including free-text ones) the guided search interface is available for inexperienced users of the package;
- in dictionary window the whole dictionary or only inverted file terms taken from the specified field can be displayed;
- retrieved terms can be highlighted in record contents during display;
- retrieved records can be marked during browsing the search results and then only the marked records can be displayed or printed;
- size of all fonts used to display record contents can be increased or decreased proportionally (using the *Zoom in* or *Zoom out* functions from the *Browse* menu),
- queries can be saved on a disk and recalled during the subsequent search sessions,
- starting from version 1.4 only records from the specified MFN range can be searched by a free-text query.

The tabular mode of displaying search results is available starting from version 1.3 of WinISIS. In this mode one part of the screen shows briefly all records retrieved and the another part of the screen presents details of one record from this list. The user can easily move between these records, mark them and print records marked or all records retrieved.

These features are discussed in the chapter *Retrieval* in detail.

2.3. Formatting language

New features of the formatting language include:

- increased length of a format (up to 26,000 characters) and its output (up to 64,000 characters);
- access to separate occurrences of repeatable fields;
- availability of new numerical and string functions, including the *LR* function enabling access to all records from which the specified inverted file term was taken (the *L* function available previously enables access to the first such record only);

- the *REF* function can be used to output not only contents of various records from the current database, but also to output data from other databases (the database being referred is specified by the database qualifier that may follow the *REF*, *L* and *LR* functions);
- a new format can be built as concatenation of several existing formats;
- in display and print formats new commands can be applied to specify type, name, size and color of fonts used to output data as well as their properties (bold, italic, underlined);
- special commands can be applied to make output of other formatting commands more attractive by drawing colorful boxes around paragraphs, centering/justifying output texts and displaying pictures or graphics (stored as BMP or JPG files).

Starting from version 1.0 of WinISIS the formatting language includes some commands typical for a high level programming language. Possibility to use:

- string and numerical variables,
- the *WHILE* command

together with the *FOR* command available previously allow preparing sophisticated formats that use such constructs as variables and various types of loops.

The most powerful innovation in the formatting language of WinISIS is availability of various hypertext commands that can be used to create interactive display formats. These commands are executed upon user's request only and therefore operations of a display format that contains such commands are controlled by the user. Hypertext commands enable among others:

- to display the current record according to another format,
- to display another record,
- to display a picture or play a sound track or movie linked to the current record.

In the latter case a hypertext command is used in fact to execute from WinISIS another Windows application with appropriate input parameters (such as an Internet browser that displays a specified WWW page). Starting from version 1.1 display formats of WinISIS can be even more interactive due to possibility to request the user to enter specified data and use this input as a parameter of the hypertext command.

Availability of hypertext commands considerably increases functionality of display formats applied by WinISIS. Additionally, when a format name is provided as the parameter of a hypertext command it is possible to change contents of this format just before its execution. Such formats are called polymorphic formats. Use of these formats greatly facilitates creation of interactive display formats for complex WinISIS applications such as an application comprising several databases with logical links between their records.

New features of the formatting language are discussed in the chapter *Formatting language* in detail.

2.4. User friendliness

Various new features makes the WinISIS system more user friendly. These features include:

- availability of various sets of system menus (so called profiles) and possibility to change the current profile during the session;
- numerous new system parameters; moreover, settings of the system parameters can be read and immediately applied by the system during the session; starting from version 1.3 WinISIS provides a user friendly interface that enables quick on-line modification of various system parameters;
- presence of a help file (at the time being the help file in English is always used irrespective of the current language of conversation).

In version 1.4 external programs can be called from system menus. These can be general purpose application programs (such as an Internet browser called to display the specified WWW page) or data-aware application programs that use WISIS.DLL library (see below) to cooperate with the WinISIS system in processing CDS/ISIS databases. Use of the system menus and system parameters is discussed in the chapter *System management* in detail.

2.5. CDS/ISIS dynamic link libraries

ISISDLL is an abbreviated name of the CDS/ISIS dynamic link library that can be used to prepare **stand-alone Windows applications** processing CDS/ISIS databases. In a sense this library plays in Windows environment a role similar to the role the CDS/ISIS Pascal language played in the DOS environment. However, there are important differences:

- while CDS/ISIS Pascal is a very limited subset the Pascal language extended by the CDS/ISIS library functions and procedures, routines of the ISISDLL library can be applied in any programming language (e.g. Visual Basic, C++ or Delphi) that supports DLL calls what gives the programmer access to all features of the programming language selected;
- while CDS/ISIS Pascal programs can be applied only in the CDS/ISIS system, application programs containing ISISDLL calls are genuine Windows applications that are independent of the WinISIS system;
- while procedures of the CDS/ISIS Pascal library allow to open one database a time as well as a single record and a single inverted file a time, a program using the ISISDLL library can open several databases simultaneously and access many records and many inverted file terms at the same time.

The ISISDLL has been developed by the Latin America and Caribbean Centre on Health Sciences Information (BIREME) that is located in São Paulo, Brazil. This library is available on the FTP site of BIREME under the following addresses:

<http://www.bireme.br/isis/P/isisdll.htm> (the page contains short information in Portuguese)

ftp://brme.bireme.br/isis_dll

Version 1.4 of the WinISIS system is distributed with another dynamic link library called **WISIS.DLL**. This library can be used to prepare **data-aware application programs** that cooperate with the WinISIS system in processing CDS/ISIS databases. A program using the WISIS.DLL library can get from the system various information about the current state of all databases being opened and can issue to the system various commands that request WinISIS to perform specific actions on these or other bases. Application programs containing WISIS.DLL calls are called **WinISIS plug-in programs** and these programs can be executed from a system menu (see the subchapter *System files* in the chapter *System management*).

Since preparation of application programs using CDS/ISIS dynamic link libraries requires programming knowledge, functions of these libraries will not be discussed in this book.

3. Data entry

In WinISIS **data entry worksheets** are much simpler than in CDS/ISIS for DOS. Generally, in contrast to a CDS/ISIS for DOS worksheet that describes data entry fields, a WinISIS worksheet simply defines a collection of fields to be filled out. In consequence a worksheet is not divided into pages. The fields are described using their names defined in the FDT table. All fields are presented in the same way - fields cannot be distinguished graphically and there are no title fields (texts without data entry boxes). If a help message is associated with the field being edited, this message is automatically displayed in the very bottom of the data entry window.

Creation and modification of data entry worksheets is much simpler in WinISIS than in CDS/ISIS for DOS. All fields defined in the FDT table are displayed on the left panel (fields belonging to the worksheet being edited are dimmed there) and the right panel shows the fields of the worksheet. Selected fields can be moved from one panel to another by clicking a button and there is also a button that transfers all the FDT fields to the worksheet. A new field is always placed as the last field of the worksheet, but there are buttons that place a selected field before its predecessor or after its successor in the worksheet.

The **data entry window** consists of the following elements:

- the MFN box, the navigator buttons, the *New record* button, the *Save* button, the *Options* submenu as well as the *Worksheet* combo box and *Add field* combo box are shown in the top of the window,
- the field edit box is the biggest box in the window (this box can be resized by positioning the cursor over its bottom edge and dragging it down or up),
- the consecutive fields are displayed in separate boxes below the field edit box,
- the record status panel and help panel are shown in the bottom of the window.

In order to edit an existing field the operator simply clicks its button. A content of this field is displayed in the field edit box where edition will take place. Occurrences of a repeatable field are shown in separate boxes preceded by sequential numbers of these occurrences. Moreover, the last occurrence of such a field is preceded by the *Occurrence* button. Clicking this button presents for editing a new empty occurrence of the field. During edition of an existing record WinISIS displays by default only the fields filled out in this record. A button with the magnification glass (visible in the very bottom of the data entry window) displays all fields of the current worksheet, including the empty ones. In order to fill out a non-existing field the operator can click this button, select a desired field and enter its value. Alternatively he can select a non-existing field in the *Add field* box. Such a field will appear empty in the worksheet and it will be presented for editing.

Note that WinISIS is much **more restrictive during data entry** than CDS/ISIS for DOS. In particular, the system checks during editing various characteristics of the field being edited that are stored in the FDT table. In consequence, WinISIS will not allow to enter for instance:

- more than one occurrence of a non-repeatable field,
- a subfield with an identifier not defined in the FDT table.

3.1. Data validation

In WinISIS a separate file called **dbase.VAL** (where *dbase* is the database name) stores information on **validation rules of fields and records** of this database. Each line of the validation file has the following general form:

tag: format

where *tag* is a tag of the field to be validated,
format is a CDS/ISIS format.

The **field validation** format is executed immediately after creation or modification of the specified field and if this format produces any output than this text is displayed as an error message. WinISIS will not leave the field being edited as long as the field validation format produces any output.

In the MTS database three fields stores consecutive lines of the publisher address - field 61 (1st line), 62 (2nd line) and 63 (3rd line). In order to ensure that address information is stored in consecutive lines (starting from the first line) the MTS.VAL file can contain the following lines:

```
62: if a(v61) then 'Enter 1st line of the publisher address first' fi
63: if a(v61) then 'Enter 1st line of the publisher address first' else
    if a(v62) then 'Enter 2nd line of the publisher address first' fi
fi
```

(in the file the format validating contents of the field 63 is written in a single line). WinISIS will not store the second line of address (field 62) if the first line (field 61) is absent and it will not store the third line (field 63) if either the first (field 61) or second (field 62) line of address is missing.

In the **last line of the validation file** the field tag may be omitted and then this line defines the **record validation** format. This format is executed before storing the record in the master file and WinISIS will not save the current record as long as the record validation format produces any output.

The MTS database stores bibliographic descriptions of various types of documents. The field 4 stores the record type identifier and this field must be filled in every record. To ensure that this condition is satisfied the last line of the MTS.VAL file may have the following contents:

```
:if a(v4) then 'You must specify "Type of record"' fi
```

In consequence WinISIS will not store a record in which the field 4 is absent.

By default WinISIS validates during data entry contents of each modified record and enables within a field several occurrences of a given subfield. In version 1.4 this behaviour can be changed by applying special system parameters (see the subchapter *System parameters* in the chapter *System management*). In consequence, during data entry WinISIS will validate all records and/or treat repeated subfields are treated as errors.

3.2. Other functions of the validation file

In version 1.4 of the WinISIS system lines of a validation file may contain other instructions than the validation formats for the fields and records. These instructions define:

- a list of values that can be entered into the specified field (the so called pick-up list),
- actions to be executed on the current record before its presentation for data entry,
- actions to be executed on this record when its editing was finished.

Separate sections of this subchapter describe these instructions in detail and give examples of their applications.

3.2.1. Pick-up list management

The line defining a list of values that can be entered into the specified field has the following general form:

tag: choice::format

where *tag* is a tag of the field,
format is a CDS/ISIS format.

When this field is edited a special button is displayed in the data entry window along with the field edit box. If the user clicks this button a list values specified by the format is presented in a separate small window. The user can choose a value of this list by clicking it and when he clicks the *OK* button the small window closes and the selected value is transferred to the edit box where it replaces the existing value of the field.

By default WinISIS interprets the output text of the format in the following way:

- the first line contains the title of the window,
- the consecutive lines contain consecutive values.

In order to define a pick-up list of record types supported in the MTS database the following lines could be added to the MTS.VAL file:

```
4:choice::'Record types'/'Monograph'/'Serial'/'Periodical'/  
>:'Article in periodical'/'Article in monograph'/'Web site'/'
```

In contrast to the field and record validation formats, the format defining a pick-up list may be written in several consecutive lines, provided that each continuation line starts from a space followed by the > sign and a colon.

The default behavior of a pick-up list can be changed by placing one or more modifiers after the **choice:** keyword. The table given below describes these **modifiers**:

Modifier	Meaning
multi:	the user may select more than one value from the list,
add:	selected value(s) will be added to the existing value of the field,
repeat:	each of selected values is stored as a new separate occurrence of the field,
notype:	contents of the field cannot be entered from the keyboard; the pick-up list window is displayed when the user presses a key;
firstdescribe:	a value is represented by a pair of lines rather than a single line of the format output text; the first line of this pair contains the description displayed in the window and the second line contains the value itself;
<>:	each of selected values is automatically enclosed into triangular brackets,
//:	each of selected values is automatically enclosed into slashes,
noansi:	the DOS to Windows conversion (see the section <i>Character conversions</i> in the chapter <i>System management</i>) is performed on the selected values; this modifier should be used for a list of values prepared using a DOS text processor;
sort:	values are automatically sorted before displaying them in the window;
files:	consecutive lines of the format output text represent names of text files storing values (each value in a separate line) rather than values themselves;
call:	the second line of the format output text represents the name of a WinISIS plug-in that will be called when the user clicks a button displayed along with the field edit box (see below).

In the MTS database the first two characters of the field 4 are interpreted a record type identifier. The list of values defined for this field above is not correct, because the first two characters of these values are not unique. These texts should be displayed in the window, but actual values stored in this field will differ from the descriptive texts displayed in the pick-up list window. Moreover, the users should not be allowed to enter values of this field from the keyboard. In consequence, the lines defining this list in the MTS.VAL file will have the following contents:

```
4:choice:notype:firstdescribe::'Record types'/'Monograph'/'Monograph'/'  
>:'Serial'/'Serial'/'Periodical'/'Periodical'/'  
>:'Article in periodical'/'APeriodical'/'  
>:'Article in monograph'/'AMonograph'/'Web site'/'Web site'/'
```

A pick-up lists defined in the validation file cannot be too long, because values of such a list are stored in the output area of the validation format. Moreover, the simple window in which such a

pick-up list is displayed is not suitable for long lists, containing hundreds or thousands of items. Such a list should be processed by a WinISIS plug-in program that will send back information necessary for filling the field. In the MTS database such a program might be used for the *Subject descriptors* field that should contain descriptors from the *Thesaurus of International Trade Terms*. This thesaurus contains nearly 1200 terms and therefore its terms cannot be displayed as a pick-up list defined in the validation file.

3.2.2. Automatic change of the data entry worksheet

A line defining actions to be executed on the current record before its presentation for data entry has the following general form:

begin: *format*

where *format* is a CDS/ISIS format. This format must produce a text containing commands of the CISIS update language of BIREME¹. These are the valid commands of this language:

Command	Meaning
d.	make the current record logically deleted
d*	delete all fields of the current record
dn	delete all occurrences of the field <i>n</i> where <i>n</i> is a field tag
dn/occ	delete occurrence <i>occ</i> of the field <i>n</i>
an#value#	add the character string <i>value</i> as a new occurrence of the field <i>n</i> (note that # serves as a character string delimiter, this sign can be replaced by any character that does not appear in <i>value</i>)
hn m value	add the character string <i>value</i> that is <i>m</i> bytes long as a new occurrence of the field <i>n</i>

WinISIS executes the commands specified by the format output text for each record just before its presentation in the data entry window. Apart from the CISIS update language commands the format output text may contain the following commands:

GetWorksheetName()	returns a character string containing a name of the current data entry worksheet (name of this command is case sensitive),
worksheet:wsname/	makes the data entry worksheet named <i>wsname</i> the current worksheet (this command must be followed by a carriage return).

A new record that has not been saved yet in a CDS/ISIS database has the MFN number 0 what makes possible to execute some formatting commands for newly created records only. The MTS.VAL file may contain the following line defining actions to be executed on each record of the MTS database before its presentation for data entry:

```
begin:if mfn=0 then 'worksheet:MTS'/'a2#',date(2),'#'/ else
if v4.1='M' then 'worksheet:MONOG'/ else
if v4.1='S' then 'worksheet:SERIA'/ else
if v4.1='P' then 'worksheet:PERIO'/ else
if v4.2='AM' then 'worksheet:AMON'/ else
if v4.2='AP' then 'worksheet:APER'/ else
if v4.1='W' then 'worksheet:WWW'/ fi fi fi fi fi fi fi
```

(in this file all these commands are written in a single line). In consequence WinISIS will automatically apply an appropriate data entry worksheet for edition of each record of the MTS database. The MTS worksheet is used for creation of a new record and a worksheet corresponding to the type of record being edited is used for modification of an existing record. Moreover, for a newly created record the current date is stored as the creation date of this record (field 2). Function *date(2)* of the formatting language returns the current date (see the section *String functions* in the chapter *Formatting language*) and this character string is provided as an argument of the command *a2* of the CISIS update language.

¹ BIREME (Latin America and Caribbean Center on Health Sciences Information in São Paulo, Brazil) cooperates closely with UNESCO in development of the WinISIS system. CISIS is the library of utility programs that process CDS/ISIS databases. The CISIS library has been also used in the WWWISIS program that acts as a server for CDS/ISIS databases on Internet.

3.2.3. Automatic entry of data

A line defining actions to be executed on the current record when its editing was finished has the following general form:

end: format

where *format* is a CDS/ISIS format. This format must produce a text containing commands of the CISIS update language (see the preceding section). Moreover, the format output text may contain the command

IsChanged() that returns logical value *True* when contents of the current record has been changed during data entry (name of this command is case sensitive).

In the MTS database two separate fields store important dates for a record:

- field 2 stores the record creation date,
- field 900 stores the date of the last update operation on this record.

The MTS.VAL file may contain the following line defining actions to be executed on the current record when its editing was finished:

```
end:if IsChanged() and mfn<>0 then
  if p(v2) then
    if v2<>date(2) then 'd900a900#',date(2),'#'/ fi
  else
    'a2#',date(2),'#'/
  fi
fi
```

(in this file all these commands are written in a single line). In consequence, in any modified record of the MTS database WinISIS will automatically store the current date as:

- the date of record creation if this date was not recorded previously,
- the date of the last update operation if the creation date is present and it is different from the current date.

Note that the CISIS update language does not contain a field replace command. Therefore if the user wants to replace contents of a field he must first delete its existing value and then add a new value. This is why the command *d900* is executed before *a900*. No action is done for a newly created record when its editing is finished, because the creation date was entered before presentation of such a record for data entry.

3.3. Exemplary validation file

This section presents the full contents of an exemplary validation file for the MTS database. In the table shown below each cell represents a single line of the MTS.VAL file.

```
begin:if mfn=0 then 'worksheet:MTS'/'a2#',date(2),'#'/ else
if v4.1='M' then 'worksheet:MONOG'/ else
if v4.1='S' then 'worksheet:SERIA'/ else
if v4.1='P' then 'worksheet:PERIO'/ else
if v4.2='AM' then 'worksheet:AMON'/ else
if v4.2='AP' then 'worksheet:APER'/ else
if v4.1='W' then 'worksheet:WWW'/ fi fi fi fi fi fi fi
```

```
4:choice:notype:firstdescribe::'Record
types'/'Monograph'/'Monograph'/'
'Serial'/'Serial'/'Periodical'/'Periodical'/'
'Article in periodical'/'APeriodical'/'Article in
monograph'/'AMonograph'/'
'Web site'/'Web site'/'
```

```
62:if a(v61) then 'Enter the first line of address' fi
```

```
63:if a(v61) then 'Enter the first line of address' else
if a(v62) then 'enter line 2' fi fi
```

```
:if v4.5='Click' then
'Edit the "Type of record" field and choose a value from the list'
fi
```

```
end:if IsChanged() and mfn<>0 then
if p(v2) then if v2<>date(2) then 'd900a900#',date(2),'#'/ fi else
'a2#',date(2),'#'/ fi fi
```

Note that the record validation line issues an error message when the record type (field 4) starts from the character string *Click*. This condition is checked because the default value of this field is:

Click a button with red arrow to display the list of available values

This contents of the field 4 serves merely as an explanation and it should be replaced by a value chosen from the pick-up list. WinISIS will not store a record with this explanatory text because its presence signals that the record type identifier is missing.

4. Retrieval

4.1. Query formulation

WinISIS offers two ways of query formulation:

- the **expert search** interface enables execution of any query, including a free-text query;
- the **guided search** interface is used to execute simple queries containing up to four search elements.

These two search interfaces are discussed in separate sections of this subchapter. In each interface search elements can be transferred to a query from the **dictionary window** that is described below in a separate subchapter.

Apart from functions providing access to the search interfaces the *Search* menu of WinISIS includes the entries that enable to:

- **save a query** executed during the current session,
- **recall a saved query**.

The latter operation executes a query and automatically displays its results in the database window, even if search results are displayed only upon the user's request (using an appropriate system parameter the system administrator may instruct WinISIS to show search results immediately after execution of each query, see the next subchapter for details).

By default the system shows merely the number of records retrieved by a query (the so called number of hits). However, if the system parameter 103=0 is applied, during execution of a query WinISIS shows in a separate window detailed information on the search results in a similar way as does CDS/ISIS for DOS.

In the formatting language of WinISIS 0 the field selector **V0** identifies any field of the current record. This non-existing field can be used in a **free-text query** to specify a text to be searched in all fields. For instance, the query

```
? v0: 'abc'
```

can be used to retrieve all records containing the character string *ABC*. In version 1.4 of WinISIS a free-text query can have two optional parameters that restrict the range of records searched by this query. In consequence a free-text query has one of the following general forms:

```
? #search_no boolean_expression
```

```
? *mfn1,mfn2 boolean_expression
```

The first form is the only form allowed in CDS/ISIS for DOS (specification of the search number is optional). In the second form the parameters *mfn1* and *mfn2* represent the MFN numbers of the first and the last record searched respectively.

4.1.1. Expert search interface

This interface enables execution of any query formulated in the CDS/ISIS search language. The default expert search window contains the following elements:

- the search operator buttons,
- the *Search expression* edit box,
- the search history box,
- the *Dictionary* button,
- the *ANY terms* button,
- the *Execute*, *Clear* and *Help* buttons.

If the tabular mode of displaying search results is applied this window contains apart from these elements

- the *Search tips* button,
- the *Display* button.

(see an appropriate section of the next subchapter for detailed description of the tabular mode).

In the *Search expression* edit box the blinking cursor shows the position on which a new query element will be entered. WinISIS facilitates in various ways creation of query elements. These facilities include:

- clicking the *Dictionary* button shows the dictionary window with all terms defined in the inverted file of the current database; double clicking such a term copies it as a new query element (if necessary this term is automatically preceded by the + operand; moreover, if the dictionary window shows only elements taken from a specified field, the selected term is automatically suffixed with an appropriate field qualifier);
- clicking the *ANY terms* button shows all such terms defined for the current database; double clicking such a term copies it as a new query element (if necessary this term is automatically preceded by the + operand);
- the search history box shows all queries formulated during the current session; these queries can be used in the following ways:
 - clicking a query inserts into the *Search expression* edit box the search number associated with this query (if necessary this number is automatically preceded by the + operand); however, if the query was executed in a database other than the current one, clicking it inserts the query itself rather than its number;
 - dragging a query and dropping it over the *Search expression* edit box appends the selected query enclosed into parentheses to the current search expression (if necessary this sub-expression is automatically preceded by the + operand);
 - double clicking a query displays its results.
- clicking one of the search operator buttons inserts an appropriate operator on the current query position; in spite that the AND, OR and NOT buttons are available the appropriate operators are *, + and ^ respectively; among the search operator buttons is the # button, clicking this button shows a window with the list of queries formulated during the current session and clicking one of elements from this list inserts into the *Search expression* edit box the search number associated with the selected query (if necessary this number is automatically preceded by the + operand).

The *Search history* window shows in separate lines the summaries of all queries executed during the current session. These queries are shown in reverse order of their execution, i.e. the last query is shown in the first line etc. Each line has the following format:

#*search_no* (*dbase*) T=*hit_no* : *search_expression*

where *search_no* is the sequential number assigned to this query and *hit_no* is the number of records retrieved according to the query.

4.1.2. Guided search interface

In contrast to the expert search interface that enables experienced users of the system formulation of sophisticated queries, the guided search interface was designed for inexperienced users who need assistance during query formulation. The guided search interface allows them to formulate easily simple queries containing up to four search elements.

Such a query is written in a table consisting of four rows. Each row represents a single query element and an operator connecting it with the following element, if any. This table contains the following columns:

- **searchable fields**

The combo box shown in this column allows to specify a field in which the query element will be searched. By default the query elements are searched in all fields. If a field has been specified, the dictionary window shows the inverted file terms taken from this field only.

- **search elements**

The text box shown in this column stores the query element. In spite that search terms can be entered from the keyboard, in order to increase the search efficiency the users should be encouraged to use the terms from the dictionary window only. When the cursor is positioned in a search element box clicking the *Dictionary* button shows the window with all the inverted file terms defined in of the current database that were taken from the field specified in the *Searchable fields* column. Double clicking such a term uses it as the search element (if necessary this term is automatically is automatically suffixed in the query with an appropriate field qualifier).

- **operators**

The combo box shown in this column allows to specify the operator that connects this query element with the following one (no operator follows the last query element). The logical operators AND, OR and NOT can be used and the default operator is AND.

A search expression consisting of these four query elements can be optionally preceded by the search number associated with one of queries executed previously. The user selects such query using the *Previous search* combo box. This box is followed by the combo box specifying the logical operator that connects this search number with the other query elements.

The user executes the query using the *Execute* button. The number of records retrieved according to this query is shown in the box labeled *Number of hits*. The user can display the query results by clicking the *Display* button. If the tabular mode of displaying search results is applied this window contains apart from these elements the *Search tips* button.

4.2. Displaying search results

By default WinISIS shows results of a query upon user's request. When the user executes the function *Search results* from the *Browse* menu the system shows first a window with the list of queries formulated during the current session and only when the user clicks one of elements from this list WinISIS displays results of the selected query. By clicking the *Display* button in the search window the user requests displaying results of the last query. Moreover, by applying the system parameter 105=1 the system administrator may instruct the system to display search results immediately after execution of each query.

WinISIS offers two modes of displaying search results that are discussed in the following sections.

4.2.1. Standard mode

By default search results are displayed in the database window. In the very bottom of this window the following message is visible:

Search #*search_no* : (*i* / *m*)

where *search_no* is the sequential number assigned to the query which results are being displayed, *i* is the sequential number of the current record in the set of all records retrieved and *m* is the total number of records retrieved. If the database window shows results of a query, the navigator buttons work on the set of all records retrieved, i.e. clicking the *First* or *Last* button displays the first or last record of this set rather than the first or last record in the whole database. The system shows in this window all records of the current database again when the user executes the function *Database* from the *Browse* menu.

The records retrieved according to the query being currently displayed can be either printed or exported. These operations are executed by the *Print* and *Export* functions that are available in the *Database* menu. The first page of the print dialog window contains the *Current browse set* option and the export dialog window contain the *Search results* option. If an appropriate option is checked only the records retrieved according to the query being currently displayed are either printed or exported.

When the database window shows a search results the **Mark button** is visible in the top of this window next to the *Format* box. Clicking this button marks the current record (if the record is a marked record the *V* sign is displayed on the *Mark* button) if it was not marked previously or removes a mark if the current record that has been already marked. Only marked records can be browsed when the *Marked records* option has been checked in the *Browse* menu (all retrieved records are displayed again when the *All records* option has been checked in this menu). WinISIS enables marking the records retrieved according to various queries and all these marks are available during the current session.

Marking defines a subset of all records retrieved according to the specified query. Such a subset can be either printed or exported. When some records have been marked in the set of all records retrieved according to the query being currently displayed and the *Marked records only* box has been checked in the print or export dialog window, WinISIS processes only the marked records during the requested operation.

WinISIS offers a **variant of the standard mode** in which search results are displayed in the database window and the message shown in very bottom of this window is repeated in red color in the first line of the window. This line contains as well the hypertext link *<STOP>* and when the user clicks this link the system displays all database records again. This mode of displaying search results is set when the system parameter 135 is applied (see in the chapter *System management* the section *Searching and search results display* in the subchapter *System parameters*). This effect can be also achieved by checking the option *Alternate search result display* in the *System settings* window that can be accessed from the *Configure* menu.

4.2.2. Tabular mode

In contrast to the standard mode when search results are displayed in the normal database window, in the tabular mode WinISIS displays retrieved records in a special window that contains two panes. The left pane shows all records retrieved in a table where each row represents a single record. Initially the right pane shows some instructions on using the tabular mode, but when the users clicks a row in the left pane, the right pane shows contents of this records using the current display format. In the left pane each row begins from the little blue icon with the + sign. Clicking this icon marks the current record (if the record is marked the row background is navy blue) if it was not marked previously or removes a mark for the record that has been already marked. The ↑ and ↓ buttons shown in the bottom of the window change the order of records shown in the left pane. By default these records are displayed in the descending order of their MFN numbers. Clicking the ↓ button changes this order to ascending, while clicking the ↑ button changes the order to descending again. A small button with the red stop sign visible in the bottom of the window closes the tabular display window and shows again all records the current database in the database window. By dragging a line separating two panes of the tabular display window the user can increase the width of one pane and simultaneously decrease the width of another pane.

The *Options* button shown in the top of the tabular display window provides the user access to the following functions:

- print the current record,
- print all marked records,
- print all retrieved records,
- export results,
- stop browsing results.

In contrast to the *Print* function available from the *Database* menu these print functions have a very few parameters. The small print parameters window contain the following elements:

- the output device box,
- the *Format* box,
- the output file box,
- the *OK.* and *Cancel* buttons

Apart from the current printer the text files with Windows and DOS character codes can be specified as an output device. Note that if output is directed a text file the name of this file must be specified the output file box and if the specified file already exists in the WinISIS work directory the existing file will be overwritten without any warning. By default the current display format is used to output data, but the *Format* box allows to select any of the formats defined for the current database.

WinISIS applies the tabular mode of displaying search results when the system parameter 980 is set to 1. This effect can be also achieved by checking the option *Display search results in a table* in the *System settings* window that can be accessed from the *Configure* menu. Parameter 144 is another system parameter that controls display of search results in the tabular mode. By default WinISIS displays results in a table if more than one record was found and uses the default database window to display a single record retrieved. If the system parameter 144=1 is applied the tabular display window is used to show search results irrespective of the number of records retrieved.

4.2.2.1. Changing the layout of the tabular display window

By default the current display format applied in the database window is used to display record contents in the right pane of the tabular display window and in the left pane the 6 digit long MFN number followed by the first field defined in FDT table of the current database is displayed as a short description of each record retrieved. The user can display another field instead of the first one by choosing an appropriate field from the *Tag* box. However, if the current display format contains hypertext commands, these commands may be useless in the tabular display window (some of them will even not work there, for detailed description of hypertext commands see the subchapter *Hypertext links in formats* in the chapter *Formatting language*). Moreover, the display format used in the right pane is the default format used in the small print window. Therefore if the display formats applied in the database window contain hypertext links it is recommended to use a default print format to show record contents in the right pane of the tabular display window.

The default layout of the tabular display window can be changed for a given database by applying special system parameters that are stored in the ***dbase_name.EXP*** file. This file is located in the WinISIS database directory and it may contain the following parameters:

500=full format name

Defines the name of a display format used to show record contents in the right pane of the tabular display window.

502=short format

Defines the default format used to display short descriptions of records retrieved in the left pane of the tabular display window. Specifying a value of this parameter the database administrator should take into account that only the first line of output text of this format will be shown in the left pane and that this format may not contain character formatting commands.

504= order of records

Defines the default order of records in the left pane. This parameter can have one of the following values:

- 0 records are displayed in ascending order of their MFN numbers,
- 1 records are displayed in descending order of their MFN numbers (default value).

506=the *Format* box is visible

Specifies whether the *Format* box is visible in the top of the tabular display window. This parameter can have one of the following values:

- 0 the *Format* box is not visible,
- 1 the *Format* box is visible (default value).

The MTS.EXP file contains the following parameters that define layout of the tabular display window for the MTS database:

```
500=print
502=mhl,'MFN=',f(mfn,1,0)," "v20
```

The *PRINT* format is used to show record contents in the right pane of the tabular display window (in contrast to the display formats this format does not contain hypertext links) and in the left pane the MFN number followed by the title (field 20) is displayed as a short description of each record retrieved.

4.2.2.2. Explanation of query formulation rules

If the tabular mode of displaying query results is applied, the search windows may contain the **Search Tips** button that displays a view window with explanation of query formulation rules for the current database. This button is visible if the SYSPAR.PAR file contains the parameter 981 and the format specified by this parameter has been defined for the current database.

The system parameter 981 gives the common name of a special format defined for various databases (if database files are stored in separate subdirectories, the formats files having the same name may have a different contents for each database). In the MTS database the *EXPLAIN* format contains the text explaining query formulation rules for this database. This format has the following contents:

```
f2,title('How to search the MTS database ?'),
{fs40,b,qc,'Searching the MTS database'}/#
{b,'MTS'},' is a bibliographic database aimed at facilitating research ',
'on the basic documents on the ',{i,{b,'M'},'utilateral ',{b,'T'}},
'rading ',{b,'S'},'ystem' },'. These publications constitute the core ',
'documentation base of the Reference Centers established by ITC under ',
'the JITAP project. The MTS database contains also descriptions of other ',
'documents (monographs, serials, periodicals, articles and Internet sites) ',
'related to the subject.'/#
'The following elements of the MTS database are searchable:.'/
{m(100,100),{i,b,'Type of record,','/words from '{b,i,'Author(s) personal,','/
'Type of material,','/Record status,','/
'words from '{b,i,'Author organization(s),','/
'words from '{b,i,'Title,','/
'words from '{b,i,'Title of the source}',' (for articles),'/
'words from '{b,i,'Former title,','/
'Frequency,','/Call number,','/IFS code','/Publisher,','/
'Year published,','/words from '{b,i,'Series,','/Document no,','/
'Sales no,','/Subject descriptors,','/Languages,','/ISBN,','/ISSN.'/'}}#
'Click the ',{i,'"Dictionary"'},
' button and select the appropriate field. Then type first characters of a
phrase you are looking for or press the > button several times until you
can see this term below. In order to apply a term in a query double click
it or drag it and drop over the search expression box.'/#
'If you like to build a complex query by combining several elements, press
an appropriate operand button before selecting the second and next
elements. Apply parentheses to change the standard order of operand
application. Remember that parentheses must be balanced (use the same number
of opening and closing parentheses).'/
```

To ensure that the *Search tips* button provides access to these explanations from the search windows the parameter

```
981=EXPLAIN
```

must be used in the SYSPAR.PAR file. The *EXPLAIN* format shows the explanatory text in English irrespective of the current language of conversation. However, it is possible to prepare a format that will check the conversation language being currently used and will show an explanatory text in this language. If English, French and Spanish are allowed languages of conversation, this is the schematic structure of such a format:

```
s0:=(&clang('')),
if s0='EN' then
  'Explanation in English'
else
```

```

if s0='FR' then
  'Explanation in French'
else
  if s0='SP' then
    'Explanation in Spanish'
  else
    'Sorry, no explanation is available in the current language of
    conversation'
  fi
fi

```

This format uses the following *CLANG* program in the CDS/ISIS Pascal language that returns a code of the current language of conversation:

```

Program CLANG ( inp : string; lw, occ : real; out : string )[FORMAT];
begin
  out:=LANG;
end.

```

4.2.3. Highlighting of retrieved terms

If the system parameter 142 has been applied WinISIS will highlight retrieved inverted file terms when search results are displayed. If a search term contains several words only the first word of such a term will be highlighted. WinISIS will highlight all occurrences of this word in the database window or in the right pane of the tabular display window.

If in the MTS database records containing the subjects descriptors *AFRICA SOUTH OF SAHARA* and *EUROPEAN UNION* were retrieved, then all words *AFRICA* and *EUROPEAN* will be highlighted during the search result display. This is an exemplary record retrieved according to this query where the words highlighted are shown in bold.

```

1997 16 000 REG
Haarlov J
REGIONAL COOPERATION AND INTEGRATION WITHIN INDUSTRY AND TRADE IN
SOUTHERN AFRICA.
Aldershot: Avebury, 1997. xix, 254 p. (Making of Modern Africa)

Deals with general approaches to regional cooperation and integration; outlines main
characteristics of industry and trade in the region and analyses SADC's attempts in this area;
identifies overall World Bank policy and practice on regional cooperation relevant to
Southern Africa; discusses regional implications of World Bank's structural adjustment
policies with Mozambique as a country case.

Subject descriptors: AFRICA SOUTH OF SAHARA ; REGIONAL COOPERATION ; SADC
English
GBP 39.95 ISBN 1-85972-412-4
Avebury, Aldershot, Hants GU11 3HR, UK

```

Note that the word *AFRICA* is highlighted in title, series statement and abstract in spite that these fields were not searched.

The parameter 142 may specify two numbers that identify the foreground and background color respectively in the current table of colors (for colors definition see the first section of the subchapter *Using fonts* in the chapter *Formatting Language*). Specification of the background color is optional and if it is omitted black background is applied (if the default table of colors is used). Of course, foreground and background colors should be different, otherwise the highlighted terms will not be visible.

4.3. Using the dictionary window

The dictionary window is displayed when the user clicks the *Dictionary* button available in the database window and in both search windows. Initially the dictionary window shows all the inverted file terms, but by selecting a field from the box shown in the top of the window the user requests WinISIS to show only terms taken from this field (application of the system parameter 104=1 is required to achieve this effect). The user can display consecutive pages of the dictionary using the < and > buttons shown in the very bottom of the dictionary window. A dictionary page is its portion that fits into the dictionary pane (note that the size of this pane changes when the dictionary window is resized). If the user looks for a particular inverted file term he may enter the first letters of this term in the term box shown below the searchable fields box. The system shows an appropriate part of the dictionary whenever contents of the term box changes and therefore in many cases entering several initial characters will be sufficient to check whether a specified term belong to the dictionary. In version 1.4 the *Printer* button visible between the < and > buttons enables to output the whole inverted file or its part to the specified output device.

If the dictionary window was open from the search window or data entry window, the dictionary terms can be easily copied to the search expression box, the current search element box or the field edit box. Copying can be done by double clicking a dictionary term or by dragging a term and dropping it over the appropriate box.

By default all terms of the dictionary are shown, but the terms taken from other fields than the selected one are dimmed and cannot be selected. In order to ensure that only the inverted file terms taken from the specified field are displayed in the dictionary window, the database administrator should apply the parameter

104=1

in the SYSPAR.PAR file.

4.3.1. Customization of the dictionary display

The searchable fields box shown in the top of the dictionary window lists all searchable fields of the current database. This list presents these fields in the order they are defined in the FDT table and for each field its name from the FDT table appears on the list. If the field identifier defined in the FST table does not correspond to a field tag defined in FDT, such a field is shown in the very end of the list as *Field id* where *id* is the field identifier.

In version 1.4 of WinISIS these default descriptions of searchable fields in the given database can be changed by application of the system parameters 560 in the ***dbase_name.EXP*** file (see the section *Changing the layout of the tabular display window* in the previous subchapter). The parameter 560 may appear several times and it has the following general form:

560=identifier description

The parameter defines a new description to be used in the searchable fields box for the field which has been indexed with the specified identifier. The following system parameters may be used in the MTS.EXP file in order to change the default names of the selected searchable fields of the MTS database:

```
560=5 Author
560=10 Author organization
560=20 Title (full)
560=292 Descriptor (full)
560=1020 Title (words)
560=1292 Descriptor (words)
```

In consequence the list of searchable fields will contain the following items:

<i>Author</i>	instead of	<i>Author(s), persons</i>
<i>Author organization</i>	instead of	<i>Author(s), organizations</i>
<i>Title (full)</i>	instead of	<i>Title(s)</i>
<i>Descriptor (full)</i>	instead of	<i>Subject descriptors</i>
<i>Title (words)</i>	instead of	<i>Field 1020</i>
<i>Descriptor (words)</i>	instead of	<i>Field 1292</i>

Note that the field 20 (titles) and 292 (subject descriptors) have been indexed twice using different indexing techniques and the inverted file terms generated with the technique 5 have the identifiers corresponding to the field tags, while the terms generated with the technique 8 have the identifiers 1020 and 1292 respectively.

In order to enable specification of searchable fields **descriptions in various languages** the following variant of the system parameter 560 is applied:

560=0 language_code

Such a parameter may appear several times and each occurrence of this parameter should immediately precede a group of the parameters that define descriptions of the searchable fields in the given language. In the MTS.EXP file the following parameters may be applied:

```
560=0 EN
560=5 Author
560=10 Author organization
560=20 Title (full)
560=292 Descriptor (full)
560=1020 Title (words)
560=1292 Descriptor (words)
560=0 FR
560=5 Auteur
560=10 Auteur collectif
560=20 Titre (entier)
560=292 Descripteur (entier)
560=1020 Titre (mots)
560=1292 Descripteurs (mots)
```

The English section (language code *EN*) is followed by the French section (language code *FR*) and both sections contain the same field descriptions in these two languages.

By default all inverted file terms are shown in the dictionary window when the user selects the "All fields" option in the searchable fields box. In version 1.4 of WinISIS this option can be inactivated for users of a given database by application of the system parameter **555=1** in the *dbase_name*.EXP file.

4.3.2. Hiding the prefixes

In CDS/ISIS for DOS the only way to ensure, that inverted file terms taken from a given field are in the inverted file display grouped together and can be easily identified, was to define a unique **inverted file prefix** for each of these groups. The FST table defining contents of the inverted file for the MTS database contains among others the following lines:

```
20 5 '\TIT=\',MHL,V20+|,|
1020 8 '\TIT:\',MHL,V20+|,|
292 5 '\DES=\',(v292/)
1292 8 (|\DES:\|v292/)
```

In consequence the inverted file of this database contains the following groups of terms:

Prefix	Group
<i>TIT=</i>	full titles
<i>TIT:</i>	words from titles
<i>DES=</i>	full descriptors
<i>DES:</i>	words from descriptors

In WinISIS appearance of prefixes in the dictionary window may be both confusing and inconvenient for the users – if the prefix is not typed in the term box no terms are shown in the dictionary window. In version 1.4 the inverted file prefixes may be hidden by application of the system parameters 550 in the *dbase_name*.EXP file (see the section *Changing the layout of the tabular display window* in the previous subchapter). This parameter may appear several times and it has the following general form:

550=identifier prefix

The parameter defines the inverted file prefix that will not appear in the dictionary window when the user selects in the searchable fields box the field that has been indexed with the specified identifier.

In the MTS.EXP file the following 550 parameters have been used:

```
550=20 TIT=  
550=1020 TIT:  
550=292 DES=  
550=1292 DES:
```

In consequence none of the prefixes used in the inverted file of the MTS database will be shown in the dictionary window when an appropriate field is selected by the user and terms without prefixes can be typed in the term box.

4.4. Using the special versions of WinISIS

UNESCO is going to distribute upon request special versions of WinISIS 1.4 that will maintain databases with **non-standard length of inverted file terms**. The inverted file of a standard CDS/ISIS database contains:

- short terms that are 10 characters long,
- long terms that are 30 characters long.

A special version of WinISIS maintains databases with other lengths of inverted file terms, for instance 20 and 60 characters. The lengths of inverted file terms maintained by the version of WinISIS being currently used are shown in the window displayed by the *View configuration* function available from the *Configure* menu.

The special version of WinISIS cannot use inverted files created by CDS/ISIS for DOS and standard versions of WinISIS. Therefore after installing a special version of WinISIS the database administrator should first use the *VF update* function from the *Database* menu of this version to **recreate inverted files of all databases** that will be used by this special version of the WinISIS system. However, the administrator should take into account that these databases **cannot be processed** directly neither by any standard version of WinISIS nor by CDS/ISIS for DOS.

In consequence, by applying a special version of WinISIS the database administrator gives up database portability that is an important feature of the family of CDS/ISIS systems. A CDS/ISIS database developed using a special version of WinISIS can be directly processed only by the persons applying a special version of the system with the same length of inverted file terms.

5. Sorting and printing

In contrast to the sorting and printing parameters of CDS/ISIS for DOS that are collected using two system worksheets, the sorting and printing parameters of WinISIS are stored on five pages of the print dialog. The last page contains the sorting parameters that are exactly the same as in CDS/ISIS for DOS. The first four pages contain various printing parameters that are described below.

General parameters

This page allows for specification of:

- the set of records to be printed
This can be either the set of records being displayed (if the retrieved records are displayed only the records marked can be printed) or the records specified by their MFN numbers. The later specification is much more flexible than in CDS/ISIS, because it may contain several specifications of MNF ranges (see the WinISIS reference manual for details).
- the output device
The list of output devices includes the default printer of the Windows system, two types of text files and a Postscript file. The text files can contain character codes used either in Windows (ASCII file Windows ANSI) or DOS (ASCII file DOS OEM).
- the print format
By default WinISIS uses the default format of the current database as the print format. However, the user can choose another format from the list of all formats defined for this database. Moreover, the user can modify the selected format before its application and even save the modified format contents.

The page contains the *Recall*, *Save*, *Preview*, *Print*, *Cancel* and *Help* buttons. The *Save* button stores the current set of printing and sorting parameters in a special file. The parameters from such a file may be applied later during next sessions of WinISIS by clicking the *Recall* button. The files storing the printing and sorting parameters of WinISIS have the extension WPR and they play a similar role to the user defined print and sort system worksheets in CDS/ISIS for DOS. The *Preview* button gives the user an idea how the printout will look like because it shows the first page of the printout.

Presentation parameters

The page allows for specification of the printout titles as well as the header and footer. As in CDS/ISIS for DOS the first, second and third title may be specified. For each element apart from its contents the following specifications can be made:

- indentation (text can be intended to the left, intended to the right or centered),
- font type and size,
- font properties (bold, italic, underlined).

By default titles are printed on the first page only, but they may be repeated on each consecutive page of the printout. Selected elements can be also printed on the special cover page preceding the first page of the printout.

Margin parameters

The page allows for specification of the printout paper size (*A4*, *Letter* and *Legal* are the predefined paper sizes, but the user defined specification is also allowed) and for detailed specification of its margins. These are the top, bottom, inside and outside margin as well as the gutter. Values of these margins can be specified in millimeters, centimeters, inches and twip². Apart from the paper size and margins specification this page allows for specification of two parameters used by CDS/ISIS for DOS, namely the data indentation and EOC (end of column) tolerance. The third parameter is the fixed height of printed records that is specified in the same

² One twip is one TWentieth of Point and there are 1440 twips in one inch and 567 twips in one centimeter.

measurement units as margins. When this option is checked all printed records will have the same "height" what is useful when database contents is used for printing the labels.

Layout parameters

This page allows for specification of two parameters of CDS/ISIS for DOS: the multi-columnar printout and the page numbers. For both parameters WinISIS supports some specifications not allowed in the DOS version. These are column decorations (a separating line or a border) and specification of the position and alignment of the page numbers. Apart from the columns and page numbers specification this page allows for specification of the magnification of printed characters (the so called zoom) and for neglecting the specified titles and sorting parameters. A special option instructs WinISIS to just produce a hit file instead of printing (in CDS/ISIS for DOS this effect is achieved when *NULL* is specified as the print file name). Another option instructs WinISIS to print only the specified number of records on a page.

6. Database definition

One of weak points of CDS/ISIS for DOS is its low level of user friendliness. This is manifested among others during database definition. This process consists of the following four steps:

- definition of the record structure (FDT table),
- definition of the default data entry worksheet,
- definition of the default format,
- definition of the database dictionary (the default FST table).

CDS/ISIS for DOS does not offer any support to the user who designs a new database. In consequence all these steps, except the first one, require good knowledge of their specific rules and are very difficult for a person without prior knowledge of the system.

In WinISIS the database definition steps are the same, but the system facilitates the user's work during execution of each step. There is relatively little assistance available during defining the FDT table, but this step involves a lot of work when the record structure is defined conceptually and the physical definition phase does not pose many difficulties. WinISIS offers a possibility to sort record fields by their tags that is especially useful during modification of an existing FDT table, because a new field is always added as the last one in the field definition table. The next steps of the database definition process are described below in the separate subchapters.

In WinISIS definition of a new database is executed by the so called **database definition wizard**. The wizard asks first the user to enter the name of a database to be created (note that the same database name restrictions are in force as in CDS/ISIS for DOS) and then it executes one after another the consecutive steps of the database definition process. During any step except the first one the user can click the button with a green left arrow in order to return to the previous step of this process. None of the database definition files is saved until the user clicks the *Terminate* button that is available during defining the dictionary of a new database. The system warns the user that a new database will be created and executes this operation only after the user's confirmation. By default a new database is stored in the database directory specified by the system parameter 5. If the user wants to store such a base in another directory he should precede the database name (specified in the very beginning of the database definition process) by an appropriate path.

6.1. Creation of a data entry worksheet

During definition of a data entry worksheet WinISIS uses the window with two panes: the left pane shows all fields defined in the FDT table and the right pane shows the fields that belong to the worksheet being edited (in WinISIS a data entry worksheet does not define a layout of data entry fields, but simply defines a collection of fields to be filled out). During definition of a new worksheet the right pane is initially empty and the user can either place there all the fields defined in the FDT table (this action is executed by the >> button) or copy there one by one only these fields that should appear in the worksheet. In order to copy a field to the worksheet the user selects this field by clicking it in the left pane and then clicks the > button (in the left pane only the fields that do not appear in the worksheet are selectable and all other fields are dimmed). The < button executes the opposite action – it removes the selected field from the worksheet. A field is always added as the last field in the worksheet and the user can move it up by clicking the ↑ button (the ↓ button moves the selected field down in the worksheet).

By default the field name defined in the FDT table is used as the field description in a data entry worksheet. The user can change the description and other characteristics of a selected field by clicking the button with a yellow pencil and notepad. Apart from the description, default value

and help message (that are associated with a data entry field in CDS/ISIS for DOS), the worksheet editor of WinISIS enables modification of the field validation format and the pick-list that have been defined for the field being edited. The two latter elements are stored in the database validation file (see the subchapter *Data validation* in the chapter *Data entry*). Apart from these elements this file may contain the record validation format as well as the so called begin code and end code formats (see the subchapter *other functions of the validation file* in the chapter *Data entry*). All these elements can be modified by clicking the *Record validation* button available in the data entry worksheet modification window.

6.2. Creation of a format

When the user starts creation of a new format WinISIS offers him the services of the **print format assistant**. This assistant creates automatically a format outputting all fields defined in the FDT table using a format template specified by the user. The following format templates are available:

- normal,
- CDS/ISIS for DOS format,
- decorated format,
- standard HTML output,
- tabular HTML output.

All these formats output in separate lines consecutive fields of the current record in the order of these fields have been defined in the FDT table. Contents of a field is preceded by the name and tag of this field that are defined in the FDT table (tags are enclosed in parentheses). In the first three templates except consecutive occurrences of a repeatable field are output in separate lines and during a HTML output these occurrences are separated by semicolons. Moreover, in the automatically created formats that generate an HTML output the heading mode is applied, while in all other automatically created formats the proof mode is used.

As appears from its name the CDS/ISIS for DOS template does not use any formatting command that is allowed in WinISIS only and before the first field it outputs the MFN number. The normal template is very similar to the CDS/ISIS for DOS template, but it outputs all texts using the 4th color of the current color table and uses the formatting commands of WinISIS to achieve an appropriate indentation of field contents. In the decorated template field names and values are output in the Arial font using different colors and before the first field a box containing the MFN number and the database name is output in the Times New Roman font. Moreover, the field names are output in italics.

As appears from their names the HTML templates output texts in the HTML format, i.e. they output the field values with literals containing appropriate commands of the HTML language. Formats based on these templates are dedicated to generate output files that will be used to present contents of CDS/ISIS databases on Internet. Both templates output the field names in italics, but the tabular HTML template outputs contents record as a table containing two rows: one with field names and another with field values. This table is preceded by the MFN number and followed by an empty paragraph. Unfortunately, the standard HTML template does not output any HTML commands that separate contents of consecutive records and in consequence users of Internet may have problems with identification of these records in HTML pages that are generated according to this template.

A new format created by the print format assistant is automatically presented by WinISIS for manual edition by the user. This process can be facilitated by use of the *B*, *I* and *U* buttons. These buttons insert at the current cursor position the character formatting commands *B*, *I* and *UL* respectively (see in the next chapter the appropriate section of the subchapter *Using fonts*). Each of these commands is automatically preceded by the { bracket. The user should enter manually the closing } bracket in such a place that only the appropriate formatting commands are affected by the respective character formatting command. A special button inserts the following template of a hypertext link:

```
link(('description'),'action')
```

The user should replace *description* by an appropriate text and *action* by the desired hypertext command.

One of weak points of CDS/ISIS for DOS is that a format modified by a respective function of the database definition services is always saved – even if the user realizes that he made wrong modifications he cannot restore an original contents of this format. In WinISIS the modified contents of a format is saved only upon the user's request. Moreover, the *Undo* button enables the user to restore an original contents of the format and continue its modification from the beginning.

In spite of these facilities offered by the *Edit format* function available in the *Edit* menu many users may prefer editing an existing format in the database window, because they can instantly see how the database records are displayed according to the modified format contents.

6.3. Creation of a FST table

When the user starts creation of a new FST table WinISIS offers him the services of the **dictionary assistant**. This assistant enables automatic creation of a simple FST table by showing the list of all fields that have been defined in the FDT table. The user can check in this list the fields to be indexed. All these fields will be indexed using the indexing technique selected in the *Technique* box that shows the nine indexing techniques of CDS/ISIS system.

A new FST table created by the dictionary assistant is automatically presented by WinISIS for manual edition by the user. During this phase an experienced user can immediately correct the selected FST entries created by the assistant. It is recommended that an inexperienced user first creates some database records, tests the new FST by indexing these records and only then modifies contents of the FST table that defines the database dictionary. The steps:

- modification of the FST table,
- indexing of the test records,
- analysis of the dictionary contents

should be repeated until a requested database dictionary is produced.

It should be stressed that the dictionary assistant producing very simple FST tables is a useful component of the WinISIS database definition wizard, because it instantly generates an FST table defining the dictionary contents for a new database. However, the is not very probable that such a table will produce a good dictionary for a database with the complex record structure. Moreover, the dictionary assistant does not generate an FST table producing expected results when the indexing techniques 5-8 are used. These techniques are intended to generate dictionary terms with prefixes, but the formats created by the assistant do not contain the literals defining these prefixes.

7. Formatting language

Separate sections of this chapter describe various categories of new features of the CDS/ISIS formatting language that are supported by WinISIS. To enable creation of more complicated formats the limits of a format size and size of its output have been significantly increased and are as follows:

max format size	26 KB
max size of format output	64 KB

Moreover, the format creation and format modification functions are much user-friendly in WinISIS than in CDS/ISIS in DOS. The Print Format Assistant (see the section *Creation of a format* in the chapter *Database definition*) creates automatically a format according to a format template chosen by the user. The database window can be split into a top and a bottom window. If the format editor is running in the bottom window, the user can modify there the current display format and immediately apply the modified format (by clicking the button with a yellow flash of lightning) in order to see its outputs in the top window. By clicking a button with the diskette the user can save the modified contents of the format in the same or another format file.

By default WinISIS uses the full name of the current database master file as a title of the window in which this database is displayed. In version 1.4 of this default title of the window can be changed by using the TITLE command in the display format. This command has the following form:

TITLE(*format*)

and it assigns an output text of the format as a title of the current window. This command can be also used in a window that is created by a hypertext command (see below the section *Working with windows* in the subchapter *Hypertext links in formats*).

7.1. Output of selected field occurrences

In WinISIS a field selector *Vtag* of a repeatable field can be followed a specification of the field occurrences to be output. This specification has the following general form:

[*from..to*]

where *from* is the number of the first occurrence output,
to is the number of the last occurrence output.

Specification of the last occurrence is optional. If the two colons are omitted too just a single occurrence is output, otherwise all consecutive occurrences starting from the given occurrence number are output.

7.2. New functions of the formatting language

7.2.1. Numeric functions

The following numeric functions have been introduced in WinISIS:

LR(*format*,*first*,*last*)

Like the *L* function the *LR* function uses an output sting produced by the *format* as a dictionary term and returns MFN numbers of all the records retrieved (only MFN of the first record

retrieved is returned by the *L* function). Optional parameters *first* and *last* allow to limit the total number of records retrieved (these parameters represent sequential numbers of records in the set of all records retrieved).

Usually the *LR* function is used as an argument in the REF function call (see below the section *Loading format files* in the subchapter *Hypertext links in formats*)

NPST(*format*)

Uses an output string produced by the *format* as a search term and returns and returns the number of postings of this term.

NOCC(*Vtag*)

Returns the number of occurrences of the specified field.

OCC

This function should be used within a repeatable group only and it returns the number of the current occurrence in this group. It may be used to produce a numbered list of occurrences.

SIZE(*format*)

Returns the size of an output string produced by the *format*. Note that size of this output string depends on the mode (for instance, a field contents is automatically followed by the period and space in the data mode).

TAG

To be used inside a repeatable group. This function returns tags of consecutive fields present in the current record (in creation order of these fields) or 0 if all fields have been processed. The following commands can be used to output in separate lines all fields of the current record preceded by their tags:

```
(if p(v0) then f(tag,0,0),' ',v0/ fi)/
```

In WinISIS the field selector V0 identifies any field of the current record.

TYPE('pattern',*format*)

Tests whether an output string produced by the *format* corresponds to the given pattern and returns 1 if this condition is satisfied or 0 otherwise.

TYPE(*number*, *format*)

Tests whether an output string produced by the *format* satisfies a condition specified by the value of *number*. This argument be an integer number from the range 1-6 and it has the following meaning:

- 1 - the string contains only alphabetic or numeric characters;
- 2 - the string contains only alphabetic characters;
- 3 - the string contains only numeric characters;
- 4 - the string is an integer (optionally signed);
- 5 - the string is a numeric value (including a value in scientific notation).

The function returns 1 if the condition is satisfied or 0 otherwise.

7.2.2. String functions

The following string functions have been introduced in WinISIS:

DATE(*type*)

Returns the current date and/or time in the format specified by the argument *type*. This argument can have one of the following values:

- 1 - returns a date and time stamp in the form *MM-DD-YYYY HH:MM:SS*;
- 2 - returns the date in the form *MM-DD-YYYY*;
- 3 - returns the time in the form *HH:MM:SS*.

Note that the actual format of the date depends on the value of the system parameter 150 (see the subchapter *System parameters* in the next chapter).

DB

Returns the name of the current database.

SS(pos,len,format)

Returns the specified substring of an output string produced by the *format*. This substring begins at the position *pos* and is *len* characters long.

7.3. Use of external formats

In WinISIS formats may include external formats. This feature greatly simplifies creation of formats for databases with complex structures. A group of commands that are common for various formats is stored as a separate format and such a group is included into a given format by applying in it the function

@fname

where *fname* is the name of format file included.

This function is used frequently in formats of the MTS database. Because this database stores several types of records its display and print formats have been divided into sections, each outputting data of the particular record type. Within each section the *DES* format is included and majority of sections the *PUB* format is included. These formats output respectively the subject descriptors and the publisher data. Schematic structure of a display format of the MTS database is shown below.

```
. . .
if v4.1='M' or v4.1='m' then
. . .
"Sales No. "v282/"##v290(3,3)/"#d290,@des,
. . .
"c30("v350(0,30)"/@pub,
. . .
fi,
if v4.1='S' or v4.1='s' then
. . .
"- "v262" vols. ", "- "v6"/"##v290(3,3)/"#d290,@des,
. . .
@pub,"Separate ed."d310/
. . .
fi,
if v4.1='P' or v4.1='p' then
. . .
"##v290(3,3)/"#d290,@des,
. . .
if S(v41,v44,v45)>' ' then c30 fi,"Free"D41,"US$ "V44,"US$ "V45@pub,
. . .
fi,
if v4.2='AP' or v4.2='Ap' or v4.2='aP' or v4.2='ap' then
. . .
"##v290(3,3)/"#d290,@des
fi,
if v4.2='AM' or v4.2='Am' or v4.2='aM' or v4.2='am' then
. . .
"##v290(3,3)/"#d290,@des
fi,
if v4.1='W' or v4.1='w' then
. . .
"##v290(3,3)/"#d290,@des,
. . .
if S(v41,v44,v45)>' ' then c30 fi,"Free"D41,"US$ "V44,"US$ "V45@pub,
. . .
fi/
fi,
. . .
```

7.4. Referencing other databases

The *REF*, *L*, *LR* and *NPST* functions may be followed by a **database qualifier** causing that these functions refer to an alternate database different from the current one. The database qualifier has the following general form:

->database_name

where *database_name* is the name of a database being referred.

Several auxiliary databases are referenced from the MTS database. This database stores in the repeatable field 300 the language codes and reference to the LANG database storing codes and names of various languages (see Annex) is made in order to display language names instead of language codes. Display and print formats of the MTS database contain the following repeatable group:

```
(if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2) else v300 fi, d300+|;|)
```

First the *NPST* function is applied to check if the given code (current occurrence of the field 300) is present in the *LANG* database. If this condition is satisfied the *REF* function is used to output the name (field 2) of an appropriate language from this database, otherwise the language code is output.

7.5. Working with format variables

In WinISIS 10 numeric and 10 string **format variables** are predefined. These variables have the following names and initial values:

Type	Names	Initial value
numeric	e0, e1, ... , e9	0
string	s0, s1, ... , s9	"

The following **assignment operations** assign values to numeric and string variables respectively:

en:=numeric expression

sn:=(format)

Note that parentheses around the *format* are required. A numeric variable may be used in formats in any place where a numeric value is allowed, e.g. as operand of a numeric expression. A string variable may be used both as operand of a string expression and as a formatting command.

The **WHILE command** defines a loop that allow to execute some formatting commands repeatedly. This command has the following general form:

WHILE condition (format)

where *condition* is a Boolean expression,

format is a CDS/ISIS format executed as long as value of *condition* is *True*.

Formatting commands, that ensure that the *condition* will not be satisfied when the loop should be terminated, should be applied inside the *format*. If the initial value of *condition* is *False* then the *format* will not be executed at all.

In the MTS database the following commands can be used to output a space after each character of the publication title (field 20):

```
s0:=(v20), e0:=1, e1:=size(s0)+1, while e0<e1( ss(e0,1,s0),' ', e0:=e0+1 ),
```

If the title is

```
WORLD TELECOMMUNICATION DEVELOPMENT REPORT
```

it will be displayed as

```
W O R L D   T E L E C O M M U N I C A T I O N   D E V E L O P M E N T  
R E P O R T
```

CDS/ISIS offers two ways to mark in contents of a long field (such as an abstract) phrases to be taken to the inverted file. These phrases may be enclosed either in slashes or in triangular brackets. The difference between these two methods is that triangular brackets in contrast to slashes can be not output when an appropriate mode is applied in the format. To avoid displaying slashes in the field 290 where these signs are used only to mark inverted file phrases we can use the following commands:

```
e0:=1, e1:=size(v290)+1,
while e0<e1( s0:=(ss(e0,1,v290)),if s0<>'/' then s0 fi, e0:=e0+1 ),
```

In this example each character of the field 290 is examined in the *WHILE* command (*e0* shows the position of a character being examined and the *SS* function is applied to extract this character that is temporarily saved as *s0*) and if this character is different than a slash it is output. Instead of outputting single characters we can collect them in the string variable *s1* and output this variable after examining all the characters.:

```
e0:=1, e1:=size(v290)+1, s1:='',
while e0<e1( s0:=(ss(e0,1,v290)),if s0<>'/' then s1:=(s1,s0) fi, e0:=e0+1
),
s1
```

7.6. Using fonts

In WinISIS various types and colors of fonts can be used. The font of a given type is specified by the sequential position of this font in the table of all fonts defined. Similarly, the color of fonts is specified by the sequential position of this color in the table of all colors defined. These tables are called the font table and the color table respectively. Elements of these tables have sequential numbers starting from 0, also the number of the 1st element is 0, of the 2nd element 1, etc.

7.6.1. Fonts and colors definition

The **font table** is defined by the **FONTS** command. This command has the following general form:

FONTS(definition_1, definition_2, ...)

where *definition_i* is the definition of *i*-th font. Such a definition has the following general form:

(family_i,name_i)

where *family_i* is the font family the *i*-th font belongs to,
name_i is the name of the *i*-th font.

The following font families may be used:

Font family	Examples of fonts that belong to this family	Description of the font family
swiss	Arial	Swiss, proportionally spaced sans serif fonts
roman	Times New Roman	Roman, proportionally spaced serif fonts
modern	Courier New	fixed-pitch serif and sans serif fonts
script	Cursive	script fonts
decor	Old English	decorative fonts
tech	Symbol	technical, symbol, and mathematical fonts.
nil		unknown or default fonts (default)

If the *FONTS* command is present in a format it must be used at the very beginning of the format, before all other formatting commands. If this command is absent WinISIS uses the font table defined by the system parameter 110 (see the subchapter *System parameters* in the next chapter). If this parameter is also absent the default font table of the Windows system is applied.

In Windows a color is defined as mixture of the three **basic colors**: red, green and blue. Such a mixture is specified by providing for each basic color an integer number from the range 0-255. This number specifies the amount of this basic color in the color being defined and it is called an index of the basic color. The extreme indexes define the following colors:

(0 , 0 , 0)	white,
(255 , 0 , 0)	red,
(0 , 255 , 0)	green,
(0 , 0 , 255)	blue,
(255 , 255 , 255)	black.

The **color table** is defined by the COLS command. This command has the following general form:

COLS(definition_1, definition_2, ...)

where *definition_i* is the definition of *i*-th color. Such a definition has the following general form:

(r_i,g_i,b_i)

where *r_i* is the index of red in the *i*-th color,
g_i is the index of green in the *i*-th color.
b_i is the index of blue in the *i*-th color.

If the COLS command is present in a format it must immediately follow the FONTS command (it means that if the COLS command is applied the FONTS command has to be applied too). If the COLS command is absent WinISIS uses the color table defined by the system parameter 110 (see the subchapter *System parameters* in the next chapter). If this parameter is also absent the default color table of the Windows system is applied.

7.6.2. Character formatting

The following commands are used in WinISIS for to specify type, size and color of the font used to output characters as well as style of these characters.

Fn

Characters are output using the *n*-th font where *n* is the number of this font in the font table. By default the first font of this table is used (font number 0).

FSn

Characters are output using the specified font size. The parameter *n* is a number that specifies the size of the current font in half points, e.g. *fs32* means that a 16-point font will be used. The default size is 12 points (*fs24*).

CLn

Characters are output using the *n*-th color where *n* is the number of this color in the color table. By default the first color of this table is used (color number 0). In the default color table the first color is black.

B

Characters are output as bold.

I

Characters are output as italic.

UL

Characters are output as underlined.

All the character formatting commands affect texts output by all commands that follow them in a format. To ensure that character formatting commands will **affect only a group of formatting commands** the commands forming such a group should be **enclosed in the { } brackets**. The following commands

```
fs40,'Welcome to the ',  
{fs60,b,'M'},'ultilateral',{fs60,b,' T'],'rading',{fs60,b,' S'],'ystem'
```

output the text *Welcome to the Multilateral Trading System*, where the capital letters *M*, *T* and *S* are output using a 30-point font in bold face and all the remaining characters are output using a 20-point font.

7.7. Paragraph formatting

The following commands specify layout of paragraphs in output texts of formats. By default all paragraphs are adjusted to the left margin.

M (*indent_all*, *indent_1st*)

Sets the left margin for each paragraph following this command. The first parameter specifies the indentation from the left for all paragraph lines and the optional second parameter specifies an indentation applied additionally to the first line of each paragraph only³. Indentations are expressed in twips. The second parameter can have a negative value. For instance, if the command `m(400,-400)` sets 400 twips indentation for all paragraph lines except the first one. The *M* command affects all paragraphs following this command.

TAB

TAB (*value*)

Tabulates to a desired position in the line. The optional parameter specifies in twips position of the next character output (its distance from the left margin). If the parameter is missing a default tabulator position is taken. For instance, the formatting commands

```
/tab(300),v290
```

have the same effect as the commands

```
/m(0,300),v290
```

except that the left margin is set for all the following paragraphs (see below how to restrict the scope of the *M* command).

QC

Centers text between the left and the right margin. Affects all paragraphs following this command.

QJ

Justifies text between the left and the right margin. Affects all paragraphs following this command.

QR

Justifies text to the right margin. Affects all paragraphs following this command. This command replaces the hidden command⁴ `'\qr'` available in early versions of WinISIS.

BOX

BOX(*color_no*)

Draws a box around the current paragraph. The border of such a box is drawn using the current color (this color is specified by the *CLn* command, see the previous subchapter). The optional parameter specifies the background color of this box. Value of this parameter is the color number in the current table of colors (see the previous subchapter). The background color should be different from the current color, otherwise the paragraph text will not be visible.

In MTS database presence of the field 8 signals that a serial or periodical is inactive, i.e. it is not subscribed any more. The following commands

```
cl4,v210,if p(v8) then box(10),'INACTIVE' fi,c30,v220
```

³ It should be stressed that in WinISIS indentation for the first line of a paragraph is calculated as a sum of the two parameters of the *M* command. Do not confuse these parameters with the indentation parameters that in CDS/ISIS for DOS can follow a field selector (the first of these parameters specifies indentation applied for the first line only).

⁴ A hidden command is an unconditional literal that begins from the backslash sign and ends with a space. This backslash is followed by a command name that is case sensitive.

output in a box the frequency (field210) and the call number (field 220) of such a serial together with the text *INACTIVE*. Border and text of this box is output using the 4th color and the 10th color is the background color of the box.

KEEPL

Ensures that text of the current paragraph will be output in a single line regardless the size of the window or page (if the line does not fit the window or page then it will be cut). This command replaces the hidden command '`\bsknone`' available in early versions of WinISIS.

The commands *M*, *QC*, *QJ* and *QR* affect all paragraphs output by other formatting commands that follow these paragraph formatting commands. To ensure that these commands will **affect only a group of paragraphs** the commands forming such a group should be **enclosed in the { } brackets**.

7.8. Outputting pictures

Some formatting commands of WinISIS output pictures rather than texts. These pictures must be stored as files in the BMP and JPEG formats and these files must reside in the image directory specified using the system parameter 141 (see the subchapter *System parameters* in the next chapter). Both the file name and extension must be provided to specify a JPEG file, while for a BMP file its extension can be omitted.

PICT(*format1*, *format2*)

This command interprets an output string produced by the *format1* as a name of the picture file and outputs this picture. The parameter *format2* is optional. If this parameter is provided and the *PICT* command is used in a display format as description of a hypertext command (see below the section *Hypertext links in formats*) the picture file specified by *format2* replaces the picture specified by *format1* when the mouse points this picture. The *PICT* command replaces the following hidden commands of early versions of WinISIS:

```
'\chpict fname1 '  
'\chpict1 fname1 fname2 '
```

where *fname1* and *fname2* are the names of picture files in the BMP format.

BPICT(*format*)

This command interprets an output string produced by the *format* as a name of the picture file and applies this picture as a **wallpaper** in the current window. Only the 8x8 pixels square cut at left top corner of the picture is used. This form of the *BPICT* command replaces the hidden command '`\isispatt fname`' available in early versions of WinISIS.

BPICT(*format*,*style*)

This command interprets an output string produced by the *format* as a name of the picture file and applies this picture as a **background image** in the current window. By default this picture is aligned to the top edge of the window. The parameter *style* is a number that specifies the picture alignment and/or scaling. This parameter can have the following values:

- 0** left aligned (default)
- 1** tile (the picture replaces the wallpaper)
- 2** scaled (the picture fills whole window)
- 4** centered horizontally
- 8** centered vertically
- 16** right aligned
- 32** bottom aligned

Moreover, any value being a sum of these values is also allowed and it specifies a combination of the basic styles. For instance, 12 can be used to display a picture in the center of the current window. This is a combination of 4 (centered horizontally) and 8 (centered vertically). This form of the *BPICT* command replaces the following hidden commands of early versions of WinISIS:

```
'\isisbgbmp fname '
```

```
'\pictscaledN '
```

BPICT((format1),(format2),style)

This command interprets the strings produced by *format1* and *format2* as names of the picture files and applies in the current window the first picture as a wallpaper and the second picture as a background image. *Style* specifies alignment and/or scaling of the background image.

In MTS database the display format *MTS* shows just some general information on this database:

```
title('MTS database'),f2,m(2000),bpict(('winisis'),0),fs40,'Welcome to the'/
{fs60,b,'M'},'ultilateral',{fs60,b,' T'},'rading',
{fs60,b,' S'},'ystem'/'database !',fs20/#
m(0){b,'MTS'},' is a bibliographic database aimed at facilitating research ',
'on the basic documents on the Multilateral Trading System. These publications
',
'constitute the core documentation base of the Reference Centers established
',
'by ITC under the JITAP project. The MTS database contains also descriptions
',
'of other documents (monographs, serials, periodicals, articles and Internet
', 'sites) related to the subject.'/#
qc,pict('un'),' MTS database is maintained by the ',
{b,'International Trade Centre UNCTAD/WTO '},pict('wto')/
qc,'using the software package CDS/ISIS for Windows developed by UNESCO.'/#
fs30,b,i,link(('Continue'),'FORMAT short')
```

The *BPICT* command displays the WinISIS logo as a background image located in the left top edge of the window, while the *PICT* commands display the UN logo in the very beginning and the WTO logo in the very end of the paragraph containing the name of ITC.

The **TEXTBOXIMG** command is a special version of the hypertext command *TEXTBOX* (see the section *Working with windows* in the next subchapter) that creates a window in which the specified picture is shown. This picture is displayed in the center of this window and size of the picture is automatically changed whenever the window is resized. The buttons *Zoom in*, *Zoom out*, *Copy to clipboard* and *Print* are shown in such a window. The *Zoom* buttons increase and decrease magnification of the picture. The user can also select with the mouse a rectangular area of the picture (by clicking a corner of this area and dragging the mouse until a desired rectangle frame is visible) and when the mouse button is released the picture magnification is changed in such a way that this area fills the window.

7.9. Hypertext links in formats

In WinISIS **display formats may be interactive**. Such formats contain **hypertext links** that can be activated by the user. A hypertext link is defined by the *LINK* command. This command has the following general form:

LINK((description), hypertext_command)

where both parameters are the formats. An output texts of the *description* appears on the screen and when the cursor is over this text the cursor shape changes what signals that this a hypertext link. When the user clicks such a description WinISIS executes a **hypertext command** that is specified by output text of the second parameter of the *LINK* command. Text of a hypertext link is displayed underlined using the second color of the current color table (see above the section *Fonts and colors definition* in the subchapter *Using fonts*). It is also possible to display a picture (see the *PICT* command in the previous subchapter) as the description of a hypertext link.

The following sections describe various hypertext commands that can be used as arguments of the *LINK* commands. The names of these commands must be given in **upper case**. Moreover, the backslash sign and the signs { and } must be preceded in hypertext commands by the backslash sign. In consequence, these characters are coded as follows:

character is coded as
\
\\

```
{      {  
}
```

7.9.1. Navigating among records

GOTO *mfn*

This command changes the MFN number of the current record to the number specified as a parameter. For instance, the following commands

```
'Go to the ',link(('first'),'GOTO 1'),' ',  
link(('previous'),'GOTO ',if val(mfn)>1 then f(mfn-1,1,0) else '1' fi)
```

displays the hypertext links *first* and *previous* which cause that either the first or the previous record of the current database is displayed in the window.

LGOTO *term*

This command uses its argument as a search term and displays the first record retrieved in the inverted file of the current database. If the specified term was not found the first record of the current database is displayed.

LAGOTO *term*

LAGOTO/*id term*

This command uses its argument as a search term and displays the first record retrieved in the inverted file of the current database. If the specified term does not exist, the next inverted file term is used instead of the term provided. If the command name is followed by the slash and a field identifier used in the inverted file FST table, only terms taken from the specified field are retrieved.

GOBACK

This command displays the record that was shown immediately before the record being displayed currently.

7.9.2. Changing the current display format

FORMAT *format_name*

This command uses the specified format as the current display format. The parameter is interpreted as format name. For example, the default display format of the MTS database shows some general information on this database and the hypertext link *Continue* that selects the format *SHORT* as the current display format:

```
link(('Continue'),'FORMAT short')
```

RESETSEARCH FORMAT *format_name*

If the *FORMAT* command with the *RESETSEARCH* prefix is executed when the search results are displayed, this command causes additionally that the system returns to the default display mode in which all records of the current database are browsed.

7.9.3. Printing

PRINTTHIS

This command displays a small print parameter window that enables printing the current page.

PRINTSEARCH

This command displays a small print parameter window that allows to print the search results being displayed or the current record.

These print windows contain boxes that enable selection of the output device and the print format. If output to a file has been requested the name of the output file should be entered into an

appropriate box. If this parameter is missing output is directed to the file *dbase.TXT* where *dbase* is the current database name. The system rewrites an existing output file without any warning if a file with the given name already exists in the working directory.

7.9.4. Executing external programs

All commands described in this section can be used either as hypertext commands in the *LINK* commands or as functions called from the WinISIS menu (see the subchapter *System files* in the next chapter).

CMD *program file_name*

This command runs the Windows or DOS program specified by the first parameter. This parameter can provide the program name preceded by its full path. It should be stressed that long file and path names should be written as short names appearing in DOS, e.g.

```
progra~1
```

should be used instead of

```
Program Files
```

The optional parameter *file_name* specifies a name of the file to be processed by the program.

OPENFILE *file_name*

This command runs an appropriate program that will process a file specified by the command parameter. Windows will analyze the extension of this file and process the file using a program associated with files having this extension. In the MTS database field 15 stores an Internet address of a web site. The following commands

```
if p(v15) then {i,'URL: '},link((v15),'OPENFILE ',v15) fi/
```

ensure that this address is displayed a hypertext link and when the user clicks this link, the default Internet browser is called to open the specified web site. The *OPENFILE* command is available in version 1.4 of WinISIS.

CALL *program_name*

This command runs a specified WinISIS plug-in program that uses the WISIS.DLL library (see the last subchapter in the chapter *General overview*) to process a CDS/ISIS database. Such a program must reside in the directory specified by the system parameter 148 (see subchapter *System files* in the next chapter). The *CALL* command is available in version 1.4 of WinISIS.

7.9.5. Opening another database

BROWSE *database_name, mfn, format_name*

Opens the data base specified by the first parameter in a new database window and if the other parameters are missing displays the first record of this database using its default display format. The optional parameters *mfn* and *format_name* specify respectively the MFN number of the record displayed initially and a name of the display format.

ITC distributed three of its CDS/ISIS databases on the *ITC Trade Information Toolkit* CD-ROM together with a read-only version of WinISIS. In order to facilitate selection of a desired database a dummy database called ITC was created and used a default database. This database contains just a single record with a single field, because presence of this record was necessary merely to execute the default display format of the ITC database that is presented below:

```
f2,{ qc,fs40,'Selected databases of '/pict('un'),
{fs60,b,'I'},'nternational',{fs60,b,' T'},'rade',
{fs60,b,' C'},'entre UNCTAD/WTO',pict('wto')}/
fs20/{b,{fs24,link(('ITD'),'BROWSE itd')},i,fs22,' - International Trade Documentation'}/
'This is the main access catalogue to the documentation collection held by the Trade ',
'Documentation Centre of ITC. ITD is a bibliographic database aimed at facilitating ',
'research for and selection of relevant publications related to various aspects ',
'of international trade, including: export promotion and marketing, international ',
'purchasing and supply management, market and product development, trade-support ',
'services, etc.'/
```

```
{b,{fs24,link(('PMR'),'BROWSE pmr')},i,fs22,' - Published Market Reports'}/
'This is a selective bibliography aimed at providing the trade promotion ',
'institutions in developing countries and economies in transition with a reference ',
'guide of selected market reports which will enable them to improve their ',
'knowledge of, and thus gaining access to, foreign markets. The reports listed in ',
'the PMR database have been published by ITC as well as various international, ',
'regional and national organizations, including trade promotion organizations, ',
'import promotion offices and research institutions.'/
{b,{fs24,link(('TITT'),'BROWSE titt')},i,fs22,' - Thesaurus of International Trade Terms'}/
'This thesaurus provides a standardized vocabulary for the bibliographical ',
'description of publications relating to the international trade. All terms are ',
'provided in three languages: English, French, and Spanish. You may change the ',
'language during browsing the TITT database.'/
#qc,{i,'These databases are maintained using the software package '},
{b,'CDS/ISIS for Windows'},i,' developed by UNESCO.'
```

This format displays short descriptions of the *ITD*, *PMR* and *TITT* database. Each description is preceded by a title line where the database name appears as a hypertext link and when the user clicks this link the *BROWSE* command opens a desired database in a new database window.

7.9.6. Executing system actions

MESSAGE *action, database_name, format_name*

This command executes a system action specified by its first parameter in a database specified by its second parameter (this database is automatically opened if necessary). The following values of the *action* parameter are allowed:

OPENSEARCH	open the expert search window,
OPENEASYSEARCH	open the guided search window,
LANGUAGE-XX	change the current language (XX is the code of a new language).

7.9.7. Working with windows

7.9.7.1. Text windows

TEXTBOX *format*

TEXTBOXCHILD *format*

TEXTBOXRCHILD *format*

Each of these commands creates a new window and displays in it an output string generated by the format provided as a command parameter. The kind of a text window created depend on the variant of the *TEXTBOX* command used, namely:

TEXTBOX	creates an independent window that is closed only when the user clicks its <i>Close</i> button or exits the WinISIS system,
TEXTBOXCHILD	creates a database dependant window (the so called child window of the current database) that is automatically closed when the user closes the current database,
TEXTBOXRCHILD	creates a record dependant window (the so called child window of the current record) that is automatically closed when the user displays another record.

In MTS database the *SHORT* format shows only the basic fields of the database records. In particular this format does not display *Notes* that are stored in the field 200. The following formatting commands can be added to the *SHORT* format in order to show these notes in a separate window that is displayed when the user clicks the hypertext link Show notes:

```
if p(v200) then link(('Show notes'),'TEXTBOXRCHILD "v200) fi/
```

Note that because a *TEXTBOXRCHILD* command has been used, the new window is created as a child window of the current record and therefore it is automatically closed when the user displays another record.

A format applied as the parameter of a TEXTBOX command is executed together with a display format in which this command has been used. In this parameter some formatting commands are not allowed. Such a format should not contain among others the character formatting commands (fortunately, presence of commands that are not allowed do not cause any error, these commands are simply ignored). The variants of TEXTBOX commands without these limitations are described in the next section.

TEXTBOXIMG *file_name*

TEXTBOXCHILDIMG *file_name*

TEXTBOXRCHILDIMG *file_name*

As the standard *TEXTBOX* commands these commands create new windows, but these windows show pictures rather than texts. The parameters are interpreted as names of picture files located in the image directory of WinISIS. For description of these windows see above the subchapter *Outputting pictures*.

7.9.7.2. Loading format files

TEXTBOXLOAD *format_name, old_text, new_text*

TEXTBOXCHILDLOAD *format_name, old_text, new_text*

TEXTBOXRCHILDLOAD *format_name, old_text, new_text*

As the *TEXTBOX* commands described in the previous section each of these commands creates a new window and displays in it an output string generated by a format specified by its first parameter. This parameter is a format name and this format is executed (loaded) only when the user clicks an appropriate hypertext link. There are no limitations on formatting commands used in formats specified as parameters of *TEXTBOXLOAD* commands. Moreover, an output text of such a format may occupy the whole display work area whereas output of a format used as the parameter of a *TEXTBOX* command shares this area with output of the display format in which this command has been used. In consequence, complex formats are without problems loaded into text windows, while application a complex format as the parameter of a *TEXTBOX* command used in a complex display format may overload the display work area.

The parameters *old_text* and *new_text* are optional. If these parameters are used, contents of the format specified by the first parameter is modified before its execution in such a way that each occurrence of the character string *old_text* is replaced by the character string *new_text*. In consequence, contents of a format that is actually executed depend on these parameters. In WinISIS display formats that can be modified before their execution are called **polymorphic formats** (see the last section of this subchapter).

In the MTS database the repeatable field 292 stores the subject descriptors. This is a simplified version of the *DES* format that is included into the *SHORT* format of the MTS database in order to display these descriptors:

```
if p(v292) then {i,'Subject descriptors: '} fi,
( if p(v292) then
  if npst->titt(|E=|v292)>0 then
    {b,link((v292),'TEXTBOXCHILDLOAD d1,xxx,'v292)},' '
  else
    v292
  fi,
fi, d292+|; | )/
```

The TITT database stores the *Thesaurus of International Trade Terms* and English terms of this thesaurus are stored in the inverted file of the TITT database with the prefixes *E=*. If the current subject descriptor (occurrence of field 292) is an English term of this thesaurus (in the TITT database the number of postings of the term *|E=|v292* is greater than 0) this descriptor is displayed a hypertext link, otherwise it is displayed as normal text.

The following command is used to display the current subject descriptor as a hypertext link:

```
link((v292),'TEXTBOXCHILDLOAD d1,xxx,'v292)
```

As you can see the descriptor is used both as a hypertext link and as the third parameter of the *TEXTBOXCHILDLOAD* command. If the user clicks this hypertext link an output text of the *D1*

format is displayed in a new window, but before execution of this format in its contents each occurrence of the character string xxx is replaced by the current subject descriptor. The *D1* format has the following contents:

```
f2, title('List of publications in the MTS database'),
if npst('DES=xxx')>1 then
  {b,fs30,qc,'Publications on "xxx"}/m(300,-300),
  ref(lr(('DES=xxx')),/link((v20),'GOTO 'mfn))
else
  {b,fs30,qc,'No other publications on "xxx"'}
fi
```

(in the MTS database subject descriptors are indexed with prefixes *DES=*). In consequence, if the user clicks a hypertext link with a subject descriptor then contents of the new window depend on the postings to this descriptor in the inverted file. If the current descriptor appears there more than once, this window contains titles (*Title* is stored as field 20) of all publications that are indexed with this descriptor, otherwise the window contains just text:

No other publications on "*Descriptor*"

To show all publications the *LR* function is applied as the first parameter of the *REF* function. The following format is used as the second parameter of the *REF* function:

```
/link((v20),'GOTO 'mfn)
```

Title of each publication is displayed as a hypertext link and if the user clicks such a link the *GOTO* command is executed in order to display in the database window description of this publication (the record with an appropriate *MFN* number).

In consequence, in the database window all subject descriptors that are terms of the thesaurus are clickable. If the user clicks such a descriptor and if there are other records indexed with this descriptor, a new text window containing titles of all these records is displayed. These titles are clickable too and when the user clicks a title in the text window, contents of an appropriate record is displayed in the database window. This effect can be achieved because the text window is a child window of the current database (the text window was created by the *TEXTBOXCHILD* command).

7.9.7.3. View windows

VIEW *database_name, mfn, format_name*

Creates a new view window, opens in it the database specified by the first parameter and displays its record specified by the second parameter using a display format specified by the third parameter.

A **view window** is similar to the database window because such a window uses the current display format to show a single record from the database open in this window. However, a view window does not contain the buttons and boxes of the database window (*MFN* box, navigator buttons, display format box and dictionary button). Moreover, if a view window is the active window the search and data entry operations cannot be executed from the menu or using the toolbar buttons. In order to increase functionality of the view window appropriate hypertext commands may be used in the display format applied in this window (see in the next chapter the section *Use of view mode* in the subchapter *Access restrictions*).

VIEWL *database_name, format_name, (dictionary_term), occurrence_no*

As the *VIEW* command this command creates a new view window, opens in it the database specified by the first parameter and displays its record specified by the third parameter using a display format specified by the second parameter. However, the record shown initially in this view window is not specified by its *MFN* number. The third parameter of the *VIEWL* command is interpreted as a dictionary term and if the optional parameter *occurrence_no* is omitted, the view window displays the first record retrieved in the database open in this window. The fourth parameter represents a sequential number of the record displayed in the set of all records retrieved.

In the MTS database there are two ways of entering publisher data:

- publisher name, address and contact information are stored in a record of the MTS database,
- a record of the MTS database contains merely the publisher code (field 52) and then detailed information on this publisher is read from the auxiliary database PUB (in the inverted file of this database publisher identifiers are stored with the prefixes *PCOD=*).

In the *FULL* format of the MTS database the following format *PUBL* is included to display the publisher data:

```

if p(v52) then {i,'Publisher : '},
  if npst->pub("PCOD="v52)>0 then
    ref->pub( l->pub("PCOD="v52),v50+|, |,{m(1200,-100),v90/v100} ),
    ' ( ',link('Publisher details'),
      'VIEWL[1] pub,source,(',"PCOD="v52,')'),' )'/
  else
    {b,i,c11,'Invalid publisher identifier: '},">>> "v52" <<<"
  fi
else
  if p(v50) then
    {i,'Publisher: '},tab(1100),v50/
    {m(1200,-100),v61/v62/v63/v90/v100/"Fax "v102/"Tel."v104/}
  fi
fi/

```

When the user clicks the hypertext link *Publisher details*, the *VIEWL* command is executed in order to open the PUB database in a view window and to display according to the *SOURCE* format the first record retrieved in this database by the search term *PCOD=Publisher_code*. The number 1 enclosed in the square brackets, that immediately follows the *VIEWL* command, is the window identifier. See the next section for more information how the windows are identified.

7.9.7.4. Identifying windows

By default any *TEXTBOX*, *VIEW* or *VIEWL* command opens a new window. Too many windows visible on the screen can confuse the user and therefore WinISIS offers a mechanism to identify the window to be used by such a command. The system checks whether a window with the specified identifier is already open and if this condition is satisfied it simply replaces contents of this window.

The **window identifier** has the following general form

[n]

where *n* is an integer number from the range 1-20. The identifier must immediately follow the *TEXTBOX*, *VIEW* or *VIEWL* command. These are general forms of the hypertext commands using the window identifiers:

```

TEXTBOX[n] format
TEXTBOX[n]CHILD format
TEXTBOX[n]RCHILD format
TEXTBOX[n]IMG file_name
TEXTBOX[n]CHILDIMG file_name
TEXTBOX[n]RCHILDIMG file_name
TEXTBOX[n]LOAD format_name, old_text, new_text
TEXTBOX[n]CHILDLOAD format_name, old_text, new_text
TEXTBOX[n]RCHILDLOAD format_name, old_text, new_text
VIEW[n] database_name, mfn, format_name
VIEWL[n] database_name, format_name, (dictionary_term), occurrence_no

```

This is the actual contents of the *DES* format used in the MTS database (a simplified version of this format was presented above in the section *Loading format files*):

```

if p(v292) then {i,'Subject descriptors: '} fi,
( if p(v292) then
  if npst->titt(|E|=v292)>0 then
    {b,link((v292),'TEXTBOX[1]CHILDLOAD d1,xxx,'v292)},' ',
  else

```

```

if npst->titt(|F|=|v292)>0 then
  {b,link((v292),'TEXTBOX[1]CHILDLLOAD d1,xxx,'v292)},' ',
else
  if npst->titt(|S|=|v292)>0 then
    {b,link((v292),'TEXTBOX[1]CHILDLLOAD d1,xxx,'v292)},' ',
  else
    v292
  fi,
fi,
fi,
fi,
fi, d292+|; | )/

```

In the MTS database the subject descriptors are entered as occurrences the repeatable field 292. The TITT database stores the *Thesaurus of International Trade Terms* and its dictionary contains the English, French and Spanish terms of this thesaurus with the prefixes *E=*, *F=* or *S=* respectively. If the current descriptor is a valid thesaurus term in any of these languages, this descriptor is displayed as a hypertext link. When the user clicks this link the TEXTBOX[1]CHILDLLOAD command is executed to display titles of all publications indexed with this descriptor (this is a result of the display format *D1*) in the text window number 1.

7.9.8. Polymorphic formats

Whenever a format name is provided as the parameter of a hypertext command two other parameters - *old_text* and *new_text* - can be optionally used in order to modify contents of this format before its execution. The system will replace in the format contents each occurrence of the *old_text* with the *new_text*. In consequence, if the same format is executed several times with various values of the *new_text* parameter and the same value of the *old_text* parameter, each time the format with another contents is actually executed. This why such a format is called a polymorphic format.

These are general forms of the hypertext command using the polymorphic formats:

```

FORMAT format_name,old_text,new_text
TEXTBOX[n]LOAD format_name, old_text, new_text
TEXTBOX[n]CHILDLLOAD format_name, old_text, new_text
TEXTBOX[n]RCHILDLLOAD format_name, old_text, new_text
VIEW[n] database_name, mfn, format_name, old_text, new_text
VIEWL[n] database_name, format_name, (dictionary_term), occurrence_no, old_text, new_text

```

The example of a polymorphic format is the *D1* format described above in the section *Loading format files*. Another example of such a format is the *TITLE* format described in the next section.

7.10. Interaction with the user

7.10.1. Prompting for the user's input

In order to collect a short text input by the user from the keyboard before execution of a hypertext command, such a command should be preceded by the **PROMPT prefix**. When the

user clicks an appropriate hypertext link the system first displays a little box beside this link. The user enters an input text there and signals the end of this text by striking the *Enter* key. WinISIS replaces then within the hypertext command any occurrence of the character string \$\$ with the text input by the user and finally executes this command.

Generally, the *PROMPT* prefix can precede any hypertext command. For instance, the command:

```
link(('Change format'),'PROMPT FORMAT $$')
```

can be applied to replace the current input format with a display format specified by the user. However, this command will be correct only when the user enters the name of a display format. If the user's input does not represent such a name, WinISIS displays an error message and applies the built-in format **ALL** that shows all fields of the record.

Frequently the *PROMPT* prefix precedes a hypertext command using a polymorphic format and then the user's input is applied as the command parameter modifying contents of such a format. In the display formats of the MTS database the hypertext command

```
link(('Look for'),'PROMPT TEXTBOX[1]CHILDLLOAD title,xxx,$$')
```

is applied to show in a separate text window the titles of all publications containing in their titles a word entered by the user. Because the character string \$\$ is replaced by a text entered by the user, the system executes actually the following hypertext command:

```
TEXTBOX[1]CHILDLLOAD title,xxx,user_input
```

This command displays in the text window no 1 output string generated by the polymorphic format *TITLE* that has the following contents:

```
f2,title('List of publications in the MTS database'),
if npst('TIT:xxx')>0 then
  {b,fs30,qc,'Publications containing "xxx" in a title'}/m(300,-300),
  ref(lr(('TIT:xxx')))/link((v20),'GOTO 'mfn))
else
  {b,fs30,qc,'No publication contains "xxx" in a title'}
fi
```

In consequence, a content of the window depends on presence of the word entered by the user in the inverted file (in the MTS database words taken from titles are indexed with prefixes *TIT*:). If this word appears there, the window displays titles (*Title* is stored as field 20) of all publications containing the specified word in their titles, otherwise the window contains just text:

No publication contains "*word*" in a title

These titles are displayed as hypertext links and when the user clicks such a link in the text window, contents of an appropriate record is displayed in the database window. This effect can be achieved because the text window is a child window of the current database (the text window was created by the *TEXTBOXCHILD* command).

The *PROMPT* prefixes precede in the display formats of the MTS database the following *VIEWL* commands:

```
'Browse thesaurus in ',
link(('English'),'PROMPT VIEWL[1] titt,titt,(E=$$),1'),' ',
link(('French'),'PROMPT VIEWL[1] titt,franc,(F=$$),1'),' ',
link(('Spanish'),'PROMPT VIEWL[1] titt,espagn,(S=$$),1')
```

These commands open in a window the thesaurus database (*TITT*) in order to show its record describing a term entered by the user (English, French and Spanish terms of the thesaurus are stored in the dictionary of the *TITT* database with the prefixes *E=*, *F=* and *S=* respectively). If the user's input is not a valid thesaurus term in the selected language WinISIS displays in the view window the first record of the *TITT* database. Because this database is opened in a view window the display formats applied (*TITT*, *FRANC* and *ESPAGN*) contain hypertext links that enable the user to navigate among records of the *TITT* database. These display formats have the same structure, but they show the thesaurus terms in English, French and Spanish respectively (see annex for detailed description of the display formats of the *TITT* database).

7.10.2. Using dialog windows

In version 1.4 of WinISIS the *EDIT* function of the CDS/ISIS Pascal language provides another way to collect input from the user. In this system only the format exits of the CDS/ISIS Pascal language are supported and this is why only selected functions and procedures of this language can be used in WinISIS (see the reference manual for the list of these library routines)

Like in CDS/ISIS for DOS the *EDIT* function of WinISIS performs screen edition of the specified character string, but in WinISIS a small dialog window is used for this purpose and a string being edited can consist of several lines. Such a dialog window contains two lines of text and the edit box as well as the *Cancel* and *OK* buttons. The *EDIT* function has the same parameters as in CDS/ISIS for DOS, but in WinISIS some of these parameters are not used and others have different meanings. In CDS/ISIS Pascal programs used as format exits in WinISIS a call of the *EDIT* function has the following general form:

ec := EDIT (STRING, x1, x2, x3, x4, x5, Title)

Where *Ec* is a numeric exit code returned by the function,

STRING is a string variable being edited,
x1, x2, x3, x4 are numeric variables that are not used,
x5 this numeric variable can have the following values:
 1 – the edit box can contain a single line of text and then striking the *Enter* key activates the *OK* button,
 2 – the edit box can contain several lines of text and then striking the *Enter* key begins a new line.
Title is a string variable containing a title of the dialog window and texts shown above the edit box; these elements are separated by the % sign, so this variable has the following general form:
title%first_text%second_text

In the next chapter the section *Restricted access to records or fields* discusses use of the CHECK program for providing access to a specified display format for the authorized users only. This program contains the following EDIT function:

```
ans:=edit(line,0,0,0,0,1,
'Password dialog%Please enter your password%and click "OK" button');
```

and it returns the character string *OK* only if the user entered a correct password in the edit box belonging to the dialog window entitled *Password dialog*. Because this window should appear only when the user requests access to classified information, the CHECK program is not invoked from the current display format (a format exit invoked from the current display format is executed whenever the format is executed, for instance any time the user clicks a navigator button in the database window), but from a display format executed only upon the user's request. The current display format contains the command

```
link(('Display details'),'FORMAT pass')
```

that applies the *PASS* format as the current display format. The CHECK program is invoked from the *PASS* format and this format is executed only when the user clicks the hypertext link *Display details*.

8. System management

8.1. System files

WinISIS uses the same categories of system files as CDS/ISIS for DOS. These are:

- system menus,
- system messages,
- system tables.

The most important difference is that a system menu of CDS/ISIS for DOS lists functions belonging to a given group (such as data entry or retrieval), whereas a system menu of WinISIS contains several groups of functions (positions of the menu). From the user's point of view a menu of CDS/ISIS for DOS corresponds to a position in a menu of WinISIS and the whole set of menus in CDS/ISIS for DOS corresponds to a menu in WinISIS.

In CDS/ISIS for DOS a system function executed from a menu option is identified by both the menu name and a letter serving as the menu option identifier. These elements cannot be changed and this is why external and internal identifiers of menu options are used in CDS/ISIS for DOS (external identifiers are visible in a menu and these can be translated, whereas internal identifiers are not visible and cannot be translated, because they are used for identification of menu options). Since the menu names cannot be changed (except for the language code that is used to distinguish various language version of the same menu), in CDS/ISIS for DOS separate sets of menus cannot reside in the menu directory. If the database administrator wants to define special sets of menus for various categories of users he must locate each set in a separate directory and ensure that each category of users applies its own path to the menu files. Moreover, because the system tables reside in the same directory as system menus, these tables must be copied into each directory containing system menus.

Because a menu in WinISIS corresponds to the whole set of menus in CDS/ISIS for DOS, in WinISIS the administrator simply defines separate menus for various categories of users. Each menu has a unique name and this is why all these menus reside in the same menu directory. A given menu contains only those functions of WinISIS that are available for a specified category of users and therefore such a menu is called a **profile**. The administrator can define a profile applied at the beginning of each WinISIS session using the parameter 102 (see the next subchapter) in the system parameter file SYSPAR.PAR. A special function of WinISIS lists all profiles that have been defined and allows to change the current profile during the session.

In CDS/ISIS for DOS system menus are stored in the special system worksheets, whereas the system messages are stored in special databases. In WinISIS both system menus (profiles) and system messages are stored in the special databases. These databases must have special names. The **profile database** name has the general form:

MNxxmm

whereas the **message database** name has the general form:

MSxx

Where *xx* is a unique two-letter code identifying the language version,

mm is a unique two-letter code identifying the profile.

System menus and messages of WinISIS are distributed in three **language versions**. These versions are identified by the codes EN (English), FR (French) and SP (Spanish). UNESCO distributes WinISIS with two profiles. These are:

- full version of the system (profile code DF),

- search only version of the system (profile code SH).

In WinISIS a system function executed from a menu option is identified by its unique code. A record of the system menu database represents a menu position (group of system functions). A field of such a record represents a system function and it has the following general form:

description[code]

Where *description* is a textual description of the function that is visible in the menu, in this text the & sign precedes a letter used to call this function from the keyboard (such a letter appears underlined in on the screen),
code is the unique code identifying the system function.

A record of the system menu database may contain up to 20 fields. The first field describes a **position of the WinISIS menu** and the consecutive fields of this record describe consecutive **menu options** of this menu positions. The dash sign (-) is used as a special code identifying the line that separates groups of functions available under a single menu position (no description is used for such a field). These are the fields of a menu database record that represents the *Search* option of the standard WinISIS menu in English:

```
1:    &Search[MENU-SR]
2:    &Expert Search[10113]
3:    &Guided Search[10118]
4:    &Close Search Window[10114]
5:    [-]
6:    Sa&ve search[10501]
7:    &Recall saved search[10504]
```

The last few records of the default system menu of WinISIS represent separate **system pop-up menus** rather than positions of the main menu of the system. The following record of a system menu database represents the English version of the pop-up menu of the data entry window:

```
1:    Data Entry Pop Up[MENU-REC]
2:    Open dictionary[11012]
3:    Create a new record[11002]
4:    Create a copy[11020]
5:    Reload record from disk[11009]
6:    [-]
7:    Delete record[11014]
8:    Undelete record[11001]
9:    [-]
10:   Define Default values (Model)[11026]
11:   Clear Default values (Model)[11027]
12:   Clear Entries[11025]
13:   [-]
14:   Help on Data entry[11013]
15:   Choose font[11018]
```

The **first record of a system menu database** has the special meaning. The first field of this record stores the description of the profile (this description is displayed in the list of available profiles) rather than a description of the menu position and the second field identifies the database as a system menu database. All remaining fields describe positions of the menu that the Windows system displays when the user clicks the WinISIS icon using the right button of the mouse. These are the fields of the first record in the full version profile of WinISIS in English:

```
1:    Full Version[NAME]
2:    [SYSTEMMENU]
3:    Restore[64]
4:    &Move[8]
5:    &Size[16]
6:    Mi&nimize[4]
```

```

7:   Ma&ximize[2]
8:   [-]
9:   &Close Alt-F4[128]

```

Preservation of an original WinISIS function code is vital if the menu option is applied to call a WinISIS system function. Only these functions can be called in versions 1.1-1.3 of WinISIS. In version 1.4 a system menu option can be used to **call an external program** as well. The code of such a menu option has one of the following forms:

```

CMD program_name
CALL program_name
OPENFILE file_name

```

In the two first forms *program_name* represents a name of the program to be called. When the *CMD* command has been applied the name identifies a general purpose Windows program such as Notepad. If the *CALL* command has been used the name identifies a WinISIS plug-in program that uses the WISIS.DLL library to process a CDS/ISIS database (such a program must reside in the directory specified by the system parameter 148, see the next subchapter). In the third form *file_name* represents a name of the file to be processed. The Windows system will analyze the extension of this file and call a program associated with files having this extension to process this file. For instance, the following menu option can be added to the default *Help* menu of WinISIS:

```

Open UNESCO &web page[OPENFILE http://www.unesco.org/webworld/isis/isis.htm]

```

When the user chooses this option the Windows system calls the default Internet browser to open the CDS/ISIS web page of UNESCO. For more details on *CMD*, *CALL* and *OPENFILE* commands of the formatting language of WinISIS see in the previous chapter the section *Executing external programs* in the subchapter *Hypertext formats*.

8.2. System parameters

WinISIS uses many of system parameters of CDS/ISIS for DOS, namely the parameter 0 (parameter file redirection), 1-5 (paths to the files of various categories), 6 (default database), 8 (separator of repeatable fields) and 14 (database access mode). The parameter 7 (default language code) has been replaced in WinISIS by the parameter 101 and all other system parameters of CDS/ISIS for DOS (9-13) are not supported in WinISIS. All remaining system parameters of WinISIS have not been used by CDS/ISIS for DOS. Within this section all the system parameters have been grouped according to areas of their application. In WinISIS a line starting with the semicolon is treated in the SYSPAR.PAR file as a comment.

Because WinISIS uses the significantly greater number of system parameters than CDS/ISIS for DOS, the *Configure* menu of WinISIS includes the **System settings** function that shows in a multi-page window the current settings for many of the system parameters and enables easy changing of their values. The *System settings* window contain the following pages:

- **System**
This page enables to define location of the system files of various categories and to specify some of database management parameters as well as the default language and menu (profile).
- **Options**
This page enables to specify the search and search result display parameters, some of the parameters defining layout of the main window and the parameters defining the character conversion tables.
- **Display**
This page enables among others to specify the parameters that define fonts used by the system for various purposes.

The *System settings* function changes values of many system parameters during the session of WinISIS. Apart from this function the *Configure* menu includes the **Reload configuration**

function that reads and immediately applies the system parameters from the SYSPAR.PAR file. This enables the system administrator to check directly results of the system parameters modifications done manually during the current WinISIS session. The system automatically sets values of various system parameters at the very end of each session. The same action is done when the user executes the **Save configuration** function from the *Configure* menu.

8.2.1. Access paths

This group of parameters defines location of files belonging to various categories. Value of each parameter is a path. Because WinISIS concatenates paths and file names in order to obtain full file names, the last character of each path must be \ .

1=program path

Defines location of the CDS/ISIS Pascal program files.

2=menu path

Defines location of the system menu files and system tables (files ISISUC.TAB and ISISAC.TAB).

3=message path

Defines location of the system message files.

4=work file path

Defines location of work files. These are files created by the system, e.g. sort files (HIT file and sort work files) or link files produced during inverted file generation, and files created by the user.

5=database path

Defines location of database files. For each database these are constituent files of the master file (MST, XRF), constituent files of the inverted file (CNT, N01, N02, L01, L02, IFP) and various definition files (FDT, FST, FMT, PFT, STW, ANY).

Location of constituent files of a given database may be separately defined in the special parameter file named **db_name.PAR**, where *db_name* is the database name. This parameter file must reside in the database directory. A similar convention as in the SYSPAR.PAR file has been employed. Numbers identify the parameters that define location of particular files as follows:

- 1 XRF
- 2 MST
- 3 CNT
- 4 N01
- 5 N02
- 6 L01
- 7 L02
- 8 IFP
- 9 ANY
- 10 FDT, FST, FMT, PFT, STW, SRT

The *db_name.PAR* file may contain the parameter 0 which defines mode of access to the given database. Value of this parameter is interpreted in the same way as value of the parameter 14 of SYSPAR.PAR (see the next section).

141=image path

Specifies location of picture files (graphical files in BMP and JPG format).

148=plug-in path

In version 1.4 this parameter specifies location of external programs applied as plug-ins of WinISIS.

8.2.2. Database management

6=default database name(s)

In WinISIS several databases to be opened at beginning of each session may be specified by providing their names separated by commas. The database name can be preceded by its full path. If the path is omitted the system uses a database path specified by the parameter 5 to build the complete database name.

14=database access mode

Defines mode of access to all databases. This parameter can have one of the following values:

- 0 single user mode (default value)
- 1 full multi-access mode
- 2 restricted multi-access mode

Multi-access modes can be applied in a network environment only. A standard mode defined by this parameter may be changed for a particular database by the parameter 0 of the *db_name.PAR* file (see parameter 5 above).

137=close database confirmation

In version 1.4 this parameter specifies whether user's confirmation is required for closing a database. This parameter can have the following values:

- 0 no confirmation,
- 1 confirmation required (default).

1001=database_name_1

1002=database_name_2

...

These parameters specify names of databases opened recently. These names appear as commands in the *Database* menu. WinISIS sets values of these parameters automatically. Usually these parameters are set manually only when the parameter 125 (see the next section) is set to 1. In this case value of the parameter 100*n* has the following general form:

100*n*=*nth_database_name**** *nth_database_description*

The database description is optional and it is presented instead of its name to the user when the database access restriction is in force (***) serve merely as a separator and therefore this character string should not be entered if the description is omitted). The description length cannot not exceed 80 characters.

8.2.3. Access restrictions

Parameters of this group are used to introduce restrictions in accessing standard functions of WinISIS or its databases (see below the subchapter *Access restrictions*).

102=default menu

Specifies the menu (profile) applied initially by WinISIS. The parameter value is a code of this profile. The default value is DF (code of the standard menu of WinISIS, see above the subchapter *System files*).

124= mode, position and size of the database window

Specifies the opening mode of the default databases. The parameter value has the following general form:

mode, left, top, right, bottom

Where *mode* is a numeric code representing the mode,

left, top, right, bottom these optional parameters define in pixels the initial position and size of this window.

The *mode* can have the following values:

- 0 databases are opened in the standard database windows (default),
- 1 databases are opened in view windows.

125= database access restriction

Specifies whether the user can open any database or he must choose one of the databases from a list. This parameter can have the following values:

- 0 any database can be opened (default),
- 1 only a database from the list databases of used recently (this list defined is defined by parameters 1001, 1002 etc.) can be opened. If the parameters above 1000 specify database descriptions these descriptions will be presented instead of database names to the user who wants to open a database.

8.2.4. Data entry

8=repeatable fields separator

Defines a character used to separate occurrences of repeatable fields during data entry. Note that this character becomes a reserved one and therefore it cannot be used as a data character. While in CDS/ISIS for DOS the repeatable field separator must be a single character, in WinISIS it can be a character string.

123=show empty fields during data entry

Specifies whether empty fields are shown or not during data entry. This parameter can have the following values:

- 0 only fields containing data are visible (default),
- 1 all fields of the current data entry worksheet, including empty one, are displayed.

The parameter specifies the initial setting. The user may click a toggle button visible in the very bottom of the data entry window in order to make empty fields visible or not.

130=automatic inverted file update

Specifies whether inverted file will be updated automatically when the user closes the data entry window. This parameter can have the following values:

- 0 no update (default),
- 1 automatic update.

134=width of field descriptions in the data entry window

In version 1.4 this parameter specifies in pixels the width of field descriptions in the data entry window.

150=date format

Specifies format of date returned by the DATE(*n*) function of formatting language or the DATESTAMP function of the CDS/ISIS Pascal language. By default date in the *Month-Day-Year* format is returned. The parameter specifies another format and it can have the following values:

- YMD* to set the format to *Year-Month-Day*
- DMY* to set the format to *Day-Month-Year*

WinISIS from its version 1.3 always uses 4 digits for the year.

155=checking for repeated subfields

In version 1.4 this parameter specifies whether a subfield can appear more than once within a field. This parameter can have the following values:

- 0 repeated subfields are allowed (default),
- 1 repeated subfields are treated as errors.

156=record validation mode

In version 1.4 this parameter specifies whether record validation is performed for all records or records modified only. This parameter can have the following values:

- 0 only contents of modified records is validated (default),
- 1 record validation is performed for each record.

8.2.5. Searching and displaying search results

103=search details are displayed

Specifies whether search details are shown in a separate window when WinISIS executes a query. This parameter can have the following values:

- 0 no display (default),
- 1 search details (number of postings for each term used in the search expression) are shown in the so called results window.

104=dictionary display mode

The parameter specifies the mode of dictionary display. This parameter can have the following values:

- 0 irrespective of the field selected always the complete dictionary is shown, but terms not occurring in the selected field are dimmed and cannot be selected (default),
- 1 only terms taken from the selected field are displayed.

105=automatic display of search results

Specifies whether WinISIS automatically displays search results after execution of a query from the expert search interface. This parameter can have the following values:

- 0 results are displayed upon user's request (default),
- 1 results are displayed automatically.

138=dictionary dump allowed

Specifies whether the dictionary dump button is visible in the bottom of the dictionary window. This parameter can have the following values:

- 0 the dictionary dump button is not visible,
- 1 the dictionary dump button is visible (default).

142=highlighting retrieved terms

If this parameter is present WinISIS highlights the words belonging to the retrieved terms when the search results are displayed (see the section *Highlighting of retrieved terms* in the chapter *Retrieval*). The parameter value has the following general format:

FC,BC

where *FC* and *BC* are numbers identifying the foreground and background color respectively in the current table of colors. Specification of the background color is optional (the first color of the current table of colors is used by default).

135=alternate display of search results

By default the search results are displayed in the same way as records of the current database and WinISIS identifies search results display merely by showing the search number and the sequential position of the retrieved record being currently displayed in the very bottom of the database window. If the user finds this mode of display confusing he may apply this parameter to precede output of the current display format by a box containing the same information as shown in the bottom of the window plus a hypertext link that enables quick termination of search results browsing. The parameter value is an integer number from 1 to 20. This number represents the color used to display texts in the box shown before the record contents.

980=tabular display of search results

If this parameter has value 1 WinISIS displays search results in a table (see in the chapter *Retrieval* the section *Tabular mode* in the subchapter *Displaying search results*).

144=mode of display when only one record is found

If the tabular mode of displaying search results is applied, WinISIS displays results in a table whenever more than 1 record have been retrieved and uses the standard mode for displaying search results consisting of a single record. If the parameter has value 1 the search results are always displayed in a table irrespective of the number of records retrieved.

981=name of the Search tips format

The parameter specifies the common name of a special format defined for various databases. If tabular mode of displaying search results is applied and the specified format has been defined for the current database, the *Search tips* button is available in the search windows. This button opens a window with output text of this format (see in the chapter *Retrieval* the section *Explanation of query formulation rules* in the subchapter *Displaying search results*).

8.2.6. Printing

112=printer setup

The parameter value stores the name and port of the printer applied recently. This parameter is set automatically by WinISIS at the end of each session.

113=index separator

This parameter specifies the character string used in printouts as a separator of index entries. The default value is the string consisting of a comma followed by a space.

8.2.7. Layout of main window and database window

120=toolbar is visible

Specifies whether the toolbar, containing buttons that enable fast access to the most frequently used functions of WinISIS, is visible in the main window just below the menu. This parameter can have the following values:

- 0 toolbar is not displayed,
- 1 the toolbar is visible (default).

121=toolbar help is displayed

Specifies whether the status bar is visible in the very bottom of the main window. When the cursor is positioned over a button of the toolbar, WinISIS displays in the status bar a short text explaining the function of this button. This parameter can have the following values:

- 0 explanations are not displayed,
- 1 explanations are visible (default).

122=status and position of the main window

The parameter value stores the status and position of the main window. This parameter is set automatically by WinISIS at the end of each session.

136=character magnification

Specifies the initial degree of character magnification applied in the database windows. The parameter value may vary from 80 to 160 (default is 120). The user may change this degree during session using the *Zoom in* and *Zoom out* commands from the *Browse* menu.

140=contents of logically deleted records is displayed

Specifies whether contents of logically deleted records are displayed in the database window. This parameter can have the following values:

- 0 a message is displayed instead of contents of such a record,
- 1 contents of all records are displayed irrespective of their status (default).

902=hypertext debugging

Specifies whether the actual text of each hypertext command is displayed in a separate window before execution of such a command. This parameter can have the following values:

- 0 no debugging of hypertext commands (default),
- 1 text of each hypertext command is displayed in a separate window.

Usually this parameter is applied only temporarily during creation of complex formats containing hypertext commands, especially in display formats that use polymorphic formats (see in the chapter *Formatting language* the section *Polymorphic formats* in the subchapter *Hypertext links in formats*).

8.2.8. Font definitions

All parameters of this group, except the first one, specify fonts applied for various purposes by WinISIS. Each parameter value is a font definition. This definition has the following general form:

font_name, font_size

Where *font_name* is the name of a font known to Windows,

font_size is the font size in half points

Specification of the font size is optional.

110=formatting defaults

Specifies the font table and optionally the color table as well as the default font and color used by WinISIS. Value of this parameter is a format that may only contain the *fonts*, *cols*, *cl*, *f* and *fs* commands (in this sequence). The default value of this parameter is:

```
fonts((nil,Courier New), (swiss,Arial)), cols((0,0,0)), cl0, f0, fs24
```

Note that this parameter must be written in a single line.

109=field edition font

Specifies the font used in the field edit box of the data entry window. WinISIS sets this parameter automatically at the end of each session.

111=format edition font

Specifies the font used in the format edit box of the database window. WinISIS sets this parameter automatically at the end of each session.

115=small font

In version 1.4 this parameter specifies the small font of WinISIS used among others for the dictionary display. This parameter should be set using the *System settings* window.

116=dialog font

In version 1.4 this parameter specifies the dialog font of WinISIS used for displaying its standard messages. This parameter should be set using the *System settings* window.

117=system font

In version 1.4 this parameter specifies the system font of WinISIS used for displaying system texts, such as group descriptions in various windows. This parameter should be set using the *System settings* window. Usually the system font is bigger than the dialog font.

133=system font of Windows

Early versions of WinISIS used a system font of Windows instead of the small, dialog and system fonts defined now by the parameters 115-117. In versions 1.1-1.3 this parameter can have the following values:

- 0 small system font (default),
- 1 large system font.

8.2.9. Language dependent parameters

101=default language code

Specifies a code of the language of conversation used initially by WinISIS. The system automatically sets this parameter to a code of the current language at the end of each session. The default value is *EN* (code of English, see the previous subchapter).

106=DOS to Windows conversion table

Specifies the name of a *DOS to Windows* character conversion table. This table is used by WinISIS to convert characters from DOS codes to Windows codes during reading database records (see below the subchapter *Characters conversions*). When this parameter is absent the standard conversion table of the Windows system is used. If this parameter is present the parameter 107 should be present as well.

107=Windows to DOS conversion table

Specifies the name of a *Windows to DOS* character conversion table. This table is used by WinISIS to convert characters from Windows codes to DOS codes during writing database records (see below the subchapter *Characters conversions*). When this parameter is absent the standard conversion the Windows system table is used. If this parameter is present the parameter 106 should be present as well.

108=screen orientation

Specifies in which direction characters are displayed on the screen. This parameter can have the following values:

- 0 from left to right (default),
- 1 from right to left (used for instance in Arabic).

131=word length

Specifies the maximum length of a word. This is required for a language that does not use spaces to separate words (e.g. Chinese).

8.2.10. Parameter file redirection

0=parameter file redirection

Specifies the name (optionally preceded by drive and/or directory specification) of the next system parameter file. If value of this parameter begins from the ? or ! sign, the text following this sign serves merely as a prompt and the actual file name is read from the keyboard. Moreover, if the prompt is preceded by the ! sign a user's answer is not displayed on the screen. All definitions following this parameter in the SYSPAR.PAR file are ignored and the system control is passed directly to the specified system parameter file. In WinISIS this file must have extension PAR, but only the file name should be entered in response to the system prompt. If next system parameter file has been used, all parameters set automatically by WinISIS are written into this file. Separate system parameter files can be defined for several users applying WinISIS on a particular computer (see below the subchapter *System personalization*).

8.2.11. Miscellaneous parameters

132=display mode of CDS/ISIS Pascal errors

Specifies whether information on errors detected by WinISIS during compilation of a CDS/ISIS Pascal program is displayed in a dialog box. This parameter can have the following values:

- 0 no dialog box displayed (default),
- 1 description of an error is shown in a dialog box.

Irrespective of the value of this parameter, errors detected in a program are displayed in the CDS/ISIS Pascal compiler window. If the program was not successfully compiled WinISIS offers in this window the possibility to browse the program listing or to edit the program code.

160=initial WinISIS plug-in

In version 1.4 this parameter specifies the name of a WinISIS plug-in program that is executed when the WinISIS session begins. This parameter may appear several times.

161=terminal WinISIS plug-in

In version 1.4 this parameter specifies the name of a WinISIS plug-in program that is executed when the WinISIS session ends. This parameter may appear several times.

918=default record and field separators

Specifies the characters used as default separators of records and fields in the ISO files created during export operations. The parameter value is a two-character string and the first character is interpreted as a record separator and the second character as a field separator. If this parameter is absent WinISIS uses by default ASCII character 29 as a record separator and ASCII 30 as a field separator. The user can change these default settings before starting the export operation.

8.3. Access restrictions

8.3.1. Restricted access to system functions

In order to limit access to some functions of WinISIS that are potentially dangerous (such as *Import*, changing the database definition files, global operations on the database records available from the *Utilities* menu or even the *Data entry* function) the administrator should use a profile that does not contain menu options enabling execution of these functions.

Creation of a new profile has been described in detail in the section *Creating a new language version* in the chapter *Adapting CDS/ISIS to local requirements* of the WinISIS manual. However, before creating a new profile the administrator should make sure that the search only version of the system (profile code *SH*) does not satisfy his needs. This profile does not contain the *Import* function and it lacks the *Edit* and *Utilities* positions of the standard main menu. Moreover, the *Configure* position provides access only to the *Change language* and *View Configuration* functions. In particular, this profile does contain the *Change profile* function, because it would enable the user to choose the full version of the system. There is also no possibility to edit the current display format in the database window.

If the administrator creates his own restricted profile he should ensure that this profile will contain neither the *Change profile* nor the *System settings* function and he should not forget to prepare this profile in all language versions that are supported by the WinISIS installation. The name of this profile should be provided as value the parameter 102 on all computers used by persons who will not have access to all functions of WinISIS. If several persons with different access rights to the WinISIS functions use the same computer, separate system parameter files should be prepared for each category of users (see below the subchapter *System personalization*).

8.3.2. Database access restrictions

Various mechanisms of WinISIS can be used to restrict users' access to specific databases. Application of the system parameter 125=1 causes that the user cannot open a database using the *Open file* dialog box (in this box he can select any CDS/ISIS database), but he has to choose only a database from the list databases of used recently. This list defined by the parameters 1001, 1002, etc.

The administrator may define a separate list of available databases for each category of users or even for each user. He should also consider specification of database descriptions in the parameters above 1000. If such a description has been defined it will be displayed in the database list in place of a database name.

8.3.3. Restricted access to records or fields

If some fields of a database contain classified information the administrator may want to ensure that for unauthorized users these fields do not appear:

- on the display,
- in printouts,
- on data entry worksheets,
- in the dictionary.

This section discusses the design of display formats that present classified information to authorized users only.

To ensure that some fields are visible in display and print formats used by authorized persons and they are not visible to unauthorized users, in CDS/ISIS for DOS the administrator should prepare separate formats for each of these categories of users. However, management of such a database is cumbersome since these files must reside in separate directories (together with FDT, FST, FMT, STW and SRT files, because only for such a group of files a separate path in the *database.PAR* file can be defined⁵) and therefore an appropriate *database.PAR* file for each category of users must be provided.

In WinISIS management of such a database is much simpler. As previously, two formats must be defined: the *SHORT* format for unauthorized users and the *FULL* format for authorized users. However, we do not need provide a proper format to a given category of users. We may use the *SHORT* format as a default format (this format does not show fields containing classified data) and provide inside this format a hypertext link that will apply the *FULL* format as the current display format. However, we cannot apply the *FULL* format directly

```
link(('Display details'),'FORMAT full')
```

because this will show classified data to any user.

To make the *FULL* format accessible to authorized users only we will use a CDS/ISIS Pascal program with the *EDIT* function (see in the previous chapter the section *Using dialog windows* in the subchapter *Interaction with the user*) that will request the user to enter a password. We will apply the *FULL* format only if this password is correct. The *CHECK* program presented below returns the character string *OK* only if the user entered the character string *ITC*.

```
Program CHECK ( inp : string; lw, occ : real; out : string )[FORMAT];
{ Collects a password and returns OK only if it is correct }
var line : string;
    ans : real;
begin
line:=inp;
ans:=edit(line,0,0,0,0,1,
'Password dialog%Please enter your password%and click "OK" button');
if line='ITC' then out:='OK' else out:='';
end.
```

Because the password dialog window should appear upon the user's request, the *CHECK* program will be invoked from the intermediate format *PASS* that is executed only when the user clicks the hypertext link produced by the command

```
link(('Display details'),'FORMAT pass')
```

The *PASS* format could have the following contents:

```
if &check('')='OK' then
@full
else
{b,fs30,cl13,'You entered a ',fs40,'wrong password !!!'}/##
```

⁵ In consequence, copies of the same FDT table are stored in several directories and each of these copies must be changed if content of the table has to be modified. Modifications of the FST tables and data entry worksheets cause similar difficulties.

```

    'Click ',link(('here'),'FORMAT short'),' to restore the previous format.'
fi

```

Note that if the password is correct the *FULL* format is executed. This causes that requested information appears immediately on the screen without any user's intervention. Unfortunately, the current display format did not change - it is still the *PASS* format. In consequence, when the user presses any of the navigator buttons this format will be executed and he will be requested to enter the password again. In order to ask the user for a password only once, the *FULL* format should replace the current display format as soon as the user provides the correct password (in order to achieve this we have to request the user to click an appropriate hypertext link). In this case the *PASS* format may have the following contents:

```

if &check('')='OK' then
  {b,fs30,'You entered the ',fs40,'correct password.'}/##
  'Click ',link(('here'),'FORMAT full'),' to change the display format.'
else
  {b,fs30,cl13,'You entered a ',fs40,'wrong password !!!'}/##
  'Click ',link(('here'),'FORMAT short'),' to restore the previous format.'
fi

```

The weak point of the solution described above is that a user who does not know the password can choose the *FULL* format as the current display format by selecting it in the format box available in the database window. This possibility can be eliminated by applying a database view window (see below) instead of the database window.

8.3.4. Use of view mode

In WinISIS an additional level of protection is provided by the system parameter 124. If this parameter is set to 1 all default databases are open in database view windows rather than in standard database windows. A database view window behaves in the same way as a window open by the VIEW command (see in the previous chapter the section *Working with windows* in the subchapter *Hypertext formats*). It means that a database view window lacks all buttons and boxes of the default database window (MFN box, navigator buttons, display format box and dictionary button). Moreover, if a database view window is active the search and data entry operations cannot be executed from the menu or using the toolbar buttons. Note, that some potentially dangerous operations (e.g. modification of database definition elements, database re-initialization) can still be activated from the menu. An appropriate profile (see above the section *Restricted access to system functions*) should be applied to ensure that the user will not be able to execute such an operation.

The administrator may apply various hypertext commands in the default display format to provide in the database open in a view window the functional equivalents of some elements of the standard database window of WinISIS. For instance, he may in this format use the commands:

```

'Go to the ',link(('first'),'GOTO 1'),' ',
link(('previous'),'GOTO ',if val(mfn)>1 then f(mfn-1,1,0) else '1' fi),' ',
link(('next'),'GOTO ',
  if val(mfn)<val(&last('')) then f(mfn+1,1,0) else &last('') fi),' ',
link(('last'),'GOTO ',&last('')), ' record'/

```

to provide functions of the database navigator buttons of WinISIS. In order to get MFN number of the last database record these commands use the following *LAST* program:

```

Program LAST ( inp : string; lw, occ : real; out : string )[FORMAT];
begin
  out:=encint(maxmfn-1,1);
end.

```

8.4. Character conversions

In order to ensure that any CDS/ISIS database can be processed by the CDS/ISIS system in both DOS and Windows, WinISIS always writes database records to the disk using DOS character codes and performs conversion to Windows character codes when it reads these records from the disk. The character set used in DOS is called the OEM⁶ character set and the character set used in Windows is called the ANSI⁷ character set.

By default WinISIS performs the standard character conversions of the Windows system:

- the **OEM to ANSI conversion** is executed when records are read from the disk,
- the **ANSI to OEM conversion** is executed when records are written to the disk.

In majority of cases these conversions work properly and no special conversion is necessary in WinISIS.

The standard character conversions of the Windows system depend on the language chosen in the international settings of this system. This causes that in some cases these conversions will not convert CDS/ISIS databases properly, for instance a database prepared in a language other than the current language of the Windows system. In such a case the administrator should use the system parameters 106 and 107 to specify appropriate character conversion tables applied by WinISIS (see above the section *Language dependent parameters* in subchapter *System parameters*). Note that both of these parameters must be provided. Values of the parameters 106 and 107 are the names of character conversion tables that (as the system tables ISISUC.TAB and ISISAC.TAB) are located in the menu directory of WinISIS.

A **character conversion table** consists of 256 rows and its *j*-th row contains an integer corresponding to the ASCII code that after the conversion will be assigned to the character with ASCII code *j*. Note that contents of the system tables depends on the language of database records and when apply your own character conversion tables you may need to modify the standard contents of the ISISUC.TAB and ISISAC.TAB tables too.

If the administrator decided that CDS/ISIS databases created by WinISIS will not be processed by CDS/ISIS for DOS, he may want to ensure that no character conversion will be executed by WinISIS. To achieve this goal TXT_TXT.TAB should be specified as a value of the system parameter 106 and 107. In this table provided by UNESCO the consecutive integer numbers are stored in consecutive rows what means that character codes are not changed during the conversion.

8.5. System personalization

If several persons use the WinISIS system from the same computer, the administrator should create a personalized system parameter file for each user and apply the parameter 0 (see above the section *Parameter file redirection* in the subchapter *System files*) in the very end of the SYSPAR.PAR file to ensure that each user will apply his own setup of the WinISIS system.

In WinISIS the personalized system parameter files must have extensions PAR, but users refer to these files by their names only. The system stores in these files all the parameters that are set automatically. For instance, if John and David use WinISIS on the same computer, the **SYSPAR.PAR** file may have there the following contents:

```
; access paths
1=c:\winisis\prog\
2=c:\winisis\menu\
3=c:\winisis\msg\
4=c:\winisis\work\
5=c:\winisis\data\
141=c:\winisis\bmp\
148=c:\winisis\plugin\
```

⁶ OEM stands for an Original Equipment Manufacturer.

⁷ ANSI stands for the American National Standard Institute.

```

; do not show whole dictionary
104=1
; show toolbar and its help
120=1
121=1
; search terms highlighting
142=4,15
; auto I/F update
130=1
; no close db confirmation
137=0
; show search results in a table
980=1
981=explain
115=MS Sans Serif, 8, 8
116=MS Sans Serif, 8, 8
117=MS Sans Serif, 8, 8
0=?Enter your name

```

The following system parameter files may be applied by each of these users:

JOHN.PAR

```

102=DF
150=YMD
109=MS Sans Serif, 13
111=Courier New, 12
112=HP LaserJet 5L PCL,PCL5EMS,LPT1:
122=0, 0, 0, 1031, 747
1001=c:\winisis\data\cds.MST
1002=c:\winisis\data\titt\titt.MST
1003=c:\winisis\data\wts\wts.MST
1004=c:\winisis\data\mts\mts.MST

```

DAVID.PAR

```

6=d:\winisis\data\mts\mts
102=SH
125=1
109=MS Sans Serif, 13
111=Courier New, 12
112=HP LaserJet IIP Plus,HPPCL,LPT1:
122=0, 0, 0, 1031, 747
1001=c:\winisis\data\mts\mts.MST***MTS database
1002=c:\winisis\data\titt\titt.MST***ITC thesaurus

```

Note that John is using the full version of the system and he can access all CDS/ISIS databases, while David is using the search only version and he can open only two databases: MTS database and the ITC thesaurus (MTS is for him the default database).

Annex. The *Multilateral Trading System* database

This annex presents the MTS database storing bibliographic descriptions of various publications on the Multilateral Trading System and several auxiliary databases used in the MTS database.

1. MTS database

1.1. Record structure

A record of the MTS database describes a document. Descriptions of the following **types of documents** can be stored in this database :

- monographs (M),
- serials (S)
- periodicals (P),
- articles in periodicals (AP),
- articles in monographs (AM),
- web sites (W).

The field 4 of each record stores information on the type of this record. The record type is identified by the first letter or two first letters of this field (the record type identifiers are given above in parentheses).

The table below presents the fields of the MTS database.

Tag	Field
2	<i>Date created</i>
4	<i>Type of record</i>
5	<i>Author(s), persons</i>
6	<i>Type of material</i>
8	<i>Record status</i>
10	<i>Author(s), organizations</i>
12	<i>Author(s) of source</i>
15	<i>URL</i>
20	<i>Title(s)</i>
21	<i>Title(s) of source</i>
23	<i>Title(s) former</i>
25	<i>How ordered (s.o./sb./m.l./o.o.)</i>
27	<i>Re-order each year / as requested (y/)</i>
28	<i>Subscription period</i>
29	<i>Renewal date</i>
30	<i>Comments for agents</i>
35	<i>Ordered for (name(s))</i>
37	<i>Ordered for (account no(s))</i>
40	<i>This copy received as a gift (y/)</i>
41	<i>Free (to all) (y/)</i>
42	<i>List price, non-US\$, currency</i>
43	<i>List price, non-US\$, value</i>
44	<i>Calculated or estimated price in US\$</i>

45	List price in US\$
47	Rebate for ITC (%) (100=free)
50	Publisher name
52	Publisher code
61	Address line 1
62	Address line 2
63	Address line 3
90	City
100	Country
102	Fax
104	Telephone
106	E-mail
110	Order from (agent, immediate source)
111	Supplier's ref. no
121	B.C. Order number
122	Date of order or request
130	Collection code
140	Claimed (date)
142	Requested (date/ed.) (serial)
143	Date cancelled (serial)
144	Date discontinued (serial)
145	Date/ed. discarded (serial)
150	Date publication received
151	Editions received
156	Last edition received
158	Editions missing
161	Date editions received
200	Notes
210	Frequency
220	Call number
222	IFS code
225	Collection code (restricted, etc)
230	City of publication (as on catalog card)
240	Publisher (as on catalog card)
250	Year published
260	Pagination
262	Number of volumes
270	Series statement
275	Annotations (supplement, etc.)
280	Document number
282	Sales number (UN docs.)
290	Abstract
292	Subject descriptors
300	Language code(s)
310	Separate editions (y/)
320	ISBN
330	ISSN
350	Availability (e.g. free to DGCs)
900	Date updated

Note that the fields 50-106 contain publisher data. There are two ways of entering these data:

- publisher name, address and contact information can be stored in a record of the MTS database,
- a MTS record contains merely the publisher code (field 52) and then detailed information on this publisher is read from the PUB database (see below).

Sometimes the city of publication and/or publisher name printed in a book differ from the official publisher data. This is why the city of publication and publisher name (entered in the same way as on a catalog card) are stored in separate fields (230 and 240) of the MTS database.

The structure of the MTS database bases on the structure of the ITD⁸ database that stores main access catalogue of the documentation collection held by the Trade Documentation Center of ITC. The ITD database was designed in late 80's and this is why this database stores dates in the YYMMDD format. Appropriate literals have been used in formats of the ITD database to output these dates in the YYYYMMDD format. In the MTS database the same solution has been applied.

1.2. Data entry worksheets

The default data entry worksheet of the MTS database contains merely the record type field. Because a pick-up list of values has been defined for this field in the validation file (see the next section) the default value of this field is:

Click a button with red arrow to display the list of available values

Separate data entry worksheets have been defined for all types of records supported in the MTS database.

These are the names of all data entry worksheets defined:

Name	Purpose
MTS	default worksheet
MONOG	data entry of monographs
SERIA	data entry of serials
PERIO	data entry of periodicals
AMON	data entry of articles in monographs
APER	data entry of articles in periodicals
WWW	data entry of web sites

1.3. Validation file

This table below shows exemplary contents of the MTS.VAL file (each cell of the right column represents a single line of this file).

No	Line contents
1	begin:if mfn=0 then 'worksheet:MTS'/'a2#',date(2),'#'/ else if v4.1='M' then 'worksheet:MONOG'/ else if v4.1='S' then 'worksheet:SERIA'/ else if v4.1='P' then 'worksheet:PERIO'/ else if v4.2='AM' then 'worksheet:AMON'/ else if v4.2='AP' then 'worksheet:APER'/ else if v4.1='W' then 'worksheet:WWW'/ fi fi fi fi fi fi fi
2	4:choice:notype:firstdescribe::'Record types'/'Monograph'/'Monograph'/ 'Serial'/'Serial'/'Periodical'/'Periodical'/ 'Article in periodical'/'APeriodical'/'Article in monograph'/'AMonograph'/ 'Web site'/'Web site'/'
3	62:if a(v61) then 'Enter the first line of address' fi
4	63:if a(v61) then 'Enter the first line of address' else if a(v62) then 'enter line 2' fi fi

⁸ ITD stands for the International Trade Documentation.

5	:if v4.5='Click' then 'Edit the "Type of record" field and choose a value from the list' fi
6	end:if IsChanged() and mfn<>0 then if p(v2) then if v2<>date(2) then 'd900a900#',date(2),'#'/ fi else 'a2#',date(2),'#'/ fi fi

The line no 1 defines actions to be done before presentation of the current record for data entry:

- for a new record the MTS worksheet is applied and the current date is added as the record creation date (field2),
- for an existing record the appropriate worksheet is applied according to the record type (field 4).

The line no 2 defines the pick-up list for the *Record type* field. The lines no 3 and 4 contain validation formats for the fields 62 and 63. The line no 5 contains the record validation format that will not allow to store a record containing in the field 4 the explanatory text instead of the record type identifier. The line no 6 defines actions to be done when the current record editing is finished:

- if the record creation date (field2) was recorded, the current date is stored as the date of the last update operation (field 900) provided that this date differs from the record creation date,
- if the record creation date is missing the current date is stored as this date.

1.4. Dictionary

The following FST table is applied to generate dictionary of the MTS database:

```

4 0 if v4='AP' or v4='ap' or v4='AM' or v4='am' then 'Article' else v4 fi
5 4 v5
6 0 v6
8 0 if p(v8) then 'INACTIVE' else 'ACTIVE' fi
10 4 MHL,V10+|,|
20 5 '\TIT=\',MHL,V20+|,|
1020 8 '\TIT:\',MHL,V20+|,|
21 4 MHL,V21+|,|
23 4 MHL,V23+|,|
50 0 if a(v52) then v50+|. | else ref->pub(1->pub("PCOD="v52), v50+|. |) fi
210 0 v210
220 0 v220
222 0 v222
240 0 v240
250 0 v250
270 4 MHL,V270
280 0 v280
282 0 v282
292 5 '\DES=\',(v292/)
1292 8 (|\DES:\|v292/)
300 0 ( if npst->lang(v300)>0 then ref->lang(1->lang(v300),v2) else v300 fi/ )
320 0 v320
330 0 v330

```

Note that *Title* (field 20) and *Subject descriptors* (field 292) are indexed using two techniques applying the inverted file prefixes. In consequence the MTS dictionary contains the following groups of terms:

Identifier	Prefix	Group
20	TIT=	full titles
1020	TIT:	words from titles
292	DES=	full descriptors
1292	DES:	words from descriptors

Note also that the *Publisher name* (field 50) is indexed directly only if the *Publisher code* (field 52) is absent. If this code is present it is used as a reference to a respective record in the auxiliary database *PUB* (see description of this database below) and the publisher name from

this record is taken to the dictionary. Similarly, the *Language code* (field 300) is not indexed directly if the auxiliary database *LANG* contains description of a language with this code. In such a case the language code is just used as a reference to a respective record of the *LANG* database (see description of this database below).

1.5. Display and print formats

1.5.1. MTS format

This is the default display format of MTS database:

```
title('MTS database'),f2,m(2000),bpict(('winisis'),0),fs40,'Welcome to the'/
{fs60,b,'M'},'ultilateral',{fs60,b,'T'},'rading',
{fs60,b,'S'},'ystem'/'database !',fs20/#
m(0),{b,'MTS'},' is a bibliographic database aimed at facilitating research ',
'on the basic documents on the Multilateral Trading System. These publications
',
'constitute the core documentation base of the Reference Centers established ',
'by ITC under the JITAP project. The MTS database contains also descriptions ',
'of other documents (monographs, serials, periodicals, articles and Internet ',
'sites) related to the subject.'/#
qc,pict('un'),' MTS database is maintained by the ',
{b,'International Trade Centre UNCTAD/WTO '},pict('wto')/
qc,'using the software package CDS/ISIS for Windows developed by UNESCO.'/#
fs30,b,i,link(('Continue'),'FORMAT short')
```

This format displays merely some general information on this database and the hypertext link Continue that selects the *SHORT* format as the current display format.

1.5.2. SHORT format

This is the main display format of the MTS database:

```
mhl,f2,
if v4.1<>'M' and v4.1<>'m' and v4.1<>'S' and v4.1<>'s' and
v4.1<>'P' and v4.1<>'p' and v4.2<>'AP' and v4.2<>'Ap' and
v4.2<>'aP' and v4.2<>'ap' and v4.2<>'AM' and v4.2<>'Am' and
v4.2<>'aM' and v4.2<>'am' and v4.1<>'W' and v4.1<>'w' then
    box,c11,b,{i,'Invalid record type '},v4
else
if v4.1='M' or v4.1='m' then
    {b,c14,v250" ",if p(v8) then box(10),'INACTIVE' fi,c30,v220}/"#d8,
    (v5+| and |)/(v10/)/{b,(v20|. |)}/v230,
    if p(v240) then ": "v240,". "N250 fi," "v250". "v260" p. " ",
    "- "v262" vols. " " ("v270) " "- "v6"."/"Doc.No. "v280" " ",
    "Sales No. "v282/"#v290(3,3)/"#d290,@des,
    ( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2) else v300 fi,
    d300+|; | )," (Separate ed.)"D310/
    "Free"D41,v42" "v43" " "US$ "v44,"US$ "v45,c40," ISBN "v320," ISSN
    "v330/
    ""c30("v350(0,30)"/@pub,
    if p(v23) then #/{i,'Former title : '},v23 fi,
fi,
if v4.1='S' or v4.1='s' then
    {b,c14,v210" ",if p(v8) then box(10),'INACTIVE' fi,c30,v220}/"#d8,
    (v10/),{b,v20"."}/if p(v23) then '({i,'Formerly: '},v23)" fi/
    if a(v143) and a(v144) then
        if p(v230) or p(v240) then v230,": "v240,". ' fi,
        "- "v262" vols. " "- "v6"."/"#v290(3,3)/"#d290,@des,
        ( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2)
        else v300 fi, d300+|; | ),
        c30"Free"d41," US$ "v44," US$ "v45," "c50"ISSN "v330/@pub,
        "Separate ed."d310/
        if p(v151) then {i,'Editions received: '},
            ( v151|/|,
            if p(v161) then
```

```

        if v161<'45' then '20'v161 else '19'v161 fi
    fi, d151+|; | )
fi
fi
    ""#"DISCONTINUED "d144,
    if p(v144) then if v144<'45' then '20'v144 else '19'v144 fi fi/
    ""#"Discarded "v145+|; | /
fi,
if v4.1='P' or v4.1='p' then
    {b,c14,v210" ",if p(v8) then box(10),'INACTIVE' fi,c30,v220}/""#d8,
    {b,v20"."}/if p(v23) then '({i,'Formerly: ',v23})" fi/
    ""#v290(3,3)/""#d290,@des,
    ( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2) else v300 fi,
      d300+|; | ),
    if S(v41,v44,v45)>' then c30 fi,"Free"D41,"US$ "V44,"US$ "V45/@pub,
    if p(v156) then {i,'Last edition received ',v156/ fi,
    if p(v143) then
        {#b,i,c11,'CANCELLED '}, if v143<'45' then '20'v143 else '19'v143 fi
    fi/
    if p(v144) then {#b,i,c11,'DISCONTINUED '},
        if v144<'45' then '20'v144 else '19'v144 fi
    fi/
fi,
if v4.2='AP' or v4.2='Ap' or v4.2='aP' or v4.2='ap' then
    {b,c14,if p(v222) then {i,'IFS code: ',v222 fi,c30,v220}/
    v5+|, |/(v10/){b,v20|. |}/
    if p(v21) then '[In :] ',{i,v21+|. |},"", "v260," (ISSN "v330)" fi/
    ""#v290(3,3)/""#d290,@des
fi,
if v4.2='AM' or v4.2='Am' or v4.2='aM' or v4.2='am' then
    {b,c14,if p(v222) then {i,'IFS code: ',v222 fi,c30,v220}/
    v5+|, |/(v10/){b,v20|. |}/
    if p(v21) then
        '[In :] ',v12+|, |": ",{i,v21|. |},"(",v270)" ", "- "V262" vols. ",
        v230,if p(v240) then " : "V240,". "N250 fi,"", "V250". ",
        v260," (ISBN "v320)"
    fi/
    ""#v290(3,3)/""#d290,@des
fi,
if v4.1='W' or v4.1='w' then
    {b,c14,v210" ",'available on Internet ',
    if p(v8) then box(10),'INACTIVE' fi,c30,v220}/""#d8,
    {b,v20"."}/if p(v15) then {i,'URL: ',link((v15),'OPENFILE ',v15) fi/
    ""#v290(3,3)/""#d290,@des,
    ( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2) else v300 fi,
      d300+|; | ),
    if S(v41,v44,v45)>' then c30 fi,"Free"D41,"US$ "V44,"US$ "V45/@pub,
    if p(v143) then {#b,i,c11,'CANCELLED '},
        if v143<'45' then '20'v143 else '19'v143 fi
    fi/
    if p(v144) then {#b,i,c11,'DISCONTINUED '},
        if v144<'45' then '20'v144 else '19'v144 fi
    fi/
fi,
/i,link(('Display details'),'FORMAT full'),' ',
link(('Look for'),'PROMPT TEXTBOX[1]CHILDL0AD title,xxx,$$'),' ',
'Browse thesaurus in ',
link(('English'),'PROMPT VIEWL[1] titt,titt,(E=$$),1'),' ',
link(('French'),'PROMPT VIEWL[1] titt,franc,(F=$$),1'),' ',
link(('Spanish'),'PROMPT VIEWL[1] titt,espagn,(S=$$),1')

fi

```

This format is divided into six sections - each section corresponds to a separate type of records and shows the most important data stored in a record of the given type. The *SHORT* format includes the *PUB* and *DES* formats (see below) that output the publisher data and the subject descriptors respectively.

Note the commands applied to show the date when the last edition of a serial was received (field 161):

```
if p(v161) then if v161<'45' then '20'v161 else '19'v161 fi
```

This field contains a date in the *YYMMDD* format and if the year is smaller than 45 the date is preceded by 20, otherwise it is preceded by 19. This way years from the range 1945-2044 are output using four digits in spite that the field stores a year using two digits only. Similar tricks have been applied in the *SHORT* format in order to display properly other dates stored in the *YYMMDD* format

The auxiliary database LANG is used in order to display language names instead of their codes (occurrences of the field 300). The TITT database is another auxiliary database used in the *SHORT* format. This database stores the *Thesaurus of International Trade Terms* and it is open a view window when the user clicks the hypertext links [English](#), [French](#) or [Spanish](#) that are shown after the current record description. The thesaurus contains terms in these languages and if the users clicks one of these links the system requests him to enter the term he is looking for. If this is a term from the thesaurus, the view window shows the corresponding record of the TITT database, otherwise it shows its first record.

The hypertext links allowing to browse the thesaurus are preceded by two other hypertext links. The link [Display details](#) applies the *FULL* format (see the next section) as the current display format and the link [Look for](#) shows in a separate text window the titles of all publications containing in their titles a word entered by the user. For this purpose the polymorphic format *TITLE* (see below) is applied.

1.5.3. FULL format

This is the alternative display format of the MTS database that is applied when the user clicks the hypertext link [Display details](#) in the main display format *SHORT*:

```
mhl,f2,
if v4.1<>'M' and v4.1<>'m' and v4.1<>'S' and v4.1<>'s' and
v4.1<>'P' and v4.1<>'p' and v4.2<>'AP' and v4.2<>'Ap' and
v4.2<>'aP' and v4.2<>'ap' and v4.2<>'AM' and v4.2<>'Am' and
v4.2<>'aM' and v4.2<>'am' and v4.1<>'W' and v4.1<>'w' then
  box,c11,b,{i,'Invalid record type '},v4
else
if v4.1='M' or v4.1='m' then
  {b,c14,V250" ",if p(v8) then box(10),'INACTIVE' fi,c30,V220}/"#d8,
  (V5+| and |)/(V10/)/{b,(V20|. |)}/V230,
  if p(v240) then ": "V240,". "N250 fi,". "V250". ",V260" p. ",
  "- "V262" vols. ", " ("V270)" ", "- "v6"."/Doc.No. "V280"      ",
  "Sales No. "V282/"#v290(3,3)/"#d290,@descr/
  ( if npst->lang(v300)>0 then ref->lang(1->lang(v300),v2) else v300 fi,
  d300+|; | )," (Separate ed.)"d310/
  "Free"D41,v42" "v43" ",
  if p(v43) then "(US$ "V44")" else "US$ "V44 fi,"US$ "V45,
  if p(v44) or p(v45) then (rebate for ITC "v47"%)" fi,c40,
  "ISBN "V320," ISSN "V330/
  ""c30("V350(0,30)"/@publ,
  if p(v25) then {i,'Type of order/how received : '},
    if v25.1='S' or v25.1='s' then 'Standing order' else
      if v25.1='M' or v25.1='m' then 'Mailing list' else
        if v25.1='O' or v25.1='o' then 'Once only (paid)' else
          if v25.1='F' or v25.1='f' then 'Once only (free)' else v25
          fi
        fi
      fi
    fi
  fi/
  if p(v27) then qc,box(10),{b,i,c11,'RE-ORDER each year'} fi
  if p(v110) then {i,'Ordered through/requested from: '},V110 fi/
  if p(v35) then {i,'Ordered for '},V35 fi/
  if p(v121) then {i,'B.C.No. '},V121" " fi,
  if p(v122) then {i,'Date ordered/requested: '},
    if v122<'45' then '20'v122' ' else '19'v122' ' fi
```

```

fi,
  "Date(s) claimed: "v140+|, |," _ _ _ ORDER CANCELLED _ _ _ "D143,
  if p(v150) then {i,'Publ.received on '},
    if v150<'45' then '20'v150 else '19'v150 fi
  fi," (this copy recd. as gift)"D40/
  if p(v23) then #/{i,'Former title : '},v23 fi,
  if p(v200) then m(800,-800)#/{i,b,'Notes: '},tab(800), v200 fi
fi,
if v4.1='S' or v4.1='s' then
  {b,c14,v210" ",if p(v8) then box(10),'INACTIVE' fi,c30,v220}/"#d8,
  (v10/),{b,v20".}"/if p(v23) then '(',{i,'Formerly: '},v23)" fi/
  if p(v230) or p(v240) then v230," : "v240,'. ' fi,
  "- "v262" vols. ", "- "v6"."/
  ""#v290(3,3)/"#d290,@descr/"ISSN "v330/
  ( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2)
    else v300 fi, d300+|; | )," (Separate ed.)"d310,
  if S(v41,v43,v44,v45)>' then c30 fi,"Free"D41,v42" "v43" ",
  if p(v43) then "(US$ "V44")" else "US$ "V44 fi,"US$ "V45,
  if p(v44) or p(v45) then " (rebate for ITC "v47"%)" fi/
  if p(v110) then {i,'Supplier : '},v110," Ref.no "v111/ fi,
  @publ,v262" vols."/
  if p(v25) then {i,'Type of order/how received : '},
    if v25='S' or v25='s' then 'Standing order' else
      if v25='M' or v25='m' then 'Mailing list' else
        if v25='O' or v25='o' then 'Once only (paid)' else
          if v25='F' or v25='f' then 'Once only (free)' else v25
        fi
      fi
    fi
  fi
  fi
  fi
fi/
if p(v151) then {i,'Editions received: '},
  ( v151|/|,
    if p(v161) then if v161<'45' then '20'v161 else '19'v161 fi fi,
    d151+|; | )
fi/
if p(v150) then {i,'First received '},v150 fi/
if p(v121) then {i,'B.C.No. '},v121" " fi,
if p(v122) then {i,' B.C.date '},
  if v122<'45' then '20'v122 else '19'v122 fi
fi/
" Requested "v142(3,3)+|; |/" Claimed "v140(3,3)+|; |/"
if p(v143) then {#b,i,c11,'CANCELLED '},
  if v143<'45' then '20'v143 else '19'v143 fi
fi/
if p(v144) then {#b,i,c11,'DISCONTINUED '},
  if v144<'45' then '20'v144 else '19'v144 fi
fi/
""#"Discarded "v145+|; |/"#d145/
if p(v200) then m(800,-800),{i,b,'Notes: '},tab(800), v200 fi/
if p(v27) then {qc,box(10),{b,i,c11,'RE-ORDER each year'}} fi/
fi,
if v4.1='P' or v4.1='p' then
  {b,c14,v210" ",if p(v8) then box(10),'INACTIVE' fi,c30,v220}/"#d8,
  {b,v20".}"/if p(v23) then '(',{i,'Formerly: '},v23)" fi/
  ""#v290(3,3)/"#d290,@descr/
  ( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2)
    else v300 fi, d300+|; | ),
  if S(v41,v43,v44,v45)>' then c30 fi,"Free"D41,v42" "v43" ",
  if p(v43) then "(US$ "V44")" else "US$ "V44 fi,"US$ "V45,
  if p(v44) or p(v45) then " (rebate for ITC "v47"%)" fi/
  if p(v25) then {i,'How received : '},
    if v25='S' or v25='s' then 'Subscription' else
      if v25='M' or v25='m' then 'Mailing list' else
        if v25='E' or v25='e' then 'Exchange' else v25
      fi
    fi
  fi
  fi
  fi
fi/
if p(v28) then {i,'Subscribed for '},v28" " fi,
if p(v29) then {i,b,c11,'Renew subscription on '},
  if v29<'45' then '20'v29 else '19'v29 fi

```

```

fi/@publ,
if p(v110) then {i,'Supplier : ',v110/ fi,
if p(v150) then {i,'First received '},
  if v150<'45' then '20'v150 else '19'v150 fi/
fi,
if p(v156) then {i,'Last edition received '},v156/ fi,
if p(v158) then {i,'Missing editions '},v158+|; | / fi,
if p(v121) then {i,'B.C.No. '},V121" " fi,
if p(v122) then {i,' B.C.date '},
  if v122<'45' then '20'v122 else '19'v122 fi
fi/
"#"Discarded "v145+|; |/"#d145/
if p(v200) then m(800,-800),{i,b,'Notes: '},tab(800), v200 fi/
if p(v143) then {#b,i,c11,'CANCELLED '},
  if v143<'45' then '20'v143 else '19'v143 fi
fi/
if p(v144) then {#b,i,c11,'DISCONTINUED '},
  if v144<'45' then '20'v144 else '19'v144 fi
fi/
fi,
if v4.2='AP' or v4.2='Ap' or v4.2='aP' or v4.2='ap' then
  {b,c14,if p(v222) then {i,'IFS code: '},v222 fi,c30,v220}/
  v5+|, |/(v10/){b,v20|. |}/
  if p(v21) then '[In :] ',{i,v21+|. |},", "v260," (ISSN "v330)" fi/
  "##v290(3,3)/"##d290,@descr/
  ( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2)
    else v300 fi, d300+|; | ),
  if p(v200) then m(800,-800),{i,b,'Notes: '},tab(800), v200 fi/
fi,
if v4.2='AM' or v4.2='Am' or v4.2='aM' or v4.2='am' then
  {b,c14,if p(v222) then {i,'IFS code: '},v222 fi,c30,v220}/
  v5+|, |/(v10/){b,v20|. |}/
  if p(v21) then
    '[In :] ',v12+|, |": ",{i,v21|. |},("v270") ", "- "V262" vols. ",
    V230,if p(v240) then ": "V240,". "N250 fi,", "V250". ",v260,
    " (ISBN "v320)"
  fi/
  "##v290(3,3)/"##d290,@descr/
  ( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2)
    else v300 fi, d300+|; | ),
  if p(v200) then m(800,-800),{i,b,'Notes: '},tab(800), v200 fi/
fi,
if v4.1='W' or v4.1='w' then
  {b,c14,v210" ", 'available on Internet ',
  if p(v8) then box(10), 'INACTIVE' fi, c30,v220}/"##d8,
  {b,v20". "}/if p(v15) then {i,'URL: '},link((v15),'OPENFILE ',v15) fi/
  "##v290(3,3)/"##d290,@descr/
  ( if npst->lang(v300)>0 then
    ref->lang(l->lang(v300),v2)
    else v300 fi, d300+|; | ),
  if S(v41,v43,v44,v45)>' then c30 fi,"Free"D41,v42" "v43" ",
  if p(v43) then "(US$ "V44)" else "US$ "V44 fi,"US$ "V45,
  if p(v44) or p(v45) then " (rebate for ITC "v47"%)" fi/
  if p(v28) then {i,'Subscribed for '},v28" " fi,
  if p(v29) then {i,b,c11,'Renew subscription on '},
    if v29<'45' then '20'v29 else '19'v29 fi
fi/@publ,
if p(v110) then {i,'Supplier : ',v110/ fi,
if p(v121) then {i,'B.C.No. '},V121" " fi,
if p(v122) then {i,' B.C.date '},
  if v122<'45' then '20'v122 else '19'v122 fi
fi/
"#"Discarded "v145+|; |/"#d145/
if p(v200) then m(800,-800),{i,b,'Notes: '},tab(800), v200 fi/
if p(v143) then {#b,i,c11,'CANCELLED '},
  if v143<'45' then '20'v143 else '19'v143 fi
fi/
if p(v144) then {#b,i,c11,'DISCONTINUED '},
  if v144<'45' then '20'v144 else '19'v144 fi
fi/
fi,

```

```

if p(v2) or p(v900) then %#
  if p(v2) then {i,'Date entered '},
    if v2<'45' then '20'v2 else '19'v2 fi
  fi,
  if p(v900) then c20,{i,'Date updated '},
    if v900<'45' then '20'v900 else '19'v900 fi
  fi
fi,
' ',i,link(('Abridged display'),'FORMAT short'),' '/
link(('Look for'),'PROMPT TEXTBOX[1]CHILDLLOAD title,xxx,$$'),' ',
'Browse thesaurus in ',
link(('English'),'PROMPT VIEWL[1] titt,titt,(E=$$),1'),' ',
link(('French'),'PROMPT VIEWL[1] titt,franc,(F=$$),1'),' ',
link(('Spanish'),'PROMPT VIEWL[1] titt,espagn,(S=$$),1')
fi

```

The *FULL* format has exactly the same structure as the *SHORT* format, but it displays all the data stored in records of the MTS database. The only differences are that it includes the formats *PUBL* and *DESCR* (see below) rather than *PUB* and *DES* and it displays the hypertext link *Abridged display* that enables the user to apply the *SHORT* format again.

1.5.4. DES format

This format is included in the *SHORT* format in order to display all subject descriptors (occurrences of the field 292):

```

if p(v292) then {i,'Subject descriptors: '} fi,
( if p(v292) then
  if npst->titt(|E|=v292)>0 then
    {b,link((v292),'TEXTBOX[1]CHILDLLOAD d1,xxx,'v292)},' ',
  else
    if npst->titt(|F|=v292)>0 then
      {b,link((v292),'TEXTBOX[1]CHILDLLOAD d1,xxx,'v292)},' ',
    else
      if npst->titt(|S|=v292)>0 then
        {b,link((v292),'TEXTBOX[1]CHILDLLOAD d1,xxx,'v292)},' ',
      else
        v292
      fi,
    fi,
  fi,
fi,
fi, d292+|; | )/

```

References to the inverted file of the TITT database are made in order to check whether the current descriptor is a term from the *Thesaurus of International Trade Terms* (English, French and Spanish terms of the thesaurus are indexed using the *E=*, *F=* and *S=* prefixes respectively) and if this condition is satisfied this descriptor is displayed as a hypertext link. When the user clicks such a link, titles of all publications that are indexed with this descriptor are displayed in a separate window. This window shows an output text of the polymorphic format *D1* (see below).

1.5.5. DESCR format

This format is included in the *FULL* format in order to output all subject descriptors (occurrences of the field 292):

```

if p(v292) then {i,'Subject descriptors: '} fi,
( if p(v292) then
  if npst->titt(|E|=v292)>0 then
    {b,link((v292),'TEXTBOX[1]CHILDLLOAD d1,xxx,'v292)},' ',
    ref->titt(l->titt(|E|=v292),
      if p(v530) then ' ({i,'BT '},
        link((v530),'TEXTBOX[1]CHILDLLOAD d2,xxx,'v530),
        ref(l(|E|=v530),
          if p(v530) then ", "d530,
            link((v530),'TEXTBOX[1]CHILDLLOAD d2,xxx,'v530)
          fi,
          ref(l(|E|=v530),
            if p(v530) then ", "d530,

```



```

        if p(v532) then ", "d532,
            link((v532),
                'TEXTBOX[1]CHILDL0AD d2,xxx, 'v532)
            fi,
        )
    ),
    )"d532
else
    if p(v583) then ' ({i,'RT '},
        ( if p(v583) then
            link((v583),
                'TEXTBOX[1]CHILDL0AD d2,xxx, 'v583),
            d583+|, |
            fi ), d583)"
        fi
    fi
)
else
    v292
    fi,
    fi,
    fi,
    fi, d292+|; | )/

```

Similarly as in the *DES* format, references to the inverted file of the TITT database are made in order to check whether the current descriptor is a term from the *Thesaurus of International Trade Terms* and if this condition is satisfied this descriptor is displayed as a hypertext link. If the referenced record of the TITT database contains the broader term an appropriate language version of this term (the fields 530, 531 and 532 store the broader terms in English, French and Spanish respectively) is displayed as a hypertext link in parentheses preceded by the *BT* literal. Moreover, existence of the of a broader term is checked for the broader term of the current descriptor and such a term is displayed too. This operation is repeated three times until the whole hierarchy of broader terms is displayed.

This is an exemplary list of descriptors displayed by the *DESCR* format:

Subject descriptors: TRADE POLICY (*BT* INTERNATIONAL TRADE); WTO (*BT* INTERNATIONAL ORGANIZATIONS, INTERNATIONAL COOPERATION); PARAGUAY (*BT* SOUTH AMERICA, LATIN AMERICA)

The current record of the MTS database has three subject descriptors: *TRADE POLICY*, *WTO* and *PARAGUAY*. All these descriptors are the thesaurus terms that have the broader terms *INTERNATIONAL TRADE*, *INTERNATIONAL ORGANIZATIONS* and *SOUTH AMERICA* respectively. Two of these terms have broader terms too: *INTERNATIONAL COOPERATION* is the broader term of *INTERNATIONAL ORGANIZATIONS* and *LATIN AMERICA* is the broader term of *SOUTH AMERICA*.

If the current subject descriptor does not have a broader term, existence of the related term is checked in the thesaurus and if such a term exists it is displayed in parentheses preceded by the *RT* literal. Both broader and related terms are displayed as hypertext links, but if the user clicks such a link output text of the polymorphic format *D2* (see below) rather than *D1* is shown in a separate window.

1.5.6. D1 format

This polymorphic format displays titles of all publications that are indexed with a subject descriptor that replaces the character string *xxx*:

```

f2, title('List of publications in the MTS database'),
if npst('DES=xxx')>1 then
    {b,fs30,qc,'Publications on "xxx"}/
    m(300,-300), ref(lr(('DES=xxx')),/link((v20),'GOTO 'mf))
else
    {b,fs30,qc,'No other publications on "xxx"'}
fi

```

The *D1* format is used in the *DES* and *DESCR* formats as a parameter of the hypertext command:

```
link((v292), 'TEXTBOX[1]CHILDLLOAD d1,xxx, 'v292)
```

that displays the current subject descriptor (field 292) as a hypertext link. If the user clicks this link, in the *D1* format each occurrence of the character string *xxx* is replaced with this descriptor and then output text generated by this format is displayed in the text window number 1. Note that if the subject descriptor appears in the inverted file only once, this window contains merely the heading:

No other publications on "*Descriptor*"

(description of the only publication indexed with this descriptor is already displayed in the database window). Otherwise the command

```
ref(lr('DES=xxx'), /link((v20), 'GOTO 'mfn))
```

is executed in order to display in this window titles (fields 20) of all publications indexed with this descriptor as hypertext links. The GOTO command, that is executed when the user clicks such a title, results in displaying description of this publication in the database window (this effect is achieved because the text window was created as a child window of the current database).

1.5.7. D2 format

As the *D1* format the polymorphic format *D2* displays titles of all publications that are indexed with a dictionary term that replaces the character string *xxx*:

```
f2, title('List of publications in the MTS database'),
if npst('DES=xxx')>0 then
  {b,fs30,qc,'Publications on "xxx"}/
  m(300,-300), ref(lr('DES=xxx'), /link((v20), 'GOTO 'mfn))
else
  {b,fs30,qc,'No publications on "xxx"'}
fi
```

The *D2* format is used in the *DESCR* format as a parameter of the hypertext command:

```
link((vnnn), 'TEXTBOXCHILDLLOAD d2,xxx, 'vnnn)
```

that shows as a hypertext link either the broader term or the related term of the current subject descriptor (these terms are stored in the TITT database in various fields and *nnn* is the tag of a field storing the appropriate term). In contrast to the *D1* format the *D2* format displays as hypertext links titles of all publications indexed with the specified term if only this term was found in the dictionary, otherwise the *D2* format displays the message:

No publications on "*Term*"

1.5.8. PUB format

This format is included in the *SHORT* and *PRINT* (see below) formats in order to output the publisher data:

```
if p(v52) then
  if npst->pub("PCOD="v52)>0 then
    ref->pub( l->pub("PCOD="v52),V50|, |,V90", ",V100 )
  else
    {b,i,c11,'Invalid publisher identifier: ', ">>> "v52" <<<"
  fi
else
  V50", ",V90", ",V100
fi/
```

As it was mentioned in the section *Record structure*, there are two ways of entering publisher data in the MTS database:

- publisher name, address and contact information are stored in a record of the MTS database,
- a MTS record contains merely the publisher code (field 52) and then detailed information on this publisher is read from the PUB database (in the dictionary of this database the publisher identifiers are stored with the prefixes *PCOD=*).

1.5.9. PUBL format

This format is included into the *FULL* format in order to output the publisher data:

```
if p(v52) then {i,'Publisher : '},
  if npst->pub("PCOD="v52)>0 then
    ref->pub( l->pub("PCOD="v52),v50+|, |,{m(1200,-100),v90/v100} ),
    ' ( ',link(('Publisher details'),
      'VIEWL[1] pub,source,(',"PCOD="v52,')'),' )'/
  else
    {b,i,c11,'Invalid publisher identifier: '},">>> "v52" <<<"
  fi
else
  if p(v50) then
    {i,'Publisher: '},tab(1100),v50/
    {m(1200,-100),v61/v62/v63/v90/v100/"Fax "v102/"Tel."v104/}
  fi
fi/
```

If the publisher data are stored in a record of the MTS database all these data are displayed, otherwise the basic publisher data are followed on the screen by the hypertext link *Publisher details*. When the user clicks this link the detailed information stored on this publisher in the PUB database is displayed in the view window number 1 (this output is generated by the display format *SOURCE*, see description of the PUB database below).

1.5.10. TITLE format

This polymorphic format displays titles of all publications that contain in their titles a word entered by the user that replaces the character string *xxx*:

```
f2,title('List of publications in the MTS database'),
  if npst('TIT:xxx')>0 then
    {b,fs30,qc,'Publications containing "xxx" in a title'}/m(300,-300),
    ref(lr(('TIT:xxx')),/link((v20),'GOTO 'mf))
  else
    {b,fs30,qc,'No publication contains "xxx" in a title'}
  fi
```

The *TITLE* format is applied in the *SHORT* and *FULL* format as a parameter of the hypertext command:

```
link(('Look for'),'PROMPT TEXTBOX[1]CHILDLOAD title,xxx,$$')
```

If the user clicks the hypertext link *Look for* the system requests him to enter the word and displays in the window number 1 output text generated by the *TITLE* format. If the word entered by the user appears in the inverted file (in the dictionary of the MTS database the words from the *Title* field are stored with the prefixes *TIT:*), the window shows titles of all publications containing the specified word in their titles, otherwise the window displays just text:

No publications contains "word" in a title

These titles are displayed as hypertext links and when the user clicks such a title in the text window, contents of an appropriate record is displayed in the database window.

1.5.11. PRINT format

This is the main print format of the MTS database:

```
mhl,f2,
  if v4.1<>'M' and v4.1<>'m' and v4.1<>'S' and v4.1<>'s' and
    v4.1<>'P' and v4.1<>'p' and v4.2<>'AP' and v4.2<>'Ap' and
    v4.2<>'aP' and v4.2<>'ap' and v4.2<>'AM' and v4.2<>'Am' and
    v4.2<>'aM' and v4.2<>'am' and v4.1<>'W' and v4.1<>'w' then
    box,c11,b,'MFN='f(mfn,1,0),c30,{i,'Invalid record type '},v4
  else
    if v4.1='M' or v4.1='m' then
      {b,c14,'MFN='f(mfn,1,0),c30,v250" ",
        if p(v8) then box(10),'INACTIVE' fi,c50,v220}/"#d8,
        (v5+| and |)/(v10/)/{b,(v20|. |)}v230,
        if p(v240) then ": "v240,". "N250 fi," "v250". "v260" p. ",
        "- "v262" vols. ", " ("v270)" ", "- "v6"/"Doc.No. "v280" " ,
```

```

"Sales No. "V282/"##v290(3,3)/"#d290,
if p(v292) then {i,'Subject descriptors: '},{b,v292+|; |} fi/
( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2) else v300 fi,
  d300+|; | )," (Separate ed.)"d310/
"Free"D41,v42" "v43" ", "US$ "V44,"US$ "V45,c40," ISBN "V320," ISSN "V330/
"c30"("V350(0,30)"/"@pub,
if p(v23) then #/{i,'Former title: '},v23 fi,
fi,
if v4.1='S' or v4.1='s' then
{b,c14,'MFN='f(mfn,1,0),c30,v210" ",
  if p(v8) then box(10),'INACTIVE' fi,c50,v220}/"#d8,
(v10/},{b,v20".}"/if p(v23) then '(',{i,'Formerly: '},v23)" fi/
if a(v143) and a(v144) then
  if p(v230) or p(v240) then v230,": "v240, ". ' fi,
  "- "v262" vols. " , "- "v6"."/"#v290(3,3)/"#d290,
  if p(v292) then {i,'Subject descriptors: '},{b,v292+|; |} fi/
  ( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2)
    else v300 fi, d300+|; | ),
  c30"Free"d41," US$"v44," US$"v45,"c50"ISSN "v330/@pub,
  "Separate ed."d310/
  if p(v151) then {i,'Editions received: '},
    ( v151|/|,
      if p(v161) then
        if v161<'45' then '20'v161 else '19'v161 fi
        fi, d151+|; | )
    fi
  fi
  "#DISCONTINUED "d144,
  if p(v144) then if v144<'45' then '20'v144 else '19'v144 fi fi/
  "#Discarded "v145+|; | /
fi,
if v4.1='P' or v4.1='p' then
{b,c14,'MFN='f(mfn,1,0),c30,v210" ",
  if p(v8) then box(10),'INACTIVE' fi,c50,v220}/"#d8,
{b,v20".}"/if p(v23) then '(',{i,'Formerly: '},v23)" fi/
"#v290(3,3)/"#d290,
if p(v292) then {i,'Subject descriptors: '},{b,v292+|; |} fi/
( if npst->lang(v300)>0 then ref->lang(l->lang(v300),v2) else v300 fi,
  d300+|; | ),
if S(v41,v44,v45)>' then c30 fi,"Free"D41,"US$ "V44,"US$ "V45/@pub,
if p(v156) then {i,'Last edition received '},v156/ fi,
if p(v143) then {#b,i,c11,'CANCELLED '},
  if v143<'45' then '20'v143 else '19'v143 fi
fi/
if p(v144) then {#b,i,c11,'DISCONTINUED '},
  if v144<'45' then '20'v144 else '19'v144 fi
fi/
fi,
if v4.2='AP' or v4.2='Ap' or v4.2='aP' or v4.2='ap' then
{b,c14,'MFN='f(mfn,1,0),c20,
  if p(v222) then {i,'IFS code: '},v222 fi,c50,v220}/
v5+|, |/(v10/){b,v20|. |}/
if p(v21) then '[In :] ',{i,v21+|. |},", "v260," (ISSN "v330)" fi/
"#v290(3,3)/"#d290,
if p(v292) then {i,'Subject descriptors: '},{b,v292+|; |} fi/
fi,
if v4.2='AM' or v4.2='Am' or v4.2='aM' or v4.2='am' then
{b,c14,'MFN='f(mfn,1,0),c20,
  if p(v222) then {i,'IFS code: '},v222 fi,c50,v220}/
v5+|, |/(v10/){b,v20|. |}/
if p(v21) then
  '[In :] ',v12+|, |": ",{i,v21|. |},"("v270)" ", "- "V262" vols. " ,
  V230,if p(v240) then ": "V240, ". "N250 fi," , "V250". " ,
  v260,," (ISBN "v320)"
fi/
"#v290(3,3)/"#d290,
if p(v292) then {i,'Subject descriptors: '},{b,v292+|; |} fi/
fi,
if v4.1='W' or v4.1='w' then
{b,c14,'MFN='f(mfn,1,0),c20,v210" ", 'available on Internet ',
  if p(v8) then box(10),'INACTIVE' fi,c50,v220}/"#d8,

```

```

{b,v20"."}/if p(v15) then {i,'URL: ',v15 fi/
""#v290(3,3)/""#d290,
if p(v292) then {i,'Subject descriptors: ',{b,v292+|; |} fi/
( if npst->lang(v300)>0 then ref->lang(1->lang(v300),v2) else v300 fi,
d300+|; | ),
if S(v41,v44,v45)>' ' then c30 fi,"Free"D41,"US$ "V44,"US$ "V45/@pub,
if p(v143) then {#b,i,c11,'CANCELLED '},
if v143<'45' then '20'v143 else '19'v143 fi
fi/
if p(v144) then {#b,i,c11,'DISCONTINUED '},
if v144<'45' then '20'v144 else '19'v144 fi
fi/
fi
fi/#

```

This format outputs exactly the same data as the *SHORT* format, but it does not contain any hypertext link. Moreover, the *PRINT* format outputs in the very beginning the MFN number of the current record and at the very end an empty line that separates consecutive records printed.

1.5.12. EXPLAIN format

This format displays the text explaining query formulation rules for MTS database:

```

f2,title('How to search the MTS database ?'),
{fs40,b,qc,'Searching the MTS database'}/#
{b,'MTS'},' is a bibliographic database aimed at facilitating research ',
'on the basic documents on the ',{i,{b,'M'},'utilateral ',{b,'T'},
'rating ',{b,'S'},'ystem' },'. These publications constitute the core ',
'documentation base of the Reference Centers established by ITC under ',
'the JITAP project. The MTS database contains also descriptions of other ',
'documents (monographs, serials, periodicals, articles and Internet sites) ',
'related to the subject.'/#
'The following elements of the MTS database are searchable: '//
{m(100,100),{i,b,'Type of record,','/'words from '{b,i,'Author(s) personal,','/
'Type of material,','/'Record status,','/
'words from '{b,i,'Author organization(s),','/
'words from '{b,i,'Title,','/
'words from '{b,i,'Title of the source}',' (for articles), '//
'words from '{b,i,'Former title, '//
'Frequency, '// 'Call number, '// 'IFS code' '// 'Publisher, '//
'Year published, '// 'words from '{b,i,'Series, '// 'Document no, '//
'Sales no, '// 'Subject descriptors, '// 'Languages, '// 'ISBN, '// 'ISSN. '//}}#
'Click the ',{i,'"Dictionary"'},
' button and select the appropriate field. Then type first characters of a
phrase you are looking for or press the > button several times until you
can see this term below. In order to apply a term in a query double click
it or drag it and drop over the search expression box.'/#
'If you like to build a complex query by combining several elements, press
an appropriate operand button before selecting the second and next
elements. Apply parentheses to change the standard order of operand
application. Remember that parentheses must be balanced (use the same number
of opening and closing parentheses).'//

```

The name of this format is provided as a value of the system parameter 981, what causes that the search windows contain the *Search tips* button if the tabular mode of displaying search results is applied. When the user clicks this button an output text of the *EXPLAIN* format is displayed in a separate window. Note that if several databases are used, the display format specified by the system parameter 981 must be defined for each database.

1.6. The MTS.EXP file

The MTS.EXP file located in the database directory stores the system parameters defining the default layout of the tabular display window for the MTS database. This file has the following contents:

```
500=print
```

```
502=mh1,'MFN=',f(mfn,1,0)," "v20
```

The *PRINT* format is used to show record contents in the right pane of the tabular display window (in contrast to the display formats this format does not contain hypertext links) and in the left pane the MFN number followed by the title (field 20) is displayed as a short description of each record retrieved.

2. TITT database

The TITT database stores the *Thesaurus of International Trade Terms* and its records contain the following fields:

Tag	Field
501	Main classification code
502	Auxiliary classification code
503	Term in English
504	Term in French
505	Term in Spanish
506	Use term in English
507	Use term in French
508	Use term in Spanish
509	Scope note in English
510	Scope note in French
511	Scope note in Spanish
512	Misc. synonyms in English
513	Misc. synonyms in French
514	Misc. synonyms in Spanish
515	Use for term in English
520	Use for term in French
525	Use for term in Spanish
530	Broader term in English
531	Broader term in French
532	Broader term in Spanish
533	Narrower term in English
543	Narrower term in French
553	Narrower term in Spanish
563	Related term in English
573	Related term in French
583	Related term in Spanish
593	Type of record
594	Notes
595	Field for internal use

The database structure has been designed for an early version of CDS/ISIS for DOS and this is why a term record stores in separate fields all language versions of thesaurus terms associated with the given term.

The following FST table is applied to generate dictionary of the TITT database:

```
501 0 v501
```

```

502 0 v502
503 0 "E="v503
504 0 "F="v504
505 0 "S="v505
593 0 if p(v593) then
      if v593.1<>'x' and v593.1<>'X' then 'Normal' else 'XRef' fi
    else
      if s(v506,v507,v508)='' then 'Normal' else 'XRef' fi
    fi

```

Note that English, French and Spanish terms of the thesaurus are indexed using the prefixes *E=*, *F=* and *S=* appropriately. In the dictionary the keyword *Normal* refers the descriptors and the keyword *Xref* refers the non-descriptors.

2.1. Display formats

Figure 1 shows the *TITT* format that is the main display format of the TITT database. This format displays the current thesaurus term and its description in English. Up to five levels of the broader and narrower terms are displayed with prefixes *BT*, *BT2*, *BT3*, *BT4*, *BT5*, *NT*, *NT1*, *NT2*, *NT3*, *NT4* and *NT5* respectively, while the related terms are displayed with the *RT* prefix. All the terms associated with the current term are displayed as hypertext links and when the user clicks such a link, description of an appropriate term is displayed in the database window. The user can search the thesaurus for a specified term by clicking the hypertext link *Look for* and if this term exists it is displayed in the database window. If the specified term does not exist the database window shows the next term available in the dictionary of the TITT database (this is result of the *LAGOTO* command that is applied as a parameter of this *LINK* command).

The hypertext links *French* and *Spanish* enable the user to browse the thesaurus in one of these languages. When the user clicks one of these links either the *FRANC* format or the *ESPAGN* format is applied as the current display format. Both these formats have exactly the same structure as the *TITT* format, but the hypertext links allow to select one of two other languages (English and Spanish in the case of the *FRANC* format or English and French in case of the *ESPAGN* format).

The *EXPLAIN* format displays the text explaining query formulation rules for TITT database. This format has the following contents:

```

f2,{fs40,b,qc,'Searching the TITT database'}/#'TITT ('
{i,{b,'T'],'hesaurus of',{b,'I'],'nternational',{b,'T'],'rade',{b,'T'],'
erms'},') is the database on containing revision 3 of this thesaurus. ',
'The following elements are provided for a term: descriptor, scope note, '
'broader terms (BTs), narrower terms (NTs) and related terms (RTs) ',
'as well as list of terms this descriptor is used for (for such a non-',
'preferred descriptor there is a list of terms to be used instead of it). ',
'All these elements are given in three languages: English, French and ',
' Spanish. Two types of classification codes are also provided for a term.'/#
'The following elements of the TITT database are searchable:/'
{m(100,100),i,b,'Main classification codes,/'
'Auxiliary classification codes,/'Descriptors in English,/'
'Descriptors in French,/'Descriptors in Spanish,/'
'Types of records'},' (',{i,'Normal'},' for preferred descriptors and ',
{i,'XRef'},' for non-preferred ones). '/#Click the',{i,'"Dictionary"'},
' button and select the appropriate field. Then type first characters of a
phrase you are looking for or press the > button several times until you
can see this term below. In order to apply a term in a query double click
it or drag it and drop over the search expression box.'/#
'If you like to build a complex query by combining several elements, press
an appropriate operand button before selecting the second and next
elements. Apply parentheses to change the standard order of operand
application. Remember that parentheses must be balanced (use the same
number of opening and closing parentheses)./'

```

```

title("Thesaurus of International Trade Terms" database'),f3,mpl,{b,fs30,v503},fs20," v5
{i,link(('French'),'FORMAT franc'),' ',link(('Spanish'),'FORMAT espagn')}/
v509(1,1)/
if p(v506) then {i,'Use '}, (if npst(|E|=v506)>0 then link((v506),'LGOTO ',|E|=v506) else v
( if p(v530) then
  {i,|BT |d530},
  if npst(|E|=v530)>0 then
    link((v530),'LGOTO ',|E|=v530)//,ref(1(|E|=v530), (
    if p(v530) then
      {i,| BT2 |d530},
      if npst(|E|=v530)>0 then
        link((v530),'LGOTO ',|E|=v530)//,ref(1(|E|=v530), (
        if p(v530) then
          {i,| BT3 |d530},
          if npst(|E|=v530)>0 then
            link((v530),'LGOTO ',|E|=v530)//,ref(1(|E|=v530), (
            if p(v530) then
              {i,| BT4 |d530},
              if npst(|E|=v530)>0 then
                link((v530),'LGOTO ',|E|=v530)//,ref(1(|E|=v530), (
                if p(v530) then
                  {i,| BT5 |d530}, if npst(|E|=v530)>0 then link((v530
                  fi ))
                else
                  v530/
              fi ))
            else
              v530/
            fi ))
          else
            v530/
          fi ))
        else
          v530/
        fi ))
      else
        v530/
      fi ))
    else
      v530/
    fi ))
  else
    v530/
  fi ))
( if p(v533) then
  {i,|NT |d533},
  if npst(|E|=v533)>0 then
    link((v533),'LGOTO ',|E|=v533)//,ref(1(|E|=v533), (
    if p(v533) then

```

```

{i,| NT2 |d533},
if npst(|E=|v533)>0 then
  link((v533),'LGOTO ',|E=|v533)/,ref(l(|E=|v533), (
  if p(v533) then
    {i,| NT3 |d533},
    if npst(|E=|v533)>0 then
      link((v533),'LGOTO ',|E=|v533)/,ref(l(|E=|v533), (
      if p(v533) then
        {i,| NT4 |d533},
        if npst(|E=|v533)>0 then
          link((v533),'LGOTO ',|E=|v533)/,ref(l(|E=|v533), (
          if p(v533) then
            {i,| NT5 |d533}, if npst(|E=|v533)>0 then link((v533
            fi ))
          else
            v533/
          fi
        fi ))
      else
        v533/
      fi
    fi ))
  else
    v533/
  fi
fi ))
else
  v533/
fi
fi ))
else
  v533/
fi
fi ))
else
  v533/
fi
fi ),
if p(v530) or p(v533) then # fi,
(if p(v563) then
  {i,|RT |d563}, if npst(|E=|v563)>0 then link((v563),'LGOTO ',|E=|v563) else v563 fi/
fi),
if p(v515) then
  if p(v563) then # fi, {i,'Used for '}, (if npst(|E=|v515)>0 then link((v515),'LGOTO ',|E
fi/
if p(v512) then {i,'Misc.synonimes: '},v512(1,1)+|; | fi/%#
i,link(('Look for'),'PROMPT LAGOTO E=$$')

```

Fig.1. Format TITT

2.1. Print formats

The main print format of the TITT database is called *PRINT* and it has the following contents:

```
@eprint/@fprint/@sprint
```

This format is a concatenation of the *EPRINT*, *FPRINT* and *SPRINT* formats that outputs descriptions of the current term in English, French and Spanish respectively. This is the content of the *EPRINT* format:

```
f3,mpl,{b,fs30,v503},fs20," "v501" "v502/  
v509(1,1)/  
if p(v506) then {i,'Use '}, (v506+| ; |) fi/  
( if p(v530) then  
  {i,|BT |d530},v530/,ref(1(|E|=v530), (  
  if p(v530) then  
    {i,| BT2 |d530},v530/,ref(1(|E|=v530), (  
    if p(v530) then  
      {i,| BT3 |d530},v530/,ref(1(|E|=v530), (  
      if p(v530) then  
        {i,| BT4 |d530},v530/,ref(1(|E|=v530), (  
        if p(v530) then {i,| BT5 |d530},v530/ fi ))  
      fi ))  
    fi ))  
  fi ))  
fi)  
( if p(v533) then  
  {i,|NT |d533},v533/,ref(1(|E|=v533), (  
  if p(v533) then  
    {i,| NT2 |d533},v533/,ref(1(|E|=v533), (  
    if p(v533) then  
      {i,| NT3 |d533},v533/,ref(1(|E|=v533), (  
      if p(v533) then  
        {i,| NT4 |d533},v533/,ref(1(|E|=v533), (  
        if p(v533) then {i,| NT5 |d533},v533/ fi ))  
      fi ))  
    fi ))  
  fi ))  
fi),  
if p(v530) or p(v533) then # fi,  
(if p(v563) then {i,|RT |d563},v563/ fi),  
if p(v515) then  
  if p(v563) then # fi,{i,'Used for '}, (v515+| ; |)  
fi/  
if p(v512) then {i,'Misc.synonimes: '},v512(1,1)+|; | fi/#
```

The *FPRINT* and *SPRINT* formats have exactly the same structure but they output descriptions in other languages. While the *PRINT* format outputs descriptions in all three languages, each of its component formats outputs descriptions in one language only and therefore any of these formats can be used an independent print format.

3. LANG database

The LANG database stores the language codes and their descriptions. Structure of this database is very simple, because its records contain only the following fields:

Tag	Field
1	Language code (two letters)
2	Language name

Both these fields are indexed and therefore the LANG.FST table contains the following lines:

```
1 0 v1  
2 0 v2
```

Indexing of the *Language code* field enables referring to the language names from the MTS database where the language codes are stored.

4. PUB database

The PUB database stores the publishers' descriptions. Records of this database contain the following fields:

Tag	Field
4	Type
47	Rebate
50	Publisher name
52	Publisher identifier
61	Address line 1
62	Address line 2
63	Address line 3
90	City
100	Country
102	Fax
104	Phone
106	Contact person
108	E-mail
200	Notes

The PUB.FST table contains the following lines:

```
47 0 v47
50 0 mhl,v50+|. |
52 0 v52
52 0 "PCOD="v52
100 0 v100
200 2 mpu,v200
```

Note that publisher codes are stored in the dictionary of the PUB database with the prefixes *PCOD=*. These codes are used to refer to the detailed publisher description from the MTS database.

4.1. Display and print formats

This is the default display format of the PUB database:

```
f2,M(1400,-1400),
if p(v50) then {i,b,c11,'Name : '},(if p(v50) then tab(1400),v50/fi) fi,
if p(v61) then
  {i,b,c11,'Address : '}, tab(1400),v61/
  if p(v62) then tab(1400),v62/ fi,
  if p(v63) then tab(1400),v63/ fi,
fi,
if p(v90) then {i,b,c11,'City : '},tab(1400),v90/ fi,
if p(v100) then {i,b,c11,'Country : '},tab(1400),v100/ fi,
if p(v104) then {i,b,c11,'Telephone : '},tab(1400),v104+|, | / fi,
if p(v102) then {i,b,c11,'Fax : '},tab(1400),v102+|, | / fi,
if p(v108) then {i,b,c11,'E-mail : '},tab(1400),v108+|, | / fi,
if p(v106) then {i,b,c11,'Contact: '},(if p(v106) then tab(1400),v106/ fi) fi,
if p(v47) then {i,b,c11,'Discount : '},tab(1400),v47"%"/ fi,
if p(v200) then {i,b,c11,'Comments : '},tab(1400),v200/ fi,
if p(v52) then #{i,b,c11,'Identifier : '}, tab(1400),v52/ fi
```

The *PRINT* format is main print format. It outputs exactly the same data as the *PUB* format, but these data are preceded by the MFN number of the current record.

The *SOURCE* format has the following contents:

```

f2,M(1400,-1400),title('Publisher ',v52),
if p(v50) then {i,b,c11,'Name : '},(if p(v50) then tab(1400),v50/fi) fi,
if p(v61) then
  {i,b,c11,'Address : '}, tab(1400),v61/
  if p(v62) then tab(1400),v62/ fi,
  if p(v63) then tab(1400),v63/ fi,
fi,
if p(v90) then {i,b,c11,'City : '},tab(1400),v90/ fi,
if p(v100) then {i,b,c11,'Country : '},tab(1400),v100/ fi,
if p(v104) then {i,b,c11,'Telephone : '},tab(1400),v104+|, | / fi,
if p(v102) then {i,b,c11,'Fax : '},tab(1400),v102+|, | / fi,
if p(v108) then {i,b,c11,'E-mail : '},tab(1400),v108+|, | / fi,
if p(v106) then {i,b,c11,'Contact: '},(if p(v106) then tab(1400),v106/ fi)
fi,
if p(v47) then {i,b,c11,'Discount : '},tab(1400),v47"%"/ fi,
if p(v200) then {i,b,c11,'Comments : '},tab(1400),v200/ fi

```

This format displays the publisher data in a view window when the user clicks the hypertext link *Publisher details* displayed by the *FULL* format of the MTS database.

The *AUTLIS* format has the following contents:

```
m(2500,-2500),f2,{b,v52},tab(2500),v50+| | /
```

This format is used for printing the authority list of publishers that contains merely the publisher codes and their names.

The *EXPLAIN* format displays the text explaining query formulation rules for PUB database. This format has the following contents:

```

f2,{fs40,b,qc,'Searching the PUB database'}/# 'PUB (' ,
{i,{b,'PUB'},'lisher'},') is the address database with publisher data. ',
'The following elements of the PUB database are searchable: '/
{m(100,100),i,b,'Rebate,'/'Publisher name,'/'Publisher id,'/'
'Country,'/'Notes.'/#}
'Click the ',{i,'"Dictionary"'},
' button and select the appropriate field. Then type first characters of a
phrase you are looking for or press the > button several times until you
can see this term below. In order to apply a term in a query double click
it or drag it and drop over the search expression box.'/#
'If you like to build a complex query by combining several elements, press
an appropriate operand button before selecting the second and next
elements. Apply parentheses to change the standard order of operand
application. Remember that parentheses must be balanced (use the same
number of opening and closing parentheses).' /

```