IFG V8.3A

IFG for FHS

Edition October 2006

Comments… Suggestions… Corrections…

The User Documentation Department would like to know your opinion on this manual. Your feedback helps us to optimize our documentation to suit your individual needs.

Fax forms for sending us your comments are included at the back of the manual.

There you will also find the addresses of the relevant User Documentation Department.

Certified documentation according to DIN EN ISO 9001:2000

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2000.

cognitas. Gesellschaft für Technik-Dokumentation mbH
www.cognitas.de

Copyright and Trademarks

Contents

1 Preface

The Interactive Format Generator IFG simplifies the creation of “screen forms”, also called “formats” or “masks”. Using IFG, anyone - even the non-dp specialist - can create formats interactively on a data display terminal. Not only does IFG make it easier for programmers to implement user prompting, it is itself based on the principle of user prompting.

User prompting on a data display terminal means

– making the display on the screen of a display terminal clear and informative,

– offering the terminal operator selection and response options,

– verifying the user’s keyboard input.

By creating a sequence of questions and answers in this way, i.e. splitting up the work into a succession of dialog steps, IFG eases the job of the terminal user, reduces possible sources of errors and allows non-dp specialists lacking any extensive training to make use of the computer.

The convenient IFG user prompting provides support for the user right up until the final stages, including storage of the format in a format library.

The formats created with IFG can be used with the Format Handling System FHS in the host computer.

IFG is itself an application program running in timesharing mode that makes use of the Format Handling System FHS for the format-driven dialog.

This User Guide is aimed not only at programmers who use formats and must be familiar with format handling in BS2000 (see “FHS V8.3A (BS2000/OSD)” User Guide or „openUTM V5.2“ manuals), but anyone who will subsequently be working at a display terminal with programs that incorporate formats.

A member of a non-dp department can, for example, draft the image of a format, say for a delivery note, to suit his or her own particular requirements (and have it printed out) and so cater better for the needs of the department.

Using IFG you can

– create formats (image drafting),
– view formats,
– modify formats,
– combine formats,
– produce key lists and messages,
– prepare formats for use,
– manage formats you have stored (print, delete, copy),
– request information selectively about IFG at any time (info mode),
– convert existing formats.

The IFG utility routine IFGFV is available for the administration functions printing, deleting, copying, creating addressing aids and object formats, as well as creating, selecting, deleting and copying user profiles, messages, and key lists. In contrast to IFG, IFGFV can be used to process multiple formats.

All the IFG images in this manual were generated on a 9750 Data Display Terminal. If you are working with IFG on another type of display terminal, the display on your screen may sometimes differ from the one shown in the manual.

1.1 Summary of contents

This manual is designed so that you can create your format step by step. The first part contains general notes on IFG: how to use it, how to execute a dialog, etc. This is followed by descriptions of how to create the format image with simple aids, of functions which simplify image creation, and of how to prepare the format for use.

The subsequent sections of the manual describe

– how to view, extend or combine existing formats;

– how to create tables

– how to print out, copy and delete formats

– how to create and administer key lists

– how to create and administer messages

– how to manage user profiles

– how to convert formats from earlier versions of the program

– how to manage formats with IFGFV

– how to use the support for Arabic/Persian formats

The appendix contains examples of addressing aids, notes on multiple field mode, an overview of how to create formats with and without a dialog extension, a description of how to convert formats generated with FHS macros, a complete list of IFG error messages, tables on the generation of terminals and a summary of 8-bit terminal support. This is followed by an index listing the screens displayed in this manual, a glossary, a list of related publications and an index.

1.2 Changes since the last version of the manual

Unicode formats

The exchange of formatted messages containing Unicode strings between application programs and terminals is supported. This is made possible by the use of Unicode formats generated by IFG V8.3A or up.

Unicode formats are formats in which the UNICODE field attribute has been assigned at least to one field of a format, see also section “Modify display attributes of fields” on page 94. IFG does not support Unicode input nor output.

The consequence of assigning the UNICODE attribute to a field is that:

– the user is able to type any character of the Unicode Basic Multilanguage Plane (BMP) into this field.

– the application program will receive the content of the field in the addressing aids in an area having a size of 2 bytes per character, see also section “What are addressing aids?” on page 37.

– the encoding of this field in the application program is UTF-16.

EURO currency symbol

Requirements

The EURO currency symbol has been introduced in XHCS 1.3A with 8-bit coded character set table EDF04F and requires at least the following environment:

– VTSU-B from V13.0A

– XHCS-SYS from V1.3A

– a terminal/emulation supporting character set ISO8859-15.

Note

In order to use IFG with an old ARABIC EDF04F table instead of the new EURO EDF04F table, the optional correction A0429830 must be applied to restore the ARABIC behaviour.

Creating a format supporting the EURO symbol

To display or process a format supporting the EURO symbol, the terminal/emulation must be set in 8-bit mode and must use an ISO8859-15 compliant variant/font.

For more information, please follow the instructions given in section “Summary of 8-bit support” on page 316.

Converting a format to support the EURO symbol

To convert an existing format into a format supporting the EURO (EDF04F character set), a conversion tool is provided for which detailed information can be found in the file SYSDOC.IFG.083.CONV-8BIT.

1.3 Changes since the version 8.0

The following changes have been incorporated in the version 8.1 in relation to its predecessor.

Introduction of the dialog manager

The dialog manager for TIAM applications is a newly introduced feature as of FHS V8.1. The dialog manager (FHS-DM) and the UTM dialog extension (FHS-DE) are essentially similar in functionality (see page 22). IFG V8.1 supports the new features of FHS V8.1.

The following abbreviations are used in the manual to indicate which product component is involved in providing the given functionality.

FHS-DE	Only applies to the DE product component in relation to dialog extension.
FHS-DM	Only applies to the DM product component in relation to dialog extension.
FHS-DE/DM	Applies to every aspect of dialog extension, i.e. the DE as well as the DM product component.
FHS-noDE	Applies to FHS without dialog extension features.
full-FHS	Applies to the complete FHS product with and without dialog extension.
FHS-noDE/DE	Applies to FHS with dialog extension but without the DM product component.

Menu bars and pull-down menus.

A new feature to create and use menu bars and pull-down menus for full screens has ben introduced for FHS-DE/DM formats.

Variables

You can define variables for FHS-DM formats that can be used as control variables or to save the contents of fields. The defined variables may be used in multiple formats.
A facility to print variables is also offered.

Internal choice number

A new feature supporting internal choice numbers is offered for single-choice fields of FHS-DE/DM formats. In contrast to the external choice numbers, internal choice numbers can be assigned independent values.

Zero value suppression

Zero value suppression is an extension for the editing attributes of fields, i.e. zero can be replaced by a fill character. Note that the zero value suppression can only be used with FHS-DE/DM formats.

Messages

New message member names have been introduced for FHS-DE/DM formats. The fourth character can be a digit.

Lists

When creating lists using FHS-DE/DM formats, you can now split the “Column title” field arbitrarily and also insert a separator line only after the column titles. Special output fields (*NUMROW, *TOPINDX, *BOTINDX) can be likewise supported in the list title. Furthermore, a new facility to include scrolling information (control variables of type NUMROW, TOPINDEX and MODINDEX) in FHS-DM format has been introduced.

Length of field names

The restriction on the length of field names has been removed. Field names in FHS-DE/DM format can now have a length of up to 255 characters.

Preparation of FHS-DE formats compatible to IFG V8.0.

Starting with FHS V8.1, field names longer than eight characters are allowed. In order to avoid incompatibilities within existing FHS-DE applications, you can set the preparation for FHS-DE formats in the user profile to be compatible with IFG V8.0.

Help panels

Static support for the scrolling information field is a newly added feature for help panels, and you can now also define whether the box size of the help panel is to be extendable. The maximum size for a help panel is 999 lines. Furthermore, help on fields within help panels can also be provided (for FHS-DE/DM only).

Format display

The function for “viewing the image of a format” has been extended by a subfunction to “view the FHS-DE/DM features” (for FHS-DE/DM only).

Alternative PLAM library types

Instead of the default member types “M” and “S” for PLAM libraries, you can now also specify your own member types to save addressing aids when creating FHS-noDE and FHS-DE formats.

FHS-DOORS conversion

The definition of default values for FHS-DOORS conversion is a new feature (full-FHS).

Arabic/Farsi formats

Support for Arabic/Farsi formats has been newly introduced. Special processing by IFG is required. For Arabic/Farsi formats due to the right-to-left notation.

LMR, MLU and COBLUR objects are no longer supported.

1.4 README file

Information on any functional changes and additions to the current product version described in this manual can be found in the product-specific README file. You will find the README file on your BS2000 computer under the file name
`SYSDOC.product.version.READ-ME.E`.
The user ID under which the README file is cataloged can be obtained from your system administrator. You can view the README file using the `/SHOW-FILE` command or an editor, and print it out on a standard printer using the following command:

```
PRINT-FILE FILE-NAME=file-name,LAYOUT-CONTROL=PARAMETERS(CONTROL-CHARACTERS=EBCDIC)
```

2 What can the Interactive Format Generator do?

The Interactive Format Generator IFG simplifies the task of creating, modifying and managing formats (masks, maps) for the dialog between the terminal user and application program.

You work with IFG interactively, drafting the image of the format directly on the screen and can examine the result at any time. The dialog can be logged on the device assigned to SYSLST.

INFO mode allows you to request information at will on the IFG function you are currently using.

2.1 What is a format?

```
                                Request Queue
_____________________________________________________________________

Please choose an item and enter a command.


Your choice : _ 1. water
                2. beer
                3. milk
                4. cheese
                5. vegetables
                6. ice-cream
                7. sweets




_____________________________________________________________________
Your Command :
F1: Help        F2: Correct        F3: Quit     F4: Print
_____________________________________________________________________
```

Example of a DE/DM format

A format is simply a form that is displayed on a terminal. Just like the forms you are confronted with every day (application forms, tax returns, betting coupons etc.), a format is made up of “boxes” (called “fields”) that can be filled in by you (input field) or by an application (output field) and text that is part of the form itself (text fields). Formats are thus forms that are stored in a computer and displayed at the terminal as and when required.

A format also contains specifications relating to

– how a field is displayed on the screen (e.g. flashing mode),

– what you can enter in the field (e.g. numerics only),

– where the cursor is to be positioned when the format is displayed,

– the kind of terminal on which the format is to be displayed.

All IFG screens are also examples of formats.

If you wish create formats which conform to the Alpha Style Guide, you must use the dialog extension. To do this, enter YES for “dialog extension” in the screen “default values for format editing attributes” (see page 72).

You can then use your new formats only with FHS-DE or FHS-DM. Note that FHS-DM formats only use data transfer areas with separate attribute blocks and field contents (see section “Addressing aid specifications” on page 191).

If you do not wish to create FHS-DE formats, enter NO at dialog extension. Your formats will then be processed in the same way as formats of previous IFG versions (< V8.0).

When creating new formats or modifying existing ones which were created using IFG versions from V8 or later, you can specify YES for dialog extension. Note that this value cannot subsequently be reset to NO.

Dialog extension enables you to

– overwrite formats with boxes and assign global help texts
– define a menu bar
– define a status area
– define a top and bottom instruction area
– define a command area
– assign a key list to a format
– define a message area
– extend the addressing aids for the format attributes (basic attributes of the format, message identifier, cursor position)

on format level, and to

– create single- and multiple-choice selection fields
– create lists
– define new validation attributes (time, area, character list, or value list)
– assign help texts to the individual format fields
– define error messages for negative field checks

on field level.

2.1.1 FHS-DE and FHS-DM formats

UTM applications are supported by the FHS dialog extension FHS-DE, TIAM applications by the FHS dialog manager FHS-DM.

The UTM dialog extension (FHS-DE) and the TIAM dialog manager (FHS-DM) are essentially the same in terms of functionality; however, FHS-DE uses a data transfer area (addressing aids) to pass information (field contents, field attributes, ...) between the application and the FHS-DE format, whereas FHS-DM uses variables as a means of transferring information.

A detailed description of FHS V8.1 can be found in the FHS manuals “Format Handling System for openUTM, TIAM, DCAM” and “Dialog Extension for TIAM and SDF-P”.

2.1.2 Differences between FHS-DE and FHS-DM

The differences between FHS-DM and FHS-DE are summarized below:

– FHS-DE supports only UTM applications
FHS-DM supports only TIAM applications

– FHS-DE uses a data transfer area and addressing aids to communicate with the application.
FHS-DM uses S variables or dialog variables to save the contents of fields and to control the dialog (control variables).

– In FHS-DE, the content of a field is always a string.
For FHS-DM, the variable can be of type CHAR or any other type (e.g. FIXEDC, NUMS, etc.). The various input handling methods for different variable types (e.g. saving input, handling of input attributes) are described in detail in the FHS manual “Dialog Extension for TIAM and SDF-P”.

– Format attributes in FHS-DE can be modified via field control blocks (FCBs) of the data transfer area.
In FHS-DM, by contrast, format attributes can only be changed by using the dialog service ATTR.

– FHS-DM does not support partial formats and the field attributes for “alignment of the input field”, “input fill character”, “undefined values”, and “exit routines”.

– FHS-DE uses tables when generating addressing aids.
FHS-DM does not use any tables.

– FHS-DM provides a convenient interface for creating lists (control variable with FHS-DM default values).

– FHS-DM does not support a language extension in the names of help panels, key lists (also called key assignment tables), and message members.

2.1.3 Variables in IFG

The use of S variables or dialog variables in FHS-DM formats essentially means that the same variable can be used in several formats. This method is far more flexible than providing information in the data transfer area when a format is output. S variables are used in SDF-P procedures, dialog variables in programs.

Variables are classified into two categories: primary variables and control variables. Primary variables contain the field contents; control variables return specific information to the user or can be used by FHS-DM for format management. Control variables may typically contain information on the available choices of a single-choice field or the specific records to be output in an list.

Variable names can be defined in IFG for:

– fields accessible to the program.
  The contents of a field are stored in a primary variable with the same name as the name of the field (see section “Modify symbolic names of fields” on page 98). This means that every name defined for a field must be accessible to the program. This can be validated during format preparations (see section “Application library specifications” on page 195).

– The selection of single-choice fields.
  LOCK control variables inform FHS-DM whether a choice is available. If no LOCK variable was defined, the choice is always available (see section “Modify names of field control variables” on page 101).

– The scrolling information in a list.
  The control variables NUMROW, TOPINDEX and MODINDEX are used by FHS-DM to control the handling of output, input, and scrolling of lists (see section “Modify names of field control variables” on page 101).

– global control variables
  Global control variables can also be defined in a format (see section “Modify names of global control variables” on page 140).

Further details on these variables are provided in the FHS manual “Dialog Extension for TIAM and SDF-P”.

2.1.4 Syntax and naming conventions for dialog and S variables

FHS-DM can be used both in applications as well as procedures. Dialog variables are used by FHS-DM in applications; S variables are used in procedures.

When variables are defined for an FHS-DM format, the names of the variables must conform to the syntax of dialog variables and/or the syntax of S variables. You can have this verified by IFG by means of an appropriate setting in the user profile (see page 195).

The syntax for S variables and dialog variables is given below:

```
<SDF_NAME> = <STRUCT_NAME 1..20>
             [<#> ( <INTEGER 1..2147483647> | <STRUCTNAME 1..20>)
             [<.> <SDF_NAME> ]

<INDEX_DIALOG_NAME> = <DIALOG_NAME> <#>
                      ( <INTEGER 1..32767> | <DIALOG_NAME> )

<DIALOG_NAME> = <STRUCT_NAME> ( <.> <ALNUM_NAME> )*

<STRUCT_NAME> = <ALPHA> ( <ALNUM> | <-> <ALNUM> )*

<ALNUM_NAME> = <ALNUM> ( <ALNUM> | <-> <ALNUM> )*
```

Conventions:

<name>	Exactly one named string may be specified.
<->,<#>,<.>	Exactly one of the corresponding characters may be specified.
|	The vertical bar separates alternative values.
( )	Parentheses are used to group character sequences
[ ]	Square brackets indicate optional character groups or character sequences.
*	The asterisk is a wildcard that may be replaced by any string, even an empty string. In other words, the preceding character group or sequence may be repeated zero or any number of times.
<ALPHA>	Sequence of letters from A to Z.
<ALNUM>	Sequence of letters from A to Z or digits.
x .. y	x represents the minimum number and y the maximum number of characters permitted in the string or x is the minimum value and y the maximum value for INTEGER.

Examples of S variables and dialog variables

Examples	S variable	Dialog variable
ABC	YES	YES
ABC#A1	YES	YES
ABC.A1	YES	YES
ABC#1	YES	YES
ABC-DEF	YES	YES
A#32767	YES	YES
NAME.LONGER.THAN20.CHARACTERS	YES	YES
1BC	NO	NO
ABC#1A	NO	NO
ABC#A#1	NO	NO
ABC#*	NO	NO
ABC#*A#1	NO	NO
A#32768	YES	NO
ABC#1.A	YES	NO
ABC#A.B#1	YES	NO
NAMELONGERTHAN20CHARACTERSINONEGO	NO	YES
ABC.1A	NO	YES

The naming conventions for dialog variables comply with the syntax rules for S variables to a large extent, so the same format can be used both in SDF-P procedures as well as applications. Note, however, that SDF-P does not allow STRUCT_NAMES of more than 20 characters, and no multi-dimensional tables are permitted with dialog variables (i.e. only one index entry “#” is allowed in the name). Furthermore, the range of table indices for dialog variables is restricted from 1 to 32767.

Note also the naming conventions in list records. The records of a list constitute a table that is composed of the individual columns of the field. Each field can be identified via the column and the number of the record in which it is defined.

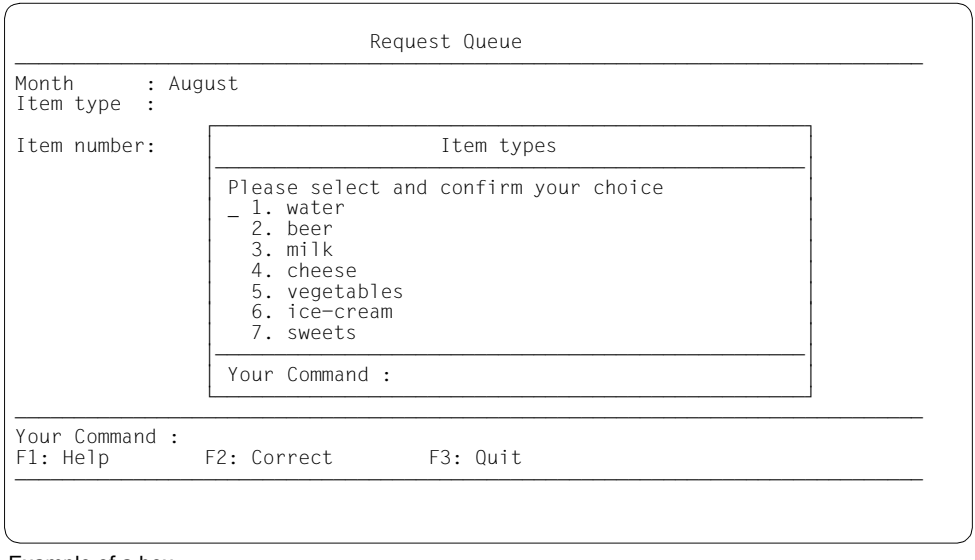
In IFG, the variable name assigned to each column is the name of the corresponding field in the first record of the list. This is because the first record of a list is the only record that IFG allows you to access.

In FHS-DM, you can reference the n-th field of a list via “FIELDNAM#n” and the entire corresponding column using “FIELDNAM#*”.

If you define a table name for SDF-P where multiple table references are allowed, you must notify FHS which index must be changed along with the record number of the list. An “*” will then be substituted for the corresponding index.

2.2 What is a dialog box?

A **dialog box** (box) is a format, which was activated from another format by means of a command or an action. The activating format can be partially or completely overwritten by the box. You can use boxes to output help texts, messages, or other formats with input and output fields or selection fields and lists. There are modal and modeless boxes. Message boxes can be either modal or modeless. All other boxes are only modal. A box is modal if the user cannot enter data in the underlying area while the box is open. A box is modeless if the user can continue working in the underlying area. However, input is only possible in fields that are visible.



Example of a box

2.3 What are menu bars and pull-down menus?

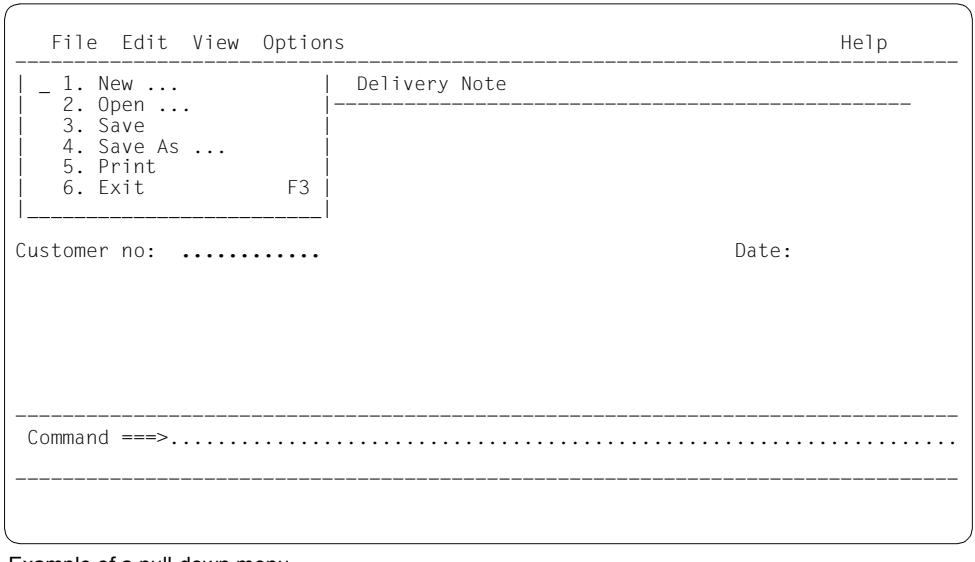
Menus are groups of options (e.g. functions, attributes, etc.) that are offered in combination on the screen. Only one function from a menu can be activated at a given time. The menu provides you with information on the options at your disposal when processing objects at the current stage of the dialog. Typical options include:

– saving a file

– searching for a particular object

The menu bar is a line at the top of a full screen. It is split into fields, called menu titles, which are always visible to the user in the menu bar. This makes the functionality of an application transparent to users and provides them with a quick means of orientation. The menu bar is always followed by a separator line. Each menu title of the menu bar is assigned a pull-down menu.

Pull-down menus are menus with options that are only made visible when the corresponding menu title in the menu bar is activated. They are displayed in the form of single-choice fields with one choice per line and without a text prompt.



Example of a pull-down menu

Such menus are created in IFG in a two-step process:

– Creation of the menu bar on creating the format (see section “Creating an FHS-DE/DM format” on page 72),

– Creation of the pull-down menu (see section “Steps performed after drafting the image” on page 79).

2.4 For which terminals can IFG formats be created?

IFG supports

– the 9750, 9755, 9758, 9763 data display terminals, IBM 3270 display terminals, and equivalent devices,

– the 9001, 9001-8931, 9002, 9003, 9004, 9011, 9012, 9013 and 9022 Printers, the PCL printers 9021 and 9022-200, and the 3287 printer (if it uses the LU1 protocol).

In addition to those mentioned above, other terminals are supported. A complete list can be found in the appendix.

IFG can be used on any of the said data display terminals, with the exception of the IBM 3270 display terminal and compatible devices. IFG is itself not executable on the 3270 display terminal; using IFG you can only generate and maintain formats for use on a 3270 display terminal. IFG takes full account, however, of the particularities of these terminals with regard to format creation and modification.

Existing formats for the 8161 and 8162 data display terminals can still be used; however, for IFG V6.0 or later, new formats can not be created for these terminals.

Note that most formats created by IFG as V8.1 are intended for BS2000 client PCs or PCs with FHS-DOORS.

2.5 In which programs can IFG formats be used?

1. FHS-noDE formats can be used in (addressing aids):
   - COBOL programs,
   - Assembler programs,
   - RPG programs,
   - PL/I programs,
   - Pascal programs,
   - C programs,
   - DRIVE procedures, and
   - Fortran programs.

   FHS-noDE formats work with any of the following access methods:
   - TIAM (Assembler, COBOL, RPG, and Fortran),
   - DCAM (Assembler, COBOL, and Fortran),
   - UTM (Assembler, COBOL, RPG, PL/I, Pascal-XT and, as of UTM V3.1, also with C, DRIVE, and Fortran).

   FHS-DE formats work with the following access method:
   - UTM (Assembler, COBOL, RPG, PL/I, Pascal-XT and, as of UTM V3.1, also with C, DRIVE, and Fortran)
2. FHS-DM formats can be used in all programming languages (COBOL, C++ etc.) that work with the TIAM access method.

The use of formats is documented in the following manuals:

- for UTM applications: in the FHS manual “Format Handling System for openUTM, TIAM, DCAM” and the “openUTM V5.2” reference manuals;
- for TIAM and DCAM applications in COBOL or Assembler: in the FHS manual “Format Handling System for openUTM, TIAM, DCAM”;
- for TIAM applications: in the FHS manual “Dialog Extension for TIAM and SDF-P”
- for programs written in RPG: in the “RPG3 (BS2000)” User Guide.

The formats are stored independently of the programs in a format library.

2.6 Support of 8-bit terminals

With IFG V7.1 or later, you can create and work with 8-bit formats These formats contain characters of a coded character set. A coded character set may contain up to 191 characters per character set, as opposed to 90.

8-bit formats can only be created on **8-bit terminals** (8-bit 9763, 9758 and 8-bit 9763). You must therefore make the appropriate terminal specifications in the current user profile. In addition, the XHCS product must be loaded.

The coded character set must be defined in the user profile (display attributes of the format). The usability of the character set and its compatibility with the display terminal are not checked until the character set is used. If a coded character set is not specified in the user profile, the default character set is used.

If you are using IFG on a **7-bit terminal** you can only create and use 7-bit formats. These formats only contain characters from the EBCDIC.DF.03 character set. IFG does not permit use of 8-bit formats on a 7-bit terminal but 7-bit formats can be used on 8-bit terminals.

Notes

– Characters which are not part of the EBCDIC.DF.03 character set are only accepted in the layout of the format, for key lists, messages, undefined values and for fill characters. Values that must also be known outside of IFG such as file names, passwords, format names, format versions, format characters and user profiles can only contain characters from the EBCDIC.DF.03 character set.

– Only certain types of 9758 and 9763 data display terminals can operate in 8-bit mode, namely 9758 M4 and 9763 D7. All others operate only in 7-bit mode.

– Please note that IFG uses two types of character sets. The first of these is the graphic character set which is created with ICE and can only be used on loadable 7-bit data display terminals (9763). The second type is the coded default character set which conforms to international standards. The two character sets cannot be used simultaneously.

– Please note that the character set of an 8-bit format cannot be changed after it has been generated.

– The “fast formatting” function is not possible for 8-bit formats.

2.7 What is a format library / What is a user profile?

Every format generated with IFG is stored in a format library.

With IFG V6 or later, formats and user profiles are stored in PLAM libraries and not in ISAM files, as was the case in earlier IFG versions. These format libraries may contain different member types (generated using IFG).

The following member types are possible:

– members of type U for user profiles (full-FHS)

– members of type F for formats (full-FHS), key lists (FHS-DE/DM), and message members (FHS-DE/DM)

– members of type R for prepared formats (full-FHS), prepared key lists (FHS-DE/DM), and prepared message members (FHS-DE/DM)

– members of type M or S for prepared formats in different languages (addressing aids; FHS-noDE/DE)

– members of type S for prepared FHS-DOORS formats (full-FHS)

– undefined member types as alternative types for addressing aids (FHS-noDE/DE)

The IFG functions to administer the format library (e.g. copy or delete) can only be applied on members of type U or F. In order to delete other member types, for example, LMS must be used. LMS can be used in any case for all member types.

Several users may access a format library at one time, providing they do not want to process the same format. The user profiles can also be shared, except in the case of the function “Modify user profile”.

Each format library must contain at least one own user profile in order to permit processing of a format. The user profile is a set of default values for

– work with IFG (e.g. characters that are used when drafting a format image),

– the attributes of the formats (e.g. the terminal type for which the formats are generated, whether or not dialog extension is used),

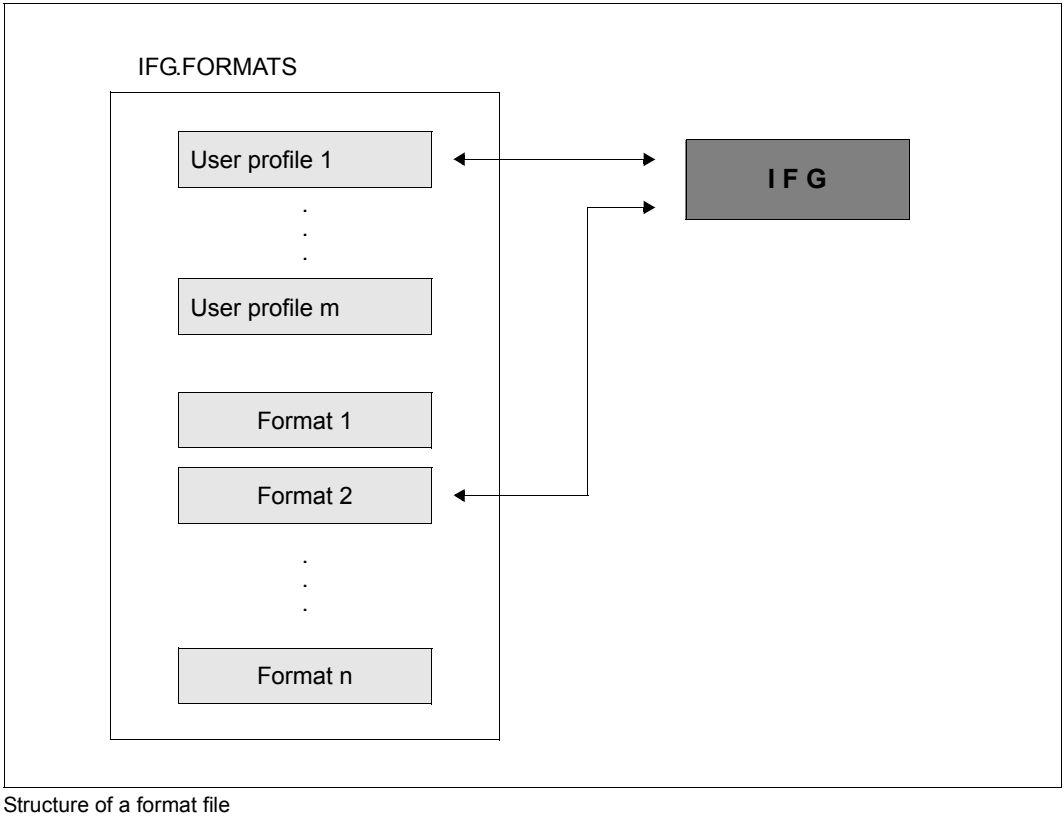
– the attributes of the fields of the format (e.g. all the fields on the screen that can be overwritten are displayed at full brightness and fields that cannot be overwritten at normal brightness),

– attributes relating to programming (e.g. the programming language in which the application program is written).

The user profile makes creating formats easier for you since you only need to specify values that differ from the defaults and at the same time it helps you keep the formats uniform.

IFG supplies a standard user profile that you can modify as you please. You are also free to change individual values from the ones in the user profile for each format.

A format library may contain several user profiles. You can switch between the various profiles with function 07: “User profile administration”.



Structure of a format file

2.7.1 What effect does the terminal type have on the use of formats?

Before you create your formats with IFG, you should also make sure that the formats that you are about to define can be used on all the desired terminals.

In addition to the terminal type, other important factors are:

– how the terminal is generated in PDN/BCAM and

– which access method is used to implement the format.

The terminal type is defined in the user profile.

```
I F G              VIEWING YOUR USER PROFILE USERPRO                  0704
                   DEFAULT VALUES FOR TERMINAL SPECIFICATIONS

 Format is to be suitable for use on the following terminal group (+ printer):
 8-Bit 9763                  : NO     9750 in multiple field mode : NO
 8-Bit 9763,9758             : NO     3270,9763,9755              : NO
 9763                        : NO     3270,9763,9755,9750         : NO
 9763,9755                   : NO     9763,9755,9750,8162         : NO
 9763,9755,9750              : YES    9763,9755,9750,816x         : NO


Default device when format is used:
 - display terminal                 - printer
   9763: NO    8162: NO               9001     : NO     9004: NO    9022: NO
   9755: NO    8161: NO               9001-8931: NO     9011: NO    PCL : NO
   9750: YES   3270: NO               9002/9003: NO     9012: NO    3287: NO
                                      9003     : NO     9013: NO

 If the default device is a printer, is it connected to
 - a printer controller: NO           - or a 9763, 9755, 9750: NO   Terminal?
                                         8161  :        8162:

Next image:                 SEND                For help           K3
To select another function: F1                  To abort function: K1
```

The screen shows the default values of the IFG user profile. The values with YES against them apply.

You must always define a standard terminal type.

If the standard terminal type is a printer, you must define the type of connection of the printer (e.g. printer control or data display terminal).

When outputting formats to printers, you must take into account the features of the respective printer.

If you select a group of terminals, you must note the features of the terminal types in the group. A format which is generated for a group of terminals only supports the features of the terminal with the lowest functionality. In other words, the more terminals there are in a group, the more restrictions apply (e.g. number of fields per line).
The format must be generated on a terminal which belongs to the group.

If the group selected is “8-bit 9763 and 9758 and 8-bit printer” or the “8-bit 9763 and 8-bit printer”, the formats generated are 8-bit formats. These formats are only permitted on 8-bit terminals and cannot be used in any other terminal group.
All other terminals are 7 bit terminals.

A 7-bit format can be output to both 7-bit and 8-bit terminals.

If you use the “fast formatting” function (default), the format is specially edited for the default terminal. FHS can then output the format to this terminal (It may also be possible for this format to be output to other terminals, see the table in section “Application library specifications” on page 195).

If you do not use the “fast formatting” function, FHS can output the format to all terminals in the defined group. If no DEVICE operand has been defined in the MDCBL macro (see the FHS manual “Format Handling System for openUTM, TIAM, DCAM”), FHS uses the default terminal type. If a DEVICE operand has been defined, this operand determines the type of terminal.

Notes

– A format generated using the IFG standard user profile (terminal group 9763, 9755, 9750 and printer), **cannot** be displayed on a 3270 display terminal. If you wish to convert such a format for a 3270 display terminal, you merely insert the image of this format into a new format with the function “create format”. Field attributes differing from the ones in the default profile and field names are lost in the process.

– If, for example, a 9763 data display terminal has been generated in your communication system as a “DSS-9750” (to find out consult your system supervisor), the display terminal will also be treated as a 9750 Data Display Terminal by UTM, TIAM COBOL or DCAM COBOL. You should remember this when you first create the format.

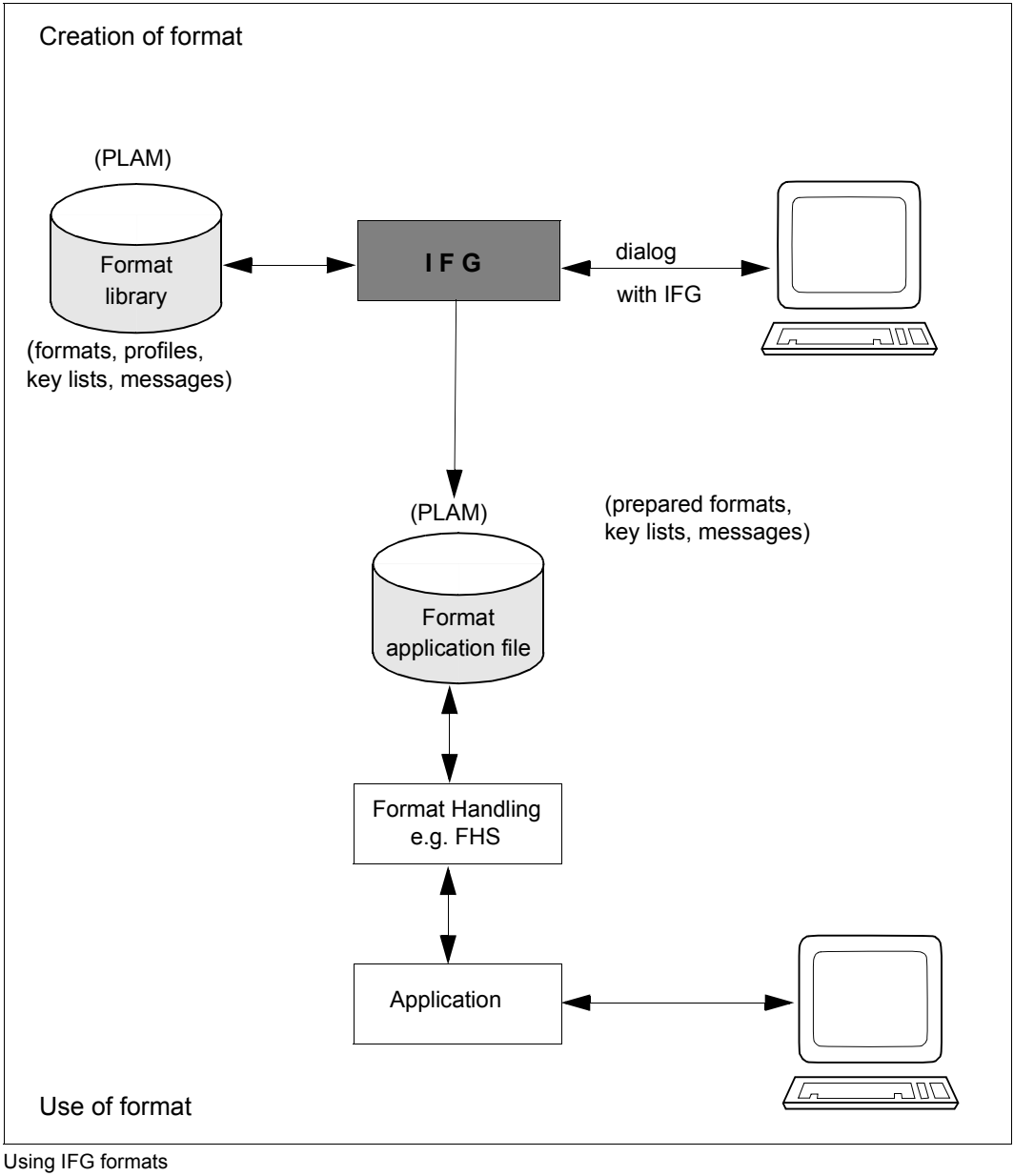
– ICE character sets are only supported by 7-bit terminals of type 9763. With all other terminals, the character sets are not converted.

– If you are working in multiple field mode, your terminal must be set to “Weight FHC as DCC” (a setting has to be changed in the controller of the data display terminal). Restrictions and further information on multiple field mode can be found in the appendix as of page 309.

– *N.B.* When a format is created, IFG recognizes the **actual** type of terminal regardless of the type generated. This is why it is possible to create a format for a 9763 terminal on a 9763 that has been generated as a 9750. This does not mean, however, that the format can also be displayed on a 9763 terminal generated as a 9750.

Possible changes in the terminal groups are given on page 134.

2.8 How do you use formats?



Using IFG formats

You create your formats with IFG interactively in the format library. Completed formats that you wish to use must be transferred to the format application file.

All members that are generated and handled by IPG can be stored in a separate library.

The format handler that is being implemented by the application program then accesses this format application file.

– Before each output the format handler structures the output message from the application program's output data and the desired format from the format application file in such a way that the desired image appears on the screen.

– After each input the format handler edits the input message so that the program only receives the desired data.

See also the "FHS V8.3A (BS2000/OSD)" and "openUTM V5.2" Reference Manuals. For "format preparation", see also chapter "Preparing a format for use" on page 113ff.

2.9 What are addressing aids?

You can address the fields of the format in the program symbolically by means of the field names (see also page 98).

To enable the fields of a format to be addressed symbolically in the data transfer (communication) area, IFG provides addressing aids for the programming languages Assembler, RPG, COBOL, PL/I, Pascal, C, DRIVE and Fortran. In the addressing aids the structure of the data transfer area is described by means of language elements in the relevant programming language. Names (field names: see page 98) are assigned to the corresponding statements of the programming language. A description of how addressing aids are used is provided in the FHS manual “Format Handling System for openUTM, TIAM, DCAM” and in the “openUTM V5.2” reference manuals.

Addressing aids generation with Unicode fields

User data transfer areas can come in different layouts with possible control over the global attributes, the field attributes and the field contents of the formats. The layout of global attributes and field attributes is not affected by Unicode, but field contents occupy twice as many bytes for Unicode fields. The table below summarizes how the content of a field is currently described in each of the programming languages for which Unicode is supported.

Programming language	Unicode
Assembler	<name> DS CL<length>
COBOL	<name> PIC X(<length>)

The table below describes how these descriptions will be modified in case of generation for Unicode.

Programming language	Unicode
Assembler	<name> DS CL<2*length> or when the CL length exceeds 256: <name> DS CL256 DS CL <2*length - 256>
COBOL	<name> PIC N(<length>)

Refer to the “Unicode in BS2000/OSD” introduction for a survey of the Unicode support in the BS2000/OSD as well as basic information on Unicode.

2.10 How are formats administered?

With IFG you can also

– copy formats

– delete formats

– print out formats.

An IFG function is available for each of these tasks.

Alternatively, you can perform these management tasks with the aid of the IFG utility routine **IFGFV**, either in batch mode or interactively from any data display terminal. IFGFV is described in the section starting on page 209.

2.11 How do you prepare a format for use?

The table below is designed to provide an overview of the individual IFG functions and at the same time serve as a guide to the manual. The structure of the table largely corresponds to the structure of the manual; the column “Page” contains the number of the page on which the description of the action within an IFG function begins.

Action	IFG function	Page
Generate format		51
– Create format file with user profile	07 User profile administration	54
– Draft image		59
– Define FHS-DE attributes		59
– Create pull-down menus	01 Create format	79
– Define choice fields		81
– Create lists		89
– Modify display attributes of the fields from the default		94
– Assign symbolic names for the fields		98
– Modify names of field control variables		101
– Modify input/validation attributes from default		103 and 107
– Modify editing attributes of the fields from default		109
Prepare format for use		113
– Store format and addressing aids in PLAM libraries		
– Create or update format application file	05 Format preparation	
– Create or update source program or macro library with addressing aids		
Display format		119
– Show image as during format application		120
– Show display attributes of fields		121
– Show attributes of an FHS-DE format		121
– Show symbolic names of fields	02 Display format	121
– Show names of field control attributes		121
– Show input/validation attributes of fields		121 and 121
– Show editing attributes of fields		121
– Show pull-down menu of the format		122
– Show terminal specifications		121
– Show display attributes of the format		123
– Show editing attributes of the format		123

Action	IFG function	Page
Modify or extend format		125
– Modify image and FHS-DE attributes of format		126
– Modify display attributes of the fields		131
– Modify symbolic names of fields		131
– Modify names of field control variables		132
– Modify input/validation attributes of the fields	03 Modify or extend format	136
– Modify editing attributes of the fields		136
– Modify pull-down menu of format		136
– Modify terminal specifications		138
– Modify display attributes of the format		136
– Modify editing attributes of the format		138
– Modify names of global control variables of format		140
– Adapt all field attributes to the user profile		132
Combine formats	04 Combine formats	141
Create tables	04 Combine formats	146
	01 Create formats	148
	03 Modify or extend formats	125
Key list administration		151
– Create key lists		152
– Display the index of key lists		158
– Display the key lists		152
– Modify the key lists		158 and 136
– Delete the key lists		152
– Print the key lists (SYSLST)		159
– Copy the key lists within the format library	09 key list administration	152
– Copy the key lists from another format library		152
Prepare key lists		152
		152

Action	IFG function	Page
Administer message members – Create message members – Create message – Display directory of the message members – Display message member – Modify message member – Delete message member – Print message member (SYSLST) – Copy message member within the format library – Copy message member from another format library – Prepare message members	10 Message management	161 165 166 171 162 162 162 162 162 162 162
Administer formats – Print to high-speed printer (SYSLST) – Delete in the format library – Copy within, and from another, format library	06 Print, delete, copy formats	175 176 178 178
Administer profile of the format library – Create a new user profile as the standard profile – Display the user profile of a format library – Modify the user profile of a format library – Copy the user profile of another format library – Delete the user profile – Select another user profile – Copy the user profile within the format library	07 User profile administration	179
Convert formats created with IFG V4.0 or IFG V5.0	08 Convert format file of former version	175
Explain functions	20 Explanation of IFG functions	49

2.12 Handling existing formats

Existing formats which were created with IFG Versions 4.0 and 5.0 must be converted before they are processed further with IFG V6.0 or later. For this purpose you must either copy the individual formats and user profiles into an existing format library (with function 06 “Copy”) or process them globally with function 06 “Convert format file of former version”.

The formats can be converted with IFG or with IFGFV. A description of IFGFV and an example of how existing formats are converted are provided in the section starting on page 209.

If you wish to use FHS-DE/DM attributes in older formats (< V8), you must enter “YES” for “dialog extension” in the “format editing attributes” function (page 138).

3 Working with IFG

The following components must be available in order to run IFG:

– the IFG format library,

– the FHS library MFHSROUT,

– the IFG module library, and

– the FHS-DOORS converter if required

IFG is called in BS2000 timesharing mode (communication application $DIALOG) with the command /START-IFG or /IFG. Please not that IFG is not executable under OMNIS if /OPTION BREAK-KEY=1.

IFG format library

The IFG format library contains the IFG messages and screens.

The default name of the IFG format library is SYSFHS.IFG.083.D for the German format library and SYSFHS.IFG.083.E for the English format library.
If this default name is cataloged under your own user ID or the default user ID $, this file can be automatically accessed by IFG.
When the default name is used, the language for messages and screens is defined by the BS2000 generation (MSGLPRI=ED/DE class 2 op).
The command /MODIFY-MSG-ATTRIBUTES TASK-LANGUAGE=D or E or *STD can be used for a subsequent assignment or reassignment.

The format library can, however, also be stored under any desired name under any user ID, in which case it must be assigned with the command /SET-FILE-LINK LINK-NAME= IFGMAPS,FILE-NAME=filename.

FHS library MFHSROUT

The library MFHSROUT is used by IFG for the output of formats. If this library is not cataloged under its default name $.MFHSROUT, it must be assigned with the command /SET-FILE-LINK LINK-NAME=MROUTLIB,FILE-NAME=filename before starting IFG.

IFG module library

The default name of the IFG module library is $.SYSLNK.IFG.083. If this name is used, IFG can be started with the command /START-IFG. If you have assigned a different name to the library, you must call IFG as follows:
/START-PROGRAM FROM-FILE=*MODULE(filename,IFG,RUN-MODE=*ADVANCED)

FHS-DOORS

If formats are also to be converted by FHS-DOORS, the module library FHS-DOORS-C (default name: $.SYSLNK.FHS-DOORS-LC.010) must always be assigned before starting IFG. The appropriate command for this is :
/SET-FILE-LINK LINK NAME=DOORSLIB,FILE-NAME=libname.

For compatibility reasons, IFG V8.3 can also be started with the command that was normally used for earlier versions, i.e.: /START-PROGRAM FROM-FILE=$.IFG (or $IFG). As a prerequisite, IFG and the IFG module library SYSLNK.IFG.083 must be installed under your own user ID or the default user ID $. However, you should use the start command /START-IFG as far as possible.

Examples

```
/BEGIN-PROCEDURE
/ASSIGN-SYSLST TO-FILE=#IFG83
/SET-FILE-LINK LINK-NAME=DOORSLIB,FILE-NAME=$.SYSLNK.FHS-DOORS-LC.010
/MODIFY-JOB-SWITCHES ON=11
/START-IFG
/MODIFY-JOB-SWITCHES OFF=11
/ASSIGN-SYSLST *PRIMARY
/END-PRCEDURE
```

```
/BEGIN-PROCEDURE
/ASSIGN-SYSLST TO-FILE=#IFG83D
/SET-FILE-LINK LINK-NAME=IFGMAPS,FILE-NAME=$.SYSFHS.IFG.083.D
/SET-FILE-LINK LINK-NAME=MROUTLIB,FILE-NAME=$.MFHSROUT
/SET-FILE-LINK LINK-NAME=DOORSLIB,FILE-NAME=$.SYSLNK.FHS-DOORS.LC.010
/MODIFY-JOB-SWITCHES ON=11
/START-PROGR FROM-FILE=*MODUL($.SYSLNK.IFG.083,IFG,RUN-MODE=*ADVANCED)
/MODIFY-JOB-SWITCHES OFF=11
/ASSIGN-SYSLST *PRIMARY
/END-PRCEDURE
```

For notes on “Logging”, see page 50.

Note that IFG cannot be run under OMNIS if `/OPTION BREAK-KEY=1` is set.

When IFG is started, it first accesses FHS and the IFG format library. The following return codes are supplied if no access is possible:

– If an incorrect version of the IFG format library is used

```
IFG0002: IFG-format and IFG-program ; program aborted
```

– If no IFG format library exists

```
IFG0003: IFG-format library not available; program aborted
```

– If the assigned file is not an IFG format library

```
IFG0004: IFG-format library is not a valid PLAM-B library
```

or

```
IFG0005: Error on accessing IFG formats; DMS: xxxx
```

– If FHS is not available

```
IFG006: FHS library not available
```

– If the FHS version is older than FHS V8.3

```
IFG007: Old version of FHS library
```

When IFG is called, it responds with the following main menu:

```
I F G                    INTERACTIVE FORMAT GENERATOR      VERS  V8.3A00    0001

Name of format library:

Function menu

01    Create format
02    View format
03    Modify, extend format
04    Combine formats
05    Prepare format
06    Print, delete, copy format

07    User profile administration
08    Convert format file of earlier versions (IFG V4.0 and V5.0)
09    Key list administration
10    Message administration

20    Keyword information

To select the desired function, mark it and press key:                    SEND
For help:                                                                 K3
To terminate I F G:                                                       F3
```

You must then specify a library name. This name must comply with the syntax rules for BS2000 file names. You will then be requested to set up a format library with a user profile (function 07), assuming no such library exists.
Note that you can also create an IFG library by using a function 08.

Functions 05, 06, 07, 08 and 20 do not output any formats and can therefore be executed from all data display terminals.

The IFG screens have the following structure:

– The header line contains the name of the function being used and the image number.

– In this area you can select a function (or in INFO mode, a keyword) from the proposed menu or enter data, e.g. in the image above you can mark and select one of the functions or enter a file name.

– The bottom part of the screen usually indicates what actions you can perform.

  You can “mark” (MAR) or “select” as follows:

  – Move the cursor to a character in the field you wish to select and press the **MAR** key. The marked field will then flash.

  – or overwrite one of the characters in the field to be selected with any character (e.g. a blank). A field marked in this way does not flash.

  Then press the SEND key.

  In selectable fields the text is always bright and italicized or underlined.

– The message line outputs the messages from IFG.

3.1 Conducting the dialog with IFG

You work with IFG in dialog steps. After outputting an image IFG always expects input from the user.
After entering any data and/or selecting a field you press one of the following keys:

SEND1 F1 F2 F3 K1 K2 K3

The meaning of keys SEND, F1, F2 and F3 is context dependent. The meaning of keys K1, K2 and K3 is constant.

Note

Depending on the data display terminal used, the SEND key is either the ↵ key or the ENTER (DÜ1) key.

The following functions can be performed at any time, regardless of the function currently in use.

K1 displays the last screen again in its unmodified state. This offers you a constant means of correcting or saving images. K1 is appropriate if

- a system or operator message has overwritten and thus obliterated the image.
- an image was incorrectly transmitted, or
- you selected the wrong option and you want your keyboard input to be ignored.
  Any incorrect input not yet transmitted can be corrected or deleted at will.

K2 aborts the current function.

You can then

- terminate work with IFGF3
- switch over to system command mode: F2
- return to the main menu: K3
- choose another subfunction: SEND
- continue working as before, if you pressed K2 inadvertently: F1

Note

“RESUME-PROGRAM” returns you to IFG from system command mode into IFG and back. IFG is continued from the point of interruption.

K3 switches on info mode.

An image appears explaining the current function and proposing relevant keywords. Each keyword appears bright and underlined.
If you are satisfied with this information, press SEND. This switches off info mode and takes you back to the current function.
To find out more about a keyword, mark it (with MAR or by overwriting) and press the SEND key.
An image telling you about this keyword then appears.
F3 takes you back to the current function from info mode.

You can also request an explanation of a particular keyword.

To do this, select function 20, “Keyword information” in the main menu. The following IFG screen will then appear:

```
I F G                        KEYWORD INFORMATION                          2001

IFG is described by means of keywords.
Each keyword is presented in an image.

   - Keyword groups for selection:


                      A - D

                      E - F

                      G - Q

                      R - Z


     To select the desired group:                        MAR and SEND

   - To view the IFG keyword index from the beginning:          SEND

   - To return to the function menu:                              F3
```

Mark the appropriate group of letters in the column. You will then be shown a series of keywords from which you can select the one you want. Each keyword is explained in a separate screen. From each screen you can select further keywords.
F3 takes you back to the function selection screen.

Note

In some functions IFG supplies preset default values of YES or NO. To change these values all you need to do is overwrite the preset value, YES with n (N, no, NO etc.) or the value NO with y (Y, yes, YES etc.). When different alternatives are available for selection (e.g. 9750 or 9755 in this case), only the required value needs to be overwritten with YES. IFG will automatically change the associated default YES to NO.

3.2 Logging your dialog with IFG

If you wish, you can have your dialog with IFG recorded on SYSLST. Before the EXEC $IFG call you enter the following system command:

```
/MODIFY-JOB-SWITCHES ON=11
```

To discontinue the log you deactivate the logging mode with the system command

```
/MODIFY-JOB-SWITCHES OFF=11
```

On one page IFG logs

– the output image and below it

– an image containing only the modifications made at the terminal.

Notes

– The listing is in uppercase only.

– When user formats are displayed, they are printed out the same way as after image drafting. In 8-bit formats with special characters you must check that the printer supports such characters. If not, printing may prove problematic.

– If you modify the image of the format, only the modified lines appear in the logging when you use the function “Display and Store the Format” (with F1).

– For input IFG also logs the key depressed (except the K2 key).

3.3 Specifying a format library before calling IFG

An existing format library can be specified before IFG is called. You do this by issuing the following system command:

```
/SET-FILE-LINK LINK-NAME=USERMAPS,FILE-NAME=filename
```

For filename, enter the name of the format library.

The name of the format library is then displayed in the IFG main menu after IFG has been called.

Note

When you terminate IFG, the LINK name USERMAPS is not automatically released. You can release the LINK name using the following system command:

```
/REMOVE-FILE-LINK LINK-NAME=USERMAPS
```

4 Creating a format

Before you generate a format, you must decide whether or not you wish to use dialog extension. The attributes of the format generated depend on this decision. If you wish to use the dialog extension, you must enter “YES” for “dialog extension” in the current user profile (for the format editing attributes). This determines that all newly generated formats use the dialog extension for as long as this user profile is active. Unless otherwise specified dialog extension is not used. By default, if you generate a format which uses the standard user profile of a newly created format library, this format is processed like a format of previous IFG versions. Likewise, you cannot use dialog extension if you use format files with user profiles of older IFG versions. The default settings in the user profile can be modified simply by overwriting. The format generated contains the values of the current user profile.

The following table gives an overview of the possible values for the generation of formats. Format generation without dialog extension is compared to format generation with dialog extension.

Values for dialog extension = NO	**Values for dialog extension = YES**
IFG returns format 0101 into which you enter a format name and possibly a format version, a password and a format identification.	IFG returns you format 0101 into which you enter a format name and possibly a format version, a password and a format identification. Format 0102 is then output. At this stage you may enter the following values: – the help panel assigned to the format – the box size – the menu bar – the status area – the top instruction area This is followed by format 0103, in which you enter the following values: – the bottom instruction area – the prompt for the command area – the size of the area for the key list – the key list allocated to the format – the size of the message area
Format 0110 format is output You can then draft the image of your format. When doing so, you must differentiate between the following types of fields: – text fields – entry fields – numeric entry fields (entry fields num) – output fields	Format 0110 is output You can then draft the image of your format. When doing so, you must differentiate between the following types of fields: – text fields – entry fields – numerical entry fields (entry fields num) – output fields – single-choice selection fields – multiple-choice selection fields – lists

Values for dialog extension = NO	Values for dialog extension = YES
You may then modify the attributes of your fields	You may then modify the attributes of your fields and – introduce new validation attributes – define messages for negative field checks – define help panels for each field When executing an application, it is possible to overwrite the format with a box when FHS-DE is outputting.

Enter the name of the format library in which the format is to be stored (current format library) and continue reading from page 54. If this library already exists, select 01 and press SEND. Then read from page 55.

4.1 Create format library with user profile

A newly created library must contain a user profile. You can create a standard user profile or copy a user profile from an existing format library. You do this by marking **07** and pressing SEND.

```
I F G                        USER PROFILE ADMINISTRATION                    0701

Name of format library: IFG.FORMATS

Do you want to  _view_____  the directory of all user profiles?

Name of current user profile  : USERPRO
Password for user profile     :

Do you want to  _view_____  or _modify____  or _delete____  the user profile?

Do you want to  _view_____  or _modify____  part of the user profile?

Do you want to  _set up___  the I F G standard profile?

Do you want to  _copy_____  an existing user profile?
from the format library:
Name of old user profile         :
Password of old user profile     :

To activate current user profile         SEND     To abort function          K2
To select the desired function:  MAR and SEND    (To delete user profile:    F1
To select another function:              F3                      For help:    K3
```

To create the standard user profile, enter a user profile name and, if desired, a password, mark “set up” in the displayed image and press SEND. IFG will then issue the message

```
5702: YOUR FORMAT LIBRARY HAS BEEN FURNISHED WITH THE IFG STANDARD PROFILE
```

to confirm that the standard profile has been created.

It is a good idea to use the name USERPRO for the standard user profile, since IFG always searches first for USERPRO. If no user profile with this name exists, you are requested by IFG to enter a name.

F3 returns you to the function menu.

4.2 Format identification

Mark **01** in the main menu and press the SEND key. If a user profile exists, you will be presented with the screen that follows:

```
I F G                          CREATING A FORMAT                              0101

Name of format library:  IFG.FORMATS

Do you wish to view the directory first?  If so, mark and press SEND.

Format name  ....  : delivery                                (up to  8 characters)
Format version ... : 01
                                                              (only if required)
Password  ....... :                                           (only if required)
Format identifier :                                           (only if required)



To draft the format image:  SEND
To select another function: F3                                  For Help: K3
```

– Enter a format name.

  The following characters are allowed in the format name: Letters, digits, $, and #. However, the $ and # must not be at the eighth position.
  The length of the format name for different format types is as follows:

FHS-noDE formats	max. 8 characters; the eighth character is a language code for UTM
FHS-DE formats	max. 8 characters; the eighth character is a language code
FHS-DM formats that are not help panels	max. 8 characters; the language code is not supported for FHS-DM
FHS-DM help panels	max. 7 characters; (see Note)

– If you do not specify a version, IFG creates this format with the version “@”, i.e. with the highest possible version. You can find out which names and which versions already exist in the current format library by viewing the format library’s directory (mark “view”).

The following characters are permitted in the format version: letters, digits, @, $, #, ., -. However, the . and - must not be in the first position.

Note

Help panels are assigned to different IFG objects (FHS-DE/DM only):

Formats that are not help panels	see page 72
Single-choice fields	see page 82
Multiple-choice fields	see page 86
Lists	see page 90
Fields	see page 94
Key lists	see page 154
Messages	see page 165

In order to enable FHS-DE to automatically insert the language code as the eighth character, the name of a help panel must not exceed seven characters. If the name is shorter, it is padded with “#”.

Key lists are restricted to 7 characters for the same reason (see page 154).

```
I F G                          DIRECTORY                                    C003
     Format library   IFG.FORMATS

Format    Version                   Identifier        Mod.  Date      Time
-------------------------------------------------------------------------------
ADDRESS   @                                           0001  04-12-10  12:18:06
EXAMPLE   V60A                                        0001  04-11-15  13:49:32
CUSTOMER  01                                          0003  04-11-11  08:12:09
DELIVERY  01                                          0001  04-11-11  08:27:45
DELIVERY  02                                          0001  04-12-10  14-04-54
SAMPLE    @                                           0007  05-04-01  09:33:51
ACCOUNT   A                                           0001  05-06-07  16:27:22
TAB       @                                           0002  04-12-10  13:41:07
TABLE     @                                           0010  04-12-10  13:59:04
_________
_________
_________
_________
_________
_________
******** End of list. To repeat from the beginning press SEND
                                                              Other info: F2
To return to the current function without selection:     F3   For help:   K3
```

Additional information is displayed when F2 is pressed.

```
I F G                          DIRECTORY                                    D003
     Format library   IFG.FORMATS

Format    Version                  Identifier       User ID  Acct.no. Password
-------------------------------------------------------------------------------
ADDRESS   @                                         USER1    ACCOUNT1 No
EXSAMPLE  V60A                                      USER2    ACCOUNT2 No
CUSTOMER  01                                        USER2    ACCOUNT2 JA
DELIVERY  01                                        USER1    ACCOUNT1 No
DELIVERY  02                                        USER1    ACCOUNT1 No
SAMPLE    @                                         USER2    ACCOUNT2 No
ACCOUNT   A                                         USER1    ACCOUNT1 No
TAB       @                                         USER2    ACCOUNT2 YES
TABLE     @                                         USER1    ACCOUNT1 No




******** End of List. To repeat from the beginning press SEND
                                                     Other Info.    F2
To return to the current function without selection:   F3      For help       K3
```

Key:

Format	Name of the format. The formats are listed alphabetically.
Version	Version of the format; “@” stands for the highest possible version. Formats with identical names are displayed in the order of their version numbers.
Identifier	Format identifier, if you specified one during format creation or in the function “modify or extend”.
Mod.	Modification counter for the format. This is set to 000 at creation or when copied. When a format is modified or extended and subsequently stored, the modification counter is incremented by 1.
Date and Time	specify when the format was created or last modified.
User ID and	ID and account number under which you worked on the
Acct. no.	format.
Password	states whether you have protected the format with a password.

You can page in the directory using SEND. F3 takes you back from the directory.

If the character string “**locked**” appears in a line after the format name, another user is currently accessing the format. In this case, IFG cannot provide any further information on this format. For your format you can optionally specify

– a version number; if no version number is specified, IFG proceeds as follows:
  – when a format is created, the highest possible version “@” is assigned to the format;
  – when a format is to be modified or viewed, the format with the specified name and the highest existing version is used.
– a password, which must be specified whenever the format is to be modified with IFG (protection against inadvertent or unauthorized modification, copying, deletion or printing)
– a format identifier that identifies the format more precisely.

SEND now takes you on to image drafting.

4.3 Draft image

By now you can already draft an image using very simple techniques. Once you have acquired a certain proficiency in the use of IFG, you will be able to make use of additional aids to simplify the drafting. This section is therefore subdivided further to allow you to proceed in steps. To begin with you will manage with the basic function; then, as you gain experience, you can make use of the additional aids.

4.3.1 Simple image drafting

The following chapter describes simple image drafting without dialog extension.

The window at the center of the screen shows the first 10 lines of the image you wish to draft. There is a column display above these 10 lines which simplifies image drafting.

```
I F G               DRAFTING AN IMAGE FOR FORMAT  DELIVERY                0110
                    (For help: K3)     VERSION  @
      Input field              > Repetition character        Delete line
    # Input field, numeric       Start-of-field char.        Repeat line
    @ Output field             " Alignment

----+----1----+----2----+--  Line 001 to line 010   --+----6----+----7----+----8
....................................................................................
....................................................................................
....................................................................................
....................................................................................
....................................................................................
....................................................................................
....................................................................................
....................................................................................
....................................................................................
....................................................................................
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
To check and edit your input:        SEND
To insert blank line or image of format                To store and display:  F1
(version:                          )                   Window fwd. 05 lines:  F2
              at line 01 :              MAR and SEND   Window back 05 lines:  F3
To switch to large window :             MAR and SEND   To abort function:     K2
```

When drafting an image you need to distinguish between the following field types:

text fields, which are not supplied with data by the application program, but contain fixed text that is output when the format is displayed. You just write the text where you want it to be.

input fields, which are fields in the format where the terminal is supposed to enter something. These are accessible to the application program. You define input fields using the field definition character shown before “input field” in the image (default:_).

input fields num, which only allow numeric input at the data display terminal (device-dependent), namely numerics and the following characters: period (.), comma (,), plus (+), minus (-), asterisk (*), and slash (/); all the other input keys are locked). Otherwise they are the same as regular input fields.
The field definition character has the default value #.

output fields, which are fields supplied with data by the application program and then displayed on the screen. They are protected against overwriting. The field definition character has the default value @.

Blanks and characters not contained in EBCDIC.DF.03 should not be used as field definition characters.

If you are using an 8-bit data display terminal and have set the terminal group “8-bit 9763 and 9758” or “8-bit 9763” in the current user profile, the format generated is an 8-bit format. This format then uses the character defined in the display attributes of the format. This means that special characters may also be entered. Further information is given on page 316.

The field types differ in their display attributes on the screen (e.g. bright/unprotected, normal brightness/protected), their input/validation attributes and their editing attributes.

For these field attributes IFG initially enters default values.

Notes for the 3270 display terminal

– adjacent fields must be separated by at least one blank or NUL character,
– successive lines must be separated by at least one blank or NUL character,
– the last column (column 80) in the last line of the format must always be left free,
– no fields for which inputs are possible should be in column 1 or line 1.

```
I F G               DRAFTING AN IMAGE FOR FORMAT  DELIVERY                0110
                    (For help: K3)     VERSION  @
      Input field               > Repetition character       Delete line
    # Input field, numeric        Start-of-field char.       Repeat line
    @ Output field              " Alignment

----+----1----+----2----+--  Line 001 to line 010   --+----6----+----7----+----8
                             Delivery note.................................
................................................................................
Company.........................................................................
Name:     -------------------------.....................................................
Street:   --------------------------....................................................
ZIP code: #####......... ........................................................
City:     ------------------------.............................................
................................................................................
Customer No:     ############                           Date:   ----------......
................................................................................
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
To check and edit your input:        SEND
To insert blank line or image of format                To store and display:  F1
(version:                          )                   Window fwd. 05 lines:  F2
               at line 01 :             MAR and SEND   Window back 05 lines:  F3
To switch to large window :             MAR and SEND   To abort function:     K2
```

Notes

- You must not use field definition characters in text fields (they are interpreted).
- The maximum format size is defined in the user profile (max. lines x columns).

Any time during image drafting you can

- check your input with **SEND**

```
                                       .
                                       .
                                       .
----+----1----+----2----+--  Line 001 to line 010   --+----6----+----7----+----8
                             Delivery note
................................................................................
Company
Name:     -------------------------
Street:   --------------------------
ZIP code: #####
City:     ------------------------
................................................................................
Customer No.:  ############                             Date:   ----------
................................................................................
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
                                       .
                                       .
                                       .
```

When error messages appear, the cursor indicates the line or field affected.

– move the window forward with **F2**.
  You can overwrite the default value “05”.

– move the window back with **F3**.
  You can overwrite the default value “05”.

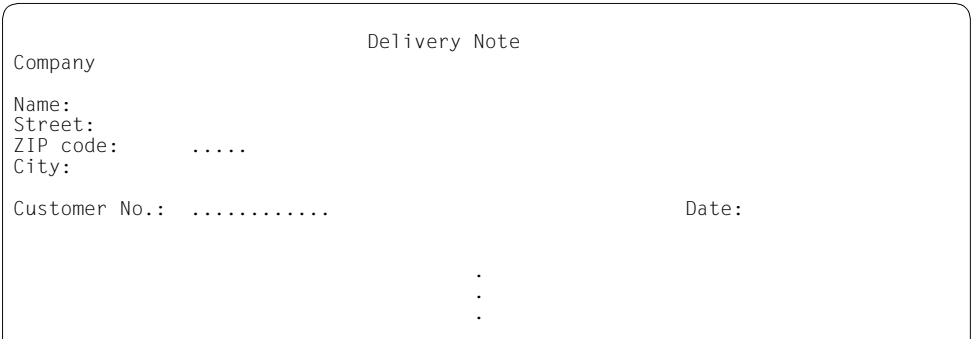
– switch to the large window with **MAR SEND**,
  i.e. you have the entire screen in front of you (except line 24, which is reserved for IFG messages). This will enable you, after some practice, to create formats enable you, after some practice, to create formats more easily that cover the whole screen. Depending on the SIDATA attributes, you may also use marks for the large window which divide the whole screen into blocks. These blocks are 5 lines long and 10 columns wide. You can revert to the small window any time you want with F2.

– insert blank lines or images of formats from the same format library with **MAR SEND**. When images are inserted in a format the character set of the format inserted must be compatible with the character set of the format to be generated. When you insert images of formats, IFG does not take the field names and field attributes of the old format (unlike IFG function 04 „Combine formats“).

How do you store the format?

Press the F1 key.

IFG will then store the format in the current format library and display the image of the format as it will appear when it is used.



Input fields	are displayed fill characters (default blank filled). They are therefore not visible on the 9750 data display terminal, for example.
Input fields num	are displayed with fill characters (default NIL characters).
Output fields	are displayed filled with the character you used in the format definition (default @).

The table below shows which characters (default values) are used for the input and output fields when drafting an image, as well as how these characters are output when the formats are used.

Field type	Characters	
	at drafting	at output
Input	_	blanks
Input num	#	.
Output	@	@

You can try filling out the format, i.e. writing in the input fields, to test their position and length. You can position the cursor anywhere you please within the format, provided “free cursor movement” has been selected in the user profile.

For format display see also page 120.

Now press SEND

The format is stored and you can select one of the proposed functions.
SEND will take you back to image creation allowing you to correct the image or extend it further.

or F3

The format is stored and you are taken straight back to image creation.

What happens if you press K2?

– You can switch on system command mode; from system command mode you can return to IFG with `/RESUME PROGRAM` and continue working where you left off.

– You can terminate format creation with SEND or IFG with F3.

– Using K3 you can go back to the main function menu.

– When you enter SEND, F3 or K3, any part of the format not yet stored is lost.

Once you have finished creating the image and have stored the format, you may decide, by selecting one of the options in the IFG screen, to

– return again to image creation,

– modify display attributes of fields,

– assign symbolic names for fields,

– modify input and validation attributes for fields,

– modify editing attributes for fields,

– create the next format, or

– select another function.

4.3.2 Additional aids for image drafting

Using various special characters you can

– create input and output fields of a particular length (REPETITION CHARACTER)

– create fields of the same type one after the other with no gap in between (START-OF-FIELD)

– justify fields (e.g. in headings, tables)

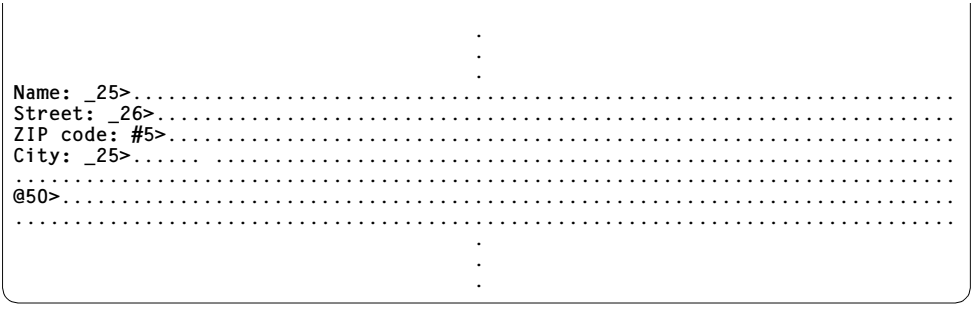
– delete (remove) lines not producing an empty line

– repeat lines already defined (REPEAT LINE)

As the default option no special characters are defined for the REPEAT LINE, START-OF-FIELD, and DELETE LINE functions. You can define characters for these functions in the user profile (see page 183).
Like the characters for field definition, you cannot use the special characters in the text fields.
No special character is required for inserting (blank) lines. You do this by specifying the line number and marking “Blank line” in the bottom section of the screen.

REPETITION CHARACTER

Enter the desired field definition character, immediately followed by the repetition factor and the special character for REPETITION CHARACTER (the default option is >). Note that subsequent text in this line is shifted. If the space available in this line is not sufficient, an IFG message to this effect is output in the bottom line of the screen.



After you have pressed SEND, the following screen is displayed:

```
                                        .
                                        .
                                        .
Name: -------------------------
Street: ---------------------------
ZIP code: #####
City: -------------------------
.....................................................................................
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
.....................................................................................
                                        .
                                        .
                                        .
```

START-OF-FIELD (the special character < was selected)

With the start-of-field character you can generate fields of the **same type** immediately following each other. During image drafting the start-of-field character is not removed. It counts as the first character of the new field.

In text fields the start-of-field character is replaced in the final image by a blank. This blank is retained in the field defined by the start-of-field character.

A start-of-field character at the beginning of a line or directly following a text field defines the beginning of a new text field. The end of this text field is taken to be before the next input/output field or at the end of the line. If no text is entered in the field, it is filled with blanks.

Notes for the 3270 display terminal

– Start-of-field characters are only permitted between text fields. They merely serve to separate text fields and are converted into blanks. The start-of-field character does not count as part of the next field.

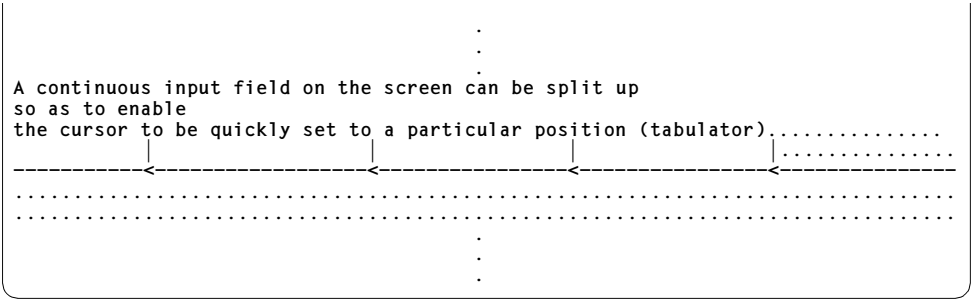
– Adjacent fields of other types must be separated by at least one blank or a NUL character.

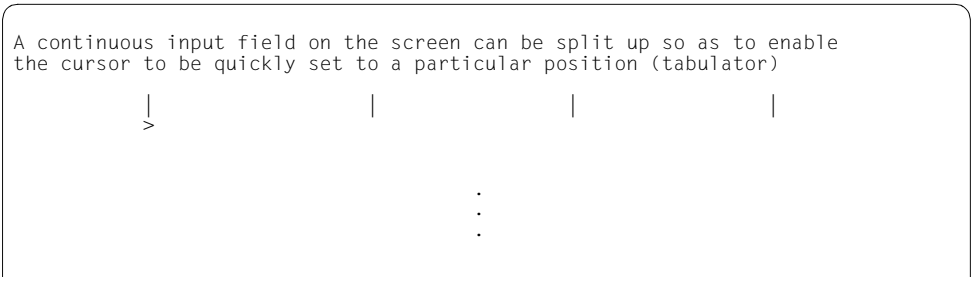
```
I F G               DRAFTING AN IMAGE FOR FORMAT  DELIVERY              0110
                    (For help: K3)     VERSION  @
      Input field                > Repetition character       Delete line
    # Input field, numeric         Start-of-field char.       Repeat line
    @ Output field               " Alignment

----+----1----+----2----+--  Line 001 to line 010   --+----6----+----7----+----8
With the start-of-field character individual<words<in a text can be............
modified, e.g. emphasized.       .........................................
................................................................................
................................................................................
                                      .
                                      .
                                      .
```

You can, for example, highlight the field (or text fields) marked with the start-of-field character using the function “Modify display attributes of the fields” (Bright=YES), or make the field accessible to the program with the function “Modify symbolic names of the fields”.

```
I F G           DISPLAY ATTRIBUTES OF FORMAT  DELIVERY              0305
                (For help: K3)      VERSION  @
                                    Field type:  Text field       Length:
006

Bright          : YES         Blanked        : NO         Printable  : YES
                              Flashing       : NO         Cursor     : YES

Underline/italics: NO         Character color: 0    (01234567)
Inverse         : NO          Character set  : 0    (*STD    )

--------------------------  Line 001 to line 010   ----------------------------
With the start-of-field character individual words in a text can be
modified, e.g. emphasized.

                                      .
                                      .
                                      .
```

You can use the tab key to position the cursor directly at the places where the start-of-field character has been entered. The cursor jumps from field to field, with each field beginning with a start-of-field character.





Combining a start-of-field character with a repetition character

If a start-of-field character immediately succeeds an input/output field, a repetition factor can be entered after it with the repetition character. The new field is of the same type as the preceding field.

ALIGNMENT (default is ”)

Using the special character for alignment (the default option is ”), you can e.g. center fields, right-justify them, etc. IFG divides up the line as follows: the number of blanks up to the end of line is added to the number of alignment characters, and then divided by the number of alignment characters. The result is the number of blanks that replace the alignment character.

```
                                        .
                                        .
                                        .
"Table"..........  ..........................................................
..............................................................................
------------------------------------------------------------------------------
"""|"|"|"|"...................................................................
..............................................................................
..............................................................................
                                        .
                                        .
                                        .
```

After you have pressed SEND, the format is displayed as follows:

```
                                        .
                                        .
                                        .
                                     Table
..............................................................................
------------------------------------------------------------------------------
                              |          |          |          |
..............................................................................
..............................................................................
                                        .
                                        .
                                        .
```

DELETE LINE (the special character % was selected)

The special character will only work if it is entered in the first column of the line.

```
                                        .
                                        .
                                        .
Name: -------------------------         First name: _20>
%irst name: --------------------
..............................................................................
Street: -----------------------------
ZIP code: #####
City: -------------------------
..............................................................................
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
 .............................................................................
                                        .
                                        .
                                        .
```

After you have pressed SEND, the format is displayed as follows:

```
                                        .
                                        .
                                        .
Name: -------------------------         First name: --------------------
.....................................................................................
Street: ------------------------------
ZIP code: #####
City: -------------------------
.....................................................................................
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
 .....................................................................................
                                        .
                                        .
                                        .
```

REPEAT LINE (the special character + was selected)

The special character will only work if it is entered in the first column of the line.

```
                                        .
                                        .
                                        .
                                    Table
.....................................................................................
-------------------------------------------------------------------------------------
                                |           |           |           |
+....................................................................................
+....................................................................................
+....................................................................................
+....................................................................................
.....................................................................................
.....................................................................................
                                        .
                                        .
                                        .
```

The definition character for “repeat line” is entered at the beginning of each line. The format is displayed as follows by pressing SEND:

```
                                        .
                                        .
                                        .
                                    Table
.....................................................................................
-------------------------------------------------------------------------------------
                                |           |           |           |
                                |           |           |           |
                                |           |           |           |
                                |           |           |           |
.....................................................................................
.....................................................................................
                                        .
                                        .
                                        .
```

The sample format may appear as follows:

```
                              Delivery Note

Company
Name:
Street:
ZIP code:       .....
City:

Customer No:     ............                                   Date:

_______________________________________________________________________________
 Item No.  |                              | Quan-  | Unit    | Total
           |                              | tity   | Price   | price
_______________________________________________________________________________
  .....    |                              | ...... | ....... | ..............
  .....    |                              | ...... | ....... | .............. €
  .....    |                              | ...... | ....... | .............. €
  .....    |                              | ...... | ....... | .............. €
_______________________________________________________________________________
                                            Total:            ............... €
                                            Sales tax:         .............. €
                                                              ________________
                                            Total:            ............... €
```

4.3.3 Creating an FHS-DE/DM format

To create an FHS-DE/DM format, you must enter “YES” for dialog extension in the current user profile (for format editing attributes) This means that all newly generated formats will use dialog extension as long as this user profile is active.

```
I F G        VIEWING YOUR USER PROFILE USERPRO                        0706
             DEFAULT VALUES FOR FORMAT EDITING ATTRIBUTES

Account is to be taken during format application of field alignment
                 in input:YES                    in output: YES

Representation of arithmetic fields
            decimal separator: .            digit separator: .
Representation of the time
                 with seconds: YES               separator: :
Representation of the date
sequence of day, month, year:  YMD (Y,M,D)        separator: .
              year, two digits: YES              four digits: NO

Dialog extensions required? : YES

Representation of undefined values
                   any char.       arithmetic    alphabetic      date/time
    character:            .               .             .               .
    hexadecimal:         00              00            00              00

Next image:                    SEND                            For help:    K3
To select another function:    F3                    To abort function:    K2
```

After you have defined the editing attributes of the formats, the screen for creating a format is output.

```
I F G                          CREATING A FORMAT                              0101

Name of format library:  IFG Formats

Do you wish to view the directory first?  If so, mark and press SEND.

Format name  ....  : Delivery                              (up to  8 characters)

Format version ... : @
                                                              (only if required)

Password  .......  :                                         (only if required)

Format identifier  :                                        (only if required)




To draft the format image:   SEND
To select another function:  F3                              For help:  K3
```

Enter a format name and, if applicable, a version no., a password, and a format ID and then press SEND. The following screen to create an FHS-DE format will then be output.

```
I F G                  EDITING THE FHS-DE/DM-PANELS: DELIVERY            0102
                       (For help: K3)       VERSION: @

                            Panel attributes:

Appears in a Box?     : NO              Box size: 22x076 (lines x columns)
Is it a help panel?   : NO              Related key list      :
-Scrolling information:                 Related help panel    :
-Extendable box size? : NO              Uses separation lines?: YES   char: -

                               Panel header:

Menu bar?             : YES             (start of field character: $)
   File.$Edit.$View.$Options                                              $Help
Panel title?          : YES              size : 1 lines
Delivery Note

Top instruction       : NO               size : 0 lines



To check and edit your input:   SEND
To process the panel trailer:   F1              To select another function: F3
```

In this function you can:

– define whether or not the format appears in a box. If it does, the box can partially or completely overwrite the format from which it was activated.

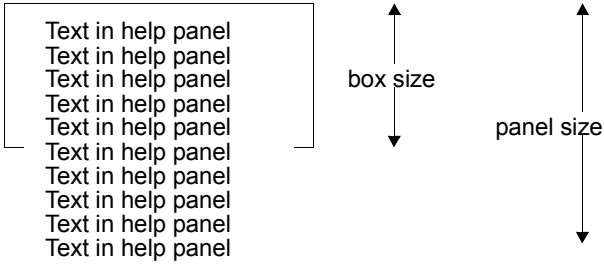
– define whether the format is to be a help panel. In a help panel, neither an input field nor an output field can be defined in the working area of the format (no choice fields, no lists, etc.). A help panel must always be defined as a box.

Note

The help panel must always be defined as a box.

Note that when you create or modify the image of a help panel, you can only process as many lines as the number of screen lines present in a full screen, since scrolling is not possible in this screen. Lines that extend beyond this limit can only be processed in the partial format.

– define the size of the box. Simply enter the number of columns and lines available for fields within a box. This size does not include the frame characters of the box; these are inserted by FHS-DE/DM.
Maximum values: (screen height -2) lines by (screen width -4) columns.
In formats which are not help panels, the specified size represents the maximum size available for the format. In help panels, it is possible to shift the text vertically. This size thus specifies the size of the box and not that of the format. As a result, you may create a format which has more lines than the box (up to a maximum of 999 lines).



Note that if you draft a format for a terminal group with the terminal 3270, the maximum number of columns you may use is the specified block size less two columns. These two columns are required for the output of your format. There is one column each on the left and right hand side of your format.

Note

All lines have the same number of columns. Unused space within a column is right-justified and added with blanks. This causes the entire width of the box to be filled. Consequently, if the box width is subsequently reduced, the original choice will not fit in the box and possibly require the number of choices per line to be reduced.

– define text to be used as a prompt for scrolling information. The scrolling information appears in the line below the format header. It indicates that only a part of the help panel is displayed in the box. The optional text that you can enter in the “Scrolling information” field precedes the actual scrolling information, i.e. the direction characters (´+´ and ´-´). The “+” and “-” indicate the position of the invisible information in relation to the visible section of the help panel.

– If you do not define any prompt, the default value from IDHSCRL is made available by FHS-DE/DM.

– define whether the help panel is to be displayed in a box with a fixed or extendable box size. If you use an extendable box size, the maximum possible number of lines from the help panel will be shown.

– specify the name of the key list (max. 7 characters) assigned to the format.

– specify the name of the help panel assigned to the format. You cannot assign a help panel to another help panel.

– define whether separator lines are to be output.

  Separation lines are always output when:

  – after the format header, if one has been defined

  – before the command or key list areas if such areas are output

  – before the message area

– define any line separation character you want.

– specify whether a menu bar, and thus the pull-down menus, are assigned to the format. Note that formats with a menu bar and pull-down menus are not displayed as a box.

  The menu bar is split into fields called menu titles. It is always followed by a separator line consisting of the defined separator character.

  The start of each menu title is defined by the start-off field character that you have defined in your user profile. The menu title ends with the last non-blank character that precedes the next menu title or at the end of the line. At least three character positions must be left free at the start and the end of the line, and the individual menu titles must be separated by at least two characters. The three characters at the start and the end of a line are protected and cannot be modified. This “three-character rule” should also be observed for formats that are smaller than the full screen.

– define whether a format receives a title. The maximum size of the title is two lines (one line is the default). The title is centered when output in the status area of the format.

– define a top instruction.
instructions are integral elements of a format that show the user how to interact with the format. Instructions can occupy a maximum of two lines. They remain visible even when the working area is shifted.

  Note that instructions can only be used with help panels.

After you have defined the attributes and header for your format, you can press the F1 key and proceed to the next screen to define a “Panel trailer”.

```
I F G                EDITING THE FHS-DE/DM PANEL  : DELIVERY              0103
                        (For help: K3)     VERSION: @



                                  Panel trailer:

Bottom instruction?      : NO          size : 0 lines


Command line?            : YES         global help :
Command:

Fkey area?               : YES         size : 1 lines

Message area?            : YES         size : 2 lines



To check and edit your input: SEND            To process the panel header: F2
To draft the format image:    F1              To select another function : F3
```

You can switch back to the previous screen at any time by pressing F2 in this screen.

This function allows you to

– define a bottom instruction. Instructions are integral elements of a format that show the user how to interact with the format. Instructions can be up to two lines in length. They remain visible even if the working area is shifted.

  Note that instructions can only be used with help panels.

– define whether the format contains a command line and whether a global help panel has been assigned to this command line. The command line can occupy up to one line. In terms of information content, global help lies between extended help, which is supplied for a complete format, and field-related help that only provides information on a single field.

- define a text as an input prompt for the command area. The command area consists of an input field, which enables the user to control the application by directly entering commands.
  The command area is generated as follows:
  - a blank
  - the prompt text
  - a blank
  - the command input field (its length is determined by other elements)
  - a blank in the last column
- reserve an area of up to two lines for the function key text. In this area, FHS-DE/DM outputs the function key number (Fxx: and Kxx:), followed by the corresponding function key text (see section “Key list administration” on page 152).

  When you press the function keys, the assigned character strings are treated as if they were entered as commands in the command area.
- define a message area and specify its size (maximum three lines). A message area is an area that can be accessed by FHS-DE/DM. When formats are used, this area is used for the exchange of messages between the application program and FHS-DE/DM.

If you press F1, screen 0110 (“Drafting an image for format”) is output, and you can draft your format as before (see page 54ff). If you press F1 while in screen 0110, the format you created is output with the inserted attributes.
Note that several different special characters exist for creating single-choice fields, multiple-choice fields, and lists.

```
   File   Edit   View   Options                                          Help
--------------------------------------------------------------------------------
                              Delivery Note
--------------------------------------------------------------------------------
Company -
Name:
Street:
ZIP Code:      .....
City    :

Customer no:  ............                                   Date:





--------------------------------------------------------------------------------
 Command ===>.................................................................
--------------------------------------------------------------------------------
```

F1, F2: If the working area of your format exceeds the number of lines available in the box (in help panels only), the working area can be scrolled forward and backward by pressing the F2 and F1 keys respectively.

SEND, F3: see page 63

4.3.4 Steps performed after drafting the image

After you have finished drafting the image for your format and saved the format, the next screen will automatically appear.

```
I F G                    CREATING THE FORMAT:  DELIVERY                  0104
                                 VERSION:  @


Name of format library: IFG.FORMATS

Do you wish to edit

-  the pull-down menus                            of the Formats?

-  the display attributes                       of the format fields?
-  the symbolic names                           of the format fields?
   the names of the control variables           of the format fields?
-  the input/validation attributes (part I)     of the format fields?
-  the input/validation attributes (part II)    of the format fields?
-  the editing attributes                       of the format fields?



To return back into last function: only SEND
To select the desired function:MAR and  SEND   To select another function: F3
To create the next format:              F2                      For help: K3
```

You can now select one of the displayed functions by marking it and pressing SEND. If you simply press SEND, you will be returned to the last function that was called.

Note that the functions to edit/create:

– the “pull-down menus” and “input/validation attributes (part II)” can only be used for FHS-DE/DM.

– the “names of control variables” is only available for FHS-DM formats.

When you have selected the function to “edit the pull-down menu”, the following screen will appear.

```
I F G       EDITING THE PULL-DOWN MENUS OF FORMAT: DELIVERY              01OZ
                                        VERSION: @


Name of format library: IFG.FORMATS

Mark the menu title of the pull-down menu to edit:
     File  Edit  View  Options


Do you wish to edit

-  the image                                      of the menu?

-  the display attributes                         of the menu fields?
-  the symbolic names                             of the menu fields?
-  the names of the control variables             of the menu fields?
-  the input/validation attributes                of the menu fields?
-  the editing attributes                         of the menu fields?


To return back in to last function:only SEND
To select the desired function:MAR and  SEND    To select another function: F3
To modify the format field attributes:  F2                          Forhelp: K3
```

You can now mark the menu title of the pull-down menu to be edited, mark one of the displayed functions, and press SEND. The currently selected menu title is highlighted.

After you have selected the function to edit the pull-down menu, the function for editing a single-choice field (see page 88) is called. You can then create/edit the pull-down menu in that screen just like a single-choice field. Note, however, that you cannot define an input prompt character and that the number of choices per line (one choice per line) cannot be modified.

Furthermore, you should note that

– the pull-down menus use the same general attributes as the format (e.g. the same extended character set).

– the pull-down menus are output by FHS in a box that encloses all fields of the pull-down menu and includes one blank each at the right and left margins. The maximum size of the pull-down menu is restricted only by the fact that the associated box must be fully visible on the screen without covering the menu bar of the format.

– the menu titles in the screen for “editing the pull-down menu of a format” are output without the ICE character set even if one or many such character sets are being used.

The function for creating/editing menu attributes is the same as the function for editing format attributes; however, only a subset of the attributes of format fields are available.

4.4 Selection fields

There are single and multiple-choice selection fields.

A single-choice selection field contains a set number of entries (at least one), which represent mutually exclusive alternatives. Of these, the user can select either one or none. A multiple-choice selection field contains a set number of entries (at least one), which are not mutually exclusive. Of these, the user can select several, one, or none.

You can only use choice fields if you are working with dialog extension.

Before you can create a choice field, you must define a special character for the choice field in the user profile. This special character then appears in the functions “Creating images” and “Modifying images” as additional field definition characters above the window. If you enter this special character in the first column of a line in the working area of the format, and then press SEND, the top left hand corner of the choice field is defined. The choice field is inserted below the line which was marked by the special character. Any existing fields are moved downwards. This and subsequent lines are then reserved for the selection field. If you specify several field definition characters, these are processed in descending order.

If you wish to create a choice field, IFG checks if there is sufficient space available. If not, a warning is output.

4.4.1 Single-choice fields

The character “%” was selected in the user profile in order to define a single-choice selection field. When this character appears in the first column, it marks a single-choice selection field.

```
I F G                DRAFTING AN IMAGE FOR FORMAT  DELIVERY                0110
                     (For help: K3)     VERSION  @
   _ Input field               > Repetition character      - Delete line
   # Input field, numeric      < Start-of-field char.      + Repeat line
   @ Output field              " Alignment
   % Single choice field       & Multiple choice field     ^List
----+----1----+----2----+--   Line 001 to line 010  --+----6----+----7----+----8

%









----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
To check and edit your input:                   SEND
To insert blank line  or  image of format                To store and display:  F1
(version:                          )                     Window fwd. 05 lines:  F2
                at line 01 :            MAR and SEND     Window back 05 lines:  F3
To switch to large window  :            MAR and SEND     To abort function:     K2
```

If you press SEND, the following screen appears. You can then specify a prompt, the position of the prompt, the name of the related global help, and the number of choices for each line.

```
I F G                EDITING A SINGLE CHOICE FIELD                        OKO3
                     (FOR HELP: K3)
Prompt text ..............: Comics:
Prompt location ..........:   (1.Above 2.Before)
Global help ..............:
Amount of choices per line : 2
 Nr.                                Text
  1 Asterix
  2 Micky Mouse
  3 Lucky Luke
  4 Superman
  5 Porky Pig
  6 Donald Duck



To edit the internal choice numbers:      MAR and SEND
To insert a choice before choice nr.      MAR and SEND
To delete the choice nr.                  MAR and SEND        Forward: F2
To store the choices:       F1                               Backward: F3
To abort function:          K2                  To check your input: SEND
```

– The prompt is the text which appears before or above the input field and requests the user to enter a choice. If the choices are not to begin in the first column of your format, enter blanks before the text for your prompt.

– The position of the prompt defines where the prompt will appear. The prompt can be located above or before the choice. Depending on your choice, specify the number “1” for “above” and “2” for “before”.

– Global help is a help panel which you can assign to the single-choice field (max. 7 characters, see page 62).

– If you wish to have several choices per line, each of these choices is output in a separate column.

Example

```
Position of prompt: before

Number of choices per line:  3

Comics: _ 1 Asterix            2 Micky Mouse          3 Lucky Luke
          4 Superman           5 Porky Pig            6 Donald Duck
```

After you have specified all of this information, you can enter the various choices of the single-choice selection field. If you define a choice identifier comprising only one freely-selectable character, the length of your input field is also only one character.

If you wish to insert an additional choice between two existing choices, mark “insert”. Then specify the identifier of the choice before which you wish to insert, and press SEND.

If you want to delete a choice, mark “delete” and enter the identifier of the choice to be deleted in the appropriate input field. Then press SEND.

If you wish to specify an internal choice number for each of the choices you have defined, mark “edit” and press SEND. The following screen will then be displayed.

```
I F G       EDITING INTERNAL CHOICE NUMBERS OF SINGLE CHOICE FIELD        OKO5

        Ext Int    Ext Int    Ext Int    Ext Int    Ext Int    Ext Int
          1   1  |   2   3  |   3   6  |   4   5  |   5   4  |   6   2
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |
                 |          |          |          |          |

To store the numbers: F1                          To check your input: SEND
To abort function:    K2                                    For Help: K3
```

The screen shows the list of external choice numbers with the corresponding internal choice numbers for each of the defined choices.

The internal choice number is a value that is passed by FHS to the underlying application when that choice is made in a single-choice field. By default, the internal and external choice numbers have the same values; however, you can set the internal choice numbers to some other value if desired. This would enable an application running in another language, for example, to reorder the choices in the choice fields without needing to change a single line of code.

When you press the F1 key, the choice numbers are stored and you are returned to the screen for “editing a single choice field”. If you press F1 in the screen for “editing a single choice field”, you return to the screen “Drafting an image for format”. This screen displays the single-choice field in accordance with your specifications for generation. If there is not enough space for the choice field to be output, a message is issued.

```
I F G                DRAFTING AN IMAGE FOR FORMAT  DELIVERY                0110
                     (For help: K3)     VERSION  @
   _ Input field                 > Repetition character      - Delete line
   # Input field, numeric        < Start-of-field char.      + Repeat line
   @ Output field                " Alignment
   % Single choice field         & Multiple choice field     ^ List
----+----1----+----2----+--  Line 001 to line 010   --+----6----+----7----+----8

Comics: _ 1 Asterix                          2 Micky Mouse
          3 Lucky Luke                       4 Superman
          5 Porky Pig                        6 Donald Duck









----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
To check and edit your input:        SEND
To insert blank line or image of format                To store and display:  F1
(Version:                          )                   Window fwd. 05 lines:  F2
                  at line 01 :          MAR and SEND   Window back 05 lines:  F3
To switch to large window :             MAR and SEND   To abort function:     K2
```

In single-choice selection fields, there is only one input field.

All entries made on one of these single choice lines are ignored in this format except:

– when entering the character for the function “single-choice selection field” in the first column of the first line of the single-choice selection field. After this entry has been made, the screen “Creating a single-choice selection field” is output. You can now modify the single-choice selection field.

– when entering the character for the function “delete line” in the first column of the first line of the single-choice selection field. When you input this character, the entire single-choice selection field (along with the relevant choices) is deleted.

Note

The same special character must be used to modify an existing choice field and to create a new one. If you wish to insert a new single-choice selection field before an existing one, you must first insert a blank line before the existing choice field. The new choice field can be inserted above this blank line.

A field control variable (LOCK variable) can be defined for every choice of a single choice field (see section “Modify names of field control variables” on page 101). This variable is used by FHS-DM to determine whether a particular choice is available. If no such variable is defined, the choice is always available.

4.4.2 Multiple-choice fields

The “&” character was selected in the user profile to define a multiple-choice selection field.

This character marks a multiple-choice field if it is positioned in the first column of a line. If you enter this character in the screen “Drafting an image for format” and press SEND, the following screen is output. You can then specify a prompt, the location of the prompt, the name of the associated global help, and the number of choices for each line.

```
I F G                EDITING A MULTIPLE CHOICE FIELD                    0K04
                     (FOR HELP: K3)
  Prompt text ..............: Comics:
  Prompt location ..........:   (1.Above 2.Before)
  Global help ..............:
  Amount of choices per line : 2

  Nr.                                  Text
  01.
  02.
  03.
  04.
  05.
  06.
  07.
  08.
  09.
  10.

To insert a choice before choice nr. --   MAR and SEND
To delete the choice nr. --               MAR and SEND        Forward: F2
To store the choices:       F1                               Backward: F3
To abort function:          K2                  To check your input: SEND
```

The prompt is the text which is located before or above the input field and which prompts the user to enter a choice. If the choices are not to begin in the first column of your format, enter blanks before your prompt.

The prompt location defines the location where the prompt will appear. The prompt text can be positioned above or before the choice. Depending on your choice, specify the number “1” for above and the number “2” for before.

Global help is a help panel which you can assign to the multiple choice field (max. 7 characters; see page 55).

If you wish to have several choices per line, each choice is output in a different column.

Example

```
Location of prompt: above

Number of choices per line:  3

Comics:

        _  Asterix          _  Micky Mouse        _  Lucky Luke
        _  Superman         _  Porky Pig          _  Donald Duck
```

Once you have entered all of the above information, you can specify the various choices of the multiple-choice field.

Should you wish to insert an additional choice between two existing choices, mark insert. Then specify the identifier of the choice before which you wish to insert and press SEND.

If you want to delete a choice, mark “delete” and enter the number of the choice to be deleted. Then press SEND.

If you press F1, you return to the screen “Drafting an image for format”. This screen displays the multiple-choice field according to your generation. If there is not enough space for the choice field to be output, a message appears.

```
I F G               DRAFTING AN IMAGE FOR FORMAT  DELIVERY                0110
                    (For help: K3)     VERSION  @
   _ Input field                > Repetition character      - Delete line
   # Input field, numeric        < Start-of-field char.      + Repeat line
   @ Output field               " Alignment
   % Single choice field        & Multiple choice field     ^ List
----+----1----+----2----+--  Line 001 to line 010   --+----6----+----7----+----8

Comics: _ Asterix                          _ Micky Mouse
        _ Lucky Luke                       _ Superman
        _ Porky Pig                        _ Donald Duck









----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
To check and edit your input:        SEND
To insert blank line or image of format                To store and display:  F1
(Version:                          )                   Window fwd. 05 lines:  F2
                 at line 01 :          MAR and SEND     Window back 05 lines:  F3
To switch to large window :            MAR and SEND     To abort function:     K2
```

In the multiple-choice field, there is an input field for each entry.

All entries made on one of these multiple-choice lines is ignored in this format except:

– when entering the character for the “multiple-choice field” function in the first column of the first line of the multiple-choice field. When this entry has been made, the screen “creating a multiple-choice field” is output. You can now change the multiple-choice field.

– when entering the character for the “delete line” function in the first column of the first line of the multiple-choice field, the entire multiple-choice field, (along with the relevant choices) is deleted when you enter this character.

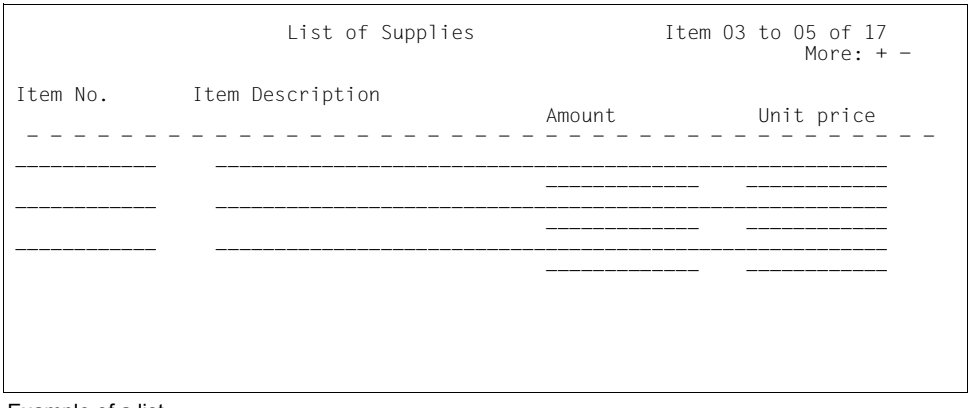
Note

The same special character must be used to modify an existing choice field and to create a new one. If you wish to insert a new multiple-choice field before an existing one, you must first insert a blank line before the existing choice field. The new choice field can be inserted above this blank line.

4.5 Lists

A list is a series of data records, where each data record consists of one or more data fields. The total number of records in the list is determined by the application using the format. The fields which correspond to each other must be assigned the same attributes and length in each data record, so that a list is defined exactly like a table. Lists can only be created when you are using dialog extension. You may create only one list per format.

Because lists can only be moved vertically, the total width of a data record must not exceed the screen width. However, you can split a data record over a number of screens.



Example of a list

Before you can create a list, you must define a special character for the list in the user profile. This special character then appears above the window for the “draft image” and “modifying image” functions. The special character “^” was selected to indicate lists. If you enter this special character in the first column of a line and then press SEND, the top left-hand corner of the list is defined. This line and the lines below it are then reserved for the list.

The generated list is inserted after the line which is marked by the special character. Existing lines are moved down. If there is not enough space to insert the list, a warning is issued.

```
I F G               DRAFTING AN IMAGE FOR FORMAT  DELIVERY                 0110
                    (For help: K3)     VERSION  @
   _ Input field              > Repetition character      - Delete line, ...
   # Input field, numeric     < Start-of-field char.      + Repeat line
   @ Output field             " Alignment
   % Single choice field      & Multiple choice field     ^ List
----+----1----+----2----+--  Line 001 to line 010  --+----6----+----7----+----8

^









----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
To check and edit your input:        SEND
To insert blank line or image of format                 To store and display:  F1
(Version:                          )                    Window fwd. 05 lines:  F2
               at line 01 :           MAR and SEND   Window back 05 lines:  F3
To switch to large window :           MAR and SEND   Line 01
```

After you have defined the start of the list, the following IFG format is output for creating your lists.

```
I F G                         EDITING A LIST                              0K02

 Scrolling information: More:                             Separation lines: NO
 Global help          :                           after column titles only: NO
 Number of records per screen: 03                     Separation character: -

 List title: (Scrolling information positioning character: ! )
                                List of supplies         Items @@ to @@ of @@
                                                                       !

 Column titles:
Item No.          Item description
                                       Amount                 Unit price

 Record layout:
 ____________     _________________________________________________________
                                             _____________    ____________

 End of Data marker:
                            ****  End of Data  ****

To store the list definition:F1                    To check your input:   SEND
To abort function          :K2                               For help:     K3
```

– Scrolling information indicates that there are list items that cannot be displayed on the screen. In the “scrolling information” field, you must enter a text to be placed before the actual scrolling information (the direction characters “+” and “-”). The “+” and “-” are displayed in an output field of four characters. They indicate the direction in which the

hidden information is located in relation to the visible segment.
Further details on scrolling information can be found in the “FHS V8.3A (BS2000/OSD)” manuals. The scrolling information is output together with the list title. Its position is defined by the positioning character “!”. In other words, when you define the list title, you should enter the positioning character at the position where the scrolling information is to appear later on.

– The global help is a help panel that you can assign to the list (max. 7 characters; see page 62).

– In the “separation line” field, you define whether separation lines are to be inserted; in the field “after column titles only”, you define whether the separation lines are to be inserted only after column titles or also between the individual records. The characters to be used for this separation line is defined in the “separation character” field. Any character may be used as a separation character.

– The “number of records per screen” field defines how many records are to be output at the same time. IFG automatically outputs the number of records that you request in this field.

– The “list title” field can be used to define special output fields in addition to text fields. These output fields provide the user with information on the current status of the list, e.g. an index of the first and last record shown and the number of records in the list. The contents of these special output fields are restricted and cannot be modified by your application. FHS uses these fields to provide the user with some global information on the list. The information to be shown is obtained by FHS via field names that are assigned to these special output fields (see section “Modify symbolic names of fields” on page 98). The following field names are used:

  “*TOPINDX”: output field for scrolling information; contains the index of the first record currently displayed.

  “*BOTINDX”: output field for scrolling information; contains the index of the last record currently displayed.

  “*NUMROW”: output field containing the total number of records defined in the list.

  Every output field in the list title must be assigned one of the above names.

  In FHS-DM formats, the values for *TOPINDX and *NUMROW are obtained from the values of the control variables TOPINDEX and NUMROW (see section “Modify names of field control variables” on page 101) if these variables exist. If no such variables are available, FHS-DM assumes the default values that are maintained by FHS-DM itself. In FHS-DE formats, the values for *TOPINDX and *NUMROW are obtained from the values of the corresponding addressing aids (there are no default values).

  Note that the special characters for defining output fields, i.e. the repetition character, start-of-field character, and the alignment character are used in the definition of the list title as usual. All other special characters are treated as normal characters.

– In the “Column title” field, you can assign a title to each of the columns in your list. This field includes only text fields, where each character other than the start-of-field is interpreted as a text character. The titles can extend over a maximum of three lines. Note that the “Column title” field can be split in any way in IFG V8.1. In IFG V8.0, by contrast, this field was internally split into the same number of titles (text fields) as the fields in the “Record layout lines”.

– In the field “Record layout”, you can specify the usual special characters for “Input field”, “Input field, numeric”, “Output field”, “Repetition character”, “Start-of-field”, and “Alignment” (output in the screens “Draft an image for format” and “Modify the image of format”).
  All fields of the first record can be assigned attributes such as names, output attributes, and input or validation attributes. These attributes are then taken over by corresponding fields of other records.

– In the field “end of Data marker”, you can specify any text. This text is used by FHS to indicate that there are no further records below the displayed text.

When you press F1, you are returned to the screen “Drafting an image for format”. The list you have generated will be output in that screen in accordance with your specifications. If there is not enough space for the output of the list, you will not be able to return to this screen, but you will receive a message in this case.

```
I F G               DRAFTING AN IMAGE FOR FORMAT  DELIVERY                 0110
                    (For help: K3)     VERSION  @
   _ Input field             > Repetition character      - Delete line, ...
   # Input field, numeric    < Start-of-field char.      + Repeat line
   @ Output field            " Alignment
   % Single choice field     & Multiple choice field     ^ List
----+----1----+----2----+--   Line 001 to line 010  --+----6----+----7----+----8

                         List of supplies              Article @@ from @@ to @@
                                                                   More: @@@@
 Item No.       Item Description
                                      Amount             Unit Price
  _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _  _ _ _ _ _ _ _ _ _ _ _ _ _
 ____________     _________________________________________________________
                                      _____________    ____________
 ____________     _________________________________________________________
                                      _____________    ____________
----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
To check and edit your input:        SEND
To insert blank line or image of format                 To store and display:  F1
(Version:                          )                    Window fwd. 05 lines:  F2
              at line 001 :             MAR and SEND   Window back 05 lines:  F3
To switch to large window :             MAR and SEND   To abort function:     K2
```

In this format, any entry made in one of these list lines (list title, column title, or list fields) will be ignored except:

– for the input of the special character for “List” in the first column of the list title. This entry causes the format “Editing a list” to be output and allows you to edit the list.

– for the input of the special character for “Delete line, ...” in the first column of the list title. Entering this character deletes the whole list.

Note

To modify an existing list and to create a new one, the same special character must be used.

4.6 Modify display attributes of fields

The display attributes of a field determine how it is displayed on the screen. This allows you to

- display fields with different intensities (fields may also be dark, i.e. blanked, e.g. for entering passwords)
- display fields which contain Unicode characters.

  Checks are performed for this function and when editing field input/validation attributes to avoid inconsistencies:

  - Only fields of the type "Any character" may be UNICODE, since the other types (arithmetic, alphabetic, date and time) can only contain characters in a strict 7-bit range.
  - The "NUM lock" and Unicode options are incompatible since "NUM lock" is a hardware option which prevents you from typing characters other than 0 through 9.
  - (In)valid characters/value lists and Unicode are mutually excluded since you can't type such Unicode characters or values.
  - Definition of tables in IFG must ensure that the usage of the UNICODE field option is identical in all rows of the tables.
  - Definition of groups in IFG must ensure that the group doesn't contain a mix of UNICODE and non-UNICODE fields (see also field attribute group FIELD LENGTH in FHS).
  - Fields containing fixed text (text not accessible by program, markable fields, fields with fixed initial text) must not be Unicode strings since it is not possible to type Unicode texts in IFG.
  - Special FHS-DE/DM fields – single/multiple choice fields, list scrolling information and command areas – must not be Unicode.

  Refer to the „Unicode in BS2000/OSD“ introduction for a survey of the Unicode support in the BS2000/OSD as well as basic information on Unicode.
- highlight fields (flashing, underlining/italics, inverse)
- specify whether or not a field can be printed out on a hardcopy unit (when “printable”: YES” is specified even blanked fields can be printed, i.e. are visible)

– display fields in different colors. The colors are represented by numbers. The default values are listed below:

  0 other attributes such as “flashing” are converted to color attributes by color stations.

  1 red

  2 green

  3 yellow

  4 blue

  5 magenta

  6 cyan

  7 white

– display fields using various ICE character sets. If you wish to use an ICE color character set for a field, you must set the “IFG” character color to “7” (white). Note that ICE character sets are only supported by 7-bit 9763 terminals.

– position the cursor on a particular field as the default option during output. “Cursor: YES” can only be specified for one field. When “cursor: YES” is specified for a field, any previous specification is set to NO.

As the default option IFG gives each field the display attributes laid down in the user profile (see page 188). These depend on the field type.

The display attributes “reverse video”, “color”, and “character set” only apply to Terminal 9763. “Color” and “reverse video” are not evaluated on other terminals, “character set” cannot be output on any other terminal. These character sets are created using ICE (only for 7-bit formats).

Note that different terminal types display the same attributes differently. For example, underlining is represented by a special color on a 9763 terminal.

When you select the function

“Modify display attributes of the fields”,

the display attributes of the first field in the format are displayed.

The field whose display attributes are currently being displayed appears in the window as a bright field; the field type and field length are supplied as additional information.

```
I F G            DISPLAY ATTRIBUTES OF FORMAT  DELIVERY                   0305
                 (For help: K3)       VERSION  01
                                      Field type: Input field       Length:
025

Bright           : YES         Blanked        : NO          Printable  : YES
UNICODE          : YES         Flashing       : NO          Cursor     : YES

Underline/italics: NO          Character color: 0    (01234567)
Inverse          : NO          Character set  : 0    (*STD    )

---------------------------   Line 001 to line 008  ----------------------------
                              Delivery Note

Company
Name:          -------------------------
Street:        -------------------------
ZIP code:            #####
City:          --------------------

--------------------------------------------------------------------------------
To process next field           SEND       To store and display format      F1
To select another field  MAR and SEND      To move window forward           F2
To abort function                K2        To move window back              F3
```

The fields are represented by their field definition characters and the contents underlined or italicized.

You can now modify the attributes of this field by overwriting the preset default values (see page 47). By pressing SEND you conclude the modification for a particular field.

If the modification was correct, the attributes of the next field are displayed. And so you go on, processing the fields one by one.

You can select a particular field by marking it.

When you come to the last field, the message

```
5303: YOU HAVE REACHED THE LAST FIELD IN THE FORMAT
```

appears.

You can now start again (move back the window) and check your input.

F1 stores the format with the modified attributes. The format is displayed as it will appear when used. The fields contain output fill characters.

```
                              Delivery Note
Company
Name:
Street:
ZIP code:      .....
City:

Customer no:  ............                              Date:

_____________________________________________________________________________
 Item no.  | Item description             | Quan-  | Unit  | Price
           |                              | tity   | price |
_____________________________________________________________________________
  00000    |                              | 000000 | 0000000 | @@@@@@@@@@@@@@ €
  00000    |                              | 000000 | 0000000 | @@@@@@@@@@@@@@ €
  00000    |                              | 000000 | 0000000 | @@@@@@@@@@@@@@ €
  00000    |                              | 000000 | 0000000 | @@@@@@@@@@@@@@ €
_____________________________________________________________________________
                                          Total:           @@@@@@@@@@@@@@ €
                                          Sales tax:        @@@@@@@@@@@@@@ €
                                                            ______________
                                  Total incl. sales tax:   @@@@@@@@@@@@@@ €
```

After the format is displayed, you can press one of the following keys:

SEND Now you can choose one of the functions offered (e.g. with SEND “Modify display attributes” again)

or F3 The format is stored and you can go back to the definition of the display attributes of the format.

As when drafting the image, you may also store and display the format using F1 at any time.

4.7 Modify symbolic names of fields

When this function is selected, the default values for the first field that is accessible to the program are displayed.

```
I F G                    FIELD NAMES OF FORMAT: DELIVERY                 0306
                         (Forhelp K3) VERSION : 01
Field accessible to program: YES     Field type: Input Field      Length:  025

Field name  :  NAME                        longer name
Group name  :                                        Start of group    :YES
                                                     End of group      : NO
Comment     :                                        Help panel        :HLPNAME
Table name  :               Table dimension: 00      Start of table         :NO
                                                     End of 1st tab. elem.  :NO
---------------------------   Line 001 to line 008   -------------------------
                              Delivery note

Company:
Name:          -------------------------
Street:        -------------------------
ZIP code:           #####
City:          --------------------

--------------------------------------------------------------------------------
To process next field            SEND       To store format                 F1
To select another field  MAR and SEND       To move window forward          F2
To abort function                K2         To move window backwards        F3
```

When you press the SEND key for formats with dialog extension, the next field is displayed. For all other formats, pressing the SEND key displays the next field accessible to the program. By “marking” and pressing SEND, you can select and modify any field, including fields that are not accessible to the program (text fields).

The fields for the field name, group name, comment, table name, and help panel may only contain characters from the EBCDIC.DF.03 character set.

With this function you define

– whether a field is to be accessible to the program
  NO means that the contents and attributes of the field are not accessible to the program even if a field name was specified.
  YES means that the field can be addressed via addressing aids or FHS-DM.

– the field name

  If you are using addressing aids, this name is the one used for addressing the fields in the application program. This name must comply with the conventions of the programming language used; however, this is not verified by IFG.

If you are using FHS-DM, this name is the name of the variable to which the field contents are written. Depending on the type of variable used, the name must comply with the syntax of dialog variables, the syntax of S variables, or both. This is validated by IFG if defined by means of an appropriate setting in the user profile (see page 195).

For formats with dialog extension, you can use field names with a length of up to 255 characters. If you wish to use a field name that is longer than 30 characters, mark “longer name” and press SEND. The following screen will appear, and you can then enter the field name.

```
I F G                    FIELD NAMES OF FORMAT : DELIVERY              030J
                         (For help K3)  VERSION: 01


Field name:
NAME









To store the field name:  F1                          To abort function:  K2
```

When you press the F1 key, you return to the previous screen.

Note that when you are editing a long field name in screen 0306, the end of the name may be truncated.

– whether the field is to form the start or end of a group .
All fields between “start of group” and “end of group” that are accessible to the program belonged to one group. They are addressed by the group name in the data transfer area when an FHS-noDE application is executed. This allows the application to modify all field attributes far more efficiently. If the data transfer area with separate attribute blocks and field contents is used, however, every field of a group is assigned its own attribute field and length field. For FHS-DE/DM formats, it is totally irrelevant in this case whether or not the fields belong to a group.
A group must not extend beyond a single table entry. The start-of-group and end-of-group fields must be accessible to the program. Further information on addressing aids can be found starting on page 191.

- the group name
- a text that is to appear as a comment in the addressing aid.
- the name of the help panel assigned to the current field (only for FHS-DE/DM formats). The maximum length is 7 characters; see page 55.
- whether this field is to be the start of a table, in which case you must also specify,
  - how many entries there will be in the table (table dimension)
  - the name to be given to the table. Defining a table name here allows you to access the table as a whole.
- whether the field terminates the table entry.
  IFG then checks that the table has been correctly defined.

Note

A table is defined as a repetition of a set of fields, where each set of fields represents an element of the table. All of the sets must contain the same field types (e.g. same length, same attributes, etc.).
For more information on the definition of tables, see page 145ff.

IFG presets default values. To change these values or assign field names, you proceed just as you would when changing display attributes.

4.8 Modify names of field control variables

On selecting this function, the default values for the first field are shown. You can access all fields of your format independently of one another.

```
I F G               FIELD-CONTROL VARIABLES OF FORMAT: DELIVERY             030G
                    (For help: K3)            VERSION: @
                                          Field type: INPUT FIELD   Length: 025
           Processing fields related to variable type: LOCK

Do you wish to select another variable type?
Variable name:




----------------------------  Line 001  to Line 008  -----------------------------
                              Delivery Note

Company:
Name:            -------------------------
Street:          -------------------------
ZIP code:            #####
City:             --------------------

--------------------------------------------------------------------------------
To process next field            SEND       To store format                  F1
To select another field  MAR and SEND       To move window forward           F2
To abort function                K2         To move window backwards         F3
```

When you press the SEND key, the next field relevant for the current variable type is output in turn, but you can select any field to be output and edited by “marking it” and pressing SEND.

This function can be used to define the names of control variables used by FHS-DM.

The following control variables can be defined:

– the LOCK variable of a single-choice field. This variable is used by FHS-DM to check whether the choice is available. If no name was defined, the choice is always available.

– the variables of type NUMROW, TOPINDEX and MODINDEX for the scrolling information of a list. FHS-DM uses these variables to control the output, input, and scrolling of the list. On output, the NUMROW variable contains the number of records defined in the list; the TOPINDEX variable contains the index of the first record to be output. FHS-DM then manages the user inputs, scrolling of records, updating of the TOPINDEX variable, and returns the indices of the modified records in the MODINDEX variable.

  If these control variables have not been defined, FHS-DM assumes the default values that are maintained by FHS-DM itself.

For more information on these variables see the FHS manual “Dialog Extension for TIAM and SDF-P”.

The name of the LOCK variable is output by default. You can, however, also select some other variable type. To do this, you must mark “select another variable” and press SEND. The following screen showing all types of field control variables will then be output.

```
I F G                        TYPES OF FIELD CONTROL VARIABLES            030I

 Variable type               Applicable on
-------------------------------------------------------------------------------
 LOCK                        the choices of a single choice field or on
                             the choices of a pull-down menu

 NUMROW                      the scroll prompt of a list
 TOPINDEX                    the scroll prompt of a list
 MODINDEX                    the scroll prompt of a list









To select a variable type, mark the relevant line and press SEND
To return to the current function without selection:        F3    For help: K3
```

Select one of the displayed variable types by marking it and pressing SEND. This will automatically return you to the “Field control variables” screen. You can then enter the variable name to be used by your application in that screen. If no variable is to be defined, delete the entry in the input field. The variable name must comply with the convention for S variables and/or dialog variables.

4.9 Modify input and validation attributes of fields (part I)

When this function is selected, the default values for the first field which is accessible to the program are displayed.

```
I F G           INPUT/VALIDATION ATTRIBUTES (I) OF FORMAT   DELIVERY        0307
                (For help: K3)               VERSION  @
Mandatory input  : NO           Automatic input: NO           Protected: NO
Min. input length: 000          Selectable     : NO           NUM lock : NO

Any character: YES
Arithmetic   : NO      Signed: NO      Decimal places: 00    Digit groups: NO
Alphabetic   : NO
Date:        : NO      Calendar check: NO      Time: NO     Message ID: *NONE
---------------------------  Line 001 to line 010   ----------------------------
                             Delivery note

Company
Name:          -------------------------
Street:        -------------------------
ZIP code:            #####
City:           --------------------

--------------------------------------------------------------------------------
To process next field           SEND       To store format                  F1
To select another field  MAR and SEND       To move window forward           F2
To abort function               K2          To move window backwards         F3
```

When you press the SEND key, only the fields accessible to the program are displayed. By “marking” and pressing SEND you can modify any field you wish.

With “Input/Validation Attributes (Part I)” you can define input protection for the field and the type of input, and also validation of the data with FHS. The input checks are thus mutually exclusive. This means for example, that arithmetic and alphabetic characters cannot be checked at the same time.

You can specify whether

– input is mandatory for the field
  the “mandatory input” attribute cannot be combined with the attributes “Automatic input”, “Protected”, and “Selectable”.

– the field should be returned to the application in any case
  Automatic input is ignored in the case of menu titles and choice fields.

– the field is protected (i.e. no user input is possible)
  The “Min. input length” must be equal to 000 for every protected field.

– a minimum input length is required,

– the field is selectable,

  A selectable field must be protected. The “selectable” attribute enables you to mark a protected field and thus access this field.

– the field is to be numeric (NUM lock).
  The attribute “numeric” (NUM lock) is a device-dependent attribute. Depending on the device, only the digits 0 to 9 and the characters period (.), comma (,), plus (+), minus (-), slash (/) and asterisk (*) can be input.

– the field is to contain any characters,

– it is to be an arithmetic field.
  Arithmetic fields are validated by FHS. Permissible characters are the digits 0 to 9 and, depending on what has been defined, decimal separators, digit separators and signs. An arithmetic field may not contain more than 15 digit positions. The number of digit positions is calculated from the field length minus the number of positions for signs, decimal separators and digit separators, provided the corresponding attributes have been selected. The combination “NUM lock” and “arithmetic” is permitted. For arithmetic fields you can also specify whether the field is to include a sign, the number of decimal places and if there is to be any digit separation.

– the field is to be a date field; you must define fields that are to contain the date as input or output fields when you create an image, making sure that you define the length of the field correctly. Depending on the specifications in the user profile (2- or 4-digit year specification), the date field must be either 8 or 10 characters long. If it is any other length, IFG rejects it and issues a corresponding error message. You can also specify whether a calendar check is to take place.

– the date is to be checked;

  The date is only checked if the attribute “Calendar check” contains the value YES.

– the field is to be a time field; (only for FHS-DE formats)
  The notational conventions of the time field correspond to those defined for the time field in the user profile.

– a message is to be assigned to the field; (only for FHS-DE formats)
  A message ID is 7 characters long. The default value *NONE indicates that no message has been assigned to the field. If no value is selected, an appropriate FHS error message is output.

– The field is to be alphabetic.
  The field may only contain letters and blanks. Letters are defined in the XHCS alphabetic table for the coded character set (CCS) used by the format. If no CCS is used letters are 'A' through 'Z' and 'a' through 'z'.

IFG specifies default values which are defined in the user profile. To change these values, proceed as you would when modifying the display attributes.

Notes

All input attributes (“Mandatory input”, “Automatic input”, “Min. input length”) are ignored by FHS for choice fields. A protected field can, however, always be converted by the application into an unprotected field.

The input and validation attributes “arithmetic”, “date”, “decimal places”, “signed”, “digit groups” and “calendar check”, “min. input length” and “mandatory input” (editing attributes) are only taken into account for format preparation if the data transfer area with separate attribute blocks and field contents is used for the format (see page 191).

When you have processed all the fields and stored the format using F1, you can check your inputs. For this purpose IFG defines preassignment characters which are dependent on the field data type, these being:

	Field data type
A	for alphabetic fields
9	for arithmetic fields
X	for fields with any characters (including numeric)

```
                          Delivery Note

Company
Name:     XXXXXXXXXXXXXXXXXXXXXXXX
Street:   XXXXXXXXXXXXXXXXXXXXXXXX
ZIP code: XXXXX
City:     XXXXXXXXXXXXXXXXXXXXXXX

Customer no:  XXXXXXXXXXXX                          Date:  30.09.1991
                                                    Time:  14:42:10
-------------------------------------------------------------------------
 Item no.  | Item description             | Quani- | Unit    | Price
           |                              | tity   | price   |
-------------------------------------------------------------------------
  XXXXX    | XXXXXXXXXXXXXXXXXXXXXXXXXXXX | 99.999 | 999,99- | 99.999.999,99- €
  XXXXX    | XXXXXXXXXXXXXXXXXXXXXXXXXXXX | 99.999 | 999,99- | 99.999.999,99- €
  XXXXX    | XXXXXXXXXXXXXXXXXXXXXXXXXXXX | 99.999 | 999,99- | 99.999.999,99- €
  XXXXX    | XXXXXXXXXXXXXXXXXXXXXXXXXXXX | 99.999 | 999,99- | 99.999.999,99- €
-------------------------------------------------------------------------
                                      Total:          999.999.999,99- €
                                      Sales tax:       99.999.999,99- €
                                                      ----------------
                              Total incl. sales tax:  999.999.999,99- €
```

For the display, the fields contain the following preassigned values:

– a date field is supplied with the current date,

– a time field is supplied with the current time (only for formats) with dialog extension),

– a field with the data type “arithmetic”, “alphabetic” or “any character” is completely filled with the preassignment character. If a sign is permitted in an arithmetic field, a negative sign is displayed. In addition, the characters defined as decimal and digit separators are displayed.

The following attributes can thus be checked from this display:

– field data type

– sign allowed

– number of decimal places

– digit group

– representation of the date (see “editing attributes of the format”)

– representation of the time (see “editing attributes of the format”)

– decimal separator (see “editing attributes of the format”)

– digit separator (see “editing attributes of the format”)

The attributes “protected”, “NUM lock” and “selectable” can be tested by means of appropriate inputs and cursor movements.

The attributes “automatic input”, “mandatory input” and “min. input length” are not displayed. The values of these attributes can only be checked when executing the application.

4.10 Modify input and validation attributes of fields (part II)

You can only choose this function if you are working with dialog extension.

On selecting this function, the default values for the first field accessible to the program are shown.

```
I F G        INPUT/VALIDATION ATTRIBUTES (II) OF FORMAT:  DELIVERY          030D
                (For help: K3)                VERSION :  @
                                        Field type:  Input Field      Length:  025
Range: from                        to                                Message ID: *NONE
Chars: EQ
                                                                     Message ID: *NONE
Value:  EQ


                                                                     Message ID: *NONE
---------------------------   Line 001 to line 010  ----------------------------
                              Delivery Note

Company
Name:              ------------------------
Street:            ------------------------
ZIP code:          #####
City:              --------------------

--------------------------------------------------------------------------------
To process next field            SEND       To store format                  F1
To select another field  MAR and SEND       To move window forward           F2
To abort function                K2         To move window backwards         F3
```

Pressing SEND causes the next field that is accessible to the program to be output. You can, however, edit any field by “marking” and pressing the SEND key.

With the “Input/Validation Attributes (part II)”, you can specify an range, a list of characters, or a list of values which are to be checked. You can allocate an error message to each check, by means of the message ID. If an error is detected during the check, the error message is output by FHS to the terminal. A blank entry in the message ID field means that the FHS default error message is output when an error occurs. The default value *NONE means that no error message is output. Otherwise the message allocated to this message ID is output.

– Range defines the minimum and maximum values which can be entered in an arithmetic field. Only numeric values can be entered. and they must not exceed 16 positions. The field to be checked must be defined as an arithmetic field in the format “Input/Validation Attributes (part I)”. The limit values (minimum maximum) can also contain a sign. A minus sign must be specified. As a separating line, you may only use the decimal separator (“,” or “.”) defined in the format “GENERAL ATTRIBUTES OF THE FORMAT”.

– “Chars” contains a list of characters for which a field should be checked. There are two possible results.
Firstly, the result is positive if only characters from the specified list are contained in the field. You must place the letters EQ before your character list.
Secondly, the result is positive if no characters from the specified are list contained in the field. You must place the letters NE before your character list.

  Blanks can be contained in the character list. They must not however be positioned last on the list.

  Example

  If you want the field to contain hexadecimal values only, you must define this field as EQ 0123456789ABCDEFabcdef.

– “Value” contains a series of values for which a field should be checked. The individual values are separated by blanks. If a value contains blanks, this value must be enclosed in quotes. If you want to specify a quote in the value list, you must specify the quote twice. There are two possible results.
Firstly, the result is positive if only values from the specified list are contained in the field. You must place the letters EQ before your list of values.
Secondly, the result is positive if no values from the specified list are contained in the field. You must place the letters NE before your list of values.

  Example

  EQ Monday Tuesday Wednesday Thursday Friday “all days” indicates that FHS-DE outputs an error message if the value entered does not match the name of a week day or the “all days” value.

4.11 Modify editing attributes of fields

When this function is selected, the default values for the first field which is accessible to the program are displayed.

```
I F G       EDITING ATTRIBUTES OF FORMAT  DELIVERY                 0308
              (For help: K3)       VERSION  01
Field data type:  any character      Field type:  Input Field        Length: 025

                                                Zero suppression     : NO
Alignment      left     : YES /YES
(Input/output) right    : NO  /NO              Floating sign        : NON
               none     : NO  /NO              Upper case only      : NO
Input/output fill char. :     /                UTM control field    : NO
Processing for undefined values: NO           Fast detection       : YES
Processing by exit routine     : NO           Exit code (remark)           :
---------------------------   Line 001 to line 008     ------------------------
                              Delivery Note

Company
Name:           -------------------------
Street:         -------------------------
ZIP code:       #####
City:           -------------------

--------------------------------------------------------------------------------
To process next field:           SEND          To store and display format:   F1
To select another field: MAR and SEND          To move window forward:        F2
To abort function:               K2            To move window back:           F3
```

When you press the SEND key, only the fields accessible to the program are displayed. By “marking” and pressing SEND you can modify any field you wish.

With the editing attributes you define how the field contents are to be edited by FHS, i.e.

In this screen, you define for all fields:

– whether the field is to be aligned on input/output (this attribute is not supported by FHS-DM formats for input)

  Note that when field alignment is used for text fields, the appearance of the format will not correspond exactly with the format that was defined when drafting the image for it. For example, if a text field begins with n blanks and is to be output left-aligned, it will be shifted by n blanks to the left on output. You can prevent this from occurring by removing field alignments for the output of text fields.

– which input/output fill characters are to be used.
Arithmetic fields may only contain zero as the input fill character; the output fill character must be neither a digit, nor a sign for date and time fields, the input and output fill character must be the same, and must not be a digit.

  For 8-bit formats, the fill characters can be any character from the defined character set. This also includes characters from the extended character set.

  Input fill characters are not supported with FHS-DM formats.

– whether undefined values are to be processed (this attribute is not supported with FHS-DM formats)

– whether the “fast” feature is to be used when checking undefined values for the field (i.e. instead of checking whether the entire field is filled with a fill character for undefined values, only the first character of the field is checked)

– whether the field is to be processed by a user-defined exit routine, and the code for that routine. The (exit routine can only consist of characters from the EBCDIC.DF.03 character set (this attribute is not supported for FHS-DM formats).

– whether field contents should automatically be converted to uppercase on input

– whether the field is a UTM controlled field

Note that the editing attributes “field alignment on input” and “exit routine” are not supported by FHS-DM.

The following additional attributes are specified for arithmetic fields:

– whether the value zero is to be completely suppressed on output and replaced by the output fill character. For example, with zero value suppression set, “045” and “000” would be output as “045” and “ ” if the output fill character is a blank.

  The combination of “Zero value suppression” and “Floating sign” does not make sense if the output fill character is not a blank, since blanks are always used as the fill character with a floating sign.
  “Zero value suppression” and an “Output fill character” of zero are mutually exclusive.

– whether leading zeros are to be replaced by fill characters on output. This means, for example, that “045” and “000” would be output as “ 45” and “ 0” if the output fill character is a blank.

– whether a floating sign is desired. “Floating sign” is only allowed in combination with “Sign” (see page 103).

Note that:

– the editing attributes “Leading zero suppression” and “Floating sign” are only taken into account for format preparation if the data transfer area with separate attribute blocks and field contents is used (see section “Addressing aid specifications” on page 191),

– “Zero value suppression” can only be used for formats with dialog extension.

IFG specifies default values which are defined in the user profile. To change these values, proceed as you would in modifying the display attributes.

When you have processed all the fields and stored the format using F1, you can check your inputs. IFG defines preassignment characters which are dependent on the field type, these being:

	Field data type
A	for alphabetic fields
9	for arithmetic fields
X	for fields with any characters (including numeric))

```
                          Delivery Note

Company
Name:     X
Street:   X
ZIP code: 0000X
City:     X

Customer no:  00000000000X                          Date:  30.09.2006
                                                    Time:  14.42.10
-------------------------------------------------------------------------
Item no.  | Item description            | Quan-  | Unit    | Price
          |                             | tity   | price   |
-------------------------------------------------------------------------
 0000X    | X                           | 00.009 | 000,09- |          0,09-
 0000X    | X                           | 00.009 | 000,09- |          0,09-
 0000X    | X                           | 00.009 | 000,09- |          0,09-
 0000X    | X                           | 00.009 | 000,09- |          0,09-
-------------------------------------------------------------------------
                                        Total:                       0,09-
                                        Sales tax:                   0,09-
                                                                  ----------
                                 Total incl. sales tax:              0,09-
```

For the display, the fields are supplied with the following values:

– a date field is supplied with the current date;

– a field with the data type “alphabetic” or “any character” is filled with a preassignment character. The rest of the field is filled with the input fill character;

– an arithmetic field is filled with 0 up to the last position, where 9 is entered. If a sign is permitted in an arithmetic field, a negative sign is supplied.

Preassignment values in an arithmetic field are right-justified; in all other fields they are left-justified.

The following attributes can thus be gathered from this display:

– alignment and fill characters upon output

– Leading zero suppression

– floating sign

– field data type

– sign permitted

– number of decimal places

– representation of the date

– representation of the time

– decimal separator

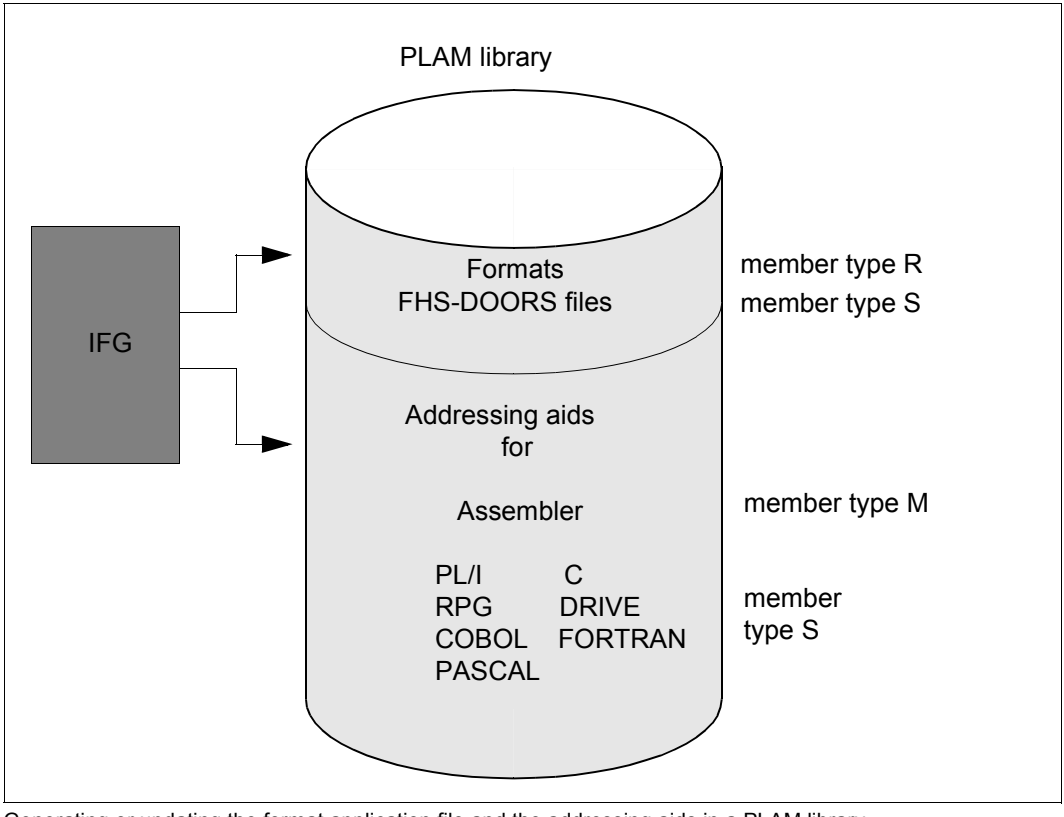
– digit separator

The attributes for “alignment and fill characters on input”, “conversion of lowercase to uppercase”, “processing undefined values”, “processing by exit routine”, “exit code”, “UTM control field”, and “zero value suppression” are not shown. The values for these attributes can only be checked by the application at runtime.

5 Preparing a format for use

Before a format can be used, it must be stored together with any associated addressing aids in a library.

Formats and addressing aids are stored directly in program (PLAM) libraries. Formats are stored as members of type R in PLAM libraries; addressing aids for Assembler are stored by default as members of type M, and addressing aids for PL/I, RPG, COBOL, Pascal, C, DRIVE, and Fortran are stored as members of type S. You can, however, also define some other member type in your profile library (see page 195) that can be used by IFG to store the addressing aids. Formats and addressing aids have the same member versions as the source format.



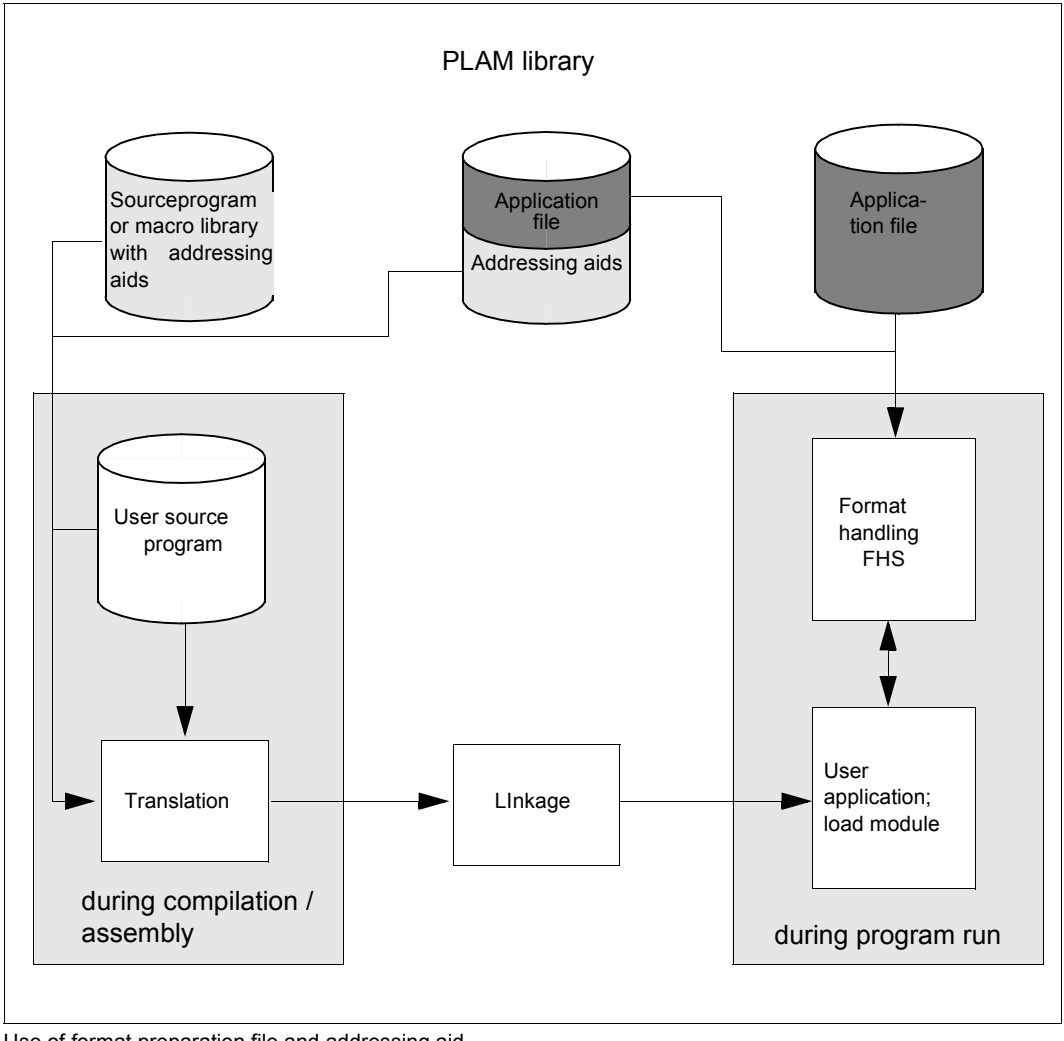
Generating or updating the format application file and the addressing aids in a PLAM library

The object modules generated by IFG have XS capability, i.e. they can also be loaded in the address space above the 16 MBytes.

Formats can also be converted for FHS-DOORS directly at the time of generating the object modules (see section “View user profile” on page 180 and section “Default values for FHS-DOORS conversion (part I)” on page 200. FHS-DOORS objects are generated as members of type S.

Note for formats which use the data transfer area with separate attribute blocks and field contents

Depending on the programming language used, IFG generates data declarations for arithmetic fields which permit the application program to process these fields conveniently (see page 191ff).



Use of format preparation file and addressing aid

Select function 05 Format Preparation.

```
I F G                        FORMAT PREPARATION                          0501


Name of format library:  IFG.Formats

Would you like to view the directory first?  If so, mark and press SEND.

Name of format to be prepared:  CUSTOMER   Version:  01
                     Password:

-  Do you want your format application file to be updated?          YES
   Library name:  formats.objmod

-  Do you want your addressing aids library to be updated?          YES
   Library name: formate.sourcelib
   Name of the addressing aid(s) without prefix or suffix (enter only if
   different from format name) :

-  Do you want a 'program-readable format extract' to be generated? NO


To execute the function:     SEND
To select another function:  F3                                 For help:  K3
```

When you view the directory of the format library, IFG shows whether format preparation (FP) has already taken place for a format. FP-Fmt indicates whether an object module was already generated for the formats, and FP-Adr indicates whether an addressing aid has been created for it.

Enter the name and the version of the format and specify the name of the PLAM library for formats and addressing aids.

You can also define default values for these library names in the user profile. These names are than automatically entered by IFG, though they can still be changed during “format preparation”.
If you specify no name for the addressing aid (generally the case), it is given the name of the format.

The function “program-readable format extract” acts as a link between IFG and Fujitsu Siemens Computers GmbH standard products, in which IFG formats are used without FHS or UTM application programs. This function has been released for internal users only and is therefore not described here.

Format preparation without FHS-DOORS conversion

You can prepare the format conversion, key lists, and messages in any order. Language dependencies are not taken into account by IFG during format preparation.

Format preparation with FHS-DOORS conversion

In this case, the key lists must be created before the corresponding non-help format. If conversion of the key list is performed with the FHS-DOORS converter, the converter searches for the appropriate object key list in the format application file. Either the language code in the format name (eighth character of the name) or the entry in the IDHSLNG is evaluated for this purpose (see the FHS manual “Format Handling System for openUTM, TIAM, DCAM”):

– if a language code is found, the FHS converter extends the name of the key list up to the seventh character with “#” and appends the language code as the eighth character.

– if no language code is found, the name of the key list remains unaltered.

Format preparation is performed even if no key list is found.

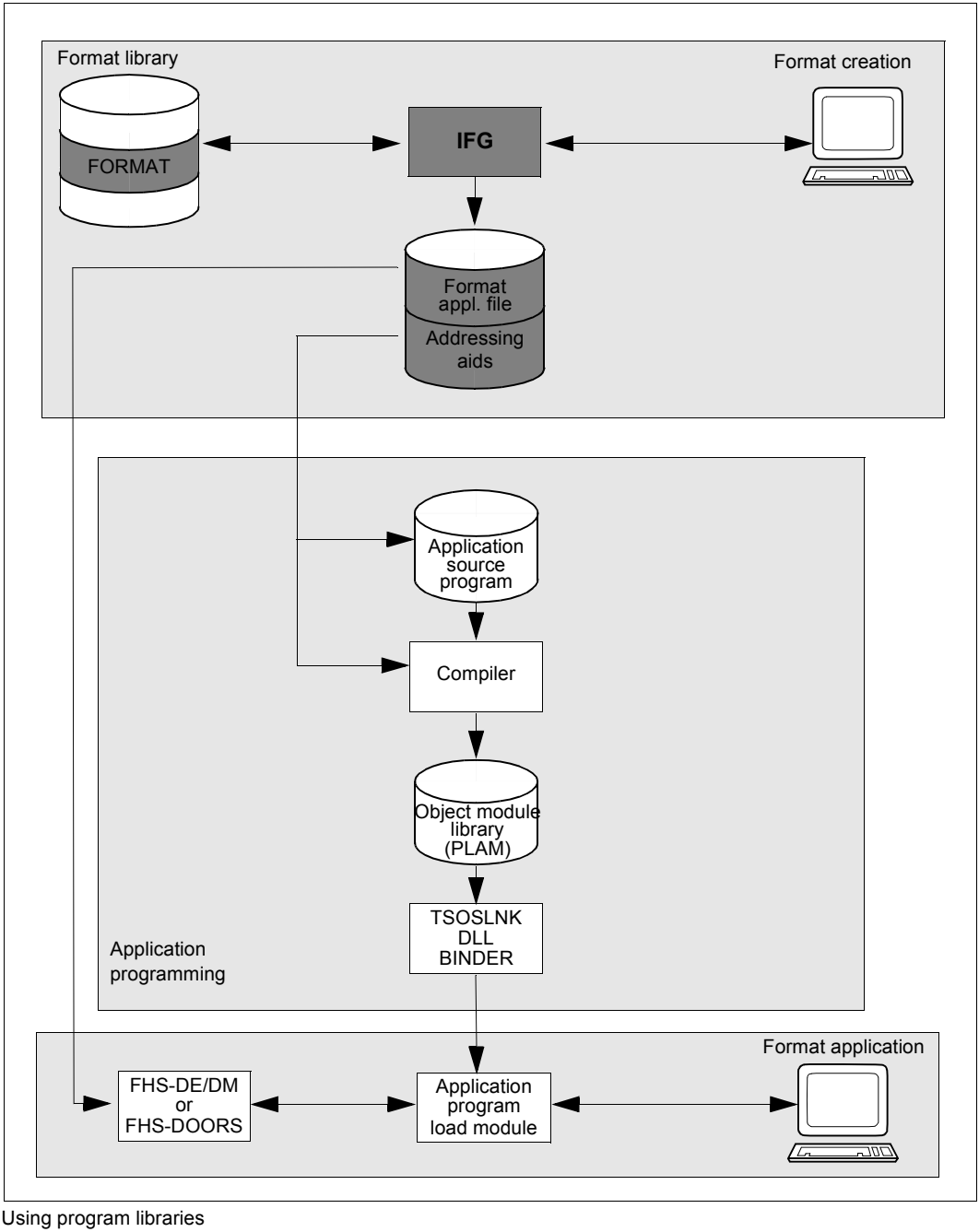
Please take note of the FHS converter messages that may be output on generating the key list and take appropriate action. In the above case, for example, you should provide the missing key list and restart the format preparation.
Although a format of type S was already generated, BLS messages (binder loader system) may still be output. Some of the BLS messages are, however, only to be understood as warnings. For example, if a format in your library is being processed for the first time, the message BLS0334 “SYMBOL IDHS.... CANNOT BE FOUND. LOADING ABORTED” is output. This message refers to the three formats IDHSLNG, IDHSHCx and IDHSHDx, which need not be present in your library (see the FHS manual “Format Handling System for openUTM, TIAM, DCAM”).

Note

The mapping of the key texts and the additional pushbuttons or menu items are not taken into consideration in the help formats.

The object module (type “R”) and the optional supplementary FHS-DOORS converter SDC file (member type “S” with .SDC as suffix) always receive the same name and the same version as the source format.

Once you have stored the formats and addressing aids in PLAM libraries and received a message, format preparation is complete.



Using program libraries

6 Viewing an existing format

Function 02 “Display format” allows you to have existing formats shown to you on the screen. You cannot use this function to change your format in any way. This function is quicker than function 03 “Modify and Extend Format” because IFG does not have to check or store any input. In addition, several users can view the same format simultaneously, an option which is not available with the function “modify and extend formats”.

The format must be displayable on the display terminal on which you are using IFG, otherwise an error message will appear.

After you have selected function 02, the following screen is displayed:

```
I F G                          VIEWING A FORMAT
0201

Name of format library: IFG.Formats

Do you wish to view the directory first?

Format name :                      Version :

Do you wish to see

-  the image, die FHS-DE-Eigenschaften oder die Feldeigensch.   of the format?

-  the Pull-Down-Menues                                         of the format?

-  the terminal specifications                                             ?
-  the display attributes                                       of the format?
-  the editing attributes                                       of the format?
-  die Namen der globalen Steuer-Variablen                      of the format



To select the desired function:  MAR and SEND
To select another function:             F3                       For help:  K3
```

Enter the name and, if applicable, the version of the format or select the format you want to see from the directory.
Mark one of the functions shown.

If you do not specify a version for the format you want to view, IFG automatically uses the highest possible version which exists for this format.

6.1 Show the image, DE/DM features and field attributes of the format

When you select the function to view the ´FHS-DE/DM features, the image or the field attributes” of the format, the following screen is displayed. You can then mark one of the indicated functions.

```
I F G            VIEWING THE IMAGE OF FORMAT: DELIVERY                  0204
                                  VERSION: @


Name of format library: IFG.FORMATS


Do you wish to view

-  the image                                    of the Format?
-  the FHS-DE/DM-features                       of the Format?

-  the display attributes                       of the format fields?
-  the symbolic names                           of the format fields?
-  the names of the control variables           of the format fields?
-  the input/validation attributes (part I)     of the format fields?
-  the input/validation attributes (part II)    of the format fields?
-  the editing attributes                       of the format fields?


To select the desired function: MAR and SEND    To select another function:  F3
To view another format:               F2                          For help:  K3
```

6.1.1 Show image as it will appear when used

The image of the format is displayed as it will appear when it is used.

This means that

– input and output fields are filled with the fill character for output defined for the relevant field. (The default for numeric input fields is “blank”; the position and the length of the fields can be tested by entering something).

– output fields that are protected against overwriting and have “blank” (default) as the fill character for output are filled with the character used to define the field.

6.1.2 Show features of an FHS-DE/DM format

The attributes of an FHS-DE/DM format are displayed as described on page 72 (Creating an FHS-DE/DM format).

6.1.3 Show display attributes of the fields

The display attributes of the fields of the format are displayed as described (as described in the section “Modify display attributes of fields” on page 94.

6.1.4 Show symbolic names of the fields

The symbolic names of the fields are displayed as described on page 98.

6.1.5 Show names of field control variables

The names of field control variables are displayed as described on page 101.

6.1.6 Show input/validation attributes of the fields (part I)

The input and validation attributes are displayed as described on page 103.

6.1.7 Show input/validation attributes of the fields (part II)

The input and validation attributes are displayed as described on page 107. You can only view the format if it is an FHS-DE format.

6.1.8 Show editing attributes of the fields

The editing attributes of the fields are displayed as described on page 109.

6.2 Show pull-down menu of the format

When you select the function to “view the pull-down menu” of the format, the following screen is output.

```
I F G        VIEWING THE PULL-DOWN-MENUS OF FORMAT: DELIVERY            02OZ
                                          VERSION: @


Name of format library: IFG.FORMATS

Mark the menu title of the pull-down-menu to view:
     File  Edit  Display  Options


Do you wish to view

-  the image                              of the menu?

-  the display attributes                 of the menu fields?
-  the symbolic names                     of the menu fields?
-  the names of the control variables     of the menu fields?
-  the input/validation attributes        of the menu fields?
-  the editing attributes                 of the menu fields?



To select the desired function:MAR and SEND     To select another function:  F3
To view another format:                F2                        For help:  K3
```

Mark the menu title of the pull-down menu that you wish to see and one of the displayed functions. The selected menu title will be shown highlighted.

The same method can be used to view pull-down menus as when viewing a format.

6.3 Show terminal specifications

Specifications for the terminal are displayed, viz.:

– the possible terminal types on which the format can be used
– the default values for application
– the controller if the terminal is a printer

More information can be found on page 136

6.4 Show display attributes of the format

The display attributes of the format are displayed, viz.:

– password
– format identifier
– ICE character set used
– screen dimensions
– start line when used as a partial format
– background color
– extended character set
– Lock for “Change-code” key

More information can be found on page 136.

6.5 Show editing attributes of the format

The editing attributes of the format are displayed, viz.:

– field alignment and fill characters for input and output
– display of decimal and digit separators in arithmetic fields
– representation of the date and time
– representation of undefined values

More information can be found on page 138.

6.6 Show names of global control variables

The names of the global control variables for the table of marked fields is displayed.

More information can be found on page 140.

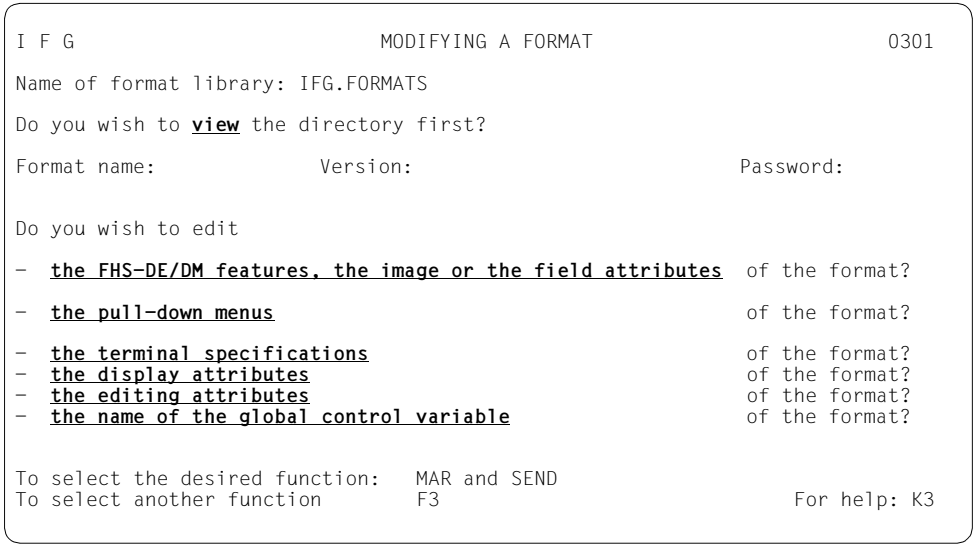
7 Modifying or extending a format

You can modify any of the following: the image, the symbolic names of the fields and the field editing attributes of the fields, i.e. everything that was defined when the format was first generated. In addition you have the option of modifying some of the general attributes of the format.

The format must be displayable on the terminal on which you are using IFG; otherwise an error message will be issued.

The terminal automatically uses the character set of the format to be processed or a character set that is compatible with the format. This prevents you from using characters which are not contained in the character set of the format. Therefore, even if you are using an 8-bit terminal, you can only use characters from EBCDIC.DF.03 when modifying a 7-bit format. Additional information is given in the appendix, starting on page 316.

Select function 03.



You can directly enter the format that you wish to modify or extend, or you can select one via the directory.
You can then mark one of the displayed functions.

If you do not specify a version for the format that you wish to modify, IFG will automatically assume the highest existing version for that format.

You may either specify the name of the desired format directly or select a format from the directory.

7.1 Modify image, FHS-DE/DM features, and field attributes of the format.

When you select the function to “edit the image of a format”, the following screen is output. You can then mark one of the displayed functions.

```
I F G            MODIFYING THE IMAGE OF FORMAT: DELIVERY                  0304
                                    VERSION: @

Name of format library: IFG.FORMATS

Do you wish to edit

 -  the FHS-DE/DM features and the image              of the format?

 -  the display attributes                           of the format fields?
 -  the symbolic names                               of the format fields?
 -  the names of the control variables               of the format fields?
 -  the input/validation attributes (part I)         of the format fields?
 -  the input/validation attributes (part II)        of the format fields?
 -  the editing attributes                           of the format fields?

Do you wish to

 -  match all format field attributes to those in the user profile?


To select the desired function:MAR and SEND     To select another function:  F3
To process another format:            F2                          For help:  K3
```

7.1.1 Modify FHS-DE/DM attributes and the image

FHS-DE attributes are edited and defined in similar ways (see page 72).

Note

When you are modifying the image of a format, you should bear in mind that each menu title of the menu bar is assigned a pull-down menu. As in the case of field names/ attributes, retention conflicts may occur if too many changes are made at the same time (see page 128). In such cases, you will be prompted to correct the error or undo the line by pressing SEND. Modifications are thus introduced in steps in order to prevent the loss of pull-down menus.
If the field attributes are to match the attributes in the user profile, only the fields of the format and not those of the pull-down menu are adapted. The method by which field attributes of the pull-down menu can be adapted to those in the user profile is described in the section “Match field attributes to those in user profile” on page 132.

You can

– insert, delete and overwrite
– define new fields
– repeat lines
– delete lines
– justify fields

the same way as in image drafting.

Two modes are available to you when modifying the image of a format:

– image modification without the retention of field names/attributes
– image modification with the retention of field names/attributes

You can specify which mode you work in either in the user profile (see page 183) or directly in IFG screen “Modifying the Image for Format ...”. The default option is image modification without retention of field names/ attributes.

```
I F G          MODIFYING THE IMAGE FOR FORMAT: DELIVERY                    0310
               (For help: K3)       VERSION: @
   _ Input field             > Repetition character      - Delete line
   # Input field, numeric    < Start-of-field char.      + Repeat line
   @ Output field            " Alignment
   % Single choice field     & Multiple choice field     ^ List
----+----1----+----2----+--  Line 001 to line 010  --+----6----+----7----+----8
                            Delivery Note
Company:
Name:     _________________________
Street:   _________________________
ZIP code:      #####
City:     _________________________

Customer No:  ############                              Date:  __________


----+----1----+----2----+----3----+----4----+----5----+----6----+----7----+----8
     Retain field names and attr. in modif. lines?  YES
To insert blank line  or  image of format:        To store and display:  F1
(Version:                          )              Window fwd. 05 lines:  F2
          at line 001   :     MAR and SEND       Window back 05 lines:  F3
To switch to large window:    MAR and SEND        To check your input: SEND
```

Modifying an image without retention of field names/attributes

Each line that you have changed in some way is regarded by IFG as a newly created line, i.e. names, as well as any attributes (insofar as they differ from the default values) defined for fields in this line are lost.

Modifying an image with retention of field names/attributes

You can modify lines without losing previously defined field names and/or field attributes which deviate from the user profile. In a single step you can

– move fields or

– delete fields or

– insert fields

The following conditions, however, must be observed:

– You can only move fields if the number of fields in a line **and** the order of the field types remain unchanged.

– You can only insert or delete fields if the position and type of the other fields remain unchanged.

In addition, you can

– extend fields,

– shorten fields and

– modify texts.

After you press the SEND or F1 key, IFG checks whether all these conditions have been observed. It thus makes sense to make changes step by step and to check whether each individual modification is correct after you have made it using the SEND key. If you have made an error, IFG issues the message:

```
5380: RETENTION CONFLICT: CORRECT THE ERROR OR RESET LINE (SEND) OR CHANGE MODE
```

The cursor indicates the first line in which a retention conflict occurs. You now have a choice of three possible responses:

– Changing the line so as to meet the requirements for retention, or

– pressing SEND to reverse the change, in which case the line is displayed again in its original form, or

– switching to the small image window and changing the retention mode. This is advisable only once all the lines modified without any retention conflict have been stored with F1.

Examples

– Moving a field

```
Old line: ZIP code: #####  City:       @@@@@@@
New line: ZIP code: #####  City: @@@@@@@
```

Here blanks were deleted causing the output field to be shifted to the left.

– Inserting a field

```
Old line: ZIP code: #####  City:      @@@@@@@
New line: ZIP code: #####  City: @@@@ @@@@@@@
```

The output field was inserted by overwriting the blanks with @.

– Deleting a field

```
Old line: ZIP code: #####  City: @@@@ @@@@@@@
New line: ZIP code: #####  City:      @@@@@@@
```

The input field was deleted by overwriting it with blanks.

– Extending/shortening a field

```
Old line: ZIP code: ########  City: @@@@@@@
New line: ZIP code: #####           @@@@@@@@@@@@@@@@
```

If you want to make more than one change in a line, you should check each individual change first using SEND. If there is no retention conflict, you can go on and perform the next one. Only when all the changes have been entered correctly should you store the format using F1.

Example of more than one change in a line

Old line

```
Name: ---------- City: ##### ----------
```

In this line

– the input field for “Name” is to be extended,

– the numeric field is to be deleted and the input field for “City” moved

– and a field for “First name” is to be inserted.

First extend the input field by the required number of positions and terminate input by pressing SEND.

New line

```
Name: ---------- City: ##### ---------- SEND
```

You must delete the numeric field in two stages. First you must overwrite the characters “####” with blanks, then press the SEND key, and finally delete the blanks and once more terminate the action with SEND.

New line

```
Name: ---------------    City: ␣␣␣␣ ---------- SEND
Name: ---------------    City: ---------- SEND
```

To insert an input field for the first name you must first insert the requisite number of blanks in front of the “City” field, press the SEND key, insert the new fields as the next step and once more terminate the action with SEND.

New line

```
Name: -------------  ␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣␣  City: ---------- SEND
Name: -------------  First name: -----------  City: ---------- SEND
*     *                                      *     *
```

The fields marked with * have retained their field names and attributes.

Notes

When fields are deleted or inserted, under some circumstances field attributes can be lost without the field concerned being deleted. When deleting or inserting, IFG compares the start positions of the fields so that it can assign existing attributes to the fields of a modified line. The position of a field is marked by the first character which is not a null character or blank. Fields can, however, also lose their attributes because these have been separated from them by the insertion of a new field.

7.1.2 Modify display attributes of fields

The procedure is the same as for modifying the display attributes (page 94ff).

7.1.3 Modify symbolic names of fields

The procedure is the same as for modifying the symbolic names from the default (page 98ff).

7.1.4 Modify names of field control variables

The procedure is the same as for modifying the names of field control variables (page 101).

7.1.5 Modify input/validation attributes of fields

The procedure is the same as for modifying the input/validation attributes from the default (part I on page 103 and part II on page 107).

7.1.6 Modify editing attributes of fields

The procedure is the same as for modifying the editing attributes from the default (page 109ff).

7.1.7 Match field attributes to those in user profile

With this function, all field-specific attributes of the fields of a format that can be set in the user profile are reset to the value specified in the current user profile. Other attributes are not affected. You can activate the function by marking it and pressing SEND. If the message

```
5384: THE FORMAT HAS BEEN CONVERTED.                    TO STORE FORMAT:F1
```

is displayed, you can save the format with F1. The following message will then appear:

```
5385: THE FIELD ATTRIBUTES HAVE BEEN CONVERTED AS REQUESTED
```

However, if the message

```
5386: THE DATA TRANSFER AREA HAS CHANGED.               TO STORE FORMAT:F1
```

is displayed, you should check the format again to verify that the changes are the same as those you requested and also meaningful.

If matching the field attributes results in attribute combinations that are illegal, the attributes are not reset, and the error message

```
5383: CONVERSION IMPOSSIBLE DUE TO ILLEGAL RESULTING ATTRIBUTE COMBINATION
```

is displayed on the screen. This may occur, for example, if a field is assigned the attributes “flashing” and “markable” or if an arithmetic field is too long.

7.2 Modify pull-down menu of format

The following screen is displayed when this function is selected.

```
I F G         EDITING THE PULL-DOWN MENUS OF FORMAT: DELIVERY              030Z
                                        VERSION: @

Name of format library: IFG.FORMATS

Mark the menu title of the pull-down menu to edit:
     File  Edit  Display  Options                                          Help
Do you wish to edit
 -  the image                                    of the menus?

 -  the display attributes                       of the menu fields?
 -  the symbolic names                           of the menu fields?
 -  the names of the control variables           of the menu fields?
 -  the input/validation attributes              of the menu fields?
 -  the editing attributes                       of the menu fields?

Do you wish to
 -  match all menu field attributes to those in the user profile?


To return to previous function:        SEND
To select the desired function:  MAR and SEND     To select another function: F3
To process another format:             F2                          For help: K3
```

You can select the menu title of the pull-down menu that you wish to edit and one of the displayed functions by “marking” and pressing SEND.

When you have selected the function to “edit the image of the pull-down menu”, the function for creating a single-choice field (see page 82) is called, since a pull-down menu is modified in exactly the same way as a single-choice field. Note, however, that you cannot define an input prompt character when modifying a pull-down menu and that the number of choices per line (restricted to one) cannot be changed.

The function for editing menu attributes corresponds to the function for creating format attributes; however, only a subset of the attributes of the format field are available.

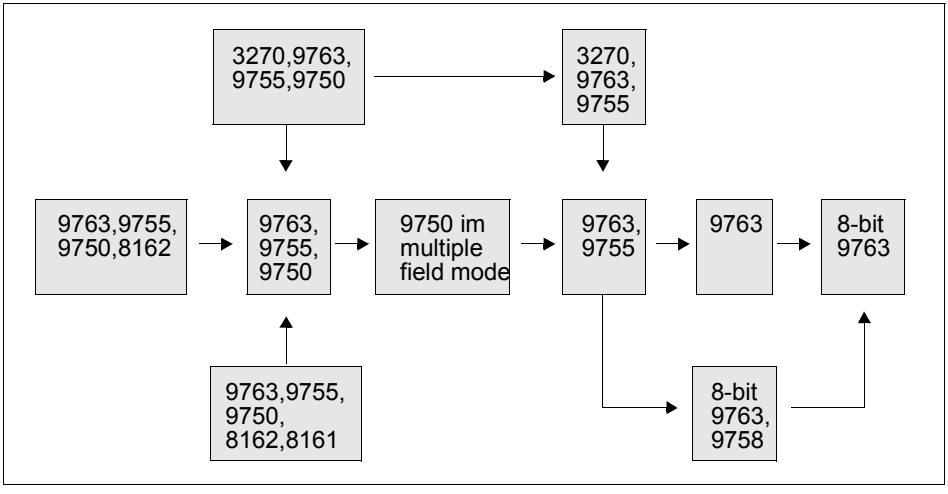
7.3 Modify terminal specifications

```
I F G              GENERAL ATTRIBUTES OF FORMAT  DELIVERY                    0309
                   TERMINAL SPECIFICATIONS   VERSION @

 Format is to be suitable for use on the following terminal group (+ printer):
 8-Bit 9763                    : NO    9750 in multiple field mode  : NO
 8-Bit 9763,9758               : NO    3270,9763,9755               : NO
 9763                          : NO    3270,9763,9755,9750          : NO
 9763,9755                     : NO
 9763,9755,9750                : YES

Default device when format is used:
 - display terminal                  - printer
   9763: NO                            9001     : NO    9004: NO    9022: NO
   9755: NO                            9001-8931: NO    9011: NO    PCL : NO
   9750: YES    3270: NO               9002/9003: NO    9012: NO    3287: NO
                                       9003     : NO    9013: NO


 If the default device is a printer, is it accessed through
 - a printer controller: NO            - or a 9763, 9755, 9750: NO   terminal?
 - RSO                 : NO

To check your input:  SEND                  To store the modified format:    F1
For help:  K3                      To return without storing any changes:    F3
```

With this function you can

– define a terminal type on which the format is to be output differing from the default laid down in the user profile (see also the section starting on page 185ff);

– change the terminal group as shown in the following figure. If a format is no longer needed for the 9750 terminal for example, you can increase the number of fields the format can have per line, by changing the terminal group.



Changing the terminal group

You store the modified format with F1.

7.4 Modify display attributes of the format

```
I F G              GENERAL ATTRIBUTES OF FORMAT  DELIVERY                  030A
                                       VERSION:  01
                                 DISPLAY ATTRIBUTES

    Password :                    Format identifier: DELIVERY NOTE

    The format uses the following character sets
          0.: *STD          1.:               2.:               3.:
          4.:               5.:               6.:               7.:


    Holes color : U (W=white, G=gray, U=undefined)

    Screen dimensions: 24 x 080 (lines x columns)

    Start line when used as a partial format: 00
    (If start line = 00 then the format is a full format)

    Coded Character Set of the format:
    Requires UNICODE support         : YES

To check your input:   SEND                    To store the modified format:  F1
For help           :   K3            To return without storing any changes:  F3
```

With this function you can

– assign a new password for the format (up to 8 characters) or change an existing one;

– assign a format identifier for the format (up to 16 characters) or change an existing one; the format identifier is displayed when the directory is output.

– enter additional character sets that are to be used for the format but are not entered in the user profile. Character sets created with ICE can be assigned to 9763 terminals only. You may enter one or more character sets to be used for the format. Each character set entered is assigned a reference number between 01 and 07. The reference number 0 stands for the default character set for the terminal. The character set specified here can be assigned to the individual fields using functions 01 or 03. When the format is displayed, the necessary character sets are loaded on the terminal. You can also use function 03 to define additional character sets for the individual fields. ICE character sets cannot be used by 8-bit formats.

– specify how the spaces between fields are to be displayed. The default value for “holes color” is “U” (undefined). The holes color can be changed to white (black type on a white background) or grey (black type on a grey background). When you are using the format with FHS, the default value “U” means that the holes color set for the terminal is the one used. A choice other than “U” is meaningful only for monochrome data display terminals; it is ignored for other terminal types.

– change the screen dimensions´

  Options:

  24 lines x 80 columns
  27 lines x 132 columns
  32 lines x 80 columns
  43 lines x 80 columns

  You can only reduce the screen dimensions if the format fits into these reduced dimensions. The default value for “screen dimensions” is 24 lines and 80 columns. Other screen dimensions, such as 32 lines x 80 columns, 43 lines x 80 columns or 27 lines x 132 columns, are only possible for 9763 terminals. You should only change the default setting 24 x 80 if all terminals on which the formats are to be output support these screen dimensions.

– specify whether the format is to be used as a partial format. Partial formats are formats that occupy only part of the screen. This enables them to be combined (only with FHS-noDE/DE) for use as part of a larger format. You specify the line where the partial format is to start on the screen. IFG then checks whether the start position and number of lines in the format are compatible with the screen size.
  When the partial format is displayed or printed with IFG, the start position is ignored; the format is always displayed/printed out from line 1 onward.

– specify the coded character set (only for 8-bit formats).
  The character set of a format cannot be modified once it has been specified.

– view the extended character set (only for 8-bit formats). Note that once an extended character set is defined, it can no longer be changed.

– specify the Unicode mode for a format or not.

  Using a mix of Unicode and non-Unicode partial formats presents a difficulty to FHS since the user can replace one of the partial formats by another at any time.
  For example, the screen could be entirely 7-/8-bit at first, then part of it could become Unicode and would confuse FHS. To avoid this FHS imposes a restriction on such a mix to work: the first partial format must be known as Unicode itself even if it doesn't contain any Unicode fields. This means that a global Unicode attribute is required for the format.

  The edition of the display attributes of the format comprises a "Requires UNICODE display" field which can be set to YES or NO. For consistency this field will be set to YES as soon as the format contains a UNICODE field. Resetting the value to NO will reset the UNICODE attributes of every field in the format.

You store the modified format with F1.

Note

Make sure that all the terminals on which the format is to be output can display the screen dimensions specified, as well as the character sets.

7.5 Modify editing attributes of the format

```
I F G            GENERAL ATTRIBUTES OF FORMAT  DELIVERY                    030B
                                    VERSION:  01
                          EDITING ATTRIBUTES

Account is to be taken during format application of field alignment
               in input: YES                 in output: YES
Representation of arithmetic fields
           decimal separator: .                 digit separator: .
Representation of the time
                with seconds: YES                     separator: :
Representation of the date
sequence of day, month, year: YMJ (YyM,J)             separator: .
             year, two digits: NO                   four digits: YES

Dialog extensions required? : YES

Representation of undefined values
                      any char.     arithmetic    alphabetic    date/time
    character:              .             .             .             .
    hexadecimal            00            00            00            00

To check your input: SEND                        To store the modified format: F1
For help:            K3                 To return without storing any changes: F3
```

With this function you can

– define the field alignment and fill characters for input and output;

  To reduce runtimes of the FHS application, you can dispense with the following when using a format:

  – alignment of field contents

  – padding with a fill character

  To do this, enter NO for “Account is to be taken during format application of field alignment”.

– define which characters are to be used to represent digit and decimal separators in arithmetic fields;

  You can define one character to represent the decimal separator and one to represent the digit separator in arithmetic fields. The permissible characters are:

  – “,” or “.” for the decimal separator

  – “,”, “.” or “ ” (blank) for the digit separator

  The decimal separator and digit separator must be different.

– define whether the time is specified with or without seconds, and define the separator.

  You can define the following for time representation:

  – whether seconds are to be specified,

  – the separator between hours, minutes, and seconds. All characters except the digits 0 and 9 are permissible.

– define for a date field the sequence of day, month, and year, the separator, and whether the year is to be indicated with two or four digits.

  You can define the following for date representation:

  – the sequence of day, month, and year,; all six possible combinations are permitted.

  – the separator between day, month, and year; all characters except the digits 0 through 9 are permissible.

  – 2- or 4-digit year specification. In the case of a 2-digit specification, it is assumed during the validity check that the date is sometime between 1901 and 2099. With 4-digit specification, the date may be within the period from 15.10.1582 to 31.12.2099.

– define whether you wish to use dialog extension. You can set this value to “YES” if your format was generated with an earlier version of IFG or was generated earlier without dialog extension. Once you have set the value to “YES”, you cannot reset it to “NO”. By default, dialog extension is not used.

– define, for each field type, whether the substitute character for undefined values can be displayed or is a hexadecimal value. For 8-bit formats, this character can be any of the characters of the format character set. This means that the substitute character cannot be a character from EBCDIC-DF.03 (see also the FHS manual “Format Handling System for openUTM, TIAM, DCAM”).

You store the modified format with F1.

7.6 Modify names of global control variables

```
I F G                GENERAL ATTRIBUTES OF FORMAT: DELIVERY          030H
                                        VERSION: @
                           GLOBAL CONTROL VARIABLE


Name of the MARK variable:









To check your input:  SEND                    To store the modified format: F1
For help:             K3               To return without storing any changes: F3
```

You can use this function to define a variable name for FHS-DM.

The global control variable is a table in which FHS-DM enters the list of all names of marked fields in a format. Fields within a list record are not entered.

If no name was assigned to the variable, the corresponding table is not generated by FHS-DM at runtime, and the information will not be available to the application program.

You can store the modified format by pressing F1.

8 Combining formats

Existing formats in the same format library can be combined to form a new format.
The attributes and names of the fields are taken over by the new format (in contrast to the subfunction “Insertion of images”).

Select function **04.**

```
I F G                          COMBINING FORMATS                          0401
                               (For help: K3)
Would you like to view the directory first?  If so, mark and press SEND

Name of the composite format:
Version                     :
Password                    :               Identifier:

        Format name        Version             after      times      Password
                                             blank lines
    01:                                         00          01
    02:                                         01          01
    03:                                         00          01
    04:                                         00          01
    05:                                         00          01
    06:                                         00          01
    07:                                         00          01
    08:                                         00          01


To check the input:          SEND               To store and display:        F1
To combine the next format:  F2                 To select another function:  F3
```

Before using the combining function you can check the directory to see which formats you have already stored in the current format library and are therefore available for combination.

When you combine formats you should state

– the name and version for the new format,

– the names and versions of the formats to be combined,

– the number of blank lines you want between the individual formats and

– if appropriate, a repetition factor indicating how many times a format is to be used at the specified place.

In addition you need to enter any passwords assigned to the formats.

Please note the following rules:

- The name and/or version of the new format must be different from the names of the old formats used.
- If you do not specify a version for the new format, IFG automatically creates the format with the highest possible version “@”.
- If you do not specify versions for the formats to be combined, IFG uses the highest available version of each of the formats.
- The new composite format must not contain itself.
- Formats that existed prior to opening function 04 cannot be overwritten.
- Existing formats are combined to form a new format from the top down (possibly separated by blank lines).
  When the individual formats are combined, the starting line of partial formats is not taken into account.
- The same format can be used for combination more than once in succession. There are two ways of doing this:
  - you can specify the same format name and version more than once, or
  - you can specify a repetition factor, in which case IFG automatically creates a table structure for the new format (see page 146). The old format becomes a table entry of the new one. The old format must not include tables itself. Field names are only taken over for the first table entry. The name of the old format is used as the table name.
- The general format attributes of the new format are taken from the current user profile. The used formats must be compatible with this user profile. The date and time specifications must be the same for all formats in the user profile. Furthermore, character set names in the formats must be specified in the user profile.
- If you are generating an **8-bit format**, the terminal groups “8-bit 9763 and 9758 and 8-bit printer” or “8-bit 9763 and 8-bit printer” must be set in the current user profile. The terminal used to generate the format must be an 8-bit terminal. All of the formats to be combined must either be 7-bit formats or have the same character set attributes. Otherwise, an error message is output and no format is generated.

  If you are generating a **7-bit format**, all formats to be combined must be 7-bit formats. 7-bit formats cannot be combined with 8-bit formats because the format generated could not be output.

  Further information is given in section “Summary of 8-bit support” on page 316.

– If you are not using dialog extension, none of the formats with dialog extension can be combined. If you are using dialog extension, any format can be combined provided the requirements listed below are met:
  – Formats may only contain one list. Therefore, only one format of the formats to be combined can contain a list.
  – The combined format has the title, top and bottom instruction lines, command and key list areas of the first format.
– A format produced by combination can be modified or extended in function 03 “Format Modification” like any other format.
– On completion of function 04, only the formats that you have properly stored and those that were displayed by IFG without any error message the last time they were stored will still exist.

```
I F G                              COMBINING FORMATS                          0401
                                  (For help: K3)
Would you like to view the directory first?  If so, mark and press SEND.

Name of the composite format: example
Version                     : V60a
Password                    :                Identifier:

        Format name        Version              after      times      Password
                                                blank lines
    01: address    @                              00          01
    02: invoice    a                              01          01
    03:                                           00          01
    04:                                           00          01
    05:                                           00          01
    06:                                           00          01
    07:                                           00          01
    08:                                           00          01


To check the input:          SEND              To store and display:        F1
To combine the next format:  F2                To select another function:  F3
```

Here the formats “address” and “invoice” are combined to form a single format. During the combining you can have the format displayed and stored as many times as you please (with F1).

If you now press SEND or F1, IFG checks whether the format complies with the specifications laid down in the user profile, issuing an error message if necessary. If the format is correct and you pressed “F1” (to store it), the combined format is displayed.

```
                              Delivery Note
Company
Name:
Street:
ZIP code:  .....
City:

Customer no.:  ............                              Date:

-------------------------------------------------------------------------------
Item no.  | Item description             | Quan-  | Unit    | Price
          |                              | tity   | pice    |
-------------------------------------------------------------------------------
 00000    |                              | 000000 | 0000000 | @@@@@@@@@@@@@@@ €
 00000    |                              | 000000 | 0000000 | @@@@@@@@@@@@@@@ €
 00000    |                              | 000000 | 0000000 | @@@@@@@@@@@@@@@ €
 00000    |                              | 000000 | 0000000 | @@@@@@@@@@@@@@@ €
-------------------------------------------------------------------------------
                                        Total:          @@@@@@@@@@@@@@@@ €
                                        Sales tax:       @@@@@@@@@@@@@@@ €
                                                          ----------------
                                  Total incl. sales tax:  @@@@@@@@@@@@@@@@ €
```

By pressing SEND again now you can continue using the combining function.

```
I F G                          COMBINING FORMATS                          0401
                             (For help: K3)
Would you like to view the directory first?  If so, mark and press SEND

Name of the composite format    : example
Version                         : V60A
Password                        :              Identifier:

        Format name        Version              after      times      Password
                                              blank lines
    01: address    @                             00          01
    02: invoice    A                             01          01
    03:                                          00          01
    04:                                          00          01
    05:                                          00          01
    06:                                          00          01
    07:                                          00          01
    08:                                          00          01


To check the input:          SEND              To store and display:        F1
To combine the next format:  F2                To select another function:  F3

5402: THE FORMAT HAS BEEN STORED IN YOUR FORMAT FILE
```

You can then modify or extend this format like any other by using function 03.

9 Creating table structures

Tables are used when generating addressing aids. They are of no relevance for FHS-DM. You can link the fields of a format that are accessible to the program to form tables. In COBOL, PL/I, Pascal, C and DRIVE programs these table structures are represented in the addressing aids by means of the appropriate language elements (e.g. in COBOL programs, by the OCCURS clause). If in addition you define a table name, you can also address the table as a whole.

At the same time IFG checks that certain conditions for tables are met:

– A table consists of up to 99 table entries.

– A table entry consists of one field or a number of fields that are adjacent in the addressing aid of the format (only fields “accessible” to the program).

– Each entry in a table has the same number of fields that are accessible to the program.

– Text fields that are between field entries or table entries and are not accessible to the program have no effect on the table structure.

– Corresponding fields in the table entries must have the same length and data type and also the same “NUM lock”, “number of decimal places”, “sign allowed” and “digit group” attributes.

– Tables must not contain other tables.

– Tables must not contain lists.

– Table entries may contain group fields. The group fields must not extend beyond one entry. Corresponding fields in different entries must have the same group field attribute. The group field attributes of the first table entry are automatically taken over for subsequent table entries if the format is stored with F1.

– For each table you can define one table name (up to 8 characters long).

9.1 Automatic generation of table structures using IFG function 04

Table structures can be generated automatically with IFG via function 04 “Combine Formats” when a format is used more than once with a repetition factor. In this case, the table structure is created automatically by IFG, as described above. The old format then corresponds to a table entry of the new one. The old format must not itself contain tables. Field names are only taken over for the first table entry. The name of the old format is used as the table name.

```
I F G                          COMBINING FORMATS                          0401
                             (For help: K3)
Would you like to view the directory first?  If so, mark and press SEND

Name of the composite format: table          Version:
                    Password:                Identifier:


        Format name        Version             after      times      Password
                                              blank lines
    01: title                                    00          01
    02: tab                                      00          13
    03:                                          00          01
    04:                                          00          01
    05:                                          00          01
    06:                                          00          01
    07:                                          00          01
    08:                                          00          01


To check the input:          SEND              To store and display:        F1
To combine the next format:  F2                To select another function:  F3
```

Now press the F1 key.

```
------------------------------------------------------------------------------------
Name                         | Address                                  | Date of birth
------------------------------------------------------------------------------------
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
                             | @@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@ | 00.00.0000
```

If you press SEND again, you can then add other formats.

```
I F G                          COMBINING FORMATS                              0401
                                 (For help: K3)
Would you like to view the directory first?  If so, mark and press SEND

Name of the composite format: table          Version: @
                    Password:                Identifier:


        Format name         Version                after       times       Password
                                                blank lines
    01: TITLE      @                                 00          01
    02: TAB        @                                 00          13
    03: footnote                                     00          01
    04:                                              00          01
    05:                                              00          01
    06:                                              00          01
    07:                                              00          01
    08:                                              00          01


To check the input:          SEND              To store and display:        F1
To combine the next format:  F2                To select another function:  F3

5402: THE FORMAT HAS BEEN STORED IN YOUR FORMAT FILE
```

You can then modify or extend this format like any other using function 03.

9.2 Explicit generation of table structures using function 01

With function 01 you create the format and in the image “Modify symbolic names of the fields” (see page 98) define the table attributes. In the first field of the first table element you must enter YES for “start of table” and the number of table elements for “table dimension”. It is also possible to define a table name.

```
I F G                    FIELD NAMES OF FORMAT: INVOICE                0306
                        (For help: K3) VERSION: 01
Field accessible to program: YES       Field type: INPUT FIELD     Length:  025

Field name: ITEM NUMBER                   longer name
Group name:                                          Start of group       : NO
                                                     End of group         : NO
Comment   :                                          Help panel      :
Table name: ITEMLINE     Table dimension: 04         Start of table         YES
                                                     End of 1st tab.elem. : NO
---------------------------   Line 001  to line 008  ---------------------------

Customer no.:  ############                          Date:  ##########

_____________________________________________________________________________
Item No.| Item description              | Amount  | Unit     | Price
|       |_______________________________|         | price    |
_____________________________________________________________________________
 #####  | ----------------------------- | #####   | ####### |    @@@@@@@@@@ €
_____________________________________________________________________________
To process next field:           SEND       To store format:                F1
To select another field: MAR and SEND       To move window forward:         F2
To abort function:               K2         To move window back:            F3
```

For the last field in the first table element, you must enter YES for “End of 1st tab.elem.”.

```
I F G                    FIELD NAMES OF FORMAT: INVOICE                  0306
                        (For help: K3) VERSION: 01
Field accessible to program: YES       Field type: INPUT FIELD     Length:  025

Field name: PRICE                          longer name
Group name:                                          Start of group       : NO
                                                     End of group         : NO
Comment   :                                          Help panel      :
Table name: ITEMLINE     Table dimension: 04         Start of table       : NO
                                                     End of 1st tab. elem.: YES
---------------------------   Line 001  to line 008  ---------------------------

Customer no.:  ############                    Date:  ##########

_____________________________________________________________________________
Item No.| Item description              | Amount  | Unit     | Price
|       |_______________________________|         | price    |
_____________________________________________________________________________
 #####  | --------------------------- | #####  | ####### |    @@@@@@@@@@ €
_____________________________________________________________________________
To process next field:           SEND       To store format:                F1
To select another field: MAR and SEND       To move window forward:         F2
To abort function:               K2         To move window back:            F3
```

When you have the format stored and displayed (F1 key), IFG checks that the table complies with all the rules and issues an error message if necessary. Group field attributes are automatically taken over from the first table entry and used for subsequent ones. Only if the table definition was error-free is the format output on the display terminal.

10 Key lists

You can assign character strings to F and K keys and store this assignment permanently as a key list in the format library. When the function keys are then activated, the assigned character strings are treated as if they were entered as commands in the command area. A key list contains a list of all the function keys from F1 to F24 and K1 to K14. When you press an F key, the input data of the format is also passed to the application together with the command.

Key lists are supported only by FHS-DE/DM. The name of a key list can be assigned in the format definition to each format. If no explicit assignment is made, the standard key list (IDHKEYnn) is assigned, which means that every format uses the same key list. A key list contains the following entry for each individual function key:

Fxx[command][designation].

Fxx	designates the function key
[command]	names the command assigned to this key
[code]	defines a freely-selectable short code (max. 12 characters)

The key list assigned to a format is shown in the key area of the mask by FHS-DE/DM. The key area is optional and consists of up to two lines. The display is implemented in the form “Fxx=code”. If no short code is defined for a key in the key list, that key is not displayed. This allows application programmers the option of displaying only important keys in the key area. If the display of all keys in the key list occupies the entire key area, the last complete entry is followed by up to three periods as an indication of further keys. The default assignments of the fields “Fxx” and “code” can also be modified by the user with IFG if required.

10.1 Key list administration

If you select the function “Key list administration”, the library name of the key list is output. You can then view or print the directory of the key list.

```
I F G                        KEY LIST ADMINISTRATION                        0901
       Key list library:
                          IFG.FORMATS

Do you wish to view or print out the directory first?
DDo you wish to print out all the key lists?

   Name of the key list: KEYLIS3                  Language : D (not for FHS-DM)
                Version: 001                      Password :

Do you wish to create,      view,      modify,
               delete   or  print out     a key list?

Do you wish to copy     a key list ?
           Old key list:                          Language : E
                Version:                          Password :
           From library: IFG.FORMATS

Do you wish to prepare  a key list  ?
          Into library:

To select the desired function: MAR and SEND       (To delete the key list: F1)
To select another function    : F3                              For help:  K3
```

The following functions are provided for all key lists:

– You can view the directory containing a list of all the key lists in your library. A specific key list can be selected by marking its name on the screen and pressing SEND.

– You can print the directory. A list of all key lists in your library is output to SYSLST.

– You can print all key lists. A listing of every key list in your library is output to SYSLST.

If you wish to use a special key list, you must first identify it. To do this, you must specify:

– the name of the key list and the letter for the optional alphabetic language identifier (e.g. D for German)

  The name of the key list may have a length of up to 7 characters (alphanumeric, #). If the language identifier is specified, the name of the key list must be 7 characters in length. If it is shorter, IFG will automatically pad the name with the hash character “#” up to the required length.

– the optional version identifier

  If you do not specify a version, IFG assumes the highest version.

This information can be entered either directly or by selecting the appropriate key list in the directory.

If you wish to use the functions modify, delete, and prepare a key list, and if a password has been assigned to the key list, this password must also be entered.

The following functions are then provided:

– You can create a key list.

  After selecting this function, you can create a key list and assign function keys. When you create a key list, IFG generates a member of type F that has the name of the key list.

– You can view a key list.

  After selecting this function, you can view the key list and the function key assignment.

– You can modify a key list.

  After selecting this function, you can modify the attributes of a key list, or create/modify the assignment of one or more function keys.

– You can delete a key list.

  To prevent a key list from being inadvertently deleted, IFG issues a warning before deleting the key list. If you really do wish to delete the key list, press the F1 key.

– You can print a key list.

  A listing of the key lists is output to SYSLST.

– You can copy a key list.

  The new key list takes the password of the key list being copied. If the library used is not the default library (defined in the user profile), you must then specify the name of the library.

– You can prepare a key list.

  This function generates an object module (of type “R”) in the library with the name of the key list. This object module is used by FHS-DE/DM. If the library used is not the default library (defined in the user profile), you must explicitly specify the name of the library.

10.2 Editing key lists

If you select the function “Editing a key list”, the library name, the name of the key list to be created, your language, and the version number are output.

```
I F G                          EDITING A KEY LIST                        0902

Key list library: IFG.FORMATS


 Name of the key list: KEYLIS3                 Language: E

              Version: 001

             Password:

  Key list identifier:                            (Up to 16 characters)

   Related help panel:

Edit a 8-bit key list         : YES
8-bit coded character set used:


To save data and edit the function keys: F1       To check and edit input: SEND
To select another function             : F3                       For help: K3
```

Using this function, you can

– assign a password to the key list (max. 8 positions) or change an existing password.

– assign a key list identifier to the key list. This identifier is used to mark the key list when viewing the directory. A key list identifier is not mandatory.

– specify a related help panel (up to 7 characters); see page 74.

– create an 8-bit key list. The 8-bit character set to be used is defined in the user profile (see section “Default values for the use of IFG” on page 183). If no character set is specified in the user profile, the default character set is used. Otherwise, the name of the 8-bit character set used is output. If the character set cannot be used, an error message is output. Please note that you cannot modify the character set of an existing 8-bit key list, nor can you convert the key list into a 7-bit key list.

Press F1 to begin assigning the function keys.

10.3 Editing function keys

If you press function key “F1” in the screen “Editing a key list”, the name of the key list and the language used are displayed. You can now assign a command and a text to the each key. You are not obliged, however, to assign a command to every key.

```
I F G                          EDITING FUNCTION KEYS                          0903

Name of the key list: KEYLIS3                                    Language: E
             Version: 001

F01 Command: HELP


                                          Text: HELP

F02 Command:


                                          Text:

F03 Command: EXIT


                                          Text: QUIT

To store the function keys: F1                                  Next keys: SEND
To select another function: F3                                   For help: K3
```

In accordance with the Alpha Style Guide, the function keys F1, F3, F12 and K3 are assigned default values (Help, Quit, Abort, and Display again, but these values may be changed.

When you activate a function key, the function key command is passed to the application by FHS-DE/DM via the command area generated using IFG (see page 77). This area is limited to one line on the screen. The exact length of the line depends on the amount of columns on the screen for which the key list is created.
If the command is too long, it is truncated.

The maximum length of the text is 12 characters. The function key number and the assigned text are inserted into the function key area of the format. This area is limited to 2 lines on the screen. The exact length of the line depends on the amount of columns on the screen for which the key list is created, so the text may be truncated (the last key detail is not output).

Example

“F1=Help F3=Quit F12=Abort K3=Display again”

By pressing SEND in the screen “EDITING FUNCTION KEYS”, you can page down as far as the K14 key. Once you have reached the last screen, you can press SEND to return to the first screen. You can store the assigned commands at each screen by pressing F1.

10.4 Viewing a key list/function keys

This function can be used to view the attributes of the key list and the function keys that were defined when creating the key list. The key list is output first.

```
I F G                                VIEWING A KEY LIST                    0904

Key list library: IFG.FORMATS


Name of the key list  : KEYLIS3           Language : E

             Version  : 001


Key list identifier   :                       (up to 16 characters)

Related help panel    :

Do you wish to use an 8-bit coded charracterr set: YES
8-bit coded character set used                  :


To view the function keys  :      SEND
To select another function :      F3                              For help: K3
```

When you have pressed SEND, the function keys are displayed:

```
I F G                              VIEWING FUNCTION KEYS                   0905

Name of the key list : KEYLIS3           Language : E

          Version    : 001


F01 Command : HELP

                                         Text : HELP

F02 Command :

                                         Text :

F03 Command : EXIT

                                         Text :


F                                                          Next keys:  SEND
To select another function:  F3                             For help:    K3
```

10.5 Modifying key lists/function keys

The procedure is the same as for creating/editing key lists (see page 154) and function keys (see page 155).

10.6 Directory of key lists

This directory contains details of all of the key lists in the specified format library. Both the library name and the name of the key list are output.

```
I F G                        DIRECTORY OF KEY LISTS                        0906

Key lists library:
                        IFG.FORMATS
TABLE    L  Version                  Identifier        Mod.  Date       Time
-------------------------------------------------------------------------------
KEYLIS3  E  1                                          0001  2005-06-22 10:24:10









******** End of list. To repeat from the beginning press SEND

To select a key list, mark the relevant line and press SEND      Other info: F2
To return to the current function without selection:   F3        For help:   K3
```

Key to above screen:

Name	Name of the key list
L	Language used
Version	Version of the key list; “@” means the latest version
Identifier	Key list identifier; if you specified this when creating/editing a key list.
Mod.	Variant number of the key list. This is set to 0001 when you are creating or copying. The number is increased by 1 each time you modify and save a key list.

Date and Time Specify when the key list was created or when it was last modified.

To obtain additional information, press the F2 key. This information is similar to that given in the format directory. A complete description is given in the chapter “Creating a format” on page 51. Press the SEND key to scroll up/down in the directory and F3 to exit and return from the directory. If you are in the “Edit” function, you can select a key list to be edited by marking it and pressing SEND.

10.7 Printing a key list

You receive the following information on every key list stored in the format file:

```
KEY LISTS LIBRARY                  name of the format file

NAME OF THE KEY LIST

LANGUAGE                           language identifier specified

KEY LIST VERSION

KEY LIST VARIANT                   key list variant

DATE OF CREATION OR LAST CHANGE    date and time

KEY LIST PREPARATION               date and time

USER ID AND                        identifier and account number under which
ACCOUNT NO.                        the key list was created or last changed

KEY LIST IDENTIFIER

PASSWORD                           specifies whether a key list is
                                   protected with a password or not

RELATED HELP PANEL

CODED CHARACTER SET                name of the coded character set

KEY      : F1
TEXT     : HELP
COMMAND  : HELP

KEY      : F3
TEXT     : QUIT
COMMAND  : EXIT
```

```
KEY       : F9
TEXT      : OPEN
COMMAND   : OPEN

KEY       : F10
TEXT      : ACTION
COMMAND   : ACTION

TASTE     : F12
TEXT      : ABORT
COMMAND   : CANCEL
```

11 Messages

Messages are only supported by FHS-DE/DM.

A message is a note that appears on the screen. There are various types of message. A message can provide information or warnings, and can also indicate errors or dangers. You can combine messages in a group, which is then called a message member. Message members are identified by a name (four letters or three letters and a digit) and an optional language identifier (one letter). A single message member can contain up to 1000 messages, which are numbered from 000 through 999. Messages are identified by means of the four characters of the message member, followed by the three digit number of the message.

Implicit messages are output by FHS without any action on the part of the application program, e.g. messages output by FHS when validating input fields. FHS-DE/DM provides standard message formats for implicit messages. These formats must either be copied to the format library of the application or be assigned as an alternative format library. The default messages have the following structure: IDHxnnn, where x can stand for the letter “F“, “I“ or “S“, and nnn is a three-digit number. If required, the message text can be modified by the user.
Explicit messages are messages initiated by the application program.

11.1 Message members administration

When you select the function “message members administration”, the library name for the message members is output.

```
I F G                  MESSAGE MEMBERS ADMINISTRATION                  0A01

Message members library:
                         IFG.FORMATS

Do you wish to view     or     print out  the directory first?
Do you wish to print out  all the message members?

 Name of message member: COMP                    Language: E (not for FHS-DM)
               Version: 4                        Password:

Do you wish to create,          view,             modify,
               delete    or     print out      a message member?

Do you wish to copy the message member?
        Old message member ;                                  Language:
                   Version :                                  Password:
              From library : IFG.FORMATS

Do you wish to prepare a message member ?
           Into library:

To select the desired function: MAR and SEND          (To delete the member: F1)
To select another function    : F3                                 For help: K3
```

The following functions are provided for all message members:

– You can view the directory. This contains a list of all the message members of your library. A specific message member can be selected by marking its name on the screen and pressing SEND.

– You can print out the directory. A list of all the message members of your library is printed to SYSLST.

– You can print out all message members. A listing is output to SYSLST for each message member of your library.

If you want to use a special message member, you must first identify it. In order to do this you must specify the following:

– The four characters that make up the name of the message member and the letter for the optional language identifier (e.g. E for English).

  The name of the message member can consist of four letters or three letters and a digit.

– The optional version number.

  If you do not specify a version, IFG uses the highest possible version available.

You can either input this information directly, or mark the appropriate message member in the directory.

If you want to use the functions modify, delete, and edit for a message member to which a password has been assigned, you must also enter this password.

The following functions are then offered:

– You can create a message member.

  When you select this function, you can create message members and messages. When a message member is created, IFG generates an element (type F) whose name comprises the 4 letters of message member name, three zeros, and the language identifier (if any).

– You can view a message member.

  When you select this function, you can view the message member and its messages.

– You can edit a message member.

  When you select this function, you can either modify the attributes of a message member, or create/edit a message.

– You can delete a message member.

  In order to prevent unintentional deletion, IFG issues a warning before deleting. If you do want to delete, press F1.

– You can print out a message member.

  A listing from the message member is printed to SYSLST.

– You can copy a message member.

  The new message member adopts the password of the message member to be copied. If the library used is not the default library (as defined in the user profile), you must also specify this library.

– You can prepare a message member.

  This function creates an object module in the specified library (type R member), which is used by FHS-DE/DM. If the library used is not the default library (as defined in the user profile), you must also specify the library. If the specified library does not exist, it is created. When a message member is created, IFG generates an member (type R) whose name comprises the 4 letters of the message member name, three zeros, and the language identifier (if any).

11.2 Creating/editing a message member

When you select the function “creating/editing a message member”, the library name, the name of the message member to be edited, its language, and the version are output.

```
I F G                          EDITING A MESSAGE MEMBER                  1002

Message members library: IFG.FORMATS



    Name of the message member: COMP                 Language: E

                       Version: 4

                      Password:

     Message member identifier:                      (Up to 16 characters)

Create a 8-bit message member : NO
8-bit coded character set used:



To save data and edit the messages: F1          To check and edit input: SEND
To select another function        : F3                        For help: K3
```

With this function you can:

– define a password. This restricts access to your message member.

– assign an identifier to the message member. This is used to identify a message member when viewing the directory. Specification of a message member identifier is optional.

– create an 8-bit message member. The 8-bit coded character set to be used is defined in the user profile (see page 183). If no character set is defined in the user profile, the default character set is used. In all other cases, the name of the 8-bit character set used is output. If the character set cannot be used, an error message is issued. Note that if an 8-bit message member already exists, you cannot modify the character set or convert the message member from 8-bit to 7-bit.

To begin editing a message, press F1.

11.3 Editing messages

When you select the function “editing messages”, the name of the message member, its language, version, and the first four letters of the message code are output.

```
I F G                              EDITING MESSAGES                          OA04

Name of the message member: COMP  Language: E  Version:

Do you wish to view          the directory of messages ?

Message number : COMP000     (3 digits)

Do you wish to copy          from message number COMP003 ?

Do you wish to delete        this message ?

Location   : 3          (1.Modal 2.Modeless 3.Message area)
Type       : 2          (1.Info  2.Error    3.Warning      4.Danger)
Help       : HELPBEG

Message    : WARNING:  In order to begin you must press ENTER.



To select the desired function: MAR and SEND    To create or edit message: SEND
To select another function    : F3                                For help: K3
```

With this function you can:

– view the directory of messages.

  Mark “view” and press SEND.

– create messages or edit existing ones.

  Here you must specify the message code. The message number comprises the name of the message member and a three digit number from 0 through 999. Entries cannot be made in the fields “Location”, “Type”, “Help”, or the message field until you have entered the message code. If the message code entered already exists, these fields are filled in by IFG, and you can then modify them. In the case of a new message code, you must fill in these fields yourself. By pressing SEND, you can check the message, while F1 saves it. A message can comprise up to 255 characters.

– copy messages.

  Messages can only be copied within a message member. You must enter the number of the message to be copied in the “copied from message code” field. You must specify the new message code in the “create or modify a message” field. Once you have

pressed SEND, the message is initialized along with the definitions of the message to be copied. You can update and save these definitions by pressing F1. The copied message is then output, and you can modify it.

– delete messages.

  You must enter the appropriate message code, select “delete” and press SEND.

The location of a message can be either modal or modeless, or it can be in the message area.

– A modal message is displayed in a box; this is the active box.
– A modeless message is displayed in a box; the user’s working area remains active.
– A message in the message area is displayed in an area reserved for messages at the bottom of the screen.

The message type specifies the category to which a message belongs. A distinction must be made here between information, warning, error, and danger.

– ”Information” provides the user with additional information on problem solving.
– “Error” refers to a situation that the user must take action to rectify. However, in this case, the danger is not immediate.
– “Warning” informs the user about situations or faults that present no danger.
– “Danger” refers to a situation that must be rectified immediately, as it poses a direct threat.

The message types “warning”, “error” and “danger” should be modal.

– A message can also be assigned a help panel, which can be requested when FHS outputs that message (7 positions); see page 74.
– Message text
  – A message can be generally understood as a series of text units. For the purposes of clarity and for practical reasons, a line feed mechanism is provided (but not in the message area).
    If a box contains the sequence “%%”, the subsequent characters are output at the beginning of the next line. The sequence “%%” is not output.
    The string “%%%%” produces the output of a blank line.

    Example

```
´text on line1.%%%%Text on line3.´ is output as
text on line1.

text on line3.
```

– If “&name” is found in the message text by FHS-DE/DM, and a field of that name is present in the format, FHS-DE/DM inserts the current contents of that field into the message text. The period following the variable “&name” is suppressed.

Examples

FHS-DE/DM

```
´Maximum is &name meters´ → ´Maximum is 100 meters
´&name.xy´ → ´valuexy´
```

FHS-DM

```
´&name#3´ →´value#3´
´&(name)#3´ → ´value#3´
´(&name#3)´ → ´value3´
´&undefinedvarname´ → ´???´
```

If the “&” character is used in the message, it must be duplicated. In addition, the following system variables may be used:
for FHS-DE: &ZPAR0, &ZPAR1, &ZPAR2, &ZPAR3

for FHS-DM: &SYS-PAR0, &SYS-PAR1, &SYS-PAR2, &SYS-PAR3,
and for compatibility reasons: &ZPAR0, &ZPAR1, &ZPAR2, &ZPAR3

11.4 Viewing an existing message member

With this function you can view the attributes of the message member and the messages that were defined when the message members were created.
The message member attributes are output first.

```
I F G                       VIEWING A MESSAGE MEMBER                      0A06

Message members library: IFG.FORMATS


     Name of the message member: COMP          Language: E

                   Version: 4


     Message member identifier:

 8-bit coded character set used:




To view the messages      : SEND
To select another function: F3                                  For help: K3
```

After you have pressed SEND, you can view the messages.

```
I F G                              VIEWING MESSAGES                          OA07

Name of the message member: COMP    Language: E    Version:

Do you wish to view the directory of messages?

Message number: COMP000     (3 digits)




Location   : 3         (1.Modal 2.Modeless 3.Message area)
Type       : 2         (1.Info  2.Error    3.Warning      4.Danger)
Help       : HELPBEG

Message    : WARNING:  In order to begin you must press ENTER.



To select the desired function: MAR and SEND          To view the message: SEND
To select another function    : F3                                For help: K3
```

In order to view a message, you must either enter the number of the appropriate message and press SEND, or select a message from the directory.

11.5 Directory of message members

This directory contains a list of all message members of the specified format library. The library name of the message member is output in each case.

```
I F G                    DIRECTORY OF MESSAGE MEMBERS                    OA03

Message members library: IFG.FORMATS

Name    L   Version                  Identifier        Mod   Date      Time
-------------------------------------------------------------------------------
COMP    E   4                                          0001  05-06-22  12:18:06









******** End of list. To repeat from the beginning press SEND

To select a member, mark the relevant line and press SEND       Other info: F2
To return to the current function without selection: F3           For help: K3
```

Key:

Name	Name of the message member
L	Language used
Version	Version of the message member; “@” stands for the highest possible version
Identifier	Identifier of the message member, if one was specified during creation or modification of the message member.
Mod	Variant number of the message member. When a message member is being created or copied, this is set to 0001. When a message member is being edited and saved, this number is incremented by 1.
Date and time	Specifies when you created or last edited the message member.

You can obtain further information by pressing F2. This information is the same as that given in the directory of formats. A complete description is provided in the chapter “Creating a format” on page 51. To scroll through the directory, press SEND; to quit and return from the directory, press F3.

If you wish to create a message, mark a message member and press SEND.

11.6 Directory of messages

This directory outputs the list of messages contained in a message member.

```
I F G                      DIRECTORY OF MESSAGES                          0A05

Message member name: COMP    Language: E    Version: 4

Message  Type       Message
number
------------------------------------------------------------------------------
 003   WARNING      In order to begin, you must press ENTER.
 005   ERROR        Press F1 to save your data.









******** End of list. To repeat from the beginning press SEND

To select a message, mark the relevant line and press SEND
To return to the current function without selection:  F3          For help: K3
```

It shows the name of the message member. For each message, the message number, the type of message, and the first 55 characters of the message text, followed by three periods (if the text is longer) are displayed.

If you wish to modify a message, mark the message number and press SEND.

If the list is not at the end, you can see following messages by pressing SEND. To return to the start of the list, press only SEND.

11.7 Printing a message member

The following information is provided for each message member saved to the format file:

LIBRARY FOR MESSAGE MEMBERS	Name of format file
NAME OF MESSAGE MEMBER	
LANGUAGE specified	Which language identifier has been
VERSION OF MESSAGE MEMBER	
VARIANT OF MESSAGE MEMBER	
MEMBER CREATION/MODIFICATION	Date and time
DATE OF PREPARATION	Specifies the date and time an object module was created for a message member
USER ID and ACCOUNT NUMBER modified	User ID and account number under which the message member was generated or last
IDENTIFIER OF MESSAGE MEMBER	Identifier of the message member
PASSWORD	Indicates whether or not the message is protected by a password
CODED CHARACTER SET	Name of the coded character set

The following information is printed out for each message of a message member: message code, location, type, help panel assigned, and text.

12 Printing, deleting or copying a format

These actions are only applicable to source formats (member type F).

```
I F G                          FORMAT ADMINISTRATION                          0601

 Name of format library:  IFG.FORMATS

 Do you wish to       view          the directory of the format library?
                      print out

 Do you wish to print out all the formats    in long form  ?
                                             in short form ?
_                                            variables only?

        Name of format:  CUSTOMER    Version:   01
        Password:
 Do you wish to        delete       the format?
 Do you wish to print out the format         in long form ?
                                             in short form?
                                             variables only?
 Do you wish to        copy       the format?
      Old format name:              Version:
              Password:
 Old format file/lib.:

To select the desired function:  MAR and SEND            (To delete format: F1)
To select another function:              F3                        For help:  K3
```

You can also perform the management functions Print, Delete and Copy using the IFG utility routine IFGFV, either

– in batch mode or

– from **any** terminal in interactive mode.

IFGFV also enables you to copy or delete all formats.

IFGFV is described in the chapter “Managing formats with IFGFV” on page 209.

12.1 Print format

You can have the following printed out:

– directory of the current format library

– all the formats in the current format library

  – with all the specs. (gen. format attributes, images of the formats, list of all the format names and format attributes) or

  – in “short” form (gen. format attributes, images of the formats)

  – only the variables that were defined in the formats. Three columns are printed per variable. These columns contain the variable type, variable length, and variable name.

– one format

  – with all the specs. (gen. format attributes, images of the formats, list of all the format names and format attributes) or

  – in “short” form (gen. format attributes, images of the formats)

  – only the variables that were defined in the formats. Three columns are printed per variable. These columns contain the variable type, variable length, and variable name.

IFG writes to the system file SYSLST, i.e. printing takes place once you have terminated the task (/LOGOFF command).

If you want the printout immediately, without terminating the task (and hence IFG as well), you should enter the following

system command before “printing”:

```
/ASSIGN-SYSLST TO-FILE=filename
```

and the following system commands after “printing”:

```
/ASSIGN-SYSLST TO-FILE=*PRIMARY
/PRINT-FILE FILE-NAME=filename,DELETE-FILE=YES,-
/LAYOUT-CONTROL=PARAMETERS(CONTROL-CHARACTER=EBCDIC)
```

The following applies to the printing of all formats (in short and long form):

– When printing 8-bit formats, you must ensure that all of the formats used within a single file are compatible. You should also check that the printer used supports the character set of these formats.
  Page 1 of the printout shows some of the general attributes of a format. The specifications requiring an explanation are described below.

– The format is printed out on page 2 with the variable fields represented by means of the appropriate field definition characters of the image draft of the current user profile. This allows you to read the basic attributes “protected”, “unprotected” and “numeric” (NUM lock) directly from the image.

N.B.

The field definition characters should be printable.

You can choose to have the print out in uppercase and lowercase letters (setting in the user profile) provided your printer also has lowercase letters in its character set.

Adjacent fields without an intervening blank are separated by the start-of-field character as in image drafting.

The grid with column and line numbers makes it easy to determine the position of a field (POS).

– Starting on page 3, all the fields of the format are listed with
  – their position (LN=line, CO=column)
  – their names
  – their length and
  – their attributes.

Fields whose attributes deviate from the specifications in the current user profile are marked with “(*)” or “(**)”.
Field attributes which cannot be selected in the user profile (e.g. blanked, exit code etc.) are marked with (**). This list of fields is not printed out in the “short form”.

The meaning of the column headings in the list is as follows:

NO	Sequence number of the format (the formats are arranged alphabetically).
FORMAT	Name of the format
VERSION	Version of the format
MOD	Modification counter of the format; the counter is incremented by 1 each time the format is modified.
CREATION DATE	specifies when the format was created or last modified.
DATE OF FORMAT PRPARATION OBJECT MODULE ADDRESSING AID	specifies when the object module and addressing aid were generated for a format.

```
USER ID             User ID and account number under which the format was
ACCT. NO            generated or last modified.

IDENTIFIER          Format identifier (comment)

PASSWORD             indicates whether the format is protected by a password
                     or not.
```

12.2 Delete format

The member (of type F) is deleted.

Formats with the same name in other format libraries are not deleted, nor are the formats in the format application file and associated addressing aids in the source program library, macro library or PLAM library (see page 113ff). The format application file can be deleted with LMS.

12.3 Copy format

You can copy a format

– from another format file (created with IFG V4 or V5)

– from another format library (created with IFG V6 or later)

– within the current format library

The new format receives the password of the old format, but is not yet prepared for use, even if the old (copied) format had been prepared before it was copied.

13 User profile administration

13.1 Select user profile

In order to make a user profile the current user profile, enter the name of the desired user profile and press SEND. Alternatively, you can select the desired user profile from the directory.

```
I F G                    USER PROFILE ADMINISTRATION                    0701

Name of format library: IFG.FORMATS

Do you want to  _view_____  the directory of all user profiles?

Name of current user profile :
Password for user profile    :

Do you want to  _view_____  or _modify____  or _delete____ the user profile?

Do you want to  _view_____  or _modify____  part of the user profile?


Do you want to  _set up___  the I F G standard profile?

Do you want to  _copy_____  an existing user profile?
from the format library:
Name of old user profile     :
Password of old user profile :

To activate current user profile         SEND   To abort function:         K2
To select the desired function:  MAR and SEND   (To delete user profile:   F1)
To select another function:              F3     For help:                  K3
```

Mark “view” and press SEND.

```
I F G                        DIRECTORY OF USER PROFILES                        07OD
     Format library: IFG.FORMATS

 Profile   Password
--------------------------------------------------------------------------------
USERPRO    NO
DSS9750    NO
DSS9755    NO
DSS9763    NO
_________
_________
_________
_________
_________
_________
_________
_________
_________
_________
_________
******** End of list. To repeat from the beginning press SEND
To select a user profile, mark the relevant line and press SEND
To return to the current function without selection:  F3          For help:  K3
```

13.2 View user profile

The “view the user profile” function shows you the user profile without allowing you to modify it. None of the fields in the user profile can be overwritten.

You can ask to see the whole user profile or only a particular part of it. Which part of the user profile contains which format attributes is shown in the following section.

13.3 Modify user profile

In the “modify user profile” function you can call the entire user profile and make changes to it or select only a part of the user profile for modification. The table on the next page shows you which part of the user profile contains which format attributes.

You can only use characters from the character set EBCDIC.DF.03 in the user profile, even when creating an 8-bit format.

What specifications does the user profile contain?

The specifications in the user profile are listed on the following screens:

– default values for the use of IFG

– terminal specifications for the formats

– display attributes of the formats

– editing attributes of the formats

– display attributes of the field types

– input/validation attributes of the field types

– editing attributes of the field types

– addressing aid specifications

– application library specifications

– specifications for field attribute blocks in the data transfer area

– specifications for global attribute blocks in the data transfer area

– default values for FHS-DOORS conversion (part I)

– default values for FHS-DOORS conversion (part II)

The values for the display, editing, input and validation attributes are transferred to a format when the format is saved on creation or modification.
As a result, changes in the user profile have no effect on formats already stored. It is only when a new format is created or an existing one is modified that the “new” user profile takes effect (when “editing formats”, only in lines where changes were made).

Notes on special characters

Field definition characters are not part of the format, i.e. the fields of a previously created format will be displayed with the modified field definition characters (if any). If the text in a previously created format contains characters that you have defined as special characters in the edited user profile, these special characters will not be initially converted. It is only when a line of the image containing such a special character is edited that the special character will also converted.

The following actions can be performed at the format level:

- You can change all profile values for the following formats:
  - terminal specifications, see page 131.
  - display attributes of the format, see page 136.
  - editing attributes of the format, see page 138.
- You can change only some profile values for the following formats:
  - application library specifications, see page 113ff.

The following actions can always be performed at the field level:

- All profile values can be changed for the following formats:
  - display attributes of field types, see page 94.
  - input/validation attributes of field types, see page 107ff.
  - editing attributes of field types, see page 109.
- The original values can be restored; see page 132.

13.3.1 Default values for the use of IFG

```
I F G                  VIEWING YOUR USER PROFILE USERPRO                    0703
                       DEFAULT VALUES FOR THE USE OF IFG

Password for the user profile:

Special characters for image drafting (if not required, blanks):
  _ input field                 >   repetition character          delete line
  # input field, numeric           start-of-field char.          repeat line
  @ output field (protected)    "   field alignment
    single choice field             multiple choice field         list

In IFG, lowercase letters in text fields are to be converted to upper case
- during image drafting       : NO              - in printouts            : YES

Image drafting/modification
Start with large window       : NO                 Retain field atttributes: NO
Max. number of used lines/columns in formats (max. 43 / 132)       : 23 / 080

Format display: with char. set attributes:NO      Free cursor movement  : YES
Character set library:


To check any changes:  SEND                                        For help:  K3
To store your input:   F1                                 To abort function:  K1
```

– You can define a password for the user profile which is up to eight characters long. When the user profile is modified or copied, this password must be specified. However, the user profile can be viewed without entering the password.

– You can modify or define the special characters for image drafting. In the “image drafting" and “image editing" functions, these characters appear above the window. The special characters can also be used in the list title (for special output fields only) or in records of the list (see also pages 91 and 92).

  Characters you do not require can be overwritten with a blank. You should not overwrite the special characters for input and output fields with a blank, however, as this may have undesirable consequences.
  The characters defined here will also be used as fill characters when the formats are printed out. They should therefore be printable.

  Example

  On page 65, the characters %, <, >, and + were selected as special characters. Only use characters that you do not require in text fields. If you wish to use the character # in text fields, you might use $ as a special character.

- You can define
  - that during image drafting lowercase letters in text fields are to be automatically converted to upper case (not to be confused with the function: conversion of lowercase to uppercase letters in input fields, see page 110
  - that formats are to be printed out in uppercase and lowercase letters provided that your printer's character set includes lowercase letters.
- You can define
  - that the large window is to appear immediately in the "image drafting" and "image modification" functions. In the "large window" the last line is not displayed as this line is reserved for IFG messages.
  - that the field names and field attributes are to be retained in the "image modification" function.
  - the maximum numbers of lines and columns in a format. If you enter more lines/columns during format drafting or modification than are permitted by the user profile, an error message is issued by IFG. If the number of lines/columns entered is greater than the screen size of the format (see section "Display attributes of the formats" on page 186), no notification is given.
- You can define
  - whether the formats are to be displayed with the specified ICE character sets or the standard character set for the terminal.
  - the character set library containing the ICE character sets you wish to use. If you have specified format display "with character set attributes", you must make an entry here; otherwise, IFG will issue an error message.
  - whether you wish to position the cursor in every field when the format is displayed (free cursor movement) or whether you only wish to position the cursor in unprotected fields. In the event of "free cursor movement", you can position the cursor at the start of the format and print out the entire format.

13.3.2 Terminal specifications

```
I F G                  VIEWING YOUR USER PROFILE USERPRO                    0704
                  DEFAULT VALUES FOR TERMINAL SPECIFICATIONS


 Format is to be suitable for use on the following terminal group (+ printer)
 8-Bit 9763                   : NO          9750 in multiple field mode : NO
 8-Bit 9763,9758              : NO          3270,9763,9755              : NO
 9763                         : NO          3270,9763,9755,9750         : NO
 9763,9755                    : NO          3270,9763,9755,9750         : NO
 9763,9755,9750               : YES         9763,9755,9750,816x         : NO


Default device when format is used:
 - display terminal                - printer
   9763: NO     8162: NO             9001     : NO     9004: NO     9022: NO
   9755: NO     8161: NO             9001-8931: NO     9011: NO     PCL : NO
   9750: YES    3270: NO             9002/9003: NO     9012: NO     3287: NO
                                     9003     : NO     9013: NO

If the default device is a printer, is it accessed through
 - a printer controller: NO          - or a 9763, 9755, 9750: NO   terminal?
 - RSO                 : NO

To check any changes:    SEND                                    For help:  K3
```

For further information, refer to the section “What effect does the terminal type have on the use of formats?” on page 32ff, and the section “Modify terminal specifications” on page 134.

Note

Unlike the function “Modify terminal specifications of format”, the user profile administration function allows the modification of each terminal group.

13.3.3 Display attributes of the formats

```
I F G                VIEWING YOUR USER PROFILE USERPRO                    0705
                DEFAULT VALUES FOR FORMAT DISPLAY ATTRIBUTES

    Tuning facility for  8161, 8162  during format application
    (less fields permitted per line, but shorter transmission times):

    ICE Character sets used:
          0.: *STD          1.:              2.:              3.:
          4.:               5.:              6.:              7.:


    Holes color   : U (W=white, G=gray, U=undefined)

    Screen dimensions  : 24 x 080 (lines x columns)

    Coded character set:

    Global mode of format is Latin:

    Change-code key is locked     :

To check any changes: SEND                                   For help:  K3
To store your input:  F1                                 To abort function:  K2
```

For more information on this screen, refer to the section “Modify display attributes of the format” on page 136.

By default, no value is assigned to the coded character set. If you create a 7-bit format, the name of the coded character set is ignored. It is not checked whether the specified coded character set can be used. This is checked when the format is created or modified. If no value is specified, the coded default character set is used.

Arabic/Farsi formats

The screens to support Arabic/Farsi formats are only output in English and differ from standard screens with respect to two additional fields:

```
Global mode of format is latin: Y/N
```

```
Change-code key is locked: Y/N
```

More information can be found on page 243.

13.3.4 Editing attributes of the formats

```
I F G                VIEWING YOUR USER PROFILE USERPRO                    0706
                DEFAULT VALUES FOR FORMAT EDITING ATTRIBUTES


Account is to be taken during format use of field alignment and fill characters?
                   in input: YES                 in output: YES

Representation of arithmetic fields
           decimal separator: .             digit separator: ,
Representation of the time
                with seconds: YES                separator: :
Representation of the date
sequence of day, month, year: YMD (Y,M,D)        separator: .
            year, two digits: YES              four digits: NO

Dialog extensions required? : YES

Representation of undefined values
                    any char.     arithmetic    alphabetic        date/time
 character              .             .             .                .
 hexadecimal           00            00            00               00

To check any changes: SEND                                        For help: K3
To store your input:  F3                                  To abort function: K2
```

For further information on this screen, refer to the section “Modify editing attributes of the format” on page 138.

Note that the default value for dialog extension is “NO´.

13.3.5 Display attributes of the field types

The following screen lists the default values for the display attributes of fields assigned by IFG when a format is created.

```
I F G              DISPLAY ATTRIBUTES FOR EACH FIELD TYPE                0707

                         Text-         Inmput     Input field    Output
                         field          field       numeric      field
------------------------------------------------------------------------------
Intensity:   bright    : NO            YES          YES           NO


Flashing               : NO            NO           NO            NO
Printable              : YES           YES          YES           YES

Underline / italics    : NO            NO           NO            NO
Inverse                : NO            NO           NO            NO

Color        (01234567) : 0            0            0             0
Character set          : 0 *STD        0 *STD       0 *STD        0 *STD




To check any changes: SEND                                      For help: K3
To store your input:  F3                               To abort function: K2
```

For more information on fields, see the section “Modify display attributes of fields” on page 94ff.

Note that all fields are visible by default.

13.3.6 Input/validation attributes of the field types

The following screen shows you the default values for the input and validation attributes of fields assigned by IFG when a format is created.

```
I F G                  VIEWING YOUR USER PROFILE USERPRO                    0708
                  INPUT/VALIDATION ATTRIBUTES FOR EACH FIELD TYPE

                             Text         Input      Input field      Output
                             field        field        numeric        field
---------------------------------------------------------------------------
Accessible to program      : NO           YES          YES            YES
Protected                  : YES          NO           NO             YES
Automatic input            : NO           NO           NO             NO
Selectable                 : NO           NO           NO             NO

Field data type: arithmetic: NO           NO           NO             NO
                 alphabetic: NO           NO           NO             NO
                 any char. : YES          YES          YES            YES

Number of decimal places   : 00           00           00             00
Signed                     : NO           NO           NO             NO
Digit grouping             : NO           NO           NO             NO


To check any changes: SEND                                     For help: K3
To store your input:  F3                               To abort function: K2
```

For more information on fields, see the section “Modify input and validation attributes of fields (part I)” on page 103ff.

13.3.7 Editing attributes of the field types

This screen displays the defaults for the editing attributes of the various field types.

```
I F G                VIEWING YOUR USER PROFILE USERPRO                    0709
                   EDITING ATTRIBUTES FOR EACH FIELD TYPE

                             Text        Input      Input field    Output
                             field       field        numeric      field
-------------------------------------------------------------------------------
Alignment          left   :  NO  /NO     YES /YES     NO  /NO      NO  /NO
(input/output)     right  :  NO  /NO     NO  /NO      YES /YES     NO  /NO
                   none   :  YES /YES    NO  /NO      NO  /NO      YES /YES
Input/output fill chars. :       /           /         0  / .          /

Leading zeros suppression:   NO          NO           NO           NO
Floating sign             :  NO          NO           NO           NO

Uppercase letters only    :  NO          NO           NO           NO

Processing by exit rout. :   NO          NO           NO           NO

Processing for und. val. :   NO          NO           NO           NO
Fast detection           :   YES         YES          YES          YES

To check any changes: SEND                                        For help: K3
To store your input:  F3                                  To abort function: K2
```

For more information on fields, see the section “Modify editing attributes of fields” on page 109ff.

13.3.8 Addressing aid specifications

This section is not applicable to FHS-DM formats.

```
I F G                  VIEWING YOUR USER PROFILE USERPRO                    070A
                  DEFAULT VALUES FOR FORMAT PREPARATION FOR FHS
                        ADDRESSING AID SPECIFICATIONS

Programming language
   COBOL : YES       ASSEMBLER: NO         RPG  : NO          PL/I   : NO
   Pascal: NO        C        : NO         DRIVE: NO          FORTRAN: NO

Structure of the data transfer area
    separate attribute blocks and field contents : YES
    aligned,   with    attribute fields          : NO
    unaligned, with    attribute fields          : NO
    unaligned, without attribute fields          : NO

Additional specifications when using COBOL, PL/I or DRIVE:
    Initial value of level number: 00     Increment: 00        (00=Standard)

Additional specifications for COBOL:
    one addressing aid     : YES     'PIC 9' for numeric fields         : NO
    two addressing aids    : NO       name prefixes, input/output       :  /
    binary for comp. clause: NO       field attribute blocks, short form: NO

To check any changes: SEND                                        For help: K3
To store your input:  F3                                  To abort function: K2
```

Here you define

– the programming language in which the application program is written.

– the structure of the data transfer area; the default is “separate attribute blocks and field contents”.
In the case of data transfer areas without attribute fields, only the field contents may be modified dynamically. It is not possible to modify attributes dynamically. For data transfer areas with attribute fields, some basic field attributes can also be modified. For data transfer areas with separate attribute blocks and field contents, all attributes can be modified dynamically (e.g. arithmetic fields, date fields). For more information on the data transfer area, refer to the FHS manual “Format Handling System for openUTM, TIAM, DCAM´.

Notes

Addressing aids for the data transfer area with separate attribute blocks and field contents are aligned with integral boundaries. If the field attribute block contains the attribute group “field length”, all field attribute blocks are aligned with halfword integral boundaries.

The data transfer area with separate attribute blocks and field contents is not supported by FHS for the terminal group “9750 in multiple field mode”.

COBOL programs support all four data transfer area structures. For the data transfer area with separate attribute blocks and field contents

– you can specify an initial value for the level number and a value for the increment;

– only one addressing aid is created;

– you can use a binary clause for all comp. clauses;

– you cannot specify “YES” for “binary for comp. clauses” in the case of UTM applications, since the bits are incompatible and result in errors;

– a “PIC 9” clause is created for an arithmetic field. If decimal places have been specified for the field, the character “V” is inserted at the appropriate point in the mask character string. If a sign is permitted, an “S” is prefixed to the mask string. A “SIGN IS TRAILING SEPARATE” clause is used for the representation of the sign. This clause is issued at a level above the field contents declarations.

– two addressing aids for input and output are created; they differ in terms of their prefix (e.g. “i” / “o”).

– the field attributes can be generated in short or long form. In short form, the field attribute blocks are only generated as a table; in long form, the field attribute blocks are additionally redefined as a table.

For the other data transfer areas

– either one or two addressing aids can be generated;

– PIC 9 or PIC X can be generated for numeric fields, in which case, however, characters may be entered in the program that are not numeric in a COBOL context; this may result in data errors when the application program executes.

The generation of both groups and tables is supported. For tables, if you have defined a table name, you can address the entire table using the name “tablename-TAB”.

You can use a binary clause for all comp. clauses and you can specify an initial value for the level number and a value for the increment.

Assembler programs support all four data transfer area structures. For the data transfer area with separate attribute blocks and field contents

– only one addressing aid is generated;

– no language elements are available for representing arithmetic fields; they are declared as a character string;

– the global attribute block is defined as a DSECT only once per compilation unit;

– the field attribute block is defined as a DSECT once only per format; the remainder of the addressing aid is generated optionally as a CSECT or DSECT, the presetting for the &SECT parameter being CSECT. The DSECT and CSECT are given the same name

as the SECTION of the current macro expansion. Therefore, when a DSECT or CSECT is generated, the macro expansion must take place in a SECTION of the same type. Further details can be found in the “ASSEMBH (BS2000)“ manual.

The generation of groups is supported, but not the generation of tables. The field names, group names and names of the addressing aids must be defined so as to be unique up to the seventh character.

RPG programs do not support the data transfer area “aligned, with attribute fields”. Two addressing aids are always generated. For the data transfer area with separate attribute blocks and field contents

– arithmetic fields are mapped using RPG language elements;

– the generation of groups is supported;

– field names and group names must be assigned as otherwise errors may occur during compilation,

– the first five characters of field names and the first four characters of addressing aids must be unique.

– unstructured tables are supported in which the first and last elements coincide. This kind of table is made up of one field repeated n times. A table name must be specified; any field names assigned are ignored. The first two characters of the table name must be unique.

– formats containing structured tables cannot be processed. The creation of addressing aids for such formats is aborted with an error message.

For the other data transfer areas

– the generation of groups and tables is not supported;

– the first five characters of the field names and the first seven characters of the addressing aid names must be unique.

PL/I programs do not support the data transfer area “aligned, with attribute fields”. Only one addressing aid is generated. The generation of groups and tables is supported. For the data transfer area with separate attribute blocks and field contents

– the PICTURE attribute is mapped for arithmetic fields. If decimal places were specified for a field, a “V” is inserted at the appropriate point in the mask string. If a sign is permitted, an “S” is appended to the mask string on the right.

– you can specify an initial value for the level number and a value for the increment.

Pascal programs do not support the data transfer area “aligned, with attribute fields”. Only one addressing aid is generated, divided into BODY and SPEC. The generation of groups and tables is supported. For the data transfer area with separate attribute blocks and field contents

– no language elements are available for representing arithmetic fields; they are declared as a character string.

C programs do not support the data transfer area “aligned, with attribute fields”. Only one addressing aid is generated. The generation of groups and tables is supported. For the data transfer area with separate attribute blocks and field contents

– no language elements are available for representing arithmetic fields; they are declared as a character string.

DRIVE procedures support only the data transfer area with “separate attribute blocks and field contents”. Only one addressing aid is generated. The field attribute blocks are redefined in tabular form. Arithmetic fields are mapped onto the data type NUMERIC, date fields with a four-digit year specification onto data type DATE. The generation of groups and tables is supported. You can specify an initial value for the level number and a value for the increment.

Fortran programs support only the data transfer area with “separate attribute blocks and field contents”. Only one addressing aid is created. The generation of groups and tables is not supported. The name of the addressing aid is 7 characters long; field names may be up to 12 characters long.

Examples of addressing aids are given in the section “Examples of addressing aids” on page 255.

13.3.9 Application library specifications

```
I F G                VIEWING YOUR USER PROFILE USERPRO                    070B
                DEFAULT VALUES FOR FORMAT PREPARATION FOR FHS
                     APPLICATION LIBRARY SPECIFICATIONS


 Format preparation for ´fast formatting´              : yes
 Format preparation for FHS-DE compatible with IFG V8.0: no

 Checks for the use of formats with FHS-DM:
 -  field names on all fields accessible to the program     : no
 -  names of variables follow the syntax of dialog variables: no
 -  names of variables follow the syntax of S variables     : no


 Preparation of:          In libraries:
 - formats        : yes
 - addressing aids: yes

 Library element type used for storing addressing aids


To check any changes: SEND                                      For help: K3
To store your input:  F3                                 To abort function: K2
```

– Format preparation for “fast” formatting

  The following rules apply when you use the “fast” formatting function (considerably enhanced performance on using the format when running the application):

  – A format prepared for “fast” formatting may only be used on a terminal other than the one specified in IFG as the "default terminal type" subject to certain restrictions (see table below).

  – A switchover between a data display terminal and a printer is not possible.

  – When formats are used with FHS implementing “fast formatting”, the meaning of the DEVICE operand in the MDCBL macro is reduced merely to a validity check.

  – “Fast” formatting is not possible for 3287 printers, 8-bit formats, or formats with dialog extension.

  – IFG V8.0 resets the field name of the input field of the command area of formats with Dialog Extensions to “CMDAREA”.

N.B.

IFG has the following default values:

– format preparation for fast formatting and

– default for application: 9750.

If you have terminals of different types connected and their formats are to be used on all the terminals, or if you want to change the terminal type in the application program, you should either

– change the default value for “fast” formatting to "NO", or

– change the terminal type so that the format can be output on all the desired terminals.

The following tables show which formats can be output on which terminals when “fast” formatting is used. These tables apply for full formats or for partial formats if the same default is defined for application for all partial formats.

	The format can be output on														
Default for application	**8160 9750**	**9755**	**9763**	**3270**	**9001**	**9001 -31/ -8931**	**9002 9003**	**9003**	**9004**	**9011**	**9012**	**9013**	**9022**	**PCL**	**3287**
9750 8160	yes	yes	yes	no	no	no	no	no	no	no	no	no	no	no	no
9755	no	yes	yes	no	no	no	no	no	no	no	no	no	no	no	no
9763	no	no	yes	no	no	no	no	no	no	no	no	no	no	no	no
3270	no	no	no	yes	no	no	no	no	no	no	no	no	no	no	no
9001	no	no	no	no	yes	yes *)	no	no	no	yes *)	yes *)	no	no	no	no
9001 -31/ -8931	no	no	no	no	yes	yes	no	no	no	yes *)	yes *)	no	no	no	no
9002 9003	no	no	no	no	no	yes *)	yes	yes *)	yes *)	yes *)	yes *)	yes *)	yes	no	no
9003	no	no	no	no	no	no	no	yes	no	no	no	no	no	no	no
9004	no	no	no	no	no	no	no	no	yes	no	no	no	yes *)	no	no
9011	no	no	no	no	yes *)	yes *)	no	no	no	yes	no	no	no	no	no
9012	no	no	no	no	no	no	no	no	no	no	yes	no	no	no	no
9013	no	no	no	no	no	no	no	no	no	no	no	yes	no	no	no
9022	no	no	no	no	no	no	no	no	yes	no	no	no	yes	no	no
PCL	no	no	no	no	no	no	no	no	no	no	no	no	no	yes	no
3287	no	no	no	no	no	no	no	no	no	no	no	no	no	no	yes

no means
- for data display terminals: formatting is aborted with a return code;
- for printers: formatting is performed, but the printout may contain errors.

*) Restriction: The attributes “bright”, “flashing”, “wide” and “tall” cannot be used. All fields must be assigned the attribute “normal”.

– Format preparation for FHS-DE compatible with IFG V8.0 .

  FHS-DE V8.0 only supports field names with a maximum of eight characters. Longer field names that are specified in IFG for addressing aids are truncated during format preparation to 8 characters.
  Starting with FHS-DE V8.1, FHS-DE supports longer file names that are no longer truncated by IFG. This could lead to incompatibilities between existing FHS-DE applications if a reference is made to such field names for cursor positioning or within the messages of message members.

  To ensure that your format is processed by IFG as before and to avoid having to modify your application, you can set the value for format preparation compatible with IFG V8.0 to “YES” in the user profile.

  IFG V8.0 also forces the field name of the input field of the command area of formats with Dialog Extensions to “CMDAREA”.

– Checks for the use of formats with FHS-DM

  FHS-DM stores field contents in variables that have the same names as the field names. A name should therefore be defined for every field that is accessible to the program. Otherwise, the application program will be unable to read both the user input as well as the output information in format fields.

  If the “Checks” options are set to “YES”, IFG prepares the format only if it can be used with FHS-DM. This means that all field names must be accessible. If field names are missing, IFG outputs the format without attributes, with the fields in question highlighted.

  If mandatory variables are missing or if the defined field names or control variables do not follow the prescribed syntax rules, FHS-DM cannot use the format.

  Depending on whether the format is used in a program or in a procedure, you must verify that the names comply with the syntax rules of dialog variables or SDF variables.

  This can be done by setting the appropriate value in the user profile. IFG will then verify that the prescribed syntax rules are observed both when creating the variable names and when preparing the format.

– Library names with format preparation

  If you do not require the preparation options for both formats as well as addressing aids, you can deactivate one of the options (e.g. FHS-DM does not require addressing aids). The preparation option is set for both formats as well as addressing aids by default; however, these values can be changed later during format preparation.

  You can specify the module library for the format application file as well as the source program or macro library for the addressing aids.

IFG will then transfer these library names to the appropriate IFG screen at format preparation. The given entries can be modified there as well.

– Library element type used for storing addressing aids

  By default, Assembler addressing aids are stored in PLAM libraries as members of type “M” (macros). All other addressing aids that are stored in PLAM libraries (e.g. C, DRIVE, Fortran addressing aids) are maintained as members of type “S” (source). Instead of the default types “M” and “S”, you may also specify your own member types. IFG will then store the addressing aids generated during format preparation in library members of these types.

  The member types you specify must have a length of at least two characters. The maximum permissible length is eight characters. They may consist of the characters A-Z, 0-9, $, # and @ and must not begin with $ or “SYS”.

13.3.10 Default values for FHS-DOORS conversion (part I)

In this screen you can specify the default values for FHS-DOORS conversion.

```
I F G                  VIEWING YOUR USER PROFILE USERPRO                    07OF
                  DEFAULT VALUES FOR FHS-DOORS CONVERSION (PART I)

 Perform FHS-DOORS conversion during format preparation? : NO
 Text separators are represented by 00 blanks  (0 for no separators)
 Language used in the conversion process:      INTERNATIONAL_____________________

 Map the key list keys texts to push-buttons/menu zone? :  NO /NO

 Add push-buttons with the following action labels for keys:
 DUE1: ____________  DUE2: ____________   K01: ____________   K02: ____________
 K03 : ____________  K04 : ____________   K05: ____________   K06: ____________
 K07 : ____________  K08 : ____________   K09: ____________   K10: ____________
 K11 : ____________  K12 : ____________   K13: ____________   K14: ____________
 F01 : ____________  F02 : ____________   F03: ____________   F04: ____________
 F05 : ____________  F06 : ____________   F07: ____________   F08: ____________
 F09 : ____________  F10 : ____________   F11: ____________   F12: ____________
 F13 : ____________  F14 : ____________   F15: ____________   F16: ____________
 F17 : ____________  F18 : ____________   F19: ____________   F20: ____________
 F21 : ____________  F22 : ____________   F23: ____________   F24: ____________

To check any changes: SEND                                          For help: K3
To store your input:  F3                                  To abort function: K2
```

The program FHS-DOORS-LC can be used to convert an FHS format application file into an FHS/FHS-DOORS format application file. This means that the format definitions are entered in the format application file in a form that can be used by FHS-DOORS (see the “FHS-DOORS (BS2000/OSD, MS-Windows)” manual).

You can call the FHS-DOORS converter for every format preparation and thus ensure that the format definitions for FHS-DOORS are always up-to-date whenever you modify or prepare your formats with IFG.
In addition to the object format (member type “R”), the converter generates an SDC file (member type “S” with “.SDC” as the suffix; SDC=Semantic Description Compressed).

The following options for conversion can be specified in this screen (the options correspond to the parameters of the MODIFY-FHS-DOORS-OPTIONS in the FHS-DOORS-LC program):

– Text separators are used to split predefined strings into multiple graphical objects. The string is split at each text separator. This allows the alignment of the generated formats to be retained regardless of which character set is used.
  This particular feature is specially intended for fixed fields that are used as titles, e.g. to mark columns in a format.

– Language used in conversion process

  If you wish to specify a language other than INTERNATIONAL, a corresponding conversion table is used to convert the values of static fields.

  The following languages are available:

  NONE, INTERNATIONAL, DANISH, ENGLISH-UK, ENGLISH-USA, FRENCH, FRENCH-BELGIAN, GERMAN, ITALIAN, NORWEGIAN, SPANISH, SWEDISH, SWISS

  During format preparation, FHS-DOORS-LC searches for the corresponding key list for a non-help format; see also page 117.

– Mapping of key lists to the push-buttons/menu zone.
  The texts of the keys in the key list are used as labels for the new push-buttons or menu entries.

– Definition of additional push-buttons for keys

  A push-button is generated in the format for every key for which a label is defined in this screen. Pressing the key can be simulated by clicking the corresponding push-button.

13.3.11 Default values for FHS-DOORS conversion (part II)

```
I F G                VIEWING YOUR USER PROFILE USERPRO                    070G
                DEFAULT VALUES FOR FHS-DOORS CONVERSION (PART II)




Labels of menu items in the 'actions' menu for keys:
 DUE1: ___________  DUE2: ___________   K01: ___________   K02: ___________
 K03 : ___________  K04 : ___________   K05: ___________   K06: ___________
 K07 : ___________  K08 : ___________   K09: ___________   K10: ___________
 K11 : ___________  K12 : ___________   K13: ___________   K14: ___________
 F01 : ___________  F02 : ___________   F03: ___________   F04: ___________
 F05 : ___________  F06 : ___________   F07: ___________   F08: ___________
 F09 : ___________  F10 : ___________   F11: ___________   F12: ___________
 F13 : ___________  F14 : ___________   F15: ___________   F16: ___________
 F17 : ___________  F18 : ___________   F19: ___________   F20: ___________
 F21 : ___________  F22 : ___________   F23: ___________   F24: ___________


To check any changes: SEND                                   For help: K3
To store your input:  F3                                 To abort function: K2
```

In this screen you can define the contents of the “actions” menu. A menu entry is generated for every key for which a label is defined in this screen. Pressing the key can be simulated by clicking on the corresponding menu entry.

13.3.12 Specifications for attribute blocks in the data transfer area

```
I F G                  VIEWING YOUR USER PROFILE USERPRO                    07OC
                    DEFAULT VALUES FOR FORMAT PREPARATION
       SPECIFICATIONS FOR FIELD ATTRIBUTE BLOCKS IN THE DATA TRANSFER AREA




            A field attribute block comprises the following attribute groups:

            Basic attributes: YES          Field length         : NO
            Type of input   : NO           Attribute combination: NO
            Representation  : NO           Color                : NO
            Cursor          : NO           EDIT return value    : NO




To check any changes: SEND                                      For help: K3
To store your input:  F3                                 To abort function: K2
```

Here you define which attribute groups are to be assigned to a field attribute block. These specifications are only valid if you are using the data transfer area with separate attribute blocks and field contents for your format. The meaning of the attribute groups is described in the FHS manual “Format Handling System for openUTM, TIAM, DCAM”. The attribute group “basic attributes” must always be present. If illegal combinations of attribute groups are defined, IFG issues an error message. The attribute group “attribute combination” cannot by combined with any of the attribute groups “type of input”, “representation” or “cursor”.

Note

Remember when selecting the attribute groups that in order to ensure optimum memory space utilization and runtime in FHS by the application program, only attribute groups actually needed should be included in the addressing aid.

13.3.13 Specifications for global attribute blocks in the data transfer area

```
I F G                    VIEWING YOUR USER PROFILE USERPRO                  07OE
                       DEFAULT VALUES FOR FORMAT PREPARATION
        SPECIFICATIONS FOR GLOBAL ATTRIBUTE BLOCK IN THE DATA TRANSFER AREA




            The global attribute block comprises the following parts:

            Basic Format Attributes: YES
            Message Identifier     : NO
            dialog cursor position : NO









To check any changes: SEND                                      For help: K3
To store your input:  F3                                 To abort function: K2
```

Here you define which attributes are to be assigned to a global attribute block. These specifications are only valid if you are using the data transfer area with separate attribute blocks and field contents for your format. The meaning of the attribute groups is given in the FHS manual “Format Handling System for openUTM, TIAM, DCAM”. The attribute group “basic format attributes” must always be present. The attribute groups “message identifier” and “cursor position” are optional.

13.4 Create IFG default user profile

The creation of user profiles is described in the chapter “Creating a format” on page 51.

13.5 Copy user profile

If you want to copy an existing user profile, you must specify:

– the name of the format library from which the user profile is to be copied

– the name and, if applicable, password of the user profile to be copied

– the name of the new user profile, possibly with password

Mark “copy” and press SEND.

After completion of the copy operation, the new user profile becomes the current user profile.

You may either copy a user profile from another format file/library or copy a user profile from the current format library and give it a new name. In both cases, the new user profile cannot be given a name that already exists, unless you want to overwrite the default user profile USERPRO.

13.6 Delete user profile

In order to delete a user profile, enter its name, mark “delete” and press the F1 key. IFG then requests you to specify a current user profile.

14 Converting existing formats

Formats created with IFG ≤ V5.0 are stored in format file (ISAM files). Since only PLAM libraries will be used in IFG V6.0 or later, formats created with IFG ≤ V5.0 must be converted. This can be done with function 08 “Conversion of format file of former version”. Formats created with more recent IFG versions need not be converted.

After selecting function 08, you only need to enter the name of the old format file and the name of the converted user profile and press the SEND key.

```
I F G              CONVERSION OF FORMAT FILE OF FORMER VERSION              0801




Name of old file:

Name of new library:  IFG.FORMATS
Name of user profile:   USERPRO









To execute the function:     SEND
To select another function:  F3                                  For help:  K3
```

If the conversion is executed successfully, you will receive the message:

```
6240: YOUR FORMAT FILE HAS BEEN CONVERTED AS REQUESTED
```

If the old format file and the new format library contain formats with the same names, these formats will not be converted.

15 Managing formats with IFGFV

Unlike IFG, the utility routine **IFGFV** can support multiple formats:

In addition, you can use IFGFV for the following tasks (like IFG):

– creating the IFG default user profile

– copying an existing user profile or all user profiles

– selecting a user profile

– deleting one or all user profiles

– converting ISAM format files into PLAM libraries

– printing out one or more formats, key lists, and messages

– printing out the directory of a format library

– deleting one or more formats, key lists, and messages

– copying individual, or a group of formats, key lists, and messages from the current format library

– copying one or more formats, key lists, and messages from another format file or format library

– generating addressing aids for one or more formats in a format library

– generating object formats or format modules for one or more formats, key lists, and messages in a format library

In addition, you can use IFGFV

– in batch mode (for example in an ENTER task) or

– in interactive mode from any terminal. The terminal does **not** have to be supported by IFG, nor must it be possible to display the formats on this terminal.

The syntax of the IFGFV statements is stored in an SDF syntax file. Hence, IFGFV can only run if the SDF (System Dialog Facility) component is installed in BS2000 (see also the “SDF-A (BS2000/OSD)” manual).

The following components must be available in order to run IFGV:

– the IFG format library,

– the FHS library MFHSROUT,

– the IFG module library, and

– the FHS-DOORS converter if required

IFG is called in BS2000 timesharing mode (communication application $DIALOG) with the command /START-IFG or /IFG. Please note that IFG is not executable under OMNIS if /OPTION BREAK-KEY=1.

IFG format library

The IFG format library contains the IFGV messages and screens.

The default name of the IFGV format library is SYSFHS.IFG.083.D for the German format library and SYSFHS.IFG.083.E for the English format library.

If this default name is cataloged under your own user ID or the default user ID $, this file can be automatically accessed by IFGV.
When the default name is used, the language for messages and screens is defined by the BS2000 generation (MSGLPRI=ED/DE class 2 op).
The command /MODIFY-MSG-ATTRIBUTES TASK-LANGUAGE=D or E or *STD can be used for a subsequent assignment or reassignment.

The format library can, however, also be stored under any desired name under any user ID, in which case it must be assigned with the command
/SET-FILE-LINK LINK-NAME=IFGMAPS,FILE-NAME=filename.

FHS library MFHSROUT

The library MFHSROUT is used by IFGV for fast formatting and for the print function. If this library is not cataloged under its default name $.MFHSROUT, it must be assigned with the command /SET-FILE-LINK LINK-NAME=MROUTLIB,FILE-NAME=file-name before starting IFGV.

IFG module library

The default name of the IFGV module library is $.SYSLNK.IFG.083. If this name is used, IFGV can be started with the command /START-IFGV. If you have assigned a different name to the library, you must call IFGV as follows:
/START-PROGRAM FROM-FILE=*MODULE(file-name,IFGV,RUN-MODE=*ADVANCED)

FHS-DOORS

If formats are also to be converted by FHS-DOORS, the module library FHS-DOORS-C (default name: $.SYSLNK.FHS-DOORS-LC.010) must always be assigned before starting IFG. The appropriate command for this is:
/SET-FILE-LINK LINK NAME=DOORSLIB,FILE-NAME=libname.

For compatibility reasons, IFGV V8.3 can also be started with the command that was normally used for earlier versions, i.e. /START-PROGRAM FROM-FILE=$.IFGV (or $IFGV). As a prerequisite, IFGV and the IFGV module library SYSLNK.IFG.083 must be installed under your own user ID or the default user ID $. However, you should use the start command /START-IFGV as far as possible.

Examples

```
/BEGIN-PROCEDURE
/ASSIGN-SYSLST TO-FILE=#IFG83
/SET-FILE-LINK LINK-NAME=DOORSLIB,FILE-NAME=$.SYSLNK.FHS-DOORS.LC.010
/MODIFY-JOB-SWITCHES ON=11
/START-IFGFV
/MODIFY-JOB-SWITCHES OFF=11
/ASSIGN-SYSLST *PRIMARY
/END-PRCEDURE
```

```
/BEGIN-PROCEDURE
/ASSIGN-SYSLST TO-FILE=#IFG81D
/SET-FILE-LINK LINK-NAME=IFGMAPS,FILE-NAME=$.SYSFHS.IFG.083.D
/SET-FILE-LINK LINK-NAME=MROUTLIB,FILE-NAME=$.MFHSROUT
/SET-FILE-LINK LINK-NAME=DOORSLIB,FILE-NAME=$.SYSLNK.FHS-DOORS.LC.010
/MODIFY-JOB-SWITCHES ON=11
/START-PROGR FROM-FILE=*MODUL($.SYSLNK.IFG.083,IFGFV,RUN-MODE=*ADVANCED)
/MODIFY-JOB-SWITCHES OFF=11
/ASSIGN-SYSLST *PRIMARY
/END-PRCEDURE
```

For notes on “Logging”, see section “Logging your dialog with IFG” on page 49.

Note that IFGFV procedures created with IFG V4.0 can still be used, provided that all ISAM format files used in this procedure have been converted beforehand into PLAM libraries. This can be done with the IFGFV statements CONVERT-FILE or TRANSFER-FORMAT or with IFG function 08 “Conversion of format file of former version”.

15.1 IFGFV commands

15.1.1 Assigning the format file

ASSIGN-WORK-FORMAT-FILE or **MAPFILE**
NAME = <full-filename 1..54>

This command must always be the first command in a program, except for the command CONVERT-FILE. It specifies the current format library.

NAME = <full-filename 1..54>
Name of the current format library which is to be assigned.

ASSIGN-INPUT-FORMAT-FILE or **COPYFILE**
NAME = <full-filename 1..54>

NAME = <full-filename 1..54>
Name of the format file or format library from which formats are to be copied.

15.1.2 Converting format files from former versions

CONVERT-FILE
FROM-FILE = ***STD** / <full-filename 1..54>
,**TO-FILE** = ***STD** / <full-filename 1..54>
,**USER-PROFILE-NAME** = ***STD** / <alphanum-name 1..8>

This command is used to convert old format files into format libraries. Formats created with IFG versions < V6.0 can be used with IFG ≥ V6.0 only if they are converted first, since ISAM files are no longer permitted for IFG V6.0 or later. After conversion, the first two arguments are the file to be copied and the destination file (this corresponds to the two commands MAPFILE and COPYFILE).

The user profile specified here becomes the current user profile if no user profile was specified previously.

This is the only command which may be executed before the command ASSIGN-WORK-FORMAT-FILE.

FROM-FILE = *STD / <full-filename 1..54>
The name of the ISAM format file to be converted.

***STD**
The file to be converted has already been assigned with a previously executed COPYFILE command (default).

TO-FILE = *STD / <full-filename 1..54>
The name of the converted format library (PLAM)

***STD**
The converted format library has already been assigned with a previously executed MAPFILE command (default).

USER-PROFILE-NAME = *STD / <alphanum-name 1..8>
The name to be assigned to the copy of the user profile.

***STD**
The user profile receives the default name USERPRO.

15.1.3 Creating the IFG standard user profile

CREATE-USER-PROFILE
USER-PROFILE-NAME = ***STD** / <alphanum-name 1..8> /

The IFG standard user profile is assigned to the current format library, which was assigned with ASSIGN-WORK-FORMATFILE. This command does not overwrite any existing user profile.

USER-PROFILE-NAME = *STD / <alphanum-name 1..8>
The name to be assigned to the newly created user profile.

***STD**
The user profile receives the default name USERPRO.

15.1.4 Copying the user profile

TRANSFER-USER-PROFILE
USER-PROFILE-NAME = ***ALL** / <alphanum-name 1..8>

This command copies the user profile from the format library which was assigned with ASSIGN-INPUT-FORMATFILE into the current format library (ASSIGN-WORK-FORMATFILE). If the input format file is an ISAM file, the copy of the user profile always receives the default name USERPRO. If a user profile in the current format library has the same name as the user profile to be copied, the user profile is not copied (unless the user profile USERPRO is being copied - this user profile is always overwritten by the copied user profile with the same name).

USER-PROFILE-NAME = *ALL / <alphanum-name 1..8>
The name of the user profile to be copied.

***ALL**
All user profiles are to be copied (default).

COPY-USER-PROFILE
FROM-USER-PROFILE = <alphanum-name 1..8>
,**TO-USER-PROFILE** = <alphanum-name 1..8>

This command copies a user profile within the currently assigned format library. The command will not overwrite an existing user profile (except for USERPRO, which is always overwritten).

FROM-USER-PROFILE = <alphanum-name 1..8>
The name of the user profile to be copied.

TO-USER-PROFILE = <alphanum-name 1..8>
The name of the copied user profile.

15.1.5 Selecting a user profile

SELECT-USER-PROFILE
USER-PROFILE-NAME = ***STD** / <alphanum-name 1..8>

This command is used to select the user profile which is to be used in the currently assigned format library. If this command is not issued before CREATE-ADDRESSING-AID or CREATE-FORMAT-MODULE, the default user profile USERPRO is used for the addressing aids and for creation of the format modules, provided it exists within the current format library. If it does not exist, the command is ignored.

USER-PROFILE-NAME = *STD / <alphanum-name 1..8>
The name of the user profile to be selected.

***STD**
The default user profile USERPRO is to be selected (default).

15.1.6 Deleting a user profile

DELETE-USER-PROFILE
USER-PROFILE-NAME = ***STD** / ***ALL** / <alphanum-name 1..8>

This command deletes user profiles from the currently assigned format library. The current user profile cannot be deleted with this command.

USER-PROFILE-NAME = *STD / *ALL / <alphanum-name 1..8>
The name of the user profile to be deleted.

***STD**
The default user profile USERPRO is to be deleted (default).

***ALL**
All user profiles are to be deleted.

15.1.7 Copying formats

COPY-FORMAT
FROM-FORMAT = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>) **,TO-FORMAT** = ***SAME** / <alphanum-name 1..8> (**VERSION** = ***SAME** / <text 0..24>)

Internal copying:

This command copies a format within the current assigned format library. The command will not overwrite an existing format with the same name; in this case, the command is ignored.

FROM-FORMAT = *ALL / <alphanum-name 1..8>
The name of the format to be copied.

***ALL**
All formats with the specified version are to be copied.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
The version of the format to be copied.

***HIGHEST-EXISTING**
The highest existing version is to be copied (default).

***ALL**
All versions of the format are to be copied.

TO-FORMAT = *SAME / <alphanum-name 1..8>
The name of the copied format.

***SAME**
The copied format is to have the same name as the original (default).

VERSION = *SAME / <text 0..24>
Version of the copied format.

***SAME**
The copied format is to have the same version as the original (default).

The default value *SAME may not be specified simultaneously for both operands TO-FORMAT and VERSION, since this would cause the format to be copied onto itself.

TRANSFER-FORMAT
FORMAT-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>)

External copying:

This command copies formats from another format file or format library, whose name must be specified beforehand. The copied format always receives the same name and version as the original. An existing format cannot be overwritten with this command.

FORMAT-NAME = *ALL / <alphanum-name 1..8>
The name of the format to be copied.

***ALL**
All formats with the specified version are to be copied.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
The version of the format to be copied.

***HIGHEST-EXISTING**
The highest existing version is to be copied (default).

***ALL**
All versions of the format are to be copied.

COPY
FROM-FORMAT = <alphanum-name 1..8> ,**TO-FORMAT** = ***SAME** / <alphanum-name 1..8>

The Copy command continues to be supported only for compatibility reasons (IFGFV < V6).

This statement can be used to copy formats within a format library (internal copy) and also from some other format file or format library (external copy). Before an external copy, the source format file or library must be first assigned (ASSIGN-INPUT-FORMAT-FILE). An existing format cannot be overwritten with this statement. The Copy statement is ignored in such cases.

FROM-FORMAT = <alphanum-name 1..8>
Name of the format to be copied. The format file or library must be assigned with ASSIGN-INPUT-FORMAT-FILE before an external copy operation.

TO-FORMAT = *SAME / <alphanum-name 1..8>
Name of the copied format

***SAME**
The format copy is assigned the name of the original format (default).

Note

If you do not specify the operand TO-FORMAT, an external copy is performed. If you specify a name, the format is copied within the current format library.

The highest version of a format is always copied with this statement; the copied format receives the same version.

If different versions of formats are to be copied, the statements COPY-FORMAT (internal copy) or TRANSFER-FORMAT (external copy) can be used.

15.1.8 Creating addressing aids

CREATE-ADDRESSING-AID
FORMAT-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / <text 0..24> **,ADDRESSING-AID-NAME** = ***SAME** / <alphanum-name 1..8> **,ADDRESSING-AID-LIB** = ***BY-USER-PROFILE** / <full-filename 1..54>

This command creates addressing aids.

FORMAT-NAME = *ALL / <alphanum-name 1..8>
Name of the format for which addressing aids are to be created.

***ALL**
Addressing aids are to be created for all formats of the assigned format library.

VERSION = *HIGHEST-EXISTING / <text 0..24>
The version of the format for which addressing aids are to be created.

***HIGHEST-EXISTING**
The highest existing version is to be used (default).

ADDRESSING-AID-NAME = *SAME / <alphanum-name 1..8>
Name of the addressing aid.

***SAME**
The addressing aid is to have the same name as the format (default).

ADDRESSING-AID-LIB = *BY-USER-PROFILE / <full-filename 1..54>
The name of the source program library in which the addressing aids are to be stored.

***BY-USER-PROFILE**
The library name is to be obtained from the user profile.

The addressing aid always receives the same version as the source format. The format library must contain at least one user profile when this command is executed. If this user profile is not USERPRO, it must also be selected beforehand; otherwise, the command is ignored.

15.1.9 Creating object modules

CREATE-FORMAT-MODULE
FORMAT-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>) **,FORMAT-MODULE-LIB** = ***BY-USER-PROFILE** / <full-filename 1..54>

This command creates object modules.

FORMAT-NAME = *ALL / <alphanum-name 1..8>
The name of the format for which an object module is to be created.

***ALL**
Object modules are to be created for all formats in the specified format library.

VERSION = *HIGHEST-EXISTING / <text 0..24>
The version of the format for which a object module is to be created.

***HIGHEST-EXISTING**
An object module is to be created for the highest existing version of the specified format (default).

FORMAT-MODULE-LIB = *BY-USER-PROFILE / <full-filename 1..54>
The name of the format application file which is to contain the object modules.

***BY-USER-PROFILE**
The module library name is to be obtained from the user profile.

The object module (type R) and the optional supplementary FHS-DOORS converter SDC file (type S) always receive the same name and version as the source format. The format library must contain at least one user profile when this command is executed. If this user profile is not USERPRO, it must also be selected beforehand; otherwise, the command is ignored.

15.1.10 Deleting formats

DELETE-FORMAT
FORMAT-NAME = ***ALL** / <alphanum-name 1..8> **VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>

This command deletes formats.

FORMAT-NAME = *ALL / <alphanum-name 1..8>
The name of the format which is to be deleted from the current format library.

***ALL**
All formats are to be deleted from the current format library.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
Version of the format to be deleted from the current format library.

***HIGHEST-EXISTING**
The highest existing version of the specified format is to be deleted (default).

***ALL**
All versions of the specified format are to be deleted.

15.1.11 Printing formats

PRINT-FORMAT
FORMAT-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>) **INFORMATION** = ***ALL-ATTRIBUTES** or **LONG** / ***FORMAT-GLOBALS** or **SHORT** / ***VARIABLES** or **VARS**

This command prints out formats to SYSLST. If no user profile is selected before this command is issued, the program assumes that USERPRO is the current user profile.

FORMAT-NAME = *ALL / <alphanum-name 1..8>
The name of the format to be printed.

***ALL**
All formats in the current format library are to be printed.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
The version of the format to be printed.

***HIGHEST-EXISTING**
The highest existing version of the specified format is to be printed (default).

***ALL**
All versions of the specified format are to be printed.

INFORMATION = *ALL-ATTRIBUTES or **LONG / *FORMAT-GLOBALS** or **SHORT / *VARIABLES** or **VARS**
specifies the type of format printout.

ALL-ATTRIBUTES or **LONG**
Full format printout:
The printout comprises the list of general format attributes, the image of the format and the list of all field names and field attributes (default).

FORMAT-GLOBALS or **SHORT**
Abbreviated format printout:
The printout comprises the list of general format attributes and the image of the format.

VARIABLES or **VARS**
Printout of the format variables. The printout consists of the list of variables defined in the format.

15.1.12 Copying key lists

COPY-KEY-LIST
FROM-KEY-LIST = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>) ,**TO-KEY-LIST** = ***SAME** / <alphanum-name 1..8> (**VERSION** = ***SAME** / <text 0..24>)

This command copies key lists within the currently assigned format library. An existing key list is not overwritten using this command; the command is simply ignored.

FROM-KEY-LIST = *ALL / <alphanum-name 1..8>
The name of the key list to be copied.

***ALL**
All key lists with the specified version are copied.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
The version of the key list to be copied.

***HIGHEST-EXISTING**
The highest possible version of the key list is copied (default).

***ALL**
All versions of the key list are copied.

TO-FORMAT = *SAME / <alphanum-name 1..8>
The name of the copied key list.

***SAME**
The copied key list receives the name of the original key list (default).

VERSION = *SAME / <text 0..24>
The version of the copied key list.

***SAME**
The copied key list receives the version of the original format (default).

The default value *SAME cannot be used simultaneously for the operands TO-KEY-LIST and VERSION, as this copy the key list onto itself.

TRANSFER-**K**EY-**L**IST
KEY-LIST-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>)

This command copies key lists from another format file or library previously specified. The new key list always receives the name and version of the original key list. It is not possible to overwrite an existing key list using this command.

KEY-LIST-NAME = *ALL / <alphanum-name 1..8>
The name of the key list to be copied.

***ALL**
All key lists are copied.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
The version of the key list to be copied.

***HIGHEST-EXISTING**
The highest possible version of the key list is copied (default).

***ALL**
All versions of the key list are copied.

15.1.13 Creating a key list module

CREATE-**K**EY-**L**IST-**M**ODULE
KEY-LIST-NAME = ***ALL**(...) / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / <text 0..24> **,FORMAT-MODULE-LIB** = ***BY-USER-PROFILE** / <full-filename 1..54>

This command creates object modules for the key list.

KEY-LIST-NAME = *ALL / <alphanum-name 1..8>
The name of the key list, for which an object module is to be created.

***ALL**
Object modules are to be created for all key lists.

VERSION = *HIGHEST-EXISTING / <text 0..24>
The version of the key list member, for which an object module is to be created.

***HIGHEST-EXISTING**
An object module is to be created for the highest possible version of the specified key list (default).

FORMAT-MODULE-LIB = *BY-USER-PROFILE / <full-filename 1..54>
The name of the application file for the key list that is to contain the object modules.

***BY-USER-PROFILE**
The name of the module library is to be taken from the user profile (default).

The object module always receives the same version as the source key list. The format library must contain at least one user profile before this command is issued. If the user profile is not USERPRO, it must be specified beforehand, otherwise the command is ignored.

15.1.14 Deleting key lists

DELETE-**K**EY-**L**IST
KEY-LIST-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>)

This command deletes key lists.

KEY-LIST-NAME = *ALL / <alphanum-name 1..8>
The name of the key list to be deleted from the current format library.

***ALL**
All key lists are to be deleted from the current format library.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
Version of the key list to be deleted from the current format library.

***HIGHEST-EXISTING**
The highest possible version of the specified key list is to be deleted (default).

***ALL**
All versions of the specified key list in the current format library are to be deleted.

15.1.15 Printing key lists

PRINT-**K**EY-**L**IST
KEY-LIST-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>)

This command prints key lists to SYSLST. If no user profile is specified before this command is issued, it is assumed that USERPRO is the current user profile.

FORMAT-NAME = *ALL / <alphanum-name 1..8>
The name of the key list to be printed.

***ALL**
All key lists in the current format library are to be printed.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
Version of the key list to be printed.

***HIGHEST-EXISTING**
The highest possible version of the specified key list is to be printed (default).

***ALL**
All versions of the specified key list are to be printed.

15.1.16 Copying message members

COPY-MSG-**M**EMBER
FROM-MSG-MEMBER = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>) ,**TO-MSG-MEMBER** = ***SAME** / <alphanum-name 1..8> **VERSION** = ***SAME** / <text 0..24>

This command copies message members within the currently assigned format library. It is not possible to overwrite an existing message member using this command; the command is simply ignored.

FROM-MSG-MEMBER = *ALL / <alphanum-name 1..8>
The name of the message member to be copied.

***ALL**
All message members with the specified version are copied.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
The version of the message member to be copied.

***HIGHEST-EXISTING**
The highest possible version of the message member is copied (default).

***ALL**
All versions of the message member are copied.

TO-MSG-MEMBER = *SAME / <alphanum-name 1..8>
The name of the copied message member.

***SAME**
The copied message member receives the name of the original format (default).

VERSION = *SAME / <text 0..24>
The version of the copied message member.

***SAME**
The copied message member receives the version of the original message member (default).

The default value *SAME cannot be used simultaneously for the operands TO-MSG-MEMBER and VERSION, as this would copy the message members onto themselves.

TRANSFER-**M**SG-**M**EMBER
MSG-MEMBER-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>)

This command copies message members from another format file or library previously specified. The new message member always receives the name and version of the original message member. It is not possible to overwrite an existing message using this command.

MSG-MEMBER-NAME = *ALL / <alphanum-name 1..8>
Name of the message member to be copied (four letters, three zeros, and the optional language identifier).

***ALL**
All message members are copied.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
The version of the message member to be copied.

***HIGHEST-EXISTING**
The highest possible version of the message member is copied (default).

***ALL**
All versions of the message member are copied.

15.1.17 Creating a message member module

CREATE-**M**SG-**M**EMBER-**MOD**ULE
MSG-MEMBER-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / <text 0..24>) ,**FORMAT-MODULE-LIB** = ***BY-USER-PROFILE** / <full-filename 1..54>

This command creates object modules for message members.

MSG-MEMBER-NAME = *ALL / <alphanum-name 1..8>
The name of the message member, for which an object module is to be created.

***ALL**
Object modules are to be created for all message members.

VERSION = *HIGHEST-EXISTING / <text 0..24>
Version of the message member for which an object module is to be created.

***HIGHEST-EXISTING**
An object module is created for the highest possible version of the specified message member (default).

FORMAT-MODULE-LIB = *BY-USER-PROFILE / <full-filename 1..54>
The name of the format application file which is to contain the object modules.

***BY-USER-PROFILE**
The name of the module library is to be taken from the user profile (default).

The object module always contains the same version as the source message member. The format library must contain at least one user profile before this command is issued. If the user profile is not USERPRO, it must be selected beforehand, otherwise the command is ignored.

15.1.18 Deleting message members

DELETE-**M**SG-**M**EMBER
MSG-MEMBER-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION =** ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>)

This command deletes message members.

MSG-MEMBER-NAME = *ALL / <alphanum-name 1..8>
Name of the message member to be deleted from the current format library (four letters, three zeros, and the optional language identifier).

***ALL**
All message members from the current format library are to be deleted.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
Version of the message member to be deleted from the current format library.

***HIGHEST-EXISTING**
The highest possible version of the specified message member is to be deleted (default).

***ALL**
All versions of the specified message member from the format library are to be deleted.

15.1.19 Printing message members

PRINT-**M**SG-**ME**MBER
MSG-MEMBER-NAME = ***ALL** / <alphanum-name 1..8> (**VERSION** = ***HIGHEST-EXISTING** / ***ALL** / <text 0..24>)

This command prints message members to SYSLST. If no user profile is specified before this command is issued, it is assumed that USERPRO is the current user profile.

MSG-MEMBER-NAME = *ALL / <alphanum-name 1..8>
The name of the message member to be printed (four letters, three zeros, and the optional language identifier).

***ALL**
All message members in the current format library are to be printed.

VERSION = *HIGHEST-EXISTING / *ALL / <text 0..24>
The version of the message member to be printed.

***HIGHEST-EXISTING**
The highest possible version of the specified message member is to be printed (default).

***ALL**
All versions of the specified message member are to be printed.

15.1.20 Printing directories

PRINT-**DIR**ECTORY

This command prints out the directory of the current format library to the system file SYSLST. Formats come first, followed by key lists, then messages.

15.1.21 Show list of previous inputs

SHOW-INPUT-**HIST**ORY
ENTRIES = **8** / <integer 1..100> / ***ALL** ,**SELECT** = ***STMT** / ***CMD** / ***ALL** ,**PATTERN** = ***NONE** / <structured-name 1..30> ,**INP**UT-**SERIAL-NUMB**ER = ***NO** / ***YES**

This command outputs a list (history) of previous inputs.

15.1.22 Output of SDF options

SHOW-SDF-OPTIONS
INFORMATION = ***ALL** / ***USER**

This command outputs the names of the SDF syntax files which are currently active and the settings of the SDF options. This is done for formats, key lists, and messages in the given order.

15.1.23 Modifying SDF options

MODIFY-**SDF-OPT**IONS
SYNTAX-FILE = ***UNCH**ANGED / ***NONE** / <full-filename 1..54> / ***STD** ,**GUID**ANCE = ***UNCH**ANGED / ***EXP**ERT / ***NO** / ***MAX**IMUM / ***MED**IUM / ***MIN**IMUM ,**LOG**GING = ***UNCH**ANGED / ***INP**UT-**FORM** / ***ACCEP**TED-**FORM** / ***INVAR**IANT-**FORM** ,**UTIL**ITY-**INTERF**ACE = ***UNCH**ANGED / ***OLD-MODE** / ***NEW-MODE** ,**PROC**EDURE-**DIALOG**UE = ***UNCH**ANGED / ***YES** / ***NO** ,**CONT**INUATION = ***UNCH**ANGED / ***OLD-MODE** / ***NEW-MODE** ,**MENU-LOG**GING = ***UNCH**ANGED / ***YES** / ***NO** ,**FUNC**TION-**KEYS** = ***UNCH**ANGED / ***STYL**E-**GUI**DE-**MODE** / ***BY-TERM**INAL-**TYPE** / ***OLD-MODE** ,**INP**UT-**HIST**ORY = ***UNCH**ANGED / ***ON(...)** / ***OFF** / ***RESET** *ON(...) | **NUMB**ER-**OF-INP**UTS = ***UNCH**ANGED / <integer 1..100>

The SDF options can be modified with this command (see the “SDF-A (BS2000/OSD)” manual).

15.1.24 Restore SDF input

RESTORE-**SDF-INP**UT
INPUT = ***LAST** / <integer -100..-1> / <integer 1..9999>

The operands and operand values of the last command entered are output.

15.1.25 Remarks

REMARK
TEXT = <cmd-rest 0..1800>

This command writes remarks to a procedure file.

15.1.26 Writing text

WRITE-TEXT
TEXT = ´␣´ / <c-string 1..1024 with-low>

This command writes text to SYSOUT.

15.1.27 Defining a restart point

STEP

This command is used to define a restart point for error recovery. It should be used in a batch run in conjunction with the following commands:

```
ASSIGN-WORK-FORMAT-FILE,
ASSIGN-INPUT-FORMAT-FILE,
CREATE-USER-PROFILE,
COPY,
COPY-FORMAT,
TRANSFER-FORMAT,
TRANSFER-USER-PROFILE,
COPY-USER-PROFILE,
SELECT-USER-PROFILE,
```

If an error occurs while one of the above commands is being analyzed, the subsequent commands are skipped until //STEP or //END is reached. The command following this is then read and processed.

15.1.28 Terminating IFGFV

END

This command terminates the program.

15.2 Examples

15.2.1 Examples of IFGFV commands

– Assigning the current format library:

```
ASSIGN-WORK-FORMAT-FILE FORMAT.MAPS
```

– Setting up the default user profile USERPRO

```
CREATE-USER-PROFILE
```

– External copying:

```
ASSIGN-INPUT-FORMAT-FILE COPY.MAPS
```

To assign the format file or format library from which formats are to be copied.

```
TRANSFER-FORMAT EXAMPLE(VERSION1)
```

The name and version of the format which is to be copied.

```
TRANSFER-FORMAT *ALL
```

The highest existing version of each format in the file/library is to be copied.

```
TRANSFER-FORMAT *ALL(*ALL)
```

All formats are to be copied.

– Internal copying:

```
COPY-FORMAT EXAMPLE1(VERSION1),EXAMPLE2(VERSION2)
```

```
COPY-FORMAT EXAMPLE1,EXAMPLE2
```

The highest existing version of the format is to be copied.

– Copying a user profile

```
TRANSFER-USER-PROFILE PROFIL1
```

– Creating the addressing aids for a format (highest possible version). The name of the addressing aid is the same as the format name; the name of the source program library is to be obtained from the user profile (the user profile is USERPRO if no other user profile has been selected beforehand).

```
CREATE-ADDRESSING-AID EXAMPLE,*SAME,*BY-USER-PROFILE
```

– Selecting the current user profile

```
SELECT-USER-PROFILE PROFILE1
```

– Creating the object modules for a format or for the highest possible version of each format in the current format library; the name of the format application file is to be obtained from the user profile.

```
CREATE-FORMAT-MODULE EXAMPLE(VERSION1),*BY-USER-PROFILE
CREATE-FORMAT-MODULE *ALL,*BY-USER-PROFILE
```

– Deleting a format, the highest possible version of each format, all formats or all versions of a format from the current format library:

```
DELETE-FORMAT EXAMPLE(VERSION1)
DELETE-FORMAT *ALL
DELETE-FORMAT *ALL(*ALL)
DELETE-FORMAT EXAMPLE(*ALL)
```

– Printing out the directory of the current format library:

```
PRINT-DIRECTORY
```

– Printing out a format in abbreviated or full form:

```
PRINT-FORMAT EXAMPLE,FORMAT-GLOBALS
PRINT-FORMAT EXAMPLE,ALL-ATTRIBUTES
```

– Printing out the highest possible version of each all format in the current format library file in the full form:

```
PRINT-FORMAT *ALL,ALL-ATTRIBUTES
```

– Terminating the program:

```
END
```

The statements can be abbreviated in accordance with the rules for SDF (see the “SDF-A (BS2000/OSD)” manual). The abbreviations must, however, remain unambiguous. It is not possible to guarantee that the abbreviations will remain valid in future versions of the program.

15.2.2 Example of how to convert a format created with IFG V5.0

```
/PROC A,(&NEWFILE,&OLDFILE,&PTIME=99)
/REMARK ** PLEASE ENTER NAME OF NEW FORMAT LIBRARY : &NEWFILE
/REMARK ** PLEASE ENTER NAME OF OLD FORMAT FILE : &OLDFILE
/SYSFILE SYSDTA=(SYSCMD)
/EXEC IFGFV
ASSIGN-WORK-FORMAT-FILE  &NEWFILE
ASSIGN-INPUT-FORMAT-FILE &OLDFILE    or
TRANSFER-USER-PROFILE                CONVERT-FILE &OLDFILE,&NEWFILE
TRANSFER-FORMAT *ALL
END
/SYSFILE SYSDTA=(PRIMARY)
/ENDP
```

Result of logging

```
(IN)     DO MIGR
(IN)     /PROC A,(&NEWFILE,&OLDFILE)
(IN)     /REMARK ** PLEASE ENTER NAME OF NEW FORMAT LIBRARY : &NEWFILE
(OUT)    &NEWFILE=
(IN)     FORMLIB.NEW
(IN)     /REMARK ** PLEASE ENTER NAME OF NEW FORMAT LIBRARY : FORMLIB.NEW
(IN)     /REMARK ** PLEASE ENTER NAME OF OLD FORMAT FILE : &OLDFILE
(OUT)    &OLDFILE=
(IN)     FORMLIB
(IN)     /REMARK ** PLEASE ENTER NAME OF OLD FORMAT FILE : FORMLIB
(IN)     /SYSFILE SYSDTA=(SYSCMD)
(IN)     /EXEC IFGFV
(OUT)    %  BLS0524 LLM 'IFGFV', VERSION 'V08.3A10' OF '2006-03-27 14:21:57'
         LOADED
         %  BLS0551 COPYRIGHT (C) FUJITSU SIEMENS COMPUTERS GMBH 2006.
         ALL RIGHTS RESERVED
(IN)     ASSIGN-WORK-FORMAT-FILE  FORMLIB.NEW
(IN)     ASSIGN-INPUT-FORMAT-FILE FORMLIB
(IN)     TRANSFER-USER-PROFILE
(IN)     TRANSFER-FORMAT *ALL
(IN)     END
(IN)     /SYSFILE SYSDTA=(PRIMARY)
(IN)     /ENDP
```

Example in SDF syntax

```
/BEGIN-PROCEDURE LOGGING=A,PAR=YES(PROC-PAR=(&NEWFILE,&OLDFILE),ESC-
CHAR=C'&')
/REMARK ** PLEASE ENTER NAME OF NEW FORMAT LIBRARY : &NEWFILE
/REMARK ** PLEASE ENTER NAME OF OLD FORMAT FILE : &OLDFILE
/ASSIGN-SYSDTA TO-FILE =*SYSCMD
/START-PROGRAM IFGFV
ASSIGN-WORK-FORMAT-FILE  &NEWFILE
ASSIGN-INPUT-FORMAT-FILE &OLDFILE     or
TRANSFER-USER-PROFILE                    CONVERT-FILE &OLDFILE,&NEWFILE
TRANSFER-FORMAT *ALL
END
/ASSIGN-SYSDTA TO-FILE=*PRIMARY
/END-PROCEDURE
```

15.2.3 Example of processing a library to obtain only English language elements without suffix E

By default, the library SYSFHS.FHS.083.FHS-DE contains German and English language elements that only differ in terms of the suffix D (for German) or E (for English). The following example shows how the library can be processed so that only English language elements without suffix E are obtained.

A work file and output file are first created with the IFGFV command

```
//ASSIGN-WORK-FORMAT
```

and the input file is assigned with the command

```
//ASSIGN-INPUT-FORMAT
```

After the user file and the language-neutral elements are transferred, the required elements are copied to the work file with

```
//TRANSFER-FORMAT
//TRANSFER-KEY-LIST
//TRANSFER-MSG-MEMBER
```

The language elements are then copied within the work file with

```
//COPY-FORMAT
```

and given new element names without suffix E. The old element names are then deleted from the file with

```
//DELETE-FORMAT,
//DELETE-KEY-LIST
//DELETE-MSG-MEMBER.
```

Finally, the commands

```
//CREATE-FORMAT
//CREATE-KEY-LIST
//CREATE-MSG-MEMBER
```

are used to generate library members of type R.

Following verification with LMS, the new library generated can be copied to the default name SYSFHS.FHS.083.FHS-DE.

Example

```
/BEGIN-PROCEDURE
/DELETE-FILE FILE-NAME=SYSFHS.FHS.081.FHS-DE.NEW
/SET-JOB-STEP
/ASSIGN-SYSDTA *SYSCMD
/START-PROGRAM FROM-FILE=$.IFGFV
//ASSIGN-WORK-FORMAT SYSFHS.FHS.081.FHS-DE.NEW   “work and output file“
//ASSIGN-INPUT-FORMAT $.SYSFHS.FHS.081.FHS-DE    “input file“
//REMARK ##### Elements not to be given new names ####
//REMARK ##### The language element IDNSLNG is no longer required ####
//TRANSFER-USER-PROFILE USER-PROFILE-NAME=USERPRO “(used by CREA-Fs)“
//TRANSFER-FORMAT FORMAT-NAME=IDHSCHC “Marking select field eg / “
//TRANSFER-FORMAT FORMAT-NAME=IDHSCRL “Scroll prompts more“
//TRANSFER-FORMAT FORMAT-NAME=IDHSCHD “Locking exclusion char eg * “
//REMARK #### Copy elements to work and output file ####
//TRANSFER-FORMAT FORMAT-NAME=IDHFERRD
//TRANSFER-FORMAT FORMAT-NAME=IDHFER1D
//TRANSFER-FORMAT FORMAT-NAME=IDHFER2D
//TRANSFER-FORMAT FORMAT-NAME=IDHHBOXD
//TRANSFER-FORMAT FORMAT-NAME=IDHHCHCD
//TRANSFER-FORMAT FORMAT-NAME=IDHHCMDD
//TRANSFER-FORMAT FORMAT-NAME=IDHHCUAD
//TRANSFER-FORMAT FORMAT-NAME=IDHHCO1D
//TRANSFER-FORMAT FORMAT-NAME=IDHHCO2D
//TRANSFER-FORMAT FORMAT-NAME=IDHHCO3D
//TRANSFER-FORMAT FORMAT-NAME=IDHHCO4D
//TRANSFER-FORMAT FORMAT-NAME=IDHHCO5D
//TRANSFER-FORMAT FORMAT-NAME=IDHHCO6D
//TRANSFER-FORMAT FORMAT-NAME=IDHHCO7D
//TRANSFER-FORMAT FORMAT-NAME=IDHHCO8D
//TRANSFER-FORMAT FORMAT-NAME=IDHHCO9D
//TRANSFER-FORMAT FORMAT-NAME=IDHHC10D
//TRANSFER-FORMAT FORMAT-NAME=IDHHEDID
//TRANSFER-FORMAT FORMAT-NAME=IDHHFKTD
//TRANSFER-FORMAT FORMAT-NAME=IDHHFTPD
//TRANSFER-FORMAT FORMAT-NAME=IDHHHLPD
//TRANSFER-FORMAT FORMAT-NAME=IDHHIDXD
//TRANSFER-FORMAT FORMAT-NAME=IDHHMEND
//TRANSFER-FORMAT FORMAT-NAME=IDHHQTSD
//TRANSFER-FORMAT FORMAT-NAME=IDHHUSED

//REMARK #### Copy elements to work and output file ####
//TRANSFER-MSG-MEMBER MSG-MEMBER-NAME=IDHIOOOD
//TRANSFER-KEY-LIST KEY-LIST-NAME=IDHKEYAD
//TRANSFER-KEY-LIST KEY-LIST-NAME=IDHKEYED
//TRANSFER-KEY-LIST KEY-LIST-NAME=IDHKEYFD
//TRANSFER-KEY-LIST KEY-LIST-NAME=IDHKEYHD
```

```
//TRANSFER-KEY-LIST KEY-LIST-NAME=IDHKEYID
//TRANSFER-KEY-LIST KEY-LIST-NAME=IDHKEYKD
//TRANSFER-KEY-LIST KEY-LIST-NAME=IDHKEYMD
//TRANSFER-KEY-LIST KEY-LIST-NAME=IDHKEYND
//TRANSFER-KEY-LIST KEY-LIST-NAME=IDHKEYSD
//TRANSFER-KEY-LIST KEY-LIST-NAME=IDHKEYUD
//TRANSFER-FORMAT FORMAT-NAME=IDHKHLDP
//TRANSFER-MSG-MEMBER MSG-MEMBER-NAME=IDHS000D
//TRANSFER-FORMAT FORMAT-NAME=IDHS011D
//TRANSFER-FORMAT FORMAT-NAME=IDHS012D
//TRANSFER-FORMAT FORMAT-NAME=IDHS014D
//TRANSFER-FORMAT FORMAT-NAME=IDHS015D
//TRANSFER-FORMAT FORMAT-NAME=IDHS020D
//TRANSFER-FORMAT FORMAT-NAME=IDHS040D
//TRANSFER-FORMAT FORMAT-NAME=IDHS050D
//TRANSFER-FORMAT FORMAT-NAME=IDHS070D
//TRANSFER-FORMAT FORMAT-NAME=IDHS080D
//TRANSFER-FORMAT FORMAT-NAME=IDHS090D


//REMARK #### Copy to work and output file to create  ####
//REMARK #### new names without suffix D.  ####
//COPY-FORMAT FROM-FORMAT=IDHFERRD,TO-FORMAT=IDHFERR
//COPY-FORMAT FROM-FORMAT=IDHFER1D,TO-FORMAT=IDHFER1
.
.
.
//COPY-FORMAT FROM-FORMAT=IDHHUSED,TO-FORMAT=IDHHUSE
//COPY-MSG-MEMBER FROM-MSG-MEMBER=IDHI000D,TO-MSG-MEMBER=IDHI000
//COPY-KEY-LIST FROM-KEY-LIST=IDHKEYAD,TO-KEY-LIST=IDHKEYA
.
.
.
//COPY-KEY-LIST FROM-KEY-LIST=IDHKEYUD,TO-KEY-LIST=IDHKEYU
//COPY-FORMAT FROM-FORMAT=IDHKHLPD,TO-FORMAT=IDHKHLP
//COPY-MSG-MEMBER FROM-MSG-MEMBER=IDHS000D,TO-MSG-MEMBER=IDHS000
//COPY-FORMAT FROM-FORMAT=IDHS011D,TO-FORMAT=IDHS011
.
.
.
//COPY-FORMAT FROM-FORMAT=IDHS090D,TO-FORMAT=IDHS090

//REMARK #### Delete old element names with suffix D ####
//DELETE-FORMAT FORMAT-NAME=IDHFERRD
//DELETE-FORMAT FORMAT-NAME=IDHFER1D
.
.
.
//DELETE-FORMAT FORMAT-NAME=IDHHUSED
```

```
//DELETE-MSG-MEMBER MSG-MEMBER-NAME=IDHI000D
//DELETE-KEY-LIST KEY-LIST-NAME=IDHKEYAD
.
.
.
//DELETE-KEY-LIST FROM-KEY-LIST=IDHKEYUD
//DELETE-FORMAT FORMAT-NAME=IDHKHLPD
//DELETE-MSG-MEMBER MSG-MEMBER-NAME=IDHS000D
//DELETE-FORMAT FORMAT-NAME=IDHS011D
.
.
.
//DELETE-FORMAT FORMAT-NAME=IDHS090D

//REMARK #### Erzeugen von Elementen vom Typ R ####
//CR-FORM-MOD FORM-NAME=IDHFERR,FORM-MOD-LIB=SYSFHS.FHS.081.FHS-DE.NEW
//CR-FORM-MOD FORM-NAME=IDHFER1,FORM-MOD-LIB=SYSFHS.FHS.081.FHS-DE.NEW
.
.
.
//CR-FORM-MOD FORM-NAME=IDHHUSE,FORM-MOD-LIB=SYSFHS.FHS.081.FHS-DE.NEW
//CR-MSG-MEMB-MOD M-M-NAM=IDHI000,FORM-MOD-LIB=SYSFHS.FHS.081.FHS-DE.NEW
//CR-KEY-LIST-MOD K-L-NAM=IDHKEYA,FORM-MOD-LIB=SYSFHS.FHS.081.FHS-DE.NEW
.
.
.
//CR-KEY-LIST-MOD K-L-NAM=IDHKEYU,FORM-MOD-LIB=SYSFHS.FHS.081.FHS-DE.NEW
//CR-FORM-MOD FORM-NAME=IDHKHLP,FORM-MOD-LIB=SYSFHS.FHS.081.FHS-DE.NEW
//CR-MSG-MEMB-MOD M-M-NAM=IDHS000,FORM-MOD-LIB=SYSFHS.FHS.081.FHS-DE.NEW
//CR-FORM-MOD FORM-NAME=IDHS011,FORM-MOD-LIB=SYSFHS.FHS.081.FHS-DE.NEW
.
.
.
//CR-FORM-MOD FORM-NAME=IDHS090,FORM-MOD-LIB=SYSFHS.FHS.081.FHS-DE.NEW
//END

/ASSIGN SYSDTA *PRIMARY
/REMARK #### Check the new library generated against the old   ####
/REMARK #### library by using the LMS utility routine          ####
/REMARK #### (“tocu *“, “tocf *“, “tocr *“). If in order,      ####
/REMARK #### replace the old library by the new library        ####
/END-PROCEDURE
```

16 Arabic/Farsi formats

The following chapter describes the support for (8-bit) Arabic/Farsi formats available with IFG V8.1 or later. In order to use this functionality, XHCS must contain the corresponding Arabic/Farsi tables (as of XHCS V1.0B), and VTSU must support the special handling required for Arabic/Farsi scripts (available as of VTSU V11.0A).
The sections below describe the special features of Arabic/Farsi support as opposed to the default functionality. Detailed information on the individual functions listed can be found in the relevant sections in which the default support is described.

Note that the term “Arabic/Farsi” is used generically for all languages that are written from right to left (e.g. Arabic, Farsi, North African). This notation is handled differently by IFG as opposed to a left-to-right notation. In order to customize IFG to the required notation, the format must be assigned a global mode. Every format can be assigned a “ Latin” mode (left-to-right notation) or an “ Arabic/Farsi” mode (right-to-left notation).

Since Latin characters as well as Arabic/Farsi characters may be combined in a format, every field of the format must also be assigned a Latin or Arabic/Farsi mode. Furthermore, the “Change-code” key can be used to mix the characters in a field, which means that you can enter both Arabic/Farsi as well as Latin characters together in the same field. It is, however, also possible to lock the “Change-code” key so that only characters of the selected mode may be entered in a particular field.

Notes

Arabic/Farsi formats are not supported by FHS-DM.

The screens for Arabic/Farsi support are only output in English and are slightly different from the default screens.

Note that when Arabic/Farsi mode is set, processing occurs from right to left. This means, for example, that the addressing aids are also generated from right to left.

16.1 Creating formats

If the character set for an 8-bit format to be generated is defined as Arabic/Farsi, the extended screen for creating a format, which includes the additional field “Global mode of format, is output automatically. You can then define the global mode of the format in this screen.

```
I F G                          CREATING A FORMAT                          0101

Name of format library:   MY.LIBRARY

Do you wish to view the directory first?  If so, mark and press SEND.

Format name  ....  : ARABIC  (only up to 7 chars for FHS-DM help panel)

Format version ..  : ONE
                                        (only if required)

Password  .......  :                    (only if required)

Format identifier  :                    (only if required)

Global mode of format is Latin?   : NO

To draft the format image:   SEND                                 For help:  K3
To select another function:  F3
```

Depending on which global mode is selected, all subsequent screens will either use the Arabic/Farsi mode or the Latin mode. Note, however, that the modes only differ with respect to the direction of writing, not the representation of the screens. The default value for the global mode can be defined in the user profile screen “Default values for format display attributes”.

Note

Note that once a global mode is selected for a format, it can no longer be changed.

```
I F G                    MODIFYING YOUR USER PROFILE  USERPRO            0705
                  DEFAULT VALUES FOR FORMAT DISPLAY ATTRIBUTES




    ICE Character sets used:
          0.:  *STD          1.:               2.:               3.:
          4.:                5.:               6.:               7.:

    Holes color   : U (W=white, G=gray, U=undefined)

    Screen dimensions  : 24 x 080 (lines x colums)

    Coded character set:

    Global mode of format is Latin: NO

    Change-code key is locked     : NO

To check any changes:  SEND                                    For help:  K3
To store your input:   F1                                 To abort function:  K2
```

If an Arabic/Farsi character set is specified in this screen, the screen “Default values for the use of IFG” is output as shown below.

```
I F G                      MODIFY YOUR USER PROFILE     USERPRO            A703
                       DEFAULT VALUES FOR THE USE OF IFG

Password for the user profile:

Special characters for image drafting (if not required, blanks):
The 1st column is for use with formats in Arabic/Farsi mode, the 2nd for Latin
     input field                < > repetition character          delete line
 # # input field, numeric       $ $ start-of-field char.          repeat line
 & & output field (protected)   " " field alignment
     single choice field            multiple choice field         list

In IFG, lowercase letters in text fields are to be converted to upper case
- during image drafting      : NO               - in printouts           : YES

Image drafting/modification
Start with large window      : NO                 Retain field attributes: NO
Max. number of used lines/columns in formats (max. 43 / 132)        : 23 / 080
Format display: with char. set attributes: NO     Free cursor movement  :  YES
Character set library:

To check any changes:  SEND                                    For help:  K3
To store your input:   F1                                 To abort function:  K2
```

Each of the special characters used for image drafting is shown duplicated in this screen. The special characters of the first column are used to create Arabic/Farsi formats with the Arabic/Farsi mode; the special characters of the second column are used in the creation of Arabic/Farsi formats in Latin mode. Only Arabic/Farsi characters (i.e. characters specific to

the Arabic/Farsi character set) may be entered in the first column; the second column can only contain Latin characters. Note that some Arabic characters have the same graphical representation on the screen as the Latin characters (e.g. # exists as a graphics character in both scripts).

16.2 Drafting an image for Arabic/Farsi formats

When you draft an image for Arabic/Farsi format, only the special characters assigned to the selected global mode of the format are output for drafting that format.

After you select the Arabic/Farsi mode, the screen “Drafting an image for format” is displayed as shown below.

```
I F G                DRAFTING AN IMAGE FOR FORMAT  ARABIC                  A110
                       (For help: K3)     VERSION  ONE
   _ Input field               < Repetition character         Delete line
   # Input field, numeric       $ Start-of-field char.         Repeat line
   & Output field              " Alignment
     Single choice field         Multiple choice field        List
8---------7---------6--------- Line 001 to line 010 --------2---------1---------









8---------7---------6---------5---------4---------3---------2---------1---------
To check and edit your input:                    SEND
To insert blank line  or  image of format                To store and display:  F1
(version:                                                Window fwd. 05 lines:  F2
              at line 001 :             MAR and SEND     Window back 05 lines:  F3
To switch to large window  :            MAR and SEND     To abort function:     K2
```

The special characters shown in this example are the Arabic/Farsi special characters. Only these characters can be used to draft the format; all other characters are output as text.

If a field of this format is assigned the Latin mode, the field will be initially treated as a field in Arabic/Farsi mode. This means that the Arabic/Farsi special characters must also be used for drafting the image. Latin mode is set only on editing the input validation attributes of the field (Inverse mode = yes, see also page 251).
By contrast, a field of an Arabic/Farsi format with Latin mode that is to be assigned the Arabic/Farsi mode will be initially treated as a field with Latin mode. The mode change only takes effect on modifying the input validation attributes of the field.

When the Arabic/Farsi mode is set, the cursor moves from right to left. The first column of the format is then at the right margin of the screen, i.e. special characters for “Delete line”, “Repeat line”, and the characters for choice fields and lists must be entered on the extreme right of the screen in Arabic/Farsi formats. If text in Latin mode is to be entered, the “Change-code” key must be pressed to switch from the Arabic/Farsi script to the Latin script.

When processing from right to left, the start-of field character must be entered at the extreme right of the field. The following order from right to left must be maintained in order to enter the repetition character: the field definition character comes first, followed by the dimension, and finally the repetition character. The dimension must be entered in Latin mode, which means that you must use the “Change-code” key to first switch to Latin mode.

If you are creating a format with dialog extension, you must enter the global attributes of the format before drafting the image for it. Although the Arabic/Farsi formats, except for being processed from right to left, do not differ from the default formats, the following must be observed:

– The start-of-field character must be introduced on the right side of the text of each menu in the menu bar.

– The prompt character of the command area is output on the right side of the screen.

– The input field of the command area occupies the remainder of the line on the left.

16.3 Single choice fields

When you create a single-choice field, the following screen is output. The screen in this example is displayed in Arabic/Farsi mode.

```
I F G                    EDITING A SINGLE CHOICE FIELD                    AKO3
                               (FOR HELP: K3)
Prompt text ..............:                                          :scimoC
Prompt location ..........: 2 (1.Above 2.Before)
Global help ..............:
Amount of choices per line : 2
                                    Text                                   Nr
                                                                  xiretsA.1
                                                              esuoM ykciM.2
                                                               ekuL ykcuL.3
                                                                 namrepuS.4
                                                                giP ykroP.5
                                                              kcuD dlanoD.6




 To    edit       the aliases of the choices      MAR and SEND
 To    insert     a choice before choice nr.      MAR and SEND
 To    delete     the choice nr.                  MAR and SEND    Forward:   F2
 To store the single choice field:  F1                          Backward:   F3
 To abort function              :  K2                To check your input: SEND
```

In contrast to the default screen, choice numbers and the text for the individual choices are shown inverted in Arabic/Farsi formats. The choices can then be entered as shown in the output of the format. The input field of the single-choice field is inserted on the right before the first choice.

The single-choice field defined earlier would then be output as follows::

```
esuoM ykciM.2          xiretsA.1 _ :scimoC
   namrepuS.4       ekuL ykcuL.3
kcuD dlanoD.6        giP ykroP.5
```

As opposed to the default case, the entire choice field is shown totally mirrored in the output. All elements of the choice can, however, also be introduced in Latin mode by pressing the “Change-code” key.

The internal choice number of a selection is entered via the screen for “Editing internal choice numbers of single-choice fields” (see page 84). Since these internal choice numbers are used in the application program, the internal choice number is always entered in Latin mode, regardless of the “Change-code” key.

16.4 Multiple-choice fields

When you create a multiple-choice field, the following screen is output. The screen in this example is shown in Arabic/Farsi mode.

```
I F G                    EDITING A MULTIPLE CHOICE FIELD                    AKO4
                                (FOR HELP: K3)
Prompt text ..............:                                          :scimoC
Prompt location ...........: 2 (1.Above 2.Before)
Global help ..............:
Amount of choices per line : 2

                                    Text                                    Nr
                                                                   xiretsA 01
                                                                esuoM ykciM 02
                                                                 ekuL ykcuL 03
                                                                   namrepuS 04
                                                                  giP ykroP 05
                                                                kcuD dlanoD 06
                                                                           07
                                                                           08
                                                                           09
                                                                           10

 To    insert     a choice before choice nr.      MAR and SEND
 To    delete     the choice nr.                  MAR and SEND    Forward:   F2
 To store the multiple choice field: F1                          Backward:   F3
 To abort function                 : K2               To check your input: SEND
```

In contrast to the default screen, the choice numbers and text for the individual choices are shown inverted in Arabic/Farsi formats. The choices can then be entered as represented in the output format. The input fields for the multiple-choice field are inserted on the right before each choice.

The multiple-choice field defined earlier would then be output as follows:

```
esuoM ykciM  _          xiretsA  _ :scimoC
   namrepuS  _        ekuL ykcuL  _
kcuD dlanoD  _         giP ykroP  _
```

As opposed to the default case, the entire choice field is completely mirrored. All elements of the choice can, however, also be introduced in Latin mode by pressing the “Change-code” key.

16.5 Creating lists

When a list is created, the screen for “Creating a list” is output. Although this screen is output in Arabic/Farsi mode, its layout corresponds to that of the default screen on page 89. However, scrolling information is entered to the left of the positioning character in the list title, i.e. the last found “!” is read from right to left.

```
I F G                          EDITING A LIST                              AKO2

 Scrolling information::erom              Separation lines           : Yes
 Global help          :                       after column titels only: Yes
 Amount of records    : 5                 Separation character       :

 List title: (Scrolling information positionning character: ! )


 Column titles:                                                      :eltit


 Record layout:                                                Bnmuloc | Anmuloc

                                                               @@@@@@@ |

 End of Data marker:                                        *** ATAD  FO DNE ***


 To store the List definition:  F1                 To check your input: SEND
 To abort function:             K2                              For Help:   K3
```

The same special characters must be used for creating the list as for drafting the image. In other words, if the Arabic/Farsi global mode was set when drafting the image, the Arabic/Farsi special characters must be used, and if the global mode is Latin, the Latin special characters must be used. Field modes can be modified later if required.

16.6 Creating a pull-down menu

A pull-down menu consists of only single-choice fields. The creation of a pull-down menu is therefore treated as the construction of single-choice fields. A detailed description can be found in the section “Single choice fields” on page 248.

16.7 Viewing a format

In this function, the Arabic/Farsi format is always shown in Arabic/Farsi mode, which means that the cursor always moves from right to left. No other special processing for Arabic/Farsi formats is involved in this function.

16.8 Modifying a format

Different format elements such as the image and attributes of the format can be modified in the same way as they were created. Note that neither the CCSN (Coded Character Set Name) nor the global mode of a format can be changed. The mode of individual fields can be changed, however, and the “Change-code” key can also be locked or unlocked.

16.8.1 Modifying the field mode

Some formats may need to have Latin and Arabic/Farsi fields at the same time, e.g. to enter an Arabic/Farsi name in a database. By default, fields are assigned the global mode that was used when drafting the image. This field mode can be optionally changed when “modifying the input validation attributes of the field”.

```
I F G           INPUT/VALIDATION ATTRIBUTES OF FORMAT  ARABIC              A307
                 (For help: K3)              VERSION  ONE
                                      Field type: INPUT FIELD      Length: 001
Mandatory input  : NO            Automatic input: NO            Protected: NO
Min. input length: 000           Selectable     : NO            NUM lock : NO
                                 Inverse mode   : NO

Any character: YES
Arithmetic   : NO      Signed: NO      Decimal places: 00    Digit groups: NO
Alphabetic   : NO
Date         : NO
------------------------------  Line 001 to line 008  ------------------------------
                                                                  ____: CIBARA
                                                                  ____:  NITAL









--------------------------------------------------------------------------------
To process next field:            SEND         To store and display format:  F1
To select another field:  MAR and SEND         To move window forward:       F2
To abort function:                K2           To move window back:          F3
```

In order to invert a field mode, YES must be entered in the “Inverse mode” field. The field will then be assigned the opposite of the global mode.

If the mode of a field is modified, the fill character of that field must be changed accordingly. In other words, if a field with Arabic/Farsi mode is changed to a field with Latin mode, the fill character must also be translated to the appropriate fill character in the other script. Note that the fill characters specified in the user profile only apply to Latin fields by default; there is no value entered there for Arabic/Farsi fill characters. The conversion from the Latin fill characters in the user profile to the corresponding Arabic/Farsi fill characters occurs directly when creating an Arabic/Farsi format.

Arithmetic fields and date and time fields in FHS may only be entered in Arabic digits. Indian-language digits are rejected immediately.

16.8.2 Locking the “Change-code” key

The script being used can be switched within a field. It is thus possible to enter Arabic/Farsi script in a field with Latin mode or Latin script in a field with Arabic/Farsi mode. The change in script is activated by pressing the “Change-code” key on the keyboard.

If desired, the “Change-code” key can be locked to prevent switching between these two scripts. If this key has been locked, only Latin script will be permitted in fields with Latin mode, and only Arabic/Farsi script may be entered in fields with Arabic/Farsi mode. The “Change-code” key can be locked in the following screen. A default value can be entered in the user profile in the screen “Default values for format display attributes” illustrated on page 245.

```
I F G                GENERAL ATTRIBUTES OF FORMAT  ARABIC                  030A
                                          VERSION  ONE
                              DISPLAY ATTRIBUTES

    Password:                        Format identifier:

    The format uses the following ICE-character sets:
          0.:  *STD          1.:                2.:                3.:
          4.:                5.:                6.:                7.:

    Holes color    : U (W=white, G=gray, U=undefined)

    Screen dimensions  : 24 x 080 (lines x colums)

    I Start line when used as a partial format: 00
    (If start line = 00 then the format is a full format)

    Coded character set of format: EDF046

    Change-code key locked: NO

To check your input:   SEND                  To store the modified format:  F1
To return without storing any changes:   F3                       For help:  K3
```

16.8.3 Limit values for arithmetic fields

When creating FHS-DE formats, you can define limits for arithmetic fields. These limits must be defined in IFG with Arabic digits, since Indian digits are rejected.

16.8.4 Color support for Arabic/Farsi formats

The default function “Modify display attributes of fields” (see page 94) can be used to assign a color to a field. This is not possible for Arabic/Farsi formats.

16.9 Combining formats

The default rules for combining formats described in the chapter “Combining formats” on page 141 apply. The only point to be noted is that formats with different global modes cannot be combined in the case of Arabic/Farsi formats.

16.10 Printing a format

As when printing standard 8-bit formats, you should ensure that only compatible formats are contained in the file to be printed. You should also make sure that the printer supports the character set of these formats.

The following additional information is printed for Arabic/Farsi formats:

– the global mode of the format

– whether the field has the same global mode as the format (only for the long form)

16.11 Converting a format

8-bit formats that were not created with XHCS need to be converted. The converter is called automatically by IFG when such formats are copied.

16.12 Processing key lists

There is no special support for Arabic/Farsi key lists. Since the commands and texts assigned to each function key consist of text only, it is not necessary to define a global mode or field mode for the function keys. The text and command fields of function keys are always treated as Latin fields. Arabic/Farsi text can only be entered by pressing the “Change-code” key.

16.13 Processing message members

There is no special support for Arabic/Farsi message members. Since the messages are all composed of text only, a global mode or field mode need not be defined for message members. Message texts are always treated as Latin fields; Arabic/Farsi text can only be entered by pressing the “Change-code” key.

17 Appendix

17.1 Examples of addressing aids

17.1.1 Assembler

Data transfer area not aligned, without attribute fields

Input formatting

```
         MACRO
         DELIVERI
*FORMAT  NAME: DELIVER
DELIVERI DS    0CL386
ADDRESSI DS    0C

NAMEI    DS    CL25
STREETI  DS    CL26
ZIPCODEI DS    CL5
CITYI    DS    CL24
ADDRESSG EQU *-ADDRESSI
CUSTMRNI DS    CL12
DATEI    DS    CL10
ARTICLEI DS    CL5
DESIGI   DS    CL28
QTYI     DS    CL6
UNITPRII DS    CL7
PRICEI   DS    CL14
         DS    CL5
         DS    CL28
         DS    CL6
         DS    CL7
         DS    CL14
         DS    CL5
         DS    CL28
         DS    CL6
         DS    CL7
```

```
          DS    CL14
          DS    CL5
          DS    CL28
          DS    CL6
          DS    CL7
          DS    CL14

SUBTOTAI DS    CL15
SALESTAI DS    CL14
TOTALI   DS    CL15
         MEND
```

Output formatting

```
          MACRO
          DELIVERO
*FORMAT   NAME: DELIVER
DELIVERO DS    OCL386
ADDRESSO DS    OC
NAMEO    DS    CL25
STREETO  DS    CL26
ZIPCODEO DS    CL5
CITYO    DS    CL24
ADDRESSF EQU *-ADDRESSO
CUSTMRNO DS    CL12
DATEO    DS    CL10
ARTICLEO DS    CL5
DESIGO   DS    CL28
QTYO     DS    CL6
UNITPRIO DS    CL7
PRICEO   DS    CL14
          DS    CL5
          DS    CL28
          DS    CL6
          DS    CL7
          DS    CL14
          DS    CL5
          DS    CL28
          DS    CL6
          DS    CL7
          DS    CL14
          DS    CL5
          DS    CL28
          DS    CL6
          DS    CL7
          DS    CL14
SUBTOTAO DS    CL15
```

```
SALESTAO DS    CL14
TOTALO   DS    CL15
         MEND
```

Data transfer area with separate attribute blocks and field contents

```
*
           .
           .
           .
         DELIVER
*FORMAT  NAME: DELIVER
*
EUAGA    DSECT
GARCMAIN DS    F           RC MAIN
GARCCTGR DS    H           RC CATEGORY
GARCREAS DS    H           RC REASON
GAFLDMOD DS    CL1         FIELDS MOD
GAFLDDET DS    CL1         FIELDS DET
GAFLDVAL DS    CL1         FIELDS VALID
GAUSEXRC DS    CL1         USER EXIT RC
GAFLDUND DS    CL1         UNDEF VALUES
GAIKEYCL DS    CL1         INPUT KEY CLASS
GAIKEYNB DS    H           INPUT KEY NUMBER
         DS    CL4         RESERVED
GAINTCTL DS    CL1         INIT CTL
GAINTOPT DS    CL1         INIT OPT
GATABCTL DS    CL1         TAB CTL
GAFCTLCK DS    CL1         FCT LOCK
GAVMICTL DS    CL1         VMI CTL
GAHMICTL DS    CL1         HMI CTL
         DS    CL2         RESERVED
GACYCCTL DS    CL1         CYCLE CTL
GACOPCTL DS    CL1         COPY CTL
GAARMCTL DS    CL1         ALARM CTL
GAHOLECO DS    CL1         HOLE COLOR
GADISSEL DS    CL1         DISPLAY SEL
GALEVSEL DS    CL1         LEVEL SEL
GAOUTMOD DS    CL1         OUTPUT MODE
GACURCTL DS    CL1         CURSOR CTL
GACURPOS DS    F           CURSOR POS
GAUSEXCT DS    CL1         USER EXIT CTL
GALANEXT DS    CL1         LANGUAGE-EXT
GASTARTL DS    H           START LINE
GAPKEYST DS    CL8         P KEY SET
EUAGAL   EQU   *-EUAGA
*
```

```
*
*
DELIVERB DSECT
DELIVERS DS    OCL4       BASIC ATTR
DELIVERI DS    CL1          INPUT STATE
DELIVERT DS    CL1          INPUT STATE ACT
DELIVERE DS    CL1          EDIT STATE
DELIVERO DS    CL1          OUTPUT CTL
DELIVERL EQU   *-DELIVERB
*
<SYSECT> CSECT
*
         DS    0F
DELIVER  DS    0CL00529
DELIVERG DS    CL(EUAGAL)
*
*
DELIVERA DS    0CL00116
NAMEA    DS    CL(DELIVERL)
STREETA  DS    CL(DELIVERL)
ZIPCODEA DS    CL(DELIVERL)
CITYA    DS    CL(DELIVERL)
CUSTMRNA DS    CL(DELIVERL)
DATEA    DS    CL(DELIVERL)
ARTICLEA DS    CL(DELIVERL)
DESIGA   DS    CL(DELIVERL)
QTYA     DS    CL(DELIVERL)
UNITPRIA DS    CL(DELIVERL)

PRICEA   DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
         DS    CL(DELIVERL)
SUBTOTAA DS    CL(DELIVERL)
SALESTAA DS    CL(DELIVERL)
TOTALA   DS    CL(DELIVERL)
```

```
*
*
DELIVERD DS    0CL00361
ADDRESSD DS    0C
NAMED    DS    CL025
STREETD  DS    CL026
ZIPCODED DS    CL005
CITYD    DS    CL024
ADDRESSG EQU   *-ADDRESSD
CUSTMRND DS    CL012
DATED    DS    CL014
ARTICLED DS    CL005
DESIGD   DS    CL028
QTYD     DS    CL005
UNITPRID DS    CL006
PRICED   DS    CL011
         DS    CL005
         DS    CL028
         DS    CL005
         DS    CL006
         DS    CL011
         DS    CL005
         DS    CL028
         DS    CL005
         DS    CL006
         DS    CL011
         DS    CL005
         DS    CL028
         DS    CL005
         DS    CL006
         DS    CL011
SUBTOTAD DS    CL012
SALESTAD DS    CL011
TOTALD   DS    CL012
*
            .
            .
            .
```

17.1.2 COBOL

Data transfer area unaligned, without attribute fields

Input formatting

```
* IFG COPY   NAME: EDELIVER  RELATED COPY NAME: ADELIVER
*     FORMAT NAME: DELIVER   USER AREA LENGTH : 00386   UTM TYPE: *
     41  ADDRESSI.
       42  NAMEI                          PIC X(025).
       42  STREETI                        PIC X(026).
       42  ZIPCODEI                       PIC 9(005).
       42  CITYI                          PIC X(024).
     41  CUSTOMERNOI                    PIC 9(012).
     41  DATEI                          PIC X(010).
     41  ARTLINE-TABI.
       42  ARTLINEI                        OCCURS 04 TIMES.
         43    ARTICLENUMBERI                 PIC 9(005).
         43    DESIGNATIONI                   PIC X(028).
         43    QUANTITYI                      PIC 9(006).
         43    UNITPRICEI                     PIC 9(007).
         43    PRICEI                         PIC X(014).
     41  SUBTOTALI                      PIC X(015).
     41  SALESTAXI                      PIC X(014).
     41  TOTALI                         PIC X(015).
```

Output formatting

```
* IFG COPY   NAME: ADELIVER  RELATED COPY NAME: EDELIVER
*     FORMAT NAME: DELIVER   USER AREA LENGTH : 00386   UTM TYPE: *
     41  ADDRESSO.
       42  NAMEO                          PIC X(025).
       42  STREETO                        PIC X(026).
       42  ZIPCODEO                       PIC 9(005).
       42  CITYO                          PIC X(024).
     41  CUSTOMERNOO                    PIC 9(012).
     41  DATEO                          PIC X(010).
     41  ARTLINE-TABO
       42  ARTLINEO                        OCCURS 04 TIMES.
         43    ARTICLENUMBERO                 PIC 9(005).
         43    DESIGNATIONO                   PIC X(028).
         43    QUANTITYO                      PIC 9(006).
         43    UNITPRICEO                     PIC 9(007).
         43    PRICEO                         PIC X(014).
     41  SUBTOTALO                      PIC X(015).
     41  SALESTAXO                      PIC X(014).
     41  TOTALO                         PIC X(015).
```

Data transfer area with separate attribute blocks and field contents

```
* IFG COPY   NAME: DELIVER
*     FORMAT NAME: DELIVER  USER AREA LENGTH : 00529   UTM TYPE: #
************************************************************
*                 GLOBAL ATTRIBUTE BLOCK                   *
************************************************************
    40 DELIVER-GLOBALS.
*        FORM-RETURNCODE
      41 RC-MAIN                      PIC 9(5) COMP SYNC.
      41 RC-CATEGORY                  PIC 9(4) COMP.
      41 RC-REASON                    PIC 9(4) COMP.
*        FORM-INDICATORS
      41 FIELDS-MOD                   PIC X.
      41 FIELDS-DET                   PIC X.
      41 FIELDS-VALID                 PIC X.
      41 USER-EXIT-RC                 PIC X.
      41 FIELDS-UNDEFINED             PIC X.
*        INPUT-IDENTIFICATION
      41 INPUT-KEY-CLASS              PIC X.
      41 INPUT-KEY-NUMBER             PIC 9(4) COMP.
      41 FILLER                       PIC X(4).
*        DEVICE-CONTROLS
      41 INIT-CTL                     PIC X.
      41 INIT-OPT                     PIC X.
      41 TAB-CTL                      PIC X.
      41 FCT-LOCK                     PIC X.
      41 VMI-CTL                      PIC X.
      41 HMI-CTL                      PIC X.
      41 FILLER                       PIC X(2).
*        OUTPUT-CONTROLS
      41 CYCLE-CTL                    PIC X.
      41 COPY-CTL                     PIC X.
      41 ALARM-CTL                    PIC X.
      41 HOLE-COLOR                   PIC X.
*        FORM-CONTROLS
      41 DISPLAY-SEL                  PIC X.
      41 LEVEL-SEL                    PIC X.
      41 OUTPUT-MODE                  PIC X.
      41 CURSOR-CTL                   PIC X.
      41 CURSOR-POS                   PIC 9(5) COMP.
      41 USER-EXIT-CTL                PIC X.
      41 LANGUAGE-EXT                 PIC X.
      41 STARTLINE                    PIC 9(4) COMP.
*
      41 P-KEY-SET                    PIC X(8).
```

```
*************************************************************
*                  FIELD ATTRIBUTE BLOCKS                   *
*************************************************************
    40 DELIVER-ATTR.
      41 ADDRESS-FAB.
        42 NAME-FAB.
          43 BASIC-ATTR.
            44 INPUT-STATE              PIC X.
            44 INPUT-STATE-ACT          PIC X.
            44 EDIT-STATE               PIC X.
            44 OUTPUT-CTL               PIC X.

        42 STREET-FAB.
          43 BASIC-ATTR.
            44 INPUT-STATE              PIC X.
            44 INPUT-STATE-ACT          PIC X.
            44 EDIT-STATE               PIC X.
            44 OUTPUT-CTL               PIC X.
        42 ZIPCODE-FAB.
          43 BASIC-ATTR.
            44 INPUT-STATE              PIC X.
            44 INPUT-STATE-ACT          PIC X.
            44 EDIT-STATE               PIC X.
            44 OUTPUT-CTL               PIC X.
        42 CITY-FAB.
          43 BASIC-ATTR.
            44 INPUT-STATE              PIC X.
            44 INPUT-STATE-ACT          PIC X.
            44 EDIT-STATE               PIC X.
            44 OUTPUT-CTL               PIC X.
      41 CUSTOMERNO-FAB.
        42 BASIC-ATTR.
          43 INPUT-STATE              PIC X.
          43 INPUT-STATE-ACT          PIC X.
          43 EDIT-STATE               PIC X.
          43 OUTPUT-CTL               PIC X.
      41 DATE-FAB.
        42 BASIC-ATTR.
          43 INPUT-STATE              PIC X.
          43 INPUT-STATE-ACT          PIC X.
          43 EDIT-STATE               PIC X.
          43 OUTPUT-CTL               PIC X.
      41 ARTLINE-TAB-FAB.
        42 ARTLINE-FAB                  OCCURS 04 TIMES.
          43 ARTICLENUMBER-FAB.
            44 BASIC-ATTR.
              45 INPUT-STATE              PIC X.
              45 INPUT-STATE-ACT          PIC X.
```

```
           45 EDIT-STATE                PIC X.
           45 OUTPUT-CTL                PIC X.
        43 DESIGNATION-FAB.
          44 BASIC-ATTR.
           45 INPUT-STATE               PIC X.
           45 INPUT-STATE-ACT           PIC X.
           45 EDIT-STATE                PIC X.
           45 OUTPUT-CTL                PIC X.
        43 QTY-FAB.
          44 BASIC-ATTR.
           45 INPUT-STATE               PIC X.
           45 INPUT-STATE-ACT           PIC X.
           45 EDIT-STATE                PIC X.
           45 OUTPUT-CTL                PIC X.
        43 UNITPRICE-FAB.
          44 BASIC-ATTR.
           45 INPUT-STATE               PIC X.
           45 INPUT-STATE-ACT           PIC X.
           45 EDIT-STATE                PIC X.
           45 OUTPUT-CTL                PIC X.

        43 PRICE-FAB.
          44 BASIC-ATTR.
           45 INPUT-STATE               PIC X.
           45 INPUT-STATE-ACT           PIC X.
           45 EDIT-STATE                PIC X.
           45 OUTPUT-CTL                PIC X.
      41 SUBTOTAL-FAB.
        42 BASIC-ATTR.
          43 INPUT-STATE              PIC X.
          43 INPUT-STATE-ACT          PIC X.
          43 EDIT-STATE               PIC X.
          43 OUTPUT-CTL               PIC X.
      41 SALESTAX-FAB.
        42 BASIC-ATTR.
          43 INPUT-STATE              PIC X.
          43 INPUT-STATE-ACT          PIC X.
          43 EDIT-STATE               PIC X.
          43 OUTPUT-CTL               PIC X.
      41 TOTAL-FAB.
        42 BASIC-ATTR.
          43 INPUT-STATE              PIC X.
          43 INPUT-STATE-ACT          PIC X.
          43 EDIT-STATE               PIC X.
          43 OUTPUT-CTL               PIC X.
    40 DELIVER-ATTR-TAB REDEFINES DELIVER-ATTR  OCCURS 029 TIMES.
      41 BASIC-ATTR.
        42 INPUT-STATE              PIC X.
```

```
        42 INPUT-STATE-ACT          PIC X.
        42 EDIT-STATE               PIC X.
        42 OUTPUT-CTL               PIC X.
************************************************************
*                     FIELD DATA PART                      *
************************************************************
    40 DELIVER-DATA SIGN IS TRAILING SEPARATE.
      41 ADDRESS.
        42 NAME                           PIC X(025).
        42 STREET                         PIC X(026).
        42 ZIPCODE                        PIC X(005).
        42 CITY                           PIC X(024).
      41 CUSTOMERNO                     PIC X(012).
      41 DATE                           PIC X(014).
      41 ARTLINE-TAB.
        42 ARTLINE                        OCCURS 04 TIMES.
          43 ARTICLENUMBER                 PIC X(005).
          43 DESIGNATION                   PIC X(028).
          43 QTY                           PIC  9(005).
          43 UNITPRICE                     PIC S9(003)V9(002).
          43 PRICE                         PIC S9(008)V9(002).
      41 SUBTOTAL                       PIC S9(009)V9(002).
      41 SALESTAX                       PIC S9(008)V9(002).
      41 TOTAL                          PIC S9(009)V9(002).
```

17.1.3 C

Data transfer area unaligned, without attribute fields

```
/* FORMAT NAME      : DELIVER */
/* USER AREA LENGTH :   386    */

   typedef struct {
           struct {
              char NAME                              [  25 ] ;
              char STREET                            [  26 ] ;
              char ZIPCODE                           [   5 ] ;
              char CITY                              [  24 ] ;
           } ADDRESS ;
           char CUSTOMERNO                       [  12 ] ;
           char DATE                             [  10 ] ;
           struct {
              char ARTICLENUMBER                     [   5 ] ;
              char DESIGNATION                       [  28 ] ;
              char QUANTITY                          [   6 ] ;
              char UNITPRICE                         [   7 ] ;
              char PRICE                             [  14 ] ;
           } ARTLINE     [  4 ] ;
           char SUBTOTAL                         [  15 ] ;
           char SALESTAX                         [  14 ] ;
           char TOTAL                            [  15 ] ;
   } DELIVER ;
```

Data transfer area with separate attribute blocks and field contents

```
/* FORMAT NAME      : DELIVER  */
/* USER AREA LENGTH :   529    */
   typedef struct {
           struct {
              char INPUT_STATE ;
              char INPUT_STATE_ACT ;
              char EDIT_STATE ;
              char OUTPUT_CTL ;
           } BASIC_ATTR ;
   } DELIVER_ATTR_T ;
   typedef struct {
/*************************************************************/
/*                 GLOBAL ATTRIBUTE BLOCK                    */
/*************************************************************/
           struct {
           /* FORM_RETURNCODE */
              long int RC_MAIN ;
```

```
            short int RC_CATEGORY ;
            short int RC_REASON ;
         /* FORM_INDICATORS */
            char FIELDS_MOD ;
            char FIELDS_DET ;
            char FIELDS_VALID ;
            char USER_EXIT_RC ;
            char FIELDS_UNDEFINED;
         /* INPUT_IDENTIFICATION */
            char INPUT_KEY_CLASS ;
            short int INPUT_KEY_NUMBER ;
            char RESERVED2            [ 4 ] ;
         /* DEVICE_CONTROLS */
            char INIT_CTL ;
            char INIT_OPT ;
            char TAB_CTL ;
            char FCT_LOCK ;
            char VMI_CTL ;
            char HMI_CTL ;
            char RESERVED3            [ 2 ] ;
         /* OUTPUT_CONTROLS */
            char CYCLE_CTL ;
            char COPY_CTL ;
            char ALARM_CTL ;
            char HOLE_COLOR;
         /* FORM_CONTROLS */
            char DISPLAY_SEL ;
            char LEVEL_SEL ;
            char OUTPUT_MOD ;
            char CURSOR_CTL ;
            long int CURSOR_POS ;
            char USER_EXIT_CTL ;
            char LANGUAGE_EXT ;
            short int STARTLINE ;
         /*               */
            char P_KEY_SET            [ 8 ] ;
         } GLOBALS ;

/*********************************************************/
/*                 FIELD ATTRIBUTE BLOCKS                */
/*********************************************************/
         union {
            struct {
               struct {
                  DELIVER_ATTR_T NAME ;
                  DELIVER_ATTR_T STREET ;
                  DELIVER_ATTR_T ZIPCODE ;
                  DELIVER_ATTR_T CITY ;
```

```
                } ADDRESS ;
                DELIVER_ATTR_T CUSTOMERNO ;
                DELIVER_ATTR_T DATE ;
                struct {
                   DELIVER_ATTR_T ARTICLENUMBER ;
                   DELIVER_ATTR_T DESIGNATION ;
                   DELIVER_ATTR_T QTY ;
                   DELIVER_ATTR_T UNITPRICE ;
                   DELIVER_ATTR_T PRICE ;
                } ARTLINE   [  4 ] ;
                DELIVER_ATTR_T SUBTOTAL ;
                DELIVER_ATTR_T SALESTAX ;
                DELIVER_ATTR_T TOTAL ;
             } ATTR ;
             DELIVER_ATTR_T ATTR_TAB [   29 ] ;
          } ATTR_UNION ;
/*********************************************************/
/*                     FIELD DATA PART                   */
/*********************************************************/
          struct {
             struct {
                char NAME                          [  25 ] ;
                char STREET                        [  26 ] ;
                char ZIPCODE                       [   5 ] ;
                char CITY                          [  24 ] ;
             } ADDRESS ;
             char CUSTOMERNO                  [  12 ] ;
             char DATE                        [  14 ] ;
             struct {
                char ARTICLENUMBER                 [   5 ] ;
                char DESIGNATION                   [  28 ] ;
                char QUANTITY                      [   5 ] ;
                char UNITPRICE                     [   6 ] ;
                char PRICE                         [  11 ] ;
             } ARTLINE   [  4 ] ;
             char SUBTOTAL                    [  12 ] ;
             char SALESTAX                    [  11 ] ;
             char TOTAL                       [  12 ] ;
          } DATA ;
   } DELIVER ;
```

17.1.4 Pascal

Data transfer area unaligned, without attribute fields

```
PACKAGE BODY DELIVER ;
BEGIN
END.

PACKAGE DELIVER ;
(* FORMAT NAME   : DELIVER  *)
(* USER AREA LEN : 00386    *)

TYPE T_DELIVER  =
  RECORD
    ADDRESS                          (0000) :
      RECORD
        NAME                             (0000) : PACKED ARRAY
                                                  (.01..025.) OF CHAR;
        STREET                           (0025) : PACKED ARRAY
                                                  (.01..026.) OF CHAR;
        ZIPCODE                          (0051) : PACKED ARRAY
                                                  (.01..005.) OF CHAR;
        CITY                             (0056) : PACKED ARRAY
                                                  (.01..024.) OF CHAR;
      END;
    CUSTOMERNO                       (0080) : PACKED ARRAY
                                              (.01..012.) OF CHAR;
    DATE                             (0092) : PACKED ARRAY
                                              (.01..010.) OF CHAR;
    ARTLINE                          (0102) : ARRAY(.01..04.) OF
      RECORD
        ARTICLENUMBER                    (0000) : PACKED ARRAY
                                                  (.01..005.) OF CHAR;
        DESIGNATION                      (0005) : PACKED ARRAY
                                                  (.01..028.) OF CHAR;
        QTY                              (0033) : PACKED ARRAY
                                                  (.01..006.) OF CHAR;
        UNITPRICE                        (0039) : PACKED ARRAY
                                                  (.01..007.) OF CHAR;
        PRICE                            (0046) : PACKED ARRAY
                                                  (.01..014.) OF CHAR;
      END;
    SUBTOTAL                         (0342) : PACKED ARRAY
                                              (.01..015.) OF CHAR;
    SALESTAX                         (0357) : PACKED ARRAY
                                              (.01..014.) OF CHAR;
    TOTAL                            (0371) : PACKED ARRAY
```

```
                                                    (.01..015.) OF CHAR;
  END;

END.
```

Data transfer area with separate attribute blocks and field contents

```
PACKAGE BODY DELIVER;
BEGIN
END.
PACKAGE DELIVER;
(* FORMAT NAME   : DELIVER   *)
(* USER AREA LEN : 00529    *)
  RECORD
    BASIC_ATTR             (00000) :
      RECORD
        INPUT_STATE            (00000) :  CHAR;
        INPUT_STATE_ACT        (00001) :  CHAR;
        EDIT_STATE             (00002) :  CHAR;
        OUTPUT_CTL             (00003) :  CHAR;
      END;
  END;
TYPE T_DELIVER_GLOBALS    =
  RECORD
    (* FORM_RETURNCODE *)
    RC_MAIN                    (00000) :  INTEGER;
    RC_CATEGORY                (00004) :  SHORT INTEGER;
    RC_REASON                  (00006) :  SHORT_INTEGER;
    (* FORM_INDICATORS *)
    FIELDS_MOD                 (00008) :  CHAR;
    FIELDS_DET                 (00009) :  CHAR;
    FIELDS_VALID               (00010) :  CHAR;
    USER_EXIT_RC               (00011) :  CHAR;
    FIELDS_UNDEFINED           (00012) :  CHAR;
    (* INPUT_IDENTIFICATION *)
    INPUT_KEY_CLASS            (00013) :  CHAR;
    INPUT_KEY_NUMBER           (00014) :  SHORT_INTEGER;
    RESERVED2                  (00016) :  PACKED ARRAY
                                          (.01..04.) OF CHAR;
    (* DEVICE_CONTROLS *)
    INIT_CTL                   (00020) :  CHAR;
    INIT_OPT                   (00021) :  CHAR;
    TAB_CTL                    (00022) :  CHAR;
    FCT_LOCK                   (00023) :  CHAR;
    VMI_CTL                    (00024) :  CHAR;
    HMI_CTL                    (00025) :  CHAR;
```

```
    RESERVED3                  (00026) :  PACKED ARRAY
                                          (.01..02.) OF CHAR;
    (* OUTPUT_CONTROLS *)
    CYCLE_CTL                  (00028) :  CHAR;
    COPY_CTL                   (00029) :  CHAR;
    ALARM_CTL                  (00030) :  CHAR;
    HOLE_COLOR                 (00031) :  CHAR;
    (* FORM_CONTROLS *)
    DISPLAY_SEL                (00032) :  CHAR;
    LEVEL_SEL                  (00033) :  CHAR;
    OUTPUT_MOD                 (00034) :  CHAR;
    CURSOR_CTL                 (00035) :  CHAR;
    CURSOR_POS                 (00036) :  INTEGER;
    USER_EXIT_CTL              (00040) :  CHAR;
    LANGUAGE_EXT               (00041) :  CHAR;

    STARTLINE                  (00042) :  SHORT INTEGER;
    (*              *)
    P_KEY_SET                  (00044) :  PACKED ARRAY
                                          (.01..08.) OF CHAR;
  END;
TYPE T_DELIVER_ATTR       =
  RECORD
    ADDRESS                        (00000) :
      RECORD
        NAME                           (00000) :  T_DELIVER_FLD_ATTR;
        STREET                         (00004) :  T_DELIVER_FLD_ATTR;
        ZIPCODE                        (00008) :  T_DELIVER_FLD_ATTR;
        CITY                           (00012) :  T_DELIVER_FLD_ATTR;
      END;
    CUSTOMERNO                     (00016) :  T_DELIVER_FLD_ATTR;
    DATE                           (00020) :  T_DELIVER_FLD_ATTR;
    ARTLINE                        (00024) :  ARRAY(.01..04.) OF
      RECORD
        ARTICLENUMBER                  (00000) :  T_DELIVER_FLD_ATTR;
        DESIGNATION                    (00004) :  T_DELIVER_FLD_ATTR;
        QTY                            (00008) :  T_DELIVER_FLD_ATTR;
        UNITPRICE                      (00012) :  T_DELIVER_FLD_ATTR;
        PRICE                          (00016) :  T_DELIVER_FLD_ATTR;
      END;
    SUBTOTAL                       (00104) :  T_DELIVER_FLD_ATTR;
    SALESTAX                       (00108) :  T_DELIVER_FLD_ATTR;
    TOTAL                          (00112) :  T_DELIVER_FLD_ATTR;
  END;
TYPE T_DELIVER_DATA       =
  RECORD
    ADDRESS                        (00000) :
      RECORD
```

```
        NAME                          (00000) :  PACKED ARRAY
                                                 (.01..025.) OF CHAR;
        STREET                        (00025) :  PACKED ARRAY
                                                 (.01..026.) OF CHAR;
        ZIPCODE                       (00051) :  PACKED ARRAY
                                                 (.01..005.) OF CHAR;
        CITY                          (00056) :  PACKED ARRAY
                                                 (.01..024.) OF CHAR;
      END;
    CUSTOMERNO                    (00080) :  PACKED ARRAY
                                             (.01..012.) OF CHAR;
    DATE                          (00092) :  PACKED ARRAY
                                             (.01..014.) OF CHAR;
    ARTLINE                       (00106) :  ARRAY(.01..04.) OF
      RECORD
        ARTICLENUMBER                 (00000) :  PACKED ARRAY
                                                 (.01..005.) OF CHAR;
        DESIGNATION                   (00005) :  PACKED ARRAY
                                                 (.01..028.) OF CHAR;
        QTY                           (00033) :  PACKED ARRAY
                                                 (.01..005.) OF CHAR;
        UNITPRICE                     (00038) :  PACKED ARRAY
                                                 (.01..006.) OF CHAR;
        PRICE                         (00044) :  PACKED ARRAY
                                                 (.01..011.) OF CHAR;
      END;
    SUBTOTAL                      (00326) :  PACKED ARRAY
                                             (.01..012.) OF CHAR;
    SALESTAX                      (00338) :  PACKED ARRAY
                                             (.01..011.) OF CHAR;
    TOTAL                         (00349) :  PACKED ARRAY
                                             (.01..012.) OF CHAR;
  END;
TYPE T_DELIVER            =
  RECORD
    DELIVER_GLOBALS               (00000) : T_DELIVER_GLOBALS;
    DELIVER_ATTR                  (00052) : T_DELIVER_ATTR;
    DELIVER_DATA                  (00168) : T_DELIVER_DATA;
  END;
END.
```

17.1.5 PL/I

Data transfer area unaligned, without attribute fields

```
/* FORMAT NAME: DELIVER */
          20 ADDRESS,
             21 NAME                              CHAR(025),
             21 STREET                            CHAR(026),
             21 ZIPCODE                           CHAR(005),
             21 CITY                              CHAR(024),
          20 CUSTOMERNO                        CHAR(012),
          20 DATE                              CHAR(010),
          20 ARTLINE                           (01:04),
             21 ARTICLENUMBER                     CHAR(005),
             21 DESIGNATION                       CHAR(028),
             21 QUANTITY                          CHAR(006),
             21 UNITPRICE                         CHAR(007),
             21 PRICE                             CHAR(014),
          20 SUBTOTAL                          CHAR(015),
          20 SALESTAX                          CHAR(014),
          20 TOTAL                             CHAR(015),
          /**/
          AREA_LENGTH_DELIVER    BIN FIXED(15) INIT(00386) STATIC;
```

Data transfer area with separate attribute blocks and field contents

```
/* FORMAT NAME: DELIVER */
/************************************************************/
/*                  GLOBAL ATTRIBUTE BLOCK                  */
/************************************************************/
          19 DELIVER_GLOBALS UNAL,
            /* FORM_RETURNCODE */
            20 RC_MAIN                BINARY FIXED(31) ALIGNED,
            20 RC_CATEGORY            BINARY FIXED(15),
            20 RC_REASON              BINARY FIXED(15),
            /* FORM_INDICATORS */
            20 FIELDS_MOD             CHAR(1),
            20 FIELDS_DET             CHAR(1),
            20 FIELDS_VALID           CHAR(1),
            20 USER_EXIT_RC           CHAR(1),
            20 FIELDS_UNDEFINED       CHAR(1),
            /* INPUT_IDENTIFICATION */
            20 INPUT_KEY_CLASS        CHAR(1),
            20 INPUT_KEY_NUMBER       BINARY FIXED(15),
            20 RESERVED2              CHAR(4),
            /* DEVICE_CONTROLS */
            20 INIT_CTL               CHAR(1),
```

```
            20 INIT_OPT               CHAR(1),
            20 TAB_CTL                CHAR(1),
            20 FCT_LOCK               CHAR(1),
            20 VMI_CTL                CHAR(1),
            20 HMI_CTL                CHAR(1),
            20 RESERVED3              CHAR(2),
            /* OUTPUT_CONTROLS */
            20 CYCLE_CTL              CHAR(1),
            20 COPY_CTL               CHAR(1),
            20 ALARM_CTL              CHAR(1),
            20 HOLE_COLOR             CHAR(1),
            /* FORM_CONTROLS */
            20 DISPLAY_SEL            CHAR(1),
            20 LEVEL_SEL              CHAR(1),
            20 OUTPUT_MOD             CHAR(1),
            20 CURSOR_CTL             CHAR(1),
            20 CURSOR_POS             BINARY FIXED(31),
            20 USER_EXIT_CTL          CHAR(1),
            20 LANGUAGE_EXT           CHAR(1),
            20 STARTLINE              BINARY FIXED(15),
            /*              */
            20 P_KEY_SET              CHAR(8),
/*************************************************************/
/*                  FIELD ATTRIBUTE BLOCKS                   */
/*************************************************************/
         19 DELIVER_ATTR UNAL,
           20 ADDRESS_FAB,
             21 NAME_FAB,
               22 BASIC_ATTR,
                  23 INPUT_STATE          CHAR(1),
                  23 INPUT_STATE_ACT      CHAR(1),
                  23 EDIT_STATE           CHAR(1),
                  23 OUTPUT_CTL           CHAR(1),
             21 STREET_FAB
                LIKE DELIVER_ATTR.NAME_FAB,
             21 ZIPCODE_FAB
                LIKE DELIVER_ATTR.NAME_FAB,
             21 CITY_FAB
                LIKE DELIVER_ATTR.NAME_FAB,
           20 CUSTOMERNO_FAB
              LIKE DELIVER_ATTR.NAME_FAB,
           20 DATE_FAB
              LIKE DELIVER_ATTR.NAME_
           20 ARTLINE_FAB                (01:04),
             21 ARTICLENUMBER_FAB
                LIKE DELIVER_ATTR.NAME_FAB,
             21 DESIGNATION_FAB
                LIKE DELIVER_ATTR.NAME_FAB,
```

```
            21 QTY_FAB
               LIKE DELIVER_ATTR.NAME_FAB,
            21 UNITPRICE_FAB
               LIKE DELIVER_ATTR.NAME_FAB,
            21 PRICE_FAB
               LIKE DELIVER_ATTR.NAME_FAB,
          20 SUBTOTAL_FAB
             LIKE DELIVER_ATTR.NAME_FAB,
          20 SALESTAX_FAB
             LIKE DELIVER_ATTR.NAME_FAB,
          20 TOTAL _AB
             LIKE DELIVER_ATTR.NAME_FAB,
/*************************************************************/
/*                     FIELD DATA PART                       */
/*************************************************************/
       19 DELIVER_DATA UNAL,
         20 ADDRESS,
           21 NAME                          CHAR(025),
           21 STREET                        CHAR(026),
           21 ZIPCODE                       CHAR(005),
           21 CITY                          CHAR(024),
         20 CUSTOMERNO                    CHAR(012),
         20 DATE                          CHAR(014),
         20 ARTLINE                       (01:04),
           21 ARTICLENUMBER                 CHAR(005),
           21 DESIGNATION                   CHAR(028),
           21 QTY                           PIC'(005)9',
           21 UNITPRICE                     PIC'(003)9V(002)9S',
           21 PRICE                         PIC'(008)9V(002)9S',
         20 SUBTOTAL                      PIC'(009)9V(002)9S',
         20 SALESTAX                      PIC'(008)9V(002)9S',
         20 TOTAL                         PIC'(009)9V(002)9S',
       /**/
       AREA LENGTH_DELIVER    BINARY FIXED(15) INIT(00529) STATIC;
```

17.1.6 RPG

Data transfer area unaligned, without attribute fields

Input formatting

```
I*  FORMAT NAME   : DELIVER
I*  USER AREA LEN : 0386
I                                          1  25 NAME
I                                         26  51 STREE
I                                         52  56 ZIPCO
I                                         57  80 CITY
I                                         81  92 CUSTM
I                                         93 102 DATE
I                                        103 107 ARTCL
I                                        108 135 DESIG
I                                        136 141 QTY
I                                        142 148 UNITP
I                                        149 162 PRICE
I                                        163 167
I                                        168 195
I                                        196 201
I                                        202 208
I                                        209 222
I                                        223 227
I                                        228 255
I                                        256 261
I                                        262 268
I                                        269 282
I                                        283 287
I                                        288 315
I                                        316 321
I                                        322 328
I                                        329 342
I                                        343 357 SUBTO
I                                        358 371 SALES
I                                        372 386 TOTAL
```

Output formatting

```
O*  FORMAT NAME   : DELIVER
O*  USER AREA LEN : 0386
O                          NAME      25
O                          STREET    51
O                          ZIPCOD    56
O                          CITY      80
O                          CUSTMR    92
O                          DATE     102
```

```
0                       ARTICL   107
0                       IDENTI   135
0                       QTY      141
0                       UNITPR   148
0                       PRICE    162
0                                167
0                                195
0                                201
0                                208
0                                222
0                                227
0                                255
0                                261
0                                268
0                                282
0                                287
0                                315
0                                321
0                                328
0                                342
0                       SUBTOT   357
0                       SALEST   371
0                       TOTAL    386
```

Data transfer area with separate attribute blocks and field contents

Input formatting

```
I*  FORMAT NAME   : DELIVER
I*  USER AREA LEN : 0529
I                                  B   1   4ODELIRM
I*                                                    MAIN RETURNCODE
I                                  B   5   6ODELIRC
I*                                                    ERROR CATEGORY
I                                  B   7   8ODELIRR
I*                                                    ERROR REASON
I*
I*                                                  FHS RETURNCODE
I                                      9   9 DELIFM
I*                                                    FIELDS MOD
I                                     10  10 DELIFD
I*                                                    FIELDS DETECTED
I                                     11  11 DELIFV
I*                                                    FIELDS VALID
I                                     12  12 DELIUE
I*                                                    USER EXIT RC
I                                     13  13 DELIFU
I*                                                    FIELDS UNDEFINED
```

```
I*
I*                                          FORMATTING INDICATOR
I                                    14  14 DELIIC
I*                                        INPUT KEY CLASS
I                                B   15  16ODELIIN
I*                                        INPUT KEY NUMBER
I                                    17  20 DELIR2
I*                                        RESERVEDO2
I*
I*                                         INPUT IDENTIFICATION
I                                    21  21 DELICI
I*                                         INIT CONTROL
I                                    22  22 DELIIO
I*                                         INIT OPTION
I                                    23  23 DELITC
I*                                         TABULATOR CONTROL
I                                    24  24 DELIFL
I*                                        FUNCTION LOCK
I                                    25  25 DELIVC
I*                                        LINE SPACE
I                                    26  26 DELIHC
I*                                        CHARACTER SPACE
I                                    27  28 DELIR3
I*                                        RESERVEDO3
I*
I*                                         DEVICE CONTROLS
I                                    29  29 DELICC
I*                                        CYCLE CONTROL
I                                    30  30 DELICL
I*                                        COPY CONTROL
I                                    31  31 DELIAC
I*                                        ALARM CONTROL
I                                    32  32 DELIBC
I*                                        HOLE COLOR
I*
I*                                         OUTPUT CONTROLS
I                                    33  33 DELIDS
I*                                        DISPLAY SELECTION
I                                    34  34 DELILS
I*                                        LEVEL SELECTION
I                                    35  35 DELIOM
I*                                        OUTPUT MODE
I                                    36  36 DELICT
I*                                        CURSOR CONTROL
I                                B   37  4OODELICP
I*                                        CURSOR POSITION
I                                    41  41 DELIUC
```

```
I*                                                    USER EXIT CONTROL
I                                  42  42 DELILE
I*                                                    LANGUAGE EXTENSION
I                                B  43  44 DELISL
I*                                                    STARTLINE
I*
I*                                                  FORMATTING CONTROLS
I                                  45  52 DELIKS
I*                                                    P KEY SET
I*
I*
I**********
I**********       ************************************
I**********       *      FIELD  ATTRIBUTE  TABLE     *
I**********       ************************************
I**********       *  SUFFIX            MEANING       *
I**********       ************************************
I**********       *    F    ---->    FLD LENGTH       *
I**********       *    I    ---->      INPUT STATE    *
I**********       *    T    ---->      INPUT STATE ACT*
I**********       *    E    ---->      EDIT STATE     *
I**********       *    O    ---->      OUTPUT CONTROL *
I**********       *    S    ---->    BASIC ATTRIBUTES *
I**********       *    M    ---->    ATTR COMB        *
I**********       *    J    ---->      INPUT CONTROL  *
I**********       *    P    ---->      PROTECTION     *
I**********       *    N    ---->    FIELD INPUT      *
I**********       *    Y    ---->      INTENSITY      *
I**********       *    V    ---->      VISIBILITY     *
I**********       *    U    ---->      UNDERLINE      *
I**********       *    W    ---->      INVERSE        *
I**********       *    Q    ---->    DISPLAY CONTROL  *
I**********       *    C    ---->    COLOR            *
I**********       *    X    ---->    INITIAL CURSOR   *
I**********       *    R    ---->    EDIT RC          *
I**********       *    Z    ---->    ALIGNMENT BYTE   *
I**********       *    L    ---->    FLD ATTR TABLE LEN*
I**********       ************************************

I**********
I                                     53  53 NAMEI
I                                     54  54 NAMET
I                                     55  55 NAMEE
I                                     56  56 NAMEO
I                                     53  56 NAMES
I                                     53  56 NAMEL
I**********
I                                     57  57 STREEI
```

```
I                                        58  58 STREET
I                                        59  59 STREEE
I                                        60  60 STREEO
I                                        57  60 STREES
I                                        57  60 STREEL
I**********
I                                        61  61 ZIPCOI
I                                        62  62 ZIPCOT
I                                        63  63 ZIPCOE
I                                        64  64 ZIPCOO
I                                        61  64 ZIPCOS
I                                        61  64 ZIPCOL
I**********
I                                        65  65 CITYI
I                                        66  66 CITYT
I                                        67  67 CITYE
I                                        68  68 CITYO
I                                        65  68 CITYS
I                                        65  68 CITYL
I**********
I                                        69  69 CUSTMI
I                                        70  70 CUSTMT
I                                        71  71 CUSTME
I                                        72  72 CUSTMO
I                                        69  72 CUSTMS
I                                        69  72 CUSTML
I**********
I                                        73  73 DATEI
I                                        74  74 DATET
I                                        75  75 DATEE
I                                        76  76 DATEO
I                                        73  76 DATES
I                                        73  76 DATEL
I**********
I                                        77  77 ARTICI
I                                        78  78 ARTICT
I                                        79  79 ARTICE
I                                        80  80 ARTICO
I                                        77  80 ARTICS
I                                        77  80 ARTICL
I**********
I                                        81  81 DESIGI
I                                        82  82 DESIGT
I                                        83  83 DESIGE
I                                        84  84 DESIGO
I                                        81  84 DESIGS
I                                        81  84 DESIGL
```

```
I**********
I                                          85  85 QTYI
I                                          86  86 QTYT
I                                          87  87 QTYE
I                                          88  88 QTYO
I                                          85  88 QTYS
I                                          85  88 QTYL
I**********
I                                          89  89 UNITPI
I                                          90  90 UNITPT
I                                          91  91 UNITPE
I                                          92  92 UNITPO
I                                          89  92 UNITPS
I                                          89  92 UNITPL
I**********
I                                          93  93 PRICEI
I                                          94  94 PRICET
I                                          95  95 PRICEE
I                                          96  96 PRICEO
I                                          93  96 PRICES
I                                          93  96 PRICEL
I**********
I                                          97  97 $0001I
I                                          98  98 $0001T
I                                          99  99 $0001E
I                                         100 100 $0001O
I                                          97 100 $0001S
I                                          97 100 $0001L
I**********
I                                         101 101 $0002I
I                                         102 102 $0002T
I                                         103 103 $0002E
I                                         104 104 $0002O
I                                         101 104 $0002S
I                                         101 104 $0002L
I**********
I                                         105 105 $0003I
I                                         106 106 $0003T
I                                         107 107 $0003E
I                                         108 108 $0003O
I                                         105 108 $0003S
I                                         105 108 $0003L
I**********
I                                         109 109 $0004I
I                                         110 110 $0004T
I                                         111 111 $0004E
I                                         112 112 $0004O
I                                         109 112 $0004S
I                                         109 112 $0004L
I**********
I                                         113 113 $0005I
I                                         114 114 $0005T
I                                         115 115 $0005E
I                                         116 116 $0005O
I                                         113 116 $0005S
I                                         113 116 $0005L
```

```
I**********
I                                        117 117 $0006I
I                                        118 118 $0006T
I                                        119 119 $0006E
I                                        120 120 $0006O
I                                        117 120 $0006S
I                                        117 120 $0006L
I**********
I                                        121 121 $0007I
I                                        122 122 $0007T
I                                        123 123 $0007E
I                                        124 124 $0007O
I                                        121 124 $0007S
I                                        121 124 $0007L
I**********
I                                        125 125 $0008I
I                                        126 126 $0008T
I                                        127 127 $0008E
I                                        128 128 $0008O
I                                        125 128 $0008S
I                                        125 128 $0008L
I**********
I                                        129 129 $0009I
I                                        130 130 $0009T
I                                        131 131 $0009E
I                                        132 132 $0009O
I                                        129 132 $0009S
I                                        129 132 $0009L
I**********
I                                        133 133 $0010I
I                                        134 134 $0010T
I                                        135 135 $0010E
I                                        136 136 $0010O
I                                        133 136 $0010S
I                                        133 136 $0010L
I**********
I                                        137 137 $0011I
I                                        138 138 $0011T
I                                        139 139 $0011E
I                                        140 140 $0011O
I                                        137 140 $0011S
I                                        137 140 $0011L
I**********
I                                        141 141 $0012I
I                                        142 142 $0012T
I                                        143 143 $0012E
I                                        144 144 $0012O
I                                        141 144 $0012S
```

```
I                                     141 144 $0012L
I**********
I                                     145 145 $0013I
I                                     146 146 $0013T
I                                     147 147 $0013E
I                                     148 148 $00130
I                                     145 148 $0013S
I                                     145 148 $0013L
I**********
I                                     149 149 $0014I
I                                     150 150 $0014T
I                                     151 151 $0014E
I                                     152 152 $00140
I                                     149 152 $0014S
I                                     149 152 $0014L
I**********
I                                     153 153 $0015I
I                                     154 154 $0015T
I                                     155 155 $0015E
I                                     156 156 $00150
I                                     153 156 $0015S
I                                     153 156 $0015L
I**********
I                                     157 157 SUBTOI
I                                     158 158 SUBTOT
I                                     159 159 SUBTOE
I                                     160 160 SUBTOO
I                                     157 160 SUBTOS
I                                     157 160 SUBTOL
I**********
I                                     161 161 SALESI
I                                     162 162 SALEST
I                                     163 163 SALESE
I                                     164 164 SALESO
I                                     161 164 SALESS
I                                     161 164 SALESL
I**********
I                                     165 165 TOTALI
I                                     166 166 TOTALT
I                                     167 167 TOTALE
I                                     168 168 TOTALO
I                                     165 168 TOTALS
I                                     165 168 TOTALL
I**********
I**********
I                                     169 193 NAME
I                                     194 219 STREE
```

```
I                                        220 224 ZIPCO
I                                        225 248 CITY
I                                        170 248 ADDRE
I                                        249 260 CUSTM
I                                        261 274 DATE
I                                        275 279 ARTIC
I                                        280 307 DESIG
I                                        308 3120QTY
I                                      R 313 3182UNITP
I                                      R 319 3292PRICE
I                                        330 334 $0001
I                                        335 362 $0002
I                                        363 3670$0003
I                                      R 368 3732$0004
I                                      R 374 3842$0005
I                                        385 389 $0006
I                                        390 417 $0007
I                                        418 4220$0008
I                                      R 423 4282$0009
I                                      R 429 4392$0010
I                                        440 444 $0011
I                                        445 472 $0012
I                                        473 4770$0013
I                                      R 478 4832$0014
I                                      R 484 4942$0015
I                                      R 495 5062SUBTO
I                                      R 507 5172SALES
I                                      R 518 5292TOTAL
I**********
I**********
I**********                               1  52 DELIGP
I**********                              53 168 DELIAP
I**********                             169 529 DELIFP
```

Output formatting

```
O*  FORMAT NAME   : DELIVER
O*  USER AREA LEN : 0529
O                         DELIRM     4B
O*                                       MAIN RETURNCODE
O                         DELIRC     6B
O*                                       ERROR CATEGORY
O                         DELIRR     8B
O*                                       ERROR REASON
O*
O*                                     FHS RETURNCODE
O                         DELIFM     9
O*                                       FIELDS MOD
```

```
0                       DELIFD    10
0*                                    FIELDS DETECTED
0                       DELIFV    11
0*                                    FIELDS VALID
0                       DELIUE    12
0*                                    USER EXIT RC
0                       DELIFU    13
0*                                    FIELDS UNDEFINED
0*
0*                                  FORMATTING INDICATOR
0                       DELIIC    14
0*                                    INPUT KEY CLASS
0                       DELIIN    16B
0*                                    INPUT KEY NUMBER
0                       DELIR2    20
0*                                    RESERVED02
0*
0*                                  INPUT IDENTIFICATION
0                       DELICI    21
0*                                    INIT CONTROL
0                       DELIIO    22
0*                                    INIT OPTION
0                       DELITC    23
0*                                    TABULATOR CONTROL
0                       DELIFL    24
0*                                    FUNCTION LOCK
0                       DELIVC    25
0*                                    LINE SPACE
0                       DELIHC    26
0*                                    CHARACTER SPACE
0                       DELIR3    28
0*                                    RESERVED03
0*
0*                                  DEVICE CONTROLS
0                       DELICC    29
0*                                    CYCLE CONTROL
0                       DELICL    30
0*                                    COPY CONTROL
0                       DELIAC    31
0*                                    ALARM CONTROL
0                       DELIBC    32
0*                                    HOLE COLOR
0*
0*                                  OUTPUT CONTROLS
0                       DELIDS    33
0*                                    DISPLAY SELECTION
0                       DELILS    34
```

```
O*                                        LEVEL SELECTION
O                          DELIOM    35
O*                                        OUTPUT MODE
O                          DELICT    36
O*                                        CURSOR CONTROL
O                          DELICP    40B
O*                                        CURSOR POSITION
O                          DELIUC    41
O*                                        USER EXIT CONTROL
O                          DELILE    42
O*                                        LANGUAGE EXTENSION
O                          DELISL    44B
O*                                        STARTLINE
O*
O*                                    FORMATTING CONTROLS
O                          DELIKS    52
O*                                        P KEY SET
O*
O*
O**********
O**********       ************************************
O**********       *      FIELD  ATTRIBUTE  TABLE      *
O**********       ************************************
O**********       *  SUFFIX             MEANING       *
O**********       ************************************
O**********       *    F    ---->    FLD LENGTH        *
O**********       *    I    ---->      INPUT STATE     *
O**********       *    T    ---->      INPUT STATE ACT *
O**********       *    E    ---->      EDIT STATE      *
O**********       *    O    ---->      OUTPUT CONTROL  *
O**********       *    S    ---->    BASIC ATTRIBUTES  *
O**********       *    M    ---->    ATTR COMB         *
O**********       *    J    ---->      INPUT CONTROL   *
O**********       *    P    ---->      PROTECTION      *
O**********       *    N    ---->    FIELD INPUT       *
O**********       *    Y    ---->      INTENSITY       *
O**********       *    V    ---->      VISIBILITY      *
O**********       *    U    ---->      UNDERLINE       *
O**********       *    W    ---->      INVERSE         *
O**********       *    Q    ---->    DISPLAY CONTROL   *
O**********       *    C    ---->    COLOR             *
O**********       *    X    ---->    INITIAL CURSOR    *
O**********       *    R    ---->    EDIT RC           *
O**********       *    Z    ---->    ALIGNMENT BYTE    *
O**********       *    L    ---->    FLD ATTR TABLE LEN*
O**********       ************************************
O**********
O                          NAMEI     53
```

```
O                       NAMET     54
O                       NAMEE     55
O                       NAMEO     56
O*                      NAMES     56
O*                      NAMEL     56
O**********
O                       STREEI    57
O                       STREET    58
O                       STREEE    59
O                       STREEO    60
O*                      STREES    60
O*                      STREEL    60
O**********
O                       ZIPCOI    61
O                       ZIPCOT    62
O                       ZIPCOE    63
O                       ZIPCOO    64
O*                      ZIPCOS    64
O*                      ZIPCOL    64
O**********
O                       CITYI     65
O                       CITYT     66
O                       CITYE     67
O                       CITYO     68
O*                      CITYS     68
O*                      CITYL     68
O**********
O                       CUSTMI    69
O                       CUSTMT    70
O                       CUSTME    71
O                       CUSTMO    72
O*                      CUSTMS    72
O*                      CUSTML    72
O**********
O                       DATEI     73
O                       DATET     74
O                       DATEE     75
O                       DATEO     76
O*                      DATES     76
O*                      DATEL     76
O**********
O                       ARTICI    77
O                       ARTICT    78
O                       ARTICE    79
O                       ARTICO    80
O*                      ARTICS    80
O*                      ARTICL    80
```

```
O**********
O                       DESIGI    81
O                       DESIGT    82
O                       DESIGE    83
O                       DESIGO    84
O*                      DESIGS    84
O*                      DESIGL    84
O**********
O                       QTYI      85
O                       QTYT      86
O                       QTYE      87
O                       QTYO      88
O*                      QTYS      88
O*                      QTYL      88
O**********
O                       UNITPI    89
O                       UNITPT    90
O                       UNITPE    91
O                       UNITPO    92
O*                      UNITPS    92
O*                      UNITPL    92
O**********
O                       PRICEI    93
O                       PRICET    94
O                       PRICEE    95
O                       PRICEO    96
O*                      PRICES    96
O*                      PRICEL    96
O**********
O                       $0001I    97
O                       $0001T    98
O                       $0001E    99
O                       $0001O   100
O*                      $0001S   100
O*                      $0001L   100
O**********
O                       $0002I   101
O                       $0002T   102
O                       $0002E   103
O                       $0002O   104
O*                      $0002S   104
O*                      $0002L   104
O**********
O                       $0003I   105
O                       $0003T   106
O                       $0003E   107
O                       $0003O   108
O*                      $0003S   108
```

```
0*                          $0003L   108
0**********
0                           $0004I   109
0                           $0004T   110
0                           $0004E   111
0                           $0004O   112
0*                          $0004S   112
0*                          $0004L   112
0**********
0                           $0005I   113
0                           $0005T   114
0                           $0005E   115
0                           $0005O   116
0*                          $0005S   116
0*                          $0005L   116
0**********
0                           $0006I   117
0                           $0006T   118
0                           $0006E   119
0                           $0006O   120
0*                          $0006S   120
0*                          $0006L   120
0**********
0                           $0007I   121
0                           $0007T   122
0                           $0007E   123
0                           $0007O   124
0*                          $0007S   124
0*                          $0007L   124
0**********
0                           $0008I   125
0                           $0008T   126
0                           $0008E   127
0                           $0008O   128
0*                          $0008S   128
0*                          $0008L   128
0**********
0                           $0009I   129
0                           $0009T   130
0                           $0009E   131
0                           $0009O   132
0*                          $0009S   132
0*                          $0009L   132
0**********
0                           $0010I   133
0                           $0010T   134
0                           $0010E   135
```

```
0                         $0010O   136
0*                        $0010S   136
0*                        $0010L   136
0**********
0                         $0011I   137
0                         $0011T   138
0                         $0011E   139
0                         $0011O   140
0*                        $0011S   140
0*                        $0011L   140
0**********
0                         $0012I   141
0                         $0012T   142
0                         $0012E   143
0                         $0012O   144
0*                        $0012S   144
0*                        $0012L   144
0**********
0                         $0013I   145
0                         $0013T   146
0                         $0013E   147
0                         $0013O   148
0*                        $0013S   148
0*                        $0013L   148
0**********
0                         $0014I   149
0                         $0014T   150
0                         $0014E   151
0                         $0014O   152
0*                        $0014S   152
0*                        $0014L   152
0**********
0                         $0015I   153
0                         $0015T   154
0                         $0015E   155
0                         $0015O   156
0*                        $0015S   156
0*                        $0015L   156
0**********
0                         SUBTOI   157
0                         SUBTOT   158
0                         SUBTOE   159
0                         SUBTOO   160
0*                        SUBTOS   160
0*                        SUBTOL   160
0**********
0                         SALESI   161
```

```
O                       SALEST   162
O                       SALESE   163
O                       SALESO   164
O*                      SALESS   164
O*                      SALESL   164
O**********
O                       TOTALI   165
O                       TOTALT   166
O                       TOTALE   167
O                       TOTALO   168
O*                      TOTALS   168
O*                      TOTALL   168
O**********
O**********
O*+                     NAME     193
O*+                     STREE    219
O*+                     ZIPCO    224
O*+                     CITY     248
O*-                     ADDRE    248
O*+                     CUSTM    260
O*+                     DATE     274
O*+                     ARTIC    279
O*+                     DESIG    307
O*+                     QTY      312
O*+                     UNITP    318R
O*+                     PRICE    329R
O*+                     $0001    334

O*+                     $0002    362
O*+                     $0003    367
O*+                     $0004    373R
O*+                     $0005    384R
O*+                     $0006    389
O*+                     $0007    417
O*+                     $0008    422
O*+                     $0009    428R
O*+                     $0010    439R
O*+                     $0011    444
O*+                     $0012    472
O*+                     $0013    477
O*+                     $0014    483R
O*+                     $0015    494R
O*+                     SUBTO    506R
O*+                     SALES    517R
O*+                     TOTAL    529R
O**********
O**********
```

```
O**********              DELIGP    42
O**********              DELIAP   168
O**********              DELIFP   529
```

17.1.7 DRIVE

Data transfer area with separate attribute blocks and field contents

```
/* FORMAT NAME: DELIVER */
/*************************************************************/
/*                  GLOBAL ATTRIBUTE BLOCK                   */
/*************************************************************/
          19 DELIVER_GLOBALS,
            /* FORM_RETURNCODE */
            20 RC_MAIN                    INTEGER,
            20 RC_CATEGORY                SMALLINT,
            20 RC_REASON                  SMALLINT,
            /* FORM_INDICATORS */
            20 FIELDS_MOD                 CHAR(1),
            20 FIELDS_DET                 CHAR(1),
            20 FIELDS_VALID               CHAR(1),
            20 USER_EXIT_RC               CHAR(1),
            20 FIELDS_UNDEFINED           CHAR(1),
            /* INPUT_IDENTIFICATION */
            20 INPUT_KEY_CLASS            CHAR(1),
            20 INPUT_KEY_NUMBER           SMALLINT,
            20 RESERVED2                  CHAR(4),
            /* DEVICE_CONTROLS */
            20 INIT_CTL                   CHAR(1),
            20 INIT_OPT                   CHAR(1),
            20 TAB_CTL                    CHAR(1),
            20 FCT_LOCK                   CHAR(1),
            20 VMI_CTL                    CHAR(1),
            20 HMI_
            20 RESERVED3                  CHAR(2),
            /* OUTPUT_CONTROLS */
            20 CYCLE_CTL                  CHAR(1),
            20 COPY_CTL                   CHAR(1),
            20 ALARM_CTL                  CHAR(1),
            20 HOLE_COLOR                 CHAR(1),
            /* FORM_CONTROLS */
            20 DISPLAY_SEL                CHAR(1),
            20 LEVEL_SEL                  CHAR(1),
            20 OUTPUT_MOD                 CHAR(1),
            20 CURSOR_CTL                 CHAR(1),
            20 CURSOR_POS                 INTEGER,
            20 USER_EXIT_CTL              CHAR(1),
            20 LANGUAGE_EXT               CHAR(1),
            20 STARTLINE                  SMALLINT,
            /*              */
            20 P_KEY_SET                  CHAR(8),
```

```
/*************************************************************/
/*                  FIELD ATTRIBUTE BLOCKS                   */
/*************************************************************/
          19 DELIVER_ATTR,
            20 ADDRESS_FAB,
              21 NAME_FAB,
                22 BASIC_ATTR,
                  23 INPUT_STATE          CHAR(1),
                  23 INPUT_STATE_ACT      CHAR(1),
                  23 EDIT_STATE           CHAR(1),
                  23 OUTPUT_CTL           CHAR(1),
              21 STREET_FAB
                 LIKE &DELIVER_ATTR.NAME_FAB,
              21 ZIPCODE_FAB
                 LIKE &DELIVER_ATTR.NAME_FAB,
              21 CITY_FAB
                 LIKE &DELIVER_ATTR.NAME_FAB,
            20 CUSTOMERNO_FAB
               LIKE &DELIVER_ATTR.NAME_FAB,
            20 DATE_FAB
               LIKE &DELIVER_ATTR.NAME FAB,
            20 ARTLINE_FAB          (04),
              21 ARTICLENUMBER_FAB
                 LIKE &DELIVER_ATTR.NAME_FAB,
              21 DESIGNATION_FAB
                 LIKE &DELIVER_ATTR.NAME_FAB,
              21 QTY_FAB
                 LIKE &DELIVER_ATTR.NAME_FAB,
              21 UNITPRICE_FAB
                 LIKE &DELIVER_ATTR.NAME_FAB,
              21 PRICE_FAB
                 LIKE &DELIVER_ATTR.NAME_FAB,
            20 SUBTOTAL_FAB
               LIKE &DELIVER_ATTR.NAME_FAB,
            20 SALESTAX_FAB
               LIKE &DELIVER_ATTR.NAME_FAB,
            20 TOTAL_FAB
               LIKE &DELIVER_ATTR.NAME_FAB,
          19 DELIVER_ATTR_TAB   (0029)  REDEFINES  DELIVER_ATTR
             LIKE &DELIVER_ATTR.NAME_FAB,
/*************************************************************/
/*                     FIELD DATA PART                       */
/*************************************************************/
          19 DELIVER_DATA,
            20 ADDRESS,
              21 NAME                               CHAR(025),
              21 STREET                             CHAR(026),
              21 ZIPCODE                            CHAR(005),
```

```
    21 CITY                              CHAR(024),
  20 CUSTOMERNO                         CHAR(012),
  20 DATE                               DATE,
  20 DATE_JUL                           CHAR(004),
  20 ARTLINE              (04),
    21 ARTICLENUMBER                     CHAR(005),
    21 DESIGNATION                       CHAR(028),
    21 QTY                               NUMERIC(05 , 00)
       CHECK &QTY                        >= 0,
    21 UNITPRICE                         NUMERIC(05 , 02),
    21 FILLER                            CHAR(001),
    21 PRICE                             NUMERIC(10 , 02),
    21 FILLER                            CHAR(001),
  20 SUBTOTAL                           NUMERIC(11 , 02),
  20 FILLER                             CHAR(001),
  20 SALESTAX                           NUMERIC(10 , 02),
  20 FILLER                             CHAR(001),
  20 TOTAL                              NUMERIC(11 , 02),
  20 FILLER                             CHAR(001),
/**/
&AREA LENGTH DELIVER  PERMANENT SMALLINT INIT 00529;
```

17.1.8 FORTRAN

Data transfer area with separate attribute blocks and field contents

```
* FORMAT NAME: DELIVER    USER AREA LENGTH: 00529
*
****************************************************************
*                    GLOBAL ATTRIBUTE BLOCK                    *
****************************************************************
*
      CHARACTER  * 52  DELIVERGLOBALS
*
*
*                                          FORM RETURNCODE
      INTEGER    *  4  DELIVERRCMAIN
*                                            RC MAIN
      INTEGER    *  2  DELIVERRCCATEGO
*                                            RC CATEGORY
      INTEGER    *  2  DELIVERRCREASON
*                                            RC REASON
*
*                                          FORM INDICATORS
      CHARACTER  *  1  DELIVERFLDMOD
*                                            FIELDS MOD
      CHARACTER  *  1  DELIVERFLDDET
*                                            FIELDS DET
      CHARACTER  *  1  DELIVERFLDVALID
*                                            FIELDS VALID
      CHARACTER  *  1  DELIVERUSEREXRC
*                                            USER EXIT RC
      CHARACTER  *  1  DELIVERFLDUNDEF
*                                            FIELDS UNDEFINED
*
*                                          INPUT IDENTIFICATION
      CHARACTER  *  1  DELIVERINPCLASS
*                                            INPUT KEY CLASS
      INTEGER    *  2  DELIVERINPNUMB
*                                            INPUT KEY NUMBER
      CHARACTER  *  4  DELIVERRESERV2
*                                            RESERVED2
*
*                                          DEVICE CONTROLS
      CHARACTER  *  1  DELIVERINITCTL
*                                            INIT CTL
      CHARACTER  *  1  DELIVERINITOPT
*                                            INIT OPT
      CHARACTER  *  1  DELIVERTABCTL
```

```
*                                                    TAB CTL
      CHARACTER  *  1  DELIVERFCTLOCK
*                                                    FCT LOCK
      CHARACTER  *  1  DELIVERVMICTL
*                                                    VMI CTL
      CHARACTER  *  1  DELIVERHMICTL
*                                                    HMI CTL
      CHARACTER  *  2  DELIVERRESERV3
*                                                    RESERVED3
*
*                                                  OUTPUT CONTROLS
      CHARACTER  *  1  DELIVERCYCLCTL
*                                                    CYCLE CTL
      CHARACTER  *  1  DELIVERCOPYCTL
*                                                    COPY CTL
      CHARACTER  *  1  DELIVERALARMCTL
*                                                    ALARM CTL
      CHARACTER  *  1  DELIVERHOLECOL
*                                                    HOLE COLOR
*
*                                                  FORM CONTROLS
      CHARACTER  *  1  DELIVERDISPLSEL
*                                                    DISPLAY SEL
      CHARACTER  *  1  DELIVERLEVELSEL
*                                                    LEVEL SEL
      CHARACTER  *  1  DELIVEROUTMODE
*                                                    OUTPUT MODE
      CHARACTER  *  1  DELIVERCURSCTL
*                                                    CURSOR CTL
      INTEGER    *  4  DELIVERCURSPOS
*                                                    CURSOR POS
      CHARACTER  *  1  DELIVERUSEXCTL
*                                                    USER EXIT CTL
      CHARACTER  *  1  DELIVERLANGEXT
*                                                    LANGUAGE EXT
      INTEGER    *  2  DELIVERSTARTL
*                                                    STARTLINE
*
*
      CHARACTER  *  8  DELIVERPKEYSET
*                                                    P KEY SET
*
      EQUIVALENCE (DELIVERGLOBALS ( 1: 4),  DELIVERRCMAIN)
      EQUIVALENCE (DELIVERGLOBALS ( 5: 6),  DELIVERRCCATEGO)
      EQUIVALENCE (DELIVERGLOBALS ( 7: 8),  DELIVERRCREASON)
      EQUIVALENCE (DELIVERGLOBALS ( 9: 9),  DELIVERFLDMOD)
      EQUIVALENCE (DELIVERGLOBALS (10:10),  DELIVERFLDDET)
```

```
      EQUIVALENCE (DELIVERGLOBALS (11:11),  DELIVERFLDVALID)
      EQUIVALENCE (DELIVERGLOBALS (12:12),  DELIVERUSEREXRC)
      EQUIVALENCE (DELIVERGLOBALS (13:13),  DELIVERFLDUNDEF)
      EQUIVALENCE (DELIVERGLOBALS (14:14),  DELIVERINPCLASS)
      EQUIVALENCE (DELIVERGLOBALS (15:16),  DELIVERINPNUMB)
      EQUIVALENCE (DELIVERGLOBALS (17:20),  DELIVERRESERV2)
      EQUIVALENCE (DELIVERGLOBALS (21:21),  DELIVERINITCTL)
      EQUIVALENCE (DELIVERGLOBALS (22:22),  DELIVERINITOPT)
      EQUIVALENCE (DELIVERGLOBALS (23:23),  DELIVERTABCTL)
      EQUIVALENCE (DELIVERGLOBALS (24:24),  DELIVERFCTLOCK)
      EQUIVALENCE (DELIVERGLOBALS (25:25),  DELIVERVMICTL)
      EQUIVALENCE (DELIVERGLOBALS (26:26),  DELIVERHMICTL)
      EQUIVALENCE (DELIVERGLOBALS (27:28),  DELIVERRESERV3)
      EQUIVALENCE (DELIVERGLOBALS (29:29),  DELIVERCYCLCTL)
      EQUIVALENCE (DELIVERGLOBALS (30:30),  DELIVERCOPYCTL)
      EQUIVALENCE (DELIVERGLOBALS (31:31),  DELIVERALARMCTL)
      EQUIVALENCE (DELIVERGLOBALS (32:32),  DELIVERHOLECOL)
      EQUIVALENCE (DELIVERGLOBALS (33:33),  DELIVERDISPLSEL)
      EQUIVALENCE (DELIVERGLOBALS (34:34),  DELIVERLEVELSEL)
      EQUIVALENCE (DELIVERGLOBALS (35:35),  DELIVEROUTMODE)
      EQUIVALENCE (DELIVERGLOBALS (36:36),  DELIVERCURSCTL)
      EQUIVALENCE (DELIVERGLOBALS (37:40),  DELIVERCURSPOS)
      EQUIVALENCE (DELIVERGLOBALS (41:41),  DELIVERUSEXCTL)
      EQUIVALENCE (DELIVERGLOBALS (42:42),  DELIVERLANGEXT)
      EQUIVALENCE (DELIVERGLOBALS (43:44),  DELIVERSTARTL)
      EQUIVALENCE (DELIVERGLOBALS (45:52),  DELIVERPKEYSET)
*
*****************************************************************
*                    FIELD ATTRIBUTE BLOCKS                     *
*****************************************************************
*
      CHARACTER  *  116  DELIVERATTR
*
*
      CHARACTER  *    4  NAMEFAB
*                                          FLD ATTRIBUTE BLOCK
      CHARACTER  *    4  NAMEBAT
*                                          BASIC ATTR
      CHARACTER  *    1  NAMEIST
*                                            INPUT STATE
      CHARACTER  *    1  NAMEISA
*                                            INPUT STATE ACT
      CHARACTER  *    1  NAMEEST
*                                            EDIT STATE
      CHARACTER  *    1  NAMEOCT
*                                            OUTPUT CTL
*
```

```
      EQUIVALENCE (DELIVERATTR  (    1:    4), NAMEFAB)
      EQUIVALENCE (NAMEFAB            ( 1: 4), NAMEBAT)
      EQUIVALENCE (NAMEBAT              (1:1), NAMEIST)
      EQUIVALENCE (NAMEBAT              (2:2), NAMEISA)
      EQUIVALENCE (NAMEBAT              (3:3), NAMEEST)
      EQUIVALENCE (NAMEBAT              (4:4), NAMEOCT)
*
*
      CHARACTER  *    4  STREETFAB
      CHARACTER  *    4  STREETBAT
      CHARACTER  *    1  STREETIST
      CHARACTER  *    1  STREETISA
      CHARACTER  *    1  STREETEST
      CHARACTER  *    1  STREETOCT
*
      EQUIVALENCE (DELIVERATTR  (    5:    8), STREETFAB)
      EQUIVALENCE (STREETFAB          ( 1: 4), STREETBAT)
      EQUIVALENCE (STREETBAT            (1:1), STREETIST)
      EQUIVALENCE (STREETBAT            (2:2), STREETISA)
      EQUIVALENCE (STREETBAT            (3:3), STREETEST)
      EQUIVALENCE (STREETBAT            (4:4), STREETOCT)
*
*
      CHARACTER  *    5  ZIPCODEFAB
      CHARACTER  *    5  ZIPCODEBAT
      CHARACTER  *    1  ZIPCODEIST
      CHARACTER  *    1  ZIPCODEISA
      CHARACTER  *    1  ZIPCODEEST
      CHARACTER  *    1  ZIPCODEOCT

*
      EQUIVALENCE (DELIVERATTR  (    9:   12), ZIPCODEFAB)
      EQUIVALENCE (ZIPCODEFAB         ( 1: 5), ZIPCODEBAT)
      EQUIVALENCE (ZIPCODEBAT           (1:1), ZIPCODEIST)
      EQUIVALENCE (ZIPCODEBAT           (2:2), ZIPCODEISA)
      EQUIVALENCE (ZIPCODEBAT           (3:3), ZIPCODEEST)
      EQUIVALENCE (ZIPCODEBAT           (5:5), ZIPCODEOCT)
*
*
      CHARACTER  *    4  CITYFAB
      CHARACTER  *    4  CITYBAT
      CHARACTER  *    1  CITYIST
      CHARACTER  *    1  CITYISA
      CHARACTER  *    1  CITYEST
      CHARACTER  *    1  CITYOCT
*
      EQUIVALENCE (DELIVERATTR  (   13:   16), CITYFAB)
      EQUIVALENCE (CITYFAB            ( 1: 4), CITYBAT)
```

```
      EQUIVALENCE (CITYBAT              (1:1), CITYIST)
      EQUIVALENCE (CITYBAT              (2:2), CITYISA)
      EQUIVALENCE (CITYBAT              (3:3), CITYEST)
      EQUIVALENCE (CITYBAT              (4:4), CITYOCT)
*
*
      CHARACTER  *    4  CUSTOMERNOFAB
      CHARACTER  *    4  CUSTOMERNOBAT
      CHARACTER  *    1  CUSTOMERNOIST
      CHARACTER  *    1  CUSTOMERNOISA
      CHARACTER  *    1  CUSTOMERNOEST
      CHARACTER  *    1  CUSTOMERNOOCT
*
      EQUIVALENCE (DELIVERATTR  (   17:   20), CUSTOMERNOFAB)
      EQUIVALENCE (CUSTOMERNOFAB       ( 1: 4), CUSTOMERNOBAT)
      EQUIVALENCE (CUSTOMERNOBAT         (1:1), CUSTOMERNOIST)
      EQUIVALENCE (CUSTOMERNOBAT         (2:2), CUSTOMERNOISA)
      EQUIVALENCE (CUSTOMERNOBAT         (3:3), CUSTOMERNOEST)
      EQUIVALENCE (CUSTOMERNOBAT         (4:4), CUSTOMERNOOCT)
*
*
      CHARACTER  *    4  DATEFAB
      CHARACTER  *    4  DATEBAT
      CHARACTER  *    1  DATEIST
      CHARACTER  *    1  DATEISA
      CHARACTER  *    1  DATEEST
      CHARACTER  *    1  DATEOCT
*
      EQUIVALENCE (DELIVERATTR  (   21:   24), DATEFAB)
      EQUIVALENCE (DATEFAB             ( 1: 4), DATEBAT)
      EQUIVALENCE (DATEBAT               (1:1), DATEIST)
      EQUIVALENCE (DATEBAT               (2:2), DATEISA)
      EQUIVALENCE (DATEBAT               (3:3), DATEEST)
      EQUIVALENCE (DATEBAT               (4:4), DATEOCT)
*
*
      CHARACTER  *    4  ARTICLENUMBEFAB
      CHARACTER  *    4  ARTICLENUMBEBAT
      CHARACTER  *    1  ARTICLENUMBEIST
      CHARACTER  *    1  ARTICLENUMBEISA
      CHARACTER  *    1  ARTICLENUMBEEST
      CHARACTER  *    1  ARTICLENUMBEOCT
*
      EQUIVALENCE (DELIVERATTR  (   25:   28), ARTICLENUMBEFAB)
      EQUIVALENCE (ARTICLENUMBEFAB     ( 1: 4), ARTICLENUMBEBAT)
      EQUIVALENCE (ARTICLENUMBEBAT       (1:1), ARTICLENUMBEIST)
      EQUIVALENCE (ARTICLENUMBEBAT       (2:2), ARTICLENUMBEISA)
```

```
      EQUIVALENCE (ARTICLENUMBEBAT       (3:3), ARTICLENUMBEEST)
      EQUIVALENCE (ARTICLENUMBEBAT       (4:4), ARTICLENUMBEOCT)
*
*
      CHARACTER  *    4  DESIGNATIONFAB
      CHARACTER  *    4  DESIGNATIONBAT
      CHARACTER  *    1  DESIGNATIONIST
      CHARACTER  *    1  DESIGNATIONISA
      CHARACTER  *    1  DESIGNATIONEST
      CHARACTER  *    1  DESIGNATIONOCT
*
      EQUIVALENCE (DELIVERATTR  (   29:   32), DESIGNATIONFAB)
      EQUIVALENCE (DESIGNATIONFAB      ( 1: 4), DESIGNATIONBAT)
      EQUIVALENCE (DESIGNATIONBAT        (1:1), DESIGNATIONIST)
      EQUIVALENCE (DESIGNATIONBAT        (2:2), DESIGNATIONISA)
      EQUIVALENCE (DESIGNATIONBAT        (3:3), DESIGNATIONEST)
      EQUIVALENCE (DESIGNATIONBAT        (4:4), DESIGNATIONOCT)
*
*
      CHARACTER  *    4  QTYFAB
      CHARACTER  *    4  QTYBAT
      CHARACTER  *    1  QTYIST
      CHARACTER  *    1  QTYISA
      CHARACTER  *    1  QTYEST
      CHARACTER  *    1  QTYOCT
*
      EQUIVALENCE (DELIVERATTR  (   33:   36), QTYFAB)
      EQUIVALENCE (QTYFAB              ( 1: 4), QTYBAT)
      EQUIVALENCE (QTYBAT                (1:1), QTYIST)
      EQUIVALENCE (QTYBAT                (2:2), QTYISA)
      EQUIVALENCE (QTYBAT                (3:3), QTYEST)
      EQUIVALENCE (QTYBAT                (4:4), QTYOCT)
*
*
      CHARACTER  *    4  UNITPRICEFAB
      CHARACTER  *    4  UNITPRICEBAT
      CHARACTER  *    1  UNITPRICEIST
      CHARACTER  *    1  UNITPRICEISA
      CHARACTER  *    1  UNITPRICEEST
      CHARACTER  *    1  UNITPRICEOCT

*
      EQUIVALENCE (DELIVERATTR  (   37:   40), UNITPRICEFAB)
      EQUIVALENCE (UNITPRICEFAB        ( 1: 4), UNITPRICEBAT)
      EQUIVALENCE (UNITPRICEBAT          (1:1), UNITPRICEIST)
      EQUIVALENCE (UNITPRICEBAT          (2:2), UNITPRICEISA)
      EQUIVALENCE (UNITPRICEBAT          (3:3), UNITPRICEEST)
      EQUIVALENCE (UNITPRICEBAT          (4:4), UNITPRICEOCT)
```

```
*
*
      CHARACTER  *    4  PRICEFAB
      CHARACTER  *    4  PRICEBAT
      CHARACTER  *    1  PRICEIST
      CHARACTER  *    1  PRICEISA
      CHARACTER  *    1  PRICEEST
      CHARACTER  *    1  PRICEOCT
*
      EQUIVALENCE (DELIVERATTR  (   41:   44), PRICEFAB)
      EQUIVALENCE (PRICEFAB           ( 1: 4), PRICEBAT)
      EQUIVALENCE (PRICEBAT             (1:1), PRICEIST)
      EQUIVALENCE (PRICEBAT             (2:2), PRICEISA)
      EQUIVALENCE (PRICEBAT             (3:3), PRICEEST)
      EQUIVALENCE (PRICEBAT             (4:4), PRICEOCT)
*
*
      CHARACTER  *    4  $DELIVER0001FAB
      CHARACTER  *    4  $DELIVER0001BAT
      CHARACTER  *    1  $DELIVER0001IST
      CHARACTER  *    1  $DELIVER0001ISA
      CHARACTER  *    1  $DELIVER0001EST
      CHARACTER  *    1  $DELIVER0001OCT
*
      EQUIVALENCE (DELIVERATTR  (   45:   48), $DELIVER0001FAB)
      EQUIVALENCE ($DELIVER0001FAB    ( 1: 4), $DELIVER0001BAT)
      EQUIVALENCE ($DELIVER0001BAT      (1:1), $DELIVER0001IST)
      EQUIVALENCE ($DELIVER0001BAT      (2:2), $DELIVER0001ISA)
      EQUIVALENCE ($DELIVER0001BAT      (3:3), $DELIVER0001EST)
      EQUIVALENCE ($DELIVER0001BAT      (4:4), $DELIVER0001OCT)
*
*
      CHARACTER  *    4  $DELIVER0002FAB
      CHARACTER  *    4  $DELIVER0002BAT
      CHARACTER  *    1  $DELIVER0002IST
      CHARACTER  *    1  $DELIVER0002ISA
      CHARACTER  *    1  $DELIVER0002EST
      CHARACTER  *    1  $DELIVER0002OCT
*
      EQUIVALENCE (DELIVERATTR  (   49:   52), $DELIVER0002FAB)
      EQUIVALENCE ($DELIVER0002FAB    ( 1: 4), $DELIVER0002BAT)
      EQUIVALENCE ($DELIVER0002BAT      (1:1), $DELIVER0002IST)
      EQUIVALENCE ($DELIVER0002BAT      (2:2), $DELIVER0002ISA)
      EQUIVALENCE ($DELIVER0002BAT      (3:3), $DELIVER0002EST)
      EQUIVALENCE ($DELIVER0002BAT      (4:4), $DELIVER0002OCT)
```

```
*
*
      CHARACTER  *    4  $DELIVER0003FAB
      CHARACTER  *    4  $DELIVER0003BAT
      CHARACTER  *    1  $DELIVER0003IST
      CHARACTER  *    1  $DELIVER0003ISA
      CHARACTER  *    1  $DELIVER0003EST
      CHARACTER  *    1  $DELIVER0003OCT
*
      EQUIVALENCE (DELIVERATTR  (   53:   56), $DELIVER0003FAB)
      EQUIVALENCE ($DELIVER0003FAB     ( 1: 4), $DELIVER0003BAT)
      EQUIVALENCE ($DELIVER0003BAT       (1:1), $DELIVER0003IST)
      EQUIVALENCE ($DELIVER0003BAT       (2:2), $DELIVER0003ISA)
      EQUIVALENCE ($DELIVER0003BAT       (3:3), $DELIVER0003EST)
      EQUIVALENCE ($DELIVER0003BAT       (4:4), $DELIVER0003OCT)
*
*
      CHARACTER  *    4  $DELIVER0004FAB
      CHARACTER  *    4  $DELIVER0004BAT
      CHARACTER  *    1  $DELIVER0004IST
      CHARACTER  *    1  $DELIVER0004ISA
      CHARACTER  *    1  $DELIVER0004EST
      CHARACTER  *    1  $DELIVER0004OCT
*
      EQUIVALENCE (DELIVERATTR  (   57:   60), $DELIVER0004FAB)
      EQUIVALENCE ($DELIVER0004FAB     ( 1: 4), $DELIVER0004BAT)
      EQUIVALENCE ($DELIVER0004BAT       (1:1), $DELIVER0004IST)
      EQUIVALENCE ($DELIVER0004BAT       (2:2), $DELIVER0004ISA)
      EQUIVALENCE ($DELIVER0004BAT       (3:3), $DELIVER0004EST)
      EQUIVALENCE ($DELIVER0004BAT       (4:4), $DELIVER0004OCT)
*
*
      CHARACTER  *    4  $DELIVER0005FAB
      CHARACTER  *    4  $DELIVER0005BAT
      CHARACTER  *    1  $DELIVER0005IST
      CHARACTER  *    1  $DELIVER0005ISA
      CHARACTER  *    1  $DELIVER0005EST
      CHARACTER  *    1  $DELIVER0005OCT
*
      EQUIVALENCE (DELIVERATTR  (   61:   64), $DELIVER0005FAB)
      EQUIVALENCE ($DELIVER0005FAB     ( 1: 4), $DELIVER0005BAT)
      EQUIVALENCE ($DELIVER0005BAT       (1:1), $DELIVER0005IST)
      EQUIVALENCE ($DELIVER0005BAT       (2:2), $DELIVER0005ISA)
      EQUIVALENCE ($DELIVER0005BAT       (3:3), $DELIVER0005EST)
      EQUIVALENCE ($DELIVER0005BAT       (4:4), $DELIVER0005OCT)
*
*
      CHARACTER  *    4  $DELIVER0006FAB
```

```
      CHARACTER  *    4  $DELIVER0006BAT
      CHARACTER  *    1  $DELIVER0006IST
      CHARACTER  *    1  $DELIVER0006ISA
      CHARACTER  *    1  $DELIVER0006EST
      CHARACTER  *    1  $DELIVER0006OCT
*
      EQUIVALENCE (DELIVERATTR  (   65:   68), $DELIVER0006FAB)
      EQUIVALENCE ($DELIVER0006FAB     ( 1: 4), $DELIVER0006BAT)
      EQUIVALENCE ($DELIVER0006BAT       (1:1), $DELIVER0006IST)
      EQUIVALENCE ($DELIVER0006BAT       (2:2), $DELIVER0006ISA)
      EQUIVALENCE ($DELIVER0006BAT       (3:3), $DELIVER0006EST)
      EQUIVALENCE ($DELIVER0006BAT       (4:4), $DELIVER0006OCT)
*
*
      CHARACTER  *    4  $DELIVER0007FAB
      CHARACTER  *    4  $DELIVER0007BAT
      CHARACTER  *    1  $DELIVER0007IST
      CHARACTER  *    1  $DELIVER0007ISA
      CHARACTER  *    1  $DELIVER0007EST
      CHARACTER  *    1  $DELIVER0007OCT
*
      EQUIVALENCE (DELIVERATTR  (   69:   72), $DELIVER0007FAB)
      EQUIVALENCE ($DELIVER0007FAB     ( 1: 4), $DELIVER0007BAT)
      EQUIVALENCE ($DELIVER0007BAT       (1:1), $DELIVER0007IST)
      EQUIVALENCE ($DELIVER0007BAT       (2:2), $DELIVER0007ISA)
      EQUIVALENCE ($DELIVER0007BAT       (3:3), $DELIVER0007EST)
      EQUIVALENCE ($DELIVER0007BAT       (4:4), $DELIVER0007OCT)
*
*
      CHARACTER  *    4  $DELIVER0008FAB
      CHARACTER  *    4  $DELIVER0008BAT
      CHARACTER  *    1  $DELIVER0008IST
      CHARACTER  *    1  $DELIVER0008ISA
      CHARACTER  *    1  $DELIVER0008EST
      CHARACTER  *    1  $DELIVER0008OCT
*
      EQUIVALENCE (DELIVERATTR  (   73:   76), $DELIVER0008FAB)
      EQUIVALENCE ($DELIVER0008FAB     ( 1: 4), $DELIVER0008BAT)
      EQUIVALENCE ($DELIVER0008BAT       (1:1), $DELIVER0008IST)
      EQUIVALENCE ($DELIVER0008BAT       (2:2), $DELIVER0008ISA)
      EQUIVALENCE ($DELIVER0008BAT       (3:3), $DELIVER0008EST)
      EQUIVALENCE ($DELIVER0008BAT       (4:4), $DELIVER0008OCT)
*
*
      CHARACTER  *    4  $DELIVER0009FAB
      CHARACTER  *    4  $DELIVER0009BAT
      CHARACTER  *    1  $DELIVER0009IST
```

```
      CHARACTER  *    1  $DELIVER0009ISA
      CHARACTER  *    1  $DELIVER0009EST
      CHARACTER  *    1  $DELIVER0009OCT
*
      EQUIVALENCE (DELIVERATTR  (   77:   80), $DELIVER0009FAB)
      EQUIVALENCE ($DELIVER0009FAB     ( 1: 4), $DELIVER0009BAT)
      EQUIVALENCE ($DELIVER0009BAT       (1:1), $DELIVER0009IST)
      EQUIVALENCE ($DELIVER0009BAT       (2:2), $DELIVER0009ISA)
      EQUIVALENCE ($DELIVER0009BAT       (3:3), $DELIVER0009EST)
      EQUIVALENCE ($DELIVER0009BAT       (4:4), $DELIVER0009OCT)
*
*
      CHARACTER  *    4  $DELIVER0010FAB
      CHARACTER  *    4  $DELIVER0010BAT
      CHARACTER  *    1  $DELIVER0010IST
      CHARACTER  *    1  $DELIVER0010ISA
      CHARACTER  *    1  $DELIVER0010EST
      CHARACTER  *    1  $DELIVER0010OCT
*
      EQUIVALENCE (DELIVERATTR  (   81:   84), $DELIVER0010FAB)
      EQUIVALENCE ($DELIVER0010FAB     ( 1: 4), $DELIVER0010BAT)
      EQUIVALENCE ($DELIVER0010BAT       (1:1), $DELIVER0010IST)
      EQUIVALENCE ($DELIVER0010BAT       (2:2), $DELIVER0010ISA)
      EQUIVALENCE ($DELIVER0010BAT       (3:3), $DELIVER0010EST)
      EQUIVALENCE ($DELIVER0010BAT       (4:4), $DELIVER0010OCT)
*
*
      CHARACTER  *    4  $DELIVER0011FAB
      CHARACTER  *    4  $DELIVER0011BAT
      CHARACTER  *    1  $DELIVER0011IST
      CHARACTER  *    1  $DELIVER0011ISA
      CHARACTER  *    1  $DELIVER0011EST
      CHARACTER  *    1  $DELIVER0011OCT
*
      EQUIVALENCE (DELIVERATTR  (   85:   88), $DELIVER0011FAB)
      EQUIVALENCE ($DELIVER0011FAB     ( 1: 4), $DELIVER0011BAT)
      EQUIVALENCE ($DELIVER0011BAT       (1:1), $DELIVER0011IST)
      EQUIVALENCE ($DELIVER0011BAT       (2:2), $DELIVER0011ISA)
      EQUIVALENCE ($DELIVER0011BAT       (3:3), $DELIVER0011EST)
      EQUIVALENCE ($DELIVER0011BAT       (4:4), $DELIVER0011OCT)
*
*
      CHARACTER  *    4  $DELIVER0012FAB
      CHARACTER  *    4  $DELIVER0012BAT
      CHARACTER  *    1  $DELIVER0012IST
      CHARACTER  *    1  $DELIVER0012ISA
      CHARACTER  *    1  $DELIVER0012EST
```

```
      CHARACTER  *    1  $DELIVER0012OCT
*
      EQUIVALENCE (DELIVERATTR  (   89:   92), $DELIVER0012FAB)
      EQUIVALENCE ($DELIVER0012FAB     ( 1: 4), $DELIVER0012BAT)
      EQUIVALENCE ($DELIVER0012BAT       (1:1), $DELIVER0012IST)
      EQUIVALENCE ($DELIVER0012BAT       (2:2), $DELIVER0012ISA)
      EQUIVALENCE ($DELIVER0012BAT       (3:3), $DELIVER0012EST)
      EQUIVALENCE ($DELIVER0012BAT       (4:4), $DELIVER0012OCT)
*
*
      CHARACTER  *    4  $DELIVER0013FAB
      CHARACTER  *    4  $DELIVER0013BAT
      CHARACTER  *    1  $DELIVER0013IST
      CHARACTER  *    1  $DELIVER0013ISA
      CHARACTER  *    1  $DELIVER0013EST
      CHARACTER  *    1  $DELIVER0013OCT
*
      EQUIVALENCE (DELIVERATTR  (   93:   96), $DELIVER0013FAB)
      EQUIVALENCE ($DELIVER0013FAB     ( 1: 4), $DELIVER0013BAT)
      EQUIVALENCE ($DELIVER0013BAT       (1:1), $DELIVER0013IST)
      EQUIVALENCE ($DELIVER0013BAT       (2:2), $DELIVER0013ISA)
      EQUIVALENCE ($DELIVER0013BAT       (3:3), $DELIVER0013EST)
      EQUIVALENCE ($DELIVER0013BAT       (4:4), $DELIVER0013OCT)
*
*
      CHARACTER  *    4  $DELIVER0014FAB
      CHARACTER  *    4  $DELIVER0014BAT
      CHARACTER  *    1  $DELIVER0014IST
      CHARACTER  *    1  $DELIVER0014ISA
      CHARACTER  *    1  $DELIVER0014EST
      CHARACTER  *    1  $DELIVER0014OCT
*
      EQUIVALENCE (DELIVERATTR  (   97:  100), $DELIVER0014FAB)
      EQUIVALENCE ($DELIVER0014FAB     ( 1: 4), $DELIVER0014BAT)
      EQUIVALENCE ($DELIVER0014BAT       (1:1), $DELIVER0014IST)
      EQUIVALENCE ($DELIVER0014BAT       (2:2), $DELIVER0014ISA)
      EQUIVALENCE ($DELIVER0014BAT       (3:3), $DELIVER0014EST)
      EQUIVALENCE ($DELIVER0014BAT       (4:4), $DELIVER0014OCT)
*
*
      CHARACTER  *    4  $DELIVER0015FAB
      CHARACTER  *    4  $DELIVER0015BAT
      CHARACTER  *    1  $DELIVER0015IST
      CHARACTER  *    1  $DELIVER0015ISA
      CHARACTER  *    1  $DELIVER0015EST
      CHARACTER  *    1  $DELIVER0015OCT
*
```

```
      EQUIVALENCE (DELIVERATTR  (  101:  104), $DELIVER0015FAB)
      EQUIVALENCE ($DELIVER0015FAB     ( 1: 4), $DELIVER0015BAT)
      EQUIVALENCE ($DELIVER0015BAT       (1:1), $DELIVER0015IST)
      EQUIVALENCE ($DELIVER0015BAT       (2:2), $DELIVER0015ISA)
      EQUIVALENCE ($DELIVER0015BAT       (3:3), $DELIVER0015EST)
      EQUIVALENCE ($DELIVER0015BAT       (4:4), $DELIVER0015OCT)
*
*
      CHARACTER  *    4  SUBTOTALFAB
      CHARACTER  *    4  SUBTOTALBAT
      CHARACTER  *    1  SUBTOTALIST
      CHARACTER  *    1  SUBTOTALISA
      CHARACTER  *    1  SUBTOTALST
      CHARACTER  *    1  SUBTOTALOCT
*
      EQUIVALENCE (DELIVERATTR  (  105:  108), SUBTOTALFAB)
      EQUIVALENCE (SUBTOTALFAB         ( 1: 4), SUBTOTALBAT)
      EQUIVALENCE (SUBTOTALBAT           (1:1), SUBTOTALIST)
      EQUIVALENCE (SUBTOTALBAT           (2:2), SUBTOTALISA)
      EQUIVALENCE (SUBTOTALBAT           (3:3), SUBTOTALST)
      EQUIVALENCE (SUBTOTALBAT           (4:4), SUBTOTALOCT)
*
*
      CHARACTER  *    4  SALESTAXFAB
      CHARACTER  *    4  SALESTAXBAT
      CHARACTER  *    1  SALESTAXIST
      CHARACTER  *    1  SALESTAXISA
      CHARACTER  *    1  SALESTAXEST
      CHARACTER  *    1  SALESTAXOCT
*
      EQUIVALENCE (DELIVERATTR  (  109:  112), SALESTAXFAB)
      EQUIVALENCE (SALESTAXFAB         ( 1: 4), SALESTAXBAT)
      EQUIVALENCE (SALESTAXBAT           (1:1), SALESTAXIST)
      EQUIVALENCE (SALESTAXBAT           (2:2), SALESTAXISA)
      EQUIVALENCE (SALESTAXBAT           (3:3), SALESTAXEST)
      EQUIVALENCE (SALESTAXBAT           (4:4), SALESTAXOCT)
*
*
      CHARACTER  *    4  TOTALFAB
      CHARACTER  *    4  TOTALBAT
      CHARACTER  *    1  TOTALIST
      CHARACTER  *    1  TOTALISA
      CHARACTER  *    1  TOTALEST
      CHARACTER  *    1  TOTALOCT
*
      EQUIVALENCE (DELIVERATTR  (  113:  116), TOTALFAB)
      EQUIVALENCE (TOTALFAB            ( 1: 4), TOTALBAT)
```

```
      EQUIVALENCE (TOTALBAT             (1:1), TOTALIST)
      EQUIVALENCE (TOTALBAT             (2:2), TOTALISA)
      EQUIVALENCE (TOTALBAT             (3:3), TOTALEST)
      EQUIVALENCE (TOTALBAT             (4:4), TOTALOCT)
*
*
*****************************************************************
*                        FIELD DATA PART                        *
*****************************************************************
*
      CHARACTER  *  360  DELIVERDATA
*
      CHARACTER  *   25  NAME
      CHARACTER  *   26  STREET
      CHARACTER  *    5  ZIPCODE
      CHARACTER  *   24  CITY
      CHARACTER  *   12  CUSTMRNO
      CHARACTER  *   14  DATE
      CHARACTER  *    5  ARTICLENUMBE
      CHARACTER  *   28  DESIGNATION
      CHARACTER  *    5  QTY
      CHARACTER  *    6  UNITPRICE
      CHARACTER  *   11  PRICE
      CHARACTER  *    5  $DELIVER0001
      CHARACTER  *   28  $DELIVER0002
      CHARACTER  *    5  $DELIVER0003
      CHARACTER  *    6  $DELIVER0004
      CHARACTER  *   11  $DELIVER0005
      CHARACTER  *    5  $DELIVER0006
      CHARACTER  *   28  $DELIVER0007
      CHARACTER  *    5  $DELIVER0008
      CHARACTER  *    6  $DELIVER0009
      CHARACTER  *   11  $DELIVER0010
      CHARACTER  *    5  $DELIVER0011
      CHARACTER  *   28  $DELIVER0012
      CHARACTER  *    5  $DELIVER0013
      CHARACTER  *    6  $DELIVER0014
      CHARACTER  *   11  $DELIVER0015
      CHARACTER  *   12  SUBTOTAL
      CHARACTER  *   11  SALESTAX
      CHARACTER  *   12  TOTAL
*
*
      EQUIVALENCE (DELIVERDATA (    1:   25), NAME)
      EQUIVALENCE (DELIVERDATA (   26:   51), STREET)
      EQUIVALENCE (DELIVERDATA (   52:   56), ZIPCODE)
      EQUIVALENCE (DELIVERDATA (   57:   80), CITY)
```

```
      EQUIVALENCE (DELIVERDATA (   81:   92), CUSTMRNO)
      EQUIVALENCE (DELIVERDATA (   93:  106), DATE)
      EQUIVALENCE (DELIVERDATA (  107:  111), ARTICLENUMBE)
      EQUIVALENCE (DELIVERDATA (  112:  139), DESIGNATION)
      EQUIVALENCE (DELIVERDATA (  140:  144), QTY)
      EQUIVALENCE (DELIVERDATA (  145:  150), UNITPRICE)
      EQUIVALENCE (DELIVERDATA (  151:  161), PRICE)
      EQUIVALENCE (DELIVERDATA (  162:  166), $DELIVER0001)
      EQUIVALENCE (DELIVERDATA (  167:  194), $DELIVER0002)
      EQUIVALENCE (DELIVERDATA (  195:  199), $DELIVER0003)
      EQUIVALENCE (DELIVERDATA (  200:  205), $DELIVER0004)
      EQUIVALENCE (DELIVERDATA (  206:  216), $DELIVER0005)
      EQUIVALENCE (DELIVERDATA (  217:  221), $DELIVER0006)
      EQUIVALENCE (DELIVERDATA (  222:  249), $DELIVER0007)
      EQUIVALENCE (DELIVERDATA (  250:  254), $DELIVER0008)
      EQUIVALENCE (DELIVERDATA (  255:  260), $DELIVER0009)
      EQUIVALENCE (DELIVERDATA (  261:  271), $DELIVER0010)
      EQUIVALENCE (DELIVERDATA (  272:  276), $DELIVER0011)
      EQUIVALENCE (DELIVERDATA (  277:  304), $DELIVER0012)
      EQUIVALENCE (DELIVERDATA (  305:  309), $DELIVER0013)
      EQUIVALENCE (DELIVERDATA (  310:  315), $DELIVER0014)
      EQUIVALENCE (DELIVERDATA (  316:  326), $DELIVER0015)
      EQUIVALENCE (DELIVERDATA (  327:  338), SUBTOTAL)
      EQUIVALENCE (DELIVERDATA (  339:  349), SALESTAX)
      EQUIVALENCE (DELIVERDATA (  350:  361), TOTAL)
*
*
      CHARACTER  *  529  DELIVER
      EQUIVALENCE (DELIVER     (    1:   52), DELIVERGLOBALS)
      EQUIVALENCE (DELIVER     (   53:  168), DELIVERATTR)
      EQUIVALENCE (DELIVER     (  169:  529), DELIVERDATA)
```

17.2 Notes on multiple field mode

You can define more fields per line (multiple field mode) if your data display terminal (9750 or 9755 only) is set to the operating mode “Weight field handling character (FHC) as display control character (DCC)”.
Formats for the group of terminal types “9750 in multiple field mode” can also be created and maintained on the 9755 Data Display Terminal (setting “Weight FHC as DCC” in SIDA installation).
When modifying the general attributes of the format you should then specify the attribute “format can be used for 9750 in multiple field mode”. For new formats you define the multiple field mode function by altering the user profile.
Existing formats can be converted to multiple field mode with function 03 (Format Modification).
Once your data display terminal has been switched over to multiple field mode, you can still work in regular mode, i.e. you can also output formats that have not been converted to multiple field mode.

The table below shows you how many fields you can define per line using the multiple field mode function.

Number of fields per line depending on the specifications in the format definition using IFG:

	Data display terminal			
	9750, 9751		9755, 9763	
	without	with	without	with
Specification in IFG	blank		blank	
Default	24	15	80	
Multiple field mode	48	23		

An exact value for the maximum number of fields for the 9750 Data Display Terminal can be calculated on the following basis:

– one field separator for each field, including text fields or empty fields

– one field separator for every output field, excluding empty fields

The total number of field separators for the 9750 Data Display Terminal is restricted to 48, without taking any leading empty fields on a line into account.

Please note

– The specifications in the columns “with blank” apply to the least favorable case, i.e. when each defined field is separated from the next by an undefined area.

– The specifications in the columns “without blank” only apply when all the columns in a line are occupied.

– IFG carries out a process of field optimization when a field is followed by a text field and the space between these two fields is less than six characters. The text field and the space beetween the fields are combined into a single field in such cases, and an undefined area remains between the two fields. The number of possible fields per line thus also depends on the spacing of the fields relative to each other and the edge of the screen. In other words, you may be able to fit more fields per line by starting/ending the fields at the extreme edges of the screen.

In multiple field mode IFG causes the current field in the window to flash when field attributes are assigned to distinguish it from the rest.

Notes

– The hardware of the device being used to generate the formats for multiple field mode must be set to “Weight FHC as DCC” (a setting has to be changed in the controller of the data display terminal). On the 9755 Data Display Terminal “Weight FHC as DCC” can be set using the SIDA installation method. IFG cannot verify this.

– For physical programming or formatting in PDN, the device should be set to normal operation. If the operating mode “Weight FHC as DCC” is set, display attributes may be corrupted or lost.

Restrictions associated with multiple field mode:

– The field attributes “bright”, “normal brightness”, “blank”, “underline/italics” and “selectable” are ignored when the formats are used.

– “bright” is automatically linked with “unprotected”.

– “normal brightness” is automatically linked with “protected”.

– The “flashing” attribute is interpreted. How it is represented on your terminal depends on the setting of the data display terminal.

– For each field in formats using multiple field mode you may specify either no comment or select field names that are not longer than 16 characters.

– In multiple field mode you can only mark by overwriting (not with MAR key).

– A data transfer area with separate attribute blocks and field contents is not supported. Similarly, FHS-DE formats are not supported either.

If the format does not fill the entire screen, the remainder of the screen is represented by NUL characters (instead of blanking) when formats are used in multiple field mode. You can avoid this by defining an empty text field in the last position of the format with a start-of-field character.

17.3 Converting formats generated with FHS macros

Any formats created with FHS macros that you wish to work on with IFG must first be processed with a conversion aid. Both the field names and field attributes are preserved. The conversion aid consists of a BS2000 procedure and the program IFGCONV, which is called by the procedure.

Conditions and restrictions

– Formats you wish to convert must exist as FHS macros in files (translated formats cannot be converted) - one file per format, no START statement.

– An error-free assembly of the formats must be possible and they must be executable under FHS.

– The MODE=IN or MODE=OUT operand in the MDMAP macro is converted to MODE=MIX; hence programs containing an MCMAP call without the specification IN or OUT must be modified accordingly.

– Fields in a format that extend over more than one line are truncated so that they end with the first line. The field length is changed accordingly. In continuation lines the field is replaced by protected blanks.

– The format library in which the format is to be stored must exist and must contain the user profile USERPRO. IFGCONV will adopt the specifications in this user profile. taking the permissible length of field names and admissibility of comments into account. If a new format library is to be created during the conversion, the length of the field names will be limited to 8 characters and no comments will be permitted.

– The following are not converted:

  formats incorporating partial formats,

  formats containing the WIDE attribute,

  formats created for printers,

  formats for which less than 80 write positions were specified in the DEVICE=operand.

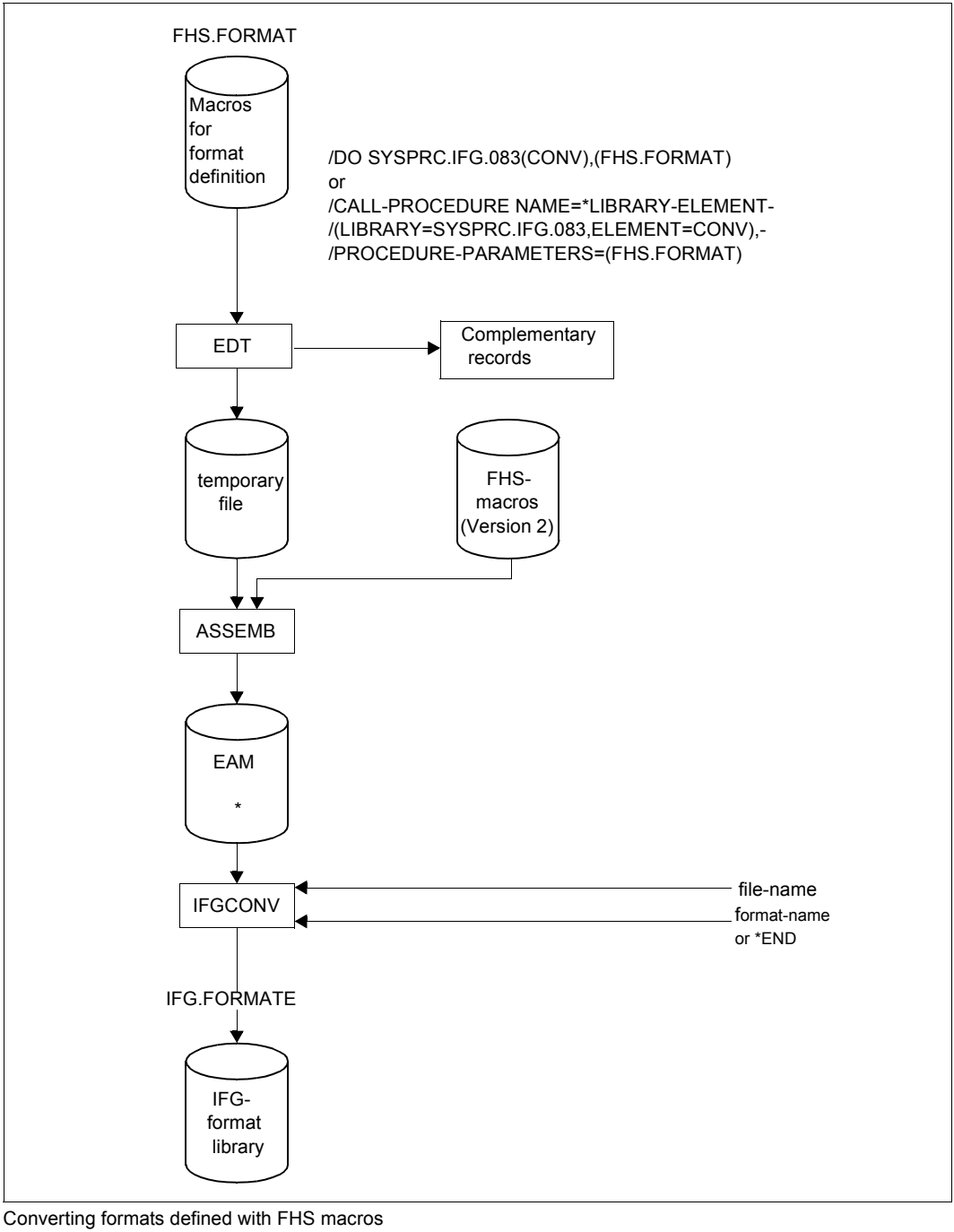
Formats that do not meet these conditions have to be created as new formats with IFG.

You initiate the conversion procedure with the command

```
/CALL-PROCEDURE NAME=*LIBRARY-ELEMENT(LIBRARY=SYSPRC.IFG.083,    -
                      ELEMENT=CONV),PROCEDURE-PARAMETERS=(FHS.FORMAT)
```

or

```
/DO SYSPRC.IFG.083(CONV)
```



Converting formats defined with FHS macros

&MAPFILE Name of the file with the FHS macro calls

&FHSMACLIB Name of the library which contains the FHS macros

IFG-FORMAT-FILENAME

Name of the IFG format library. This library must already exist and must contain the user profile USERPRO.

MAPNAME Name of the FHS format to be converted or *END if you wish to terminate the program IFGCONV.

If there were flags during the assembly, *END can be entered immediately.
In the following example the format UMTEST in the file FHS.FORMAT is converted and then stored in the file IFG.FORMATS.

```
/do sysprc.ifg.083(conv)
%&FHSMACLIB=syslib.fhs.083
ISAM FILE - DOING @GET
 EDT NORMAL END
V30.0A20 SIEMENS BS2000 ASSEMBLER READY
FLAGS IN 00000 STATEMENTS, 000 PRIVILEGED FLAGS, 000 MNOTES
IFG-FORMAT-FILE NAME ?
*ifg.formats
ENTER MAPNAME OR "*END"
*umtest
MDMAP-PARAMETER "MODE=IN/OUT" NOT SUPPORTED BY IFG, CHANGED TO "MIX"
MAP UMTEST  CONVERTED
/
```

17.4 Terminal generation

The following tables show how terminals supported by IFG and FHS are generated.

DT type	generated in IFG/FHS as
3270	3270
8161	8161
8162	8162
8160 9748 9749 9750 9751 9752 9753 9754	9750
9755 9756 9758 9762	9755
9763 9756-12x 9759	9763

Printer	**generated in**	
	IFG as	**FHS as**
3287	3287	3287
4011-N11/20 -N61/70	9011	9011 (ECMA emulation)
4810-P10	PCL	PCL
4813-I101 -I601	9011	9011 (ECMA emulation)
4819-P10	PLC	PLC
4820-P10/1000	PLC	PLC
4821-Pxx	PLC	PLC
4824-P20	PLC	PLC
9001-8931/832	9001-8931	9001-8931
9001-31/32	9001-8931	9001-31
9001-xxx	9001	9001
9002	9002	9002
9003	9003	9003
9004	9004	9004
9011-18/28	9011	9011-18
9011-19/29	9011	9011-19
9012	9012	9012
9013	9013	9013
9014-11	9013	9013
9014-15	9013	9013
9014-16	9013	9013
9021-2	PCL	PCL
9022-200	PCL	PCL
9022-200U	PCL	PCL
9022	9022	9022 (not 9022-300/300U)
9025	9013	9013
9097-10/20	9011	9011 (ECMA emulation)

17.5 Summary of 8-bit support

With IFG V7.1 or later it is possible to create and work with 8-bit formats. These formats contain characters from coded character sets. A coded character set can contain up to 191 characters, as opposed to the normal 90 characters per character set.

8-bit formats can only be created on 8-bit terminals (8-bit 9763, 9758 and 8-bit 9763) You must therefore make the appropriate terminal specifications in the current user profile. In addition, the XHCS product must be loaded.

The coded character set must be defined in the user profile (display attributes of the format). The usability of the character set and its compatibility with the display terminal are not checked until the character set is used. If a coded character set is not specified in the user profile, the default character set is used.

Creating a format

8-bit formats can only be created if you are using an 8-bit terminal and have specified an 8-bit terminal group in the user profile.
If you are using a 7-bit terminal and have specified an 8-bit terminal group in the user profile, formats cannot be created. In this case, an error message is output. If a 7-bit terminal group is defined for a 7-bit or 8-bit terminal, the terminal is automatically set to 7-bit mode, and a 7-bit format is created.

An 8-bit format is created using an 8-bit user standard character set. You must define this character set in VTSCUB (see “XHCS“ manual). If the character set used is not compatible with the display terminal or is a 7-bit character set, an error message is output, and no format is created.

Modifying and extending a format

If you are working on a 7-bit terminal, you can (as before) only modify a 7-bit format using the characters of EBCDIC.DF.03. 8-bit formats cannot be modified; if you attempt to do so, an error message is output.

If you are working on an 8-bit terminal and want to modify a 7-bit format, the terminal is automatically set to 7-bit mode. This prevents the use of invalid characters.
Your format can only be modified using an 8-bit user standard character set. If the user standard character set is not compatible with the terminal or with a 7-bit character set, the format is not modified and an error message is output.

Combining formats

If you are working on a 7-bit terminal, you can only combine 7-bit formats. The defined terminal group must be a 7-bit terminal group. A combination including an 8-bit format is not permissible.

If you are working on an 8-bit terminal with a defined 7-bit terminal group, you can only combine 7-bit formats. A combination including an 8-bit format is not permissible.

If you are working on an 8-bit terminal with a defined 8-bit terminal group, you can combine 7-bit or 8-bit formats. In this case, all of the 8-bit formats to be combined must use the character set specified in the user profile.

Converting terminal group formats

A 7-bit format or 7-bit terminal group can always be converted to an 8-bit format or 8-bit terminal group. In this instance, you use the coded character set defined in the user profile for this format. However, it is not possible to convert an 8-bit format or terminal group to a 7-bit format or terminal group.

Inserting images of formats

In a 7-bit format, only images of 7-bit formats can be inserted.

In an 8-bit format, you can insert images of an 8-bit format with the same character set, or a 7-bit format.

ICE character set

ICE character sets can only be used for 7-bit formats.

17.6 Index of screens listed

The index below lists the screens referred to in this manual. The relevant page numbers are provided.

Glossary

addressing aid

An addressing aid defines the data structure in the data transfer area. When formats are used, it enables the programmer to address the fields of a format by means of symbolic names.

attribute

Characteristic of a format or field for representation, editing or validation purposes. An attribute is defined either when a format is created using IFG (static attribute) or in the application program via the global attributes and field attributes (dynamic attributes).

attribute field

Field in the addressing aid in which the attributes can be entered.

bottom instruction

A basic format element that teaches the user about dialog with the format and when using the application (for help panels only).

character set file

File that contains character sets created with ICE.

command area

The command area comprises an input field which allows the user to control the application through the direct input of commands.

data transfer area (message area)

Area containing the fields and attributes which are accessible to the program. When formats are used, this area is used for data exchange between the application program and FHS.

dialog box

A dialog box (box) is an enclosed area on the screen in which the user can receive information, or define settings for operands and attributes by input or selection.

fast formatting

Function which can help considerably improve the performance of a format, but which is subject to certain restrictions. This function must be specified when a format is first created using IFG.

FHS-DE

Format Handling System V8 - Dialog extension

format

Logical data structure that defines a "form".

format application file

Library in which the format definitions are stored.

group field

Consecutive field can be combined to form a group field. All the fields of a group which are accessible to a program can then be addressed under a single name (group name) in the data transfer area.

input field

Field in which the terminal user enters data for the application program.

key list

Table of the keys to which commands are assigned.

list

A list is a series of data records. Each data record contains one or more different fields.

menu bar

The menu bar is the first line in the full screen. It is subdivided into fields known as menu titles. The menu bar is followed by a separator line. A pull-down menu is assigned to each menu title of the title bar.

menu title

The menu title is a field in the menu bar that displays a pull-down menu when it is selected.

messages

Messages contain explanations (information/warnings) which are aimed at the application group concerned.

message area

Area to which the program has access. When formats are being used, this area is used for the exchange of messages between the application program and FHS.

modal box

A box is modal when the user is unable to make entries in the underlying area while the box is open.

modeless box

A box that allows the user to continue working in the underlying area is defined as modeless.

multiple choice field

A multiple choice field contains a constant number of entries, from which the user can select one or more, or none at all. The entries are not mutually exclusive.

multiple field mode

Function that permits more fields per line in formats when a data display terminal is set to ’operating mode with a field control character’.

output field

Field in which the application program outputs data.

partial format

Format that occupies only a specific part of the screen.

prompt

Text or a character that requests you to make entries in the input field.

pull-down menu

A pull-down menu consists of a group of options (e.g. functions, characteristics) that are represented in the form of single-choice fields (one selection per line, no prompt text).

single choice field

A single choice field contains a constant number of entries, from which the user can select a maximum of one. The entries are mutually exclusive.

status area

The status area is a protected area in the upper part of the format which provides information on the standard format actions.

table

A table is defined as the repetition of a record of fields. Each record of fields constitutes a table element. All records must contain the same field types (e.g. same length, same attributes, etc.). The fields of a record can be addressed using one of the elements in a series.

text field

Field that contains fixed text defined when a format is created.

top instruction

A basic format element that prompts the user during dialog with the format and when using the application (for help panels only).

working area

The working area is the user's action area, containing all of the (basic) elements used to process the current task.

XHCS (Extended Host Code Support)

Extended support for the host code.

Related publications

Ordering manuals

The manuals are available as online manuals, see *http://manuals.fujitsu-siemens.com*, or in printed form which must be paid and ordered separately at *http://FSC-manualshop.com*.

[1] **FHS V8.3A (BS2000/OSD)**
Format Handling System for openUTM, TIAM, DCAM
User Guide

Target group
Programmers
Contents
Program interfaces of FHS for TIAM, DCAM and UTM applications. Generation, application and management of formats.

[2] **FHS V8.1A (BS2000/OSD, TRANSDATA)**
Dialog Extension for TIAM and SDF-P
User Guide

Target group
Application developers
Contents
The manual describes the program interface for using the FHS dialog manager in TIAM and SDF-P applications.

[3] **WIN-DOORS/FHS-DOORS** (BS2000/OSD, MS-Windows)
Optimizing Panels with the DOORS Editor
User Guide

Target Group
The manual addresses BS2000 developers who wish to optimize formats for use under WIN-DOORS/FHS-DOORS.
Contents
The manual describes how converted formats can be processed using the DOORS Editor. It explains how you work with the DOORS Editor, and the op-tions available for user-specific extensions using the Dialog Builder. It also contains a reference section on the object attributes. A description of the interface covers the online help system for the DOORS Editor.

[4] **FHS-DOORS** (BS2000/OSD, MS-Windows)
Graphical Interface for BS2000/OSD Applications
User Guide

Target group
The manual addresses BS2000 developers who wish to equip BS2000 applications with a graphical interface.
Contents
The manual describes the usage model and the functions of FHS-DOORS. A sample session provides an example of how you work with FHS-DOORS. The manual also describes the parameters with which sessions can be configured as required by applica-tions, and the interfaces for a library for format recognition and OLE automation. It also contains a description of the format converter FHS-DOORS-LC and of the Event Stream Service (ESS-DOORS).

[5] **openUTM V5.2**
Concepts and Functions
User Guide

Target group
Anyone who wants information about the functionality and performance capability of openUTM.
Contents
The manual contains a general description of all the functions and features of openUTM, plus introductory information designed to help first-time users of openUTM.

[6] **openUTM V5.2** (BS2000/OSD, UNIX, Windows)
Administering Applications
User Guide

Target group
This manual is intended for everyone responsible for administering openUTM applications and generating administration programs.
Contents
The manual describes the program interface to administration, which enables you to generate your own administration programs. It also describes the command interface to administration and the options available for the administration of message queues and printers.

[7] **openUTM V5.2** (BS2000/OSD, UNIX, Windows)
Generating Applications
User Guide

Target group
This manual is designed for use by application planners and developers as well as operators of UTM applications.
Contents
This manual describes how to define the configuration for a UTM application using the UTM tool KDCDEF and how to create the KDCFILE. One chapter also goes into more detail about the generation of selected objects and functions of the application.
Additional topics include the dynamic configuration of an application and the updating of the KDCFILE using the tool KDCUPD.

[8] **openUTM V5.2** (BS2000/OSD, UNIX, Windows)
Programming Applications with KDCS for COBOL, C and C++
User Guide

Target group
This manual is intended for programmers who wish to use the KDCS program interface for programming UTM applications.
Contents
The manual describes the KDCS interface in the form valid for COBOL, C and C++. This interface incorporates both the basic functions of the Universal Transaction Monitor and the calls for distributed processing. It also contains a description of working together with databases.

[9] **openUTM V5.2** (BS2000/OSD)
Messages, Debugging and Diagnostics
User Guide

Target group
This manual is intended for programmers, generators and administrators of UTM applications in BS2000/OSD.
Contents
The manual describes the debugging of UTM applications, the format of UTM dumps, behavior in the event of errors, and the openUTM message concept. It also includes all messages and return codes output by openUTM.

[10] **UTM** (TRANSDATA)
Supplement for Pascal-XT
User Guide

Target group
Programmers of UTM Pascal-XT applications
Contents
– Translation of the KDCS program interface into the language Pascal-XT
– All the information required by programmers of UTM Pascal-XT applications
Applications
BS2000 transaction processing

[11] **UTM** (TRANSDATA)
Supplement for PL/I
User Guide

Target group
Programmers of UTM PL/I applications
Contents
Translation of the KDCS program interface into the PL/I language, and all the information required by programmers of UTM PL/I applications

[12] **UTM (**RANSDATA)
Supplement for FORTRAN
User Guide

Target group
Programmers of UTM FORTRAN applications
Contents
Translation of the KDCS program interface into the FORTRAN language, and all the information required by programmers of UTM FORTRAN applications
Applications
BS2000 transaction processing

[13] **ASSEMBH** (BS2000)
Reference Manual

Target group
Users in a BS2000 environment who want to write programs in the assembly or macro language, and to use structured programming.
Contents
– Description of the language set of the ASSEMBH assembler in BS2000
– Structure of the assembly language; assembler instructions
– Structure and elements of the macro language; macro language instructions
– Structured programming with ASSEMBH
– Predefined macros for structured programming
– ILCS interface for structured programming

[14] **SDF-A** (BS2000/OSD)
User Guide

Target group
This manual is intended for experienced BS2000 users and system administration staff.
Contents
It describes how to process syntax files and explains the SDF-A functions on the basis of examples. The SDF-A statements are listed in alphabetical order.
The manual also includes a description of the SDF-SIM utility routine.

[15] **SDF-P** (BS2000/OSD)
Programming in the Command Language
User Guide

Target group
The manual addresses BS2000/OSD users and systems support.
Contents
SDF-P is a structured procedure language in BS2000. The introduction is followed by a detailed description of commands, functions and macros.

[16] **Packet Switching**
X.25PORT (PDN) and VTSU-X.29 (BS2000)
User Guide

Target group
Organizers and application planners, programmers, system and network administrators
Contents
Definition of important packet switching concepts, transporting packet data via circuit switching networks, link options with X.25PORT and VTSU-X.29, X.25-specific parts of the generation of a communication system, application interfaces.

[17] BS2000
Datenkommunikationssystem *
Technische Beschreibung
(Data Communication System, Technical Description)

Target group
- BS2000 users with an interest in the technical background of their systems (software engineers, systems analysts, computer center managers, system administrators)
- Computer scientists interested in studying a concrete example of a general-purpose operating system

Contents
Functions and principles of implementation of
- BCAM (Basic Communication Access Method)
- DCAM (Data Communication Access Method)
- TIAM (Terminal Interactive Access Method)
- RBAM (Remote Batch Access Method)
- VTSU-B (Virtual Terminal Support Basic)

Order number
U3214-J-Z53-1

[18] **RPG3** (BS2000)
RPG Compiler
User's Guide

Target group
RPG users in BS2000
Contents
- Calling and controlling the RPG3 compiler
- Input and compilation of source programs
- Generation and management of object and load modules
- Controlling program execution
- Runtime error handling
- File processing
- Terminal mapping support (FHS/IFG interface)
- Language interfacing (COBOL, assembler)
- /COPY statement
- DMS-Monitor

[19] **XHCS**
(BS2000/OSD)
8-Bit Code and Unicode Support in BS2000/OSD
User Guide

Target group
Application programmers and system administrators
Contents
XHCS (Extended Host Code Support) is a software package of BS2000/OSD that lets you use extended character sets and/or the Unicode character set in conjunction with 8-bit terminals. XHCS is also the central source of information on the coded character sets in BS2000/OSD.

[20] **Style Guide**
Guidelines on the Design of User Interfaces
User's Guide

Target group
Developers of application programs
Contents
The Style Guide contains rules and recommendations for the development of uniform user interfaces. It describes their structure and contents, and how they are used.

[21] **Unicode in BS2000/OSD**
Introduction

Target group
Application programmers and system administrators, who want to get an overview, to which extent the Unicode support is provided in BS2000/OSD, and which BS2000/OSD components you need for the Unicode support
Contents
This manual gives an overview of the Unicode support in BS2000/OSD and describes basics, concepts and correlations, which apply for all BS2000/OSD products concerned by Unicode. Thus it completes the product-specific description in the respective manuals. Helpful tables from the Unicode conversion surroundings complete the manual.

Index

Fujitsu Siemens Computers GmbH
User Documentation
81730 Munich
Germany

Fax: (++49) 700 / 372 00001

e-mail: manuals@fujitsu-siemens.com
http://manuals.fujitsu-siemens.com

Comments
Suggestions
Corrections

Submitted by

Comments on IFG V8.3A
IFG for FHS

Information on this document

On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document from the document archive refers to a product version which was released a considerable time ago or which is no longer marketed.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format *…@ts.fujitsu.com*.

The Internet pages of Fujitsu Technology Solutions are available at *http://ts.fujitsu.com/*...
and the user documentation at *http://manuals.ts.fujitsu.com*.

Hinweise zum vorliegenden Dokument

Zum 1. April 2009 ist Fujitsu Siemens Computers in den alleinigen Besitz von Fujitsu übergegangen. Diese neue Tochtergesellschaft von Fujitsu trägt seitdem den Namen Fujitsu Technology Solutions.

Das vorliegende Dokument aus dem Dokumentenarchiv bezieht sich auf eine bereits vor längerer Zeit freigegebene oder nicht mehr im Vertrieb befindliche Produktversion.

Bitte beachten Sie, dass alle Firmenbezüge und Copyrights im vorliegenden Dokument rechtlich auf Fujitsu Technology Solutions übergegangen sind.

Kontakt- und Supportadressen werden nun von Fujitsu Technology Solutions angeboten und haben die Form *…@ts.fujitsu.com*.

Die Internetseiten von Fujitsu Technology Solutions finden Sie unter *http://de.ts.fujitsu.com/*..., und unter *http://manuals.ts.fujitsu.com* finden Sie die Benutzerdokumentation.