

Host Forms Description Language 3.2 for IBM MVS Creating Forms

HFDL 3.2
August 1995
720P30610

Xerox Corporation
701 South Aviation Boulevard
El Segundo, California 90245

©1995 by Xerox Corporation. All rights reserved.

Copyright protection claimed includes all forms and matters of copyrightable material and information now allowed by statutory or judicial law or hereinafter granted, including without limitation, material generated from the software programs which are displayed on the screen such as icons, screen displays, looks, etc.

Printed in the United States of America.

Xerox® and all Xerox products mentioned in this publication are trademarks of Xerox Corporation. Products and trademarks of other companies are also acknowledged.

IBM® and all IBM products mentioned in this publication are trademarks of International Business Machines Corporation.

Changes are periodically made to this document. Technical updates will be included in subsequent editions.

Table of contents

Introduction	vii
About this manual	vii
Contents	vii
Conventions	viii
Related publications	viii
Xerox publications	viii
IBM publications	ix
1. Overview	1-1
HFDL features	1-1
Prerequisites	1-2
Operating environment	1-2
Hardware requirements	1-2
Software requirements	1-2
Xerox laser printer support	1-2
Centralized printers	1-2
Decentralized printers	1-3
Contrasting HFDL with a typical forms application	1-3
Forms you can create with HFDL	1-4
2. Planning your forms	2-1
Determining format	2-1
Page orientation	2-1
Page frames	2-2
System page	2-2
Physical page	2-3
Virtual page	2-3
Form origin	2-3
Positioning form elements	2-4
Grids	2-5
X and Y coordinates	2-5
Negative coordinates	2-6
Unit value	2-6
Data types	2-6
Forms data	2-6

	Variable data	2-6
Fonts		2-7
	Fixed and proportionally spaced fonts	2-7
	Font orientation	2-7
	Font orientation when using short-edge paper feed	2-8
Virtual keyboard and font character maps		2-9
	Virtual keyboard character maps	2-10
	Font character maps	2-11
Adding color to forms		2-11
	Ink color availability and control	2-11
Choosing paper size		2-12
Forms design ruler		2-12
	Ruler scales	2-12
	Suggested uses	2-13

3. Using HFDL commands		3-1
Creating forms with HFDL commands		3-1
	Available specifications	3-1
Command format		3-2
	Reserved words	3-2
	Special characters	3-2
	Understanding the command syntax flow charts	3-3
Setting up your form		3-3
	FORM command	3-4
	PAPER command	3-4
	LANDSCAPE/PORTRAIT command	3-5
	GRID command	3-7
	FONT command	3-8
	ICATALOG command	3-9
	PALETTE command	3-9
	INK command	3-10
	Inkname format	3-11
	IRESULT command	3-12
Description commands		3-12
	LINE command	3-13
	BOX command	3-14
	TEXT AT command	3-17
	TEXT IN BOX command	3-21
	LOGO command	3-25
	GRAPHIC command	3-26
	SECTION command	3-27

	DO SECTION command	3–28
	COLUMN command	3–30
	INTERWORD command	3–30
	The interword gap formula	3–30
	Specifying the base gap	3–31
	Specifying the interword multiplier	3–31
	Using SPACING and INTERWORD together	3–31
	DOT LEADER command	3–32
	LET command	3–33
	Relative coordinate	3–33
	COMMENT command	3–34
	END command	3–35
4.	Creating a sample form	4–1
	Setting up the form	4–2
	Using the sample description command	4–3
5.	Managing and printing sample forms	5–1
	JCL requirements	5–1
	Using the COMPILE command	5–1
	Command coding conventions	5–1
	Command syntax	5–1
	Command syntax examples	5–5
	Batch processing COMPILE commands	5–5
	HFDL compiler JCL	5–6
	Downloading and printing a sample	5–10
	Accessing compiled forms	5–12
	Editing an existing form	5–12
	Specifying monochrome forms	5–12
6.	Messages	6–1
	Message indicators	6–1
Appendices		
	A. System considerations	A–1
	Line density limits	A–1
	Local density and page setup errors	A–1
	Boxes and their image	A–2
	Text in boxes	A–2
	Using lines within sections	A–2
	B. FMT grid format properties	B–1
	C. Approved abbreviations for reserved words	C–1
	D. Comparison of HFDL and FDL	D–1

E.	Supported paper sizes	E-1
Glossary		GLOSSARY-1
Index		INDEX-1

This manual describes the Host Forms Description Language (HFDL) and how you use it to create an electronic version of your form. The HFDL commands and statements you use comprise the Forms Source Language (FSL). The host computer compiles the FSL and stores it in the database. Using the resources management utilities, you can access and print your forms on Xerox laser printing systems (LPS) connected to your host computer.

About this manual

This manual is designed for users who are familiar with the MVS operating system and have experience using Xerox printers. For more information, refer to the "Overview" chapter.


This manual describes HFDL commands and statements, and how to plan and lay out your forms.

Before using this manual, become familiar with its contents and conventions.



Contents

This manual contains the following sections:

- | | |
|-------------------|---|
| Chapter 1 | "Overview" describes the HFDL features, operating environment, and hardware and software requirements. It also lists supported Xerox laser printing systems and common forms you can create and manage with HFDL. |
| Chapter 2 | "Planning your forms" describes how to plan and design forms. It shows you how to determine page orientation and format, add colors, select paper size, and use the design ruler. |
| Chapter 3 | "Using HFDL commands" describes the available commands and how to set up your forms. It also contains the command statement conventions and definitions you use to create the FSL and examples of the JCL. |
| Chapter 4 | "Creating a sample form" describes how to create a form using HFDL. |
| Chapter 5 | "Managing and printing sample forms" describes how to compile, store, and download forms to the printer. |
| Chapter 6 | "Messages" lists the error messages that may appear during forms compilation. |
| Appendix A | "System considerations" describes the printer limitations you may encounter during the creation of complex forms. |
| Appendix B | "FMT grid format properties" lists the grid and origin substitutions you can use for the supported paper sizes. |

Appendix C	"Approved abbreviations for reserved words" lists the abbreviations you can use for reserved words.
Appendix D	"Comparison of HFDL and FDL" compares the commands available for FDL, HFDL 2.1D, HFDL 3.1, and HFDL 3.2.
Appendix E	"Supported paper sizes" lists the paper sizes that HFDL supports.
	For detailed lists of virtual keyboard maps and font character maps used with HFDL, see the "Virtual keyboard and font character maps" appendix in the <i>Xerox Print Resources Manager/Host Forms Description Language 3.2 for IBM MVS Managing Resources</i> manual.

Conventions

	This manual uses the following conventions:
	In this manual, the word "character" refers to an "alpha-numeric character," except where noted.
"Supported paper sizes" appendix	References to chapters, sections, and appendices appear in quotation marks.
ICATALOG	Commands, operands, and parameters that you enter appear in bold.
<i>Host Forms Description Language 3.2 for IBM MVS Creating Forms</i>	Document names, library names, variable information, and user-defined data you enter appears in italic.
LPI	System commands and parameters appear in uppercase letters.
	Notes are hints that help you perform a task or understand the text.
	CAUTION: Cautions appear immediately before any action that may destroy the data stored on your network or damage your equipment. Make sure you understand the potential impact of the action before you perform it.

Related publications

You can find additional information related to HFDL in the following publications.

Xerox publications

Xerox related publications include the following:	
Publication	Number
<i>Xerox 871 Communications Module System Guide</i>	600P86733
<i>Xerox 3700 Laser Printing System Release Five Series Reference Manual</i>	720P84320
<i>Xerox 4235 Laser Printing System Operations Guide</i>	720P82510
<i>Xerox 4235 Laser Printing System Production Print Mode PDL/DJDE Reference</i>	720P82530
<i>Xerox 4235 Laser Printing System Operator Maintenance and Problem Solving Guide</i>	720P82540

Publication	Number
<i>Xerox 4235 Laser Printing System Xerox Distributed Print Mode XES Printer Command Reference</i>	720P82520
<i>Xerox 9790/8790 Laser Printing Systems Reference Manual</i>	720P90000
<i>Xerox Host Forms Description Language 3.2 for IBM MVS Installation Guide</i>	720P30600
<i>Xerox LPS Forms Creation Guide</i>	720P90081
<i>Xerox LPS Operations Reference</i>	720P90041
<i>Xerox LPS PDL/DJDE Reference</i>	720P90051
<i>Xerox LPS Print Description Language Quick Reference Card</i>	720P60681
<i>Xerox LPS Standard Font Library User Guide</i>	720P86174
<i>Xerox LPS Tape Formats Manual</i>	720P86175
<i>Xerox Print Resources Manager/ Host Forms Description Language 3.2 for IBM MVS Managing Resources</i>	720P30560
<i>Xerox Print Resources Manager/ Host Forms Description Language 3.2 for IBM MVS Installation Guide</i>	720P30550

IBM publications

IBM publications related to the HFDL include the following:

IBM MVS/ESA JCL Reference
IBM MVS/ESA User's Guide
IBM MVS/XA JCL Reference
IBM MVS/XA User's Guide
IBM MVS/XA JCL

The Host Forms Description Language (HFDL) operates in a standard IBM/MVS environment that includes Xerox high-speed, high-volume centralized printers. You use Xerox Print Resources Manager (XPRM) to store form files and resources in the XPRM database.

This chapter describes the HFDL features, prerequisites, operating environment, and the supported printers. Refer to the *Xerox Print Resources Manager/Host Forms Description Language 3.2 for IBM for MVS Managing Resources* manual for more information.

HFDL features

HFDL, combined with XPRM, provides the following features:

- Flexible language statement syntax. The syntax allows you to quickly and easily develop your form electronically. The syntax is similar to the source language that resides on the Xerox laser printing system (LPS) printer and that the Forms Description Language (FDL) compiler uses. HFDL also allows additional command extensions, such as dot leaders, which the FSL compiler does not support.
- More online printer time. Through HFDL and XPRM, you have the ability to compile and store forms on your host computer, freeing up your Xerox laser printer for other printing jobs.
- Powerful utilities. The XPRM utilities allow you to manage your resources through a system interface. These utilities allow you to access stored forms for updating, downloading to printers, or deleting.
- Additional flexibility. You have the ability to download host-generated forms, which were compiled in different formats, to specific laser printers.
- Increase in several FDL limits.
- Standard entries. You have the ability to enter standard HFDL reserved words and identifiers in uppercase or lowercase letters.
- Host system interface. XPRM provides an enhanced resource management capability through a host system interface. This allows you to take advantage of the enhanced features offered by the new family of Xerox laser printers such as the Xerox 4135 LPS Printer, the Xerox 4235 LPS Printer, the Xerox 4700 Color Document Printer, and the Xerox 4850 HighLight Color LPS Printer.

- Conversion from .FRM to metacode. This feature is implemented as a stand-alone utility program. For details see Appendix D.

HFDL also allows you to take advantage of the added features of the Xerox 4650 Printer (600 dpi forms).

Prerequisites

You should have a working knowledge of your Multiple Virtual Storage (MVS) operating system, and be able to perform the following procedures:

- Create and edit a text file on your system.
- Use JCL to compile a job on your system.
- Use JCL to access a file through your resource management interface, and download your selected file to your Xerox laser printing system.

Operating environment

The host computer, peripheral devices, printers, and software make up your printing system. This section describes the hardware and software requirements to operate HFDL.

Hardware requirements

HFDL 3.2 operates on IBM S/370 and S/390 series computers running the MVS/XA or MVS/ESA operating system.

Software requirements

HFDL operates in batch mode under the IBM MVS/ESA and MVS/XA operating systems. It does not use operating system features specific to any MVS release level. Any specific requirements for your site are provided during installation and described in the *Host Forms Description Language 3.2 for IBM MVS Installation Guide*.

Xerox laser printer support

HFDL supports Xerox centralized and decentralized printers.

Centralized printers

Centralized printers are high-speed, high-volume printers generally used in central locations for printing data processing output from mainframe computers. They may be attached to the host computer by an S/370 channel or attached remotely by an 871 Communications Module (CM) or BARR SNA.

Table 1-1 lists the centralized printers that HFDL supports and the associated operating system software (OSS).

Table 1–1. Supported centralized printers

Centralized printers	OSS version
4050	V2.1, V3.5, or V3.8
4090	V2.1, V3.5, or V3.8
4135	V3.6
4650	V3.5 or V3.8
4850	V3.7, V3.8, or V5.0
4890	V4.0 or V5.0
8790	V2.1 or V3.9
9790	V2.1 or V3.9

Decentralized printers

HFDL also supports some Xerox decentralized printers. These printers are generally in remote locations away from the mainframe. An SNA/RJE communications link or a BSC/RJE communications link connects them to the host. Also, the Xerox Flex Box protocol converter allows you to connect the Xerox 4235 printer to the host.

Table 1–2 lists the decentralized printers that HFDL supports and the associated OSS.

Table 1–2. Supported decentralized printers

Decentralized printers	OSS version
3700	V5.2
4235	V1.2 or V1.5 (XPPM and XDPM modes)
4700	V1.0 (XES mode)



HFDL 3.2 generates forms on the Xerox 4235 and Xerox 4700 printers operating in XES mode.

Contrasting HFDL with a typical forms application

HFDL allows you to create electronic versions of your forms, compile them, store them in your system database, and download them to your printer as needed. This eliminates the need to stock large quantities of preprinted forms.

Once you download your form to a Xerox LPS, the printer merges your form with variable data. This capability allows you to create, store, and print any number of forms, in any order, on plain paper.

In a typical forms application that does not use HFDL, you have to prepare and load preprinted forms (often printed in large quantities) on a printer connected to a computer. You can load

only one set of forms at a time, and must remove and replace any forms currently loaded with the preprinted forms you want.

The typical forms application usually requires a stock of preprinted forms. If there are different forms, storage costs can be high. If you have to modify preprinted forms, you usually must discard old versions, making your costs even higher.

Forms you can create with HFDL

The following list provides examples of general business forms you can create with HFDL:

- Bar codes
- Billing statements and invoices
- Calendars
- Checklists
- Financial statements
- Floor plans
- Flow diagrams
- Marketing surveys
- Order forms
- Organizational charts
- Packing and shipping slips
- Production and inventory control forms
- Reorder lists
- Reports
- Routing forms
- Schedules
- Status report forms.

Depending on your business, you may need to create forms to meet specific industry standards. HFDL allows you to create the following types of specialized forms:

- Bank statements
- Government compliance forms
- Insurance forms
- Tax return forms.

This chapter describes the parameters you must determine and specify when designing your form.

Careful planning and design are critical to creating electronic versions of your forms. Form planning and design specify such things as page size, orientation, layout, fonts, inks, and highlight colors.

For additional information about designing complex forms and Xerox laser printer capabilities, refer to the "System considerations" appendix.

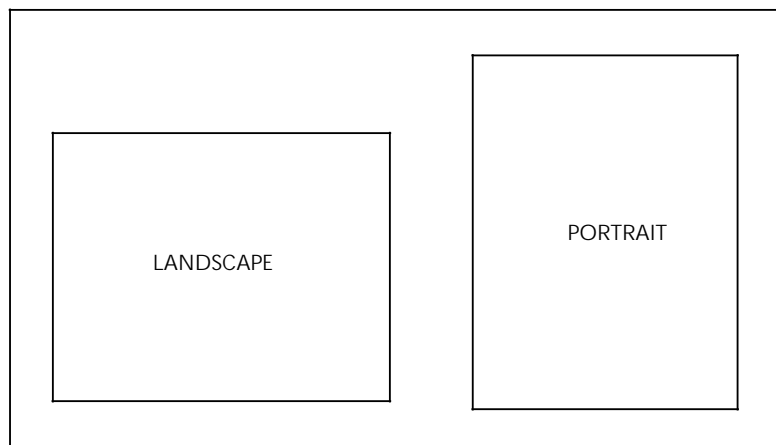
Determining format

Before you create a form with HFDL command statements, you must plan its layout, content, and appearance according to the requirements described in the following sections.

Page orientation

The orientation of a page is either portrait or landscape. Figure 2-1 shows these orientations.

Figure 2-1. **Page orientation**



The landscape orientation, sometimes called the horizontal format, refers to a page whose horizontal side is longer than its vertical side.

The portrait orientation, sometimes called a vertical format, refers to a page whose vertical side is longer than its horizontal side. Use the portrait orientation for text material such as letters, manuals, reports, and forms that need the tall vertical format.

Landscape format is ideal for forms, spreadsheets, and reports that require columns across the page. For example, the landscape orientation is appropriate for a Gantt chart that lists six or twelve months horizontally, and lists the tasks vertically.

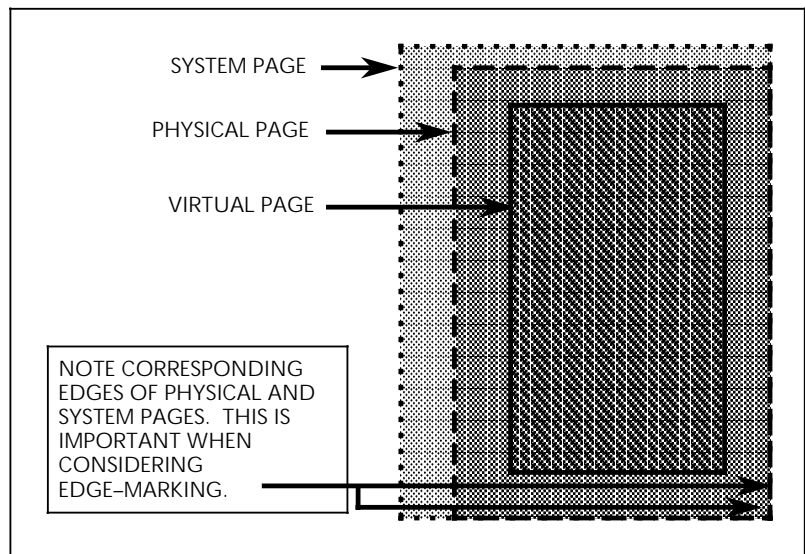


Large paper size forms in short edge feed (SEF) made for the Xerox 4135 printer require you to specify the **FORMATS** command in the opposite orientation. For example, for a portrait orientation, use landscape fonts, and for a landscape orientation, use portrait fonts. Make sure you specify the orientation you want. For example, for a portrait form, specify **PORTRAIT** for the orientation, and for a landscape form, specify **LANDSCAPE**.

Page frames

The page frame is a set of boundaries associated with a page as a unit of printing or imaging. There are three types of page frames: system, physical, and virtual. Figure 2-2 shows the relationship between the system, physical, and virtual page.

Figure 2-2. **Page frames**



For additional information about page frames, refer to the *Xerox LPS Forms Creation Guide*.

System page

The system page is the maximum area the printer allows you to image graphics and text. The system page is slightly larger than the physical page. If the default paper sizes have been established for your printer, the system page has little bearing on your form design.

If you decide to place marks along the edge of the physical page (edge-marking) in your form, be familiar with the edge-marking limitations. To accommodate edge-marking, the system page must be larger than the physical page. Since the system page boundaries correspond to at least two edges of the physical page boundaries for all paper sizes, you can only use edge-marking on the two edges of the physical page that do not correspond to the system page.

The product guide for each printer contains specific information about edge-marking.

Physical page



CAUTION: Unexpected results occur if you plan a form for a physical page paper size that is not available.

The physical page is the actual paper size. On a Xerox LPS, you define the default size of the physical page during system generation (SYSGEN). During planning, you must consider the default paper size or other paper sizes available for your printer. See the "Supported paper sizes" appendix for more information.

Virtual page

The virtual page is the area you specify for printing. The virtual page can be as large as the physical page (actual paper size), but can never be larger than the physical page. If you do not specify the virtual page size, the defined (planned) area of the virtual page is the same as the physical page.

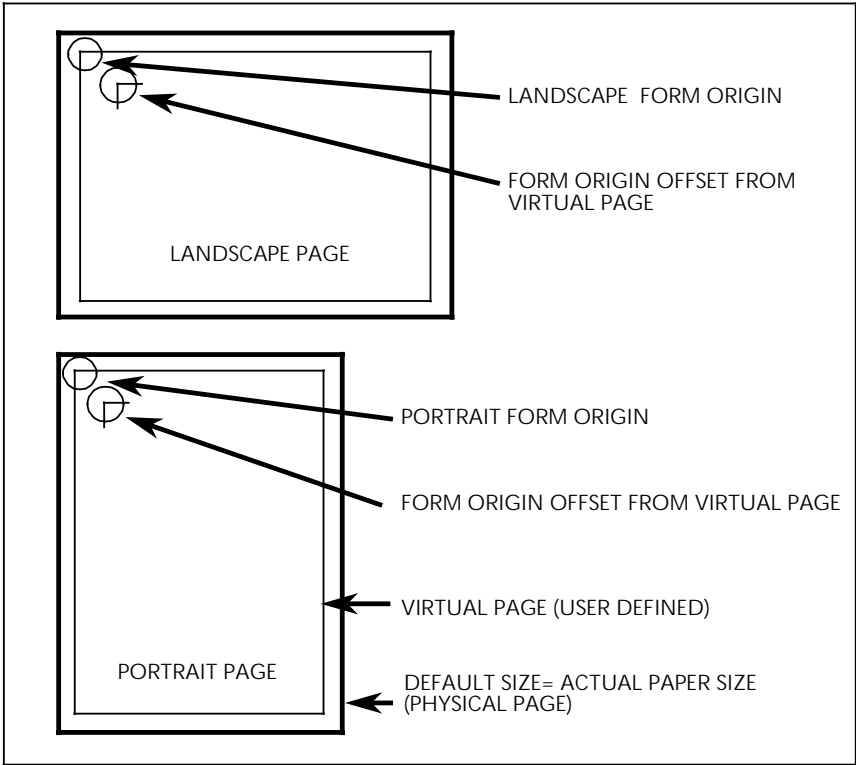
Since the Xerox 4700 Color Document Printer has a limitation on the print area of the physical page, make sure that the virtual page does not include the area the printer grippers cover. Refer to your printer documentation for more information on print area limitations.

Form origin

All form data described by HFDL is positioned on the page relative to a point, called the form origin. The form origin is a reference point from which you place all elements on a form using HFDL commands. The form origin is a point that is offset from the upper left corner of the virtual page. You can specify this point during form description. If you do not define a virtual page and specify a form origin, the default form origin is the origin of the physical page.

Figure 2-3 shows the location of the form origin of the virtual page in relation to the actual page size.

Figure 2-3. **Form origin location**



Positioning form elements

You can place the following elements on a form with HFDL:

- Boxes
- Fixed text
- Images
- Lines
- Logos and signatures
- Sections.

You locate an element by determining the horizontal (x) and vertical (y) distance from the upper-left corner (origin) of the form. These coordinates (x, y) comprise the origin of the document. The form origin (upper left corner of the page) is always (0,0).

Form element	Element origin
Vertical ruled line	Top of the line at the midpoint of the thickness of the line.
Horizontal ruled line	Left end of the line at the midpoint of the thickness of the line.
Box (outlined)	Upper-left corner of the box at the midpoint of the thickness of the outline.
Box (shaded)	Upper-left corner of the box.

Form element	Element origin
Fixed text (single line)	Upper-left corner of the first character cell when the line appears upright.
Fixed text (multiple lines)	Upper edge of the topmost character cell and the leftmost edge of the leftmost character cell when viewed in an upright position.
Graphics	Upper-left corner of the image.
Logos and signatures	Upper-left corner of the first character cell.
Sections (of a form)	Upper-left corner of the section.

Grids

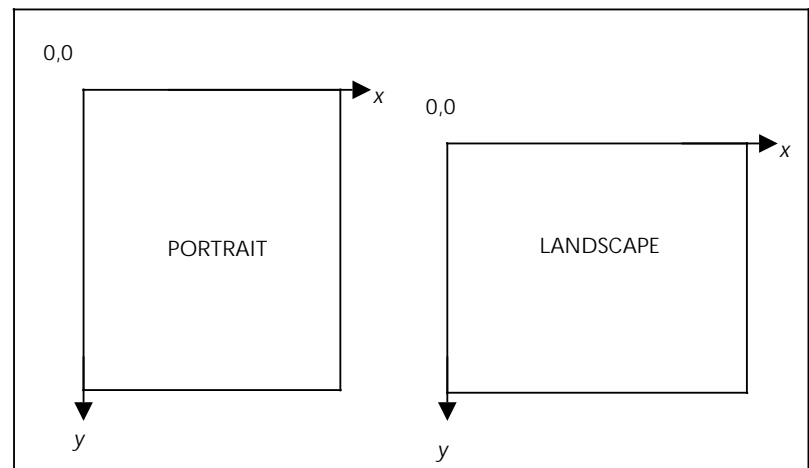
You specify the location of a form element (alphanumeric character, line, or box) on a page in terms of its horizontal and vertical displacement from the form origin. You can use these units of measure to define this displacement:

- Characters per inch (cpi) horizontally
- Dots per inch (300 dpi)
- Linear units (inches or centimeters)
- Lines per inch (lpi) vertically
- Xdots (600 dpi).

X and Y coordinates

If you draw a horizontal and vertical line through the form origin to create *x* and *y* coordinates, the location of the form origin is expressed as *y*=0, *x*=0. When specifying both coordinates, provide the *y* value first. Figure 2-4 shows the correct placement for *x* and *y* coordinates.

Figure 2-4. **X and Y coordinates**



If you draw additional lines horizontally and vertically away from the form origin so that each additional line is one unit of measure apart, you create a grid. You can now position form elements by specifying a grid position.

Negative coordinates

You can use negative x and y values to place the form elements above and to the left of the form origin. You can only specify negative values within the system page boundaries.

Unit value

You can specify the unit of measure you want to use to position form elements on the page. However, predefined formats automatically select the grid values for you. The "FMT grid format properties" appendix lists the predefined grid format properties for various page sizes.

Data types

There are two types of data: forms data and variable data. Make sure you consider them when you create forms for printing on Xerox LPS printers.

Forms data

Forms data is the information on a form that remains the same, such as form titles and column titles. Typically, fixed-length forms data changes only when you modify the form. Forms data is the only data you enter in the design process.

When designing a column size you must consider the amount of variable data the column accommodates, not the size of the fixed-length column title.

Variable data

Variable data is the data you overlay in the space provided by the designed form. Variable data varies in length. An example of variable data is a name and address.

You must allow sufficient space in the columns to accommodate the variable data. Otherwise, variable data may be invalid or unusable. For example, if you do not provide sufficient space for an address, the zip code could be truncated. Spacing for variable data is extremely important and is more accurately calculated with fixed spacing of the font characters.

Fonts

Fonts are character sets, each having a unique typeface (Times, Roman or Helvetica), type size (points), and type style (normal, bold, or italic).

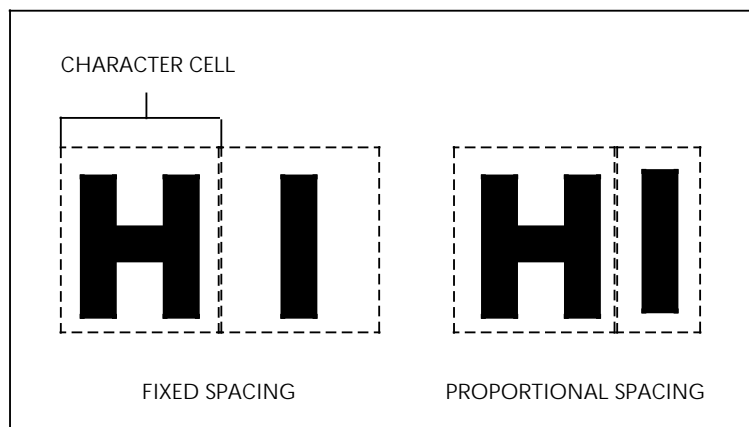
Fonts must reside in the host database and on the printer. Check with your System Administrator to identify the fonts available on your printer. If you specify a font that does not reside in the host database, you receive an error message when you compile the form.

Fixed and proportionally spaced fonts

You can use fixed and proportionally spaced fonts on Xerox LPS printers. Each character occupies an area of the form cell called a character cell. With fixed fonts, all character cells in the set are the same width. With proportional fonts, character cells vary in width.

Usually, you use fixed fonts for variable data and proportional spacing for forms data. Figure 2-5 shows examples of fixed and proportional spacing.

Figure 2-5. **Character spacing**



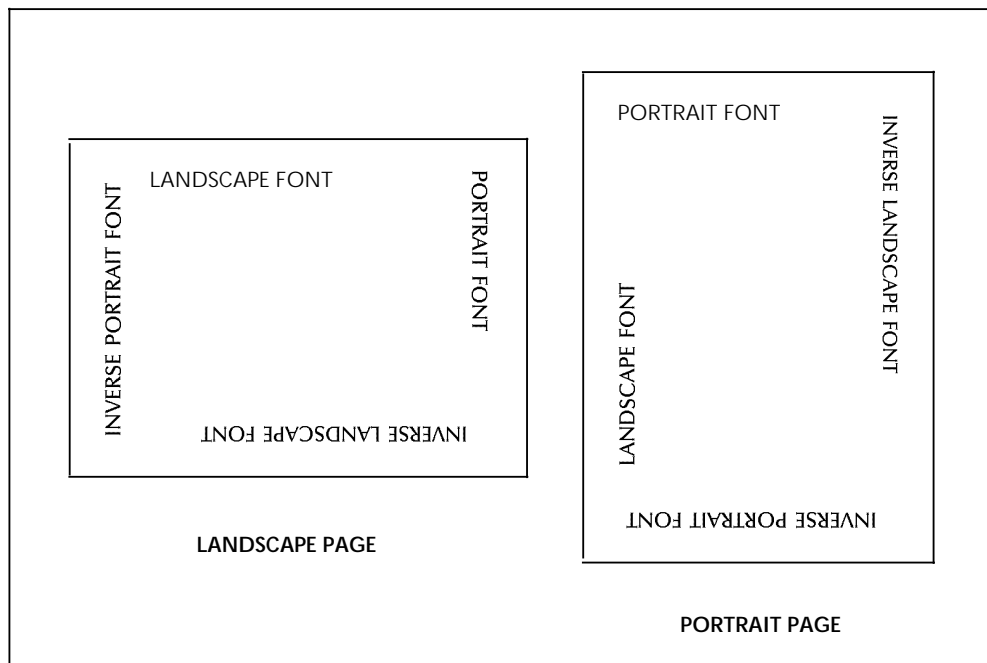
Font orientation

In addition to typeface, style, and size, you can also define fonts by their orientation:

- Landscape
- Portrait
- Inverse landscape
- Inverse portrait.

Figure 2–6 shows the font orientations relative to the physical page.

Figure 2–6. **Font orientation**



Font orientation when using short-edge paper feed

Normally, stock feeds into the printer moving in the direction of the page length. You assign each job a font with a landscape orientation (characters align along the long edge of the page) or a font with a portrait orientation (characters align along the short edge of the page).

Some printers, such as the Xerox 4135 LPS, support paper sizes up to 14.33 by 17 inches by allowing the stock to feed into the printer moving in the direction of the page width. This is the short edge feed (SEF) option. This reversal of the feed edge reverses the font orientation the job must use. When printing a landscape document using the SEF option, specify portrait fonts. When printing a portrait document using the SEF option, specify landscape fonts.

Virtual keyboard and font character maps

You can create virtual keyboard and font character maps and store them in the XPRM database after installation. During the compilation of forms, you can set the compiler to use the character maps to enable keyboard-to-font character mapping. Use the **KEYBOARD** parameter of the **COMPILE** command to specify the keyboard map you want to use. The keyboard characters are mapped to the corresponding graphic representations of the characters in the Xerox printer fonts. For more information on setting the compiler, refer to the "Using the COMPILE command" section in the "Managing and printing sample forms" chapter.

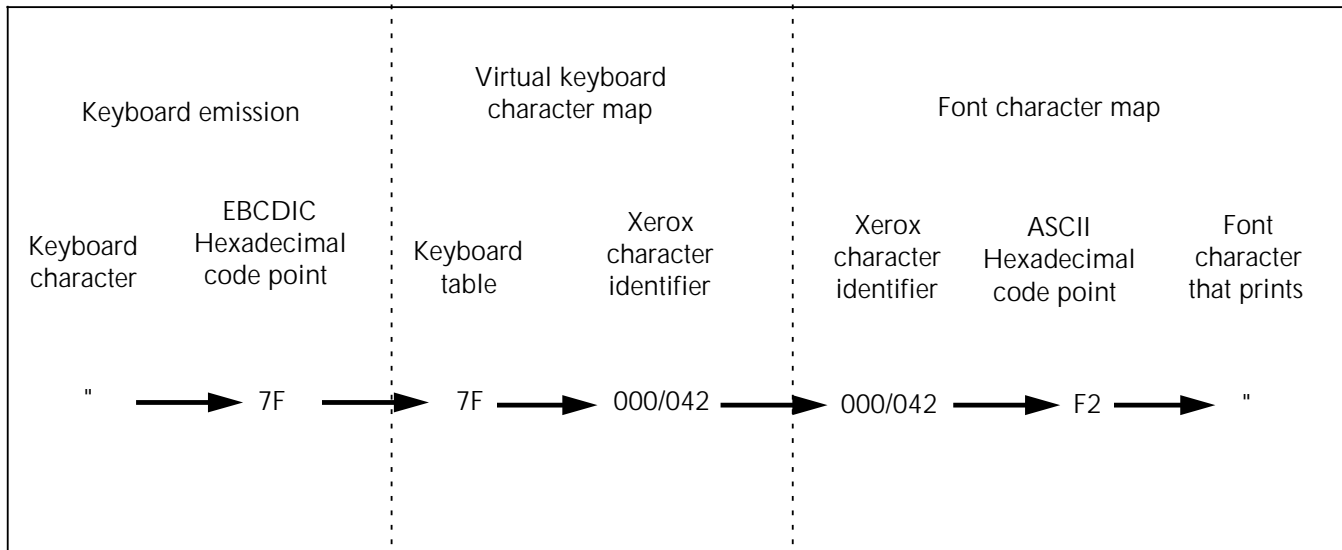
Normally, mapping is not required in the United States, but it is required in the following situations:

- When there are differences between your terminal keyboard configurations
- To accommodate varying character sets
- To accommodate the internal organization of characters within different fonts.

You must use a keyboard character map to access a font character map. In the United States, use the U.S. English keyboard character map.

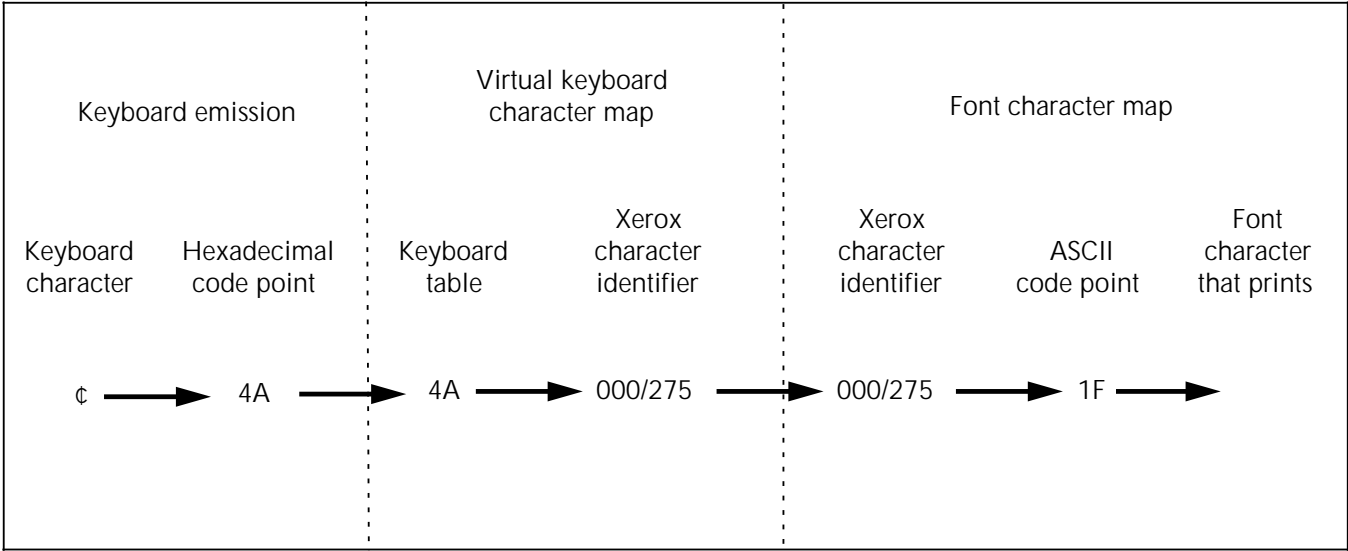
Generally, you map the keyboard character to the corresponding character in each font used in the form. Figure 2-7 shows this process. In the example, the neutral double quote (") on the keyboard is mapped to the neutral double quote in the font.

Figure 2-7. Mapping process



You can also use the mapping process to assign a different character to a key. Figure 2–8 shows you how to map a little used keyboard character, such as the cent sign (¢), to a new font character required in your form, the fraction one-half (½).

Figure 2–8. Mapping to a different font character



You cannot map the following keyboard characters to a different character: A to Z, a to z, and 0 to 9.

Virtual keyboard character maps

When you press a key on the terminal keyboard, it emits a unique code. This special code is referred to as a hexadecimal code point.

A specific hexadecimal code point corresponds to the same character, regardless of the keyboard you are using. For example, hexadecimal code point C1 always corresponds to an uppercase A.

Keyboards may vary from country to country, so there are a few hexadecimal code points that correspond to different characters depending on the keyboard you are using. For example, on a U.S. English keyboard, hexadecimal code point 4A corresponds to the cent symbol (¢); on a UK English keyboard, hexadecimal code point 4A corresponds to the dollar symbol (\$). U.S. English is the default keyboard, but you can customize a virtual keyboard character map for each unique keyboard configuration you need.

A unique Xerox character identifier is assigned to each character in the Xerox centralized and decentralized fonts. The printer uses the Xerox character identifier to locate the graphic representation of the character. The Xerox character identifier matches the hexadecimal code point to the character it represents. For example, the Xerox character identifier 0/101 corresponds to the uppercase A.

For more information on keyboard character maps, see the “Virtual keyboard and font character maps” appendix in the *Xerox Print Resources Manager/Host Forms Description Language 3.2 for IBM MVS Managing Resources* manual.

Font character maps

Once the compiler equates a keyboard code point to a Xerox character identifier, another translation must locate the graphic representation of the character within the font. A font character map identifies the location of each graphic character within a font. The graphic representation is the actual bitmap that prints the character in the form. Usually the character is in ASCII format. For example, hexadecimal code point 41 is the ASCII code for the uppercase A.

Some character sets may not support all font graphic characters. For example, Character Set E does not contain the left brace ({}).

For more information on font character maps, see the "Virtual keyboard and font character maps" appendix in the *Xerox Print Resources Manager/Host Forms Description Language 3.2 for IBM MVS Managing Resources* manual.

Adding color to forms

The Xerox highlight color LPS printers and the Xerox 4700 Color Document Printer allow you to use color to emphasize important parts of your form. With a highlight color printer, you can use different shades of a single color to highlight parts of your form. The 4700 printer uses full color technology, which allows you to choose from a wide selection of colors. The following examples show typical color enhancements for forms:

- Highlighted titles
- Highlighted columns representing totals or important results
- Lines setting off groups of columns or items or used as a border for your form
- Boxes filled with any of the colors and tints available in the ink catalog.

You use the HFDL command statements to control color. For more information, refer to the "Using HFDL commands" chapter.

On a highlight color printer, text, lines, and box borders always print in the primary color of the printer, even if you specify a different color, tint, and so on.

You can send a form using red, blue, or green palettes to either a highlight color printer or a full color printer. To convert red, blue, and green highlight colors to full color versions, recompile the form with `VERSION=XC` on the **COMPILE** command. The full-color palette is available only on the 4700 printer.

Ink color availability and control

When planning colors, you should determine if your company has any regulations and controls for using colors on forms. You may need to use a specific ink list, which contains an ink catalog, palette, and ink for items such as logos and form titles. Contact your System Administrator to determine the available ink colors.

For additional information regarding default ink palettes, catalogs, color models, ink priority and substitutions, and modifying output, refer to the *Xerox LPS PDL/DJDE Reference Manual*.

Choosing paper size

Once you determine the actual page size (physical page) available for your printer, you can specify any size within the physical page size for your form.

Contact your System Administrator to determine available paper sizes.

For a list of paper sizes for the family of Xerox laser printers, refer to the “Supported paper sizes” appendix.

Forms design ruler

Careful spacing and form measurements can help ensure a smooth conversion to an electronic format. As an option, a design ruler can be ordered to measure character and line spacing on your draft form or paper original. Contact your appropriate Xerox customer representative for ordering information.

Ruler scales

The design ruler is approximately 15 inches long, with eight scales, four on each side. The scales are defined as follows:

- | | |
|---------------|---|
| Side 1 | Scale 1 (10/INCH)—Output character spacing. You can also use this scale to measure inches, graduated in tenths.
Scale 2 (6/INCH)—Output line spacing for 6 lines per inch (lpi).
Scale 3 (8/INCH)—Output line spacing for 8 lpi.
Scale 4 (12/INCH)—Output line spacing for 12 characters per inch (cpi) or 12 lpi. |
| Side 2 | Scale 5 (13.6/INCH)—Default output character spacing for 13.6 characters per inch (cpi).
Scale 6 (8.1/INCH)—Default output line spacing for 66 lines per landscape page.
Scale 7 (10.7/INCH)—Output line spacing on for 88 lines per landscape page.
Scale 8 (15/INCH)—Output line spacing for 15 cpi or 15 lpi. |

Suggested uses

You can use the design ruler for character spacing or line spacing. For example, you can use scale 1 for 10 characters per inch (cpi) or 10 lines per inch (lpi). Typically, you use side 1 of the ruler for measuring impact-printed computer output reports in which the character spacing is 10 cpi (scale 1) and line spacing is 6 lpi (scale 2) or 8 lpi (scale 3). You use side 2 for measuring LPS output.

The default spacing for the LPS is 13.6 cpi (scale 5) and 8.1 lpi (scale 6). These defaults allow optimal readability for the standard format of 132 columns by 66 lines.

You use the Host Forms Description Language (HFDL) keyword commands to translate your form design into a format for the host computer. The command set representing your form becomes the Forms Source Language (FSL), which the host compiles and stores. You can download the compiled form to a Xerox LPS or XES printer. This chapter describes the HFDL commands and their functions.

Creating forms with HFDL commands

The HFDL command language syntax is usually the same as the Forms Description Language (FDL) command language syntax. You should be familiar with the FDL commands and how to use them at the printer.

For more information on FDL commands and printer usage, refer to the *Xerox Laser Printing Systems Forms Creation Guide* and the "Troubleshooting syntax errors" section in the *HFDL Release Notes*.

The "Comparison of HFDL and FDL" appendix in this manual compares the commands for HFDL 2.1D, HFDL 3.1, HFDL 3.2, and FDL.

Available specifications

The HFDL commands allow you to specify the following features:

- **Form name** identifies the forms you are developing.
- **Page orientation** specifies a page as portrait or landscape.
- **Font selection** specifies the fonts you want to use.
- **Vertical and horizontal grid positions** specify grid position and position form elements in line printer measurements (line number and character position) or inches, centimeters, dots, and xdots.
- **Highlight and full color** enhance graphs, charts, forms, and other publications.
- **Shading** mixes color with black to produce shades of a primary color (red, green, or blue).
- **Caption position** specifies locations of captions on a page or selects automatic placement within a specified area.
- **Logo and graphic position** specifies the location of selected logos and graphics on a page.

Command format

Each HFDL command consists of a command identifier, parameters, and reserved words associated with the function. The following conventions apply:

- A semicolon (;) terminates a command.
- A comment line begins with the word **COMMENT**, followed by a comment and semicolon (;).
- A blank space separates reserved words and parameters.
- Commas are legal only as shown in the command syntax flow charts.

Command information appears within the first 72 columns of each record. Commands can continue for multiple records; however, there is a limit of 4,096 characters allowed in one text block. Multiple commands can appear in one record if you separate them with semicolons.

The system ignores records containing all blanks. You can use them to separate command statements.

Reserved words

Reserved words appear in bold uppercase text in the command syntax flow charts in this chapter. You can abbreviate many reserved words to the first three characters (except **FMT***n*, where you must specify the integer *n*). Refer to the "Approved abbreviations for reserved words" appendix for the list of abbreviations you can use. Avoid using reserved words as section or form names.

Special characters

Use the following special characters to meet certain requirements within the command statements:

- A single quote (') encloses a text string, for example, 'This is a font count' produces "This is a font count" in the printed version. When you use a reserved word such as "font" in a string, it is not processed as a command.
- A pound symbol (#) within a text string shifts the text between uppercase and lowercase characters, and vice versa. Each string begins with an uppercase character. For example, F#IRST produces "First" in the printed version. To shift back to uppercase characters, you must enter another pound symbol (#) character in the text string. For example, R#EV#A produces "RevA" in the printed version.

You can substitute the pound symbol (#) with any other character by using the SHIFT operand at the COMPILE command. Refer to the "Command Syntax" section in the "Managing and printing sample forms" chapter for information on how to use this operand.

- An exclamation point (!) specifies a character in the font character set, not the keyboard character. Enter an exclamation point (!) and the ASCII hexadecimal code point for the character you want to print. For example, 65!7B

produces 65°. No mapping or translation is done to these characters.

You can substitute the exclamation point (!) with any other character by using the LITERAL operand at the COMPILE command. Refer to the "Command Syntax" section in the "Managing and printing sample forms" chapter for information on how to use this operand.

- Special characters print only if you type them twice in succession. For example, 'IT'S HERE!!' produces "IT'S HERE!" in the printed version.

Understanding the command syntax flow charts

Many HFDL statements offer several options. This chapter represents HFDL command syntax in the form of flow charts, which allow you to visualize the main command statement and the way options branch from that statement.

If you want to enter a command in its short form, without options, follow the main command line in the flow chart until you reach a semicolon. Otherwise, follow the branching below the main command line to include any parameters you want.

The main command line contains the required words and operands. Under the command line, or under a branching line, you can use optional words to clarify the command. Parameters that are vertically stacked (one above the other) under the main command line or under a branching line indicate choices for that particular parameter. Ellipses (...) indicate multiple occurrences of the preceding parameter.

Words, characters, or phrases shown in lowercase are variables. Replace these variables with names or values that you have selected.

Setting up your form

Before you enter commands to describe a form, you enter setup commands to specify the following:

- **FORM** specifies the name and resolution of the form.
- **PAPER** specifies the paper size.
- **LANDSCAPE/PORTRAIT** specifies page orientation and virtual page size.
- **GRID** specifies grid unit and form origin.
- **FONTS** specifies fonts.
- **ICATALOG** and **PALETTE** specifies ink catalog and palette.
- **INK** specifies inks and colors valid only when using Xerox color printers.
- **IRESLT** specifies results when objects imaged with different inks overlap on Xerox highlight color printers only.

The name parameter of the **FORM** command is required. The other commands are optional. If you do not specify the optional

commands, the defaults are used. If you want to include text on the form, you must specify the **FONT** command.

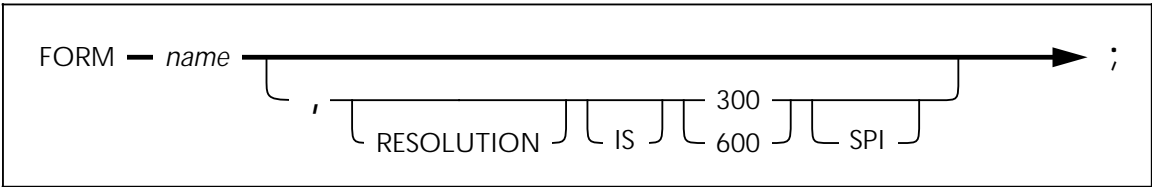
FORM command

You must first specify the **FORM** command when you describe your form. You must also completely contain it in the first record of the form description.

Enter a form name between one and six characters. This is the only required parameter in the **FORM** command. **FORM** must be the first command of the form file and must precede any comments. Once the form is compiled, this name identifies the form stored on the printer.

Figure 3–1 shows the command syntax flow.

Figure 3–1. FORM command syntax



Parameters	<i>name</i> identifies the name you want to use for the form. The form name must be from one to six characters. <i>name</i> is a required parameter, but you can omit the remainder of the FORM command. RESOLUTION: You can specify the resolution (300 or 600 SPI) or leave the resolution blank. IS and SPI are optional. The 600 SPI resolution is only valid on the Xerox 4650 LPS.
Default	300 SPI if you do not specify SPI . There are no defaults for the form <i>name</i> .
Examples	Long form: FORM NEWFOR, RESOLUTION IS 600 SPI; Short form: FORM NEWFOR;

PAPER command

Use the **PAPER** command to specify the paper size for your form. This command replaces the ALIN option (from HFDL version 2.1).

If you specified a predefined form using the **GRID** command, the paper size is determined automatically and you do not need to use the **PAPER** command.

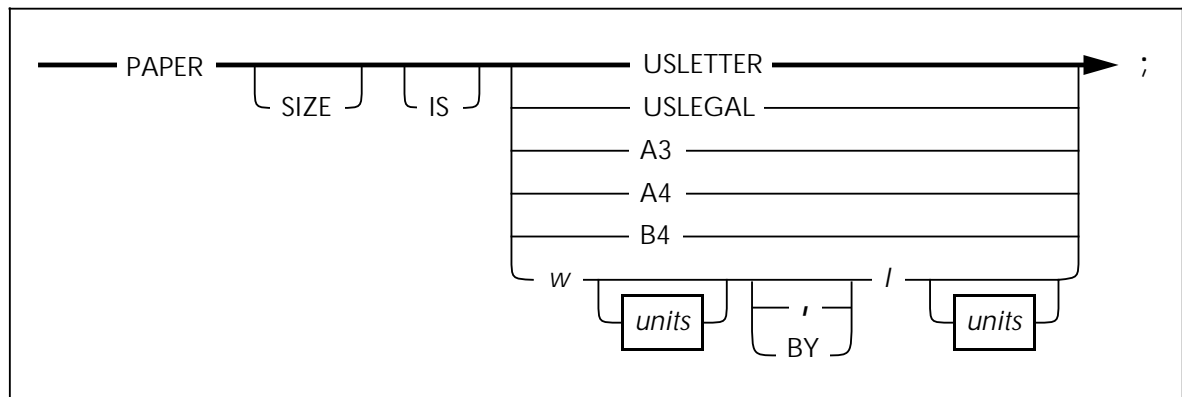
If you use the **PAPER** command to specify a paper size at run time, make sure the paper size matches the size specified in the JDE or JDL for the variable data.

Before printing your form, verify that the printer contains the correct paper size. Use the printer status key on the printer console. An error message displays if a paper size mismatch occurs at run time. The job aborts and your form does not print.

Refer to the output statement **PAPER SIZE** command in the *Xerox LPS PDL/DJDE Reference* manual.

Figure 3–2 shows the command syntax flow.

Figure 3–2. **PAPER** command syntax



Parameters

SIZE and **IS** are optional words you can use for clarity in the long form of the language coding.

USLETTER, **USLEGAL**, **A3**, **A4**, and **B4** are the standard paper sizes you can use with the Xerox LPS.

Width (*w*) and length (*l*) represent the decimal paper size for printers that allow you to print forms in nonstandard paper sizes.

units indicates the unit of measure you want to use when specifying the size of the page. The *units* you can choose are **INCHES**, **CENTIMETERS**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You must express **DOTS** and **XDOTS** in integers.

Refer to the “Supported paper sizes” appendix for more information.

Default

Inches, if you do not specify units.

If you do not specify the **PAPER** command and a predefined format is not defined in the **GRID** command, HFDL assumes **USLETTER**.

Examples

Long form:

PAPER SIZE IS USLETTER;

PAPER SIZE IS 8.5 INCHES BY 11 INCHES;

Short form:

PAPER USLETTER;

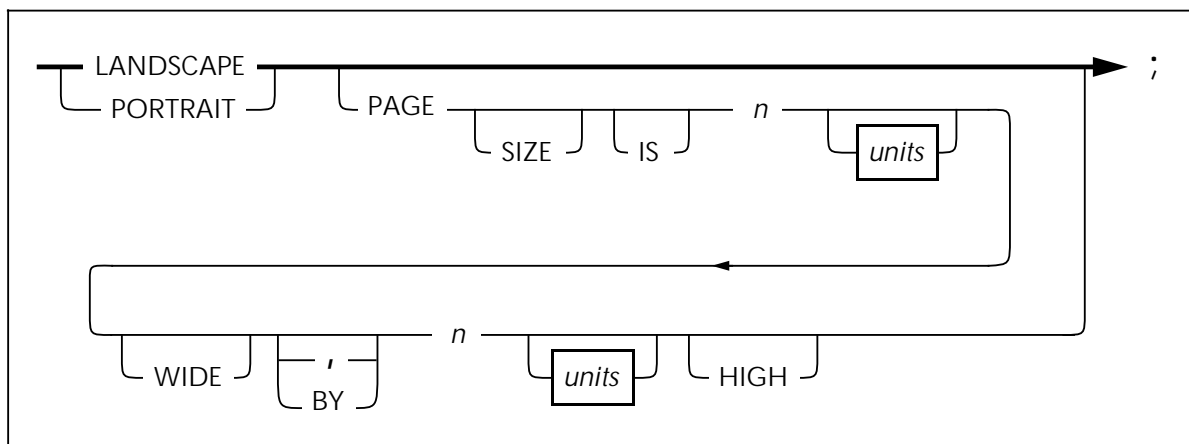
LANDSCAPE/PORTRAIT command

Use the **LANDSCAPE/PORTRAIT** command to specify the orientation of the form, and the size and origin of the virtual page.

If you specified the predefined format in the **GRID** command, the form orientation is established automatically and you do not need to use the **LANDSCAPE/PORTRAIT** command.

Figure 3–3 shows the command syntax flow.

Figure 3–3. **LANDSCAPE/PORTRAIT** command syntax



Parameters

LANDSCAPE specifies the landscape form orientation. If you only specify **LANDSCAPE**, grid units and form origin correspond to the standard grid format (FMT1).

PORTRAIT specifies the portrait form orientation. If you only specify **PORTRAIT**, grid units and form origin correspond to the standard grid format for the portrait page (FMT6), unless you change them with the **GRID** command.

Refer to the “FMT grid format properties” appendix for more information.

PAGE, **SIZE**, and **IS** specifies the width and height of the virtual page.

The value you use to specify the special page size is *n*. You must use positive numbers and you cannot exceed the size specified in the **PAPER SIZE** command. You can use two decimal places when you specify inches. You must express **DOTS** and **XDOTS** as integers.

units indicates the unit of measure you want to use when specifying the size of the virtual page. The *units* you can choose are **INCHES**, **CENTIMETERS**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You must express **DOTS** and **XDOTS** as integers.

Defaults

If you do not use the **LANDSCAPE/PORTRAIT** command, HFDL uses these defaults:

- **LANDSCAPE** (orientation)
- FMT1 (grid units and form origin)
- 8.5 by 11 inches (virtual page)
- **INCHES** (if you do not specify *units*).

Examples

Long form:

PORTRAIT PAGE SIZE IS 8.5 INCHES WIDE BY 11 INCHES HIGH;

Short form:

PORTRAIT;

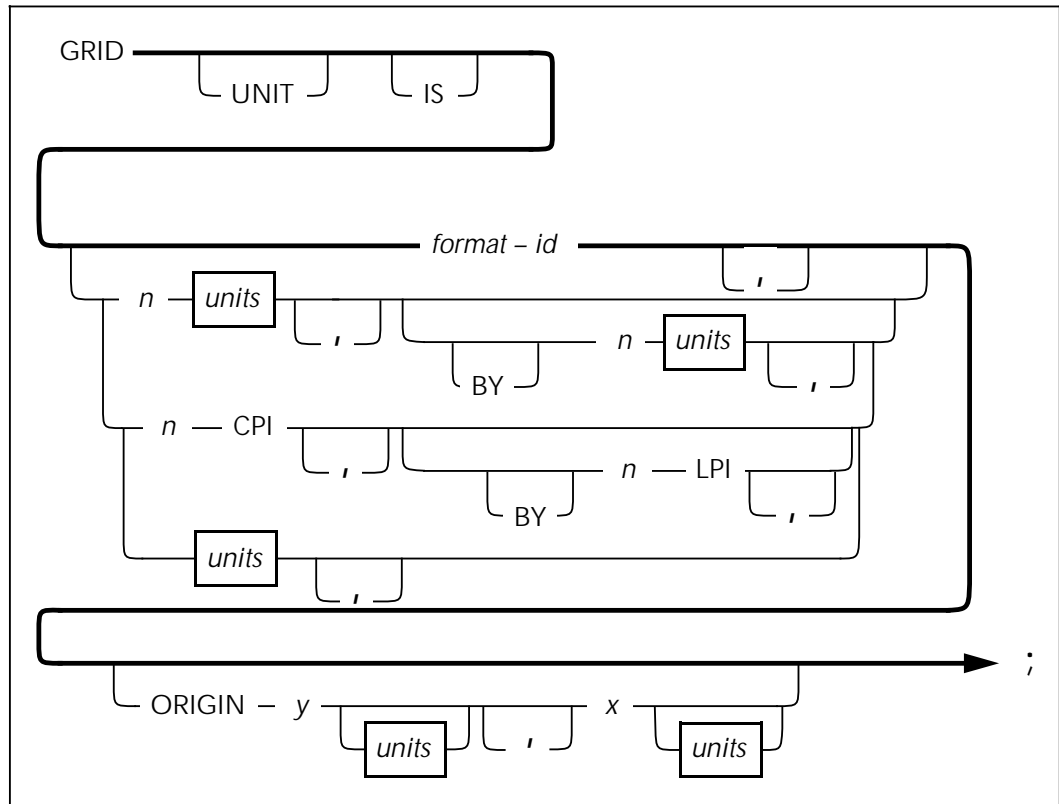
It allows HFDL to default to the FMT6 predefined format.

GRID command

Use the **GRID** command to specify the dimensions of the grid unit and the location of the form origin with respect to the virtual page origin. All HFDL description commands create and place form elements (lines and boxes) relative to the form origin.

Figure 3–4 shows the command syntax flow.

Figure 3–4. **GRID command syntax**



Parameters

UNIT and **IS** are optional words you can use as a lead-in to the *format - id* and *units* identifiers.

The name of the predefined format that you have selected is *format - id*. Refer to the “FMT grid format properties” appendix for more information.

The size of the grid unit is *n*, expressed in positive numbers of units (for example, 6 inches). If you specify **inches** or **centimeters**, the value represents one side of a square grid unit. If you only specify one **DOTS** or **XDOTS** parameter, the grid unit is the same number of dots both horizontally and vertically.

CPI specifies the number of characters per inch for the horizontal size of the grid unit.

LPI specifies the number of lines per inch for the vertical size of the grid unit.

ORIGIN specifies the origin of the form relative to the virtual page origin. **ORIGIN** establishes the 0.0 X and Y coordinates.

The **GRID** sizes must fall into the boundaries of the physical page. The smallest grid allowed is 1 **XDOT** by 1 **XDOT**. The largest grid allowed is 1 physical page by 1 physical page.

Y specifies the form origin offset downward from the top-left corner of the virtual page.

X specifies the form origin offset to the right of the left edge of the virtual page.

units indicates the unit of measure you want when specifying the offset origin. The *units* are **INCHES**, **CENTIMETERS**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You must express **DOTS** and **XDOTS** in integers.

Defaults Predefined format FMT1, if you do not specify the **GRID** or **LANDSCAPE/PORTRAIT** commands.

Predefined format FMT6, if you specify **PORTRAIT** (**LANDSCAPE/PORTRAIT** command), but not **GRID**. If you specify a predefined format, all **GRID** format parameters are automatically provided, unless you change them with the **ORIGIN** parameters.

INCHES, if you do not specify *units*.

A value of 1, if you omit an *n* value when you specify **DOTS** or **XDOTS**.

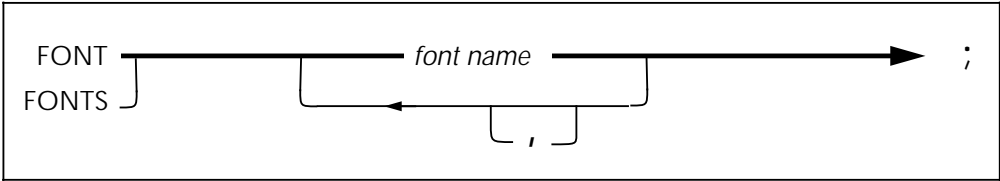
Examples Long form:
GRID UNIT IS 12 CPI BY 8 LPI ORIGIN 0.18 INCH 0.66 INCH;
Short form:
GRID FMT1;

FONT command

You cannot specify any text in your form until you specify the fonts you are going to use with the **FONT** command. Check with your System Administrator for the fonts available for your forms.

Figure 3-5 shows the command syntax flow.

Figure 3-5. **FONT command syntax**



Parameters *Font name* identifies the standard or custom font for your form. *Font name* is a maximum 6-character string for a centralized font card and a maximum 20-character string for a decentralized font card. Use a blank space or a comma to separate multiple font names. There is a maximum of 32 fonts per form.

Do not confuse font names with the **FONT** command. You can specify up to 32 font names in a HFDL command structure, such as **font1**, **font2**, through **font32**. You can specify only one **FONT** command in an HFDL command structure.

When you compile a form in LPS format, fonts convert to uppercase. When you compile a form in XES format, fonts do not convert. The system accepts them in uppercase and lowercase. Refer to the “Managing and printing sample forms” chapter.

Defaults None.

Examples Long form:
FONT Font1;
There is no short form.

ICATALOG command

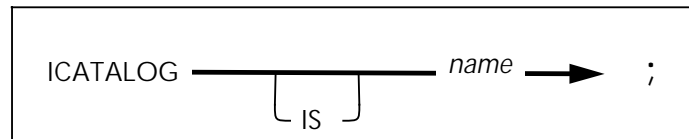
Use the **ICATALOG** command to specify the ink catalog to be used on highlight and full color printers.

The ink catalog is a file that contains color information. It resides on highlight color printers and in the XPRM database. You can list all the ink catalogs stored in your XPRM database by using the LIST COLORMAP command. Refer to the “Resource management” chapter in the *XPRM/HFDL 3.2 for IBM MVS Managing Resources* manual.

Full color printers require the ink catalog name. If you do not specify a catalog for a highlight color printer, the printer uses its default catalog.

Figure 3–6 shows the command syntax flow.

Figure 3–6. **ICATALOG command syntax**



Parameter *name* represents the ink catalog to use for color information. Your ink catalog name must not exceed six alphanumeric characters. You cannot use spaces.

Default Printer default catalog and palette, if you do not specify the **ICATALOG** (highlight color only).

Examples Long form:
ICATALOG IS CAT1;
Short form:
ICATALOG CAT1;

PALETTE command

Use the **PALETTE** command to specify the ink palette to use for color information on highlight and full color printers. The ink palette is a subset of inks in the ink catalog.

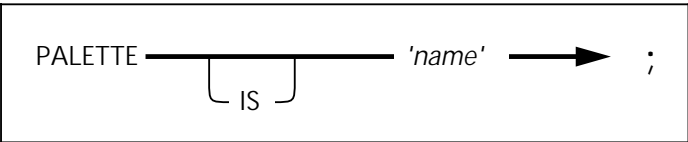
Full color printers require the palette name. If you do not specify a palette for a highlight color printer, the printer uses its default palette.

You can list all the palette names stored in your XPRM database by using the LIST COLORMAP command. Refer to the “Resource

management” chapter in the *XPRM/HFDL 3.2 for IBM MVS Managing Resources* manual.

Figure 3–7 shows the command syntax flow.

Figure 3–7. **PALETTE command syntax**



- Parameter** *name* represents the ink palette you use for ink information. Your palette name must not exceed 32 alphanumeric characters, including spaces.
- Default** Printer default catalog and palette, if you do not specify the **PALETTE** (highlight color only).
- Examples** Long form:
PALETTE IS 'PAL1';
Short form:
PALETTE 'PAL1';

INK command

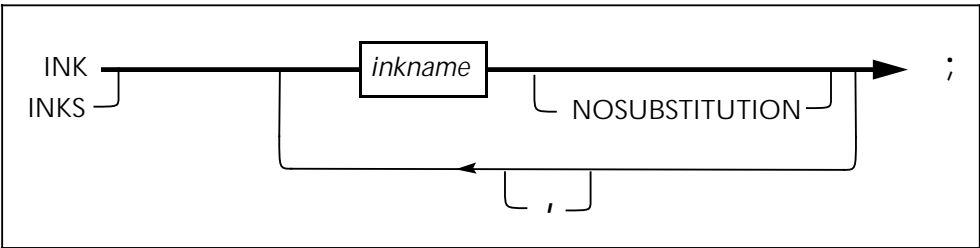
The **INK** command is only valid on Xerox highlight color and full color printers. This command specifies all of the inks and colors you use when you create a form on Xerox highlight color and full color printers. The inks you specify with the **INK** command are referenced as an index number when you use the **LINE**, **TEXT** and **BOX** commands. They are not referenced as their identification name.

The **ICATALOG** and **PALETTE** commands must precede the **INK** command, as the printer uses the palette and catalog specified in these commands. If you do not specify a catalog or palette for a highlight color form, the printer uses its default catalog and palette.

Ink index numbers begin with 1 for the first ink you specify, 2 for the second, and so forth. This numbered list of ink names is the ink list you use for your form. The ink you selected for a specific description command is identified by the associated ink number found in the ink list. This command follows the **FONT** command. The maximum number of ink names that can be defined is 255.

Figure 3–8 shows the command syntax flow.

Figure 3–8. **INK command syntax**



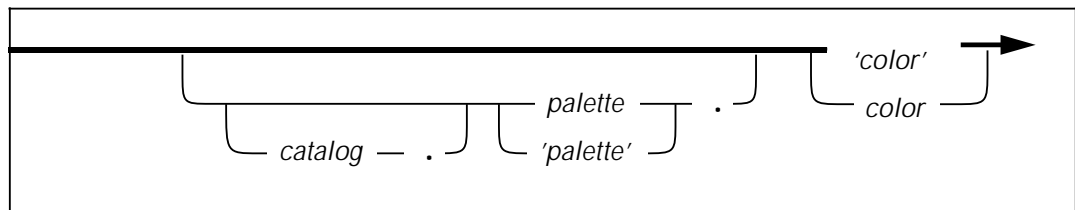
Parameters	<p><i>Inkname</i> identifies all of the inks you use to create your form. If you use a reserved word such as BLACK as an ink name, you must enclose it in single quotes: 'BLACK'. You must also enclose ink names with embedded blanks in single quotes.</p> <p>NOSUBSTITUTION specifies that ink substitutes are not allowed for this ink. If you do not specify this parameter, the printer attempts to substitute another ink. Full color printers ignore this parameter.</p>
Default	The printer default ink, if you do not specify the INK (highlight color only).
Example	<p>Long form:</p> <p>INKS RED, PINK;</p> <p>There is no short form.</p>

Inkname format

This format describes a variety of ways you can specify an ink. Use this statement to describe forms you are compiling for Xerox highlight color or full color printers.

Figure 3–9 shows the inkname syntax.

Figure 3–9. Inkname syntax



Once you establish an **INK** command parameter (catalog, palette, or ink color), it remains in effect until you change it or another parameter overrides it. A color or palette name containing one or more blank spaces requires quotes around the name.

Defaults	<p>The user catalog specified in the ICATALOG command, if you do not specify a catalog in the INK command.</p> <p>System default catalog, if you do not specify a default catalog (highlight color only).</p> <p>The palette you specify in the PALETTE command, if you do not specify a palette in the INK command.</p> <p>System default palette, if you do not specify a default palette (highlight color only).</p> <p>When the ink color begins with a digit, you must specify it within quotes.</p>
-----------------	---

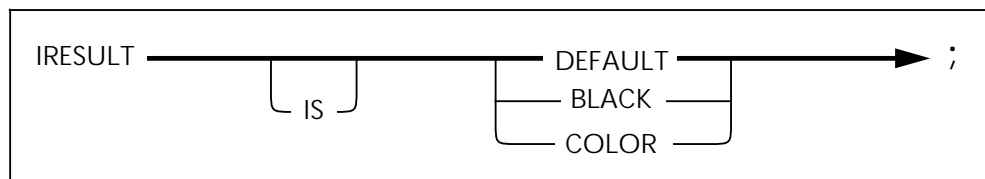
Examples	<p>This example shows you how to use the single quotes in describing specific inks:</p> <p>RED or 'RED'</p> <p>PAL1.'RED', or 'PAL1'.'RED', or PAL1.RED</p> <p>CAT1.PAL1.'RED', or CAT1.'PAL1'.'RED', or CAT1.'PAL1'.RED, or CAT1.PAL1.RED</p>
-----------------	--

IRESULT command

The color result you receive from overlapping objects is either black or color for the entire page. The **IRESULT** command allows you to specify the result when objects imaged with different inks overlap. The **IRESULT** command appears in the HFDL, FDL, and PDL OUTPUT statements and DJDE. You can use the **IRESULT** command only on a Xerox highlight color printer.

Figure 3–10 shows the command syntax flow.

Figure 3–10. **IRESULT** command syntax



Parameters

DEFAULT uses the printer default **IRESULT**, which may be **BLACK** or **COLOR**.

BLACK specifies a black pixel to print when you place two objects with different inks that overlap, and black and highlight pixels coincide.

COLOR specifies a color pixel to print when you place two objects with different inks that overlap, and black and highlight pixels coincide.

Default

DEFAULT, if you do not enter the command.

Examples

Long form:

IRESULT IS BLACK;

Short form:

IRESULT COLOR;



Because the **IRESULT** command is a highlight color command, the Xerox 4700 printer ignores it.

Description commands

Once you complete the setup commands, you can use the following commands in any sequence to enter descriptive data:

- LINE
- BOX
- TEXT AT
- TEXT IN BOX
- LOGO
- GRAPHIC
- SECTION
- DO SECTION
- COLUMN
- INTERWORD
- DOT LEADER
- LET

- COMMENT
- END.

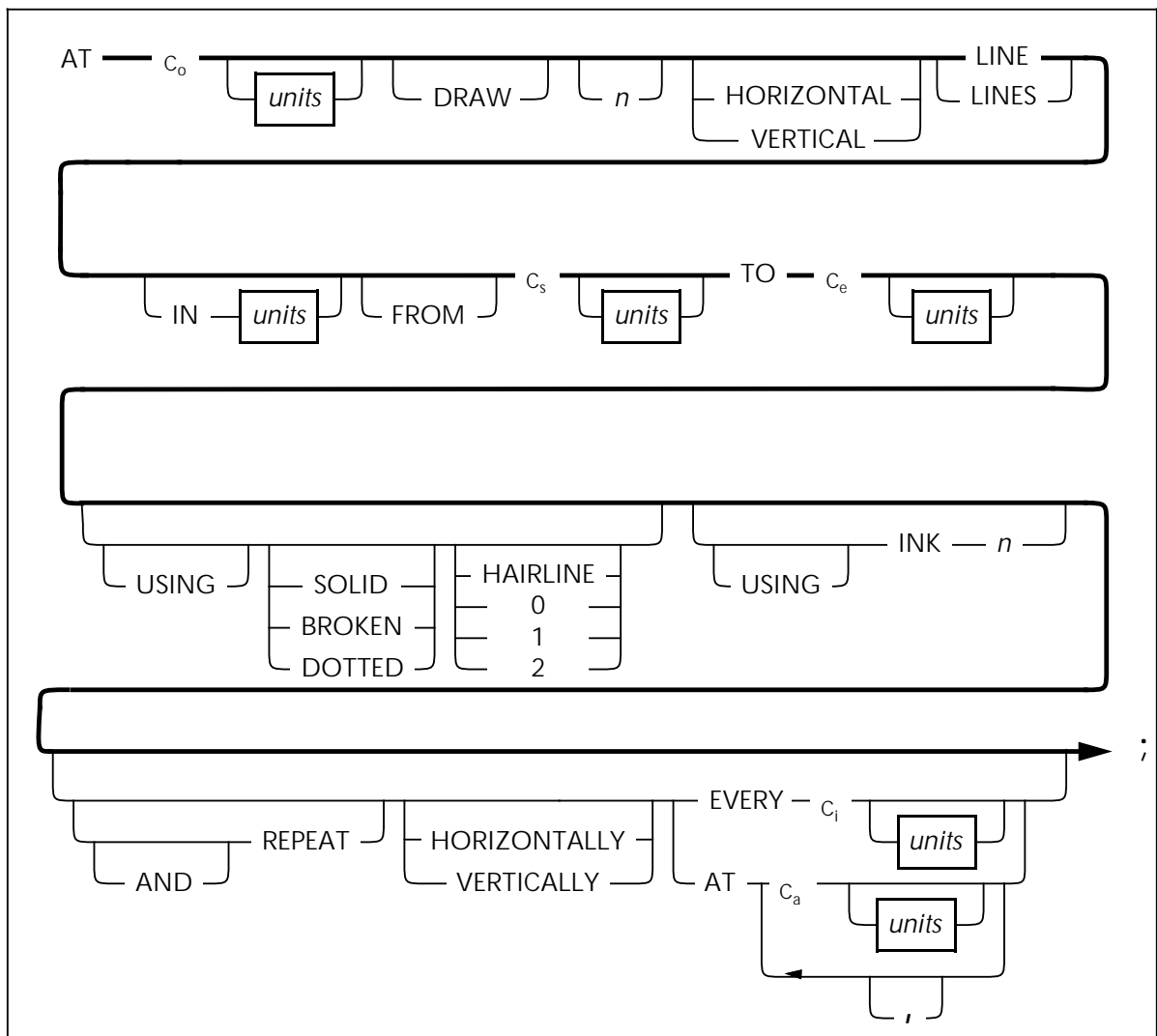
You must place coordinates within the boundaries of the physical page. If coordinates stay within the physical page, but extend beyond the virtual page, HFDL issues a warning message. The size of the lines table, currently set at 1,500, determines the maximum number of lines and boxes. Text must fit on the physical page. You can enter up to 4,096 characters in a text block.

LINE command

Use the **LINE** command to specify one or more lines in the same direction on your form.

Figure 3-11 shows the **LINE** command syntax.

Figure 3-11. **LINE** command syntax



Parameters

C_o allows you to specify a line origin location on the form grid, measured from the origin **AT** location, the point at which the line begins. If the line is horizontal, **C_o** represents the origin in the vertical direction (y-axis coordinate); if the line is vertical, **C_o**

represents the origin in the horizontal direction (x-axis coordinate).

units indicates the unit of measure for specifying the line origin. The *units* are **INCHES**, **CENTIMETERS**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You must express **DOTS** and **XDOTS** in integers.

DRAW is an optional reserved word.

The number of lines you want drawn on your form is *n*.

HORIZONTAL or **VERTICAL** specifies the direction of your lines.

LINES specifies that this reserved word is required.

IN *units* is a shorthand method of specifying units for the entire command. **IN** *units* overrides all other units used. For example, if you want to draw a line using inches but specify grid FMT1, **IN** *units* uses inches as the grid for all units in the specific **LINE** statement.

FROM, **TO** specifies the coordinate at the coordinate start (**C_s**) point and the coordinate end (**C_e**) point.

SOLID, **BROKEN**, or **DOTTED** specifies the type of line you want to draw.

HAIRLINE, **0** point (invisible), **1** point, or **2** points specifies the thickness of the line you want to draw.

INK *n* specifies the ink color index from the ink list for a Xerox highlight or full color LPS printer.

REPEAT specifies that you want a selected line repeated according to the following commands.

HORIZONTALLY or **VERTICALLY** specifies the orientation of the lines you want repeated.

EVERY **C_i** indicates that you want a line repeated at the increment (**C_i**) you have specified.

AT **C_a** indicates that you want the line repeated at the absolute coordinate (**C_a**) you have specified.

Defaults **HORIZONTAL**, if you do not specify a line orientation.

SOLID, if you do not specify a line type.

1, if you do not specify a line weight.

VERTICALLY, if you do not specify the **REPEAT** orientation.

Examples Long form:

AT 3 INCHES DRAW 5 HORIZONTAL LINES IN INCHES FROM 4 INCHES TO 10 INCHES USING SOLID 1 USING INK 3 AND REPEAT HORIZONTALLY EVERY 0.50 INCHES;

Short form:

AT 3 5 LINES 4 TO 10 REPEAT EVERY 0.50;

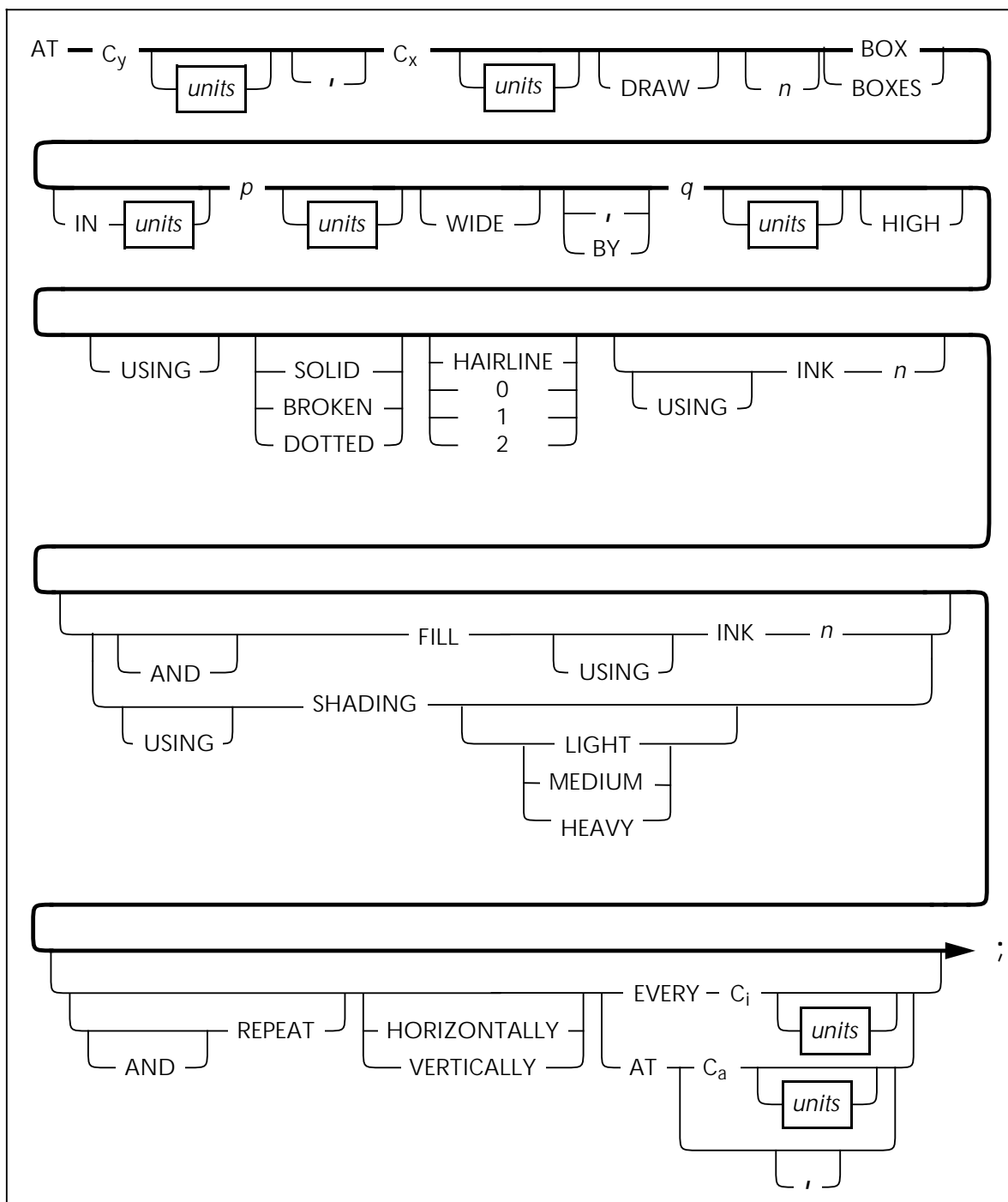
BOX command

Use the **BOX** command to describe the size and shape of a box on your form. Similar to lines, you can repeat boxes at fixed intervals or at specified locations. The origin of the box is its upper left corner at the midpoint of the thickness of the outline. If you are using the 4850 HighLight Color or 4700 LPS printer,

the **BOX** command allows you to outline the frame of the box in color.

Figure 3-12 shows the **BOX** command syntax flow.

Figure 3-12. **BOX** command syntax



Parameters

C_y specifies the y-axis coordinate for the box origin location.

C_x specifies the x-axis coordinate for the box origin location.

units indicates the unit of measure for specifying the box origin. The *units* are **INCHES**, **CENTIMETERS**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You must express **DOTS** and **XDOTS** in integers.

The number of boxes you want to draw is *n*.

IN *units* is a shorthand method of specifying units for the entire command. **IN** *units* overrides all other units used. For example, if you want to draw a box using inches but specify grid FMT1, **IN INCHES** uses inches as the grid for all units in the specific **BOX** statement.

The width of the box you have specified is *p*.

The height of the box you have specified is *q*.

SOLID, **BROKEN**, or **DOTTED** specifies the type of line.

HAIRLINE, **0** point (invisible), **1** point, or **2** points specifies the thickness of the line for your box.

INK *n* specifies the **INK** number you selected from the ink list to draw the line in color for use on a Xerox highlight or full color LPS printer. You may specify the same, or different, ink numbers for the outline of the box.

FILL and **INK** *n* specifies the ink you want to use to fill a box. The box fill can be a color or a tint of color. The ink you specify must be defined in the ink list of the **INK** command.

SHADING specifies shadowing: **LIGHT**, **MEDIUM**, or **HEAVY**.

REPEAT specifies that you want a box repeated according to the following commands.

HORIZONTALLY or **VERTICALLY** specifies the orientation of the repeated boxes.

EVERY **C_i** indicates that you want a box repeated at the increment (**C_i**) you have specified.

AT **C_a** indicates the absolute x or y coordinate where you want the box repeated in a horizontal or vertical direction.

Defaults

SOLID, if you do not specify the line type.

1, if you do not specify the line weight or shading.

0, if you do not specify the line weight but do specify shading.

MEDIUM, if you do not specify the degree of **SHADING**.

HORIZONTALLY, if you do not specify the **REPEAT** orientation.



When you specify a box and shade with one command and do not specify a line type for your box, your box will be shaded but you will not have a line around it.

Examples

Long form:

AT 4,6 DRAW 4 BOXES 14 WIDE BY 4 HIGH USING SOLID HAIRLINE INK 2 AND REPEAT EVERY 5;

Short form:

AT 4,6 BOX 14 WIDE BY 4 HIGH;

TEXT AT command

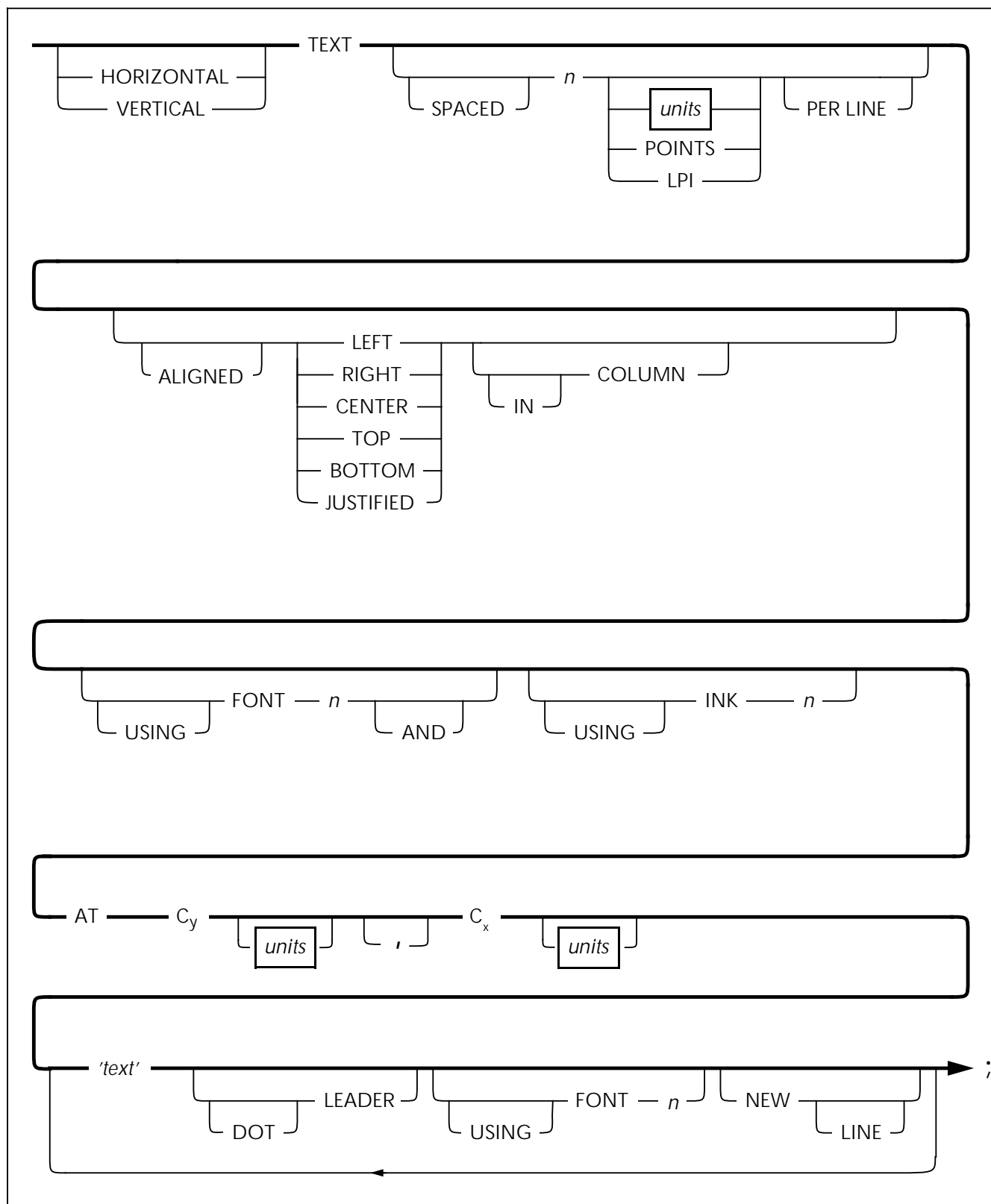
Use the **TEXT AT** command to incorporate text into your form. To include text in a form, you must specify at least one font. You use the **FONT** setup command to specify the font, and you can only specify one **FONT** command. The fonts you specify with the **FONT** command are referenced as an index number. They are not referenced as their font name. Font index numbers begin with 1 for the first font you specify, 2 for the second, and so on.

The number of fonts you can specify for a form depends on the point size, how many fonts you use, and the size of the font memory of the printer. Once you specify a font index number in a **TEXT** command, it remains in effect until you specify a new font index.

When you use multiple text segments within a **TEXT AT** command and they reference different font indexes, the last font you use remains in effect until you specify a new font index.

Figure 3-13 shows the **TEXT AT** command syntax flow.

Figure 3-13. **TEXT AT** command syntax



Parameters

HORIZONTAL or **VERTICAL** specifies the direction of the text relative to the page orientation.

SPACED defines the amount of vertical space (in *n units*) a line of text occupies. If you specify a value (not zero), it replaces the line spacing value of the font being used for this **TEXT AT** command. An **LPI** value specifies line height in terms of lines per inch. All other specifications are actual line height measurements. The *n units* value must be positive. The units are **INCHES**, **CENTIMETERS**, **CPI**, **LPI**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You must express **DOTS** and **XDOTS** in integers. **POINTS** specifies that you want the **TEXT SPACED** in *n POINTS* (1/72 inch).

ALIGNED specifies the text justification alignment (**LEFT**, **RIGHT**, **CENTER**, **TOP**, **BOTTOM**, **JUSTIFIED**). See the *Xerox Laser Printing Systems Forms Creation Guide* for alignment examples. You can center multiple horizontal strings beneath each other or right- or left-justify them. You can center multiple vertical strings alongside each other or align them on the top or bottom.

The **JUSTIFIED** alignment mode allows you to justify text. In simple justification (with no **COLUMN**), you must specify where the lines break. You can justify lines by finding the longest line in a block of lines, then justify other text strings to this length. This method of justification is best for justifying small blocks of text or when you must justify the last line of a paragraph. You can use the **JUSTIFIED** parameter with the **INTERWORD SPACE** and the **COLUMN SIZE IS** commands. In this mode, text begins at the coordinates you specify and extends to the right column boundary. Line breaks occur automatically. Enter text in one continuous string without line breaks.

Example

JUSTIFIED parameters:

```
TEXT JUSTIFIED AT 1,1 'This is a sample'
                        'of justified text' 'in block format.';
```

This produces the following text on the form:

```
This is a sample
of justified text
in block format.
```

You specify **IN COLUMN** for any alignment mode and use it with the **COLUMN SIZE IS** command. Line breaks occur automatically when you use **TEXT ALIGNED IN COLUMN** or **TEXT JUSTIFIED IN COLUMN**. You can enter a complete paragraph of text in one continuous string. A quote indicates the end of a paragraph rather than a line break. The system ignores multiple line breaks. Columns do not have a fixed location on the page. You place them relative to the **TEXT** command you specify. The horizontal coordinate you specify sets the left column boundary. You set the right column boundary to the sum of the column width and the left column boundary.

You can include a maximum of 4096 characters in one continuous text string.

Example

Text justified in a column:

COLUMN 1 INCH WIDE;

TEXT JUSTIFIED IN COLUMN AT 1,1

'four four four five five five six six six'

This produces the following text on the form:

```
four four four
five five five
six six six
```

FONT *n* specifies the index number of the font you selected. Refer to the **FONT** command for additional information.

INK *n* specifies the index number of the ink color you selected from the ink list to use with a Xerox highlight or full color LPS printer. The specified **INK** is in effect for this command and not for subsequent **TEXT** commands.

C_y defines the coordinate you selected on the y-axis where the text origin is located.

C_x defines the coordinate you selected on the x-axis where the text origin is located.

'**text**' specifies a text string that appears exactly as you enter it (except for special characters).

DOT LEADER specifies dot leaders you have selected to place between text or at a column boundary. Dot leaders extend the full width of the column you specify in the **COLUMN** command, unless interrupted by a '**text**' string. You can place dot leaders between two text strings and a right boundary previously defined by you in the **COLUMN** command. If you place dot leaders between two text strings, the right most text string aligns with the right column boundary.

Do not use **DOT LEADER** in the **JUSTIFIED** mode. **DOT LEADER** overrides any alignment mode and places the text flush with the column boundaries.

USING FONT *n* specifies a change of font for the next body of text which is not the dot leader string. When you specify dot leaders and a change of fonts together, the **DOT LEADER** keywords must precede the **FONT** *n* keywords.

NEW LINE specifies a new line to follow a text string. You can only use the **NEW LINE** keyword once between text strings.

NEW LINE is similar to a blank text string (' '). If you want to use multiple blank lines between text strings, use multiple blank text strings (' ' ' ' ' '), not **NEW LINE**. **NEW LINE** is ignored if it is not used with **DOT LEADER** or **FONT** *n*.

Defaults

HORIZONTAL and **CENTER**, if you do not specify the orientation and alignment for the **TEXT AT** command.

Use the font index of the last **TEXT** command, if **USING FONT** is not entered. If you omit **FONT** in all **TEXT** commands, FONT1 from the FONT list is used.

DOTS, if you do not specify *units*.

If a **TEXT** command containing a dot leader is not preceded by a **COLUMN** command, the system uses a default column width of 6.5 inches (portrait) or 9.5 inches (landscape).

Examples

Single dots repeated, if you do not use a **DOT LEADER** command to specify the **DOT LEADER** text string.

The following is an example of a **DOT LEADER** used between two text strings:

TEXT AT 1,1 'John Doe' DOT LEADER '987-6543';

This produces the following text on the form:

John Doe.....987-6543

The following is an example of a **DOT LEADER** following a text string and ending at the column boundary:

TEXT AT 1,1 'Sales Tax' DOT LEADER NEW LINE 'Total Now Due' DOT LEADER;

This produces the following text on the form:

Sales Tax

Total Now Due.....

The following is an example of the long form of the **TEXT AT** command:

HORIZONTAL TEXT SPACED 12 POINTS PER LINE ALIGNED CENTER IN COLUMN USING FONT 1 AND USING INK 1 AT 1,1 'Phone List' NEW LINE 'John Doe' DOT LEADER '987-6543';

The following is an example of the short form:

TEXT SPACED 12 POINTS PER LINE ALI CEN INK 1 AT 1,1 'Phone List' NEW LINE 'John Doe' DOT LEA '987-6543';

This produces the following text on the form:

Phone List

John Doe987-6543

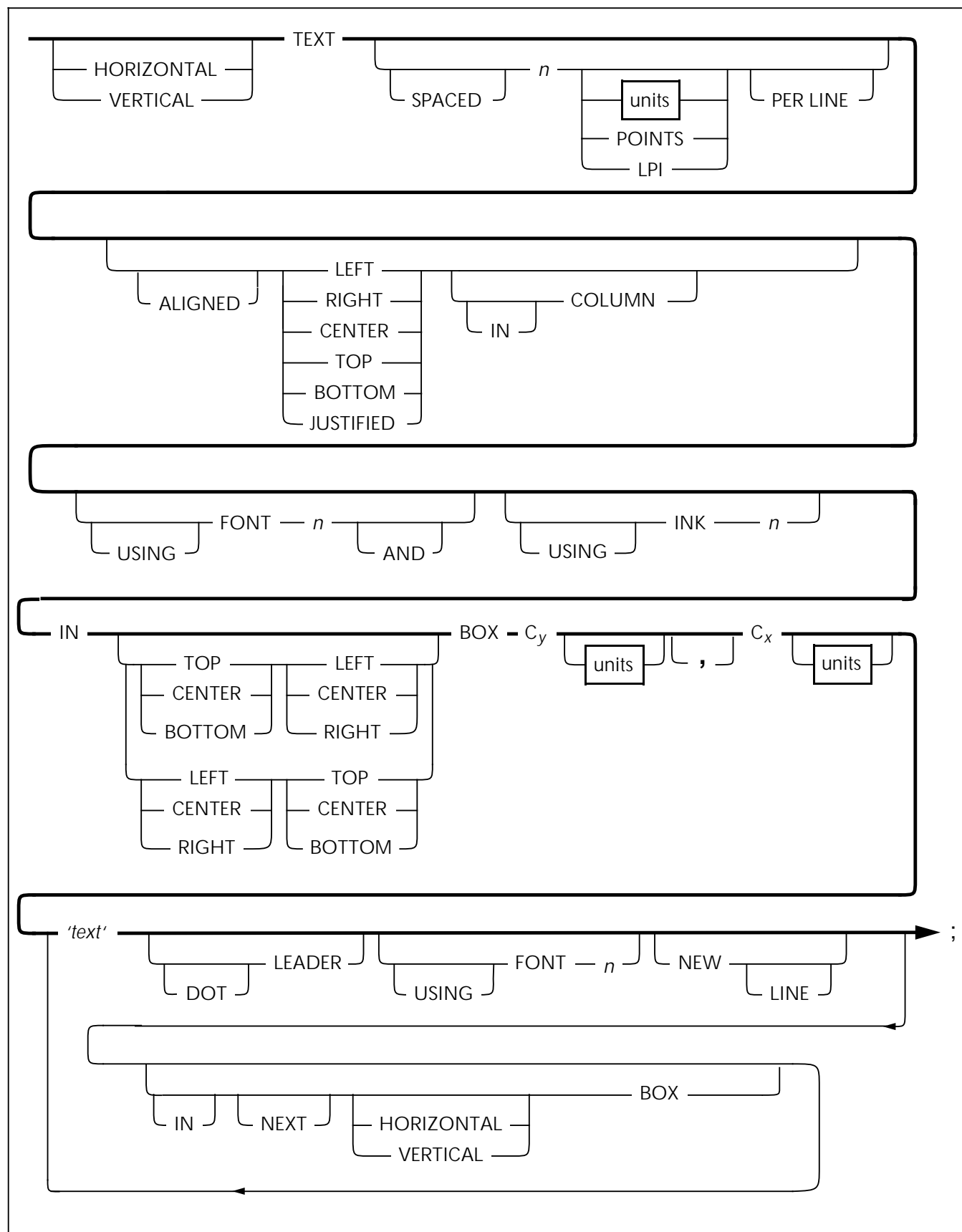
TEXT IN BOX command

Use the **TEXT IN BOX** command to incorporate text into a box on your form. The command syntax format designates the proper type and placement of text in a box according to the planned form. This command format is similar to the **TEXT AT** command, except it includes those parameters necessary to designate the placement of text in a box.

You must define a box with matching coordinate values before you can reference it in a **TEXT IN BOX** command. You can create boxes and adjoining boxes with **BOX** commands, **LINE** commands, or both.

Figure 3-14 shows the **TEXT IN BOX** command syntax flow.

Figure 3-14. **TEXT IN BOX** command syntax



Parameters

HORIZONTAL or **VERTICAL** specifies the text orientation of the text within the box. Do not confuse **TEXT** alignment with **TEXT** placement in the box. You can align horizontal text **LEFT** or **RIGHT**. You can align vertical text **TOP** or **BOTTOM**.

SPACED defines the amount of vertical space (in *n units*) you want a line of text to occupy. If you specify a value (not 0), it replaces the line spacing value of the font being used for this **TEXT AT** command. All specifications except **LPI** are actual line height measurements. An **LPI** value specifies line height in lines per inch. The *n units* value you specify must be positive. The *units* are **INCHES**, **CENTIMETERS**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You can express **DOTS** and **XDOTS** only in integers. **POINTS** specifies that you want the **TEXT SPACED** in *n POINTS* (1/72 inch).

ALIGNED specifies that you want the text aligned **LEFT**, **RIGHT**, **CENTER**, **TOP**, **BOTTOM**, **JUSTIFIED**. See the *Xerox Laser Printing Systems Forms Creation Guide* for alignment examples.

The **ALIGNED** and **JUSTIFIED** commands including **IN COLUMN**, are not recommended with the **TEXT IN BOX** command. These are more appropriate with the **TEXT AT** command. If you want to use **JUSTIFIED IN COLUMN**, refer to the “**TEXT AT** command” section for a description of how to use these parameters.

FONT *n* specifies the index number of the font you selected. Refer to the “**FONT** command” section for additional information.

INK *n* specifies the index number of the ink color you selected from the ink list to use with a Xerox highlight or full color LPS printer. The specified **INK** is in effect for this command and not for subsequent **TEXT IN BOX** commands.

IN specifies the position in a box where you want to place the text. There are nine positions as follows:

TOP LEFT	TOP CENTER	TOP RIGHT
LEFT CENTER	CENTER CENTER	RIGHT CENTER
BOTTOM LEFT	BOTTOM CENTER	BOTTOM RIGHT

C_y defines the coordinate of the box origin on the y axis.

C_x defines the coordinate of the box origin on the x axis.

'text' specifies a text string that displays on the form exactly as you enter it (except for special characters).

DOT LEADER specifies dot leaders you have selected to place between text or to a column boundary. Dot leaders extend the full width of the column you specify in the **COLUMN** command, unless interrupted by a **'text'** string. You can place dot leaders between two text strings and a right boundary previously defined in the **COLUMN** command. If you place dot leaders between two text strings, the right most text string aligns with the right column boundary.

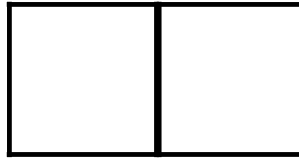
Do not use **DOT LEADER** in the **JUSTIFIED** mode. **DOT LEADER** overrides any alignment mode and places the text flush with the column boundaries.

USING FONT *n* specifies a change of font for the next body of text which is not the dot leader string. When you specify dot leaders and a change of fonts together, the **DOT LEADER** keywords must precede the **FONT** *n* keywords.

NEW LINE specifies a new line to follow a text string. You can only use the **NEW LINE** keyword once between text strings. **NEW LINE** is similar to a blank text string (' '). If you want to use multiple blank lines between text strings, use multiple blank text strings (' ' ' ' ' '), not **NEW LINE**. **NEW LINE** is ignored if it is not used with **DOT LEADER** or **FONT** *n*.

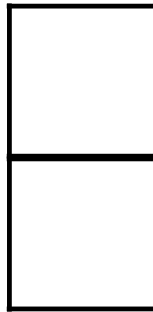
IN NEXT HORIZONTAL or **VERTICAL BOX** specifies the text to place in an adjacent box (horizontally or vertically). Adjacent boxes must have a common, congruent side. For example, **IN NEXT HORIZONTAL BOX...** produces the results shown in Figure 3-15.

Figure 3-15. **IN NEXT HORIZONTAL BOX**



IN NEXT VERTICAL BOX... produces the results shown in Figure 3-16.

Figure 3-16. **IN NEXT VERTICAL BOX**



Defaults

HORIZONTAL, if you do not specify the box orientation. The same applies for **[IN] [NEXT] [HORIZONTAL/VERTICAL] [BOX...]**;

The font index of the last **FONT** command, if **USING FONT** is not entered. If you omit **FONT** in all text commands, then **FONT 1** from the **FONT** list is used.

DOTS, if you do not specify *units*.

CENTER, if you do not specify the alignment. The same applies for the **IN [TOP,BOTTOM] BOX....** for text placement within a box.

Examples

Long form:

HORIZONTAL TEXT SPACED 12 POINTS PER LINE USING FONT 1
AND USING INK 1 IN CENTER CENTER BOX 4 3 'FIRST'
'NATIONAL' 'TRUST' IN NEXT HORIZONTAL BOX 'FIRST'
'DEPOSIT' 'INSURANCE';

Short form:

TEXT 12 POINTS INK 1 IN BOX 4 3 'FIRST' 'NATIONAL' 'TRUST'
BOX 'FIRST' 'DEPOSIT' 'INSURANCE';

LOGO command

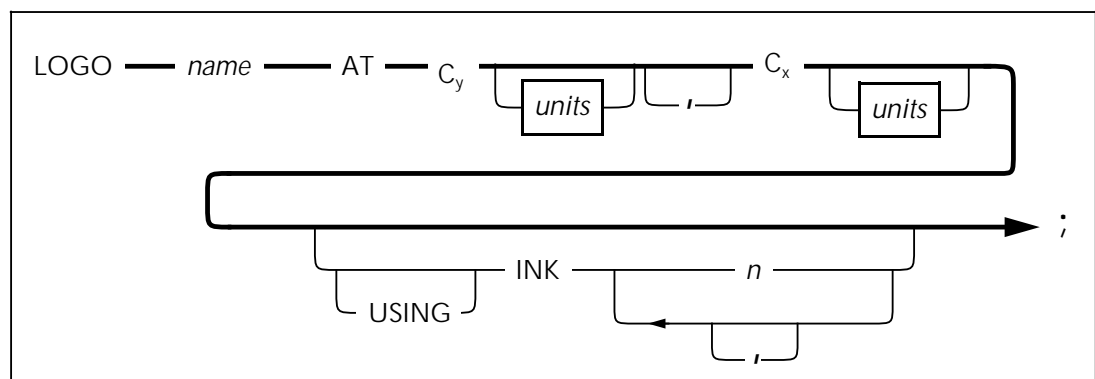
Use the **LOGO** command to place logos in the form. The logo is a special font of one or more characters, which carries with it the relative position of the characters to make up the logo. Although a logo may comprise many characters, it is positioned as a single unit. There is an option that overrides the colors given to a color logo file. For example, if you reference a logo in a color other than blue on a blue form, you must specify **'USING INK *n*'** to override the existing color specification of the logo. If you do not use this option, the printer requests an ink substitution prior to printing the form. Logos you specify in a form you are printing on a Xerox highlight color LPS printer must be in color format. You receive a 'LOGO NOT FOUND' message if you reference a black and white logo in a highlight color (HC) form. The command format designates the proper placement of a logo according to the planned form.



CAUTION: The XPRM database does not support XES logos. To include a logo on an XES form, you must convert the logo to a font and load it into the XPRM database as a font.

Figure 3-17 shows the **LOGO** command syntax flow.

Figure 3-17. **LOGO command syntax**

**Parameters**

name identifies the unique file name (or font identifier) for the logo. A logo name may consist of one to six characters.

C_y defines the y-axis coordinate for the logo origin location.

C_x defines the x-axis coordinate for the logo origin location.

units indicates the unit of measure for placement of the logo.

The *units* are **INCHES**, **CENTIMETERS**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You must express **DOTS** and **XDOTS** in integers.

INK *n* specifies the ink number from the ink list that overrides the corresponding ink specified in the ink list of the logo file. If the logo has a **NOSUBSTITUTION** flag, then ink overrides are not allowed (highlight color only).

When you incorporate a logo into your form, make sure you do not exceed the font memory capacity.

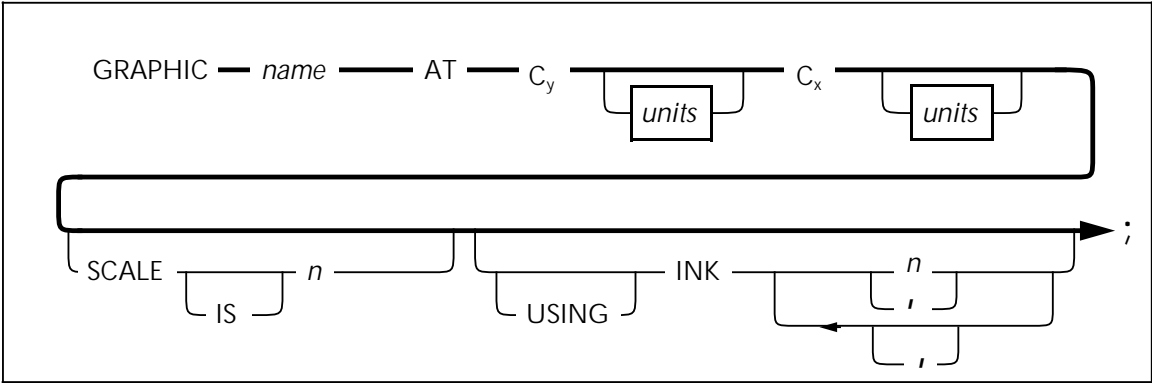
- Default** None.
- Examples** Long form:
LOGO NEW1 AT 1,1 USING INK 3;
There is no short form.

GRAPHIC command

A form may consist only of graphic images. You can have up to 16 images per page. Xerox highlight color LPS printers allow you to print colored graphics on your form. For highlight color forms, you can also override any ink specified in the ink list of a color graphic file.

You must specify each graphic associated with a form in the forms source language (FSL) file with the **GRAPHIC** command. Figure 3-18 shows the command syntax flow.

Figure 3-18. GRAPHIC command syntax



- Parameters** *name* identifies the IMG or XES graphic file name you specify. This is the name of the graphic stored on the printer.
- C_y* defines the coordinate on the y-axis to locate the logo origin.
- C_x* defines the coordinate on the x-axis where you want the logo origin located.
- units* indicates the unit of measure for placement of the graphic. The *units* are **INCHES**, **CENTIMETERS**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You must express **DOTS** and **XDOTS** in integers.
- SCALE** *n* specifies the scaling of the printed graphic that you have selected. The scaling factor value *n* must be an integer. You can only scale on a factor from 1 to 8.
- INK** *n* specifies the ink number from the ink list which overrides the corresponding ink list of the graphic (highlight color only).

- Default** None.

Examples Long form:
 GRAPHIC SYMBOL AT 1,1 SCALE IS 4 USING INK 3;
 There is no short form.

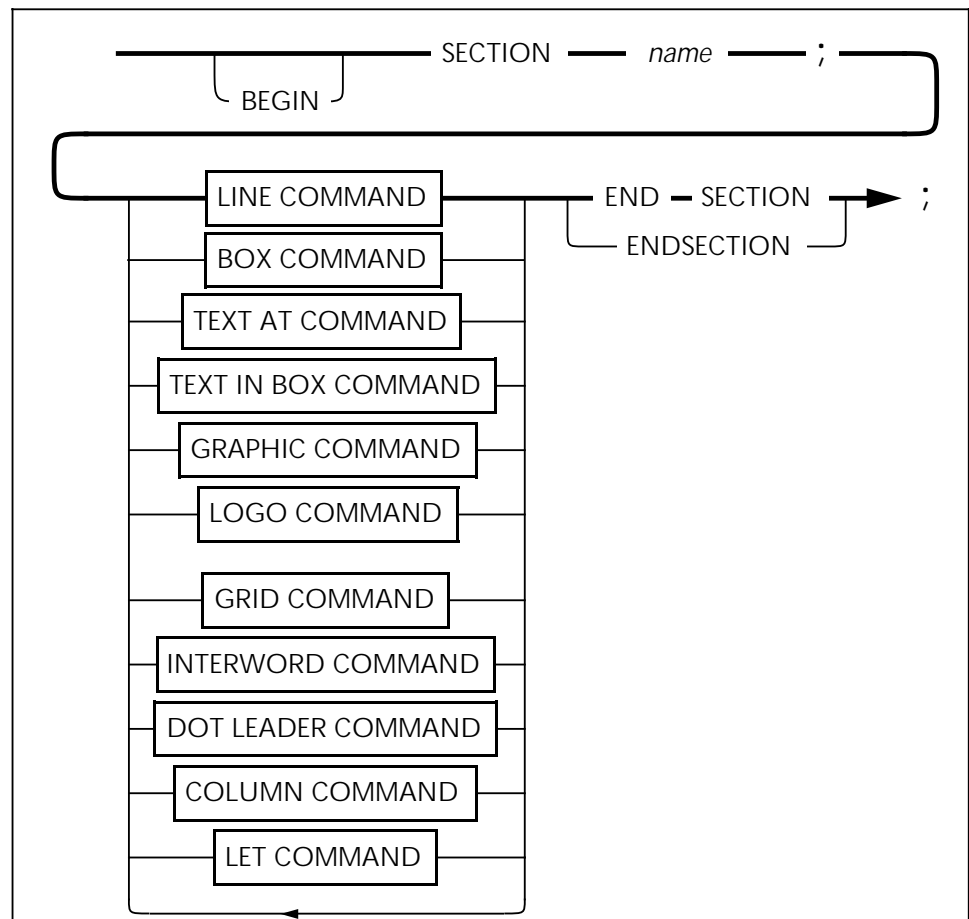
SECTION command

Use the **SECTION** command when a form consists of identical blocks of commands. You can save time by defining one section and then repeating it at the locations you want.

You can also use this command to define an entire form as a section. Within the section, you can use any other command, statement, or specification except another **SECTION** command, **DO SECTION** command, **FORM** command, or **END** command.

Figure 3–19 shows the **SECTION** command syntax flow.

Figure 3–19. **SECTION** command syntax



Parameter *name* specifies the name of the **SECTION** you want to use to invoke the **SECTION** command. You may define and invoke more than one section, but you must terminate each section with the command **END SECTION** or **ENDSECTION** before you define another section. Do not use a reserved word or keyword as a section name. *name* is a maximum 16-character string.

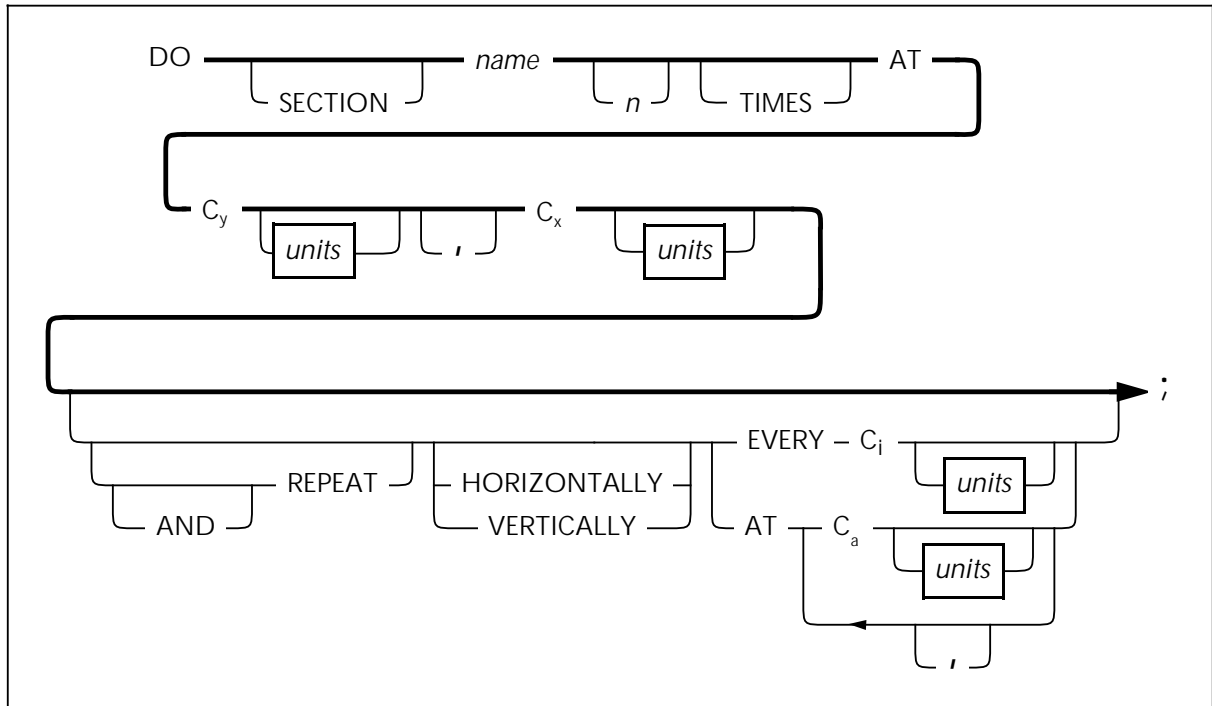
Default None.

Examples	<p>Long form:</p> <pre>BEGIN SECTION NUSEC; LOGO NEW0 AT 0,0 USING INK 2; TEXT AT 1,1 USING FONT 3 AND INK 1 'Acme Weather Balloons' END SECTION;</pre> <p>There is no short form.</p>
SECTION definitions	<p>The BEGIN SECTION command informs the compiler that what follows includes the END SECTION command and lines up to form a logical collection of commands that may be invoked as needed to build a complete form.</p> <p>Most HFDL commands may be found within a SECTION and these commands are documented in the SECTION command description. The key point to remember is that coordinates within the SECTION definition are relative to 0, 0. Also, if the GRID command is specified, then the GRID is only valid within the bounds of the defining SECTION. The GRID command is a local variable and it's scope is that of the defining SECTION. Other statements within the section will then use the local GRID command to determine the placement of lines, boxes, text, etc. Remember that these placement values are still relative to 0, 0.</p> <p>The final placement of the elements is defined by the DO SECTION command.</p>
SECTION invocation	<p>A SECTION is invoked by using the DO SECTION command. This command names the desired SECTION as well as the origin of the SECTION. The AT parameter takes on the role of the ORIGIN command for the SECTION and is used to determine the final placement of lines, boxes, text, etc. The final placement is computed as follows:</p> $x_{\text{Final}} = x_{\text{Origin}} + x_{\text{Value}} * x_{\text{Grid}}$ <p>and</p> $y_{\text{Final}} = y_{\text{Origin}} + y_{\text{Value}} * y_{\text{Grid}}$ <p>xValue and yValue are taken from the selected command. xGrid and yGrid are taken from the active GRID command. xOrigin and yOrigin are taken from the DO SECTION command that invokes the section.</p> <p>If a GRID command has not been defined within the SECTION, the GRID that is in effect for the FORM is used.</p>

DO SECTION command

Use the **DO SECTION** command to invoke a defined section. You must first define a section before you can invoke it. You can repeat sections at absolute or incremental coordinates. You can also repeat a section in the same location. This command is useful for sections containing relative coordinates.

Figure 3–20 shows the **DO SECTION** command syntax flow.

Figure 3–20. **DO SECTION** command syntax**Parameters**

name specifies the name of the section you want to invoke.

n **TIMES** specifies the number of times you want the section repeated.

C_y specifies the *y*-axis where you want the section origin located.

C_x specifies the *x*-axis where you want the section origin located.

units indicates the unit of measure for placement of the section origin. The *units* are **INCHES**, **CENTIMETERS**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You must express **DOTS** and **XDOTS** in integers.

REPEAT specifies the number of times you want a section repeated in the orientation you specified.

HORIZONTALLY or **VERTICALLY** specifies the orientation of the lines you want repeated.

EVERY C_i indicates that you want to repeat a section at the increment you have specified.

AT C_a indicates that you want to repeat the section at the absolute coordinate you have specified.

Default

VERTICALLY, if you specified **REPEAT**.

Examples

Long form:

```
DO SECTION NUSEC 3 TIMES AT 1,1 AND REPEAT
HORIZONTALLY EVERY 4;
```

Short form:

```
DO NUSEC 3 AT 1,1 REPEAT HORIZONTALLY EVERY 4;
```

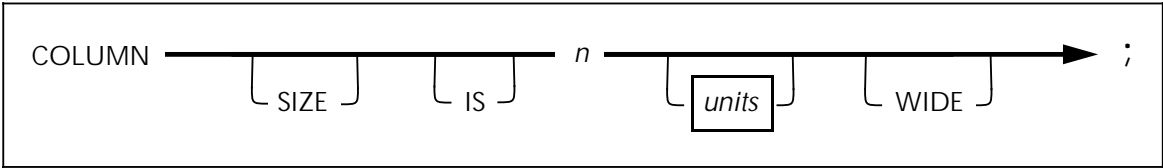
COLUMN command

Use the **COLUMN** command to establish the column width when you specify **TEXT ALIGNED JUSTIFIED IN COLUMN**. You can also use the **COLUMN** command to establish a right boundary for the **DOT LEADER** command.

Parameters *units* indicates the unit of measure for the column width. The *units* are **INCHES**, **CENTIMETERS**, **DOTS**, and **XDOTS**. You can use two decimal places when you specify inches and centimeters. You can express **DOTS** and **XDOTS** only in integers.

Figure 3–21 shows the **COLUMN** command syntax flow.

Figure 3–21. **COLUMN** command syntax



Defaults The default units specified for the **GRID** command, if you do not specify *units*.

Column width of 6.5 inches (portrait) or 9 inches (landscape), if you do not specify column width using the **COLUMN** command.

Examples Long form:
COLUMN SIZE IS 3.8 INCHES WIDE;
Short form:
COLUMN IS 3.8 IN WIDE;

INTERWORD command

The **INTERWORD** command specifies the interword multiplier used to determine the interword gap.

The interword gap formula

The interword gap is the blank space between words in printed text. It is determined in HFDL with this formula:

$$\text{interword gap} = \text{interword multiplier} * \text{base gap}$$

where

base gap is one of these values, based on the first named font:

- the width of ASCII code point X'20', usually a "blank"
- an EN space, which is 1/2 the point size of the font

interword multiplier is one of these values:

- the value specified by the **INTERWORD** command
- 1.0, if no **INTERWORD** command is specified.

HFDL uses the interword gap you specify as the minimum value. For fully justified text, HFDL adjusts the interword gap when needed.

Specifying the base gap

The base gap for text not fully justified is always ASCII code point X'20', the "blank".

The base gap for fully justified text is determined by the SPACING operand of the COMPILE command. If SPACING is specified in the JCL for such a job, the base gap is the EN space.

If SPACING is specified in the JCL for a job which does not include fully justified text, SPACING is not used, and the base gap remains ACSII code point X'20', the "blank".

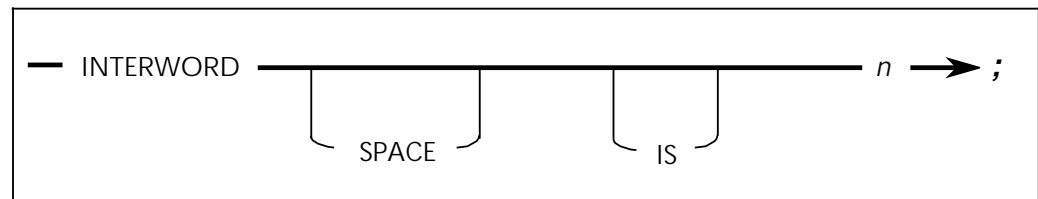
For an example of how to specify the SPACING operand, refer to the “Command syntax” section of the “Managing and printing sample forms” chapter of this manual.

Specifying the interword multiplier

The INTERWORD command specifies the interword multiplier in the formula above. The command is not an FDL command; it will not work on a form compiled at the printer.

Figure 3-22 shows the syntax for the INTERWORD command.

Figure 3–22. **INTERWORD** command syntax



n is the interword multiplier used to determine the interword gap. This value is multiplied by the base gap.

Examples:

INTERWORD SPACE IS 0.8;

INTERWORD 0.8;

Using SPACING and INTERWORD together

For justified text, use the SPACING operand of the COMPILE command along with the INTERWORD command to specify the size of the interword gap.

Table 3-1 illustrates how the two parameters work together to determine the interword gap.

Table 3-1. Determining interword

The interword multiplier:	applied to the base gap, specified by:	equals an interword gap of:
INTERWORD .5	no SPACING operand	0.5 * the width of ASCII code point X'20'
INTERWORD .5	COMPILE SPACING ¹	0.5 * the EN space
INTERWORD 1.0	no SPACING operand	1.0 * the width of ASCII code point X'20'
INTERWORD 1.0	COMPILE SPACING ¹	1.0 * the EN space
(no INTERWORD command specified, defaults to 1.0)	no SPACING operand	(1.0) * the width of ASCII code point X'20'
(no INTERWORD command specified, defaults to 1.0)	COMPILE SPACING ¹	(1.0) * the EN space

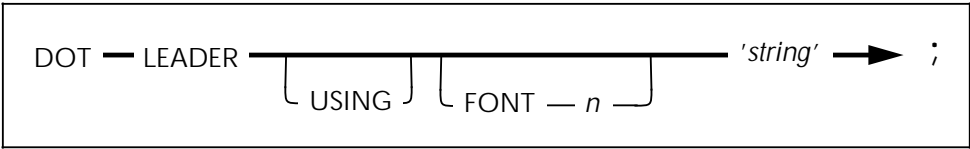
¹ Valid for fully justified text only

DOT LEADER command

Use the **DOT LEADER** command to define the character string you specify when you use **DOT LEADER** in the **TEXT AT** command. The text string portion of the **TEXT AT** command allows for **DOT LEADER** requests.

Figure 3–23 shows the command syntax flow.

Figure 3–23. DOT LEADER command syntax



- Parameters

'string' specifies the type of dot leader you want to use. Strings cannot exceed 12 characters.
- USING FONT** *n* specifies the font you want to use for the dot leader characters.
- Defaults

Single dots of the font currently in use, if you do not specify a font and you use **DOT LEADER** in the **TEXT** command.
- Examples

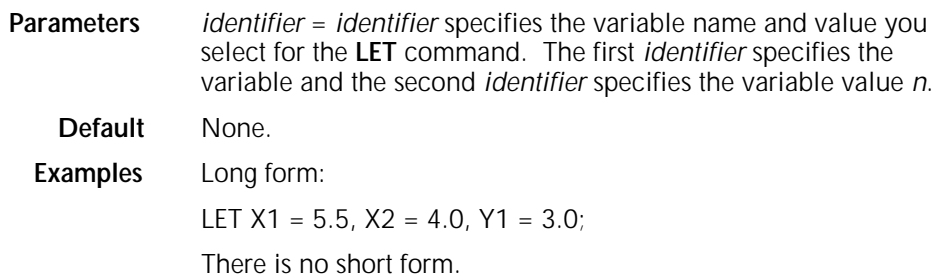
Long form:
DOT LEADER USING FONT 1 '*' ;

[illegible]

Use the **LET** command to define variable names. You can specify all x and y coordinates with either literal values or variable names. If you use variable names, you must first define them with the **LET** command. The **LET** command allows an *identifier* (variable) to equal an *n* value. Variable names must begin with an alphabetic character and may be up to 16 characters in length.

You can only set variables to a specific numeric amount. You cannot specify units in the **LET** statement. However, you can specify units (if needed) when the variable is used as a coordinate. For example, if variable Y1 is set to 3 in a **LET** statement and a line is drawn at Y1 inches, it is drawn at 3 inches. If the grid units are dots and no units are specified, then the line Y1 is drawn at 3 dots.

Figure 3-24. LET command syntax



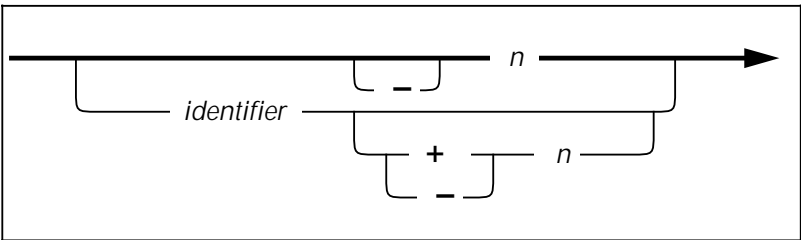
You can specify all coordinates as literal values or variables. You must first define variables in the **LET** statement. Refer to the “**LET** command” section for additional information. If you use variables outside of the **LET** command, they are called relative coordinates. These coordinates can also have a positive or a negative value. This allows you to place text items relative to one another. If you want to shift the whole block of text up or

down, you only have to change the initial values. You can shift text automatically when you add or delete text items.

HFDL uses spaces to delimit keywords. Therefore, relative coordinates must not contain spaces between the variable name Y1 and the value +.5.

Figure 3–25 shows the variable syntax flow.

Figure 3–25. **Relative coordinate syntax**



- Parameters** *identifier* is the variable name you specify for the relative coordinate.
- + or – specifies a positive or negative value that you can apply to the *n* value.
- Default** None.
- Examples** The following examples show the long form of the relative coordinate variables.
- In this example, you can define the Y coordinate as the value of Y1 plus 3 centimeters:
- AT Y1+3 CM
- In this example, you can define the Y coordinate as the value of Y2 minus 4.5 grid units:
- AT Y2–4.5
- There is no short form.



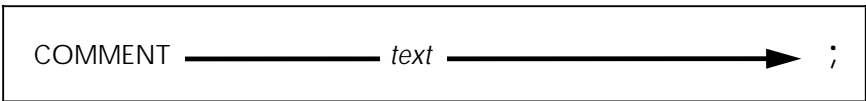
In the examples above, the value of **Y1** and **Y2** change. When you refer to **Y1** in the FSL, it is 3 larger. Likewise, **Y2** is 4.5 smaller.

COMMENT command

Use the **COMMENT** command to retain a record of your comments in the source statement. Comments do not appear on the form, and the compiler ignores them. You do not need to enclose the **COMMENT** command text in single quotes.

Figure 3–26 shows the **COMMENT** command syntax flow.

Figure 3–26. **COMMENT command syntax**



- Parameter** *text* specifies the comment text string you want to include in the HFDL command statement.
- Default** None.

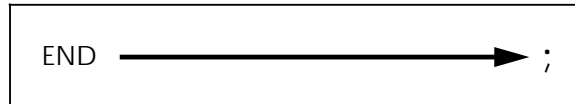
Examples Long form:
 COMMENT ***HFDL IS ABLE TO DESCRIBE MANY TYPES OF
 FORMS***;
 There is no short form.

END command

Use the **END** command to terminate a form description.

Figure 3–27 shows the command syntax flow.

Figure 3–27. **END command syntax**



Parameter **END** specifies an end-of-file condition from the source input disk file or from the input data stream.

Default There is no default for the **END** command. It is an optional command.

Example Long form:
 FORM HFDL1;; END;
 There is no short form.

Setting up the form

Follow these steps to set up the basic elements of your form:

1. Use the **FORM** command to specify both a unique form name that identifies the form to the system and a resolution.

In this example, the form name is XEROX2 and the resolution is 300 SPI.

FORM XEROX2, RESOLUTION IS 300 SPI;

2. Specify the physical paper size.

This example specifies U.S. letter size:

PAPER SIZE IS USLETTER;

3. Specify the grid unit and form origin. The form origin should coincide with variable data origin and form grid units should correspond with the character and line spacing of the variable data.

The example below shows the long form of the command using FMT1 values:

GRID UNIT IS 13.6 CPI 8.1 LPI ORIGIN .18 IN .66 IN;

This example shows the short form of the command:

GRID FMT1;

You do not need to use the **LANDSCAPE/PORTRAIT** command to specify page orientation and virtual page size if you specify a predefined format in the **GRID** command.

4. Specify the fonts you want to use.

In this example, you specify a large font (UN214A) for the report title, a smaller font (UN107A) for the column headings, and a very small font (UN104A) for the form identifier:

FONTS UN214A, UN107A, UN104A;

When you refer to these fonts in later descriptive commands, the first font specified (UN214A) is identified as FONT 1, the second as FONT 2, and the third as FONT 3.

This example shows entering all of these setup commands in their long form:

FORM XEROX2 RESOLUTION IS 300 SPI;

PAPER SIZE IS USLETTER;

GRID UNIT IS 13.6 CPI 8.1 LPI ORIGIN .18 IN .66 IN;

FONTS UN214A, UN107A, UN104A;

Since the sample form uses the default resolution, default physical page size, and default grid format, you can omit these commands. This example shows entering all of the setup commands required in their short form:

FORM XEROX2;

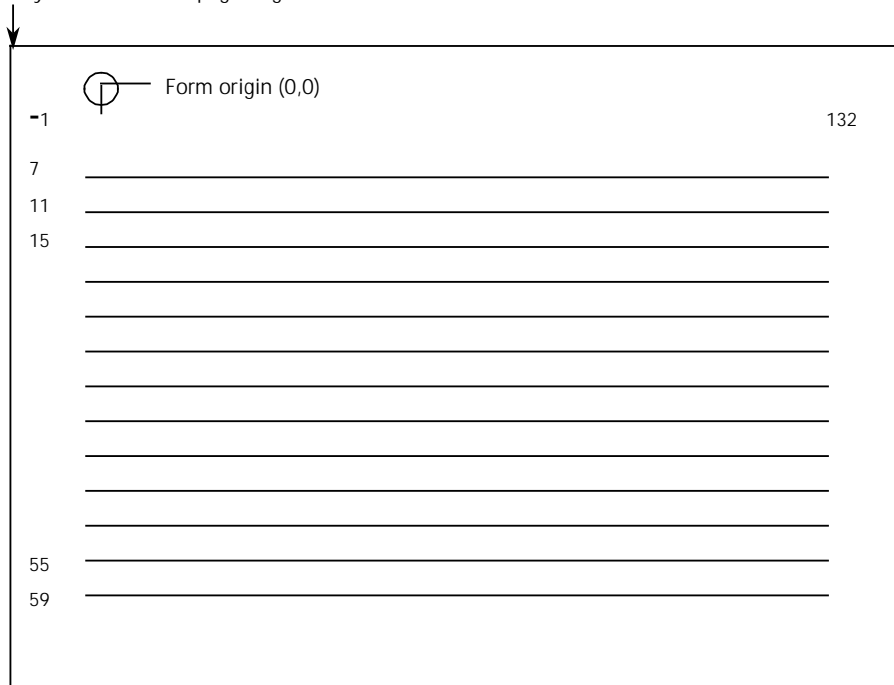
FONTS UN214A, UN107A, UN104A;

Using the sample description command

After you finish setting up the sample form, follow these steps to enter descriptive data:

1. Draw 14 parallel horizontal lines as shown.

Physical and virtual page origin



The following example shows the long form of the LINE command:

AT 7 DRAW 14 HORIZONTAL LINES FROM -1 TO 132
USING SOLID 1 AND REPEAT VERTICALLY EVERY 4;

The LINE command draws lines in a horizontal direction under character row 7 and every fourth row thereafter, beginning two character positions to the left of character position 1 and ending at character position 132.

SOLID 1 specifies the appearance of the lines, which are solid and one point wide. Make sure that you stay within the boundaries of the system page when you specify a negative character position.

VERTICALLY is the default direction of the repetition process when printing is horizontal, since the usual intent is to draw parallel lines.

If you omit optional keywords and system defaults, the short form of the command looks like this example:

AT 7 14 LINES -1 TO 132 EVERY 4;

2. Draw 15 parallel vertical lines as shown.

[illegible]

This example shows the long form of the **LINE** command:

AT -1 DRAW VERTICAL LINES FROM 7 TO 59 USING SOLID 1 AND REPEAT HORIZONTALLY AT 4, 12, 39, 48, 55, 64, 72, 79, 87, 96, 104, 112, 121, 132;

The **L**INE command draws lines in a vertical direction, beginning at character position -1 and repeated at the specified intervals, with each line extending from character row 7 to character row 59. HORIZONTALY is the default direction of the repetition process when printing is vertical.

The following example, which omits optional keywords and system defaults, is the short form of the command:

AT -1 VERTICAL LINES 7 TO 59 AT 4, 12, 39, 48, 55, 64, 72,
79, 87, 96, 104, 112, 121, 132;

3. Use the **BOX** command to draw an invisible box and center it above the table.

This example shows the long form of the command:

AT 2, -1 DRAW 1 BOX 133 WIDE BY 5 HIGH USING SOLID
0;

The **BOX** command draws an invisible box 133 grid units wide by 5 grid units high under character row 2, beginning two character positions to the left of character position 1. Use SOLID 0 as the thickness of the box line when you want to draw an invisible box. Notice that the horizontal sides of the box begin and end at the same character positions as the horizontal lines of the table.

This example shows the short form of the command:

AT 2, -1 BOX 133, 5 SOLID 0;

4. Position the title in the center of the invisible box.

This example shows the long form of the TEXT IN BOX command:

HORIZONTAL TEXT ALIGNED CENTER USING FONT 1 IN
CENTER BOX 2, -1 'COMPOSITE STOCK INVENTORY
STATUS REPORT';

The **TEXT IN BOX** command places the title COMPOSITE STOCK INVENTORY STATUS REPORT in the center of the box whose upper left corner begins at character row 2, at character position -1. FONT 1 is UN214A, the first font you specified with the FONT command; FONT 2 is UN107A, the second font specified.

5. Specify the column headings.

This example shows the long form of the command:

HORIZONTAL TEXT ALIGNED CENTER USING FONT 2
IN CENTER BOX 7, -1 'CAT' 'NO'

IN NEXT HORIZONTAL BOX 'ITEM' 'NO'

IN NEXT HORIZONTAL BOX 'DESCRIPTION'

IN NEXT HORIZONTAL BOX 'REOR' 'POINT'

IN NEXT HORIZONTAL BOX 'REOR' 'QTY'

IN NEXT HORIZONTAL BOX 'OPEN' 'STOCK'

IN NEXT HORIZONTAL BOX 'VENDOR' 'RECEIPTS'

IN NEXT HORIZONTAL BOX 'STOCK' 'ORDERS'

IN NEXT HORIZONTAL BOX 'COMM' 'STOCK'

IN NEXT HORIZONTAL BOX 'SALES' 'ISSUES'

IN NEXT HORIZONTAL BOX 'BACK' 'ORDERS'

IN NEXT HORIZONTAL BOX 'VENDOR' 'RETURNS'

IN NEXT HORIZONTAL BOX 'STOCK' 'TRANSFERS'

IN NEXT HORIZONTAL BOX 'CURRENT' 'STOCK';

This command places a column heading in each column of the table, using FONT 2. The system automatically centers column headings.

The **TEXT IN BOX** command simplifies the task of placing text in a series of adjacent boxes. You only need to specify the coordinates of the upper-left corner of the first box. Subsequently, when you enter a **BOX** command, the system automatically recognizes the coordinates of a box directly to the right of the current box. The keywords IN NEXT HORIZONTAL are optional. It is also possible to specify the box directly below the current box, however, you must enter the keywords IN NEXT VERTICAL.

This example shows the short form of the command:

```
TEXT FONT 2 IN BOX 7, -1 'CAT' 'NO'  
BOX 'ITEM' 'NO'  
BOX 'DESCRIPTION'  
BOX 'REOR' 'POINT'  
BOX 'REOR' 'QTY'  
BOX 'OPEN' 'STOCK'  
BOX 'VENDOR' 'RECEIPTS'  
BOX 'STOCK' 'ORDERS'  
BOX 'COMM' 'STOCK'  
BOX 'SALES' 'ISSUES'  
BOX 'BACK' 'ORDERS'  
BOX 'VENDOR' 'RETURNS'  
BOX 'STOCK' 'TRANSFERS'  
BOX 'CURRENT' 'STOCK';
```

6. Apply shading to the form.

This example shows the long form of the command:

```
At 11, -1 DRAW 6 BOXES 133 WIDE BY 4 HIGH USING  
SHADING MEDIUM AND REPEAT VERTICALLY EVERY 8;
```

This command draws six shaded boxes 133 grid units wide by 4 grid units high and an identical series of shaded boxes every eighth row. The coordinates of the upper left corner of the first box are character row 11, character position -1.

This example shows the short form of the command:

```
AT 11, -1 6 BOXES 133, 4 SHADING VERTICALLY EVERY 8;
```

7. Place the form's identifier and date of issue at the bottom of the form.

This example shows the long form of the command:

```
HORIZONTAL TEXT USING FONT 3 AT 61, -1 'XEROX2  
(5/93)';
```

This command places the words XEROX2 (5/93) at character row 61, character position -1, or on the line immediately below the table, flush with the left edge of the table. Font 3 is UN104A, the third font specified in the **FONT** command.

This example shows the short form of the command:

```
TEXT FONT 3 AT 61, -1 'XEROX2 (5/93)';
```

8. Terminate the form description with an **END** command, as follows:

```
END;
```

When you have created your form design using Host Forms Description Language (HFDL) commands, you can compile, batch process, store, and print a sample of it. After this process completes, your form becomes available for use. This chapter describes the **COMPILE** command and the job control language (JCL) to invoke the HFDL compiler.

JCL requirements

During installation, your System Administrator or system installer modifies the items in your JCL, as appropriate, for your specific site.

You enter items in the JCL that are specific to your job when you are ready to compile your form with HFDL. If you are not familiar with the items you need to change in the JCL, contact your System Administrator.

Using the COMPILE command

You use the **COMPILE** command to indicate how you want to compile the form. The **COMPILE** command is the only command available for compiling forms.

Command coding conventions

Use the following conventions when you code the **COMPILE** command:

- Follow the command syntax carefully.
- Enter operands in uppercase or lowercase characters.
- Type operands in the first 72 columns of a record.
- Abbreviate as indicated by the capital letters.

Command syntax

Figure 5–1 shows the **COMPILE** command syntax flow.

Figure 5–1. **COMPILE command syntax**

COMPILE FSL=DDname[(name)] optional other parameters
--

Parameters	<p>FSL is the only valid resource you can use.</p> <p>FSL=DDname[(PDS member name)] where DDname = XFFSL and PDS member name is the member name of the PDS where the FSL is stored. The DDname can be either a sequential or a partition data set. When the DDname is sequential, the PDS member name is not applicable.</p>
Operands	<p>LIST produces the full compiler listing. To specify no listing, delete this operand.</p> <p>VERSION=Other specifies the form version you want to compile. 2 produces a form suitable for the version 2 and version 3 printers (4050, 4090, 4135, 4650, 8790, 9790). HC produces a form suitable for a Xerox highlight color LPS printer. X1 or XC produces a form suitable for Xerox decentralized printers. The default is 2.</p> <p>Form versions are not significant unless you are creating a highlight color form or an XES form. Version HC only prints on a Xerox highlight color printer. You should use version HC for compiling any form that you will use on a highlight color printer, even if it is a black and white form. When you specify VERS=2 or VERS=10 and you have referenced color anywhere in the form, VERSION is automatically set to HC. When you specify VERS=X1 and you have referenced color anywhere in the form, VERSION is automatically set to XC. Version XC only prints on a Xerox full color decentralized printer.</p> <p>OBJECT specifies that the database stores the compiled form. The default is not to store the compiled form in the database.</p> <p>SPACING is used for fully justified text. It specifies the base gap used in this formula to determine the interword gap:</p> $\text{interword gap} = \text{interword multiplier} * \text{base gap}$ <p>If you include this operand for the COMPILE command, the base gap is the EN space. The EN space is 1/2 the point size of the first named font.</p> <p>If you omit this operand, the base gap is the width of ASCII code point X'20'. This code point is usually a "blank", and is always derived from the first named font.</p> <p>This operand is valid for fully justified text only.</p> <p>The INTERWORD command specifies the interword multiplier in the interword gap formula above. Table 5-1 illustrates how the INTERWORD command and the SPACING operand work together to determine the interword gap for fully justified text.</p>

Table 5-1. Determining interword gap

The interword multiplier:	applied to the base gap, specified by:	equals an interword gap of:
INTERWORD .5	no SPACING operand	.5 * the width of ASCII code point X'20'
INTERWORD .5	COMPILE SPACING¹	.5 * the EN space
INTERWORD 1.0	no SPACING operand	1.0 * the width of ASCII code point X'20'
INTERWORD 1.0	COMPILE SPACING¹	1.0 * the EN space
(no INTERWORD command specified, defaults to 1.0)	no SPACING operand	(1.0) * the width of ASCII code point X'20'
(no INTERWORD command specified, defaults to 1.0)	COMPILE SPACING¹	(1.0) the EN space

¹ Valid for fully justified text only

HFDL uses the interword gap you specify as the minimum value. For fully justified text, HFDL adjusts the interword gap when needed.

For the INTERWORD command description and syntax, refer to the "Using HFDL commands" chapter of this manual.

The example in Figure 5-2 shows MYFORM being compiled with SPACING specified. The EN space will be used along with the value specified by the INTERWORD FSL command to determine the interword gap for this form.

Figure 5-2. **SPACING compile syntax**

```
COMPILE FSL=XFFSL(MYFORM) VERSION=HC SPACING
```

GROup=name specifies a user-assigned optional "name" to the form. This allows you to group forms in the database for subsequent listing, downloading, and so on. For example, list all forms with GROUP=MEDINS. The maximum size of the field is 8 bytes.

KEYboard=name specifies the keyboard map you want to use in the compilation of the form. **name** is a maximum 8-character string. The compiler retrieves the requested keyboard map and associated font map. Then the compiler maps the characters entered on the form to the corresponding graphic representations in the Xerox printer fonts. If no keyboard name is entered, no keyboard mapping occurs. The font characters are translated directly from EBCDIC to ASCII and then located in the font based on their ASCII value. For more information, refer to the *Xerox Print Resources Manager/Host Forms Description Language 3.2 for IBM MVS Managing Resources* manual.

MEMber=selection specifies which member(s) you want to compile. Specify only one member by using the exact member name for **selection**. Specify a group of members by using a DOS style command such as * and ?. Specify all members in the partition data set by substituting **selection** with *.

FORm=selection specifies which member(s) you want to compile in a multi-form file. Specify only one form by using the exact form name for **selection**. Specify a group of forms by using a DOS style command such as * and ?. Specify all forms in the partition data set by substituting **selection** with *.

LITeral=character specifies a user-assigned character for introducing a literal. A literal is a character from the font character set, as opposed to a keyboard character. The character must be coded in ASCII hexadecimal after the literal character. The default literal introduction character is the exclamation point. To use the exclamation point as a normal keyboard character without having to key it twice in succession, override the default literal introduction using this operand.

Table 5-2 shows an example of how a different literal introduction character can be used to make frequent keying of the exclamation point easier:

Table 5-2. **Coding alternative literal introduction characters**

To print:	When literal intro is !, key:	When literal intro is &, key:
65%	65!25	65&25
SOLD!	SOLD!!	SOLD!



The code point for % is 25.

SHIFT=character specifies a user-assigned character for shifting text between uppercase and lowercase. The default shift character is the # character. To use the # character as a normal keyboard character without having to key it twice in succession, override the default shift using this operand.

Table 5-3 shows an example of how a different shift character can be used to make frequent keying of a normal # character easier:

Table 5-3. **Coding alternative shift characters**

To print:	When shift character is #, key:	When shift character is *, key:
RevA	R#EV#A	R*EV*A
#4	##4	#4

PAPersize=name specifies the default paper size you want for your form. **name** is one of the following values: LT, LG, A4, A3, or B4.

Command syntax examples

Figure 5–3 shows an example of the command you use to compile a form called MYFORM where the DDname is XFFSL and the database will store the compiled form.

Figure 5–3. **Sample COMPILE syntax**

```
COMpile FSL=XFFSL(MYFORM) LISt OBJECT
```

This example compiles MYFORM with a full compiler report in version 2 format.

The example in Figure 5–4 shows compiling MYFORM without a compiler listing. This example compiles MYFORM with no compiler listing in the HC form format. The database does not store the compiled form.

Figure 5–4. **Highlight color COMPILE syntax**

```
COMPILE FSL=XFFSL(MYFORM) VERsion=HC
```

The example in Figure 5–5 shows how to compile a number of member names that are MMBRxx in a partition data set XFFSL. For each member that matches the indicated criteria, only a form name starting with FORM will be compiled. All the forms are compiled for the 4700 printer.

Figure 5–5. **Multiple member and multiple form COMPILE syntax**

```
COM FSL=XFFSL VER=XC MEM=MMBR?? FOR=FORM*
```

The example in Figure 5–6 shows how to compile a form stored in a sequential data set XFFSL with the user definable literal character set to "\$" and the shift character set to "@." The form is compiled for the decentralized printers.

Figure 5–6. **User definable literal and shift COMPILE syntax**

```
COMPILE FSL=XFFSL VER=X1 LITERAL=$ SHIFT=@ OBJECT
```

Batch processing COMPILE commands

Batch processing allows you to perform multiple sets of **COMPILE** commands in one step. You use the **BATCH** command instead of the **COMPILE** command. You can also perform nested batch processing, in which a batch command is nested within another batch command. Figure 5–7 shows the **BATCH** command syntax.

Figure 5–7. **BATCH command syntax**

`BATCH DDname[(PDS member name)]`

The *PDS member name* identifies the file that contains the **COMPILE** commands. The PDS member must be in the same data set as the FSLs you are compiling.

If you are compiling forms in the XYZFORMS data set, and the DDname is XFFSL, you can create the batch command statement. Figure 5–8 shows an example of the BATCH command with XFFSL as the DDname and BATCH01 as the PDS member name.

Figure 5–8. **Sample BATCH command**

`BATCH XFFSL (BATCH01)`

The XYZFORMS data set contains the BATCH01 PDS member, which contains the **COMPILE** commands for the forms you are compiling.

An example of the batchfile is:

```
COMPILE FSL=XFFSL(XYZFM1) LIST OBJECT VER=2
COMPILE FSL=XFFSL(XYZFM2) LIST OBJECT VER=HC
COMPILE FSL=XFFSL(XYZFM3) LIST OBJECT VER=X1
COMPILE FSL=XFFSL(XYZFM3) LIST OBJECT VER=XC
```

HFDL compiler JCL

The Virtual Storage Access Method (VSAM) data sets store the HFDL compiler resources (fonts, logos, forms, and colormaps). These, along with printer device definitions, comprise the XPRM database.

Table 5–4 identifies the DDnames that the compiler uses to reference data sets for centralized printers. You must include a DD statement for each in your compiler invocation JCL.

Table 5–4. **Resource DDnames for centralized printers**

DDname	Description
XSFNTATT	Data set containing font and logo attributes.
XSFORMS	Data set containing forms. Each record contains a resource occurrence in Xerox LPS printer file format – .FRM or .FR6.
XSSUPTAB	Data set containing database references.
XSFRMATT	Data set containing form attributes.

Table 5–5 identifies the DDnames that the compiler uses to reference data sets for decentralized printers.

Table 5–5. **Resource DDnames for decentralized printers**

DDname	Description
XSFNTATT	Data set containing font and logo attributes.
XSFORMS	Data set containing forms. Each record contains a resource occurrence in Xerox LPS printer file format – XES.
XSFRMATT	Data set containing form attributes.
XSCLRATT	Data set containing color attributes.
XSSUPTAB	Data set containing database references.



Centralized and decentralized resources are in the same data sets.

Table 5–6 identifies the DDnames and the statements required in addition to the resource data sets for centralized printers.

Table 5–6. **HFDL execution DDnames for centralized printers**

DDname	Description
SYSPRINT	Writes the compiler listings when the LIST parameter of the COMPILE command is coded. Compile errors and the copyright message are written to SYSPRINT as well.
SYSTEM	Indicates a malfunction. Contact your system administrator. Information is not written to this DDname under normal conditions.
XFLOG	Condition codes log for all forms compiled. This file is useful when you compile forms and would like to find out quickly which form causes the program to return the condition code to other than 0.
FSL DDname	Data set containing the forms source language that is input to the compiler (FSL). This data set may be a sequential disk file or a member of a partitioned data set (PDS).
XFINP	Temporary data set that the compiler uses.
XFIFF	Temporary data set that the compiler uses.
XFFRM	Temporary data set that the compiler uses.
XFTMP	Temporary data set that the compiler uses.

Table 5–7 identifies the DDnames and the statements required in addition to the resource data sets for decentralized printers.

Table 5–7. **HFDL execution DDnames for decentralized printers**

DDname	Description
SYSPRINT	Writes the compiler listings when the LIST parameter of the COMPILE command is coded. Compile errors and the copyright message are written to SYSPRINT as well.
SYSTEM	Indicates a malfunction. Contact your system administrator. Information is not written to this DDname under normal conditions.
XFLOG	Condition codes log for all forms compiled. This file is useful when you compile forms and would like to find out quickly which form causes the program to return the condition code to other than 0.
FSL DDname	Data set containing the forms source language that is input to the compiler (FSL). This data set may be a sequential disk file or a member of a partitioned data set (PDS).
XFINP	Temporary data set that the compiler uses.
XFIFF	Temporary data set that the compiler uses.
XFFRM	Temporary data set that the compiler uses.
XFTMP	Temporary data set that the compiler uses.
XFIFO	Temporary data set that the compiler uses.

You must customize the HFDL compiler job control language (JCL) to meet the standards and conventions for your MVS installation. The minimum changes that you should make follow below.

- Modify the JOB statement to meet installation standards.
- Modify the DSNAMES in the DD statements for the resource data sets, execution data sets, and XSSUPTAB to meet installation standards.
- Modify the UNIT name for the temporary data sets.
- Modify the EXEC statement for each compilation for specific form names and parameters governing the compile.
- Do not remove the COND parameter as shown in step 2 in Figure 5–10. These parameters are required for proper condition code handling.

For complex forms, increase your region size and temporary data set space allocation to avoid ABENDS.

If you want to download and print a form sample following the compilation, the DDnames listed in Table 5–8 are required.

Table 5–8. **Data sets required to download and print**

DDnames	Description
XSUIN	XPRM Utility commands are read from the file specified by the XSUIN DD statement. XSUIN may specify either instream data or reference a sequential data set or a member of a partitioned data set (PDS).
XSUACTL	All messages from the XPRM utility are written to XSUACTL.
XSSUPTAB	Identifies the data set that contains the printer device definitions.
Printer DDname	A DD statement for a printer to which the utility will download and sample the form. For more information on this DDname see the <i>Xerox Print Resources Manager/Host Forms Description Language 3.2 for IBM MVS Managing Resources</i> manual.

Figure 5–9 shows an example of the JCL you use to compile a form and store it in the database.

Figure 5–9. Invoking the HFDL compiler

```
//LOGONIDX      JOB 'RUN HFDL COMPILER',REGION=8M user job card parameters
//*
/*JOBPARM      LINE=0
//STEP1        EXEC PGM=HFDL32,PARM='COMPILE FSL=XFFSL(XVERL) LIST OBJECT'
//STEPLIB      DD DSN=HFDL.V3R2M00.LOAD,DISP=SHR
//             DD DSN=XPRM.V3R2M00.SASCRDLB.C550,DISP=SHR
//             DD DSN=XPRM.V3R2M00.LOAD,DISP=SHR
//SYSPRINT     DD SYSOUT=*
//SYSTEM      DD SYSOUT=*
//*           ****The following data sets are specific to the form compiler
//*           ----Condition codes log for all files compiled
//XFLOG        DD SYSOUT=*
//*           ----The fsl source file is a customer defined pds or seq
//XFFSL        DD DSN=HFDL.SOURCE.FSL,DISP=SHR
//*           ----The inp file is a temporary file
//XFINP        DD DSN=&&INP,DISP=(NEW,PASS),UNIT=SYSDA,
//             SPACE=(TRK,(10,1)),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)
//*           ----The iff file is a temporary file
//XFIFF        DD DSN=&&IFF,DISP=(NEW,PASS),UNIT=SYSDA,
//             SPACE=(TRK,(10,1)), DCB=(RECFM=FB,LRECL=320,BLKSIZE=10240)
//*           ----The compiled forms are temporary files until stored
//*           in the database
//XFFRM        DD DSN=&&FRM,DISP=(NEW,PASS),UNIT=SYSDA,
//             SPACE=(TRK,(10,10,10)),DCB=(RECFM=F,LRECL=128,BLKSIZE=128)
//XFTMP        DD DSN=&&TMP,DISP=(NEW,PASS),UNIT=SYSDA,
//             SPACE=(TRK,(10,1)),DCB=(RECFM=FB,LRECL=320,BLKSIZE=10240)
//*           *****The following data sets are the resource data sets
//*           ---- XSFNTATT is a VSAM dataset containing FONT and LOGO
//*           ATTRIBUTES
//XSFNTATT     DD DSN=XPRM.V3R2M00.XSFNTATT,DISP=SHR
//*           ---- XSFORMS is a VSAM dataset containing FORMS
//XSFORMS      DD DSN=XPRM.V3R2M00.XSFORMS,DISP=SHR
//*           ---- XSFRTATT is a VSAM dataset containing FORM ATTRIBUTES
//XSFRTATT     DD DSN=XPRM.V3R2M00.XSFRTATT,DISP=SHR
//XSSUPTAB     DD DSN=XPRM.V3R2M00.XSSUPTAB,DISP=SHR
//*
//*           *****THE JCL FOR DOWNLOAD AND SAMPLE VIA XPRMUTIL GOES HERE
```

Downloading and printing a sample

If you want to download and print a sample of the form, run the XPRMUTIL job step immediately following the compile step.

Figure 5–10 shows an example of a JCL you use to compile, store, download, and sample a form on the printer.

Figure 5–10. Downloading and printing a sample

```

//LOGONIDX      JOB 'RUN HFDL COMPILER',REGION=8M user job card parameters
//*
/*JOBPARM      LINE=0
//STEP1        EXEC PGM=HFDL32,PARM='COMPILE FSL=XFFSL(XVERL) LIST OBJECT'
//STEPLIB      DD DSN=HFDL.V3R2M00.LOAD,DISP=SHR
//              DD DSN=XPRM.V3R2M00.SASCRDLB.C550,DISP=SHR
//              DD DSN=XPRM.V3R2M00.LOAD,DISP=SHR
//SYSPRINT     DD SYSOUT=*
//SYSTEM       DD SYSOUT=*
//*            ****The following data sets are specific to the form compiler
//*            ----Condition codes log for all files compiled
//XFLOG        DD SYSOUT=*
//*            ----the fsl source file is a customer defined pds or seq
//XFFSL        DD DSN=HFDL.SOURCE.FSL,DISP=SHR
//*            ----the inp file is a temporary file
//XFINP        DD DSN=&&INP,DISP=(NEW,PASS),UNIT=SYSDA,
//              SPACE=(TRK,(10,1)),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)
//*            ----the iff file is a temporary file
//XFIFF        DD DSN=&&IFF,DISP=(NEW,PASS),UNIT=SYSDA,
//              SPACE=(TRK,(10,1)),DCB=(RECFM=FB,LRECL=320,BLKSIZE=10240)
//*            ----The compiled forms are temporary files until stored
//*            in the database
//XFFRM        DD DSN=&&FRM,DISP=(NEW,PASS),UNIT=SYSDA,
//              SPACE=(TRK,(10,10,10)),DCB=(RECFM=F,LRECL=128,BLKSIZE=128)
//XFTMP        DD DSN=&&TMP,DISP=(NEW,PASS),UNIT=SYSDA,
//              SPACE=(TRK,(10,1)),DCB=(RECFM=FB,LRECL=320,BLKSIZE=10240)
//*            ****The following data sets are the resource data sets
//*            ---- XSFNTATT is a VSAM dataset containing FONT and LOGO
//*            ATTRIBUTES
//XSFNATT      DD DSN=XPRM.V3R2M00.XSFNTATT,DISP=SHR
//*            ---- XSFORMS is a VSAM dataset containing FORMS
//XSFORMS      DD DSN=XPRM.V3R2M00.XSFORMS,DISP=SHR
//*            ---- XSFRMATT is a VSAM dataset containing FORM ATTRIBUTES
//XSFRMATT     DD DSN=.XPRM.V3R2M00.XSFRMATT,DISP=SHR
//XSSTUPTAB    DD DSN=.XPRM.V3R2M00.XSSTUPTAB,DISP=SHR
//*
//*            ****THE FOLLOWING JCL IS FOR DOWNLOAD AND SAMPLE
//STEP2        EXEC PGM=XPRMUTIL,COND=(8,LE,STEP1)
//STEPLIB      DD DSN=XPRM.V3R2M00.LOAD,DISP=SHR
//              DD DSN=XPRM.V3R2M00.SASCRDLB.C500,DISP=SHR
//SYSPRINT     DD SYSOUT=*
//SYSTEM       DD SYSOUT=*
//XSUACTL      DD SYSOUT=*          ACTIVITY LOG
//XSFUNTS      DD DSN=XPRM.V3R2M00.XSFUNTS,DISP=SHR
//XSFNATT      DD DSN=XPRM.V3R2M00.XSFNTATT,DISP=SHR
//XSFORMS      DD DSN=XPRM.V3R2M00.XSFORMS,DISP=SHR
//XSFRMATT     DD DSN=XPRM.V3R2M00.XSFRMATT,DISP=SHR
//XSSTUPTAB    DD DSN=XPRM.V3R2M00.XSSTUPTAB,DISP=SHR
//ddnamea      DD SYSOUT=*,DCB=(RECFM=FBA),
//              DEST=prtrid
//XSUIN        DD *
//              REPRO FORM WHERE(NAME=XVERL) OUTDEV(HCPRNTR) SAMPLE;
//*

```

Accessing compiled forms

Use the XPRMUTIL utility to access a compiled form for deletion, sample printing, or downloading to Xerox LPS printers. If you are not familiar with this utility, contact your System Administrator or refer to the *Xerox Print Resources Manager/Host Forms Description Language 3.2 for IBM MVS Managing Resources* manual.

Editing an existing form

If you want to edit an existing form, select the forms source language (FSL) with your ISPF editor, make the necessary changes, and then compile your edited form with HFDL.

Specifying monochrome forms

Specifying the monochrome mode for a Xerox highlight color LPS printer allows you to process monochrome forms (using only the black primary color) much faster and at a higher quality than printing forms in version 2 or 10 format. If you use the monochrome mode, you also reduce the toner consumption of a Xerox highlight color LPS printer. You can specify the monochrome mode in the input Parm (**VERSION=HC**) if you want to compile and print a monochrome form on a Xerox highlight color LPS printer.

To print a monochrome form on a Xerox full color printer, you specify the version **X1**.

This chapter lists the error messages that appear when problems occur, an indication of the cause of the message, and the action you can perform to correct the problem.

Message indicators

The letters in the error numbers indicate the severity of the error as follows:

- E (error) indicates a syntax or similar error.
- F (fatal) indicates a serious error that can prevent you from compiling your form or may abnormally end (ABEND) your session.
- I (informative) indicates the status or activity of the system where no action is needed by the operator.
- W (warning) indicates an error in the description of your form. It also indicates the action you may be required to take to correct the problem.

XFG6001F No parameters found

Indication	There were no parameters in the EXEC statement of the JCL. There is nothing to compile.
Action	Correct and recompile the form.

XFG6002F Unidentified parameter keyword <parameter keyword>

Indication	The parameter keyword encountered could not be identified.
Action	Correct and recompile the form.

XFG6003F Unidentified parameter value <parameter value>

Indication	A parameter value entered in the EXEC statement is invalid.
Action	Correct and recompile the form.

XFG6004F No FSL specified

Indication	No FSL file specified in the JCL parm.
Action	Modify the JCL to include the FSL you want to compile.

XFG6005W Group name too long, using <name>

Indication The group name used exceeds the maximum 8 characters allowed.

Action None required. The system truncates the identifier at 8 characters.

XFG6006W Member/Form name <name> is too long

Indication The member or form name specified in the parm statement is too long. The member name is only 8 characters or less, and the form name is only 6 or less.

Action Correct the member/form name in the parm statement.

XFG6007F DDname <name> is not defined

Indication The DDname is not defined in the JCL.

Action Modify the JCL to include the missing DDname.

XFG6008F Dsname <name> does not exist

Indication The Dsname defined by the DDname does not exist in the system.

Action Check and modify the JCL.

XFG6009F Invalid paper size specified <paper size>

Indication the paper size encountered could no be identified.

Action Make sure that the paper size value is one of the following: LT, LG, A3, A4, or B4, and then rerun the job.

XFG6010F Keymap id too long <keymap id>

Indication The compiler parsed a keymap id that exceeded the maximum eight characters.

Action Specify a keymap id that is eight characters or less.

XFG6011F Cannot open batch file <file name>

Indication There is a serious system error. The compiler was not able to open the immediate IFF file created by the compiler.

Action Verify that the file name is correct and rerun the job. If the problem persists, contact your System Administrator.

XFG6012F Batch files nested too deep

Indication The maximum number of nested batch files is now only 4.

Action Correct the batch file such as the deepest nested level to 4 or less.

XFG6013F Cannot open FSL

Indication There is a serious system error. The compiler was not able to open the input FSL file.

Action Verify that the file name is correct and rerun the job. If the problem persists, contact your System Administrator.

XFG6014F Cannot open scratch file <file name>

Indication There is a serious system error. The compiler was not able to obtain a scratch file used during compilation.

Action Verify that the file name is correct and rerun the job. If the problem persists, contact your System Administrator.

XFG6020I There are N forms compiled

Indication There are n forms compiled in this job, either through multi-form or multi-member selection.

Action None.

XFM8nnn Internal Failure

Indication There is a serious internal error.

Action Contact your System Administrator and indicate the error number. These are the descriptions of the possible error numbers:

error numbers	Description
-000	FRM Generator completed successfully
-001	Invalid <grid record> format
-002	Memory Allocation Error
-003	Open File Error
-004	Invalid <IFF version> record format
-005	Invalid <device code> record format
-006	Invalid <resolution> record format
-007	Invalid <form version> record format
-008	Invalid <orientation> record format
-009	Invalid <size record> format
-010	Invalid <origin record> format
-011	Invalid <page number record> format
-012	FRM Generator failed

error numbers	Description
-013	Invalid <line record> format
-014	Invalid <box record> format
-015	Invalid <text record> format
-016	Invalid format
-017	Invalid <logo record> format
-018	Invalid <graphic record> format
-019	Invalid <picture record> format
-020	Data overflow
-021	Insufficient input parameters
-022	Resources overflow
-023	Error writing FRM file
-024	Not found
-025	Error opening IFF file
-026	Error opening FRM file
-027	Invalid <name record> format
-028	Unexpected IFF command sequence
-029	Invalid <ink> command format
-030	Invalid <palette/catalog> command format
-031	Invalid <ink priority> command format

XFP1001W Discarding excess FDL

Indication	Excess FSL statements were found beyond the END statement.
Action	Verify that these statements do not belong in the form, then move or delete them.

XFP1002W Semicolon expected

Indication	A semicolon was expected in the statement. The compiler inserted one and continued with the compile.
Action	None required.

XFP1003W Character not in FDL alphabet

- Indication** A character was found in the FSL statements that does not exist in the FSL alphabet. The character was ignored.
- Action** Verify that the character does not belong in the FSL statements, then move or delete it.

XFP1004W DRAW N and REPEAT EVERY disagree, using 2

- Indication** The number of lines entered for the DRAW command must be greater than 1. The compiler substituted 2 for the invalid existing number.
- Action** Enter the correct values and recompile the form.

XFP1005W Undefined Section Name <name>, command ignored

- Indication** The section name used has not been defined.
- Action** Verify that the section name was spelled correctly or make sure the section is defined prior to invoking it with the **DO SECTION** command.

XFP1006W Invalid line weight, using HAIRLINE

- Indication** An invalid line weight was entered. The compiler substituted a value of **HAIRLINE**.
- Action** If **HAIRLINE** is not the value you want, correct the FSL and recompile the form.

XFP1007W Invalid border weight, using HAIRLINE

- Indication** An invalid border weight was entered for a box. The compiler substituted a value of **HAIRLINE**.
- Action** If **HAIRLINE** is not the value you want, enter a valid border weight.

XFP1008W Possible missing or unmatched quotes

- Indication** The compiler cannot detect a closing quote.
- Action** Make sure that all quotes in the preceding commands in your FSL are matched.

XFP1009E 600 spi cannot be used with highlight color form

- Indication** There is no Xerox printer support for highlight color in 600 spi mode.
- Action** Use 300 spi for color forms or use black and white mode for 600 spi.

XFP1010W Box not found

- Indication** A box with the origin specified, or the **NEXT BOX** could not be found.
- Action** Recalculate the box coordinates. To ensure the box is found, use dot coordinates for the box origin.

XFP1012W Unexpected fraction, using integer

- Indication** A fraction was entered where an integer was expected. The integer part of the number is used.
- Action** Change fraction to an integer value and recompile the form.

XFP1014W Invalid resolution, using 300 spi

- Indication** An invalid resolution was entered in the **RES** statement.
- Action** Change the resolution to 300 or 600 spi.

XFP1015W Unexpected end of text (terminating code expected)

- Indication** The terminating code character is missing at the end of the text string. If you entered the text correctly, this could indicate a damaged file.
- Action** Reenter the text with the terminating code. If the problem persists, contact your System Administrator.

XFP1016W Text will not fit in column

- Indication** The entered text is too long to fit in the specified column.
- Action** Reduce the length of the text string or increase the size of the column to fit the text. Otherwise the text will not be printed.

XFP1018W Text might not fit in box

- Indication** The entered text block exceeds or is equal to the dimensions of the box in which it is to be placed.
- Action** Check the printed form. If text crosses the boundary of the box, reduce the length of the text string or increase the size of the box to fit the text.

XFP1020W Missing END statement

- Indication** End of file was found with no **END** statement.
- Action** No action is required, since the **END** statement is optional. You may want to add the **END** statement and recompile.

XFP1021W Setup command out of sequence

- Indication** A setup command (**FORM/RESOLUTION**, **FONT**, **GRID**, **LANDSCAPE/PORTRAIT**, **INK**, **ICATALOG**, or **PALETTE**) was entered after a formatting command (such as **LINE**, **TEXT**, **BOX**).
- Action** Place the setup command with the rest of the setup commands at the beginning of the form source and after the **FORM/RESOLUTION** command.

XFP1022W Identifier name too long, using <name>

- Indication** An identifier was used that exceeds the maximum 16 characters allowed.
- Action** None required. The system truncates the identifier at 16 characters.

XFP1024W Section name too long, using <name>

- Indication** The section name used exceeds the maximum 16 characters allowed.
- Action** None required. The system truncates the section name at 16 characters.

XFP1025W Logos not supported for XES forms

- Indication** You used the **LOGO** command while generating an XES form. In this release, logos are treated as fonts in XES forms.
- Action** Do not use the **LOGO** command while generating XES forms or, if possible, enter the logo as a font.

XFP1026W Catalog name too long, using <six-character catalog name>

- Indication** The catalog name used exceeds the maximum six characters allowed.
- Action** None required. The system truncates the catalog name at six characters.

XFP1027W Dot Leader too big for column

- Indication** You specified a dot leader that exceeds the column setting.
- Action** Change the dot leader or the column setting.

XFP1028W Text String Longer than 256 characters

- Indication** A text string contains more than the maximum 256 characters.
- Action** Reduce the size of the text string.

XFP1029W No substitution allowed for Logo <logo name>

- Indication** You entered the **USING INK** parameter in the **LOGO** command when the NO SUBSTITUTION flag was set for the specified logo in the logo resource.
- Action** Do not use the **USING INK** parameter with this logo.

XFP1030W Can not close data base

- Indication** The compiler is not able to close the database because it is being used by another user.
- Action** Try again in a few minutes. If the problem persists, note the return code number and contact your System Administrator.

XFP1031W Palatte name too long, using <thirty-two-character palatte name>

- Indication** The palatte name used exceeds the maximum thirty-two characters allowed.
- Action** None required. The system truncates the palatte name at thirty-two characters.

XFP1032W Column off right of physical page

- Indication** The specified column width exceeds the physical page boundary.
- Action** Resize the column width to fit the physical page boundary.

XFP3001E Syntax error on <command segment>

- Indication** The specified command segment contains a syntax error.
- Action** Correct the error and recompile the form.

XFP3002E DRAW N and REPEAT AT counts disagree

- Indication** The **DRAW** and **REPEAT AT** counts do not agree.
- Action** Correct and recompile the form.

XFP3003E N TIMES and REPEAT AT counts disagree

- Indication** The number of **TIMES** and the **REPEAT AT** counts do not agree.
- Action** Correct and recompile the form.

XFP3004E Horizontal text inconsistent with top/bottom

- Indication** Horizontal text is aligned **TOP** or **BOTTOM** instead of **LEFT** or **RIGHT**.
- Action** Correct the text direction with the proper alignment.

XFP3005E Vertical text inconsistent with left/right

Indication Vertical text is aligned **LEFT** or **RIGHT** instead of **TOP** or **BOTTOM**.
Action Correct the text direction with the proper **ALIGNMENT**.

XFP3006E Identifier exceeds 6 characters

Indication An identifier name, such as **FORM** name, exceeded six characters.
Action Correct the identifier name and recompile the form.

XFP3007E Stacked text and justify conflict

Indication Stacked text cannot be justified.
Action Remove the **JUSTIFY** and recompile the form.

XFP3008E Setup command not allowed within section

Indication One of the setup commands such as **FONT** or **INK** was found within a section.
Action Move the setup command to the beginning part of the FSL.

XFP3009E Code point <code point> in font has a zero width

Indication Some of the characters in this font have a zero width value.
Action Reload the font.

XFP3010E DO SECTION not allowed within section

Indication A **DO SECTION** command was found within a section.
Action Remove the **DO SECTION** command and recompile the form.

XFP3011W Ambiguous positioning of <position>, <position> given. <position>, <position> used

Indication Ambiguous positioning of text in a box such as Left, Left instead of Left with Bottom, Center, or Top.
Action Correct the ambiguous position.

XFP3012E BEGIN SECTION not allowed within section

Indication A **BEGIN SECTION** command was found within a section.
Action Remove the command or end the current section.

XFP3014E Value out of range

- Indication** A zero value or a value that exceeds page limits was encountered.
- Action** Correct the value to one within the valid limits.

XFP3015E Keyword out of context/sequence

- Indication** You sequenced a keyword incorrectly.
- Action** Enter the keyword in the correct order.

XFP3016E FONT command not defined

- Indication** Text is referenced in the form and no **FONT** command was specified.
- Action** Enter the **FONT** command with the valid fonts you want to use in the form.

XFP3017E Invalid font index

- Indication** The font index in the **TEXT** command is outside the range of those defined in the **FONT** command.
- Action** Correct the font number to a value within the range of fonts defined, or define a new font if the font is missing from the font list.

XFP3018E INK command not defined

- Indication** Inks are referenced in the form and there is no **INK** command specified.
- Action** Enter the **INK** command with the valid inks you want to use in the form.

XFP3019E Invalid ink index

- Indication** The ink index referenced in a **LINE**, **TEXT**, **BOX**, **GRAPHIC**, or **LOGO** command is outside the range of those defined in the **INK** command.
- Action** Correct the ink number so that it is within the range of inks defined or define a new ink if the ink is missing from the ink list.

XFP3020E Line off (top/bottom/left/right) of physical page

- Indication** The line went off either the top, bottom, left, or right of the physical page.
- Action** Correct the line coordinates so that it does not extend beyond the physical page.

XFP3021E Font name exceeds 20 characters

Indication The font name exceeded the 20 character maximum length.

Action Use a valid font name, which should consist of 20 characters or less.

XFP3024E Line end coordinate <= start coordinate

Indication The line ending coordinate is less than or equal to the line starting coordinate.

Action Correct the line coordinates and recompile the form.

XFP3030E Box off (top/bottom/left/right) of physical page

Indication The box went off either the top, bottom, left, or right of the physical page.

Action Correct the box coordinates so the box does not extend beyond the physical page.

XFP3034E Box width is zero

Indication The width of the box was zero.

Action Correct the box width and recompile the form.

XFP3035E Box height is zero

Indication The height of the box was zero.

Action Correct the box height and recompile the form.

XFP3040W Text off (top/bottom/left/right) of physical page

Indication The text went off either the top, bottom, left, or right of the physical page.

Action Correct the text coordinates so the text does not extend beyond the physical page.

XFP3050W Logo off (top/bottom/left/right) of physical page

Indication The logo went off either the top, bottom, left, or right of the physical page.

Action Correct the logo coordinates so the logo does not extend beyond the physical page.

XFP4001F Unexpected end of file

Indication The **END** statement was encountered before any description commands. There is nothing to compile.

Action Move the **END** command to the last line of the FSL and recompile the form.

XFP4006F Cannot open FSL

- Indication** There is a serious system error. The compiler was not able to open the input FSL file.
- Action** Verify that the file name is correct and rerun the job. If the problem persists, contact your System Administrator.

XFP4009F Cannot open IFF

- Indication** There is a serious system error. The compiler was not able to open the immediate IFF file created by the parser.
- Action** Contact your System Administrator.

XFP4010F Grid format not defined <format name>

- Indication** The format specified in the **GRID** command does not exist.
- Action** Correct and recompile the form.

XFP4012F Too many fonts specified

- Indication** The **FONT** command specified more than the limit of 32 fonts.
- Action** Delete the number of fonts used so the maximum of 32 is not exceeded and recompile the form.

XFP4014F Cannot open scratch file <file name>

- Indication** There is a serious system error. The compiler was not able to obtain a scratch file used during compilation.
- Action** Verify that the file name is correct and rerun the job. If the problem persists, contact your System Administrator.

**XFP4020F (300/600) spi font not found **

- Indication** One of the fonts in the indicated font list is not a 300 or 600 spi font.
- Action** Load the necessary font into the font database or correct the **FONT** command with the correct name and recompile the form.

XFP4022W Logo not found <logo name>

- Indication** The logo specified in the **LOGO** command was not found in the database.
- Action** Correct the name of the logo or load the required logo into the database and recompile the form.

XFP4023F Undefined font index

- Indication** A font index was referenced in a **TEXT** command that was not defined in the font list.
- Action** Correct the **TEXT** command to the correct reference from the font list.

XFP4030F**Database access fault: mode =<return code>**

Indication There is a serious system error. The compiler was not able to access the resource database for the reason based on the return code.

Action Contact your System Administrator with the return code (**rc**) number. These are the descriptions of the possible return codes:

Return code	Description
-161	Allocation error
-162	Fatal error doing BLOB add
-163	Fatal error deallocating resource
-164	File corrupted
-165	Fatal file system error
-166	Fatal file system error (see error codes)
-167	Missing segment, no BLOB access
-168	DB dictionary not successfully loaded
-169	Programming error detected
-170	Duplicate record
-171	End of file
-172	Record does not exist
-173	Operation successful
-174	Error deleting attribute record
-175	Error updating attribute record
-176	Error deleting BLOB, attribute record deleted
-177	BLOB maximum size has been exceeded
-178	In mass delete, record not deleted, OPEN error
-179	Could not access BLOB for READWRITE
-180	Unable to move BLOB data
-181	Data list not found
-182	Duplicate resource
-183	Resources to deallocate not allocated
-184	File already opened
-185	File does not exist
-186	Non-secured file not open for READWRITE

Return code	Description
-187	File not open
-188	File system warning (see error codes)
-189	Identifier already exists
-190	Invalid array specification
-191	Invalid array starting position
-192	Invalid buffer length
-193	Invalid buffer pointer
-194	Invalid value, buffer pointer
-195	Invalid CID
-196	Invalid data list resource
-197	Invalid data set name
-198	Invalid element data
-199	Invalid element for resource
-200	Invalid element type
-201	Invalid element pointer
-202	Invalid file organization
-203	Invalid file type
-204	Invalid file handle
-205	Invalid variable, identifier pointer
-206	Invalid key length
-207	Invalid offset pointer
-208	Invalid array size pointer
-209	Invalid key pointer
-210	Invalid mode
-211	Invalid logical operator
-212	Invalid logical operator pointer
-213	Invalid query resource
-214	Invalid query syntax
-215	Query syntax incomplete
-216	Invalid relational operator
-217	Invalid relational operator

Return code	Description
-218	Invalid read mode
-219	Invalid resource
-220	Invalid resource pointer variable
-221	Invalid return pointer variable
-222	Invalid security type flag
-223	Invalid element value
-224	Lock request failed
-225	Validation check failed
-226	Maximum query lists has been exceeded
-227	Maximum data lists has been exceeded
-228	Missing data on a retrieval request
-229	Missing segment number
-230	Security request made on resource with no BLOBs
-231	Updates to key elements not allowed
-232	None of the selected set of records updated in mass modification
-233	Updates to protected elements not allowed
-234	No record found for request
-235	Unable to open files
-236	Partial set of records updated in mass modification
-237	Query list not found
-238	Query stack full
-239	"Wait for lock" value out of range
-240	Record buffer size exceeded
-241	Record maximum size has been exceeded
-242	Record truncated
-243	Request for data beyond end of array
-244	Resource not allocated for READWRITE
-245	Access security denied
-246	Secured file not open for READWRITE

Return code	Description
-247	Outstanding resource sharing already exists
-248	Access security errors detected but processing continued
-249	Data stack full
-250	Duplicate element in list
-251	Duplicate automatically generated data list
-252	Verification routine has detected an invalid element value
-253	Data dictionary not loaded
-254	Caller supplied an invalid resource name
-255	Caller supplied an invalid element name
-256	Caller supplied too many occurrences of element values to be validated
-257	Invalid identifier variable value
-258	No current position established for resource, transaction invalid
-259	No current record established for resource, transaction invalid
-260	A single record must be specified for transaction
-261	Requested resource not allocated for reading
-262	Resource requested not allocated for use
-263	Caller supplied NULL file pointer for data dictionary list file
-264	Database dictionary and database data set are incompatible
-265	Attribute file only opened
-266	Attribute and two BLOB files opened
-267	Attribute file and secured BLOB opened
-268	Attribute file and non-secured BLOB opened
-269	Invalid access type requested for resource
-270	Invalid resource count
-271	All physical files opened successfully

Return code	Description
-272	Physical files containing attribute records opened
-273	Physical files containing attribute and secured BLOB opened
-274	Physical files containing attribute and non-secured BLOB opened
-275	Partial deallocation has occurred
-276	No outstanding resource sharing currently exists
-277	Database verification KEY1
-278	Database verification DD name
-279	Database verification file segment
-280	Database verification key size
-281	Database verification record size
-282	Database verification version
-283	Database verification revision
-284	Database verification resource name
-285	Database verification element segment
-286	Database verification number of elements
-287	Database verification element name
-288	Database verification element tag
-289	Database verification element type
-290	Database verification element flag
-291	Database verification element size
-292	Could not access BLOB for READ
-293	Invalid query length size

XFP4034F Invalid graphic scale factor

Indication	An incorrect graphic scaling factor was entered. The scaling factor must be an integer in the range 1 to 8.
Action	Correct the scaling factor and recompile the form.

XFP4040F Text string exceeds buffer limits

- Indication** The text string entered exceeded the text buffer limits of 4096 characters.
- Action** Shorten the text string by breaking the text up into two or more **TEXT** commands.

XFP4050F Keyboard map not found

- Indication** The keyboard map specified in the compile command line was not in the resource database.
- Action** Correct the keyboard map name if it is incorrect or add the specified keyboard map to the resource database.

XFP4051F Keyboard map error: rc= <return code>

- Indication** The compiler could not find a valid font character map in the database.
- Action** Load font character maps into the database for the fonts used in this form.

XFP4052F Keymap id too long <keymap id>

- Indication** The compiler parsed a keymap id that exceeded the maximum eight characters.
- Action** Specify a keymap id that is eight characters or less.

XFP4053F Maximum number of inks allowed (255) exceeded

- Indication** Your FSL uses more than the maximum 255 inks (colors) allowed.
- Action** Reduce the number of inks in your FSL to 255 or less.

XFP4054F Cannot reassign identifier to itself

- Indication** You made an invalid use of the **LET** command. An identifier cannot be on both the right-hand and left-hand side of the assignment statement.
- Action** Remove the identifier from the right-hand side of the assignment statement.

XFP4062F Invalid LPS tape label

- Indication** There is a serious system error. The compiler was not able to process the tape label of the compiled form it just created.
- Action** Contact your System Administrator.

XFP4063F Data List Error

- Indication** There is a serious system error.
- Action** Contact your System Administrator.

XFP4064F Query List Error

Indication There is a serious system error.
Action Contact your System Administrator.

XFP4065F Invalid XES Tape Label

Indication There is a serious system error. The compiler was not able to process the tape label or the compiled form it just created.
Action Contact your System Administrator.

XFP4066F DELETE old Attribute Error

Indication There is a serious system error.
Action Contact your System Administrator.

XFP4067F Attribute data list Error

Indication There is a serious system error.
Action Contact your System Administrator.

XFP4068F ADD Attribute Error

Indication There is a serious system error.
Action Contact your System Administrator.

XFP4069F BLOB Data List Error

Indication There is a serious system error.
Action Contact your System Administrator.

XFP4070F Can not add 1st BLOB

Indication There is a serious system error.
Action Contact your System Administrator.

XFP4071F Can not add BLOB

Indication There is a serious system error.
Action Contact your System Administrator.

XFP4080F The <database name> resource is in use by another user

Indication The database is locked and in use by another user.
Action Try to compile your form later.

XFP4081F Can not open Database. Check DB reference.

- Indication** The database cannot be opened. There may be a database reference error.
- Action** Check your JCL for database reference errors or contact your System Administrator.

XFP4082F DB security fault. No access privilege for user.

- Indication** The database has been secured and you do not have access privileges.
- Action** Contact your System Administrator.

XFP4084F Invalid virtual keyboard character map <name>

- Indication** You specified an unknown virtual keyboard character map.
- Action** Use a valid keyboard character map. See the "Virtual keyboard and font character maps" appendix in the *Xerox Print Resources Manager/Host Forms Description Language 3.2 for IBM MVS Managing Resources* manual.

XFP4085F Font character map not found for font <name>

- Indication** You specified a character map for this font which cannot be found.
- Action** Contact your System Administrator.

XFP4086F Virtual keyboard character map not found <map name>s

- Indication** You specified a virtual keyboard character map for this font which cannot be found.
- Action** Contact your System Administrator.

XFP4087F No character map is yet specified for font s

- Indication** No character map has been specified for this font.
- Action** Contact your System Administrator.

XFP5000F Memory allocation fault: rc=

- Indication** There is a serious system error. The compiler was not able to process the tape label of the compiled form it just created.
- Action** Contact your System Administrator.

XFP8nnn Internal Failure

- Indication** There is a serious internal error.
- Action** Contact your System Administrator and indicate the error number. Refer to error code XFM8nnn for the descriptions of the possible return codes.

XFU4030F Database access fault: mode=<return code>

Indication There is a serious system error. The compiler was not able to access the resource database for the reason based on the return code.

Action Contact your System Administrator with the return code (**rc**) number. Refer to error code XFP4030F for the descriptions of the possible return codes.

XFU4060F Can not open FRM

Indication There is a serious internal error.

Action Contact your System Administrator and indicate the error number.

XFU4061F Invalid FRM format

Indication There is a serious internal error.

Action Contact your System Administrator and indicate the error number.

XFU4062F Invalid LPS tape label

Indication There is a serious internal error.

Action Contact your System Administrator and indicate the error number.

XFU4063F Data List Error

Indication There is a serious internal error.

Action Contact your System Administrator and indicate the error number.

XFU4064F Query List Error

Indication There is a serious internal error.

Action Contact your System Administrator and indicate the error number.

XFU4065F Invalid XES tape label

Indication There is a serious internal error.

Action Contact your System Administrator and indicate the error number.

XFU4066F DELETE old Attribute Error

Indication There is a serious internal error.

Action Contact your System Administrator and indicate the error number.

XFU4067F Attribute data list Error

Indication	There is a serious internal error.
Action	Contact your System Administrator and indicate the error number.

XFU4068F ADD Attribute Error

Indication	There is a serious internal error.
Action	Contact your System Administrator and indicate the error number.

XFU4069F BLOB Data List Error

Indication	There is a serious internal error.
Action	Contact your System Administrator and indicate the error number.

XFU4070F Can not add 1st BLOB

Indication	There is a serious internal error.
Action	Contact your System Administrator and indicate the error number.

XFU4071F Can not add BLOB

Indication	There is a serious internal error.
Action	Contact your System Administrator and indicate the error number.

XFU4072F Can not DELETE old Record. Delete via XPRM UTIL

Indication	There is a serious internal error.
Action	Contact your System Administrator and indicate the error number.

XFX0000I FRM Generator completed successfully

Indication	The XES generator completed successfully.
Action	None.

XFX0001F XES generator failed, Invalid color format detected for ink tag.

Indication	There is a programming error.
Action	Contact your System Administrator.

XFX0002F Memory allocation error

Indication The system has failed to acquire sufficient memory.
Action Increase your memory size for the job or contact your System Administrator.

XFX0003F Open file error

Indication The system has failed to open a file.
Action Contact your System Administrator.

XFX0004F Invalid <IFF version> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

XFX0005F Invalid <device code> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

XFX0006F Invalid <name record> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

XFX0007F Invalid <resolution> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

XFX0008F Invalid <form version> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

XFX0009F Invalid <orientation> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

XFX0010F Invalid <size> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

AFX0011F Invalid <origin> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

AFX0012F Invalid <grid> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

AFX0013F Invalid <line> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

AFX0014F Invalid <box> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

AFX0015F Invalid <text> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

AFX0016F Invalid record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

AFX0018F Invalid <graphic> record format

Indication The system found an error in the IFF file.
Action Contact your System Administrator.

AFX0020F Insufficient input parameters

Indication There is a programming error.
Action Contact your System Administrator.

AFX0022F Error opening IFF file

Indication The system failed to open the IFF file.
Action Make sure your JCL contains the DD statement //XFIFO.

XFX0023F Error opening XES file

Indication The system failed to open the IFF file.
Action Make sure your JCL contains the DD statement //XFXES.

XFX0024F Error writing XES file

Indication There is a programming error.
Action Contact your System Administrator.

XFX0025F Unexpected IFF command sequence

Indication There is a programming error.
Action Contact your System Administrator.

XFX0026F Error sorting IFF File

Indication There is a programming error.
Action Contact your System Administrator.

XFX0027F Error Opening Sorted IFF File

Indication The system failed to open a temporary sorted IFF file.
Action Contact your System Administrator.

XFX0028F Error Opening Temporary Sorted Work File

Indication The system failed to open a temporary sorted work file.
Action Contact your System Administrator.

XFX0029W Keyboard name truncated to MAX size

Indication You specified a keyboard name of more than eight characters.
Action Use the keyboard name specified when you invoke HFDL.

XFX0031I XES Generator message log started on DAY MON dd hh:mm:ss yyyy

Indication The system wrote the XES Generator message to the indicated log.
Action Refer to the indicated error log for the message.

XFX0032F Fatal program error ... invalid message number

Indication There is a programming error.
Action Contact your System Administrator.

**XFX0033W Possible print errors ... no character map for font **

- Indication** The program could not locate a character map in the XPRM font database for the indicated font and keyboard combination.
- Action** The program continues with the last used character map.

XFX0034F Error initializing to XPRM databases

- Indication** The system failed to open one or more XPRM databases.
- Action** Make sure the XPRM databases are allocated in your JCL.

XFX0035F Fatal error return from XPRM API.

- Indication** There is a programming error.
- Action** Contact your System Administrator.

XFX0036W Error terminating XPRM

- Indication** The system located an error when closing one or more XPRM databases.
- Action** Refer to the system messages.

XFX0038F Error reading ink header record

- Indication** The system found an error in the IFF file.
- Action** Contact your System Administrator.

XFX0039F Error reading ink name record

- Indication** The system found an error in the IFF file.
- Action** Contact your System Administrator.

XFX0040W Missing XPRM color map ... ink number <ink tag>

- Indication** The system cannot locate an ink match in the XPRM color map database.
- Action** The system is substituting black ink.

**XFX0041F Invalid font orientation for font **

- Indication** The XPRM database contains an invalid font orientation attribute value.
- Action** Check the database for the listed font.

XFX0042W Assigning space to hex value <xx>

- Indication** The FSL contains a HEX ASCII value in !xx format and <xx> cannot be translated to EBCDIC using the current character map.
- Action** The system is substituting EBCDIC SPACE ox40.

EXIT **Terminating due to internal error**

Indication	There is a serious internal error.
Action	Contact your System Administrator.

A.

System considerations

Consider the following items when designing and planning your form:

- Line density limits
- Local density limits
- Superimposed edges when adjoining boxes
- Text size in boxes
- Lines with sections.

These items are especially important when designing complex forms. You can avoid error messages and save the time it takes to correct the problems by being aware of the system considerations described in this appendix.

Line density limits

There is a limit to the number of characters and lines you can image on a page. The basic limit is measured in units called dispatchable items.

A dispatchable item has a maximum of 32 dots and can be the following:

- Full character—of point size 8, 9 or less
- Fraction of a character—greater than point size 8 or 9
- Line segment—equal to or smaller than 32 dots in the scan direction.

Local density and page setup errors

You may encounter a page setup error while the printer is imaging a page. This error indicates there was not enough time for the printer to image a page. One of the following conditions may have caused the problem:

- Excessive amount of data to be imaged
- Disk errors
- Imaging problems
- Local density problem.

If you design a form to avoid exceeding line density limitations, you may cause page setup errors because of local density. Long lines of small text set with close spacing cause a local density problem. The imaging hardware accepts and dispatches characters at two different rates. If the form structure imposes too heavy a dispatchable load, the hardware may run out of input before all of the data has been processed. You rarely encounter this problem in readable forms.

If you have described a form that causes a page setup error, examine both the form and the variable data to see if there are many lines or a large amount of text in a small area. If you encounter this type of error, reduce the density of the area by spreading the material out on the form or by deleting parts of the form.

Boxes and their image

You can draw boxes with just a single box statement or with line-by-line statements. The box statement may not work correctly if it specifies two boxes adjoined by a single side. This causes the adjoining side lines of the boxes to become superimposed. Since the two lines use the same pair of coordinates, a line density problem may occur.

Text in boxes

If you are using a **TEXT IN BOX** command, HFDL determines if the given text can fit in the box. If your text font size exceeds the maximum allowable size, HFDL generates an error message (XFP1018W). If you see this error you have the following choices:

- Make the box larger.
- Reduce the number of lines.
- Use a smaller font.
- Use a **TEXT AT** command rather than a **TEXT IN BOX** command if the text fits visually.

Refer to the "Messages" chapter for more information.

Using lines within sections

When you create a form by invoking defined sections, make sure you do not begin and end lines at section boundaries. It is more efficient to allow the line to continue through a series of sections.

B. FMT grid format properties

This appendix lists the **FMT** parameters you use in the **LANDSCAPE/PORTRAIT** and **GRID** commands. It includes the FMT properties for 8.5 by 11-inch paper, 8.5 by 14-inch paper, and A4 paper. The **FMT** defaults are **FMT1** (landscape) and **FMT6** (portrait).

Table B-1 lists the grid and origin substitutions you can use when you print on 8.5 by 11-inch paper.

Table B-1. FMT properties for 8.5 by 11-inch paper

Specification	Substitution
FMT1	LANDSCAPE; GRID UNIT IS 13.6 CPI, 8.1 LPI, ORIGIN 0.18 INCH, 0.66 INCH;
FMT2	LANDSCAPE; GRID UNIT IS 15 CPI, 8.1 LPI, ORIGIN 0.18 INCH, 0.50 INCH;
FMT3	LANDSCAPE; GRID UNIT IS 13.6 CPI, 10.7 LPI, ORIGIN 0.14 INCH, 0.66 INCH;
FMT4	LANDSCAPE; GRID UNIT IS 15 CPI, 10.7 LPI, ORIGIN 0.14 INCH, 0.50 INCH;
FMT5	LANDSCAPE; GRID UNIT IS 10 CPI, 6 LPI, ORIGIN 0.17 INCH, 0.50 INCH;
FMT6	PORTRAIT; GRID UNIT IS 13.6 CPI, 8.1 LPI, ORIGIN 0.57 INCH, 0.58 INCH;
FMT7	PORTRAIT; GRID UNIT IS 12 CPI, 6 LPI, ORIGIN 0.50 INCH, 0.50 INCH;
FMT8	PORTRAIT; GRID UNIT IS 10 CPI, 6 LPI, ORIGIN 0.50 INCH, 0.50 INCH;
FMT9	LANDSCAPE; GRID UNIT IS 20 CPI, 10 LPI, ORIGIN 0.25 INCH, 0.25 INCH;
FMT10	PORTRAIT; GRID UNIT IS 17.6 CPI, 12.5 LPI, ORIGIN 0.22 INCH, 0.51 INCH;
FMT11	PORTRAIT; GRID UNIT IS 20 CPI, 12.5 LPI, ORIGIN 0.22 INCH, 0.50 INCH;

Table B-2 lists the grid and origin substitutions you can use when you print on 8.5 by 14-inch paper.

Table B-2. **FMT properties for 8.5 by 14-inch paper**

Specification	Substitution
FMT12	LANDSCAPE; GRID UNIT IS 13.6 CPI, 8.1 LPI, ORIGIN 0.18 INCH, 0.66 INCH;
FMT13	PORTRAIT; GRID UNIT IS 13.6 CPI, 8.1 LPI, ORIGIN 0.57 INCH, 0.58 INCH;

Table B-3 lists the grid and origin substitutions you can use when you print on A4 paper.

Table B-3. **FMT properties for A4 paper**

Specification	Substitution
FMT1A	LANDSCAPE; GRID UNIT IS 12.5 CPI, 8.3 LPI, ORIGIN 0.18 INCH, 0.57 INCH;
FMT2A	LANDSCAPE; GRID UNIT IS 14.3 CPI, 8.3 LPI, ORIGIN 0.18 INCH, 0.60 INCH;
FMT3A	LANDSCAPE; GRID UNIT IS 12.5 CPI, 11.1 LPI, ORIGIN 0.18 INCH, 0.57 INCH;
FMT4A	LANDSCAPE; GRID UNIT IS 14.3 CPI, 11.1 LPI, ORIGIN 0.18 INCH, 0.60 INCH;
FMT5A	LANDSCAPE; GRID UNIT IS 10.0 CPI, 6.0 LPI, ORIGIN 0.22 INCH, 0.85 INCH;
FMT6A	PORTRAIT; GRID UNIT IS 13.6 CPI, 8.1 LPI, ORIGIN 0.91 INCH, 0.46 INCH;
FMT7A	PORTRAIT; GRID UNIT IS 12.0 CPI, 6.0 LPI, ORIGIN 0.85 INCH, 0.39 INCH;
FMT8A	PORTRAIT; GRID UNIT IS 10.0 CPI, 6.0 LPI, ORIGIN 0.85 INCH, 0.39 INCH;
FMT9A	LANDSCAPE; GRID UNIT IS 20.0 CPI, 10.0 LPI, ORIGIN 0.14 INCH, 0.85 INCH;
FMT10A	PORTRAIT; GRID UNIT IS 17.6 CPI, 12.5 LPI, ORIGIN 0.57 INCH, 0.39 INCH;
FMT11A	PORTRAIT; GRID UNIT IS 20.0 CPI, 12.5 LPI, ORIGIN 0.57 INCH, 0.39 INCH;

C.

Approved abbreviations for reserved words

Abbreviations can be helpful in reducing the number of characters or lines you need to describe a function or command. However, while abbreviations are helpful, they can create problems if you do not use them properly. If you use abbreviated reserved words, make sure they are identical to those in the approved list.

The following list contains approved abbreviations for reserved words. The approved abbreviation is shown in capital letters or in parentheses beside the word. If an abbreviation is not shown, you cannot abbreviate the reserved word.

ALigned	GRAPhic	page
and	grid	PALETTE
A3	HAIRline	PAPER
A4	HEAVy	per
BEGin	HIGH	POInts
BLAck	HORizontal	PORtrait
BOTtom	ICAtalog	pts
BOXes	in	REPeat
BROken	INch	RESolution
by	INKs	RIGHt
B4	INTERword	SCALE
CENter	IREsult	SECTIon
CENTImeters (cm)	is	SHADing
color	JUSTified	SHIfT
COLUMn	LANDscape	SIZE
COMment	LEAders	SOLid
characters per inch (CPI)	LEFT	SPACed
DEFault	let	spots per inch (SPI)
do	LIGHt	TEXT
dot	LINes	TIMes
dots	logo	to
DOTTed	lines per inch (LPI)	top
DRAW	LITeral	UNIT
end	MEDium	USIng
ENDSection	MEMber	USLEGal
EVERy	new	USLETter
FONTs	NEXT	VERTical
FORM	NOSubstitution	WIDE
FROM	ORIGIN	XDOTs

D.

Comparison of HF DL and FDL

This appendix compares commands available for HF DL 2.1D, HF DL 3.1, HF DL 3.2, and FDL, as shown in Table D-1.

Table D-1. Command comparison matrix

Command/Feature	HF DL 2.1D	HF DL 3.1	HF DL 3.2	FDL	Extension *
BATCH			•		
BOX	•	•	•	•	
COLUMN	•	•	•		•
COMMENT	•	•	•	•	
DO SECTION	•	•	•	•	
DOT LEADER	•	•	•		•
END	•	•	•	•	
FONT	•	•	•	•	
FORM	•	•	•	•	
GRAPHIC	•	•	•	•	
GRID	•	•	•	•	
ICATALOG		•	•	•	
INK		•	•	•	
INTERWORD	•	•	•		•
INVERSE LANDSCAPE FONTS	•	•	•		
IRESULT		•	•	•	
LET	•	•	•		•
LINE	•	•	•	•	
LOGO	•	•	•	•	
LOWER CASE	•	•	•		
LANDSCAPE/ PORTRAIT	•	•	•	•	
PALETTE		•	•	•	
PAPER		•	•	•	
SECTION	•	•	•	•	
TEXT AT	•	•	•	•	
TEXT IN BOX	•	•	•	•	

*Extension denotes an additional command in HF DL2.1 that is not in FDL.

E.

Supported paper sizes

You can design forms for a variety of paper sizes, however, you must be aware of the paper sizes your printer supports. Even though HFDL supports a paper size, it may not be available or supported by your printer. If you are not sure of the size of paper your printer supports, contact your System Administrator.

Table E-1 lists the paper sizes that HFDL supports.

Table E-1. **Paper sizes available for HFDL**

Paper size (inches)	HFDL designation
8.5 by 11	USLETTER
8.5 by 14	USLEGAL
16.54 by 11.69	A3
8.27 by 11.69	A4
10.12 by 14.33	B4
8 by 10.2	(None)
8 by 10.5	(None)
8 by 13	(None)
8.27 by 10.63	(None)
8.27 by 13	(None)
8.37 by 10.78	(None)
8.46 by 10.83	(None)
8.46 by 12.4	(None)
8.46 by 14.02	(None)
8.5 by 10.75	(None)
8.5 by 13	(None)
11 by 17	(None)

871 CM	Xerox 871 Communications Module. Provides a low-cost, remote communication interface to Xerox printers.
abort	To prematurely terminate the printing of a job, execution of a program, or execution of a nonprinting task.
algorithm	A step by step procedure for solving a problem.
alignment	Positioning of an image on a page for printing.
alphanumeric	Set of letters and whole numbers.
ASCII	American Standard Code for Information Interchange. Standard 7-bit code that represents alphanumeric information. Each alphanumeric character and several nonprinting characters are assigned a binary number covering 128 possible characters. It is used for information interchange among data processing systems, data communication systems, and associated equipment. See also <i>EBCDIC</i> .
application	Use to which a computer program or system is applied (for example, sorting employee records).
application software	Host- or LPS-resident software that directs the computer to perform specific tasks or functions. Common business applications include payroll, accounting, inventory, and so on.
argument	Independent variable of a function, such as a parameter.
ascender	Portion of the lowercase letter that extends above the main body of the character (such as b, d, or h).
background job	Low-priority job, usually batched, which is executed automatically, in the background, as system resources become available.
band	Unit in a printed page that optimally handles image processing in sequence.
batch	Body of data accumulated and grouped by kind before processing.
batch processing	Process that allows repetitive operations to be performed sequentially on batched data without much involvement from the computer operator.
binary	Numbering system with a base of 2. All numbers are represented by combinations of 1 and 0.

bit	<p>Binary digit. Smallest unit of information recognized by a computer. Numeric digits of 1 or 0 represent the binary scale of notation. In computers and related devices, the binary numeric values can be represented by a positive or negative electrical pulse. As a general rule, 1 equals ON, TRUE, or YES; and 0 (zero) equals OFF, FALSE, or NO. A number of bits strung together can represent a character to the computer.</p> <p>Multipliers of the bit are as follows:</p> <ul style="list-style-type: none">• 1 or 0 byte = 8 bits• 1 kilobyte (KB) or 1,024 bytes = 8,192 bits• 1 megabyte (MB) or 1,048,576 bytes = 8,388,608 bits. <p>Computer space equivalents (in KB) are as follows:</p> <ul style="list-style-type: none">• 1.5 KB = about one single-spaced typed page• 30 KB = about 20 typed pages• 150 KB = about 100 typed pages.
bitmap	<p>Visual representation of graphic images in which a bit defines a picture element (pixel). For example, if a bit is 1, the corresponding pixel is printed.</p>
bitmapped	<p>Display image generated bit by bit for each point or dot. A software-driven scanner is used to create characters and graphics.</p>
blocking	<p>Process of combining two or more records into a single block of data, which can be moved, operated upon, stored, and so on, as a single unit by the computer.</p>
block length	<p>Number of characters or bytes contained in a block of data. The block is treated as a unit in the computer. Block lengths may be specified in units such as records, words, computer words, or characters.</p>
boot	<p>To load initial operating instructions for a program into the memory of the computer. These instructions direct the loading of the rest of the program. Booting may require the entry of a few commands or keystrokes (soft boot) or the turning on of a switch to begin the process (hard boot).</p>
buffer	<p>Compartment of memory in which data is stored during transfer from one device to another.</p>
byte	<p>Fixed number of bits operated upon as a unit (in data processing, usually 8). A byte may represent a character, a machine instruction, or some other logical unit of information.</p>
carriage return	<p>Control character that causes the printing system to start printing at the left margin of the current line (unless set to be interrupted as a line end).</p>
centralized printer	<p>Class of Xerox printers that includes the 4050, 4090, 4135, 4650, 4850, 4890, 8790, and 9790 printers. Centralized printers are frequently placed in a central location to handle a large volume of output.</p>
channel	<p>1. In data communications, path or line that allows two or more devices to communicate. 2. In computers, a path for communication between the central processing unit (CPU) and peripheral devices.</p>
character	<p>Single printable letter (A to Z), numeral (0 to 9), symbol (& % #), or punctuation marks (, . ! ?) used to represent data. Characters can also be nonprinting, such as a space, tab, or carriage return.</p>

character cell	Digitized space containing a single character within a font set.
character code	Code representing alphanumeric information, such as ASCII.
character map	Grid of blocks, where each block corresponds to a letter, number, punctuation mark, or special character.
character set	Number of different characters used by a particular device, including alphabetic, numeric, and special characters such as symbols.
CLPS	Centralized laser printing system. Technology that uses a laser to transfer character forms to a page. See also <i>LPS</i> .
cpi	Characters per inch. Designates the number of characters per inch for a particular typeface. See also <i>pitch</i> .
cluster	Group of related feeder trays, usually containing the same size and type of paper stock. Each cluster has a name, consisting of one to six alphanumeric characters.
code	1. Set of symbols representing data or instruction to a computer. 2. To write a list of software instructions that cause the product or system to perform specified instructions.
code conversion	Translation of one type of character or symbol code to another.
collate	To arrange or assemble into ordered sets.
column	Vertical arrangement of characters.
command	User instruction to a computer. Commands are words, mnemonics, or characters that cause a computer to perform predefined operations.
command language	Set of commands that can be used to instruct a system on how to perform a task.
comment	Explanation added to program instructions. They are ignored by the computer.
compatibility	Characteristics of computer equipment permitting one device to use the same information or programs as another device without conversion or code modification.
compiler	Software that translates instructions written in high-level language into machine language for execution by a system.
composite character	Character that has multiple bitmap references combined into one bitmap and stored on disk in font memory.
computer language	Computer-oriented language consisting solely of computer instructions. See also <i>machine language</i> .
computer program	Sequence of instructions suitable for processing by a computer.
concatenate	To connect or link in a series, such as files grouped together for faster processing.
condition code	Code returned by a job that is used to verify successful execution of the job.
configuration	One or more computers and related devices (terminals, printers, and so forth) interconnected and programmed to operate as a system.
console	Functional unit containing devices used by an operator to communicate with an operating system. It may consist of a display, keyboard, and certain switches or other controls.
consumable supplies	Supplies such as paper and dry ink that are depleted during the course of normal operation.

control program	Program that supports the operating system by monitoring the flow of jobs, tasks, processing, and so forth, within the system.
coordinate	Point on the x-axis and y-axis that determines a grid position.
copy	To duplicate data in a new location or on an additional storage medium. For example, to copy files from disk to tape.
copy-sensitive	Term used to indicate jobs in which multiple copies of a report will contain different data, such as paychecks and banking statements.
data	1. In general, facts, numbers, letters, symbols, and so forth, that can be processed or produced by a computer. 2. In data processing, the source data, or raw data, entered for processing, as opposed to the results obtained by processing.
database	Information to meet specific processing and retrieval needs. Generally applies to integrated files of data, arranged for access by many subsystems.
data file	Collection of related data records organized in a specific manner so that each record is similarly structured, for example, a payroll file set up with one record for each employee, structured with the last name first, followed by the rate of pay and all deductions.
data processing	Operations carried out on data by means of algorithms and programs to produce information or a specific result. The rearrangement of data into a suitable form for further use.
data set	Collection of logically related records stored in a database. See also <i>file</i> .
data storage	Preservation of data on various media, such as tape and disks.
debug	To detect and correct errors in a program.
decentralized printer	Class of Xerox printers that includes the 3700, 4045, 4235, and 4700 printers. Decentralized printers are frequently placed in a remote location to handle a lower volume of output.
default	Value assigned to a field by the system. Default fields may be used for such items as document formats, menu selections, input fields, font selection, and paper or image size. You can change the default value of a field.
device	Any piece of hardware other than the central processing unit (CPU).
digitize	To express or represent data in digital (binary) format so it can be processed electronically.
display	Viewing device (monitor) used to visually display the results of operator interaction with the system.
DJDE	Dynamic job descriptor entry. Command within an input data stream used to modify the printing environment dynamically.
dot	Unit of measurement representing a fraction of an inch; for example, 300 dots per inch (dpi). Dot may also refer to a picture element (pixel) or spot.
dpi	Dots per inch. Indicates the number of dots per inch displayed on a terminal screen or printed to form a character or graphic.
download	To load files from the host onto LPS system disks.
dry ink	Minute, dry particles of resin and carbon black used to create images. Dry ink can accept an electrical charge.

duplex	Printing or copying on both sides of the paper. See also <i>simplex</i> .
easy values	Font point sizes, orientations, and names explicitly defined in Font Interchange Standard (FIS).
elite	Smallest size standard typewriter type: 12 characters per horizontal inch. See also <i>pica</i> .
embedded blanks	Blank spaces within a command line.
enabler	Hardware devices or software packages that allow the printer to perform as specified or better.
escapement	Lateral positioning of characters or font families.
EBCDIC	Extended binary coded decimal interchange code. Coded character set consisting of 8-bit coded characters. It can accommodate 256 characters. See also <i>ASCII</i> .
FCB	Forms Control Buffer. Buffer for controlling the vertical format of printed output.
field	1. Part of a record that serves a similar function in all records of the group (for example, name and address field). 2. Area or setting of practical activity or application.
file	Set of records or text that can be stored and retrieved. An organized, named collection of records treated as a unit. For offline, it is the data between the two tape marks. For online, it is the data between banner pages.
fixed font	Font containing characters with fixed spacing.
fixed pitch	Font set in which every character cell has the same width. In reference to character sets, this term describes typefaces in which all character cells are of equal width. Characters are monospaced as opposed to proportionally spaced.
fixed spacing	Arrangement of characters on a line so that all characters occupy the same amount of horizontal space.
flag	Small indicator marking the occurrence of an event or the existence of a certain condition while the program is executing.
floating accent	Nonspacing accent characters that can be combined with characters and printed as a composite.
font	<p>Data file for a particular electronic printer that prints a character set of a specific typestyle (letter, form, and shape). Each font has its own unique characteristics, including the following:</p> <ul style="list-style-type: none"> • Size—expressed in points • Weight—expressed as light, medium, or bold • Stress—Roman or italic • Width—regular, condensed, or expanded • Orientation—the print direction of the characters, relative to the page. <p>In addition to alphanumeric characters, a font can contain various symbols or logos.</p>
font white space	Spacing character in the font.

form	Electronic version of a form that can include lines, boxes, text, shading, logos, and embedded text. Forms can be printed without variable data or merged with variable data during the printing process.
form feed	Keyboard printer control character that causes the printer to skip to the top of the next page.
format	1. Layout of a document, including margins, page length, line spacing, typeface, and so forth. 2. In data storage, the way the surface of a disk is organized to store data. 3. To prepare the surface of a disk for the acceptance of data.
FDL	Forms description language. Keyword oriented language used to define the appearance of an electronic form including horizontal and vertical lines, boxes, descriptive text, and so forth. The source language is referred to as forms source language (FSL) and the object language (after compilation) is referred to as .FRM. See also <i>FSL</i> .
FSL	Forms source language. Uncompiled collection of user-created files containing FDL commands. See also <i>FDL</i> .
graphics	Use of lines and figures to display data, as opposed to using text.
grid	Imaginary pattern of evenly spaced horizontal and vertical lines on a page.
grid unit	Smallest rectangle enclosed by horizontal and vertical lines on a grid. The size of a grid unit is expressed as the length of one side of the rectangle.
hardcopy	Machine output in permanent form, such as printed reports, listings, and so forth.
hard values	Nonoptimal adjustment of particular Font Interchange Standard (FIS) fonts in terms of point size and orientation.
hardware	Physical components of a system (processors, printers, terminals, and keyboards).
hexadecimal	Numbering system with a base of 16. The numbers 10 through 15 are represented by A through F, respectively.
HFDL	Host Forms Description Language. Xerox host implementation of forms description language.
hierarchy	Relative priority assigned to rank and order of items.
high-level language	Computer programming language consisting of symbols and English-like command statements that are easily understood and read by the user. High-level languages are considered the source code and are usually compiled. FDL and HFDL are high-level languages.
host	Computer accessed by users that serves as a source of high speed data processing for workstations with less computer power. Commonly referred to as a mainframe.
host interface	Connection between the network and the host computer.
IDEN	Parameter that notifies the system that DJDE records may be part of the input data stream.
identifier (ID)	Characters used to identify or name data and possibly to indicate certain properties of that data.
image area	Area on a physical page that may contain text or graphics.
input	Data or text introduced into a computer-based system.

input devices	Keyboards, magnetic media, or any device used to give a system information.
input output (I/O)	General term encompassing the flow of data into and out of a computer system.
input processing	Formatting control for the pages of a report.
insert	To add text or graphics to a document.
interface	Device by which two systems connect and communicate with each other.
Interpress	Xerox standard that defines digital representation of lines for printing. You can print Interpress documents on any sufficiently powerful printer equipped with Interpress print software.
JCL	Job control language. High-level language used with the MVS or DOS operating systems to identify job requirements such as user name, program name, CPU processing time, input/output devices, and files needed for batch processing.
JDE	Job descriptor entry. Collection of job descriptions. See also <i>job</i> .
JDL	Job descriptor library. Collection of compiled job descriptions. See also <i>job source library (JSL)</i> .
job	Set of instructions (JDEs) defining a unit of work for the system.
jobcard	First JCL statement in a job, identifying the beginning of the job, the user, the job name, and similar information.
job control	Program called into storage to prepare each job or job step to be run.
job management	Collective functions of job scheduling and command processing.
JSL	Job source library. Collection of uncompiled job descriptions. See also <i>JDE</i> ; <i>JDL</i> .
keyword	Reserved word that is essential to the meaning and structure of a statement. Required part of a command that must be entered in the command language syntax exactly as it is written. In XPRM, keywords can be written in uppercase or lowercase letters.
label	1. In data storage, a reference to a file saved on tape or disk, a record indicating the file name or date created, as well as other control information. 2. In programming, a name assigned to a particular instruction or portion of a program as a location reference. The computer translates the label into a locatable address.
landscape	Orientation of print lines or top of an illustration parallel to the long edge of the paper. See also <i>orientation</i> .
language	Defined set of characters and symbols combined together by specific rules. Used as an operator interface with the computers system software to perform certain tasks. See also <i>high-level language</i> ; <i>low-level language</i> .
laser printing	Technology that uses a laser to scan and transfer character images onto a page by direct or indirect means.
leading	1. Vertical distance between lines (also called line space) measured from the baseline of one line to the baseline of the next. 2. Extra spacing between lines of type. 3. In typography, spacing between lines and paragraphs.
legal size	Page the standard size of legal briefs, 8.5 by 14 inches.

library	In data storage, a collection of related files or programs.
line	One horizontal flow of characters.
line feed	Control character that causes the printing system to begin printing in the current character position of the next line (unless set to be interpreted as an end of line).
line printer	High-speed printer that prints an entire line of characters at the same time.
line tables	Internal data structures that provide a record in memory of the line to be drawn on a page.
listing	Printout or display of the statements in a program, usually used as a convenience in examining or editing programs.
literal	Any alphanumeric character or symbol used to define itself rather than something it might represent.
load	To enter data into storage or working registers.
logo	Small illustration, signature, or design, typically used to identify a company.
log off	Procedure used to end a computer session.
log on	Procedure used to begin a computer session.
low-level language	Computer language that is close to machine language so each written instruction in a program can be converted to machine language on a one-to-one basis.
LPS	Laser printing system. Technology that uses a laser to transfer character forms to a page.
machine language	Binary code that can be read directly by the central processing unit (CPU) of a computer. All programming language is eventually converted into machine language for execution by the computer. See also <i>computer language</i> .
mainframe	The central processing unit (CPU) of a large computer. Most often used to denote any large computer that is used as the central source of computing for remote dumb terminals or interconnected micro- or mini-computers.
margins	White space on the top, bottom, and each side of the printed text.
media	Vehicles or devices on which information is stored or transmitted.
MB (megabyte)	1,048,576 bytes.
menu	List of available functions, commands, or options.
merge	Operation combining two or more items into a predetermined order. For example, merging two separate alphabetically ordered lists into one large alphabetically ordered list.
mnemonic	Word that possesses qualities helpful to human memory. Software commands are often rooted in mnemonic terms or abbreviations to make it easy to associate the functions with the name or keys pressed to implement the function.
mode	Manner in which an activity or process is carried out.
module	Cohesive unit within a program.
monitor	1. Program used to supervise, control, or verify the operations of a system. 2. A monochrome or color cathode ray tube (CRT) used to display computer information to the operator.

MVS	Multiple Virtual Storage. Operating system that supports the IBM System 370 hardware. MVS is designed to handle large scale production batch applications.
national use characters	Includes the @, \$, and # sign. The system recognizes the hexadecimal representations of the U.S. national characters: @ as X'7C'; \$ as X'5B'; and # as X'7B'. In some countries, the keyboard-generated characters may generate different hexadecimal representation and cause errors.
nonimpact printer	Printer that forms characters or symbols on a medium (usually paper) without striking or impacting the medium. Laser printers are nonimpact printers.
object file	Source file converted into machine language (binary code) that is usable by the CPU.
operand	Data that is acted upon (for example, in a computer operation or process).
operating system	Basic host- or LPS-resident controlling program that governs the operations of a computer, such as job entry, input output (I/O), and data management.
orientation	Direction in which text or images are positioned on a page. When text and images are positioned with the lines of text parallel to the shorter sides of the paper, the orientation is portrait. When text and images are positioned with the lines of text parallel to the longer sides of the paper, the orientation is landscape.
origin	References a point on the image area of a page that is usually in the upper left corner of the page.
OSS	Operating system software. See also <i>operating system</i> .
overprinting	Printing more than one character at the same position.
overstrike	Printing characters over each other.
page	1. In computer programming, a block of instructions, data, or both, that can be located in main or auxiliary storage. 2. In word processing or form generation, a defined section of a document.
page end	Command character (form feed) that terminates the current page.
pagination	Process of separating text or data into pages.
parameter	Part of a command, other than the keyword. See also <i>keyword</i> .
PARM	Keyword parameter in the JCL that is coded on the EXEC statement of the JCL after the program or procedure name and only applies to the step being executed.
parsing	To read or interpret a command. To build up a parameter list from information within a command.
physical page	Area on which printing is done.
pica	1. Unit of measurement equal to 0.166 inch. Picas are often used to express line measurements or column width. 2. A type size that measures ten characters to the inch. See also <i>elite</i> ; <i>pitch</i> ; <i>point</i> .
pitch	Horizontal character spacing; 10-pitch spacing is called pica (10 characters per inch); 12-pitch spacing is called elite (12 characters per inch). See also <i>cpi</i> .

pixel	Acronym for picture element. Smallest addressable point of a bitmapped screen that can be independently assigned color and intensity. Pixels are definable locations on a display used to form images. For graphics displays, more pixels generally provide higher resolution. Spots, dots, and pixels are used interchangeably.
point	Unit of measurement equal to 0.0139 inch. Points are always used to express type, size, and leading. There are 12 points to a pica and about 72 points to an inch. See also <i>pica</i> .
point size	Height of character set from the top of its ascender to the bottom of its descenders in units (points). Point size does not always include leading.
portrait	Orientation of print lines or the top of an illustration parallel to the short edge of the paper. See also <i>orientation</i> .
PDL	Print description language. Language used to describe printing jobs to an LPS. PDL describes the input (type, format, and characteristics), performs the processing functions (logical processing), and describes the output (type format, font selection, and accounting options).
printer	Output device that produces hardcopy printouts.
printout	Informal expression referring to almost anything printed by a computer peripheral device.
process	1. To perform a systematic sequence of operations, such as add, edit, or delete. 2. To produce a specific result by manipulating data.
processor	Applies to any system that is capable of receiving and performing operations upon data and supplying the results to those operations.
program	Complete set of instructions in a language compatible with the device to be used. A program directs a system to perform each operation at the right time in the proper sequence.
programmer	Person involved in designing, writing, and testing computer programs.
prompt	Message or symbol displayed on a system console requiring the operator to take action.
proportional font	Font containing characters that vary in width.
proportional spacing	Text in which each alphanumeric character is given a weighted amount of space is proportionally spaced. Such output has print-like appearance. Proportional spacing allows more space for wide characters and less space for narrow characters.
proportional type	Characters that vary in width.
purge	To delete data from a system.
query	Request for data or other information entered by an operator while the system is processing.
queue	List of documents waiting to be processed.
real time	To process transactions as they are entered into a system. User and system interchange allow transactions to be processed and the results returned to the user.
record	Collection of data or words treated as a unit.

recovery	Act of overcoming a problem or error during processing. Typically, a specialized software recovery routine gains control and attempts to resolve the error without crashing the system.
remote access	Access to a central computer by terminals or devices geographically separated from the computer.
replace	Process of exchanging one portion of text or symbols with another. This process encompasses two functions: deleting the old and inserting the new.
resolution	Number of dots per inch (dpi). The LPS imaging system converts a character from digitized data into a printed image composed of these tiny dots. The greater the number of dots per inch, the higher the resolution and the clearer the image. The terms dots, spots, and pixels are synonymous.
resource	File that is stored in a library (in machine-readable format) and used during the printing of a document. Each file holds a different type of resource; for example, forms or fonts.
RSTACK	Parameter that defines a series of reports in a single file.
save	To store programs, data, or text for retrieval at a later time.
scale	To adjust font size according to given proportions.
simplex	Printing on one side of a page. See also <i>duplex</i> .
software	Programs, including operating systems, procedures, utilities, and application programs written for a computer system. Software can be supplied by the hardware manufacturer or other firms but does not include programs written by the user.
sort	To rearrange data records according to a particular order. This order is generally selected by character (alphabetic, numeric, or special symbol) and can be ascending or descending.
source file	File containing source language statements or commands.
source language	High-level or low-level language that is used or created by a programmer. A source language must be converted by a compiler into machine language (object language) for instruction to be executed.
source program	Program written in a source language, generally a high-level programming language.
space	Blank area between words, recognized as a character by word and data processing systems.
spooling	Process of releasing data from main memory and storing it temporarily until it is ready to be accepted by the peripheral device. For example, storing text before sending it to a printer.
spot	Unit of measurement representing a fraction of an inch, for example, 300 spots per inch (spi). Spots may also be referred to as a picture element (pixel) or dot.
statement	Detailed written instructions in a program step. A statement is written according to specific rules called syntax.
string	Connected sequence of alphanumeric characters treated as one unit of data by a program.
submit	Term used when sending a JCL to the computer.
symbol	Characters used in a computer language to specify a particular function.

syntax	Rules governing the structure of expression in a programming language.
syntax error	System response to a mistake in structure of expression.
system	1. In data processing, a collection of parts and procedures organized to accomplish a set of specific functions. 2. Assembly of components united by some form of regulated interaction to form an organized whole. 3. Operation or procedures through which a business activity is accomplished.
system disk	Magnetic disk storage medium, usually of large capacity, that is not removable, as opposed to removable floppy disks or disk packs.
sysgen (system generation)	Process whereby the system is made ready to operate. Typically involves selecting the operative parameters and activating the relevant software.
system page	Maximum area in which text and graphics can be imaged on a printing system.
system software	Software programs that support and control system functions by governing hardware operation and input and output process. This includes interpreting source programs and converting them into machine language, as well as distributing tasks among various processors.
tape	Recording media for data or computer programs. Tape can be in a permanent form, such as perforated paper tape, or a reusable magnetic form. The magnetic form has a higher storage capacity than disk storage, but it takes longer to write or recover data from tape than from disk.
task	1. Any major job performed by a computer. 2. One of several programs being executed by a system.
terminal	Device equipped with a keyboard and connected to a computer or network.
testing	1. Process of running a program for the express purpose of discovering errors. 2. For computer-oriented systems, the process of verifying the ability of a system to meet performance objectives in a simulated environment, or validating its actual performance in a real-time environment.
text	Portion of a message, program, or input that contains the information you want to convey.
text string	Consecutive series of characters that are output exactly as specified in the command or program.
translation	Function of a language processor that converts certain source programs, such as Basic, into a machine-readable language.
transmit	To send data from one place to another.
truncate	Cut off before completion, as when data transferred from a host to a printer is cut off before all data is transmitted.
TSO	Time Sharing Option. Host software that allows multiple terminals to communicate efficiently with a host. TSO command processors can edit data sets, submit jobs for batch execution, and perform data set manipulation.
typeface	1. Type of a single design. 2. Set of character with design features that make them similar to one another.

type size	Height of a typeface expressed in points (measured from the bottom of its descenders to the top of its ascenders).
type style	Italic, condensed, bold, and other variations of typeface that form a type family.
UCS	Universal Character Set. Printer feature that permits the use of a variety of character arrays.
variable	Information of a changeable nature that is merged with a standard or a repetitive document to create a specialized or personalized version of that document.
variable data	Data that varies from page to page. Variable data is not a part of form design.
variable text	Text that varies from page to page. An example would be various names and addresses combined with a form letter to make a complete document.
virtual page	Page area selected by a forms designer for printing.
x-axis	Horizontal axis on a form grid.
xdot	Unit of measurement representing a fraction of an inch. Xdots may also be referred to as a picture element (pixel) or spot.
XDPM	Xerox Decentralized Print Mode or Xerox Escape Sequence (XES). See also <i>XES</i> .
XES	Xerox Escape Sequence. See also <i>XDPM</i> .
XPPM	Xerox Production Print Mode.
XPRM	Host-resident utility providing an interface between printer resources and the printer. XPRM allows you to store and manage resources generated by multiple applications in a central repository.
y-axis	Vertical axis on a form grid.

Numerals

4135 printer, SEF, 2-2
4700 Color Document Printer, 2-3, 2-11
871 Communications Module (CM), 1-2

A

abbreviating, reserved words , 3-2, C-1
accessing compiled forms, 5-12
alignment, 3-19
ASCII format, 2-11

B

BARR SNA, 1-2
BATCH command, 5-5 to 5-6
batch processing, COMPILE command, 5-5 to 5-6
bitmaps, 2-11
BOX command, 3-14 to 3-16
boxes
 images, A-2
 text, see *TEXT IN BOX command*
BSC/RJE communications link, 1-3

C

catalog, ink, 2-11 to 2-12
centralized
 font card, 3-8
 printers, 1-2 to 1-3, 5-6 to 5-7
character
 cells, 2-7
 identifier, 2-10 to 2-11
 maps, 2-9 to 2-11
 per inch (CPI), 3-7
 sets, 2-7
 spacing, 2-13
 special, 3-2 to 3-3
 text block, 3-13
code point, 2-11
coding conventions, COMPILE command, 5-1
color, adding, 2-11
COLUMN command, 3-30
commands, HFDL
 comparison, FDL, D-1
 description, 3-12 to 3-35
 format, 3-2 to 3-3
 forms, creating, 3-1
 setup, 3-3 to 3-12
 syntax, 3-3
COMMENT command, 3-34 to 3-35
communications link, 1-3
COMPILE command, 2-9, 2-11, 3-3, 5-1 to 5-10
compiled forms, accessing, 5-12

coordinates

 description commands, 3-13
 elements, 2-4 to 2-5
 negative, 2-6
 relative, 3-33 to 3-34
 X and Y, 2-5 to 2-6

D

data sets, 5-6 to 5-9
data types, 2-6
database, 3-9, 5-6
DDnames, 5-6 to 5-9
decentralized
 font card, 3-8
 printers, 1-3, 5-7 to 5-8
density limits, lines, A-1
description commands, 3-12 to 3-35
design ruler, see *forms, design ruler*
DO SECTION command, 3-28 to 3-29
documentation, other, viii to ix
DOT LEADER command, 3-32 to 3-33
DOTS, 3-8
downloading, sample, 5-10 to 5-11

E

edge-marking, 2-3
elements, form, 2-4 to 2-5
END command, 3-35
environment, operating, 1-2 to 1-3
errors
 local density, A-1 to A-2
 messages, 6-1 to 6-27
 page setup, A-1 to A-2

F

FDL commands, 3-1, D-1
features, HFDL
 commands, 3-1
 software, 1-1
fixed fonts, 2-7
FMT grid format properties, B-1 to B-2
FONT command, 3-8 to 3-9
fonts
 character maps, 2-9 to 2-11
 character sets, 2-7
 index number, 3-17
 names, 3-8
 orientation, 2-7 to 2-8
 types, 2-7
FONTS command, SEF, 2-2

form

- elements, positioning, 2-4 to 2-5
- orientation, 3-5
- origin, 2-3

formats

- ASCII, 2-11
- commands, 3-2 to 3-3
- determining, 2-1 to 2-6
- inkname, 3-11
- XES, 3-9

forms

- color, 2-11 to 2-12
- compiled, accessing, 5-12
- data, 2-6
- design ruler, 2-12 to 2-13
- existing, editing, 5-12
- HFDL application, 1-3 to 1-4
- monochrome, specifying, 5-12
- planning, 2-1 to 2-13
- samples, 4-1 to 5-12
- setting up, 3-3 to 3-12, 4-2 to 4-3
- types, 1-4

FORMS command, 3-4

Forms Description Language, see *FDL*

Forms Source Language, see *FSL*

frames, page, 2-2 to 2-3

FSL, 3-1, 5-12

G

GRAPHIC command, 3-26 to 3-27

graphic representation, see *bitmaps*

GRID command, 3-7 to 3-8, B-1 to B-2

grids, 2-5 to 2-6

H

hardware requirements, 1-2

HFDL commands, using, 3-1 to 3-35

highlight color printer, 2-11 to 2-12

horizontal format, see *landscape orientation*

I

IBM

- S/370, 1-2

- S/390, 1-2

ICATALOG command, 3-9

indicators, messages, 6-1

ink

- catalog, 2-11, 3-9
- color, 2-11 to 2-12
- palette, see *PALETTE command*

INK command, 3-10 to 3-11

inkname format, 3-11

INTERWORD command, 3-30 to 3-32

interword gap

- formula, 3-30
- multiplier, 3-31
- SPACING operand, 3-31 to 3-32
- specifying base, 3-31

IRESULT command, 3-12

ISPF editor, 5-12

J

JCL

- HFDL compiler, 5-6 to 5-10

- requirements, 5-1

job control language, see *JCL*

K

keyboard character maps, 2-9 to 2-10

KEYBOARD parameter, 2-9

keyword commands, see *commands*

L

landscape, orientation, 2-1 to 2-2, B-1 to B-2

LANDSCAPE/PORTRAIT command, 3-5 to 3-6

LET command, 3-33 to 3-34

line

- density, limits, A-1
- origin, location, 3-13 to 3-14
- per inch (LPI), 3-7
- size table, 3-13
- spacing, 2-13
- using, sections, A-2

LINE command, 3-13 to 3-14

LIST COLORMAP command, 3-9

LITERAL operand, 3-3

LOGO command, 3-25 to 3-26

M

mapping process, 2-9

maps

- fonts, 2-9, 2-11

- keyboard, 2-9

message indicators, 6-1

messages, see *errors, messages*

metacode, conversions, 1-2

monochrome mode, , 5-12

multiple font names, 3-8

Multiple Virtual Storage, see *MVS*

MVS/ESA, 1-2

MVS/XA, 1-2

N

negative coordinates, 2-6

O

operating

- environment, 1-2 to 1-3
- system software, see *OSS*

orientation

- fonts, 2-7 to 2-8
- form, 3-5
- page, 2-1 to 2-2
- text, 3-23

origin

- elements, 2-4 to 2-5
- form, 2-3 to 2-4
- line, 3-13
- virtual page, 3-7

OSS, 1-2 to 1-3

P

page

- frames, 2-2 to 2-3
- orientation, 2-1 to 2-2
- setup errors, A-1 to A-2

palette, 2-11 to 2-12

PALETTE command, 3-9 to 3-10

PAPER command, 3-4 to 3-5

paper size, 2-3, 2-12, 3-4 to 3-5, E-1

physical page , 2-2 to 2-3, 3-13

PORTRAIT command, 3-5 to 3-6, B-1 to B-2

portrait orientation, 2-1 to 2-2

prerequisites, 1-2

printer

- console, 3-4
- grippers, 2-3

printers

- 4135 printer, SEF, 2-2
- 4700 Color Document, 2-3, 2-11
- highlight color, 2-11
- supported, 1-2 to 1-3

printing samples, 5-10 to 5-11

procedures

- descriptive data, entering, 4-3 to 4-6
- forms, setting up, 4-2 to 4-3

proportional fonts, 2-7

protocol converter, 1-3

publications, related, viii to ix

R

relative coordinate, 3-33 to 3-34

requirements

- hardware, 1-2
- JCL, 5-1
- software, 1-2

reserved words, 3-2, C-1

resolution, 3-4

ruler scales, 2-12

S

S/370 channel, 1-2

sample

- description command, 4-3 to 4-6
- forms, printing, 5-1 to 5-12

scales, design rule, 2-12

SECTION command, 3-27 to 3-28

SEF option, 2-2

setup commands, 3-3 to 3-12

short edge feed option, see *SEF option*

size

- lines, table, 3-13
- paper, 2-3, 2-12, 3-4 to 3-5, E-1

SNA/RJE communications link, 1-3

software requirements, 1-2

SPACING operand, 3-31

special characters, 3-2 to 3-3

syntax

- BATCH command, 5-5 to 5-6
- command flow, HFDL, 3-3
- COMPILE command, 5-1 to 5-5

system

- considerations, A-1 to A-2
- page, 2-2 to 2-3

T

TEXT AT command, 3-17 to 3-21

text block, size, 3-13

TEXT IN BOX command, 3-21 to 3-25, A-2

toner consumption, 5-12

types

- fonts, 2-7 to 2-8
- forms , 1-4
- page frames, 2-2 to 2-3

U

unit value, 2-6

V

variable data , 2-6

vertical format, see *portrait orientation*

virtual keyboard character maps, 2-9 to 2-11

virtual page

- description commands, 3-13
- origin, 3-7
- size, 2-2 to 2-4

Virtual Storage Access Method, see *VSAM*

VSAM data sets, 5-6

X

X and Y coordinates, 2-5 to 2-6

XDOTS, 3-8

Xerox 4135 printer, SEF, 2-2

Xerox 4700 Color Document Printer, 2-3, 2-11

Xerox Flex Box protocol converter, 1-3

XES

- format, 3-9
- logos, 3-25

YY coordinates, see *X and Y coordinates*

There should only be one blank line above each top entry
in the columns of this page.
(Remove this text and arrows.)

If there is NOT a blank line at the top
of this 2nd column . . . put one in.

This layout guide specifies the appearance of a desired index. There are several fields below, and by editing inside them you determine how the index will look.

(Don't delete the fields or change their properties. Edit only their content.)

The fields will not appear in the resulting index.

This frame will not be present in the index created.

NOTE: The white space frame is purposefully 11pts less high than the usual similar frame in other 8.5×11 templates.

REQUIRES a blank first line in each column of the first page of the finished document for the correct white space.

Examples of the three levels, three different paragraphs. Remember that each level has that number of tabs prior to the first character.

Notes to DELETE when finished.

You can use <SAME> on these next three paragraphs.

Entry - Notice that this line wraps at the same point as other levels.

Subentry - This line also wraps at the same point as other levels.

Tertiary - Notice that this line wraps at the same point as other levels.

A

item, 3, 5 to 7

subentry

Sig & Bld Ex

tertiary entry

00 IT Main Entries

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
boxes	boxes(1)			AN
centralized	centralized(1)			AN
character	character(1)			AN
commands, HFDL	commands, HFDL(1)			AN
coordinates	coordinates(1)			AN
decentralized	decentralized(1)			AN
errors	errors(1)			AN
features, HFDL	features, HFDL(1)			AN
fonts	fonts(1)			AN
form	form(1)			AN
formats	formats(1)			AN
forms	forms(1)			AN
IBM	IBM(1)			AN
ink	ink(1)			AN
interword gap	interword gap(1)			AN
JCL	JCL(1)			AN
line	line(1)			AN
maps	maps(1)			AN

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
operating	operating(1)			AN
orientation	orientation(1)			AN
origin	origin(1)			AN
page	page(1)			AN
printer	printer(1)			AN
printers	printers(1)			AN
procedures	procedures(1)			AN
requirements	requirements(1)			AN
sample	sample(1)			AN
size	size(1)			AN
syntax	syntax(1)			AN
system	system(1)			AN
types	types(1)			AN
virtual page	virtual page(1)			AN
XES	XES(1)			AN

00 IT Main (see) entries

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
design ruler, see <i>forms, design ruler</i>				AN
Forms Description Language, see <i>FDL</i>				AN
Forms Source Language, see <i>FSL</i>				AN
graphic representation, see <i>bitmaps</i>				AN
horizontal format, see <i>landscape orientation</i>				AN
job control language, see <i>JCL</i>				AN
keyword commands, see <i>commands</i>				AN
Multiple Virtual Storage, see <i>MVS</i>				AN
short edge feed option, see <i>SEF option</i>				AN
vertical format, see <i>portrait orientation</i>				AN
Virtual Storage Access Method, see <i>VSAM</i>				AN
Y coordinates, see <i>X and Y coordinates</i>				AN
messages, see <i>errors, messages</i>				AN

00 IT Subentries

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
text, see <i>TEXT IN BOX</i> command		boxes(1)		AN
palette, see <i>PALETTE</i> command		ink(1)		AN
system software, see <i>OSS</i>		operating(1)		AN

000 IT Introduction

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
documentation, other	nal info			OV
	ons rela			
publications, related	elated publi			OV
	M public			

01 IT Overview

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
871 Communications Module (CM)	dule (C			OV
BARR SNA	RR S			OV
BSC/RJE communications link	SC/R			OV
printers	zed pr	centralized(1)		OV
	ized pr			
communications link	ations l			OV
printers	ece	decentralized(1)		OV
environment, operating	ng envir			OV
	ers tha			
software	L feat	features, HFDL(1)		OV
HFDL application	asting H	forms(1)		OV
	r cost			
types	pes	forms(1)		OV
hardware requirements	ardwa			OV
S/370	M S/37	IBM(1)		OV
S/390	90 se	IBM(1)		OV
metacode, conversions	M to meta			OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
MVS/ESA	SA ope			OV
MVS/XA	S/X			OV
environment	onm	operating(1)		OV
	he asso			
OSS	ating system s			OV
	ted OS			
prerequisites	rereq			OV
supported	ter sup	printers(1)		OV
	pported d			
protocol converter	ocol con			OV
hardware	re requ	requirements(1)		OV
software	ftw	requirements(1)		OV
S/370 channel	es on IB			OV
SNA/RJE communications link	JE com			OV
software requirements	are re			OV
forms	ess fo	types(1)		OV
Xerox Flex Box protocol converter	lex Bo			OV

02 IT Planning your forms

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
4135 printer, SEF	5 prin			OV
4700 Color Document Printer	x 47			OV
4700 Color Document Printer	700 p			OV
ASCII format	r is in AS			OV
bitmaps	itma			OV
catalog, ink	nk ca			OV
	ata			
cells	acter cel	character(1)		OV
identifier	cter ide	character(1)		OV
	cter iden			
maps	r map	character(1)		OV
	er map			
	ter map			
sets	arac	character(1)		OV
spacing	ter spac	character(1)		OV
code point	ode poi			OV
color, adding	ddi			OV
COMPILE command	ILE com			OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
COMPILE command	LE com			OV
elements	orm ele	coordinates(1)		OV
	ft corn			
negative	ega	coordinates(1)		OV
X and Y	Y coo	coordinates(1)		OV
	cifying a g			
data types	ta ty			OV
edge-marking	dge-mar			OV
elements, form	ning f			OV
	orm elem			
fixed fonts	ixe			OV
character maps	nt cha	fonts(1)		OV
	ont chara			
	ont char			
character sets	er set	fonts(1)		OV
orientation	nt ori	fonts(1)		OV
	ient			
types	d and p	fonts(1)		OV
FONTS command, SEF	NTS com			OV
elements, positioning	osit	form(1)		OV
	emen			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
origin	m ori	form(1)		OV
ASCII	e ASC	formats(1)		OV
determining	ining forma	formats(1)		OV
	daries asso			
	ion (SY			
	ace the foll			
	ying bot			
	a and var			
color	or to f	forms(1)		OV
	lor models, i			
data	ta: fo	forms(1)		OV
design ruler	ms des	forms(1)		OV
	uler for m			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
planning	ning your f	forms(1)		OV
	opriate for a G			
	rea you sp			
	ise the o			
	gh the f			
	sider them wh			
	riable d			
	ction of the p			
	etting the co			
	mal code poin			
	es the loc			
	gn ruler ca			
	e side 2 fo			
frames, page	ge fra			OV
	ke sure			
grids	Gri			OV
	MT gr			
highlight color printer	ght color p			OV
	ns, and m			
catalog	ontai	ink(1)		OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
color	k color av	ink(1)		OV
	ult in			
keyboard character maps	yboar			OV
	al keybo			
KEYBOARD parameter	OARD par			OV
landscape, orientation	pe ori			OV
	pe forma			
spacing	g or li	line(1)		OV
mapping process	pping proc			OV
fonts	nt character m	maps(1)		OV
fonts	iler equat	maps(1)		OV
keyboard	rd-to-f	maps(1)		OV
negative coordinates	ve co			OV
fonts	eir orien	orientation(1)		OV
	on when u			
page	age o	orientation(1)		OV
	entation, an			
elements	wing ele	origin(1)		OV
	ons (of a			
form	d the form ori	origin(1)		OV
	m origin lo			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
frames	ame i	page(1)		OV
	ude the are			
orientation	ienta	page(1)		OV
	cape orie			
palette	ette, an			OV
	nd sub			
paper size	aper s			OV
paper size	osing pa			OV
physical page	hys			OV
	hysi			
portrait orientation	er portr			OV
	ait fo			
grippers	rippe	printer(1)		OV
4135 printer, SEF	rox 41	printers(1)		OV
4700 Color Document	r Doc	printers(1)		OV
4700 Color Document	x 4700 C	printers(1)		OV
highlight color	hlight c	printers(1)		OV
proportional fonts	ortiona			OV
ruler scales	uler sca			OV
scales, design rule	cales are de			OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
SEF option	d (SEF) m			OV
paper	aper siz	size(1)		OV
paper	r siz	size(1)		OV
page	ystem p	system(1)		OV
	yste			
fonts	ach hav	types(1)		OV
	gn ea			
page frames	t of bou	types(1)		OV
	n never b			
unit value	nit va			OV
variable data	iable da			OV
virtual keyboard character maps	irtual k			OV
	rtual ke			
	yboard and font c			
size	ual pa	virtual page(1)		OV
	rtual pag			
	o the actu			
X and Y coordinates	rdina			OV
	u can no			
Xerox 4135 printer, SEF	arge pa			OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
Xerox 4700 Color Document Printer	ment Printer h			OV
Xerox 4700 Color Document Printer	nter uses full co			OV

03 IT Using FDL commands

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
abbreviating, reserved words	rved w			OV
alignment	nmen			OV
BOX command	X com			OV
	X comm			
	ng fo			
font card	rd and a m	centralized(1)		OV
per inch (CPI)	PI sp	character(1)		OV
special	ecia	character(1)		OV
	pecia			
text block	ters in a te	character(1)		OV
COLUMN command	MN com			OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
description	escr	commands, HFDL(1)		OV
	• EN			
	ts is a short			
	re 3-12			
	w the li			
	fy at leas			
	ure 3-13			
	n center mu			
	xt' spec			
	x with matc			
	re 3-14			
	N COLU			
	... pro			
	ve a 'LO			
	6 imag			
	hin the s			
	nal=xOr			
	is where			
	D JUSTIF			
	ula abo			
	ng' sp			
	es xoxo			
	T Y1+3			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
format	nd for	commands, HFDL(1)		OV
	erstandi			
forms, creating	reati	commands, HFDL(1)		OV
setup	ettin	commands, HFDL(1)		OV
	he fir			
	w) and			
	TS, and XD			
	xes) re			
	ntime			
	3–6 sho			
	irst ink y			
	RED, PI			
	lack pix			
syntax	and syn	commands, HFDL(1)		OV
COMMENT command	MMENT com			OV
	ENT ***H			
COMPILE command	t the COMP			OV
description commands	rdi	coordinates(1)		OV
relative	ativ	coordinates(1)		OV
	+ or – sp			
database	abase b			OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
font card	ard. Us	decentralized(1)		OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
description commands	ipt			OV
	axim			
	nd (C _e) p			
	ure 3–12			
	wo decima			
	nd are refer			
	ig			
	uote i			
	pted by a 't			
	T 1,1 'Jo			
	e 3–14			
	nt of ve			
	NT <i>n</i> k			
	lue fo			
	LE <i>n</i> sp			
	ifies the nam			
	alue * yGr			
	Y C_i ind			
	MN SIZE IS 3			
	ax" secti			
	eed 12 c			
	uces –. –.			
	T Y2–4			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
DO SECTION command	DO SEC			OV
	AT specifies th			
DOT LEADER command	ADER co			OV
	es ***			
DOTS	ress DOT			OV
END command	ND comm			OV
FDL commands	guage (FD			OV
commands	FDL comma	features, HFDL(1)		OV
FONT command	ONT com			OV
	ONT Fon			
index number	nt index nu	fonts(1)		OV
names	ies the stan	fonts(1)		OV
orientation	ntation of the f	form(1)		OV
commands	wing conventi	formats(1)		OV
	nder the c			
inkname	nkna	formats(1)		OV
XES	m in XE	formats(1)		OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
setting up	g up y	forms(1)		OV
	d a prede			
	S USLET			
	xceed th			
	DL descripti			
	mat FMT			
	OG (hig			
	ned is 25			
	ribe fo			
	nly on a X			
FORMS command	RM com			OV
FSL	rce Lang			OV
GRAPHIC command	APHIC co			OV
	C SYM			
GRID command	RID co			OV
	2 CPI BY 8			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
HFDL commands, using	g HFDL com			OV
	tiple com			
	ses (.			
	tion (30			
	d B4 are t			
	AIT, grid u			
	de of a squ			
	em Admi			
	apter in the XP			
	id on Xe			
	e compiling for X			
	jects ima			
	p to 4,09			
	umber of lin			
	ame of th			
	D, BROK			
	ory of the p			
	ho			
	uous st			
	n specif			
	ax' DOT LE			
	mm			
	ex numb			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
ICATALOG command	OG com			OV
catalog	he ink c	ink(1)		OV
INK command	NK comma			OV
	his sta			
inkname format	e forma			OV
INTERWORD command	RWO			OV
	X'20', t			
	erwo			
formula	ord ga	interword gap(1)		OV
multiplier	tiplie	interword gap(1)		OV
SPACING operand	CING op	interword gap(1)		OV
	ACIN			
specifying base	e base ga	interword gap(1)		OV
IRESULT command	SULT com			OV
LANDSCAPE/PORTRAIT command	RAIT com			OV
	APE/PO			
LET command	ET comm			OV
	nate as the v			
origin, location	e orig	line(1)		OV
	rizon			
per inch (LPI)	I specifie	line(1)		OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
size table	ze of the lin	line(1)		OV
LINE command	INE com			OV
	RY 0.50 IN			
LIST COLORMAP command	OLORMAP c			OV
LITERAL operand	TERAL op			OV
LOGO command	GO com			OV
	OGO NEW			
multiple font names	arate multi			OV
form	m, and the	orientation(1)		OV
text	he text orien	orientation(1)		OV
line	cify a lin	origin(1)		OV
virtual page	al page o	origin(1)		OV
PALETTE command	ETTE com			OV
	tte nam			
PAPER command	APER com			OV
	mal pap			
paper size	per siz			OV
	zes you ca			
physical page	ical pag			OV
PORTRAIT command	E/POR			OV
	ORTRAIT sp			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
console	onsole	printer(1)		OV
relative coordinate	e coordi			OV
	es show the lo			
reserved words	eserv			OV
resolution	ON: You			OV
SECTION command	CTION com			OV
	ECTION comman			
setup commands	u enter setu			OV
	m one to si			
	SLETT			
	rid format fo			
	zontal size of			
	d is 1 XDO			
	alog is a file tha			
	ALETTE 'PA			
	mes with embe			
	ixels coin			
lines, table	s table, c	size(1)		OV
paper	ect pape	size(1)		OV
	R, USLE			
SPACING operand	ACING is sp			OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
special characters	pecial charac			OV
	rs print only if y			
command flow, HFDL	x flow ch	syntax(1)		OV
TEXT AT command	XT AT com			OV
	EXT A			
	ustification is be			
	dth of the colu			
	st' NEW L			
text block, size	loc			OV
TEXT IN BOX command	XT IN BOX com			OV
	EXT IN B			
	ne of text to occu			
	xes must have a commo			
	X 4 3 'FIRS			
description commands	ond the vir	virtual page(1)		OV
origin	igin with re	virtual page(1)		OV
XDOTS	nd XDO			OV
format	S format, fon	XES(1)		OV
logos	ort XES lo	XES(1)		OV

04 IT Creating a sample form

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
samples	g a sam	forms(1)		OV
	sing FMT1 v			
	4 paralle			
	5 parall			
	NTAL BOX 'BAC			
	6 BOXES 13			
setting up	up the fo	forms(1)		OV
	S UN214			
descriptive data, entering	nter descriptive da	procedures(1)		OV
	Y AT 4, 12, 3			
	NTAL BOX 'STOC			
	HADING MEDIU			
forms, setting up	et up the basic ele	procedures(1)		OV
	N107A, UN1			
description command	mple description com	sample(1)		OV
	7, 96, 104, 11			
	L BOX 'REOR' 'P			
	ING FONT 3 AT 61, -1			

05 IT Managing and printing sample forms

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
accessing compiled forms	essing compiled f			OV
BATCH command	e the BAT			OV
	TCH com			
batch processing, COMPILE command	ssing COMPILE co			OV
	YZFORMS d			
printers	ntraliz	centralized(1)		OV
	s for cent			
coding conventions, COMPILE command	oding conve			OV
COMPILE command	sing the C			OV
	ap = inte			
	YFORM be			
	Teral=chara			
	hat are MMB			
	YZFM2) LIS			
	ddition to the reso			
	atements re			
	SUACT			
	ows an ex			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
compiled forms, accessing	Acc			OV
data sets	ata sets sto			OV
	nce data se			
	ttioned da			
	uential da			
database	abas			OV
DDnames	fies the DDna			OV
	Dnam			
	cution DD			
	pilation, the DD			
printers	mes for dec	decentralized(1)		OV
	ames for dec			
downloading, sample	wnlo			OV
	ownlo			
compiled, accessing	ccess a compi	forms(1)		OV
existing, editing	diting an ex	forms(1)		OV
monochrome, specifying	fying monoc	forms(1)		OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
samples	escription Lan	forms(1)		OV
	St produ			
	EN spac			
	le 5-2 sh			
	atches the indic			
	ch in your compiler in			
	ble 5-6 id			
	tten to SYSPR			
	ted in Table 5-8 a			
	ant to dow			
	ding a			
	ochrome mo			
FSL	uage (FS			OV
ISPF editor	r ISPF e			OV
HFDL compiler	FDL compi	JCL(1)		OV
	FNTA			
	FINP			
	SUPTA			
	gure 5-9 s			
requirements	CL requireme	JCL(1)		OV
monochrome mode,	onochrome mode			OV

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
printing samples	d printing a sa			OV
	ure 5–10			
JCL	ms in your JCL, as app	requirements(1)		OV
forms, printing	aging and prin	sample(1)		OV
	ERsion=Oth			
	0) the EN sp			
	HIFT=chara			
	XC MEM=M			
	BJECT VER=X			
	alfuncio			
	FFR			
	y the XSUI			
	king the HF			
	inting a sa			
	put Parm (VE			
BATCH command	atch pr	syntax(1)		OV
	ZFORMS da			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
COMPILE command	MPILE com	syntax(1)		OV
	ersion XC on			
	Oup=nam			
	EMber=sele			
	e DDname is XFFSL a			
toner consumption	oner consump			OV
VSAM data sets	Virtual Storage Access Me			OV

06 IT Messages

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
messages	essa	errors(1)		OV
	XFG6006W			
	FG6020I			
	FP1001W			
	FP1005W			
	FP1015W			
	FP1024W			
	FP1031W			
	FP3007E			
	FP3016E			
	FP3030E			
	FP4010F			
	FP4030F			
	d variable, identifier poin			
	imum data lis			
	abase dictionary and dat			
	FP4034F			
	FP4051F			
	FP4066F			
	FP4084F			

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
message indicators	ndicate			OV
indicators, messages	age indi			OV

Appendix A. IT System considerations

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
images	boxes a	boxes(1)		OV
density limits, lines	ne densi			OV
local density	ocal de	errors(1)		OV
	pe of err			
page setup	ge setup er	errors(1)		OV
	nsity of the ar			
density, limits	y limi	line(1)		OV
using, sections	ithin sect	line(1)		OV
setup errors	ounter a pag	page(1)		OV
	es a page setup err			
considerations	em conside	system(1)		OV
	f your text font siz			
TEXT IN BOX command	T IN BOX comma			OV

Appendix B. FMT grid format properties

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
FMT grid format properties	MT grid f			OV
	ble B-3 li			
GRID command	RID command			OV
	rint on A4 p			
landscape, orientation	MT prope			OV
	CAPE; GR			
PORTRAIT command	ORTRAIT and G			OV
	RTRAIT; GRI			

Appendix C. IT Approved abbreviations for reserved words

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
abbreviating, reserved words	oved abb			OV
reserved words	eserved wo			OV

Appendix D. IT Comparison of HFDL and FDL

Hint: To begin filling in a blank Index Table, press <NEXT>.

ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
comparison, FDL	arison of HF	commands, HFDL(1)		OV
FDL commands	nd FD			OV

Appendix E. Supported paper sizes

Hint: To begin filling in a blank Index Table, press <NEXT>.

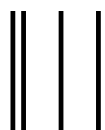
ITEM	VARIATIONS (Optional)	LIST UNDER (Optional)	SORT AS (Optional)	CODES (Optional)
paper size	orted paper si			OV
paper	or a variety of pape	size(1)		OV

Reader Satisfaction Survey

We would appreciate your comments and suggestions for improving this publication.					
Publication title and number			Current date		
Your name and title			Telephone number		
Your company and address					
	Very Satisfied	Somewhat Satisfied	Neither Satisfied nor Dissatisfied	Somewhat Dissatisfied	Very Dissatisfied
1.	How satisfied are you with this manual's ease of use?				
2.	How satisfied are you with the clarity and usefulness of the illustrations and tables?				
3.	How satisfied are you with the effectiveness of the index in helping you find information?				
4.	How satisfied are you with the procedures and/or command descriptions found in this manual? (Are they complete and easy to follow?)				
5.	How satisfied are you with the way the information is distributed among the chapters? (Is the information easy to find? Is too much information repeated?)				
6.	Rate your overall satisfaction.				
Do you have any questions that are not answered by the documentation? If so, please list them.					
What specific things can we do to make the documentation better?					
Please write any other comments or suggestions. Please be specific and give page and paragraph references where applicable.					

Thanks for your feedback. Cut, fold, and fasten as shown OR fax to the number at the top of this page.

Please fold here



Place Stamp
Here
Post Office
Will Not
Deliver
Without Proper
Postage

Xerox Corporation
Printing Systems Software
280 Wekiva Springs Road, Suite #340
Longwood, Florida 32779-3606

Attn: Documentation



Please fold here