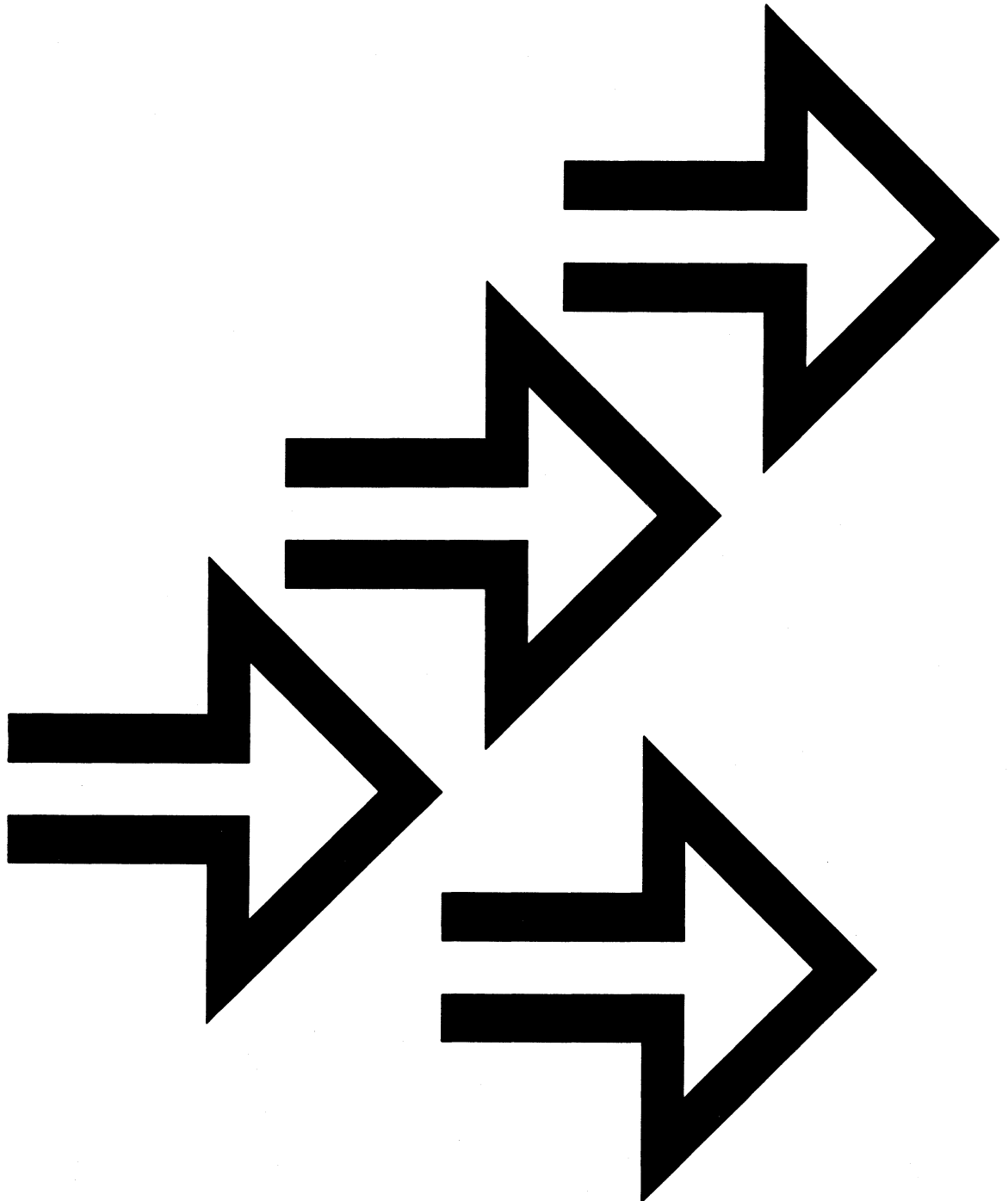




**Document Composition Facility:  
Generalized Markup Language  
Applications Guide**

G544-3305-00

**Release 3.2**



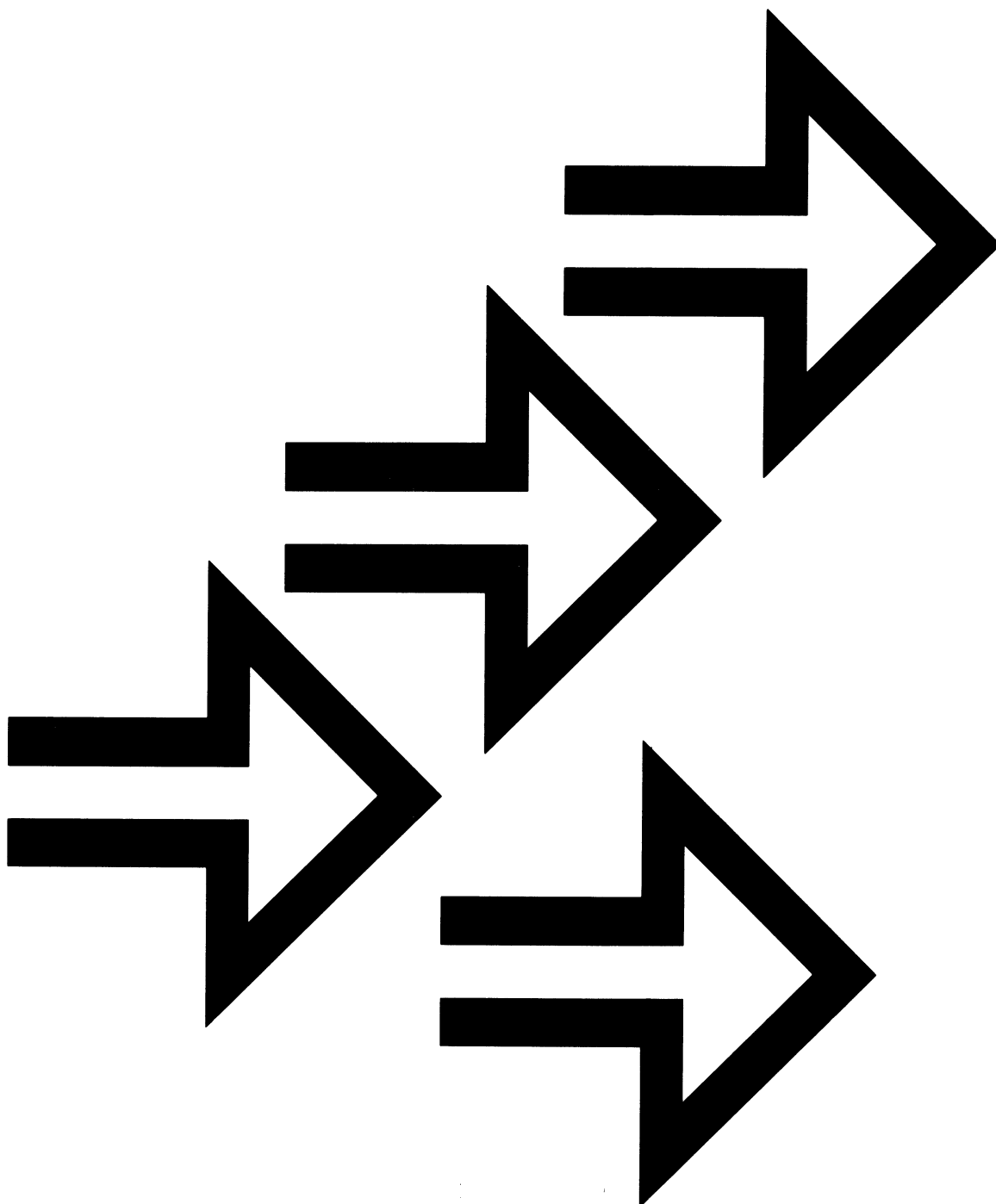




# **Document Composition Facility: Generalized Markup Language Applications Guide**

G544-3305-00

Release 3.2



### **First Edition (August 1989)**

This edition applies to a set of Generalized Markup Language Applications of the IBM Document Composition Facility (DCF), Release 3.2, Licensed Program (Program Number 5748-XX9), which operates with VM/CMS and MVS/TSO systems that use DCF 3.2. This edition applies to any subsequent release of this feature until otherwise indicated in new editions or technical newsletters.

Changes are made periodically to the information herein. Before using this publication in connection with the operation of IBM systems, consult the latest *IBM System/370, 30xx, 4300, and 9370 Processors Bibliography*, GC20-0001, for the editions that are applicable and current.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM licensed program in this publication is not intended to state or imply that only IBM's licensed program may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the address given below; requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for reader's comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Box 1900, Department 588, Boulder, Colorado, U.S.A. 80301-9191. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever. You may, of course, continue to use the information you supply.

# Contents

<b>Chapter 1. Introduction</b>	<b>1</b>
How This Book Is Designed	1
The GML Applications	2
What's a Memo?	2
What's an Overhead Transparency?	2
What's a Schedule?	2
Prerequisites	4
<b>Chapter 2. Creating Memos</b>	<b>5</b>
Memo Tags and Their Attributes	6
:LOGO	6
:TO	6
:FROM	7
:MEMODATE.	7
:SECURITY.	7
:SUBJECT.	8
:REF.	8
:MEMOTXT	8
:SIGN.	9
:CC	9
Examples of Memos	10
Internal Memo	10
External Memo	12
Things to Remember	14
<b>Chapter 3. Creating Overhead Transparencies</b>	<b>15</b>
Overhead Transparency Tags and Their Attributes	16
:TRHEAD	16
:TRFOOT	17
Examples of Overhead Transparencies	18
Title-Page Example	18
Head-Level Examples	20
Text Example	24
Things to Remember	26
<b>Chapter 4. Creating Schedules</b>	<b>27</b>
Schedule Tags and Their Attributes	28
:SCHEDULE	28
:SCHI	30
:ANNOTATE	33
:ESCHEDULE.	33
:SCHDATE.	33
:ASSUM	34
:EASSUM	34
:DEPEND	34
:EDEPEND	35
:XPO	35
:EXPO	35
:KEYC	35
:EKEYC	35
Schedule Examples	36

Schedule with :ASSUM, :DEPEND, and :XPO . . . . .	36
Schedule with :ANNOTATE . . . . .	38
Schedule with STITLE and DATETYPE . . . . .	40
Schedule with Staffing Information . . . . .	42
Things to Remember . . . . .	44
 <b>Appendix A. Messages</b> . . . . .	 45
Generalized Markup Language (GML) Starter Set Messages . . . . .	46
 <b>Appendix B. Related Publications and Products</b> . . . . .	 53
Publication Library Guide for the Document Composition Facility . . . . .	54
DCF Publications . . . . .	55
Related Products . . . . .	56
 <b>Glossary</b> . . . . .	 57
 <b>Index</b> . . . . .	 67

---

## Chapter 1. Introduction

This publication introduces a set of Generalized Markup Language Applications that are available with the IBM Document Composition Facility (DCF) licensed program (program number 5748-XX9). DCF is a text-processing program; its main component is the text formatter called SCRIPT/VS.

One of the major features of DCF is the ability to use the Generalized Markup Language (GML). With DCF IBM supplies a *starter set* of GML to give users a starting point in using GML (and in developing their own GML to meet their specific needs).

This book describes the tags and attributes of three additional sample GML applications for creating memos, overhead transparencies, and schedules. If you create these types of documents as a part of your job, the GML application tags can help increase your productivity and make your job easier.

IBM Publishing Systems BookMaster licensed program (Program Number 5688-015) provides an alternative set of tags and attributes for creating memos, overhead transparencies, and schedules. For more information about BookMaster, contact your local IBM Branch Office.

---

### How This Book Is Designed

The *Document Composition Facility: Generalized Markup Language Applications Guide* is designed differently than other books in the DCF library. This design uses a fold-out at the beginning of each chapter. With the fold-out opened all the way, the syntax of the tags, attributes, and values are always displayed on your left as you read about them and learn about how they're used in examples.

This book has the following chapters and appendixes:

- Chapter 1, "Introduction" describes this publication.
- Chapter 2, "Creating Memos" on page 5 describes how to use the GML application tags for creating memos.
- Chapter 3, "Creating Overhead Transparencies" on page 15 describes how to use the GML application tags for creating overhead transparencies.
- Chapter 4, "Creating Schedules" on page 27 describes how to use the GML application tags for creating schedules.
- Appendix A, "Messages" on page 45 describes the GML messages issued by DCF.
- Appendix B, "Related Publications and Products" on page 53 lists not only the books in the DCF Library and other publications related to this publication, but related products as well.

The first part of each chapter has a foldout page with the syntax for the particular set of tags that chapter describes. The pages following the foldout describe each tag, its attributes, and how they are used. Finally, the pages following the description of the tags and attributes have examples of the input you would use to create a

document and examples of what that formatted input looks like after it has been printed.

This book was formatted with the IBM Publishing Systems BookMaster licensed program and was printed on an IBM 3825 page printer. When you use the GML starter set tags and printers other than the one used to print this book, your formatted output may differ from the examples shown because of differences in column length, fonts, line spacing, device types, indentation, and other formatting differences between GML and BookMaster.

---

## The GML Applications

The GML application tags and attributes described in this book make creating memos, overhead transparencies, and schedules easy.

### What's a Memo?

Most often memos are brief, informal, written interoffice notes. Memos are also used as a form of communication from a person in one business to a person in another business. Memos have a general format that is consistent, whether the memo is to remind fellow employees of the upcoming company picnic or to acknowledge receipt of information from another company.

The GML application tags have taken some of the work out of writing memos. These tags make it possible for you to include all of the information you might want in your memo and to place that information in a logical order, from the date at the top to the copy list at the bottom. From now on, with the use of these new tags, it's possible for all of your memos to be formatted consistently.

### What's an Overhead Transparency?

Overhead transparencies are clear plastic sheets used with overhead projectors to display text or graphics on a white screen or wall. They are a valuable aid to meetings and group discussions. Overhead transparencies can be used to show meeting agendas, key points of text from a presentation or speech, and graphic displays, such as charts or illustrations.

The GML overhead transparency tags allow you to use the versatility of your Advanced Function Printing (AFP) printers or PostScript<sup>1</sup> devices to make paper masters from which overhead transparencies can be made. These tags make it possible to have a running heading, running footing, and a large, easy-to-read typeface.

### What's a Schedule?

A schedule is a tool for tracking the time and steps needed to implement a project from initial plan to completion. Schedules are used for tracking the progress of projects that can be as simple as having offices cleaned on a weekly basis or as complex as a product development project involving many functions of your business (or other businesses) over several years at various locations.

---

<sup>1</sup> PostScript is a trademark of Adobe Systems, Incorporated.



The GML schedule tags allow you to construct a graphic display showing all of the information about your project from start to finish. There are tags for displaying the start date, end date, different actions of the project, and parties responsible for those actions (just to name a few of the many functions available with schedules). There are also tags for showing how things have changed during a project (such as completing a specific action before the projected completion date).

---

## Prerequisites

Before you learn how to create memos, overhead transparencies, or schedules, you should:

1. Have DCF Release 3.2 or higher installed.
2. Understand the concepts in *Document Composition Facility: Introduction to Generalized Markup Language*.
3. Know how to use a text editor, such as XEDIT or MARKUP, to create and revise a document.
4. Know how to use GML tags to identify the parts of a document.
5. Know how to specify attributes on GML tags.
6. Know how to format and print a document using DCF.
7. Understand how to use the SCRIPT command (see *Document Composition Facility: Starter Set User's Guide*).

See Appendix B, "Related Publications and Products" on page 53 for a listing of publications that contain more information about XEDIT, MARKUP, GML, and DCF.

---

## Chapter 2. Creating Memos

To create memos, you need to specify the DSMMPROF profile. You can do this either by specifying it on the SCRIPT command profile option or by including the following control word as the first line of your input file:

```
.im dsmmprof
```

**Note:** If you imbed the profile in your input file, do not imbed it more than once. Do not imbed any other profiles when you are making memos, as it may cause unpredictable results for some tags. When you specify DSMMPROF, the DSMMPROF3 profile is automatically imbedded.

The GML starter set tags can be used along with the memo tags; however, a few formatting options will be ignored (see “Things to Remember” on page 14 for a list of these options).

The memo tags are used to specify information about the following:

- The sender of the memo.
- The receiver of the memo.
- The security-classification level of the memo.
- The subject of the memo.
- The reference of the memo.
- A copy list for the memo.

IBM Publishing Systems BookMaster provides an alternative set of tags and attributes for creating cover letters (memos). The starter set memo tags are not compatible with BookMaster cover-letter tags.

---

### Memo Tags and Their Attributes

In “Syntax for Memo Tags and Attributes” (on the fold-out page at the beginning of this chapter), there is a chart with the syntax for using the memo tags. In “Examples of Memos” on page 10, there are some examples of how to mark up a memo in a text file and how that markup looks when printed.

In this part of the chapter, you’ll find a listing of the memo tags and their attributes as well as a brief description of how they are used.

#### :LOGO

The :LOGO tag specifies the logo to be used on the memo.

**LOGONAME** Specifies the name of the graphic that will be used as a logo on the memo.

The LOGONAME should be the external name of the file that contains the image to be imbedded. The graphic file you choose as your LOGONAME must reside in storage that is accessible to SCRIPT/VS (see *Document Composition Facility: SCRIPT/VS User’s Guide* for more information about imbedding segments). The graphic must also be compatible with the printer you’re going to use to print the memo.

**Usage:** The graphic specified with the LOGONAME attribute is used as the logo for the memo. The logo is placed in the top, right-hand corner of the memo. If a logo is not specified with the :LOGO tag, no graphic is used.

**Note:** If the logo you specify does not fit in the space provided, see your system programmer.

#### :TO

The :TO tag specifies the recipient of the memo. The attributes for this tag are used to specify the name and address of the recipient.

<b>NAME</b>	Specifies the recipient’s name.
<b>COMPANY</b>	Specifies the recipient’s company or workplace for external memos. This attribute is only valid for external memos; it is ignored if it is specified for an internal memo.
<b>ADDR1</b>	Specifies the first line of the recipient’s address (such as street or building).
<b>ADDR2</b>	Specifies the second line of the recipient’s address (such as city or state).

**Usage:** All of the text for each attribute must be within single quotation marks (').

The :TO tag does not cause any text to be placed on the page. This tag only collects the information about the receiver of the memo. Use the :MEMOTXT tag to print the receiver information on the page.

**Note:** If you are using the INTERNAL value on the MEMTYPE attribute with the :MEMOTXT tag (see “:MEMOTXT” on page 8), you can specify multiple :TO tags and attributes to print at the top of the memo. If you are using the EXTERNAL value on the MEMTYPE attribute with the :MEMOTXT tag, only one name and address will print in at the top of the memo. If you’ve specified

---

## Syntax for Memo Tags and Attributes

---

<i>TAG</i>	<i>ATTRIBUTE</i>	<i>VALUE</i>	<i>TEXT</i>
:LOGO	LOGONAME =	'logo graphic name'.	
:TO	NAME = COMPANY = ADDR1 = ADDR2 =	'receiver's name' 'receiver's company name' 'first line of receiver's address' 'second line of receiver's address'.	
:FROM	NAME = ADDR1 = ADDR2 = PHONE = DEPT = TITLE =	'sender's name' 'first line of sender's address' 'second line of sender's address' 'sender's telephone number' 'sender's department' 'sender's title'.	
:MEMODATE.			<i>date of the memo</i>
:SECURITY.			<i>security level of the memo</i>
:SUBJECT.			<i>subject of the memo</i>
:REF.			<i>reference of the memo</i>
:MEMOTXT	MEMTYPE =	INTERNAL EXTERNAL.	<i>The text of your memo follows the :MEMOTXT tag.</i>
:SIGN.			<i>name of the memo's sender</i>
:CC	NAME = ADDR1 = ADDR2 =	'name of person on copy list' 'first line of address of person on copy list' 'second line of address of person on copy list'.	



multiple :TO tags and attributes, only the last name and address you've specified will print.

## :FROM

The :FROM tag specifies information about the sender of the memo. The attributes for this tag are used to specify the name and address of the sender.

- NAME** Specifies the sender's name.
- ADDR1** Specifies the first line of the sender's address (such as street or building address).
- ADDR2** Specifies the second line of the sender's address (such as city and state).
- PHONE** Specifies the sender's telephone number.
- DEPT** Specifies the sender's department.
- TITLE** Specifies the sender's title.

**Usage:** All of the text for each separate attribute must be within single quotation marks (').

If the PHONE, DEPT, or TITLE attributes are not used, space is not reserved for them in the sender's address block. Space will always be reserved for the sender's name and two address lines (attributes ADDR1 and ADDR2).

The :FROM tag does not cause any text to be placed on the page. This tag only collects the information about the sender of the memo. Use the :MEMOTXT tag to print the sender information on the page.

## :MEMODATE.

The :MEMODATE tag specifies the date the memo's sender wants to use in the address block of the sender. If this tag is not used, the default will be the current date.

**Usage:** The text associated with the :MEMODATE tag must be entered on a single line of the input document (either all on the same line as the :MEMODATE tag, or all on the next line) and cannot contain any other tags.

The :MEMODATE tag does not cause any text to be placed on the page. This tag determines the date used in the sender's address block. Use the :MEMOTXT tag to cause the date to print on the page.

## :SECURITY.

The :SECURITY tag specifies the security classification of the memo.

**Usage:** The text associated with the :SECURITY tag must be entered on a single line of the input document (either all on the same line as the :SECURITY tag, or all on the next line) and cannot contain any other tags.

If specified, the security-classification level appears in the address block at the top of the first page of the memo and at the top of all subsequent pages.

The :SECURITY tag does not cause any text to be placed on the page. This tag only collects the security level in the sender's address block. Use the :MEMOTXT tag to cause the security-level information to print on the page.

### :SUBJECT.

The :SUBJECT tag specifies the subject of the memo.

**Usage:** The text associated with the :SUBJECT tag must be entered on a single line of the input document (either all on the same line as the :SUBJECT tag, or all on the next line) and cannot contain any other tags.

The :SUBJECT tag does not cause any text to be placed on the page. This tag only collects information about the subject of the memo. Use the :MEMOTXT tag to cause the subject information to appear on the page.

### :REF.

The :REF tag specifies the reference of the memo.

**Usage:** The text associated with the :REF tag must be entered on a single line of the input document (either all on the same line as the :REF tag, or all on the next line) and cannot contain any other tags.

The :REF tag does not cause any text to be placed on the page. This tag only collects information about the reference of the memo. Use the :MEMOTXT tag to cause the reference information to appear on the page.

### :MEMOTXT

The :MEMOTXT tag causes the address of the sender and receiver of the memo to appear at the top of the memo and identifies the beginning of the text of the memo.

**MEMTYPE** Specifies whether the memo is for internal distribution or external distribution.

The INTERNAL value specifies that the memo is for internal distribution. The INTERNAL value also allows you to specify several names in the :TO section at the top of your memo. To specify more than one name and address, you must use the :TO tag for each name and address. The first name and address will print across the top of the memo, the second name and address will print under the first, and so forth. The INTERNAL value can be abbreviated as a minimum of either "I" or "i."

The EXTERNAL value specifies that the memo is for external distribution. If you are using the EXTERNAL value to the MEMTYPE attribute, only one name and address will print at the top of the memo. If you've specified more than one name and address, only the last name and address you've specified will print in this block. The EXTERNAL value can be abbreviated as a minimum of either "E" or "e."

**Note:** If no value is specified for MEMTYPE, you will get an INTERNAL memo as the default.



**Usage:** The :MEMOTXT tag causes the sender and receiver information to appear at the top of the first page of the memo.

The :MEMOTXT tag should be placed after the :TO, :FROM, :MEMODATE, :SUBJECT, and :REF tags, but prior to the start of the text of the memo. Any GML tags that can be used with :BODY tag can be used with the :MEMOTXT tag.

**Note:** The :MEMOTXT tag can be used only once in a memo. If it is used more than once, unpredictable results will occur.

## :SIGN.

The :SIGN tag ends the text of the memo so it should not be used until you have added all the text that you want in the body of the memo. The :SIGN tag should be placed after the :MEMOTXT tag and before the copy list (if a copy list is specified) in your memo.

If a name is specified using the :SIGN tag, it will be used in the signature line. If a name is not specified on the :SIGN tag, the name of the sender, as specified with the :FROM tag, will be used in the signature line.

**Usage:** The text associated with the :SIGN tag must be entered on a single line of the input document (either all on the same line as the :SIGN tag, or all on the next line) and cannot contain any other tags.

## :CC

The :CC tag is used for specifying a list of names of the people who are to receive copies of the memo. Attributes to this tag are used for specifying their names and addresses. You must use a separate :CC tag and attributes for each name you want on the copy list.

**NAME** Specifies the first name on the copy list.

**ADDR1** Specifies the first line of the address of the first name on the copy list (such as street or building address).

**ADDR2** Specifies the second line of the address of the first name on the copy list (such as city and state).

**Usage:** If your memo needs a copy list, use the :CC tag to start the list of names to be included in the copy list. All of the text for each attribute must be within single quotation marks (''). The :CC tag should appear after the :MEMOTXT and :SIGN tag or at the end of the text of the memo if the :SIGN tag isn't specified. If the :CC tag does not come *after* the :MEMOTXT tag, unpredictable DCF error messages will occur.

---

### Examples of Memos

On the following pages are some examples of how to mark up a memo in a text file and how that markup looks when printed.

#### Internal Memo

Here is an example of an internal memo:

```
:logo logoname=earthsta.
:security.Internal Use Only
:memodate.July 15, 1989
:to
  name='T. Williams, Ph.D.'
  addr1='Dept. M'
  addr2='Building 12'.
:from
  name='Dr. J. J. Jackson'
  addr1='Dept. X'
  addr2='Building 7'
  title='Head of Operations'
  dept='Captial Equipment Acquisitions'
  phone='ext. 2068'.
:subject.Computer purchase
:ref.Your memo of 07/12/89
:memotxt memtype=internal.
:p.
I am in total agreement with your findings regarding the necessity
for your department to computerize its operation.
Your study was complete and to the point.
Order the equipment outlined in the referenced memo; funds have
been made available for this purpose.
:p.
Keep up the good work!
:sign.
:cc
  name='Ms. L. Dominguez'
  addr1='Dept. J'
  addr2='Building M'.
:cc
  name='Mr. J. Tinker'
  addr1='Dept. I'
  addr2='Building B'.
:cc
  name='Mr. C. Evers'
  addr1='Dept. X'
  addr2='Building M'.
:cc
  name='Ms. S. Chance'
  addr1='Dept. X'
  addr2='Building M'.
```

**To:**                      Name                      Int Zip                      Area  
                              T. Williams, Ph.D.    Dept. M    Building 12

Classification: *Internal Use Only*



**From:**

Name: Dr. J. J. Jackson  
 Dept/Phone: Capital Equipment Acquisitions, ext. 2068  
 Title: Head of Operations  
 Int Zip: Dept. X  
 City & State: Building 7  
 Date: July 15, 1989

Subject: Computer purchase  
 Reference: Your memo of 07/12/89

I am in total agreement with your findings regarding the necessity for your department to computerize its operation. Your study was complete and to the point. Order the equipment outlined in the referenced memo; funds have been made available for this purpose.

Keep up the good work!

Dr. J. J. Jackson

**CC:**                      Name                      Int Zip                      Area  
                              Ms. L. Dominguez    Dept. J                      Building M  
                              Mr. J. Tinker            Dept. I                      Building B  
                              Mr. C. Evers            Dept. X                      Building M  
                              Ms. S. Chance           Dept. X                      Building M

### External Memo

Here is an example of an external memo:

```
:logo logoname=earthsta.
:security.Your Eyes Only
:memodate.July 15, 1989
:to
  name='Ms. B. Weaver'
  company='Buffalo Computing
  addr1='1876 Nathan Brittles Blvd.'
  addr2='Buffalo, WY 82834'.
:from
  name='T. Williams, Ph.D.'
  addr1='Satelite Uplink, Inc.'
  addr2='Shiprock, NM 87420'
  title='Head of Operations'
  dept='Department M'
  phone='ext. 9298'.
:subject.Computer purchase
:ref.Our telephone conversation of 07/12/89
:memotxt memtype=external.
:p.
Funds have been made available for purchase of the computer system
that we discussed the other day by telephone.
Please start the process by which we can order this system.
:p.
Thank you for all your assistance in this matter.
:sign.
:cc
  name='Dr. J. J. Jackson'
  addr1='Dept. X/Building 7'
  addr2='Shiprock, NM'.
:cc
  name='Ms. L. Dominguez'
  addr1='Dept. J/Building M'
  addr2='Shiprock, NM'.
:cc
  name='Mr. J. Tinker'
  addr1='Dept. I/Building B'
  addr2='Shiprock, NM'.
:cc
  name='Mr. C. Evers'
  addr1='Dept. X/Building M'
  addr2='Shiprock, NM'.
:cc
  name='Ms. S. Chance'
  addr1='Dept. X/Building M'
  addr2='Shiprock, NM'.
```



Classification: ***Your Eyes Only***

**To:**

Name: Ms. B. Weaver  
Company: Buffalo Computing  
Address: 1876 Nathan Brittles Blvd.  
City & State: Buffalo, WY 82834

**From:**

Name: T. Williams, Ph.D.  
Dept/Phone: Department M, ext. 9298  
Title: Head of Operations  
Int Zip: Satellite Uplink, Inc.  
City & State: Shiprock, NM 87420  
Date: July 15, 1989

Subject: Computer purchase  
Reference: Our telephone conversation of 07/12/89

Funds have been made available for purchase of the computer system that we discussed the other day by telephone. Please start the process by which we can order this system.

Thank you for all your assistance in this matter.

T. Williams, Ph.D.

**CC:**

Dr. J. J. Jackson	Dept. X/Building 7	Shiprock, NM
Ms. L. Dominguez	Dept. J/Building M	Shiprock, NM
Mr. J. Tinker	Dept. I/Building B	Shiprock, NM
Mr. C. Evers	Dept. X/Building M	Shiprock, NM
Ms. S. Chance	Dept. X/Building M	Shiprock, NM

---

### Things to Remember

There are a few things that you should remember when using the GML memo tags:

- Blank lines in your text file are ignored and will not leave space on your formatted and printed page.
- The following is a list of formatting commands that are ignored when you use the memo tags:
  - SYSVAR S is ignored; a one-column format will be used.
  - SYSVAR X is ignored; a cross-reference listing will not be created.

---

## Chapter 3. Creating Overhead Transparencies

To create overhead transparencies, you need to specify the DSMTPROF profile. You can do this either by specifying it on the SCRIPT command profile option or by including the following control word as the first line of your input file:

```
.im dsmtprof
```

The DSMTPROF profile allows you to use larger, easier-to-read fonts to prepare overhead transparencies for your presentations. You can use the overhead transparency tags (:TRHEAD and :TRFOOT) and their attributes for specifying running headings and running footings.

**Note:** If you imbed the profile in your input file, do not imbed it more than once. Do not imbed any other profiles when you are making overhead transparencies, as it may cause unpredictable results for some tags. When you specify DSMTPROF, the DSMPROF3 profile is automatically imbedded.

Most GML starter set tags can be used along with the overhead transparency tags; however, there are a few tags and formatting options that will be ignored (see “Things to Remember” on page 26 for a list of these tags and options).

IBM Publishing Systems BookMaster provides a named style for creating overhead transparencies.

---

### Overhead Transparency Tags and Their Attributes

In “Syntax for Overhead Transparency Tags and Attributes” (on the fold-out page at the beginning of this chapter), there is a chart with the syntax for using the overhead transparency tags. In “Examples of Overhead Transparencies” on page 18, there are some examples of how an overhead transparency is marked up in your text file and of how that markup looks when printed.

In this part of the chapter, you’ll find a listing of the overhead transparency tags and their attributes as well as a brief description of how they are used.

#### **:TRHEAD**

The **:TRHEAD** tag specifies information to be placed in the heading of each transparency. Attributes on this tag are used to specify where that information will be placed in the heading and how it will appear on the printout.

**Note:** To ensure that running headings do *not* appear on the title page of your overhead transparencies, place the **:TRHEAD** tag after the **:BODY** tag.

**LEFT** Specifies the text you want placed on the left side of the heading. If text is not specified for the left side of the heading, the default is the value of the security level in the **:GDOC** tag. If no text is specified and no security level was specified in the **:GDOC** tag, the **LEFT** side of the heading is left blank.

**RIGHT** Specifies the text you want placed on the right side of the heading. If no text is specified for the right side of the heading, it is left blank.

**SUBJECT** Specifies the text you want placed in the centered heading. If no text is specified for the **SUBJECT** of the heading, it is left blank.

**LOGONAME** Specifies the name of the graphic to be used as the logo.

The **LOGONAME** should be the external name of the file that contains the image to be imbedded. The graphic file you choose as your **LOGONAME** must reside in storage that is accessible to **SCRIPT**, and must be available to **PSF** (Print Services Facility) at print time. (See *Document Composition Facility: SCRIPT/VS User’s Guide* for more information about imbedding segments.) The graphic must also be compatible with the printer you’re going to use to print the memo.

**LOGOPOS** Specifies the horizontal alignment of the logo within the heading. The values of **LEFT** or **RIGHT** (or any abbreviation of those values) can be specified. For example the letter “L” can be used to specify **LEFT**, or the letter “R” can be used to specify **RIGHT**. Unless otherwise specified, the default for **LOGOPOS** is **RIGHT**.

**Note:** If you have specified text for the same position as your logo, the text will be printed and the logo will not.

**Usage:** If a value of an attribute contains a blank or special characters, they must be enclosed in single quotation marks.



---

## Syntax for Transparency Tags and Their Attributes

<i>TAG</i>	<i>ATTRIBUTE</i>	<i>VALUE</i>
:TRHEAD	LEFT =	'information in left heading'
	RIGHT =	'information in right heading'
	SUBJECT =	'subject information'
	LOGONAME =	'logo graphic name'
	LOGOPOS =	LEFT RIGHT.
:TRFOOT	LEFT =	'information in left footing'
	RIGHT =	'information in right footing'
	CENTER =	'information in center footing'
	LOGONAME =	'logo graphic name'
	LOGOPOS =	LEFT RIGHT.



**Processing:** If no :TRHEAD tag is specified, no text will print in the heading unless a security attribute was specified on the :GDOC tag. The security level is then left aligned in the heading. A horizontal rule will separate the heading from the text of the transparencies.

The :TRHEAD information can be used many times in the document but will take effect on the current formatted page. If more than one :TRHEAD is specified for the same page, the most recent is used.

## :TRFOOT

The :TRFOOT tag specifies information to be placed in the footing of each transparency. Attributes on this tag are used to specify where that information will be placed in the footing and how it will appear on the printout.

**Note:** To ensure that running footings do *not* appear on the title page of your overhead transparencies, place the :TRFOOT tag after the :BODY tag.

- LEFT** Specifies the text you want placed on the left side of the footing. If text is not specified, the date you used on the :DATE tag will be placed here. If no text or date is specified, the left side of the footing contains the current system date.
- RIGHT** Specifies the text you want placed on the right side of the footing. If text is not specified, the page number will be placed here.
- CENTER** Specifies the text you want placed in the center of the footing. If text is not specified, the default text will be placed in this area in the following order:
1. Text from the most recent :H0 or :H1 tag.
  2. Residual text from the most recent :H0 or :H1 tag.
  3. The STITLE from the :TITLE tag.
  4. The title from the :TITLE tag.
- LOGONAME** Specifies the name of the graphic to be used as the logo.
- The LOGONAME should be the external name of the file that contains the image to be imbedded. The graphic file you choose as your LOGONAME must reside in storage that is accessible to SCRIPT, and must be available to PSF at print time. (See *Document Composition Facility: SCRIPT/VS User's Guide*, for more information about imbedding segments.) The graphic must also be compatible with the printer you're going to use to print the memo.
- LOGOPOS** Specifies the horizontal alignment of the logo within the footing. The values of LEFT or RIGHT (or any abbreviation of those values) can be specified. For example, the letter "L" can be used to specify LEFT, or the letter "R" can be used to specify RIGHT. Unless otherwise specified, the default for LOGOPOS is RIGHT.
- Note:** If you have specified text for the same position as your logo, the text will be printed and the logo will not.

**Usage:** If a value of an attribute contains a blank or special characters, it must be enclosed in single quotation marks. Only one line is available for text in a footing. If the text specified will not fit in the footing, it will print over itself until all of the specified text is printed.

**Processing:** If no :TRFOOT tag is specified, the default footing will be formatted.

The :TRFOOT information can be specified many times in the document but will take effect on the current formatted page. If more than one is specified for a page, the most recent is used.

---

## Examples of Overhead Transparencies

On the following pages are some examples of how to mark up overhead transparencies in a text file and of how that markup looks when printed.

### Title-Page Example

Here's an example of how to mark up a title page:

```
:gdoc sec='Internal Use Only'.  
:titlep.  
:title  
  stitle='Transparencies'.Creating Overhead Transparencies  
:etitlep.
```

## ***Creating Overhead Transparencies***

***Internal Use Only***

***Transparencies***

### Head-Level Examples

Here is an example of a head-level one:

```
:body.  
:trhead left='Left Heading'  
      subject='Subject Text'  
      logoname='ibm800'  
      logopos=right.  
:trfoot left='Left Footing'  
      right='Right Footing'  
      center='Center Footing'.  
:h1.Example of a Head-Level One
```

*Left Heading*

**IBM**

---

***Subject Text***

---

***Example of a Head-Level One***

*Left Footing*

*Center Footing*

*Right Footing*

Here are some examples of other head levels:

:h2.Example of a Head-Level Two  
:h3.Example of a Head-Level Three  
:h4.Example of a Head-Level Four



*Left Heading*

**IBM**

---

## ***Subject Text***

---

***Example of a Head-Level Two***

**Example of a Head-Level Three**

***Example of a Head-Level Four***

*Left Footing*

*Center Footing*

*Right Footing*

### Text Example

Here's an example of some text:

```
:p.This paragraph was started using the &gml.p tag.  
Following is an unordered list.  
:ul.  
:li.This is the first item of the unordered list.  
:li.This is the second item of the unordered list.  
:eul.  
:p.This paragraph is followed by an ordered list.  
:ol  
:li.This is the first item of the ordered list.  
:li.This is the second item of the ordered list.  
:eol.  
:note.  
This note was made using the &gml.NOTE tag.  
:egdoc.
```

*Left Heading***IBM**

---

## ***Subject Text***

---

This paragraph was started using the :p tag. Following is an unordered list.

- This is the first item of the unordered list.
- This is the second item of the unordered list.

This paragraph is followed by an ordered list.

1. This is the first item of the ordered list.
2. This is the second item of the ordered list.

**Note:** This note was made using the :NOTE tag.

*Left Footing**Center Footing**Right Footing*

---

## Things to Remember

There are a few things that you should remember when using the overhead transparency tags:

- Blank lines in your text file are ignored and will not leave space on your formatted and printed page.
- Leading blanks in your line of text are ignored.
- The following is a list of SCRIPT formatting commands that are ignored when you use the overhead transparency tags:
  - SYSVAR S is ignored; a one-column format will be used.
  - SYSVAR D is ignored; simplex is done.
  - SYSVAR X is ignored; a cross-reference listing will not be created.
- There are a few GML starter set tags that will not print when you are using DSMTPROF. The following tags are not valid in an overhead transparency. The results of these tags are not included in your transparencies and an error message is issued.

```
:INDEX
:IH1
:IH2
:IH3
:I1
:I2
:I3
:FRONTM
:BACKM
```

- Use of this profile is not supported on the 4224 printer, the 3812 printer, and line printers.

---

## Chapter 4. Creating Schedules

To create schedules, you need to specify the DSMSPROF profile. You can do this either by typing it on the SCRIPT command line or by including the following control word as the first line of your input file:

```
.im dsmsprof
```

**Note:** If you imbed the profile in your input file, do not imbed it more than once. Do not imbed any other profiles when you are making schedules, as it may cause unpredictable results for some tags. When you specify DSMSPROF, the DSMPROF3 profile is automatically imbedded.

The GML starter set tags can be used along with the schedule tags.

IBM Publishing Systems BookMaster provides an alternative set of tags and attributes for creating schedules, but there are some differences between those tags and the GML starter set schedule tags.

---

## Schedule Tags and Their Attributes

In “Syntax for Schedule Tags and Their Attributes” (on the fold-out page at the beginning of this chapter), there is a chart with the syntax for using the schedule tags. In “Schedule Examples” on page 36, there are some examples of how to mark up a schedule in a text file and of how that markup looks when printed.

In this part of the chapter, you’ll find a listing of the schedule tags and their attributes as well as a brief description of how they are used. All of the text for each attribute must be enclosed in single quotation marks (').

### :SCHEDULE

The :SCHEDULE tag starts the schedule. The text on the :SCHEDULE tag is the schedule title. The :ESCHEDULE tag ends the schedule.

The schedule title text associated with the :SCHEDULE tag must be entered on a single line of the input document (either all on the same line as the :SCHEDULE tag or all on the line following the :SCHEDULE tag) and cannot contain any other tags.

If you do not add text to this tag as the schedule title, you can use a head tag before the :SCHEDULE tag to specify the title of the schedule, and it will appear in the table of contents. If you use both a head tag and text on the :SCHEDULE tag, the text in the :SCHEDULE tag acts as a conditional page break causing the schedule to print on the following page.

**Note:** If the values for the START and END attributes are invalid, an error occurs that stops processing. There are no default values for these attributes.

**START** Specifies the start date of the schedule. This attribute is required.

**Note:** If MONTH or GMONTH is specified on the INTERVAL attribute, or the INTERVAL attribute is not specified at all, the date must be in the format “mm/yy” where “mm” is the month, and “yy” is the year. If QUARTER is specified on the INTERVAL attribute, the date must be in the format “q/yy” where “q” is the quarter and “yy” is the year. If YEAR is specified on the INTERVAL attribute, the date must be in the format “yy” where “yy” is the year.

**END** Specifies the end date of the schedule. This attribute is required. The date must be in the same format that was used on the START attribute.

**DATE** Specifies the date to be used as the current date for the schedule. If this attribute is used, the date must be in either the USA format or the EUROPEAN format. (See the description of the DATETYPE attribute for further details.) If this attribute is invalid or not specified, the current system date is used.

**ACTSIZE** Specifies the width of the activities column for the activities specified with the :SCHI tag. Any valid space unit notation can be used. If this attribute is not used, one and one-half inches is reserved for the activities in the first column of the schedule. If ACTSIZE is specified, it should be at least one-quarter of an inch. If the ACTSIZE is invalid, unpredictable DCF error messages occur.

## Syntax for Schedule Tags and Their Attributes

<i>TAG</i>	<i>ATTRIBUTE</i>	<i>VALUE</i>	<i>TEXT</i>
:SCHEDULE	START = END = DATE = ACTSIZE = DATETYPE = STITLE = LEGEND = TODAY INTERVAL =	'schedule start date' 'schedule end date' 'current schedule date' 'activity size' USA EUROPEAN 'short title' YES NO MONTH GMONTH QUARTER YEAR.	<i>schedule title</i>
:SCHI	PSTART = PEND = ASTART = AEND = SSTART = SEND = RESP = HEAD NODBOX.	'projected start date' 'projected end date' 'actual start date' 'actual end date' 'slip start date' 'slip end date' 'responsible party'	<i>text for SCHI</i>
:ANNOTATE	DATE = ALIGN =	'date for annotation' LEFT RIGHT CENTER.	<i>annotation text</i>
:ESCHEDULE.			
:SCHDATE.			
:ASSUM :LI. :EASSUM.	COMPACT.		<i>ASSUM text</i>
:XPO :LI. :EXPO.	COMPACT.		<i>XPO text</i>
:KEYC :LI. :EKEYC.	COMPACT.		<i>KEYC text</i>
:DEPEND :LI. :EDEPEND.	COMPACT.		<i>DEPEND text</i>





<b>DATETYPE</b>	Specifies the format in which the date will be displayed. The valid values for DATETYPE are:
<b>USA</b>	Indicates the USA date format: 'mm/dd/yy' where 'mm' indicates month, 'dd' indicates day, and 'yy' indicates year. The value USA can be abbreviated as a minimum of either "u" or "U."
<b>EUROPEAN</b>	Indicates the European date format: 'dd/mm/yy' where 'dd' indicates day, 'mm' indicates month, and 'yy' indicates year. The value EUROPEAN can be abbreviated as a minimum of either "e" or "E."
	If DATETYPE is not specified, the USA format for dates is expected.
<b>STITLE</b>	Specifies the short title of the schedule. The short title is used in the running footing, rather than the complete title.
<b>LEGEND</b>	Specifies that a legend describing the symbols used in the schedule is to be included at the end of the schedule.
<b>YES</b>	Indicates that you want to have a legend included at the end of the schedule. The value YES can be abbreviated as a minimum of either "y" or "Y."
<b>NO</b>	Indicates that you do not want to have a legend included at the end of the schedule. The value NO can be abbreviated as a minimum of either "n" or "N."
	The default for LEGEND is NO, but if an invalid value is specified, the default is YES.
<b>TODAY</b>	Specifies that an arrow is to be placed at the bottom of the schedule, indicating where the current system date occurs on the schedule.
<b>INTERVAL</b>	Specifies the interval used in the schedule. Valid values and the way those values format for the INTERVAL attribute are as follows:
<b>MONTH</b>	Indicates that each year in the schedule is divided into 12 months, and the months are labeled: "Jan" "Feb" and so forth. The value MONTH can be abbreviated as a minimum of either "m" or "M."
<b>GMONTH</b>	Indicates that each year in the schedule is divided into 12 months of 30 days each, and the months are labeled generically: "Month 1," "Month 2," and so forth. The value GMONTH can be abbreviated as a minimum of either "g" or "G."
<b>QUARTER</b>	Indicates that each year in the schedule is divided into four quarters and the quarters are labeled: "1st Qtr," "2nd Qtr," and so forth. The value QUARTER can be abbreviated as a minimum of either "q" or "Q."
<b>YEAR</b>	Indicates that each year in the schedule is not divided. The value YEAR can be abbreviated as a minimum of either "y" or "Y."

The default for INTERVAL is "MONTH."

**Usage:** The :SCHEDULE tag starts a schedule. After the :SCHEDULE tag, the :SCHI tags specify the different activities in the schedule. After the last :SCHI tag in the schedule, the :ESCHEDULE tag is used to end the schedule.

**Processing:** When the :SCHEDULE tag is encountered, a new page is started, using the schedule title in a level 1 heading, then a box, with months, quarters, or years as the schedule heading, is begun. The START, END, and INTERVAL attributes determine the intervals used in the schedule heading. If a schedule title is not supplied with the :SCHEDULE tag, no page eject occurs and a level 1 heading is not created.

The default width of the schedule is the current line length for 0° or 180° rotated devices. For 90° and 270° rotated devices, the width is determined from the page width.

Each occurrence of the :SCHI tag causes a new activity to be added to the schedule.

When the :ESCHEDULE tag is encountered, the box created for the schedule is ended. A legend prints explaining the symbols used in the schedule if the LEGEND attribute is used.

### :SCHI

The :SCHI (schedule item) tag specifies an activity in the schedule. The text on the :SCHI tag is the activity description. The text associated with the :SCHI tag must be entered on a single line of the input document (either all on the same line as the last attribute to the :SCHI tag or all on the line following the last attribute to the :SCHI tag) and cannot contain any other tags.

Attributes determine the dates associated with the activity. For all attributes that require a date, the date should be in either the USA format or the EUROPEAN format (see the description of the DATETYPE attribute on page 29).

**Note:** Do not specify a single start or end date that is the same as the last day of the schedule.

**PSTART** Specifies the *projected* start date of the activity. If the PEND attribute is used, the PSTART attribute is required; otherwise, this attribute is optional. The date given on this attribute must be earlier than the date given on the PEND attribute.

**Note:** If a date that is outside the START and END date attributes of the :SCHEDULE tag has been specified, an error message is issued and the default for PSTART will be the START date specified on the :SCHEDULE tag. If an invalid date is specified, the PSTART attribute is ignored.

**PEND** Specifies the *projected* end date of the activity. If the PSTART attribute is used, this attribute is required; otherwise, this attribute is optional. The date given on this attribute must be later than the date specified on the PSTART attribute.

**Note:** If a date that is outside the START and END date attributes of the :SCHEDULE tag has been specified, an error message is issued and the default for PEND will be the END date specified on the

:SCHEDULE tag. If an invalid date is specified, the PEND attribute is ignored.

**ASTART** Specifies the *actual* start date of the activity. If the AEND attribute is used, the ASTART attribute is required; otherwise, this attribute is optional. The date given on this attribute must be earlier than the date specified on the AEND attribute.

**Note:** If a date that is outside the START and END date attributes of the :SCHEDULE tag has been specified, an error message is issued and the default for ASTART will be the START date specified on the :SCHEDULE tag. If an invalid date is specified, the ASTART attribute is ignored.

**AEND** Specifies the *actual* end date of the activity. This attribute is optional. The date given on this attribute must be later than the date specified on the ASTART attribute.

**Note:** If a date that is outside the START and END date attributes of the :SCHEDULE tag has been specified, an error message will be issued and the default for AEND will be the END date specified on the :SCHEDULE tag. If an invalid date is specified, the default is the value specified with the DATE attribute on the :SCHEDULE tag.

**SSTART** Specifies the *slip* start date of the activity. This attribute is optional.

**Note:** If a date that is outside the START and END date attributes of the :SCHEDULE tag has been specified, an error message is issued and the default for SSTART will be the START date specified on the :SCHEDULE tag. If an invalid date is specified, the SSTART attribute is ignored.

**SEND** Specifies the *slip* end date of the activity. This attribute is optional.

**Note:** If a date that is outside the START and END date attributes of the :SCHEDULE tag has been specified, an error message is issued and the default for SEND will be the END date specified on the :SCHEDULE tag. If an invalid date is specified, the SEND attribute is ignored.

**RESP** Specifies the party responsible for the activity. The text given on this attribute is placed in *italics* on the right side of the activities column. This attribute is optional.

**HEAD** Specifies that this :SCH tag should be treated as a heading within the schedule. Any dates (PSTART, PEND, ASTART, AEND, SSTART, and SEND) are ignored. HEAD can be abbreviated as a minimum of either "h" or "H." This attribute is optional.

**NODBOX** Specifies the information from this :SCH tag should not appear on the datelist box created by the :SCHDATE tag. NODBOX can be abbreviated as a minimum of either "n" or "N." This attribute is optional.

**Usage:** The SSTART and SEND attributes are independent of each other. As many SSTART attributes can be specified as desired, regardless of the number of or absence of SEND attributes, and the converse is also true.

The dates on all six attributes must be in either the USA format or the EUROPEAN format. (See the description of the DATETYPE attribute on 29.) If

the year ("yy") is not given, the current system year is used. To be safe, always specify the year. If you allow the processing to use the default year, the default year changes on January 1 of the following year. For example, your schedule has a PSTART date of 10/25/88, and instead of specifying 1988, you allow the year to default to the current system year. If you format your schedule *after* 01/01/89, the default year is now 1989 so your PSTART date of 10/25/88 will change to 10/25/89.

You can specify multiple start and end dates for a given activity by using the PSTART, PEND, ASTART, AEND, SSTART, and SEND attributes. If too many dates are specified, you may get undesirable results. The first PSTART attribute specified is paired with the first PEND attribute specified, and so forth.

**Note:** Depending on the size of the schedule, the INTERVAL value specified on the :SCHEDULE tag, and the device type you are formatting for, dates that are too close together may cause an error message about overlapping dates. This same message will occur if the dates are not specified in chronological order.

Likewise, the first ASTART attribute specified is paired with the first AEND attribute specified. SSTART and SEND dates are not paired, because they are independent of each other. For each pair of projected and actual dates, the start date must precede the end date. There must be an equal number of PSTART and PEND dates. There can be one more ASTART date than AEND date. The missing AEND date uses the value specified with the DATE attribute on the :SCHEDULE tag. See "Schedule with :ANNOTATE" on page 38 for an example of multiple start and end dates.

Use the HEAD attribute to indicate a :SCHI tag that should be treated as a heading within the schedule. Use the NODBOX attribute to indicate a :SCHI tag that should not have its information placed in the datelist box created with the :SCHDATE tag.

**Processing:** A box is drawn for the *projected* dates of an activity. The box starts at the *projected* start date (PSTART) and ends at the *projected* end date (PEND).

If the activity does not span a multiple number of days but occurs on one day, the projected start date and the projected end date are the same.

**Note:** The symbols you get on your printed schedule may vary from one printer or device to another. Use the LEGEND attribute for the :SCHEDULE tag to print a legend of the symbols you will get on your schedule.

A thick horizontal rule is drawn to show the *actual* dates of an activity. For an actual date that has an AEND attribute specified, a solid horizontal rule is used. If the AEND date was not specified, and the current system date is used as an ending date, a shaded horizontal rule is used. The rule starts at the *actual* start date (ASTART) and ends at the *actual* end date (AEND). If the activity has not yet ended, there is no AEND attribute for that activity, so the rule is drawn up to the current system date. See "Schedule Examples" on page 36 for examples of the difference in rules for the actual dates for an activity that has ended and for an activity that has not ended.

**Note:** If you do not specify an AEND date, the default is the value specified with the DATE attribute on the :SCHEDULE tag. If the ASTART date is *later* than the DATE value in this case, you will receive an error message.

If the activity does not span a multiple number of days but occurs on one day, the actual start date and the actual end date are the same.

When the HEAD attribute is used, all date-type attributes (PSTART, PEND, ASTART, AEND, SSTART, and SEND) are ignored. The activity description is printed in a bold font and the vertical rules depicting months (or quarters or years) are omitted.

## :ANNOTATE

The :ANNOTATE tag creates an annotation on the schedule for the specified date.

**DATE** Specifies the date at which the annotation is to start. This attribute is required. The date must be in either the USA format or the EUROPEAN format. (See the description of the DATETYPE attribute for further details.) If this attribute is not specified, a message will be issued. If an invalid date is specified, the default is the value specified with the DATE attribute on the :SCHEDULE tag.

**ALIGN** Specifies the horizontal alignment of the annotation at the date specified. The valid values for ALIGN are LEFT, RIGHT, and CENTER. LEFT, RIGHT, and CENTER may be abbreviated as a minimum of either "l" or "L," "r" or "R," and "c" or "C," respectively. If this attribute is not specified, the annotation text is left-aligned at the date specified.

**Usage:** The :ANNOTATE tag is used to place annotation text within the schedule. The text associated with the :ANNOTATE tag must be entered on a single line of the input document (either all on the same line as the :ANNOTATE tag or all on the line following the :ANNOTATE tag) and cannot contain any other tags.

The ALIGN attribute is optional.

To annotate a specific activity, place the :ANNOTATE tag *after* the :SCHI tag for that activity or you will get an error message.

This tag is useful in distinguishing between multiple start and end dates on the same activity. This tag is also useful for creating staffing information. (See "Schedule with Staffing Information" on page 42 for an example of this usage.)

**Processing:** The text specified on the :ANNOTATE tag is placed in the schedule, starting at the date specified and using the alignment specified. The text is not wrapped, so you must take care that the text from one annotation does not print over text from another annotation on your printed schedule. If an alignment is specified, but the text will not fit in the available space, the annotation text that doesn't fit will be moved. See "Schedule with :ANNOTATE" on page 38 for an example of annotation.

## :ESCHEDULE.

An :ESCHEDULE tag is required.

## :SCHDATE.

The :SCHDATE tag creates a table showing the dates of the schedule in numeric format.

**Usage:** The :SCHDATE tag produces a table separate from the schedule itself. This table contains the dates in the schedule. The :SCHDATE tag cannot be placed between a :SCHEDULE and :ESCHEDULE tag or prior to the first :SCHEDULE tag in the document or you will get an error message.

Normally, the :SCHDATE tag follows the :ESCHEDULE tag that ends the schedule for which you want a list of dates printed. If you have the following input:

```
:schedule.  
...  
:eschedule.  
:schdate.  
:schedule.  
...  
:eschedule.  
:schdate.
```

the first :SCHDATE tag produces the list of dates for the first schedule, and the second :SCHDATE tag produces the list of dates for the second schedule. If you do not place your :SCHDATE tag immediately following an :ESCHEDULE tag, the :SCHDATE tag produces a list of dates for the most recent schedule.

**Processing:** When the :SCHDATE tag is encountered, a table is produced that contains all the dates in the most recent schedule.

### :ASSUM

The :ASSUM tag starts a list of assumptions for the schedule. The list of assumptions follows the schedule.

**COMPACT** Specifies that the list should be compacted. This provides the same function as the COMPACT attribute on the GML starter set list tags.

**Usage:** The :ASSUM tag starts a list of assumptions in the schedule. Each assumption is identified by a :LI tag.

**Processing:** When the :ASSUM tag is encountered, a level 4 heading with the word “Assumptions” is formatted, and an unordered list is started. Each assumption appears as an unordered list item.

### :EASSUM

The :EASSUM tag must be specified to end the list.

### :DEPEND

The :DEPEND tag starts a list of dependencies for the schedule. The list of dependencies follows the schedule.

**COMPACT** Specifies that the list should be compacted. This provides the same function as the COMPACT attribute on the GML starter set list tags.

**Usage:** The :DEPEND tag starts a list of dependencies in the schedule. Each dependency is identified by a :LI tag.

**Processing:** When the :DEPEND tag is encountered, a level 4 heading with the word “Dependencies” is formatted, and an unordered list starts. Each dependency appears as an unordered list item.

## :EDEPEND

The :EDEPEND tag must be specified to end the list.

## :XPO

The :XPO tag starts a list of exposures for the schedule. The list of exposures follows the schedule.

**COMPACT** Specifies that the list should be compacted. This provides the same function as the COMPACT attribute on the GML starter set list tags.

**Usage:** The :XPO tag starts a list of exposures in the schedule. Each exposure is identified by a :LI tag.

**Processing:** When the :XPO tag is encountered, a level 4 heading with the word “Exposures” is formatted, and a unordered list is started. Each exposure appears as an unordered list item.

## :EXPO

The :EXPO tag must be specified to end the list.

## :KEYC

The :KEYC tag starts a list of checkpoints for the schedule. The list of checkpoints follows the schedule.

**COMPACT** Specifies that the list should be compacted. This provides the same function as the COMPACT attribute on the GML starter set list tags.

**Usage:** The :KEYC tag starts a list of checkpoints in the schedule. Each checkpoint is identified by a :LI tag.

**Processing:** When the :KEYC tag is encountered, a level 4 heading with the words “Key Checkpoints” is formatted, and a unordered list is started. Each checkpoint appears as an unordered list item.

## :EKEYC

The :KEYC tag must be specified to end the list.

---

### Schedule Examples

The following pages show some sample input and printed output using the schedule tags.

#### Schedule with :ASSUM, :DEPEND, and :XPO

This example uses the :ASSUM. tag, the :DEPEND tag, and the :XPO tag.

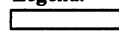
```
:schedule actsize='1.75i' start='3/88' end='7/88' today
      legend=yes date='10/1/87'.New Racing Bicycle Project
:schi pstart='3/30/88' pend='3/30/88'
      astart='3/30/88' aend='3/30/88'.Specification Review
:schi pstart='4/21/88' pend='6/5/88'
      astart='4/29/88' aend='6/1/88'
      sstart='4/28/88'.Education
:schi pstart='4/02/88' pend='4/12/88'
      astart='4/4/88' aend='4/14/88'.Design Reviews
:schi pstart='3/30/88' pend='3/30/88'
      astart='4/16/88' aend='4/16/88'.Specifications Approved
:schi pstart='4/14/88' pend='4/16/88'
      astart='4/16/88' aend='4/17/88'.Prototype Review
:schi pstart='5/10/88' pend='5/26/88'
      astart='5/12/88' aend='5/26/88'.Early Production Review
:schi pstart='5/26/88' pend='5/30/88'
      astart='5/26/88' aend='5/30/88'
      sstart='5/30/88' send='6/03/88'.Informal Track Test
:schi pstart='6/15/88' pend='6/20/88'
      astart='6/17/88' aend='6/22/88'.Race Test
:schi pstart='6/30/88' pend='6/30/88'.Announce Product
:schi pstart='7/4/88' pend='7/9/88'.Final Race Test
:schi pstart='7/15/88' pend='7/15/88'.Ship Product
:eschedule.
:assum.
:li.The material necessary to manufacture production frames is
in stock.
:li.There are nine manufacturing workers and five racers
assigned to this effort.
:li.Testing will be done in all applicable environments.
:eassum.
:depend compact.
:li.Early production model available to run the tests.
:li.Experienced personnel available to test the product.
:li.No additional education needed by manufacturing personnel.
:li.Maintenance support for all races.
:edepend.
:xpo.
:li.Announce before end of formal test.
:expo.
```



## New Racing Bicycle Project

Activity	1988				
	March	April	May	June	July
Specification Review	◆				
Education		▬			
Design Reviews		▬			
Specifications Approved	◆	◆			
Prototype Review		▬			
Early Production Review			▬		
Informal Track Test			▬	▲	
Race Test				▬	
Announce Product				◆	
Final Race Test					▬
Ship Product					◆

### Legend:



projected day range



projected single date



actual day range (complete)



actual single date



actual day range (ongoing)



slip start date



slip end date

### Assumptions

- The material necessary to manufacture production frames is in stock.
- There are nine manufacturing workers and five racers assigned to this effort.
- Testing will be done in all applicable environments.

### Dependencies

- Early production model available to run the tests.
- Experienced personnel available to test the product.
- No additional education needed by manufacturing personnel.
- Maintenance support for all races.

### Exposures

- Announce before end of formal test.

### Schedule with :ANNOTATE

This example uses the INTERVAL attribute on the :SCHEDULE tag, the :ANNOTATE tag, multiple start and end dates, and the RESP attribute on the :SCHI tag:

```
:schedule start='1/88' end='4/90' interval=quarter today
      date='6/24/88'.Computer X39 and Operating System
:schi pstart='1/1/88' pend='3/1/88' pstart='5/1/88'
      pend='7/1/88' astart='1/15/88' aend='3/15/88' astart='5/15/88'
      resp='J22'.Planning
:annotate date='1/1/88'.Software
:annotate date='5/1/88'.Hardware
:schi head.Design
:schi pstart='6/1/88' pend='12/1/88' resp='G51'
      astart='5/30/88'.Hardware
:schi pstart='8/1/88' pend='2/1/89' resp='G47'.Software
:schi head.Test
:schi pstart='11/1/88' pend='11/1/89' resp='T47'.Hardware
:schi pstart='8/1/89' pend='8/1/89' resp='T48'.Software
:schi head.
:schi pstart='1/1/88' pend='9/1/88'
      pstart='10/1/88' pend='6/1/89'
      pstart='7/1/89' pend='3/1/90'.
:annotate date='1/1/88'.Planning
:annotate date='10/1/88'.Microcode
:annotate date='7/1/89'.Test
:eschedule.
:schdate.
```

## Computer X39 and Operating System

Activity		1988				1989				1990-			
		1st Qtr	2nd Qtr	3rd Qtr	4th Qtr	1st Qtr	2nd Qtr	3rd Qtr	4th Qtr	1st Qtr	2nd Qtr	3rd Qtr	4th Qtr
Planning	J22												
		Software	Hardware										
<b>Design</b>													
Hardware	G51												
Software	G47												
<b>Test</b>													
Hardware	T47												
Software	T48												
		Planning				Microcode				Test			



Activity		Projected Start	Projected End	Actual Start	Actual End
Planning	J22	1/1/88	3/1/88	1/15/88	3/15/88
		5/1/88	7/1/88	5/15/88	
<b>Design</b>					
Hardware	G51	6/1/88	12/1/88	5/30/88	
Software	G47	8/1/88	2/1/89		
<b>Test</b>					
Hardware	T47	11/1/88	11/1/89		
Software	T48	8/1/89	8/1/89		
		1/1/88	9/1/88		
		10/1/88	6/1/89		
		7/1/89	3/1/90		

### Schedule with STITLE and DATETYPE

This example shows the use of the STITLE and DATETYPE attributes on the :SCHEDULE tag with EUROPEAN dates.

```
:schedule actsize='2i' start='3/88' end='8/88' date='27/4/89'  
    datetype=european stitle='Road Repair' today  
    .Road Repair for Downtown Area  
:schi pstart='30/3/88' pend='30/4/88'.Repair Main Street  
:schi pstart='30/3/88' pend='30/3/88'.Place Detour Signs  
:schi astart='1/4/88' aend='3/4/88'.Fill Pot Holes  
:schi astart='2/4/88' aend='2/4/88'.Actual Sign Placement  
:schi pstart='30/6/88' pend='30/7/88'  
    astart='2/7/88' aend='2/8/88'.Repair 3rd Avenue  
:schi pstart='19/6/88' pend='16/7/88'  
    astart='24/6/88' aend='24/7/88'.Repair 2nd Street  
:schi pstart='10/7/88' pend='10/7/88'  
    astart='25/7/88' aend='25/7/88'.Replace Water Main  
:schi pstart='11/7/88' pend='11/7/88'  
    astart='21/8/88' aend='26/8/88'.Restore Full Service  
:eschedule.  
:schdate.
```

## Road Repair for Downtown Area

Activity	1988					
	March	April	May	June	July	Aug
Repair Main Street						
Place Detour Signs		◊				
Fill Pot Holes		■				
Actual Sign Placement		◆				
Repair 3rd Avenue						
Repair 2nd Street						
Replace Water Main					◊	◆
Restore Full Service					◊	■

↑

Activity	Projected Start	Projected End	Actual Start	Actual End
Repair Main Street	30/3/88	30/4/88		
Place Detour Signs	30/3/88	30/3/88		
Fill Pot Holes			1/4/88	3/4/88
Actual Sign Placement			2/4/88	2/4/88
Repair 3rd Avenue	30/6/88	30/7/88	2/7/88	2/8/88
Repair 2nd Street	19/6/88	16/7/88	24/6/88	24/7/88
Replace Water Main	10/7/88	10/7/88	25/7/88	25/7/88
Restore Full Service	11/7/88	11/7/88	21/8/88	26/8/88

Road Repair

1

### Schedule with Staffing Information

An important part of some schedules is the staffing information for the project—the personnel required for each stage of a project. The HEAD and NODBOX attributes on the :SCHI tag, and the ALIGN attribute on the :ANNOTATE tag provide a way to include staffing information in a schedule.

The HEAD attribute is used on the :SCHI tag. It contains the activity description “Staffing” to set this text off from the rest of the schedule. The NODBOX attribute on the :SCHI tag is used for the :SCHI tags that produce the staffing information, so that this information does not appear in the datelist box created by the :SCHDATE tag. A date of the 15th of each month, plus the ALIGN = CENTER attribute on the :ANNOTATE tag, cause the staffing information to be centered within each month.

```
:schedule start='7/88' end='6/89' date='1/20/89' today.Testing Project
:schi pstart='7/1/88' pend='9/1/88'
    astart='7/1/88' aend='9/15/88' resp='Software Education'.Education
:schi pstart='9/1/88' pend='10/1/88'
    pstart='10/15/88' pend='10/15/88'
    astart='9/15/88' aend='10/5/88'
    astart='10/20/88' aend='10/20/88' resp='Software Test'.Test Plan
:annotate date='9/1/88'.create
:annotate date='10/15/88'.approval
:schi pstart='10/1/88' pend='1/1/89'
    astart='10/1/88' resp='Programming Test'.Create Testcases
:schi pstart='1/1/89' pend='2/15/89'
    astart='12/15/88' resp='Programming Development'.Debug Testcases
:schi pstart='2/15/89' pend='5/15/89' resp='Programming Test'.Run Testcases
:schi pstart='3/15/89' pend='4/15/89' resp='Programming Test'.Claims Testing
:schi pstart='6/1/89' pend='6/1/89'
    resp='Programming Development'.Final Assessment
:schi head nodbox.Staffing
:schi nodbox.Projected
:annotate align=center date='7/15/88'.3
:annotate align=center date='8/15/88'.3
:annotate align=center date='9/15/88'.3
:annotate align=center date='10/15/88'.4
:annotate align=center date='11/15/88'.4
:annotate align=center date='12/15/88'.5
:annotate align=center date='1/15/89'.6
:annotate align=center date='2/15/89'.6
:annotate align=center date='3/15/89'.5
:annotate align=center date='4/15/89'.5
:annotate align=center date='5/15/89'.4
:annotate align=center date='6/15/89'.2
:schi nodbox.Actual
:annotate align=center date='7/15/88'.2
:annotate align=center date='8/15/88'.3
:annotate align=center date='9/15/88'.4
:annotate align=center date='10/15/88'.4
:annotate align=center date='11/15/88'.4
:annotate align=center date='12/15/88'.5
:annotate align=center date='1/15/89'.5
:eschedule.
:schdate.
```

## Testing Project

Activity	1988						1989					
	July	Aug	Sept	Oct	Nov	Dec	Jan	Feb	March	April	May	June
Education <i>Software Education</i>												
Test Plan <i>Software Test</i>			create	approval								
Create Testcases <i>Programming Test</i>												
Debug Testcases <i>Programming Development</i>												
Run Testcases <i>Programming Test</i>												
Claims Testing <i>Programming Test</i>												
Final Assessment <i>Programming Development</i>												
<b>Staffing</b>												
Projected	3	3	3	4	4	5	6	6	5	5	4	2
Actual	2	3	4	4	4	5	5					

↑

Activity	Projected Start	Projected End	Actual Start	Actual End
Education <i>Software Education</i>	7/1/88	9/1/88	7/1/88	9/15/88
Test Plan <i>Software Test</i>	9/1/88	10/1/88	9/15/88	10/5/88
	10/15/88	10/15/88	10/20/88	10/20/88
Create Testcases <i>Programming Test</i>	10/1/88	1/1/89	10/1/88	
Debug Testcases <i>Programming Development</i>	1/1/89	2/15/89	12/15/88	
Run Testcases <i>Programming Test</i>	2/15/89	5/15/89		
Claims Testing <i>Programming Test</i>	3/15/89	4/15/89		
Final Assessment <i>Programming Development</i>	6/1/89	6/1/89		

Testing Project

1

---

### Things to Remember

There are a few things that you should remember when using the GML schedule tags:

- Blank lines in your text file are ignored and will not leave space on your formatted and printed page.
- The following is a list of formatting commands that are ignored when you use the memo tags:
  - SYSVAR X is ignored; a cross-reference listing will not be created.
  - SYSVAR S is ignored; a one column format will be used.
- GML starter set tags can't be used within the :SCHEDULE and :ESCHEDULE tags.
- Use of this profile is not supported on line printers.



---

## Appendix A. Messages

All messages are listed in order by message number **nnn**.

Several messages include return codes that indicate the reason for the error. These return codes and their meanings can be found in the appropriate systems manuals as listed in Appendix B, "Related Publications and Products" on page 53.

---

## Generalized Markup Language (GML) Starter Set Messages

**002**      *tag name* **FOUND OUTSIDE** *element*

**Explanation:** The given tag was found outside the element within which it is valid.

**System Action:** The tag is ignored.

**User Response:** Change the markup of the document to include the tag within the given document element.

**003**      *element* **PREMATURELY ENDED BY** *tag name*

**Explanation:** The given tag is not allowed within the given element. An element end tag was probably omitted.

**System Action:** The element is ended before the tag is processed.

**User Response:** Insert an element end tag at the appropriate place in the document.

**004**      *tag name* **FOUND WITHIN** *element* **AND IGNORED**

**Explanation:** The given tag is invalid within the given element.

**System Action:** The tag is ignored.

**User Response:** Remove the tag from the element.

**005**      **EXTRANEOUS [GLOSSARY | DEFINITION] TERM** '*term*'  
**IGNORED**

**Explanation:** Two successive definition or glossary term tags have been encountered. A definition or glossary description tag was probably omitted.

**System Action:** The given definition or glossary term is ignored.

**User Response:** Add the missing definition or glossary description or remove the extraneous definition or glossary term.

**006**      **[GLOSSARY | DEFINITION] TERM TAG MISSING**

**Explanation:** Two successive definition or glossary description tags have been encountered. A definition or glossary term tag was probably omitted.

**System Action:** A question mark (?) is used in place of the missing definition or glossary term.

**User Response:** Add the missing definition or glossary term or remove the extraneous definition or glossary description.

**007**      **UNRECOGNIZED LIST TYPE:** *list type*

**Explanation:** An invalid list item identifier number has been specified in the profile. The number must be 1-4 for an unordered list on a line device, 1-5 for an unordered list on a page device, and 1-8 for an ordered list on any device.

**System Action:** A simple list is used in place of the unrecognized list.

**User Response:** Ensure that the list item identifier number is one of the correct numbers for your output device.

**008**      *element ID 'identifier'* **TRUNCATED TO SEVEN CHARACTERS**

**Explanation:** The given element identifier exceeds the maximum allowed length.

**System Action:** The identifier is truncated at 7 characters.

**User Response:** Change the markup to use identifiers that are no more than 7 characters long.

**009**      **DUPLICATE** *element ID 'identifier'* **IGNORED**

**Explanation:** Two elements with the same identifier have been encountered.

**System Action:** The duplicate identifier is ignored.

**User Response:** Change the markup to specify a unique identifier for the element.

**011**      *tag name* **END-TAG FOUND OUTSIDE** *element* **AND IGNORED**

**Explanation:** The given end tag was encountered outside the given element. An element tag was probably omitted.

**System Action:** The tag is ignored.

**User Response:** Change the markup to include the element end tag within the proper element.

**014**      **MISSING TERM FOR LEVEL** *level* **INDEX ENTRY**

**Explanation:** One of the terms given for an index entry of the given level was null, either because no term was given on this or a higher level index tag or because an index level was skipped.

**System Action:** The tag is ignored.

**User Response:** Change the markup to include the missing index term or missing index level tag.

**015**      *value* **NOT A VALID VALUE FOR** *attribute name* **ATTRIBUTE OF**  
*tag name* **TAG**

**Explanation:** A value encountered on an attribute was not valid for that particular attribute.

**System Action:** The value is ignored, and the default value is used. If there is no default value and the severity code is SEVERE, processing stops. See the description of the tag to determine what the default attribute value is.

**User Response:** Ensure that the attribute was spelled correctly. Correct the attribute value specified.

**016**      *identifier* **NOT A VALID IDENTIFIER FOR** *tag name* **TAG**

**Explanation:** The identifier specified on the ID attribute of the tag contains invalid characters.

**System Action:** The attribute is ignored.

**User Response:** Specify an identifier that contains only valid characters. The valid characters are the alphanumeric characters and the characters @#\$.

**017**      *definition type* **DEFINITION identifier DOES NOT EXIST**

**Explanation:** The REFID identifier specified on a :TABLE, :ROW, :TFT, :THD, or :BARCODE tag refers to a definition that has not been defined.

**System Action:** The tag is ignored.

**User Response:** Ensure that the REFID identifier is spelled correctly. Delimit the definition with the :RDEF or :BARDEF tag prior to referring to the identifier.

**019**      **REQUIRED ATTRIBUTE *attribute name* MISSING FROM tag name TAG**

**Explanation:** A required attribute is missing from a tag.

**System Action:** The tag is ignored, or if the severity code is SEVERE, processing stops.

**User Response:** Specify the attribute on the tag.

**020**      **CELL WIDTH FOR CELL *cell number* TOO SMALL, ZERO INDENTS USED**

**Explanation:** The width of a cell in a table is so small that the standard left and right indents in the cell cannot be performed.

**System Action:** The left and right indents for the cell are set to zero.

**User Response:** Define the cell to have a larger cell width with the CWIDTHS attribute on the :RDEF tag.

**021**      **INVALID ARRANGE SPECIFICATION, DEFAULT OF 1 USED**

**Explanation:** The ARRANGE values specified on the :RDEF tag did not specify a valid arrangement. A different number of values were specified on different lines of the arrangement.

**System Action:** The default arrangement of one cell is used.

**User Response:** Correct the values on the ARRANGE attribute so that the same number of values are specified between slashes, or on different occurrences of the ARRANGE attribute, if multiple occurrences are used.

**022**      **MINDEPTH REQUIRED, BUT MISSING FOR CELL NUMBER *cell number***

**Explanation:** For cells that are rotated 90, -270, 270, or -90 degrees in a table, the MINDEPTH attribute is required, but was not specified.

**System Action:** The rotation of the cell is changed to 0.

**User Response:** Specify a MINDEPTH value for the particular cell, or change the rotation value of the cell to 0, 180, or -180.

**023**      **CANNOT START A NEW ROW IN A HEADER OR FOOTER, C TAG IGNORED**

**Explanation:** An attempt was made to start a new row in a table header or footer (as defined with the :THD or :TFT tag), by specifying a :C tag without a cell number, and no more cells were available in the table header or footer.

**System Action:** The :C tag is ignored.

**User Response:** When specifying the :C tag without a cell number in a table header (:THD) and table footer (:TFT), ensure that there is another cell within the table header or footer that has a higher cell number than the last cell.

**024      *tag name* TAG NOT VALID AFTER FIRST ROW IN TABLE HAS STARTED**

**Explanation:** The **tag-name** tag is valid only if it is encountered after the :TABLE tag and prior to the first :ROW tag in the table.

**System Action:** The tag is ignored.

**User Response:** Relocate the tag so that it falls between the :TABLE and the first :ROW tag.

**025      MSI CHECK SUM RESULTS IN AN ERROR VALUE FOR *bcddata***

**Explanation:** The check digit value was 10, which results in an error when the specified type of check digit is IR, INE, NR or NNE.

**System Action:** The :BARCODE tag is ignored. No bar code is generated.

**User Response:** Change either the input data or the type of check digit so that the check digit value is not in error.

**026      DATA *bcddata* CONTAINS INVALID CHARACTERS FOR GENERATION OF *bctype***

**Explanation:** Invalid data was given on the :BARCODE tag (includes invalid data format for UPCE).

**System Action:** The :BARCODE tag is ignored. No bar code is generated.

**User Response:** Remove the invalid data characters. CODABAR start and stop characters may need to be added or changed. Ensure that the UPCE data conforms to one of the four valid formats.

**027      INVALID NUMBER OF CHARACTERS IN *bcddata* FOR GENERATION OF *bctype***

**Explanation:** The length of data given on the :BARCODE tag for CODABAR, EAN8, EAN13, MSI, UPCA, or UPCE is invalid.

**System Action:** The :BARCODE tag is ignored. No bar code is generated.

**User Response:** For CODABAR, ensure that the length of the input data is a minimum of 3 characters. For EAN8, ensure that the length is 7 characters. For EAN13, ensure that the length is 12, 14, or 17 characters. For MSI, ensure that there are no more than 15 characters. For UPCA or UPCE, ensure that the length is 11, 13, or 16 characters.

**028      WIDE TO NARROW RATIO IS LESS THAN 2:1 FOR ID *id***

**Explanation:** The wide-to-narrow ratio is less than 2:1.

**System Action:** The :BARDEF tag is ignored.

**User Response:** Change one or both attribute values so that the WIDE attribute value is at least twice as large as the NARROW attribute value.

**029**      *tag name* **TAG IS IGNORED FOR OUTPUT DEVICE:** *pdev*

**Explanation:** The :BARDEF and :BARCODE tags are not valid for physical device *pdev*.

**System Action:** The :BARDEF and :BARCODE tags and their data are ignored.

**User Response:** Invoke SCRIPT/VS with a valid device type.

**030**      *value* **SPECIFIED FOR THE** *attribute* **ATTRIBUTE EXCEEDS THE**  
*measurement* **LENGTH**

**Explanation:** Invalid sizes were specified for WIDE or HEIGHT.

**System Action:** The :BARDEF tag is ignored.

**User Response:** Specify smaller values for WIDE or HEIGHT.

**031**      **ATTRIBUTE** *attribute* **IGNORED FOR** *bctype:* **BAR CODE ID** *id*

**Explanation:** The :BARDEF tag specifies an attribute that is ignored for the particular type of bar code specified.

**System Action:** Attribute is ignored.

**User Response:** Remove the ignored attributes from the :BARDEF tag.

**032**      **GENERATED BAR CODE EXCEEDS THE** *measurement* **LENGTH**

**Explanation:** Generated bar code exceeds the printable area remaining on the page.

**System Action:** The barcode is generated but may print off the page or overlay the surrounding text.

**User Response:** Either specify smaller values for NARROW, WIDE or HEIGHT, or change the position of the bar code on the page.

**033**      **ATTRIBUTE** *attribute* **ON** *tag name* **TAG INVALID WITHIN**  
*element:* *tag name* **IGNORED**

**Explanation:** The given tag with this attribute is invalid within the given element.

**System Action:** The tag is ignored.

**User Response:** Remove the tag from the element.

**034**      **DATE FOR** *attribute name* **ATTRIBUTE MUST BE EARLIER THAN**  
**DATE FOR** *attribute name* **ATTRIBUTE**

**Explanation:** The given attribute must have a date that comes before the date on the second attribute.

**System Action:** The attribute is ignored, or if the severity code is SEVERE, processing stops.

**User Response:** Change the date on the first attribute so that it comes before the date on the second attribute.

**035**      *attribute name* **ATTRIBUTE MISSING CORRESPONDING** *attribute end-name* **ATTRIBUTE**

**Explanation:** The given attribute must have its corresponding end attribute.

**System Action:** The attribute is ignored.

**User Response:** Specify the corresponding end attribute or delete the given attribute.

**036**      **SCHEDULE EXCEEDS PAGE WIDTH FOR THIS DEVICE BY**  
*estimated value*

**Explanation:** The schedule is too large to fit on the page when formatted for the given device.

**System Action:** Processing stops.

**User Response:** Shorten the schedule by the estimated value and reformat, or reformat the schedule for a 90° or 270° rotated device.

**037**      *attribute name* **ATTRIBUTE DEFAULT NOT AVAILABLE**

**Explanation:** The default value for the attribute is not available. It is not specified.

**System Action:** The attribute is ignored.

**User Response:** Add the attribute and a value so the default does not need to be used, or include the default.

**038**      *tag name* **TAG FOUND OUT OF SEQUENCE**

**Explanation:** The given tag was not found in the right location. The sequence of tags in the document must come in a specific order to format correctly.

**System Action:** The tag is ignored.

**User Response:** Place the given tag in the correct sequence in the document.

**039**      **DEVICE** *physical device* **NOT SUPPORTED BY** *application name*  
**APPLICATION**

**Explanation:** This application will not format for particular device types.

**System Action:** Processing stops.

**User Response:** Format the application for a valid device type.

**040**      *attribute-name* **ATTRIBUTE** *value* **OVERLAPS PREVIOUS**  
*attribute-name* **ATTRIBUTE** *value*

**System Action:** The tag is ignored.

**Explanation:** A value encountered on an attribute overlaps previous values for that particular tag.

**User Response:** Ensure that the values on the attributes for that particular tag do not overlap each other and are in ascending order. (See description of the tag to determine valid attribute values.)

**041**      *attribute-name* **ATTRIBUTE DATE MUST FALL WITHIN** *element*  
**DATE RANGE**

**System Action:** The value is ignored and the default value is used. See the description of the tag to determine what the default attribute value is.

**Explanation:** The given attribute must have a date that falls within the range of dates of the element.

**User Response:** Change the date on the attribute so that it falls within the date range of the element. (See description of the tag to determine valid attribute values.)



---

## **Appendix B. Related Publications and Products**

## Publication Library Guide for the Document Composition Facility

The following table lists the Document Composition Facility publications by number as they relate to user tasks. “DCF Publications” on page 55 lists the titles and the order numbers that correspond to the numbers listed in the table.

Number	User Tasks	Typical Audience	Brief Description
(1) (2) (3) (20)	Planning and introducing DCF/DLF	Users, system planners	Provide a general overview of text processing, library facility, and available books.
(3) (4) (5) (12) (16) (20)	Formatting documents (using the GML starter set)	Novice to experienced end users	Provide an introduction to the Generalized Markup Language (GML) starter set and describes the GML starter set tags and SCRIPT/VS messages.
(6)	Creating bar codes with DCF/GML	Experienced end users	Provides information about using GML to create bar codes.
(9) (10) (11) (12) (17) (18) (19) (22)	Formatting documents (using SCRIPT/VS control words)	Knowledgeable to experienced end users	Describe the function and use of all SCRIPT/VS control words, macros, diagnostic aids, and the formatting features and messages.
(14) (15)	Converting RFTDCA for SCRIPT/VS formatting	Novice to experienced RFTDCA users	Describe the function and use of the optional Office Document Feature, including diagnostic aids and messages.
(4) (5) (7) (9) (10) (11) (19) (22)	Modifying GML starter set <sup>2</sup>	Document administrator and text programmer <sup>3</sup>	Contain material about GML starter set tags, SCRIPT/VS control words, and GML starter set modifications.
(4) (5) (7) (8) (9) (10) (11) (16) (19) (22)	Creating GML application processing functions	Document administrator and text programmer <sup>3</sup>	Provide information about designing your own GML and about GML concepts, GML starter set tags, SCRIPT/VS control words, and usage guidelines.
(10) (11) (12) (13) (19) (21) (22)	Installing, modifying, and maintaining DCF	System programmer	Give information on error isolation, program tailoring, and use of SCRIPT/VS.
(23)	Creating mathematical formulas with SMFF	Experienced end users	Describes the function and use of the SCRIPT Mathematical Formula Formatter (SMFF), including .EQ control word and messages.
(24)	Creating memos, transparencies, and schedules with GML applications	Novice to experienced end users	Describes the use of the memo, transparencies, and schedule applications, including messages.

<sup>2</sup> Central Programming Service support and maintenance are provided *only* on the unmodified GML applications shipped with DCF. If you modify any of these GML applications shipped with DCF, it is recommended that you also maintain an *unmodified* copy for diagnostic purposes.

<sup>3</sup> The document administrator is responsible for defining markup conventions and procedures for an organization. The text programmer implements application processing functions (APFs) that provide the processing specified by the document administrator.

## DCF Publications

Number	Titles and Order Numbers
(1)	<i>Document Composition Facility and Document Library Facility Executive Overview and Product Summary</i> , GX20-2332.
(2)	<i>Document Composition Facility and Document Library Facility General Information</i> , GH20-9158.
(3)	<i>Document Composition Facility: Introduction to Generalized Markup Language</i> , G544-3192.
(4)	<i>Document Composition Facility: Generalized Markup Language Starter Set User's Guide</i> , SH20-9186.
(5)	<i>Document Composition Facility: Generalized Markup Language Starter Set Reference</i> , SH20-9187.
(6)	<i>Document Composition Facility: Bar Code User's Guide</i> , S544-3115.
(7)	<i>Document Composition Facility: Generalized Markup Language Starter Set Implementation Guide</i> , SH35-0050.
(8)	<i>Document Composition Facility: Generalized Markup Language Starter Set Concepts and Design Guide</i> , SH20-9188.
(9)	<i>Document Composition Facility: SCRIPT/VS User's Guide</i> , S544-3191.
(10)	<i>Document Composition Facility: SCRIPT/VS Text Programmer's Guide</i> , SH35-0069.
(11)	<i>Document Composition Facility: SCRIPT/VS Language Reference</i> , SH35-0070.
(12)	<i>Document Composition Facility SCRIPT/VS Messages</i> , SH35-0048.
(13)	<i>Document Composition Facility: Diagnosis Guide and Reference</i> , LV32-0523.
(14)	<i>Document Composition Facility: Office Document Feature User's Guide</i> , G544-3129.
(15)	<i>Document Composition Facility: Office Document Feature Reference</i> , S544-3130.
(16)	<i>Using the Document Composition Facility</i> , SR21-0515 (Training Course 32291).
(17)	<i>Using DCF with the 4250 Printer</i> , SR20-8486 (Training Course 32908).
(18)	<i>Using DCF with Page Printers</i> , SR21-1211 (Training Course 32243).
(19)	<i>Document Composition Facility—Release 3 (SCRIPT/VS) for Document Administrators and Text Programmers</i> , SR20-7525 (Training Course).
(20)	<i>Document Composition Facility (SCRIPT/VS) Student Text</i> , SC20-1894 (Training Course).
(21)	DCF Program Directory.
(22)	<i>Document Composition Facility: TSO Enhancements Update Guide</i> , G544-3345.
(23)	<i>Document Composition Facility: SCRIPT Mathematical Formula Formatter User's Guide</i> , S544-3306.
(24)	<i>Document Composition Facility: Generalized Markup Language (GML) Applications User's Guide</i> , S544-3305.

**Note:** The following reference booklets and a three-ring binder are also available:

*Document Composition Facility: GML Starter Set Quick Reference*, SX26-3723.  
*Document Composition Facility: SCRIPT/VS Text Programmer's Quick Reference*, SX26-3719.  
*Document Composition Facility Post-Processor Examples*, S544-3484.  
*Document Composition Facility SCRIPT/VS*, SH35-0086 (binder).

---

## Related Products

The following products are related to GML:

- **MARKUP:** A PC-based editor that helps you create GML documents. Refer to *MARKUP User's Guide and Tutorial*, which you can order with the MARKUP product, 6476161. For information about ordering MARKUP, contact your local IBM Branch Office.
- **Publishing Systems BookMaster:** A host-based application that runs under control of DCF and is designed for high-volume in-house publishing applications. This application contains a superset of the GML Starter Set. For information about ordering BookMaster (Program Number 5688-015), contact your local IBM Branch Office.

## Glossary

The glossary defines words and phrases that have special meanings in SCRIPT/VS or special meanings in a typographical sense. The terms are defined as they are used in this book. If you do not find the term you are looking for, refer to the index or to the *IBM Vocabulary for Data Processing, Telecommunications, and Office Systems*, GC20-1699.

This glossary includes definitions developed by the American National Standards Institute (ANSI) and the International Organization for Standardization (ISO). This material is reproduced from the American National Dictionary for Information Processing, copyright 1977 by the Computer and Business Equipment Manufacturers Association, copies of which can be purchased from the American National Standards Institute, 1430 Broadway, New York, New York 10018.

### A

**advanced function printing (AFP).** The ability of licensed programs to use the all-points-addressable concept to print text and images on a printer.

**alignment.** The horizontal placement of text in a column or cell.

**all-points addressability.** The capability to address, reference, and position text, overlays, and images at any defined point on the printable area of a sheet. See *page device* and contrast with *line device*.

**alphameric string.** A sequence of characters consisting solely of the letters a through z and the numerals 0 through 9.

**ampersand.** The & character. When an ampersand begins a character string, SCRIPT/VS assumes the character string is a symbol name. If the symbol name is defined, SCRIPT/VS replaces the symbol with its value (unless symbol substitution is off). In running footings, running headings, and running titles, the ampersand is usually the page number symbol. When encountered by itself on the right side of a .SE [Set Symbol] control word, it is interpreted as the page number symbol.

**application processing function (APF).** In GML processing, the processing that is performed when a document element or attribute is recognized. In SCRIPT/VS, an APF is implemented as a sequence of control words, possibly intermixed with text and symbols, in one of three forms: macro definition, value of a symbol, or imbedded file.

**application programming interface (API).** An external, published interface that can be programmed to by another application.

**ascender.** (1) In a font, the distance from the baseline to the top of the character. See *maximum ascender*. (2) The part of a lowercase letter that rises above the body of the letter. Letters with ascenders are b, d, f, h, k, l, and t.

**ATMS-II.** Advanced Text Management System.

**attribute.** A characteristic of a document (or document element) other than its type or content. For example, the security level of a document or the depth of a figure.

**attribute label.** In GML markup, a name of an attribute that is entered in the source document when specifying the attribute's value.

### B

**back matter.** In a book, those sections (such as glossary and index) that are placed after the main chapters or sections.

**balancing.** In multicolumn formatting, the process of making column depths on a page approximately equal by re-distributing the text in the columns. See also *vertical justification*.

**baseline.** An imaginary horizontal line that most of the letters in a line of text appear to rest on.

**basic document element.** In a general document, one of a group of elements that occurs frequently; for example: note, paragraph, and definition list.

**batch environment.** The environment in which non-interactive programs are executed.

**binding edge.** The edge of a page to be bound, stapled, or drilled. Defined with the BIND option of the SCRIPT command.

**body.** (1) Of a printed page, that portion between the top and bottom margins that contains the text. (2) Of a book, that portion between the front matter and the back matter.

**boldface.** A heavy-faced type. Also, printing in this type.

**bottom margin.** On a page, the space between the body or the running footing, if any, and the bottom edge of the page.

**break.** An interruption in the formatting of input lines so that the next input line is printed on a new output line.

**bug.** An error in a program or in a document markup.

## C

**call.** Used in reference to macros. It means to invoke the macro.

**caps.** Capital letters. See also *initial caps*.

**caption.** Text accompanying and describing an illustration.

**case-sensitive.** Whether a group of letters is uppercase or lowercase has relevance. ABC is different from Abc which is different from ABc.

**CDPF.** Composed Document Printing Facility.

**cell.** A single unit within a table in which text or other expressions may appear. A cell is always rectangular and usually bounded by horizontal and vertical rules.

**centimeter (cm).** A measurement equal to 0.39 inch. 100 cm = 1 meter (m).

**chapter.** In a general document, a name given to a first-level heading segment that occurs within the body of a document. See also *heading segment*.

**character.** A symbol used in printing. For example, a letter of the alphabet, a numeral, a punctuation mark, or any other symbol that represents information.

**character arrangement table.** Translates input data into printable characters and identifies associated character sets and graphic character modification modules.

**character set.** A finite set of different characters that is agreed to be complete for some purpose. For example, in printing, the characters that constitute a font.

**character space.** The horizontal size of a character. This size depends upon which font the character is from and on which physical device the character is printed.

**character spacing.** The space between characters in a word.

**cicero.** In the Didot point system, a unit of 0.1776 inch (4.512 millimeters) used in measuring typographical material.

**CMS.** Conversational Monitor System—an interactive processor that operates within VM/370.

**code page.** A font library member name that gives the association between code points and the character names of a font.

**code point.** An eight-bit binary code representing one of 256 possible characters.

**coded font.** (1) The combination of a code page and a font library. (2) A font that is fully described in terms of typeface, point-size, weight, width, and attribute.

**column.** A vertical arrangement of characters or other expressions on a printed page.

**column balancing.** The process of redistributing lines of text among a set of columns so that the amount of text in each column is as equal as possible.

**column width.** The width of each text column on a page. Specified with the .CL [Column Line Length] control word. (In multicolumn formatting, all columns on a page usually have the same width.)

**command.** A request from a terminal or a request specified in a batch processing job for the performance of an operation or the execution of a particular program. For example, a request given at a terminal for SCRIPT/VS to format a document or for an editor to edit a line of text.

**comment.** A control word line that is ignored by SCRIPT/VS. Such lines begin with either .\* or .cm.

**composed text.** Text that has been formatted and that contains control information to direct the presentation of the text on page printers.

**composite.** The act or result of formatting a document.

**composite rotation.** The total amount of rotation done by the printer to place text in the correct orientation on the page.

To determine the composite rotation, add all the current rotations, such as the rotation of the page as specified with the DEVICE command option, the rotation of the current area (as specified on the .DA [Define Area] control word), the rotation of the current table (as specified in the TABLE parameter of the .TD [Table Definition] control word), and the rotation of the contents of the current cell (as specified in the CELL parameter of the .TD [Table Definition] control word). For example: if the page rotation is 90 and there is no current area, and the table rotation is 180, and the rotation of the contents of the cell is 180; then the composite rotation of the table would be 270 and the composite rotation of the contents of the cell would be 90.

**compositor.** A person or program that composes text.

**concatenation.** The forming of an output line that contains as many words as the column width allows, by placing the first words from an input line after the last words from the preceding input line. When words from an input line would reach beyond the right margin and

**hyphenation** cannot be performed, they are placed at the beginning of the next output line, and so on.

**control word.** An instruction within a document that identifies its parts or tells SCRIPT/VS how to format the document. See also *macro*.

**control word line.** An input line that contains at least one control word.

**control word statement.** A control word and its parameters.

**copy group.** A portion of a FORMDEF that defines a set of modifications that can be used when printing a page.

**current left margin.** The left limit of a column that is in effect for formatting. Each column's left margin is specified with the .CD [Column Definition] control word. However, the current left margin (that is, the left boundary for an output line) might vary to the right of the column's left margin when indentation is changed with the .IN [Indent], .UN [Undent], .IL [Indent Line], and .OF [Offset] control words.

**current line.** The line in a source document at which a computer program (such as an editor or a formatter) is positioned for processing.

## D

**debug.** To detect, trace, and eliminate errors in computer programs and SCRIPT/VS documents.

**default value.** A value assumed by a computer program when a control word, command, or control statement with no parameters is processed. In GML processing, the value assumed for an attribute when none is specified.

**descender.** (1) In a font, the distance from the baseline to the bottom of the character. See *maximum descender*. (2) The part of a letter that falls below the body of the letter. Letters with descenders are g, j, p, q, y, and Q.

**destination.** The physical device to which data is sent.

**dictionary.** A collection of *word stems* that is used with the spelling verification and automatic hyphenation functions.

**Didot point system.** A standard printer's measurement system on which type sizes are based. A Didot point is 0.0148 inch (0.376 millimeter). There are 12 Didot points to a cicero. See also *cicero* and *point*.

**document.** (1) A publication or other written material. (2) A machine-readable collection of lines of text or images, usually called a source document. See also *output document* and *source document*.

**document administrator.** One who is responsible for defining markup conventions and procedures for an organization.

**document conversion processor.** A computer program that processes a machine-readable document that includes formatting controls written in one formatter language, to produce a machine-readable document that includes formatting controls appropriate for another formatter language.

**document library.** A set of VSAM data sets, accessible in a batch environment, which contain documents and related files.

**dot leader.** A set of periods that fills in the space between two pieces of split text such as a chapter title and its page number in a table of contents.

**duplex.** A mode of formatting appropriate for printing on both sides of a sheet.

## E

**EBCDIC.** Extended binary-coded decimal interchange code. A coded character set consisting of 8-bit coded characters.

**edit.** To create or modify the contents of a document or file. For example, to insert, delete, change, rearrange, or copy lines.

**editor.** A computer program that processes commands to enter lines into a document or to modify it.

**eject.** In formatting, a skip to the next column or page.

**element.** Any part of a document: a single character or a word or a sentence. Also refers to any part of a document you can identify with a GML tag (tagged element), such as a paragraph or figure or heading.

**em.** A unit of measure usually equal to the width or the height of the character "m" in a particular font.

**en.** A unit of measure usually equal to one-half the width of an em. For many typefaces, lowercase characters tend to average the width of an en.

**enable.** Used in reference to a tag. Means that the tag is mapped to its appropriate APF.

**epifile.** The second portion of a profile (after a .EF control word) that is processed *after* a main document has been processed.

**escapement.** The space unit of movement (either vertical or horizontal) that is built into a physical device. For the 1403, with a 10-pitch train, the horizontal escapement unit is 1/10th of an inch; for the 4250, that value is 1/600th of an inch; and for the 3800 Model 3

and the 3820 Page Printer, that value is 1/240th of an inch.

**exposure.** The amount of risk associated with a schedule item or items.

**extended symbol processing.** The processing of a symbol whose value causes the remainder of the line to be stacked and later processed as a new input line.

## F

**factor.** A dimensionless scalar value used to form a product with another value. Factors can also be expressed as percentages.

**figure space.** (1) The width of the figure zero (0) is commonly used as the figure space of a given typeface. This is the definition figure space as it is used in DCF. (2) A unit of measure equal to the width of the “en” space in a particular font.

**fill character.** The character that is used to fill up a space; for example, blanks used to fill up the space left by tabbing.

**float.** (1) (noun) A keep (group of input lines kept together) whose location in the source file can vary from its location in the printed document. (2) (verb) Of a keep, to be formatted in a location different from its location in the source file.

**flush.** Having no indentation.

**fold.** (1) To translate the lowercase characters of a character string into uppercase. (2) To place that portion of a line that does not fit within a column on the next output line.

**folio.** Page number.

**font.** (1) An assortment of type, all of one size and style. (2) A font library member that contains characters that must be used in conjunction with a code page font library member.

**font object.** Refers to a member of a font library. In CMS, a font object is a file whose filetype matches the name of the font library. In MVS, a font object can be a member of a partitioned data set (PDS).

**font set.** The set of fonts to be used in formatting a source document.

**footing.** Words located at the bottom of the text area. See also *running footing*.

**footnote.** A note of reference, explanation, or comment, placed below the text of a column or page, but within the body of the page (above the running footing).

**foreground.** The environment in which interactive programs are executed. Interactive processors reside in the foreground.

**format.** (1) (noun) The shape, size, and general makeup of a printed document. (2) (verb) To prepare a document for printing in a specified format.

**formatter.** (1) A computer program that prepares a source document to be printed. (2) That part of SCRIPT/VS that formats input lines for a particular logical device type.

**formatting mode.** In document formatting, the state in which input lines are concatenated and the resulting output lines are justified.

**form definition.** A resource object that defines the characteristics of the form which include: overlays to be used (if any), text suppression, the position of page data on the form, and the number and modifications of the page. Synonymous with FORMDEF.

**FORMDEF.** Synonym for form definition.

**front matter.** In a book, those sections (such as preface, abstract, table of contents, list of illustrations) that are placed before the main chapters or sections.

## G

**general document.** A type of document whose description can apply to a variety of documents, from memoranda to technical manuals. It can be used as a catch-all category for documents that do not conform to any other type description.

**Generalized Markup Language (GML).** A language that can be used to identify the parts of a source document without respect to particular processing.

**GML delimiter.** A special character that denotes the start of GML markup. In the starter set, it is initially a colon (:).

**GML end tag delimiter.** A special character that denotes the end of GML markup. In the starter set, it is initially a period (.).

**GML interpretation.** Interpretation of GML markup consists of recognizing the start or end of an element (or an attribute label), associating it with an APF, and executing the APF. In SCRIPT/VS, interpretation is performed jointly by SCRIPT/VS itself and by APFs.

**graphic character modification (GCM) module.** Modules that contain scan patterns of IBM-supplied character sets and/or user-defined character sets, without respect to particular processing.

**Graphical Data Display Manager (GDDM).** An IBM licensed program that creates page segments.



**gutter.** In multicolumn formatting, the space between columns.

## H

**hanging indentation.** The indentation of all lines of a block of text following the first line (which is not indented the same number of spaces). Specified with the .OF [Offset] or .UN [Undent] control word.

**head-level.** The typeface and character size associated with the words standing at the beginning of a chapter or chapter topic.

**heading.** Words located at the beginning of a chapter or section or at the top of a page. See also *head-level* and *running heading*.

**heading segment.** An element that begins with a heading, followed by basic document elements and lower-level heading segments.

**hexadecimal.** Pertaining to a number system based on 16, using the sixteen digits 0, 1, . . . 9, A, B, C, D, E, and F. For example, X'1B' equals decimal 27. See also *EBCDIC*.

**highlighting.** Emphasis associated with a document element. In formatting, highlighting is usually expressed by changing font, overstriking, underscoring, and/or capitalizing the highlighted element.

**horizontal justification.** The process of redistributing the extra horizontal white space at the end of a line of text in between the words and letters of the line so as to exactly fill the width of the column with the text.

## I

**IEBIMAGE.** Utility program that creates and maintains various 3800 Printing Subsystem Model 1 modules (for example, character arrangement table and graphic character modification (GCM) modules) and stores them in SYS1.IMAGELIB.

**image.** A likeness or imitation of an object, such as a picture or logo.

**impact printer.** A printer, such as the 1403 and the 3211, in which printing is the result of mechanical impacts.

**implied paragraph structure.** An element that begins with an implied paragraph; that is, one for which you do not specifically enter a paragraph tag. The existence of the paragraph is understood from the existence of the implied paragraph structure, for example, as with notes, figure captions, and footnotes.

**indent.** To set typographical material to the right of the left margin.

**indention.** The action of indenting. The condition of being indented. The blank space produced by indenting. Specified with the .IN [Indent], .IR [Indent Right], .UN [Undent], .OF [Offset], and .IL [Indent Line] control words. See also *hanging indentation*.

**initial caps.** Capital letters occurring as the first letter of each word in a phrase. To set a phrase in initial caps is to capitalize the first letter of each word in the phrase.

**initial value.** A value assumed by SCRIPT/VS for a formatting function until the value is explicitly changed with a control word. The *initial value* is assumed even before the control word is encountered, whereas the *default* value is assumed when the control word is issued without parameters. See also *default value*.

**initialize.** This is a general programming term which means to set everything up correctly at the beginning before you actually do any processing. For the Document Composition Facility: Generalized Markup Language Starter Set it means doing things such as mapping tags to APFs and setting up symbol names and values.

**inline space.** An amount of horizontal white space in a line that usually occurs between two words.

**input device.** A machine used to enter information into a computer system (for example, a terminal used to create a document).

**input line.** A line, as entered into a source file, to be processed by a formatter.

**interactive.** Pertaining to an application in which entries call forth a response from a system or program, as in an inquiry system. An interactive system might also be conversational, implying a continuous dialog between the user and the system. Interactive systems are usually communicated with via terminals, and respond immediately to commands. See also *foreground*.

**interactive environment.** The environment in which an interactive processor operates.

**intercharacter space.** Extra horizontal white space inserted *between* characters of a word. This space is in addition to the space included as part of the characters by the designer of the font.

**interword space.** See *word space*.

**Interactive System Productivity Facility (ISPF).** A dialog manager for interactive applications that provides control and services to allow processing of the dialogs in different host environments.

**italic.** A typestyle with characters that slant upward to the right.

### J

**job control language (JCL).** A language of job control statements used to identify a computer job or describe its requirements to the operating system.

**job control statement (JCS).** A statement that provides an operating system with information about the job being run.

**justification.** See *horizontal justification* and *vertical justification*.

**justify.** (1) (ISO) To control the printing positions of characters on a page so that the left-hand and right-hand margins of the printing are regular. (2) See *left-justify* and *right-justify*.

### K

**Kanji.** The non-phonetic Japanese writing system. In a font representing Kanji characters, each character is represented by a double-byte code. Contrast with Katakana.

**Katakana.** A character set consisting of symbols used in one of the two common Japanese phonetic alphabets. Each character is represented by one byte. Contrast with Kanji.

**keep.** (noun) In a source document, a collection of lines of text to be printed in the same column. When the vertical space remaining in the current column is insufficient for the block of text, the text is printed in the next column. (In the case of single-column format, the next column is on the next page.)

### L

**layout.** The arrangement of matter to be printed. See also *format*.

**leader.** (1) Dots or hyphens (as in a table of contents) used to lead the eye horizontally. (2) The divider between text and footnotes on a page (usually a short line of dashes that can be redefined).

**left-hand page.** The page on the left when a book is opened; usually even-numbered.

**left-justify.** (ISO) To control the printing positions of characters on a page so that the left-hand margins of the printing are regular.

**legend.** An explanatory list of the symbols, lines, and other components of a schedule.

**ligature.** A single character (piece of type or font raster) that represents two or more input characters: ff and ffi

are examples of characters that may be represented by (printed as) a ligature.

**line device.** Any of a class of printers that accept one line of text from the host system at a time. SCRIPT/VS supports such line devices as the 1403 and 3800 Model 1.

**line space.** The vertical distance between the baseline of the current line and the baseline of the previous line.

**line spacing.** See *line space*.

**logical output device.** The combination of a physical output device and such logical variables as page size and number of lines per vertical inch (for line devices). A specification of 1403W6 is an example of a logical output device.

**logical page.** Synonym for page.

**lowercase.** Pertaining to small letters as distinguished from capitals; for example, a, b, g rather than A, B, G.

### M

**machine-readable.** Data in a form such that a machine can acquire or interpret (read) it from a storage device, from a data medium, or from another source.

**maclib.** See *macro library*

**macro.** See *macro instruction*

**macro instruction.** An instruction in a source language that is to be replaced by a defined sequence of instructions in the same source language. In SCRIPT/VS, a macro is a sequence of one or more control words, symbols, and input lines. A macro's definition can be recursive.

**macro library.** A collection of macros. The form the library takes will vary by environment, being a MACLIB in CMS, a PDS in TSO, and so on.

**macro substitution.** During formatting, the substitution of control words, symbols, and text for a macro.

**map.** Associate a tag with an APF using the .AA [Associate APF] control word.

**margin.** (1) The space above, below, and on either side of the body of a page. (2) The left or right limit of a column.

**mark up.** (1) (verb) To determine the markup for a document. (2) (verb) To insert markup into a source document.

**markup.** (noun) Information added to a document that enables a person or system to process it. Markup can describe the document's characteristics, or it can specify

the actual processing to be performed. In SCRIPT/VS, markup consists of GML tags, attribute labels and values, and control words.

**markup-content separator.** A delimiter used in GML markup which indicates the end of the markup and the beginning of the text. The default markup content separator for GML is a period (.).

**maximum ascender.** The maximum height from the baseline to the top of the character in the font character set.

**maximum descender.** The maximum depth from the baseline to the bottom of the character in the font character set.

**meter (m).** Basic unit of linear measurement.

**MCS.** Markup/content separator.

**millimeter (mm).** One thousandth of a meter. There are 10 millimeters in one centimeter. (25.4 millimeters = 1 inch.)

## N

**nonimpact printer.** A printer, such as the 3800 Printing Subsystem, in which printing is not the result of mechanical impacts, but is instead produced by another process such as laser beam, ink-jet, or electro-erosion. The 3800 Printing Subsystem, for example, uses a laser based technology and the 4250 Printer uses an electro-erosion process.

## O

**object.** A sequential collection of control records that represent documents, pages, fonts, and so on.

**offset.** (1) (verb) To indent all lines of a block of text, except the first line. (2) (noun) The indentation of all lines of a block of text following the first line.

**option.** Information entered with a SCRIPT command to control the execution of SCRIPT/VS.

**orientation.** The angle between the top or bottom edge of the page and the baselines within a column, measured in a clockwise direction.

**output device.** A machine used to print, display, or store the result of processing.

**output document.** A machine-readable collection of lines of text or images that have been formatted, or otherwise processed, by a document processor. The output document can be printed or it can be filed for future processing.

**output line.** A line of text produced by a formatter.

## P

**page.** A collection of data that can be printed on a physical sheet of paper. Synonymous with logical page.

**PAGEDEF.** Synonym for page definition.

**page definition.** An object containing a set of formatting controls for printing pages of data. Includes controls for number of lines per printer form, font selection, print direction, and for mapping individual fields in the data to positions on the forms. Synonymous with PAGEDEF.

**page device.** A device that prints a formatted page that has graphics and text merged.

**page printer.** Any of a class of printers that accept composed pages, constructed of composed text and images, among other things. SCRIPT/VS supports such page printers as the 4250 Printer, the 3800 Model 3, and the 3820 Page Printer.

**page segment.** See *segment*.

**paginate.** To number pages.

**paragraph unit.** An element that has the same structure as a paragraph. In a General Document, the paragraph units are: paragraph, note, and paragraph continuation.

**parameter.** Any one of a set of properties whose values determine the characteristics or behavior of something. The syntax of some SCRIPT/VS control words includes parameters, which establish the properties of a formatting function or a printed page.

**part.** In a general document, a part is a zero-level heading segment. See also *heading segment*.

**patch PSC element.** A PSC element that is used temporarily to modify the normal output.

**pel.** The unit of horizontal measurement for the 3800 Printing Subsystem and the 4250 Printer. On the 3800 Printing Subsystem Model 1, one pel equals approximately 1/180th inch. On the 3800 Model 3 and the 3820 Page Printer, one pel equals approximately 1/240th inch. On the 4250 Printer, one pel equals approximately 1/600th inch.

**physical output device.** A physical device, such as a terminal, a disk file, a line printer, or a nonimpact printer. The 1403 printer is an example of a physical output device.

**pica.** A unit of about 1/6 inch used in measuring typographical material. Similar to a cicero in the Didot point system.

## Glossary

**pitch.** A number that represents the amount of horizontal space a font's character occupies on a line. For example, 10-pitch means 10 characters per inch, or each character is 0.1 (1/10) inch wide. 12-pitch means 12 characters per inch, and 15-pitch means 15 characters per inch.

**point.** (1) A unit of about 1/72 of an inch used in measuring typographical material. There are twelve points to the pica. (2) In the Didot point system, a unit of 0.0148 inch. There are twelve Didot points to the cicero.

**PostScript language.** A programming language designed to convey a description of virtually any desired page to a printer capable of interpreting the page description.

**PostScript devices.** Any of a class of devices, such as the IBM 4216 Personal Pageprinter, that are configured to accept 8-bit ASCII and include PostScript files in DCF documents.

**PostScript image file.** Any file containing encapsulated PostScript that is imbedded in a DCF document by means of the .PO [PostScript] control word. PostScript image files can include any combination of images or text.

**profile.** (1) In SCRIPT/VS processing, a file that is imbedded before the primary file is processed. It can be used to control the formatting of a class of source documents. When processing GML markup, the profile usually contains the mapping from GML to APFs and the symbol settings that define the formatting style. (2) In the DLF library, a collection of information that identifies a batch SCRIPT/VS user (user profile) or a document processor (attribute profile) or that defines certain library parameters (system profile).

**proportional spacing.** The spacing of characters in a printed line so that each character is allotted a space proportional to the character's width.

## R

**ragged right.** The unjustified right edge of text lines. See also *left-justify*.

**ragged left.** The unjustified right edge of text lines. See also *right-justify*.

**reference element.** In a general document, an element whose content is a reference to another element that is generated by an APF. There are five: figure reference, footnote reference, heading reference, index entry reference, and list item reference.

**required blank.** A character that prints as a blank, but does not act as a word separator.

**residual text.** The line of text following the markup/content separator of a GML tag.

**right-hand page.** The page on the right when a book is opened; usually odd-numbered.

**right-justify.** (ISO) To control the printing positions of characters on a page so that the right-hand margins of the printing are regular.

**row.** A horizontal arrangement of characters or other expressions on a printed page.

**rule.** (1) A straight horizontal or vertical line used, for example, to separate or border the parts of a figure or box. (2) A solid black rectangle of a given width, extending horizontally across the column or vertically down the column.

**running footing.** A footing that is repeated above the bottom margin area on consecutive pages (or consecutive odd- or even-numbered pages) in the body of the page (text area).

**running heading.** A heading that is repeated below the top margin area on consecutive pages (or consecutive odd- or even-numbered pages) in the body of the page (text area).

## S

**SCRIPT/VS.** The formatter component of the Document Composition Facility. SCRIPT/VS provides capabilities for text formatting and document management, macro processing and symbol substitution, and GML tag recognition and processing.

**section.** When an output page has two or more single-column parts with the same or different column-widths, or a single-column part and a multicolumn part, or two or more different multicolumn parts, each part of the output page is called a section.

**segment.** An object containing composed text and images, prepared before formatting and included in a document when it is printed.

**set.** This term is used in reference to a symbol. It implies the .SE [Set Symbol] control word.

**set size.** The set size of a given typeface determines the number of characters that will fit in a line of a given width when it is printed or set.

**slip end date.** The end date of a schedule or project that has moved or *slipped* to a later time than originally planned.

**slip start date.** The start date of a schedule or project that has moved or *slipped* to a later time than originally planned.

**small caps.** Capital letters in the same style as the normal capital letters in a font, but approximately the size of the lowercase letters.

**source document.** A machine-readable collection of lines of text or images that is used for input to a computer program.

**space.** A blank area separating words or lines.

**space unit.** A unit of measure of horizontal or vertical space. In GML markup, the *em* is used when a measure that is relative to the current font size is required. When an absolute measure is required, as in specifying the depth of a figure, recommended space units are inches (*nnI*), millimeters (*nnW*), picas/points (*nnPnn*), or Ciceros/Didot points (*nnCnn*), where *nn* is the number of units. See also *em*, *pica*, *point*, *Cicero*, and *Didot point system*.

**starter set.** An example of GML support that is provided with the Document Composition Facility. It consists of a document type description for general documents, a profile, and a library of APFs.

**SYSVAR.** An option of the *SCRIPT* command that permits the user to specify values for symbols. In the starter set, *SYSVAR* symbol values determine whether certain processing variations will occur, such as heading numbering, duplex formatting, and two-column printing.

**structure.** A characteristic of a document (or element) that expresses the type and relationship of the elements of the content. See also *element*.

**structured field.** A self-identifying string of bytes, analogous to a logical record. A structured field consists of an introducer, which identifies and characterizes the structured field, and data or parameters.

**symbol.** A name in a source document that can be replaced with something else. In *SCRIPT/VS*, a symbol is replaced with a character string. *SCRIPT/VS* can interpret the character string as a number, a character string, a control word, or another symbol.

**symbol substitution.** During formatting, the replacement of a symbol with a character string that *SCRIPT/VS* can interpret as a value (numeric, character string, or control word) or as another symbol.

**string.** A linear sequence of entities such as characters or physical elements.

## T

**tab.** (1) (noun) A preset point in the typing line of a typewriter-like terminal. A preset point in an output line. (2) (verb) To advance to a tab for printing or typing. (3) (noun) a tab character, *X'05'*.

**table (ISO).** (1) An array of data each item of which may be unambiguously identified by means of one or more arguments. (2) An arrangement of cells in rows and columns.

**tag.** In GML markup, a name for a type of document (or document element) that is entered in the source document to identify it. For example, *:p.* might be the tag used to identify each paragraph.

**terminal.** A device, usually equipped with a keyboard and some kind of display, capable of sending and receiving information over a communication channel.

**text item.** Explicitly marked (tagged) elements that occur within text, such as within a paragraph unit. In a general document, for example, quotations and phrases are text items.

**text programmer.** One who implements APFs that provide the processing specified by the document administrator. In *SCRIPT/VS*, this involves writing *SCRIPT/VS* macros and organizing macro libraries and profile files so that the appropriate composition will be done for each tag.

**text line.** An input line that contains only text.

**text variable.** A symbol whose final value is to be treated as text only.

**token.** A string of characters that is treated as a single entity. In *SCRIPT/VS*, a parameter passed to a macro in one of the local variables *&\*1*, ... *&\*n*.

**top margin.** On a page, the space between the body or running heading and the top edge of the page.

**translate table.** That 256-byte portion of the character arrangement table that translates the user's data code for a character recognizable by the 3800 Printing Subsystem Model 1.

**transparency.** A master or copy on material that transmits light without diffusion.

**TRC.** Table reference character. In printer *SYSOUT* data sets, a second control byte, following the carriage control byte, which indicates which font the record is to be printed in. The presence of TRCs is indicated by the JCL parameter *DCB=OPTCD=J*.

**TSO.** An interactive processor within *OS/VS2*.

## Index

**type posture.** A typeface style variation indicating whether a typeface is upright (as in Roman) or slanted to the right (as in italic or cursive).

**type size.** The vertical height (point size) of a given typeface, such as 10 point.

**type style.** Style variations in a typeface. Among these variations are posture, weight, and width.

**type weight.** The relative thickness of the strokes of a typeface. Usually described in such terms as light, demi bold, bold, and so on.

**type width.** The horizontal size (set size) of a given typeface. The width may be given in units of measurement, such as set 9 point, or it may be descriptive: ultra condensed, condensed, expanded, and so on.

**typeface.** All type of a single style. There might be several fonts (different sizes) with the same typeface or style.

**typeface family.** A collection of fonts of a common typeface that vary in size and style.

**typeset.** (1) (verb) To arrange the type on a page for printing. (2) (adjective) Pertaining to material that has been set in type.

## U

**underscore.** (1) (noun) A line printed under a character. (2) (verb) To place a line under a character.

**unformatted mode.** (1) In document formatting, the state in which each input line is processed and printed without formatting. Other SCRIPT/VS control words remain in effect and are recognized. (2) In document printing using the UNFORMAT option, the state in which each input line (control words as well as text) is printed as it exists in the input, in the order in which it is processed. No formatting is done.

**unique identifier (ID).** In a general document, an attribute whose value serves as a name which can be used to refer to the element. See also *reference element*.

**unit space.** The minimum amount of additional spacing acceptable for purposes of horizontal justification, as specified by the font designer.

**uppercase.** Pertaining to capital letters, as distinguished from small letters; for example, A, B, G rather than a, b, g.

## V

**variable.** A quantity that can assume any of a given set of values.

**variable text.** For the .VT [Variable Text] control word, text to be inserted in a formatted document by a post processor.

**vertical justification.** The process of redistributing the extra vertical white space at the end of a column in between the lines of text, so as to make each column in a set appear to be equal in depth.

## W

**widow.** One or two lines or words at the end of a paragraph that are printed separately from the rest of the paragraph.

**word space.** The horizontal white space placed between words; referred to as an interword blank.

**word spacing.** The space between words in a line. See also *word space*.

**Writable Character Generation Module (WCGM).** A 64-position portion of the 3800 Printing Subsystem Model 1's character generation storage that holds the scan elements of one character set. There are two WCGMs in the basic 3800, and optional added storage provides two more.

## X

**XPO.** (see exposure)

# Index

## A

- attributes
  - for memo tags (see memo tag attributes)
  - for overhead transparency tags (see overhead transparency tag attributes)
  - for schedule tags (see schedule tag attributes)

## B

- blank lines in entry
  - with memos 14
  - with overhead transparencies 26
  - with schedules 44
- BookMaster
  - program number 1
  - related applications 1
  - related products 56

## C

- courses
  - self-study 55
  - training 55
- creating memos
  - attributes for memo tags (see memo tag attributes)
  - examples of rmemos
    - external memo 12
    - internal memo 10
  - memo tags
    - :CC 9
    - :FROM 7
    - :LOGO 6
    - :MEMODATE 7
    - :MEMOTXT 8
    - :REF 8
    - :SECURITY. 7
    - :SIGN 9
    - :SUBJECT 8
    - :TO 6
  - things to remember
    - blank lines 14
    - formatting command 14
- creating overhead transparencies
  - attributes for overhead transparency tags (see overhead transparency tag attributes)
  - examples of overhead transparencies
    - head-level examples 20
    - text example 24
    - title-page example 18
  - overhead transparency tags
    - :TRFOOT 17
    - :TRHEAD 16
  - things to remember
    - blank lines 26
    - GML starter set tags 26

- creating overhead transparencies (*continued*)
  - things to remember (*continued*)
    - script formatting commands 26
- creating schedules
  - attributes for schedule tags (see schedule tag attributes)
  - examples of schedules
    - schedule with staffing information 42
    - schedule with STITLE and DATETYPE 40
    - schedule with :ANNOTATE 38
    - schedule with :ASSUM, :DEPEND, and :XPO 36
  - schedule tags
    - :ANNOTATE 33
    - :ASSUM 34
    - :DEPEND 34
    - :EASSUM 34
    - :EDEPEND 35
    - :EKEYC 35
    - :ESCHEDULE 33
    - :EXPO 35
    - :KEYC 35
    - :SCHDATE 33
    - :SCHEDULE 28
    - :SCHI 30
    - :XPO 35
  - things to remember
    - blank lines 44
    - formatting command 44
    - GML starter set tags 44

## D

- dates
  - for memos 7
  - for overhead transparencies 17
  - for schedules
    - AEND (actual end date) 31
    - ANNOTATE date 33
    - ASTART (actual start date) 31
    - date 28
    - DATETYPE (USA or EUROPEAN) 29
    - end date 28
    - PEND (projected end date) 30
    - PSTART (projected start date) 30
    - SCHDATE date table 33
    - SEND (slip end date) 31
    - SSTART (slip start date) 31
    - start date 28
- DCF program number 1
- DCF publications 55
  - library guide 54
- diagrams, syntax (see syntax diagrams)

### E

- examples
  - of memos (see memo examples)
  - of overhead transparencies (see overhead transparency examples)
  - of schedules (see schedule examples)
- external memo example 12

### F

- fold-out page, use of 1
- footings
  - on overhead transparencies 17
  - on schedules 29

### H

- headings
  - on overhead transparencies 16
  - on schedules 28

### I

- internal memo example 10

### L

- library guide
  - for DCF publications 54

### M

- memo examples
  - external memo 12
  - internal memo 10
- memo profile 5
- memo tag attributes
  - for the :CC tag
    - ADDR1 9
    - ADDR2 9
    - NAME 9
  - for the :FROM tag
    - ADDR1 7
    - ADDR2 7
    - DEPT 7
    - NAME 7
    - PHONE 7
    - TITLE 7
  - for the :LOGO tag
    - LOGONAME 6
  - for the :MEMOTXT tag
    - MEMTYPE 8
  - for the :TO tag
    - ADDR1 6
    - ADDR2 6
    - COMPANY 6
    - NAME 6
- memo tags
  - :CC 9

#### memo tags (*continued*)

- :FROM 7
- :LOGO 6
- :MEMODATE 7
- :MEMOTXT 8
- :REF 8
- :SECURITY. 7
- :SIGN 9
- :SUBJECT 8
- :TO 6

- memo tags and attributes syntax (see syntax diagrams)
- memos, creation of (see creating memos)

### O

- overhead transparencies, creation of (see creating overhead transparencies)
- overhead transparency examples
  - head-level examples 20
  - text example 24
- overhead transparency profile 15
- overhead transparency tag attributes
  - for the :TRFOOT tag
    - CENTER 17
    - LEFT 17
    - LOGONAME 17
    - LOGOPOS 17
    - RIGHT 17
  - for the :TRHEAD tag
    - LEFT 16
    - LOGONAME 16
    - LOGOPOS 16
    - RIGHT 16
    - SUBJECT 16
- overhead transparency tags
  - :TRFOOT 17
  - :TRHEAD 16
- overhead transparency tags and attributes syntax (see syntax diagrams)
  - title-page example 18
- overhead transparency things to remember
  - blank lines 26
  - GML starter set tags 26
  - script formatting commands 26

### P

- products, related (see related products)
- profiles
  - memo profile 5
  - overhead transparency profile 15
  - schedule profile 27
- publications
  - for DCF 55
- publication, related (see related publications)



**R**

- related products
  - BookMaster 56
  - MARKUP 56
- related publications 53

**S**

- schedule examples
  - schedule with staffing information 42
  - schedule with STITLE and DATETYPE 40
  - schedule with :ANNOTATE 38
  - schedule with :ASSUM, :DEPEND, and :XPO 36
- schedule profile 27
- schedule tag attributes
  - for the :ANNOTATE tag
    - ALIGN 33
    - DATE 33
  - for the :ASSUM tag
    - COMPACT 34
  - for the :DEPEND tag
    - COMPACT 34
  - for the :KEYC tag
    - COMPACT 35
  - for the :SCHEDULE tag
    - ACTSIZE 28
    - DATE 28
    - DATETYPE 29
    - END 28
    - GMONTH 29
    - INTERVAL 29
    - LEGEND 29
    - MONTH 29
    - QUARTER 29
    - START 28
    - STITLE 29
    - TODAY 29
    - YEAR 29
  - for the :SCHI tag
    - AEND 31
    - ASTART 31
    - HEAD 31
    - NODBOX 31
    - PEND 30
    - PSTART 30
    - RESP 31
    - SEND 31
    - SSTART 31
  - for the :XPO tag
    - COMPACT 35
- schedule tags
  - :ANNOTATE 33
  - :ASSUM 34
  - :DEPEND 34
  - :EASSUM 34
  - :EDEPEND 35
  - :EKEYC 35
  - :ESCHEDULE 33

schedule tags (*continued*)

- :EXPO 35
- :KEYC 35
- :SCHDATE 33
- :SCHEDULE 28
- :SCHI 30
- :XPO 35
- schedule tags and attributes syntax (see syntax diagrams)
- schedule things to remember
  - GML starter set tags 44
- schedules, creation of (see creating schedules)
- self-study courses 55
- syntax diagrams
  - for memos (see fold-out page, Chapter 2)
  - for overhead transparencies (see fold-out page, Chapter 3)
  - for schedules (see fold-out page, Chapter 4)

**T**

- tags
  - for memos (see memo tags)
  - for overhead transparencies (see overhead transparency tags)
  - for schedules (see schedule tags)
- things to remember
  - for memos
    - blank lines 14
    - formatting command 14
  - for overhead transparencies
    - blank lines 26
    - GML starter set tags 26
    - script formatting commands 26
  - for schedules
    - blank lines 44
    - formatting command 44
    - GML starter set tags 44
    - GML starter set tags 26, 44
- training courses 55

**U**

- use of fold-out page 1

**Special Characters**

- :ANNOTATE 33
- :ASSUM 34
- :CC 9
- :DEPEND 34
- :EDEPEND 35
- :EKEYC 35
- :ESCHEDULE 33
- :EXPO 35
- :FROM 7
- :KEYC 35
- :LOGO 6
- :MEMODATE 7

## **Index**

:MEMOTXT 8  
:REF 8  
:SCHDATE 33  
:SCHEDULE 28  
:SCHI 30  
:SECURITY. 7  
:SIGN 9  
:SUBJECT 8  
:TO 6  
:TRFOOT 17  
:TRHEAD 16  
:XPO 35

Order No. G544-3305-00

Use this form to provide comments about this publication, its organization, or subject matter. Understand that IBM may use the information any way it believes appropriate, without incurring any obligation to you. Your comments will be sent to the author's department for the appropriate action. Comments may be written in your language.

**Note:** IBM publications are not stocked at the location to which this form is addressed. Direct requests for publications or for assistance in using your IBM system, to your IBM representative or local IBM branch office.

- |   | Yes                      | No                       |
|---|--------------------------|--------------------------|
| • Does the publication meet your needs? | <input type="checkbox"/> | <input type="checkbox"/> |
| • Did you find the information:         |                          |                          |
| Accurate?                               | <input type="checkbox"/> | <input type="checkbox"/> |
| Easy to read and understand?            | <input type="checkbox"/> | <input type="checkbox"/> |
| Easy to retrieve?                       | <input type="checkbox"/> | <input type="checkbox"/> |
| Organized for convenient use?           | <input type="checkbox"/> | <input type="checkbox"/> |
| Legible?                                | <input type="checkbox"/> | <input type="checkbox"/> |
| Complete?                               | <input type="checkbox"/> | <input type="checkbox"/> |
| Well illustrated?                       | <input type="checkbox"/> | <input type="checkbox"/> |
| Written for your technical level?       | <input type="checkbox"/> | <input type="checkbox"/> |
| • Do you use this publication:          |                          |                          |
| As an introduction to the subject?      | <input type="checkbox"/> | <input type="checkbox"/> |
| As a reference manual?                  | <input type="checkbox"/> | <input type="checkbox"/> |
| For advanced knowledge of the subject?  | <input type="checkbox"/> | <input type="checkbox"/> |
| To learn about operating procedures?    | <input type="checkbox"/> | <input type="checkbox"/> |
| As an instructor in class?              | <input type="checkbox"/> | <input type="checkbox"/> |
| As a student in class?                  | <input type="checkbox"/> | <input type="checkbox"/> |
| • What is your occupation?              |                          |                          |

**Comments:**

To receive a reply provide your name, mailing address, and date:

---

---

---

---

Thank you for your input and cooperation.

**Note:** If mailed in the U.S.A., no postage stamp is necessary. For residents outside the U.S.A., your local IBM office or representative will forward your comments.

Reader's Comment Form



Fold and Tape

Please Do Not Staple

Fold and Tape

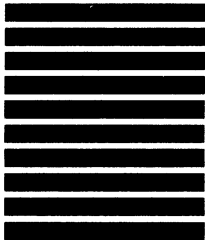


NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**

FIRST CLASS    PERMIT NO.40    ARMONK, NY

POSTAGE WILL BE PAID BY ADDRESSEE



International Business Machines Corporation  
Department 588  
P. O. Box 1900  
Boulder, Colorado U.S.A. 80301-9191



Fold and Tape

Please Do Not Staple

Fold and Tape







Program Number  
5748-XX9

File Number  
S370-40

G544-3305-00



Printed in U.S.A.