

Not Measurement  
Sensitive

MIL-HDBK-28001  
30 June 1995

## **MILITARY HANDBOOK**

# **DEPARTMENT OF DEFENSE APPLICATION OF MIL-PRF-28001 USING STANDARD GENERALIZED MARKUP LANGUAGE (SGML)**



AMSC N/A

AREA IPSC

DISTRIBUTION STATEMENT A: Approved for public release; distribution is unlimited.

## **MIL-HDBK-28001**

### **FOREWORD**

1. This military handbook is approved for use by all Departments and Agencies of the Department of Defense.
2. Beneficial comments (recommendations, additions, deletions) and any pertinent data which may be of use in improving this document should be addressed to: ATTN: CALS Digital Standards Office, DISA, Center for Standards, Code: JIEO/JEBEB, 10701 Parkridge Blvd., Reston, VA 22091-4398 by using the self-addressed Standardization Document Improvement Proposal (DD Form 1426) appearing at the end of this document or by letter.
3. This document supplements departmental manuals, directives, and military standards, and provides basic and fundamental information on Standard Generalized Markup Language (SGML).
4. The use of Courier font in this handbook represents SGML notation.

`<!ELEMENT charfill - O EMPTY>.`

**MIL-HDBK-28001****CONTENTS**

<b><u>PARAGRAPH</u></b>		<b><u>PAGE</u></b>
1.	<b>SCOPE</b> .....	1
1.1	Scope .....	1
1.2	Applicability .....	2
2.	<b>REFERENCE MATERIAL</b> .....	3
2.1	Government documents .....	3
2.1.1	Specifications, standards, and handbooks. ....	3
2.2	Non-Government publications .....	4
2.3	Informational documents .....	4
3.	<b>DEFINITIONS</b> .....	5
3.1	Acronyms used in this handbook .....	5
3.2	Terms .....	6
4.	<b>CONTINUOUS ACQUISITION AND LIFE-CYCLE SUPPORT (CALs)</b>	
	<b>SGML</b> .....	12
4.1	Purpose of SGML .....	12
4.2	Intended use .....	12
4.3	Background of SGML .....	12
4.4	CALS SGML documents .....	13
4.5	CALS document delivery/interchange .....	15
5.	<b>DOCUMENT ANALYSIS</b> .....	16
5.1	Document analysis .....	16
5.2	DTD considerations. ....	17
5.2.1	Use and modification of existing DTDs .....	17
5.2.2	Identifying elements and their relationships .....	18
5.2.2.1	Element and attribute naming guidelines .....	18
6.	<b>DOCUMENT TYPE DEFINITION (DTD)</b> .....	20
6.1	Document Type Definition (DTD) .....	20
6.1.1	Document type declaration .....	21
6.1.2	Entity declarations .....	22
6.1.2.1	General entities .....	22
6.1.2.2	Parameter entities .....	22
6.1.2.3	General versus parameter entities .....	23
6.1.2.4	Data type entities .....	23
6.1.2.5	External entities .....	24
6.1.2.6	System versus public entities .....	24
6.1.3	Element declaration .....	24
6.1.3.1	Model group .....	25
6.1.3.2	Start- and end-tag minimization .....	27

**MIL-HDBK-28001****CONTENTS**

<b><u>PARAGRAPH</u></b>		<b><u>PAGE</u></b>
6.1.3.3	Declared content or content model . . . . .	28
6.1.3.4	Exceptions to content models . . . . .	28
6.1.3.5	Attribute list declaration . . . . .	29
6.1.3.5.1	Attribute definitions . . . . .	29
6.1.3.5.2	Attribute name . . . . .	29
6.1.3.5.3	Declared values . . . . .	29
6.1.3.5.4	Default values . . . . .	30
6.1.4	Notation declaration . . . . .	30
6.1.5	Comments . . . . .	30
6.1.6	Modular DTDs . . . . .	31
6.2	Example DTD . . . . .	31
<b>7.</b>	<b>CREATING THE DOCUMENT INSTANCE . . . . .</b>	<b>32</b>
7.1	Background . . . . .	32
7.2	SGML coded text source file (document instance) contents. . . . .	32
7.3	Authoring environments . . . . .	32
7.3.1	Templates . . . . .	32
7.3.2	Entity libraries . . . . .	33
7.3.2.1	Entities . . . . .	33
7.3.2.2	Public entity libraries (text, file, graphic) . . . . .	33
7.3.2.3	Legacy information . . . . .	33
7.4	Parsing . . . . .	33
7.5	Tagging philosophies . . . . .	33
7.5.1	Structure tagging . . . . .	33
7.5.2	Content tagging . . . . .	34
7.5.3	Combined structure and content tagging . . . . .	34
7.6	Tagging guidance . . . . .	34
7.6.1	Security . . . . .	34
7.6.2	Space characters . . . . .	35
7.6.3	Wordspace . . . . .	35
7.6.4	Comment declarations . . . . .	35
7.7	Tag reuse and standardization . . . . .	35
7.7.1	DoD efforts to standardize tags . . . . .	36
7.7.2	New tags . . . . .	36
<b>8.</b>	<b>MATHPAC GUIDANCE . . . . .</b>	<b>37</b>
8.1.	Introduction . . . . .	37
8.1.1	Formatting of mathematical elements . . . . .	37
8.1.2	General guidance . . . . .	37
8.2	Formulas . . . . .	38
8.2.1	Inline formulas . . . . .	38
8.2.2	Display formulas . . . . .	39

**MIL-HDBK-28001****CONTENTS**

<b><u>PARAGRAPH</u></b>		<b><u>PAGE</u></b>
8.3	Tagging various formulas . . . . .	40
8.3.1	Notation . . . . .	40
8.3.1.1	Superscripts and subscripts . . . . .	40
8.3.1.2	Fractions . . . . .	40
8.3.1.3	Powers . . . . .	40
8.3.1.4	Roots . . . . .	41
8.3.1.5	Derivatives . . . . .	41
8.3.1.6	Integral . . . . .	41
8.3.1.7	Summations . . . . .	42
8.3.1.8	Products . . . . .	42
8.3.1.9	Vectors . . . . .	42
8.3.1.10	General operators . . . . .	42
8.3.1.11	Matrices . . . . .	42
8.3.1.12	Tensors . . . . .	43
8.3.1.13	Functional notation . . . . .	43
8.3.2	Presentation and placement . . . . .	44
8.3.2.1	Fonts . . . . .	44
8.3.2.2	Character embellishment . . . . .	44
8.3.2.3	Breaking formulas . . . . .	44
8.3.2.4	Aligning formulas . . . . .	45
8.3.2.5	Columns (piles) . . . . .	45
8.3.2.6	Fences . . . . .	45
8.3.2.7	Boxes . . . . .	46
8.4	Detailed examples . . . . .	46
8.5	Problems . . . . .	47
8.5.1	Mixed content problems . . . . .	47
8.5.2	Mathpac tagging problems . . . . .	47
9.	<b>PARTIAL DOCUMENTS . . . . .</b>	<b>48</b>
9.1	Partial documents . . . . .	48
9.1.1	Concepts . . . . .	48
9.1.2	Transmittal master . . . . .	48
10.	<b>CREATING AND USING FORMATTING OUTPUT SPECIFICATION</b>	
	<b>INSTANCES (FOSI) . . . . .</b>	<b>51</b>
10.1	Scope . . . . .	51
10.2	Statement of purpose, premises . . . . .	51
10.3	Media . . . . .	51
10.4	Design goals . . . . .	51
10.5	Page integrity and page fidelity . . . . .	51
10.6	Machine parseability . . . . .	52
10.7	The Output Specification (OS) goal . . . . .	52

**MIL-HDBK-28001****CONTENTS**

<b><u>PARAGRAPH</u></b>		<b><u>PAGE</u></b>
10.8	The Output Specification Document Type Definition (OS DTD) . . . . .	52
10.9	Organization of the Output Specification Document Type Definition (OS DTD) (categories of composition characteristics) . . . . .	52
10.10	Formatting Output Specification Instance (FOSI) . . . . .	52
10.10.1	Definition of a FOSI . . . . .	52
10.10.2	When to use a FOSI . . . . .	53
10.10.3	Why use a FOSI . . . . .	53
10.11	FOSI development guidance . . . . .	53
10.11.1	Understanding the requirements . . . . .	53
10.11.2	Understanding OS concepts . . . . .	54
10.11.2.1	Purpose of a FOSI . . . . .	54
10.11.2.2	Identification and treatment of source data . . . . .	54
10.11.2.3	Cognizance of source DTD . . . . .	54
10.11.2.4	Significant record ends . . . . .	54
10.11.3	Organizing and documenting a FOSI . . . . .	55
10.11.3.1	Specifying elements-in-context (e-i-c) . . . . .	55
10.11.3.1.1	Technique - e-i-c order in the FOSI . . . . .	55
10.11.3.2	Technique - documenting a FOSI . . . . .	56
10.11.4	Setting up the resource description . . . . .	56
10.11.5	Setting up the security description . . . . .	56
10.11.6	Setting up page models . . . . .	56
10.11.6.1	Technique - using page sets . . . . .	56
10.11.6.2	Technique - setting page parameters . . . . .	57
10.11.6.2.1	Layout areas . . . . .	57
10.11.6.2.2	Page set . . . . .	57
10.11.6.3	Technique - using page references . . . . .	58
10.11.6.4	Technique - setting up headers and footers . . . . .	60
10.11.6.5	Positioning techniques . . . . .	60
10.11.6.6	Using security values . . . . .	60
10.11.6.7	Using page numbers . . . . .	60
10.11.6.8	Technique - setting up footnote areas . . . . .	61
10.11.6.9	Technique - controlling floating elements. . . . .	61
10.11.6.9.1	Floating figure - title at top of figure . . . . .	61
10.11.6.9.2	Floating figure - title at bottom of figure . . . . .	63
10.11.6.9.3	Multi-sheet figure - title at top of figure . . . . .	64
10.11.6.9.4	Inline figure - title at bottom of figure . . . . .	65
10.11.6.9.5	Figure on same page with associated text - title at top of figure . . . . .	67
10.11.6.9.6	Figure on facing page with associated text - title at bottom of figure . . . . .	68
10.11.6.9.7	Figure on same sheet with associated text - title at bottom of figure . . . . .	70
10.11.6.9.8	All figures placed in a separate section . . . . .	72
10.11.7	Setting up the style description . . . . .	73
10.11.8	Setting up the document default . . . . .	73

## MIL-HDBK-28001

## CONTENTS

<u>PARAGRAPH</u>		<u>PAGE</u>
10.11.9	Setting up environments . . . . .	74
10.11.9.1	Charsubsets - using characteristic list subsets . . . . .	74
10.11.10	Determining element categories . . . . .	74
10.11.11	Describing elements . . . . .	74
10.11.11.1	Technique - grouping elements . . . . .	74
10.11.12	Other techniques . . . . .	74
10.11.12.1	Using context and occurrence . . . . .	74
10.11.12.2	Using inheritance and defaulting . . . . .	75
10.11.12.3	Determining the font . . . . .	75
10.11.12.4	Specifying leading . . . . .	76
10.11.12.5	Controlling hyphenation . . . . .	76
10.11.12.6	Specifying word spacing, letter spacing, and kerning . . . . .	76
10.11.12.7	Using indents . . . . .	77
10.11.12.8	Boxing . . . . .	77
10.11.12.9	Specifying quadding . . . . .	78
10.11.12.10	Using highlighting . . . . .	78
10.11.12.11	Specifying change marks . . . . .	78
10.11.12.12	Specifying prespace and postspace . . . . .	78
10.11.12.13	Specifying keeps . . . . .	79
10.11.12.14	Specifying vertical justification parameters . . . . .	79
10.11.12.15	Specifying text breaks (textbrk) . . . . .	79
10.11.12.16	Using spans . . . . .	80
10.11.12.17	Using borders . . . . .	80
10.11.12.18	Using rules . . . . .	80
10.11.12.19	Using character fills . . . . .	80
10.11.12.20	Using automatic numbering . . . . .	80
10.11.12.21	Suppressing text . . . . .	80
10.11.12.22	Puttext - using generated text . . . . .	81
10.11.12.23	Putgraph - using pre-defined graphics . . . . .	81
10.11.12.24	Savetext - saving text for multiple uses . . . . .	81
10.11.12.25	Usetext - using saved text . . . . .	82
10.11.12.25.1	Sort userule - userule #1 . . . . .	83
10.11.12.25.2	System/Subsystem/Subject Number (SSSN) userule - userule #2 . . . . .	86
10.11.12.26	Interaction of puttext, putgraph, ruling, and usetext . . . . .	87
10.11.12.27	Using string . . . . .	87
10.11.12.28	Using cross references . . . . .	88
10.11.12.28.1	How to set up the FOSI . . . . .	88
10.11.12.29	Defining leader dots . . . . .	89
10.11.12.30	Building a Table of Contents (TOC) . . . . .	90
10.11.12.31	Pseudo-elements . . . . .	91
10.11.13	Handling source attributes . . . . .	91
10.11.13.1	Technique - identifying source attributes . . . . .	91

## MIL-HDBK-28001

## CONTENTS

<u>PARAGRAPH</u>		<u>PAGE</u>
10.11.13.2	Technique - specifying the use of the attribute . . . . .	91
10.11.13.3	Technique - interaction of attributes and other values . . . . .	92
10.11.14	Describing graphics . . . . .	92
10.11.14.1	Sources of graphics information . . . . .	92
10.11.14.2	Interaction of the reproduction window, sizing, and placement . . . . .	93
10.11.14.3	World coordinates . . . . .	93
10.11.14.4	Assumptions . . . . .	93
10.11.14.5	Available space . . . . .	94
10.11.15	Graphic characteristics . . . . .	94
10.11.15.1	Reproduction area dimensions . . . . .	94
10.11.15.2	Graphic sizing (sizing) . . . . .	94
10.11.15.3	Text block . . . . .	95
10.11.15.3.1	Placement (placemnt) . . . . .	95
10.11.15.4	Elements and graphics attributes . . . . .	96
10.11.16	Important terms related to graphics . . . . .	97
10.11.17	FOSI graphic environment . . . . .	97
10.11.17.1	Default characteristic values . . . . .	98
10.11.18	Graphic . . . . .	99
10.11.18.1	Required attribute . . . . .	99
10.11.18.1.1	Boardno . . . . .	99
10.11.18.2	Optional (implied) attributes . . . . .	100
10.11.18.2.1	Reprowid & reprodep . . . . .	100
10.11.18.3	Implied values of remaining attributes . . . . .	101
10.11.18.3.1	Llccorda & rucorda . . . . .	101
10.11.18.4	Implied values of remaining attributes . . . . .	103
10.11.18.4.1	Hscale & vscale . . . . .	103
10.11.18.5	Implied values of remaining attributes . . . . .	104
10.11.18.5.1	Scalefit . . . . .	104
10.11.18.6	Implied values of remaining attributes . . . . .	105
10.11.18.6.1	Hplace & vplace . . . . .	105
10.11.18.7	Implied values of remaining attributes . . . . .	107
10.11.18.7.1	Coordst & coordend . . . . .	107
10.11.18.8	Implied values of remaining attributes . . . . .	109
10.11.18.8.1	Rotation . . . . .	109
10.11.18.9	Implied values of remaining attributes . . . . .	110
10.11.18.10	Graphsty . . . . .	110
10.11.18.11	Describing tables . . . . .	110
10.11.18.11.1	Tables as graphics . . . . .	111
10.11.18.11.2	The structure of tables . . . . .	111
10.11.18.11.3	Source structure . . . . .	111
10.11.18.11.4	Output structure and table objects . . . . .	112
10.11.18.12	Mapping source structure to output structure . . . . .	112

**MIL-HDBK-28001****CONTENTS**

<b><u>PARAGRAPH</u></b>		<b><u>PAGE</u></b>
10.11.18.13	Text flow rules of tables . . . . .	114
10.11.18.14	Specifying the style of a table . . . . .	115
10.11.18.14.1	Table characteristics . . . . .	115
10.11.18.14.2	Table subset group and table subset characteristics . . . . .	115
10.11.18.14.3	Scope of table characteristics . . . . .	115
10.11.18.14.4	The content of tables . . . . .	116
10.11.19	Overriding FOSI values through the source markup . . . . .	116
10.11.19.1	Table characteristics . . . . .	116
10.11.20	Standard cell characteristics . . . . .	117
10.11.20.1	Table subset group characteristics . . . . .	118
10.11.20.1.1	Table subset characteristics . . . . .	119
10.11.20.1.2	Column characteristics . . . . .	119
10.11.20.1.3	Row characteristics (rowatts) . . . . .	121
10.11.20.1.4	Cell characteristics (cellatts) . . . . .	121
10.11.21	Describing footnotes . . . . .	121
10.12	Parsing the FOSI . . . . .	124
11.	<b>ELECTRONIC REVIEW . . . . .</b>	<b>125</b>
11.1	Electronic review . . . . .	125
11.1.1	Electronic review process improvement . . . . .	125
11.1.2	Electronic review comments . . . . .	125
11.1.3	Electronic review declaration set overview . . . . .	125
11.2	Electronic review declaration set . . . . .	126
11.3	Using the electronic review toolkit . . . . .	127
11.4	Electronic review functionality . . . . .	128
11.5	Electronic review user interface . . . . .	129
11.6	Electronic review process . . . . .	129
11.7	Modreq tag set description . . . . .	131
12.	<b>SGML REUSE LIBRARY (SRL) AND SGML TAGSET REGISTRY (STR) . . . . .</b>	<b>132</b>
12.1	Background . . . . .	132
12.2	DoD SGML standardization and reuse program . . . . .	132
12.3	CALS SGML Library (CSL) . . . . .	133
12.3.1	CSL Librarian . . . . .	133
12.3.2	SGML Reuse Library (SRL) . . . . .	134
12.3.3	SGML Tagset Registry (STR) . . . . .	134
12.4	Summary of operations . . . . .	135
12.4.1	Scenario 1, direct reuse . . . . .	135
12.4.2	Scenario 2, modified reuse . . . . .	135

## MIL-HDBK-28001

## CONTENTS

<u>APPENDICES</u>	<u>PAGE</u>
A. EXAMPLE DOCUMENT TYPE DEFINITION .....	137
B. COORDINATED SAMPLE DOCUMENT .....	223
C. MATHPAC TAG SET .....	228
D. MODREQ TAG SET .....	237
E. MODULAR DTD .....	242
F. ELEMENT AND ATTRIBUTE INDEX .....	243
CONCLUDING MATERIAL .....	251
 <u>FIGURES</u>	
FIGURE 1. Page model layout .....	59
FIGURE 2. Sample graphic 1 .....	99
FIGURE 3. Sample graphic 2 .....	101
FIGURE 4. Sample graphic 3 .....	102
FIGURE 5. Sample graphic 4 .....	102
FIGURE 6. Sample graphic 5 .....	103
FIGURE 7. Sample graphic 6 .....	104
FIGURE 8. Sample graphic 7 .....	104
FIGURE 9. Sample graphic 8 .....	105
FIGURE 10. Sample graphic 9 .....	106
FIGURE 11. Sample graphic 10 .....	106
FIGURE 12. Sample graphic 11 .....	107
FIGURE 13. Sample graphic 12 .....	108
FIGURE 14. Sample graphic 13 .....	109
FIGURE 15. Sample graphic 14 .....	110
FIGURE 16. DoD SGML standardization and reuse program organization .....	132
FIGURE 17. Sample output .....	227
 <u>TABLES</u>	
I Element and attribute description .....	157
II Attribute set description .....	217
III Mathpac tag set description .....	228
IV Mathpac attribute set description .....	235
V Modreq tag set description .....	237

## **MIL-HDBK-28001**

### **1. SCOPE**

**1.1 Scope.** This handbook provides guidance in the application of MIL-PRF-28001, Markup Requirements and Generic Style Specification for Electronic Printed Output and Exchange of Text, which is based on ISO 8879, Information processing - Text and office systems - Standard Generalized Markup Language (SGML). Data prepared in accordance with these guidelines will facilitate the automated storage, retrieval, interchange, and processing of technical documents from varied data sources. Section 2 of this document lists applicable reference material and section 3 provides useful definitions. Section 4 provides an overview of SGML in the Continuous Acquisition and Life-Cycle Support (CALS) strategy. Sections 5 through 12 of this handbook describe the following tasks:

- a. Performing a document analysis.
- b. Developing a Document Type Definition (DTD) based on a document analysis.
- c. Creating the document instance.
- d. Tagging mathematical formulae and constructs.
- e. Preparing partial documents.
- f. Preparing a Formatting Output Specification Instance (FOSI) in accordance with the output specification in MIL-PRF-28001.
- g. Using the electronic review declaration set.
- h. Using the SGML Reuse Library and SGML Tagset Registry.

Appendix A contains an example DTD and its corresponding element and attribute definition tables for a complex technical document.

Appendix B contains an example of an entire SGML application of a simple business letter (document type declaration, document instance, FOSI, and printed output).

Appendix C provides descriptions of SGML tags for mathematical notations (mathpac).

Appendix D provides descriptions of SGML tags for electronic review and comment (modreq).

Appendix E is reserved for a description of modular DTDs. The content of this appendix will be provided in a change notice or a revision to this handbook.

An index to the SGML elements and attributes used in this handbook is provided in appendix F.

## **MIL-HDBK-28001**

**1.2 Applicability.** This handbook applies to digital data and document development which are required to be compliant with MIL-PRF-28001. It is in the interest of both DoD and industry to agree on the most widely applicable set of conventions for the preparation and interchange of textual data for both defense and non-defense use. To use this handbook, a minimal knowledge of SGML will be helpful but not essential. In addition, the reader is expected to have a knowledge of document content and style specifications and knowledge of publishing and composition terminology.

**MIL-HDBK-28001****2. REFERENCE MATERIAL****2.1 Government documents.**

**2.1.1 Specifications, standards, and handbooks.** The following specifications, standards, and handbooks form a part of this document to the extent specified herein. Unless otherwise specified, the issues of these documents are those listed in the Department of Defense Index of Specifications and Standards (DODISS) and supplements thereto, cited in the solicitation.

**SPECIFICATIONS****DEPARTMENT OF DEFENSE**

- |               |   |  |
|---------------|---|--|
| MIL-PRF-28000 | - | Digital Representation for Communication of Product Data: IGES Application Subsets and IGES Application Protocols. |
| MIL-PRF-28001 | - | Markup Requirements and Generic Style Specification for Electronic Printed Output and Exchange of Text.            |
| MIL-PRF-28002 | - | Raster Graphics Representation in Binary Format, Requirements For.   |
| MIL-PRF-28003 | - | Digital Representation for Communication of Illustration Data: CGM Application Profile.                            |
| MIL-M-38784   | - | Technical Manuals: General Style and Format Requirements.  |

**STANDARDS****FEDERAL INFORMATION PROCESSING STANDARDS**

- |              |   |  |
|--------------|---|--|
| FIPS PUB 152 | - | Standard Generalized Markup Language (SGML). |
|--------------|---|--|

(Copies of the Federal Information Processing Standards (FIPS) are available to Department of Defense activities from the Standardization Document Order Desk, 700 Robbins Avenue, Building 4D, Philadelphia, PA 19111-5094. Others must request copies of FIPS from the National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161.)

**DEPARTMENT OF DEFENSE**

- |              |   |  |
|--------------|---|--|
| MIL-STD-12   | - | Abbreviations for use on Drawings, and in Specifications, Standards and Technical Documents. |
| MIL-STD-1840 | - | Automated Interchange of Technical Information.  |

**MIL-HDBK-28001****REGULATIONS****DEPARTMENT OF DEFENSE**

DOD 5200.1-R - Information Security Program Regulation.

(Unless otherwise indicated, copies of federal and military specifications, standards, handbooks are available from: Defense Printing Service Detachment Office, Attn: Customer Service, Building 4D, 700 Robbins Avenue, Philadelphia PA 19111-5094.

**2.2 Non-Government publications.** The following documents form a part of this document to the extent specified herein. Unless otherwise specified, the issues of the documents which are DoD-adopted are those listed in the issue of the Department of Defense Index of Specifications and Standards (DODISS) cited in the solicitation.

**INTERNATIONAL ORGANIZATION FOR STANDARDIZATION (ISO)**

ISO 646	-	Information Processing - ISO 7-Bit Coded Character set for Information Interchange - 3rd Edition.
ISO 8879	-	Information Processing - Text and office systems - Standard Generalized Markup Language (SGML).
ISO/IEC TR 9573	-	Information Processing - SGML Support Facilities - Techniques for Using SGML.

(Copies are available from the Defense Printing Service Detachment Office, Attn: Customer Service, Building 4D, 700 Robbins Avenue, Philadelphia PA, 19111-5094 for issue to DoD activities only. All other requesters may obtain documents from the American National Standards Institute, 11 West 42nd Street, 13 Floor, New York, NY 10036.

**2.3 Informational documents.** The following documents are listed for informational purposes only and do not form a part of this handbook. They may be obtained from their publishers.

The SGML Handbook, Charles M. Goldfarb, Oxford University Press, 1990.

Practical SGML, Eric van Herwijnen, Kluwer Academic Publishers, 1990.

SGML: An author's guide to the Standard Generalized Markup Language, Martin Bryan, Addison-Wesley, 1988.

SGML: The User's Guide to ISO 8879, Joan M. Smith, and Robert Stutely, John Wiley, 1988.

SGML and Related Standards Document Descriptions and Processing Languages, Joan M. Smith, Ellis Horwood, 1992.

**MIL-HDBK-28001****3. DEFINITIONS**

**3.1 Acronyms used in this handbook.** The acronyms used in this handbook are defined as follows:

a.	<b>ARL</b>	-	<b>ASSET Reuse Library</b>
b.	<b>ASCII</b>	-	<b>American Standard Code for Information Interchange</b>
c.	<b>ASSET</b>	-	<b>Asset Source for Software Engineering Technology</b>
d.	<b>CALS</b>	-	<b>Continuous Acquisition and Life-Cycle Support</b>
e.	<b>CCITT</b>	-	<b>Consultative Committee for International Telegraphy and Telephony</b>
f.	<b>CFS</b>	-	<b>Center For Standards</b>
g.	<b>CDRL</b>	-	<b>Contract Data Requirements List</b>
h.	<b>CGM</b>	-	<b>Computer Graphics Metafile</b>
i.	<b>CSL</b>	-	<b>CALS SGML Library</b>
j.	<b>CSR</b>	-	<b>CALS SGML Registry</b>
k.	<b>CSSPO</b>	-	<b>CALS SGML Standardization Policy Office</b>
l.	<b>DISA</b>	-	<b>Defense Information Systems Agency</b>
m.	<b>DoD</b>	-	<b>Department of Defense</b>
n.	<b>DODISS</b>	-	<b>Department of Defense Index of Specifications and Standards</b>
o.	<b>DTD</b>	-	<b>Document Type Definition</b>
p.	<b>e-i-c</b>	-	<b>Element in Context</b>
q.	<b>FOSI</b>	-	<b>Formatting Output Specification Instance</b>
r.	<b>IGES</b>	-	<b>Initial Graphics Exchange Specification</b>
s.	<b>ISO</b>	-	<b>International Organization for Standardization</b>
t.	<b>NET</b>	-	<b>Null End Tag</b>

**MIL-HDBK-28001**

u.	<b>OS</b>	-	Output Specification
v.	<b>PDL</b>	-	Page Description Language
w.	<b>SGML</b>	-	Standard Generalized Markup Language
x.	<b>SRL</b>	-	SGML Reuse Library
y.	<b>SSSN</b>	-	System/Subsystem/Subject Number
z.	<b>STR</b>	-	SGML Tagset Registry
aa.	<b>WYSIWYG</b>	-	What You See Is What You Get

**3.2 Terms.**

- a. **ANY.** This means that the element's content can consist of #PCDATA or the content of all elements defined in the Document Type Definition (DTD) in any order. The content of an element must conform to that element's content model.
- b. **Application.** The related set of processes performed on documents of related types.
- c. **ATTLIST.** See attribute definition list.
- d. **Attribute.** A characteristic quality, other than type or content. An attribute provides more information about an element such as classification level, unique reference identifiers, or formatting information (ISO 8879).
- e. **Attribute definition list.** A set of one or more attribute definitions defined by the attribute definitions list parameter of an attribute definition list declarations (ISO 8879).
- f. **CDATA - (character data).** Zero or more characters that occur in a context in which no markup is recognized, other than the delimiters that end the character data. Such characters are classified as data because they were declared to be so (ISO 8879).
- g. **Concrete syntax.** Association of the abstract syntax with specific delimiter characters, quantities, markup declaration names, etc (ISO 8879).
- h. **CONREF - (content reference attribute).** An implied attribute whose default value is #CONREF. If this attribute is not specified on its element's tag, the element's content is specified by the element's content model. If this attribute is specified on its element's tag, the element's content in the document instance is empty, and the tag is processed by the application in some special way.

**MIL-HDBK-28001**

- i. **Content model.** A parameter in an element declaration consisting of either ANY or a model group along with any exclusions or inclusions that define the element's allowable content (ISO 8879).
- j. **CURRENT - (#CURRENT).** The attribute defaults to the previous value on all subsequent usage of that element type.
- k. **Data content notation.** An application's translation of an element's data content, or of a data entity, that may be different from the normal meaning of the document character set (ISO 8879).
- l. **Descriptive markup.** Markup that describes the structure and other attributes of a document in a non-system-specific manner, independently of any processing that may be performed on it. In particular, it uses tags to express the element structure.
- m. **DOCTYPE.** See document type declaration.
- n. **Document.** A collection of information, which may include data (text, graphics, tables, and numbers).
- o. **Document instance.** Instance of a document type. A document or computer file that contains (textual) data content annotated with SGML markup in accordance with the applicable DTD (ISO 8879).
- p. **Document type.** A class of documents having similar characteristics; for example, journal, article, technical manual, or memo (ISO 8879).
- q. **Document type declaration.** The declaration which invokes a DTD in an SGML document. The document instance of an SGML document must always be preceded by a document type declaration.
- r. **Document type declaration subset.** The element, entity, and short reference sets occurring within the declaration subset of a document type declaration. The external entity referenced from the document type declaration is considered part of the declaration subset (ISO 8879).
- s. **Document type definition.** Defines the markup language or rules for a distinct class or type of document.
- t. **DTD.** See document type definition.
- u. **Element.** A component of the hierarchical structure defined by a document type definition; it is identified in a document instance by descriptive markup, usually a start-tag and end-tag (ISO 8879).

**MIL-HDBK-28001**

- v. **Element in context (e-i-c).** The complete set of information necessary to identify a particular set of instances of an element type within a document, including its generic identifier, context, and occurrence.
- w. **Element type.** A class of elements having similar characteristics; for example, paragraph, chapter, abstract, footnote, or bibliography (ISO 8879).
- x. **Element type name.** See generic identifier.
- y. **e-i-c.** See element in context.
- z. **EMPTY.** Used to specify an element which can have no content in the markup of a document. The content is either implied or automatically generated by the system.
- aa. **Entity.** A collection of characters that can be referenced as a unit (ISO 8879).
- ab. **ENTITY.** Syntactically conforms to NAME and refers to a declared, externally defined SDATA, CDATA, or NDATA entity.
- ac. **ENTITIES.** One or more ENTITY.
- ad. **External entity.** An entity whose replacement text is not incorporated in the entity declaration. The system identifier and/or public identifier is specified instead (ISO 8879).
- ae. **External identifier.** A parameter that identifies an external entity or data content notation. There are two kinds of external identifiers, system identifiers and public identifiers (ISO 8879).
- af. **FIXED - (#FIXED).** Specifies an attribute's value to be unchangeable and equal to its default value.
- ag. **Fragment.** A piece of a FOSI or DTD that is modular for flexibility in usage in other applications.
- ah. **Formatting Output Specification Instance (FOSI).** An instance of the Output Specification (OS) that assigns values to the style characteristics for a particular document type declaration. The FOSI uses the syntax of an SGML declaration.
- ai. **Generic identifier (GI).** A name that identifies the element type of an element (ISO 8879).
- aj. **ID.** A unique identifier conforming syntactically to NAME.
- ak. **IDREF.** References an ID.

**MIL-HDBK-28001**

- al. **IDREFS.** One or more IDREF.
- am. **IMPLIED - (#IMPLIED).** Specifies an attribute as optional where the processing system may supply a value.
- an. **Implied attribute.** See IMPLIED.
- ao. **Markup.** Text that is added to the data of a document in order to convey information about the data.
- ap. **Model group.** Part of a content model which defines that part of an element's content in terms of delimiters, subelements, connectors, and #PCDATA. It is subject to possible modification with respect to any exceptions, i.e., exclusions and inclusions for the content model.
- aq. **Modular DTDs.** DTDs that are designed to be flexible and easy to reference for use in other SGML applications. For example, a base DTD may reference a table declaration set that is used in many applications.
- ar. **NAME.** A declared value keyword defining an attribute's value to be a name. NAME conforms in length to the NAMELEN parameter from the SGML Declaration, begins with a name start character (alpha by default), and has name characters (alpha, numeric) as well as "-" and "." for the subsequent characters.
- as. **NAMES.** One or more NAME separated by one or more parameter separator characters--space, tab, or carriage return (record end).
- at. **Name.** A name token whose first character is a name start character (ISO 8879).
- au. **Name token.** A character string containing the name characters a-Z, 0-9, ".", and "-". The number of characters in a name token must not exceed the value of NAMELEN quantity.
- av. **NAMELEN - (name length).** Quantity in the concrete syntax that limits the maximum length of entity and element names, name tokens, etc.
- aw. **NDATA - (non-SGML data).** Non-SGML data is data that is not parsable in accordance with ISO 8879. It is either data in an undefined character set, bit data, or some mix of the two. In undefined character set data, the bit combinations represent characters, but not in the document character set. In bit data, the bit combinations, although they can be managed as characters, do not represent a character repertoire in the usual way.
- ax. **NOTATION.** Where the external entity contains only details of the notation to be used for processing non-SGML Data.

**MIL-HDBK-28001**

- ay. **NMTOKEN - (name token).** A declared value keyword defining an attribute's value to be a name token, begins with and contains name characters.
- az. **NMTOKENS.** *One or more NMTOKEN.*
- ba. **NUMBER.** A declared value keyword defining an attribute's value to be a number conforming to NAMELEN.
- bb. **NUMBERS.** *One or more NUMBER.*
- bc. **Number.** A name token containing only the characters 0-9. The number of characters in a number must not exceed the value of the NAMELEN quantity (ISO 8879).
- bd. **Number token.** A name token whose first character must be a 0-9. The number of characters in a number token must not exceed the value of the NAMELEN quantity.
- be. **NUTOKEN - (number token).** A declared value keyword defining an attribute's value to be a number token. Beginning with a digit and then containing name characters.
- bf. **Owner identifier.** The portion of a public identifier that identifies the owner or originator of public text (ISO 8879).
- bg. **Page model.** Documents are typically made up of different types of pages, such as two column, one column or foldout pages. A page model is used in the FOSI to convey information about each type of page to be generated.
- bh. **Parameter entity.** An entity that is referenced from a markup declaration parameter (ISO 8879).
- bi. **Parser.** A software program that does two things. First it validates a DTD both with respect to the SGML grammar and syntax of ISO 8879 and the concrete syntax (SGML environment) of an SGML Declaration (e.g., the CALS SGML Declaration in MIL-PRF-28001). Secondly, it validates the document instance against the DTD with respect to the markup rules defined by the DTD, the SGML grammar and syntax of ISO 8879, and the SGML environment. Usually, a parser provides error returns for any non-conformance identified.
- bj. **PCDATA - (parsed character data).** Zero or more characters that occur in a context in which text is parsed and markup is recognized. They are classified as data characters because they were not recognized as markup during parsing (ISO 8879).
- bk. **Public entity.** An external entity with a public identifier that may be shared.

**MIL-HDBK-28001**

- bl. **Formal public identifier.** A public identifier that is constructed according to rules defined in ISO 8879 so that its owner identifier and the components of its text identifier can be distinguished (ISO 8879).
- bm. **Public Identifier - (PUBLIC).** An external identifier containing the keyword "PUBLIC" assigned to an externally stored DTD, entity, data content notation, etc..
- bn. **RCDATA - (replaceable character data).** Declared content that is similar to CDATA, except that entity references and character references are recognized. In RCDATA declared content a general entity reference is replaced with its replacement text which in turn may contain general entity or character references which must be replaced.
- bo. **Reference concrete syntax.** Specific values associated with the abstract syntax that are given in the standard. It also serves as a model from which other (variant) concrete syntaxes are formed.
- bp. **REQUIRED (#REQUIRED).** The attribute must always be given a value.
- bq. **SDATA - (system dependent data) entity.** Specific character data entity whose replacement text is treated as character data when referenced. The replacement text is dependent on a specific system, device, or application process.
- br. **Standard Generalized Markup Language (SGML).** A language for document representation that formalizes markup and frees it of system and processing dependencies.
- bs. **SGML Reuse Library.** A repository for document type declaration sets (see 12.3.2).
- bt. **SGML tagged text source file (referred to as a document instance in ISO 8879).** A text file marked up according to the structure and content defined by a DTD. This document contains data content and SGML markup in accordance with a specific DTD.
- bu. **SGML Tagset Registry.** A repository for SGML tags (see 12.3.3).
- bv. **Tag - (descriptive markup).** Delimiter for a logical element inside a document. The SGML standard defines a tag as descriptive markup. Tags are the names defined by the generic identifier in an element declaration.
- bw. **Tagging.** Inserting descriptive markup (tags) into the text to create an document instance which may be validated against a specific DTD.
- bx. **Transfer unit.** A collection of files consisting of one transfer unit declaration file and one or more data files (the smallest collection of files to make a successful interchange of technical information) (see MIL-STD-1840).

## MIL-HDBK-28001

### 4. CONTINUOUS ACQUISITION AND LIFE-CYCLE SUPPORT (CALS) SGML

4.1 **Purpose of SGML.** SGML as applied to MIL-PRF-28001 enables the following:

- a. Description of the logical structure and/or content of documents in unambiguous syntax (see 5.2),
- b. Assurance of automated quality control over adherence to that structure and/or content,
- c. Delivery and storage of technical publication text in an easily maintained and updatable form, and
- d. Provides for vendor, software, and platform independence.

4.2 **Intended use.** Preparation of documents in an automated support environment typically consists of the following steps:

- a. Creating a document type declaration for publication control if one does not exist,
- b. Creating a Formatting Output Specification Instance (FOSI) if one does not exist to specify the formatting to be applied to documents conforming with the document type declaration,
- c. Creating a document instance,
- d. Verifying that the syntax is correct according to the rules of SGML,
- e. Using a FOSI and document type declaration to direct the composition of the document so that the produced (printed or displayed) copy corresponds to the proper format and style,
- f. Optionally reviewing the document electronically using SGML for the comments, and
- g. Optionally generating a text presentation metafile in a page description language (PDL) to drive the display device, such as a printer or typesetter.

This handbook addresses these steps in the publication preparation process with the exception of the authoring process. Refer to applicable functional specifications for detailed authoring requirements.

4.3 **Background of SGML.** The term markup originated from the typesetting instructions handwritten on author's manuscripts. These instructions described the desired appearance of the text on a printed page. A similar scheme was adopted with the evolution of electronic publishing in order to provide instructions to an automated printing device. Generally, electronic publishing systems do not display formatting instructions along with the textual information, but may display the information in a What You See Is What You Get (WYSIWYG) form. The format or appearance instructions are usually stored in the same file as the textual information and are usually in a unique or proprietary

## MIL-HDBK-28001

format. This practice makes direct use of this file in another word processor difficult if not impossible. In addition, local, national, and international corporations and governments are forging new cooperative agreements which will require information sharing on massive scales. This means organizations will need to manage the growing volumes of information being generated and shared. These information management needs are being implemented through database paradigms. Traditional electronic publishing methods do not produce information structures that facilitate the database information management paradigm.

SGML was developed to address these information exchange and management challenges. The International Organization for Standardization (ISO) participated in the development of the SGML standard and adopted it in 1986 as ISO 8879. This standard completely defines the terms and syntax necessary to specify the structure and content of a document. ISO 8879 (SGML) does not provide a functionality for specifying appearance or formatting requirements except by means of system dependent processing instructions. Use of processing instructions is deprecated (unless necessary) in CALS practice. However, MIL-PRF-28001 contains an Output Specification (OS) Document Type Definition (DTD) for specifying appearance and formatting requirements for a class of documents through instances of the OS DTD. An instance of the OS DTD is referred to as a FOSI.

The Department of Defense (DoD) prepares and uses massive amounts of information, mostly in the form of printed documentation. Managing information in this form is difficult, error-prone, and costly. Seeing the potential benefits, DoD is implementing the SGML paradigm as part of the CALS strategy. The DoD application of SGML is specified in MIL-PRF-28001. ISO 8879 defines SGML as a meta-language. That is, many different SGML languages satisfying the grammar and syntax specified in ISO 8879 can be obtained by choosing among different features and options afforded by ISO 8879. MIL-PRF-28001 specifies the DoD CALS SGML implementation (or DoD SGML language) by choosing certain SGML features and setting certain parameters. MIL-PRF-28001 also uses SGML to provide a standard for specifying the formatting of CALS SGML documents.

**4.4 CALS SGML documents.** The parts of a CALS SGML document and how each relates to the others are described in the following paragraphs.

- a. **SGML Declaration.** The SGML Declaration defines what characters will be allowed in the rest of the document and how they will be encoded. Since there are many different encoding schemes available, the SGML Declaration is needed to ensure document transportability between different publishing systems. Both ISO 8879 and MIL-PRF-28001 specify the use of the ISO 646 character set commonly referred to as the American Standard Code for Information Interchange (ASCII). The SGML Declaration also specifies how certain characters are to be interpreted. For example, the reference concrete syntax for the character < is the start-tag open delimiter in the document instance. Other special rules and definitions are also included in the SGML Declaration. MIL-PRF-28001 specifies the SGML Declaration for CALS SGML applications.
- b. **Document type declaration.** The document type declaration associates a document type name with a DTD. An example of a document type declaration is provided in appendix B. The DTD occurs within a part of the document type declaration called

**MIL-HDBK-28001**

the document type declaration subset. The acronym DTD is occasionally (however incorrectly) used as an abbreviation for document type declaration. The document type declaration subset could occur entirely within the same entity as the document type declaration. More often, though, all or part of it is an external entity that is referenced within the document type declaration. When external, the document type declaration subset entity is often identified by a public identifier. DTD is an SGML reserved word used in public identifiers to indicate that the identified entity is a document type declaration subset. The document type declaration subset defines the content and structure of a document. It refers to how the information content in a class of documents such as technical manuals, books, memos, etc. is related and defines any dependencies. For example, a book may consist of front matter, followed by body matter, followed by rear matter. The body may be made up of chapters which consist of a required title followed by multiple sections.

The document segments are declared in the document type declaration subset as element type declarations and then referred to as elements. Each element type declared is given a unique name called an element type name (same as generic identifier in ISO 8879). These element type names are used as the tags to markup the document's information content and are also used by a parser to verify the content and structure of the document. Markup of a document provides an unambiguous definition of its structure and/or content. An automated data processing software, such as a FOSI processor, can be used to process the document in a predictable manner. For example, a document may require an index. If the document type declaration subset defines an index element type and the document instance is properly tagged, a processor designed to interpret FOSIs, will be able to automatically generate the index for the entire document.

Also defined in the document type declaration subset are entity declarations. Entities may be used to define data once and reference it many times. Entities are typically defined in the document type declaration subset and referenced in either the document instance or the DTD itself. The different types of entities that may be used are detailed in 6.1.2.

DTDs may be used for any class of documents that have the same content model and structure. DoD and industry have developed many DTDs and declared them to be PUBLIC. A PUBLIC DTD may be used to produce documents reflecting the structure defined in the DTD. Section 12 of this handbook describes the procedures needed to access some public DTDs.

- c. Document instance (referred to as an SGML coded text source file in MIL-STD-1840). The source file is the information content of the document, marked up in accordance with a DTD. This markup may include declarations, tags, and entity references. The first part of a document instance contains the document type declaration. This declaration identifies the DTD (usually identified as an external entity to the document type declaration) required to correctly interpret the document. Document type declarations may identify a PUBLIC DTD. A repository of some of the public tag sets

**MIL-HDBK-28001**

has been developed. The procedures to access the repository are described in section 12 of this handbook.

- d. **Formatting output specification instance.** The FOSI specifies the desired appearance of the information content of the document. This output formatting description capability is not contained in ISO 8879, but is found in MIL-PRF-28001. MIL-PRF-28001 contains an OS DTD that defines the requirements as to how FOSIs are to be developed and interpreted. Document-wide formatting requirements such as page layout and hyphenation rules are specified in the FOSI, as well as how document elements such as paragraph titles, tables, and lists are to be formatted. The FOSI, which is an instance of the OS DTD, provides the composition and imaging characteristics to be applied to the SGML tagging (including attributes) of a document instance to present the text material in paginated or screen presentation form.

**4.5 CALS document delivery/interchange.** The schedule, method, and format of document delivery will be governed by the contract. The contract statement of work and/or specification describes the information content of the document, and the DD Form 1423, Contract Data Requirements List (CDRL), describes the document delivery schedule and format. If the SGML Declaration, document type declaration, document instance (also known as a SGML tagged text source file in MIL-STD-1840), and FOSI are required and are not available, the CDRL should specify that each should be prepared in accordance with MIL-PRF-28001. If delivery is required, the CDRL should specify delivery in accordance with MIL-STD-1840.

MIL-STD-1840 describes the packaging structure and format for the transfer of digital data. If an SGML document and related files are required for delivery, then MIL-STD-1840 identifies what can be contained in an SGML transfer unit. The minimum number of files required by MIL-STD-1840 to be in an SGML transfer unit are two, the transfer unit declaration file (format specified in MIL-STD-1840) and one SGML coded text source file (also known as a document instance). Many other files may be included in an SGML transfer unit, such as a document type declaration file (if not already part of the SGML coded text source file), additional SGML coded text source files, entity files, illustration files, a FOSI data file, special word files, and other files as specified in the contract. There are several SGML document delivery options that may be specified in the contract.

Describing all the possible combinations and permutations of SGML document delivery is beyond the scope of this handbook. The organization requesting delivery needs to carefully determine their requirements and completely describe them in the contract. Contractors responsible for delivery of SGML documents must prepare deliveries in accordance with the CDRL.

## MIL-HDBK-28001

## 5. DOCUMENT ANALYSIS

**5.1 Document analysis.** Document analysis is a top-down approach to describe the boundaries of a document's structure and content and develop a document's structure and/or content description. A description of the document's content can be more explicit and detailed than a mere description of its structure. Greater detail can be provided in content tagging, giving better direction to the authors and enabling more extensive use of automated quality control. The structure of a document can be extracted via careful examination of the document. The key is to define the information in the document, not the format of the document. However, it should be recognized that *formatting requirements may influence DTD development*. People familiar with the document's structure/content are the best resource available for determining the elements required for the document and how these elements are related. For example, a paragraph may be on the same level as a numbered list for one document while in another document lists may be subelements of paragraphs. From this document structure definition, the DTD is written. The following steps describe one method for performing a document analysis:

- a. Determine if the document type fits into an existing class. There are many types of documents that are used regularly. After the analysis team has determined the type of document, check existing classes by consulting the SGML Reuse Library to determine if a DTD suitable for the application already exists. The contract should specify the DTDs which must be used and should contain requirements for Government approval of any new DTDs or deviations from existing DTDs. If it is necessary to modify the existing DTD or develop a new DTD, the DTD preparing activity should be contacted for approval. Modification of existing DTDs should be discouraged. With prior approval, it may be possible to modify the structure of the document instance and avoid modifying or creating a new DTD. There are many classes that can be considered such as:

Administrative Manuals  
Handbooks  
Plans  
Specifications  
Users Manuals

Amendments  
Maintenance Manuals  
Regulations  
Standards

Flight Manuals  
Pamphlets  
Reports  
Technical Manuals

- b. Collect document guidelines. Determine and collect the standards or specifications to which the DTD must conform, if applicable. Use the guidelines set forth in the specifications and/or standards when developing the document structure definition.
- c. Use many document examples. If available, assemble several representative documents of the same type to begin your analysis. Try to include examples of simple, average, and complex documents of this type. This will help to ensure that unusual as well as trivial components are not overlooked by the analysis team.

**MIL-HDBK-28001**

- d. Define the document purpose. Decide how data in the document may be used. Designs should be flexible to allow for future modifications. If guidelines are required, incorporate them from the beginning of the analysis.
- e. Identify the users. Who will be using the document and how they will be using the document must be identified in the analysis process. The document structure/content definition may need to be modified to incorporate all of the requirements of different users.
- f. Use a team effort to analyze the document. If possible, every person associated with the document should be involved in the document analysis procedure. People who are familiar with the document will be the most qualified to identify all of the necessary elements in the document. It is not important for the analysis team members to understand the intricacies of SGML. At all conclusion of the analysis, the team should understand and agree on all structure/content of the document type they have defined.
- g. Identify the logical elements and their relationships. Start by reviewing the documents. Develop a document structure outline. Use the table of contents, if there is one, as a good high level overview of the general document type structure. Determine the significance of typographic differences such as bold, underline, and italics. Look for subtle components such as index terms or keywords. Define the relationships among the identified components. Identify the optional, repeating, and required components.
- h. Produce a DTD. Using the defined document structure, a person familiar with SGML will be able to generate a DTD for the document type.
- i. Test the DTD. Tag representative samples of the document to ensure that the DTD correctly reflects the document structure and/or content.

**5.2 DTD considerations.** Both structural (chapter, titles, section, paragraph, list, etc.) and content (description, maintenance, operation, test, repair, task, etc.) elements can be used in describing documents; however, content elements enable more explicit and detailed description than is possible by merely describing a document with structural elements.

**5.2.1 Use and modification of existing DTDs.** An evaluation of the similarities between document types should be done in order to determine if modifying an existing DTD would be beneficial. If the documents are similar, it may be beneficial to modify the already existing DTD by modifying content models or by simply modifying sequence or occurrence indicators. If the documents at all are similar, it may be easier to work from an already existing DTD than it is to create one from the beginning. If it is necessary to modify the existing DTD or develop a new DTD, the DTD preparing activity should be contacted for approval.

Existing DTDs will reside in the SGML Reuse Library (SRL) (see 12.3.2). Procedures for obtaining a user account to the SRL are contained in section 12. Upon user account approval DTDs can be downloaded as needed. These DTDs and fragments will be in a usable form as they are verified (parsed) before being accepted into the library.

**MIL-HDBK-28001**

**5.2.2 Identifying elements and their relationships.** Using the defined document structure from the document analysis, the following suggestions may help in the identification of elements:

- a. Decide which items identified in the document should be elements.
  1. Items that look different in terms of format (bold, italic, underlined, indented).
  2. Items that make up the structure of the document (chapter, section, table).
  3. Different types of specific information (content such as names, addresses, or telephone numbers).
- b. Decide how those elements appear in relationship to each other.
  1. The order in which they appear (sequence).
  2. The number of times they appear (occurrence).
  3. Items which should be treated as inclusion or exclusion exceptions.
- c. Decide if certain content models or attribute lists will be used in multiple places, and if so, declare as parameter entities.

**5.2.2.1 Element and attribute naming guidelines.** The following guidelines facilitate establishing new element and attribute names when necessary:

- a. Tags from the SGML Tagset Registry (STR) should be used whenever possible.
- b. Tag name may relate to military specification terminology, but should be as generic as possible.
- c. Tags to denote multiple words may be in acronym or abbreviated form.
- d. Abbreviations should conform to MIL-STD-12 when appropriate.
- e. Where abbreviations are not in MIL-STD-12 or if abbreviations are in conflict with other words, a new abbreviation should be used (subject to approval by the STR Registrar).
- f. There will be a list of names, maintained by the STR Registrar, which should not be abbreviated.
- g. Some names may be abbreviated when used as part of a multiple word name, but the name when used alone should not be abbreviated.

## MIL-HDBK-28001

- h. Names should not be abbreviated if the abbreviation can be misinterpreted.
- i. An element may only be defined once in a DTD. Therefore, there may be a necessity to define an additional element when a different content model is needed for the same type of data in several places in a document.
- j. The xref elements should be used for internal cross-references; the extref element should be used for external cross-references. Avoid the use of page numbers and other format-driven information for cross-referencing.
- k. Number referencing should use "no" and not "num" for number.
- l. Tag descriptions and use should have generic meaning rather than a narrow meaning.
- m. Attributes must be in the STR and are associated with the elements to which they apply. The following apply:
  - 1. A new attribute may be added to an existing tag.
  - 2. New elements should have their attributes submitted with them.
  - 3. Attribute names should be reviewed and described.
  - 4. Attribute values and defaults will not be standardized in the library, the values and defaults should be assigned in the DTD in which they appear.
- n. Tags in the STR should reference the originating source DTDs in which they appear (when possible).
- o. When notes and notices have specific content names, they should have note or notice as a part of their name.
- p. Parameter entities denoting content models will not be a part of the STR with the exception of public entities.

## MIL-HDBK-28001

### 6. DOCUMENT TYPE DEFINITION (DTD)

**6.1 Document Type Definition (DTD).** The document type definition consists of a group of declarations that describes the components of a document. These components are called elements in a DTD, and are defined (or declared) for each unique part of the document, as identified in the document analysis. For example, a document may have structural elements such as chapter, title, section, para, and list; and content elements such as description, maintenance, operation, test, repair, and task. The introduction of an element into a DTD's structural hierarchy may also necessitate the introduction of other lower-level elements, i.e., the subelements of its content model (if any). An example of this is the element chapter which would have a content model comprising of a title, and then section(s); then the element's title and section would be declared as elements and introduce new elements in their content models. These element type names, or Generic Identifiers (GIs) as called in ISO 8879, are then used as tags to identify those parts of a document when creating the SGML instance. An element in a DTD may have attributes assigned to it in an attribute definition list provided by an ATTLIST declaration. Attributes allow the user to attach more information to the element. There may be multiple attributes in an ATTLIST for an element. An attribute would have a name similar to an element name, a value, and a default. The attribute may appear with its element or tag in the document instance. An example of this is `<para security="c">`. The DTD defines the structure and the rules for markup of a given class of documents. A DTD defines the following:

- a. The name and allowed content of each element.
- b. Frequency and order of appearance of each element.
- c. Start- and end-tag omission status for each element. With the exception of the mathpac elements, omissible start-tags are not allowed for elements in DoD DTDs.
- d. The attribute name, declared value, and default value for each of an element's attributes.
- e. All entities (general and parameter) referenced in the DTD.

All markup declarations begin with a markup declaration open (MDO) `<!` and end with the markup declaration close (MDC) `>`.

Markup declarations other than comment declarations and marked section declarations contain a keyword such as DOCTYPE, ELEMENT, ATTLIST, NOTATION, or ENTITY.

The following document type declaration models an office memo in order to demonstrate appropriate use of delimiters, keywords, and overall syntax.

```
<!--Document type declaration for a simple office memo -->
<!DOCTYPE memo [
<!ENTITY % mixed      "#PCDATA | list" >
<!ELEMENT memo - -    (header,body,sign) >
```

**MIL-HDBK-28001**

```

<!ATTLIST memo      route (internal | public) #IMPLIED
                      release (confid | general) "general">
<!ELEMENT header - - (to+ & from & subj & date) >
<!ELEMENT (to | from | subj | date) - o (#PCDATA) >
<!ATTLIST to distrib (group | unique) "group" >
<!ATTLIST from origin (person | depart) "person" >
<!ATTLIST date day (mon | tue | wed | thu | fri | sat | sun) "mon" >
<!ELEMENT body - - (para+) >
<!ELEMENT para - - (%mixed;)* >
<!ELEMENT list - - (item+) >
<!ELEMENT item - o (%mixed;)* >
<!ELEMENT sign - - (name,title?,phonext) >
<!ELEMENT (name,title,phonext) - o (#PCDATA) >
]>

```

The following declarations are described below:

- a. Document type declaration,
- b. Entity declaration.
- c. Element and associated attribute definition list declarations.
- d. Notation declaration.

Comments and comment declarations can be used throughout the document, as long as appropriate syntax is used (see 6.1.5).

DTDs shall also be submitted for placement in the CALS SGML Library.(see 12.0).

**6.1.1 Document type declaration.** A document type declaration identifies the document type or class of the document. For example, docmilspec could be used to identify military specification documents.

The first declaration is the document type declaration itself:

```

<!DOCTYPE document_type_name optional_external_identifier
[optional_document_type_declaration_subset]>

```

The document type declaration can provide the document type definition in several ways:

- a. The entire DTD can be provided in the document type declaration subset, the so-called internal subset.
- b. The entire DTD can be invoked from an external file (the so-called external subset) by the document type declaration using the optional external identifier.
- c. Part of the DTD can be supplied in the internal subset and the remainder in the external subset. In particular, parameter entities can be declared and referenced in the

**MIL-HDBK-28001**

document type declaration subset to provide part or all of the internal subset. It also should be remembered that an entity declaration in the internal subset takes precedence over any subsequent entity declaration for the same entity in the external or internal subsets.

**6.1.2 Entity declarations.** Entities may be either general or parameter. General entities are typically used for data that will be declared once and referenced many times. Parameter entities are often used as a shorthand method for specifying long model groups or other constructs that may be used many times. Entities whose replacement text is specified directly in the declaration are called internal and those whose replacement text is located elsewhere are called external. External entities in some sense have an existence of their own and may be declared and referenced in many documents. Entities are declared in the DTD or the document type declaration subset. They may be redefined. Entity names, unlike other SGML names, are case-sensitive.

The following subsections will discuss general entities, parameter entities, general versus parameter entities, external entities, and system versus public entities.

**6.1.2.1 General entities.** A general entity declaration follows the form:

```
<!ENTITY entity_name "entity_text">
```

The entity name is the information needed to identify the entity. It may be a quoted version of the entity's replacement text or a quoted string identifying an entity whose replacement is located elsewhere. The entity's replacement text is the real text of the entity, that is, the text that directly replaces a reference to the entity when the reference is detected during parsing. The entity's replacement text may be additional SGML text or non-SGML data. If it is non-SGML data, the replacement text must be located externally to the declaration and the entity text must contain the name of an appropriate data content notation.

An example of a general entity declaration is:

```
<!ENTITY dod "Department of Defense">
```

This entity would be referenced in the document instance with:

```
&dod;
```

**6.1.2.2 Parameter entities.** A parameter entity declaration follows the form:

```
<!ENTITY % name "entity_text">
```

Parameter entities are commonly referenced in the marked section declarations in instances (e.g., `%math;`) to provide the status keyword specification (INCLUDE, IGNORE, etc.) for a marked section. Parameter entities are often used as a shorthand method for specifying long model groups or other constructs that may be used many times or as a method of allowing changes to be made to existing content models. An example of this is the declaration:

**MIL-HDBK-28001**

```
<!ENTITY % text      "(#PCDATA | ftnref | indxflag | verbatim | xref |
                        graphic | subscript | supscrt | tool | testeq |
                        material | torqueval | dataiden | hrule | emergency |
                        change | emphasis | applicabil)+">
```

The parameter entity is referenced with:

```
%text;
```

### 6.1.2.3 General versus parameter entities.

- a. A general entity name can be up to 32 characters long and the parameter entity name can be up to 31 characters long.
- b. Parameter entities are declared using a percent sign followed by at least one space or carriage return before an SGML name. The name of a general entity in a general entity declaration is merely an SGML name.
- c. A parameter entity reference opens with a percent sign (%) immediately preceding the entity name, and a general entity reference opens with an ampersand (&).
- d. Both types of entity references are normally terminated by a semicolon immediately following the entity name.
- e. A space or a record end (hard return) or record end code following an entity reference will be interpreted as a reference closing delimiter. Therefore, textual constructions such as R&D and AT&T should be avoided as the &D and the &T will be interpreted as entities. In such circumstances, an entity reference can be used in place of the ampersand (& from the Numeric and Special Graphic character set). It is good practice to terminate an entity reference with a semicolon.

**6.1.2.4 Data type entities.** A data content notation must be specified on the ENTITY declaration for a data type entity and a NOTATION declaration must be provided for that data content notation. Consider the following example:

```
<!ENTITY fig-1 SYSTEM "C:\tms\cm59\fig1" NDATA cgmbin>
<!NOTATION cgmbin PUBLIC
    "ISO 8632/3//NOTATION CGM Binary text encoding//EN">
```

The following points should be noted. First, the NOTATION declaration does not have to precede the entity declaration. Secondly, system dependent external identifiers such as

```
SYSTEM "C:\tms\cm59\fig1"
```

do not have to provide a system identifier such as C:\tms\cm59\fig1 if the SGML system can locate the entity, DTD, etc. without one.

## MIL-HDBK-28001

**6.1.2.5 External entities.** External entities are commonly used entities kept separate from the declarations. The entity declaration simply contains a special identifier, either publicly known or known only to a particular system. A common use of external parameter entities is to contain an often-used set of declarations that might be incorporated by reference within the document type declaration of a document. The replacement text of such an entity is a document type declaration set. Similarly, if the replacement text of such an entity is entirely comment and entity declarations, it is known as an entity set. The terms document type declaration set and entity set apply to the replacement text of these entities. These terms are also used to describe the external entities themselves, provided the resolution of any parameter entities contained within the external entity is consistent with the containing entity's role as either a document type declaration set or entity set. External PUBLIC entities take the form:

```
<!ENTITY Entity_name PUBLIC "PUBLIC_identifier">
```

An example of an external PUBLIC parameter entity follows:

```
<!ENTITY % ISOTech PUBLIC
    "ISO 8879:1986//ENTITIES General Technical//EN">
```

**6.1.2.6 System versus public entities.** System entities are used to describe processing system dependencies. The following example shows how the file C:\text\sig.txt is identified as the system entity, signature.

```
<!ENTITY signature SYSTEM "C:\text\sig.txt">
```

On the other hand, public identifiers are external entities which are known beyond the context of the processing system. The entity is given a public identifier by an international, national, or industry standard, or by a group of users who wish to share the public identifier. The following example shows how the set of Greek letters is identified as the public entity, ISOgrk1.

```
<!ENTITY % ISOgrk1 PUBLIC "ISO 8879:1986//ENTITIES Greek Letters//EN">
```

**6.1.3 Element declaration.** Element declarations are used to define how and when elements may be used. They contain the name of the element type (or a parenthetical group of element type names), the start- and end-tag minimization, and the declared content or content model of the element type. If an element type has declared content, it will be specified as either CDATA, RCDATA, or EMPTY (see 3.2). In each of these cases, the element is a terminal node of the document's hierarchy and as such may contain no other subelements. If an element type has a content model, its content can be a model group or ANY. Additionally, exceptions may be part of the content model.

An element is declared to have mixed content if its content model contains #PCDATA besides sub-elements; otherwise the element has element content. For example,

```
<!ELEMENT para - o (#PCDATA | ftnote)* >
```

## MIL-HDBK-28001

is a mixed content element. It is recommended that only the or connector (|) is used in mixed content model groups. Mixed content should be avoided in cases where separators would be interpreted as data content. The following is an example of mixed content to avoid:

```
<!ELEMENT section - o (#PCDATA, subsect+) >
```

6.1.3.1 Model group. Groups are used in markup declarations for a list of elements and are called model groups.

- a. Use of connectors. Elements listed within a group are separated by connectors. There are three types of connectors:
  1. The sequence (seq) connector is the comma (.). It indicates that the elements within the group occur in the order or sequence in which they are encountered.
  2. The or connector is the vertical bar (|). It indicates that only one element within the group may be used each time the group is evaluated.
  3. The and connector is the ampersand (&). It indicates that the members of the group may occur in any order. The and connector adds contextual ambiguity and generally should be avoided in new document type declaration development.

Only one type of connector may join elements at the same level within a group. Of course, within a subsidiary parenthetical group a different connector could be used.

Some examples follow:

```
<!ELEMENT book - - (front, body, rear)>

<!--A book is made up of the elements front, body, and rear, each
occurring in that order. -->

<!ELEMENT front - - (title & author)>

<!--The front matter of the book is made up of a title and author;
however, either one may occur first. -->

<!ELEMENT body - - (chapter | section)>

<!--The body is made up of either a chapter or a section, but only
one can occur.-->

<!ELEMENT title - o (#PCDATA)>

<!-- A title is made up of parsed character data; no further
subdivision of elements is allowed. -->

<!ELEMENT rear - - ((appendix & index), (biblio | glossary))>
```

**MIL-HDBK-28001**

<!--Both an appendix and an index must occur; either one may occur first, the other immediately afterward. Following them, either a bibliography or a glossary must occur, but not both. Notice that each type of connector is used in a different group. One group is "(xxx & yyy)". Another group is "(xxx | yyy)". A third group is "((xxx), (yyy))". In this last example, the comma is connecting two groups rather than two element type names. Connectors can also be used between element types and groups, such as: (xxx, (yyy)). The last two examples also illustrate that a group may have only one element type name within it. -->

- b. Use of occurrence indicators. In addition to connectors, SGML also provides occurrence indicators. Occurrence indicators may modify a group or individual element type. They are:
1. The optional (opt) occurrence indicator, the question mark (?), indicates that the element type or group may occur either one time or not at all.
  2. The plus occurrence indicator, the plus sign (+), indicates that the element type or group must occur at least once.
  3. The repetition (rep) occurrence indicator, the asterisk (\*), indicates that the element type or group may occur zero or more times.
  4. The absence of an occurrence indicator means that the element type or group must occur once and only once.

Some examples follow:

```
<!ELEMENT book      - -    (front, body, rear?)>

<!--The front and body element types each must occur only once,
but the rear is optional. -->

<!ELEMENT front     - -    (title & author+)>

<!--There is one title allowed and at least one author must be
specified; however, there may be more than one author. The title
can be specified first and then the authors or all authors may be
specified and then the title. -->

<!ELEMENT body      - -    (chapters+ | sections+)>

<!--The body of the book may be made up of one or more chapters or
one or more sections. -->

<!ELEMENT title     - o    (#PCDATA)>

<!--#PCDATA consists of zero or more characters. Any general
entities and character references are recognized and resolved when
#PCDATA content is parsed. Moreover, start-tags and end-tags are
recognized.-->
```

## MIL-HDBK-28001

```
<!ELEMENT rear      - -      ((appendix+ & index)?, (biblio |
                                glossary)?)>
```

<!--The rear may or may not have a grouping of appendices and an index, but it may not have one without the other. Further, it may have more than one appendix, but only one index. All of the appendices or the index may occur first.

Then, either a bibliography or a glossary may optionally occur. If the group had been specified as (biblio | glossary)\* or (biblio | glossary)+, it would have meant that the group could be evaluated multiple times. This would mean that multiple bibliographies, multiple glossaries, one of each, or multiples of both might have occurred in any order.

Also notice that both groups "(appendix+ & index)?" and "(biblio | glossary)?" comprising the content of the rear element are optional so that it is possible for the content of rear to be nothing. -->

**6.1.3.2 Start- and end-tag minimization.** OMITTAG is an SGML feature that allows start- and end-tags to be omitted in the document instance in accordance with the rules of ISO 8879. Start-tag omission is discouraged (except for the mathpac tags) in CALS DTDs. In the element declaration, the tag minimization parameters follow the element type name, and precede the element's declared content or content model. A hyphen (-) specifies no omission; the upper or lower case "o" specifies that the tag need not be used if omissible under the rules of minimization outlined in the ISO standard. Once a parser has determined whether or not minimization of a tag is allowable based on the standard, it must also check the element type declaration of the element in question, to determine if the application designers have specified that the tag may be eliminated. For example, consider the following element type declarations:

```
<!ELEMENT doc      - -      (front, body, rear)>
<!ELEMENT front    - -      (titlepg, contents)>
```

In this example, the document contains front, body, and rear matter. The two hyphens between the word doc and the content model signify that the start- and end-tags for doc and front must be used.

The following example demonstrates a different effect:

```
<!ELEMENT safesum  - o      (para?, list, para)>
<!ELEMENT para     - o      (biblio | glossary)>
<!ELEMENT list     - -      (item+)>
```

The start-tag for the safety summary may not be minimized, but the end-tag may be omitted. Once the safety summary is started, an optional paragraph may appear or a list may start. Since the paragraph is not contextually required in this instance, its start-tag must be used if a paragraph occurs. Whether or not a paragraph is used, a list must occur. After the list is over, one paragraph must be used. In this example, one paragraph is contextually optional and one is contextually required. By looking at the element declaration for a paragraph, we see that the application designer has specified

**MIL-HDBK-28001**

that the end-tags for the paragraph are not required where they may be omitted under the rules for tag omission. In other words, if under the rules for tag omission, the start-tag of an element may not be omitted, an error must be reported if the start-tag is not encountered.

**6.1.3.3 Declared content or content model.** An element can have either declared content or a content model. The three types of declared content are:

<b>CDATA</b>	Consists of character data. Start tags, general entity references, and character references ARE NOT recognized as such. No markup is recognized other than the delimiters that end the character data.
<b>EMPTY</b>	Indicates that an element has no content. The end tag is not allowed in the document instance.
<b>RCDATA</b>	Consists of replaceable character data. Start tags are not recognized as such but general entity references and character references ARE recognized and resolved. No markup is recognized other than the delimiters that end the character data.

The content model of an element consists of either ANY or a model group comprised of proper subelements followed by the exceptions which consist of the exclusions and inclusions.

<b>ANY</b>	Consists of character data and the content of any element declared in the DTD. General entity references and character references are recognized and resolved.
------------	--

An example of a content model is the following:

```
(day | month | year)
```

**6.1.3.4 Exceptions to content models.** Exceptions to a content model may be exclusions, inclusions, or both. Exclusion exceptions modify the effect of model groups to which they apply by precluding content options that otherwise would have been available. Inclusion exceptions are used to specify elements that are not logically part of an element's content. Inclusion exceptions allow independent element types to occur anywhere within the instance of the element. Examples are footnotes or tables. The notation for exclusion exceptions is `-(element_names)`. The notation for inclusion exceptions is `+(element_names)`. The notation for an exception consisting of both exclusion and inclusion exceptions is `-(element_names) +(element_names)`. Exclusions always take precedence over inclusions and cannot be overridden by them. For example, exceptions for the element "ftnote" can be applied in the following manner:

```
<!ELEMENT ftnote - - (%list; | para) -(ftnote | table)>
<!ELEMENT ftnote - - (%list; | para) +(symbol)>
```

The above examples exclude footnotes and tables from footnotes but allows symbols anywhere in footnotes. These exceptions can also be applied in one declaration as demonstrated in the following example:

## MIL-HDBK-28001

<!ELEMENT ftnote - - (%list; | para) -(ftnote | table) +(symbol)>

6.1.3.5 Attribute list declaration. The last type of declaration is the attribute list declaration.

<!ATTLIST element\_name attribute\_definition\_list>

Attributes are additional data to be provided about element types.

6.1.3.5.1 Attribute definitions. Each list of attribute definitions is associated with one or more element types while an element type may have associated with it at most one attribute definition list. For example, in the following sample application of this specification, security classification is one of the attributes of many types of elements. A simplified version follows:

```
<!ATTLIST doc      security      (u | c | s | ts)      "u"
                  service      (af | army | navy | dla | cg | mc)  #REQUIRED
                  change      NUMBER      "0">
```

This example specifies that u (unclassified) is the default if the attribute is not specified for any particular element of that type. The second attribute is service. It lists six possible values and, through the use of the REQUIRED keyword, requires that one of them be specified. If it is not specified, an error will be reported as there is no default. The third attribute is the change level. Its value must be a digit. The default is 0.

6.1.3.5.2 Attribute name. A member of an attribute definition list within an attribute list declaration; it declares an attribute name, specifies the form and SGML-specific aspects of possible values, and specifies the action (such as providing a default value) to be taken if an attribute's value is not specified.

6.1.3.5.3 Declared values. An attribute's declared value can be any of the following keywords:

- a. CDATA
- b. NAME
- c. NAMES
- d. NMTOKEN
- e. NMTOKENS
- f. NUMBER
- g. NUMBERS
- h. NUTOKEN

**MIL-HDBK-28001**

- i. NUTOKENS
- j. ENTITY
- k. ENTITIES
- l. ID
- m. IDREF
- n. IDREFS
- o. NOTATION

ISO 8879 contains more values that are not used in MIL-PRF-28001 applications.

**6.1.3.5.4 Default values.** An attribute's default value can be a specific value or any one of the following keywords (see 3.2) preceded by the reserved name indicator #:

- a. FIXED
- b. REQUIRED
- c. CURRENT
- d. IMPLIED
- e. CONREF

**6.1.4 Notation declaration.** A notation declaration identifies a data content notation used within the document. This is used in the accompanying application to identify drawings or illustrations which are non-SGML data (NDATA), such as Initial Graphics Exchange Specification (IGES), Computer Graphics Metafile (CGM), Consultative Committee for International Telegraphy and Telephony (CCITT) Group 4, and others. A notation declaration follows the form:

```
<!NOTATION notation_name notation_identifier>
```

**6.1.5 Comments.** An SGML comment consists of text which is not intended to be processed by either a parser or an application and which is delimited by "--". Comments can be embedded in SGML Declarations wherever SGML syntax allows. The following is an example of such an embedded comment:

```
<!ATTLIST pageref pgrefid IDREF -- reference to a pgid -- #IMPLIED>
```

A comment declaration consists of zero or more SGML comments joined together or separated by white space or carriage returns which are immediately preceded by <! and followed by >. The following are valid comment declarations:

## MIL-HDBK-28001

```
<!-->  
<!--text not to be processed-->  
<!--text1-- --text2----text3-- >
```

6.1.6 Modular DTDs. See appendix E for discussion of modular DTDs.

6.2 Example DTD. See appendix A.

## MIL-HDBK-28001

### 7. CREATING THE DOCUMENT INSTANCE

**7.1 Background.** SGML uses generic tags to mark up the structure/content of the document based on the structure and content defined in a DTD. Each tag is used to identify a logical piece of information, such as a title, a paragraph, or a task. The basic way of tagging a document is to surround a piece of information with generic markup tags. The start-tag shows the beginning of the structure/content of the information and the end-tag shows the end. In order to tag the document instance appropriately, the person inserting the tags must be familiar with the DTD and understand the type of document being tagged, especially when content tags are being used. For example, the content model defined in a DTD might specify that a repair procedure must contain a list of necessary tools. The person inserting tags in this document instance must be aware of the requirement for a list of tools and must use the appropriate tags. If inappropriate tags are used, errors will be given during parsing (see 7.4).

Another example of inappropriate tagging would occur if procedural steps are defined in the structure/content of a DTD and the person inserting the tags uses the sequential list tag instead of the step tag. The result may appear to be the same; however, the sequential list tag has been used inappropriately and may not conform to the structure/content defined in the DTD. An error may or may not be given during parsing depending on the content model of the element as defined in the DTD.

Generic markup tags may also be used to identify the structure of a document, such as chapters, chapter titles, sections, and paragraphs. The person inserting the tags must be aware, for example, that a chapter title must be the first element after a chapter tag based on the structure defined in the DTD. If the chapter title tag is not used, an error will be given during parsing.

**7.2 SGML coded text source file (document instance) contents.** According to MIL-STD-1840 an SGML coded text source file, also known as a document instance, is composed of SGML coded, ASCII text files, marked up (tagged) in accordance with MIL-PRF-28001 (see 4.4c).

**7.3 Authoring environments.** Several types of authoring tools are available. One author may use a WYSIWYG type editor, another a simple ASCII text editor. Care should be taken in determining which authoring tool is selected, depending on author preference, ease of use, training required, and cost. The selected authoring tool should handle any DTD after it has been parsed.

Authors should not have to abandon their current way of composing. The ideal environment should include tools such as global search and replace, spell checking, dictionary, and thesaurus usage. Some authors have become accustomed to intuitive cutting and pasting to make the authoring process easier. The use of dynamic external entities should be encouraged to help the author manage information.

**7.3.1 Templates.** A customized template modeled from the DTD should be developed to assist those not trained in DTD development but required to author a document instance. The template can be implemented in a text editor, a WYSIWYG editor, a data base input form, or an SGML authoring/composition system.

## MIL-HDBK-28001

**7.3.2 Entity libraries.** Entity libraries are used to store text that can be used by many others in an enterprise. By allowing others to have access to these libraries much time can be avoided by not re-keying this information.

**7.3.2.1 Entities.** An entity is a separate unit of data, text, graphic, or partial document. An entity may be either general (see 6.1.2.1) or parameter (see 6.1.2.2). Entities whose content is defined external to the DTD and the document instance are called external entities (see 6.1.2.5). By using entities, authors can reduce text entry time and errors. Many enterprises have boilerplate text with reusable content. The contents may be controlled and maintained by the enterprise. This content can be maintained as an SGML external entity in an entity library. To use this entity in a document, the DTD developer or author must reference the entity identifier in the document instance or DTD. If the document will be used outside the enterprise, external entities must be supplied in the MIL-STD-1840 transfer unit (see 4.5).

**7.3.2.2 Public entity libraries (text, file, graphic).** External entities that can be used in multiple documents are called public entities. These are collected and maintained in public entity libraries. These public entities are usually given a unique public identifier by either international, national, or industry standards bodies or a group of users who wish to share the entity among themselves. Others could be sets of entity declarations for specific graphic symbols or specialized document subjects, such as petroleum, airline, or chemical (see 6.1.2.5). Other public entities could be generic graphic symbols (character entities) such as <sup>TM</sup>.

**7.3.2.3 Legacy information.** Legacy data may contain items that could be better used for an entity instead of re-keying that data. Decisions must be made as to whether this legacy data should be changed into an entity or whether an existing external entity can be used without changing the document's intent.

**7.4 Parsing.** A parser is a specialized software program that recognizes SGML markup in a document. A parser that reads a DTD and checks and reports on markup errors is a validating SGML parser. A parser can be built into an SGML editor to prevent incorrect tagging and to check whether a document contains all the required elements.

The document instance should be run against several SGML parsers to check compliance with SGML syntax and its DTD. SGML parsers cannot detect text errors, meaning errors, or incorrect cross-references. Additionally, the SGML parser cannot determine if the document's content was tagged with an appropriate element in the document instance.

**7.5 Tagging philosophies.** Two tagging philosophies are structure and content tagging. The philosophy applied to a particular application will depend on the goals and the applications for which the information is created. The following paragraphs describe the philosophies and when they may be used.

**7.5.1 Structure tagging.** Structure tagging is commonly used to model and encode information according to the structure of a document or class of documents. Structure tagging can be applied when the document structure can be enforced and validated. Structure tagging promotes document consistency, ease of training, ease of revision, etc. because document structures and structure

## MIL-HDBK-28001

tagging rules can be clearly defined and easily standardized. Some examples of structure tagging would be the use of generic element names such as chapter, section, subsection, para, and step1.

**7.5.2 Content tagging.** Content tagging is commonly used to identify document components by the nature of the information contained in the component. Content tagging can be applied when the organization of the document is dependent on the meaning of its contents. Content tagging promotes information reuse and automated information management process application. The contents of content-tagged documents can be supplied by data bases because data base elements can be easily mapped to content tags. Similarly, data bases can be populated with data from content-tagged documents. Some examples of content tags include remove and replace procedure, warning, introduction, checklist, concluding material, and part identifier.

**7.5.3 Combined structure and content tagging.** A document or document class may combine structure and content tagging. This combination can be applied when generic structure tags such as title and section are part of the content model of a content tag. For example, a military specification may state that the body of a document must contain a maintenance chapter, an assembly chapter, and an undetermined number of chapters in that specific order. Elements such as maintenance, assembly, and chapter can be used to provide the content and structure requirements. The element declarations can be written as follows:

```
<!ELEMENT body      - -      (maint, assem, chapter+)>
<!ELEMENT (maint | assem | chapter) - - (title, section+)>
```

This allows the content to be defined explicitly for the maintenance and assembly chapter while still allowing multiple non-content specific chapters to be defined; they all have the same content model.

**7.6 Tagging guidance.** The following paragraphs describe tagging guidance that can be used to prepare documents for DoD.

**7.6.1 Security.** If the content of an element is secret or confidential, this should be indicated with the security attribute. The allowable attribute values for the security attribute are u, c, and s which represent unclassified, confidential, and secret respectively. This attribute is defaulted to unclassified. Security is to be marked at the lowest level element possible so that it is clear exactly what the classified material is.

Each page is to be marked with the highest level of security which appears on that leaf (the front and back of a page). The title of the closest parent is to be marked with the appropriate security marking. As an example, if the paragraph inside of a primary paragraph is marked confidential, then immediately after the paragraph number but before the title itself, the symbol for classified material would appear. This symbol would also appear in the table of contents. To continue with the example, the following tags:

```
<para0><title>Primary Paragraph Title<para secur="c">Content of a confidential
paragraph</para0>
```

would produce the following output:

**MIL-HDBK-28001**

**X.X (C) Primary Paragraph Title.**  
 Content of a confidential paragraph.

where X.X represents the paragraph number. In the table of contents this title would appear as:

**X.X (C) Primary Paragraph Title .....Y-Z**

where Y-Z represents the page number. For complete information about security marking refer to DOD 5200.1-R.

As can be seen in the above example the person creating the document instance need only identify the lowest level element which contains the classified material. The output system is responsible for generating the appropriate marking on the content of other elements, such as title. A title element should only have the security attribute declared as c or s when the title itself is confidential or secret.

The same ideas apply to marking security inside of a table. A table element should declare the security attribute as either c or s only when the entire content of the table is confidential or secret. In most cases only one or several entries will be classified. The tags for these entries should have the security attribute appropriately declared. It will again be up to the output system to properly mark the title of the table as well as any pages on which the classified material appears.

In DTD design, care should be taken to ensure that only those elements which can directly contain classified material be given a security attribute. Elements that contain other elements which may contain classified material but which would not themselves directly contain classified material, should not have a security attribute.

**7.6.2 Space characters.** White space function characters are used in separator strings to make the document markup more readable. The space character as assigned by the concrete syntax has a special function as a separator with SGML declarations and between words. Multiple white spaces are either discarded when the markup is parsed, or they are normalized into a single space so that markup strings can be compared without regard to record boundaries or space.

**7.6.3 Wordspace.** MIL-PRF-28001 appendix B states "The white space between words ... may be adjusted for readability and line justification." In some documents, additional white space may be required at the end of a sentence for esthetics. The insertion of extra space characters into the document instance may not provide this desired result due to the behavior of various composition systems.

**7.6.4 Comment declarations.** Comment declarations (see 6.1.5) may be used within the document instance to provide instructions or reminders. The comment's content will be ignored during parsing and composition.

**7.7 Tag reuse and standardization.** Tag reuse and standardization are encouraged to save DTD development time and tagging time. The reuse of fragments of an existing DTD can save time because the DTD developer and author are able to agree on a predefined set of tags and definitions. The reuse of tags saves time in document analysis by allowing the DTD developer and author to spend

**MIL-HDBK-28001**

more time understanding the document's structure. Less time is used in research and development of new tags. Moreover, the use of familiar tags reduces the time needed by the author to tag the text source file. By using these familiar tags, the author can also reduce content tagging errors.

**7.7.1 DoD efforts to standardize tags.** The tri-service/industry committee has standardized baseline tags which are currently being used for document tagging. The baseline set of tags is being expanded by the services to meet new functional requirements as new document classes are explored. To enable the DoD to require standard tags, the STR is being developed for storage and reuse of these tags.

**7.7.2 New tags.** New tags are developed when tagging requirements can not be satisfied using standards tags. Standardization of new tags is accomplished through the STR tag submission process (see 12.2).

## MIL-HDBK-28001

### 8. MATHPAC GUIDANCE

**8.1. Introduction.** The following guidelines to the use of the mathpac elements and entities are to be used in the SGML tagging of mathematical notation (equations, formulas, etc.) that occurs in documents conforming to MIL-PRF-28001. This mathpac material is based on similar material of the technical report ISO/IEC TR 9573 Information Processing - SGML Support Facilities - Techniques for Using SGML. For the reader's convenience, a data dictionary of the mathpac tags is included as appendix C.

**8.1.1 Formatting of mathematical elements.** The MIL-PRF-28001 OS does not support the specification of formatting characteristics for mathematical elements. When creating an SGML-tagged source file tagged for a specific document or contract, mathematical elements must be handled in one of four ways:

- a. Simple in-line mathematical notation can be formatted as regular text using special characters, superscripts, and subscripts.
- b. Some complex mathematical notation can be generated with an illustration program and included or referenced as graphics in the SGML-tagged source file.
- c. Other complex mathematical notation can be formatted using a specialized mathematical formatting tool where the composed notation is provided in the form of graphics files which are specified in the SGML-tagged file. It is recommended that the source material for these graphics files be provided with the SGML-tagged source file to permit modification of the notation at a later date.
- d. SGML tagging can be employed for marking up simple and complex mathematics. However, since the current state of the Output Specification does not support the association of complex mathematical formatting specifications to arbitrary SGML declaration sets, this option must be evaluated in terms of the practical availability of composition systems that will be able to format such SGML-tagged mathematics, and it should not be assumed that such systems necessarily are or will be widely available.

**8.1.2 General guidance.** Clearly some rudimentary knowledge of both mathematical notation and SGML is required to use the mathpac tags. If necessary the author can assist with the tagging of complicated mathematical notation by answering questions with regard to unfamiliar symbols, their meaning, usage, etc. The author can also assist with the formatting of such notation by advising how it should be presented and arranged on the page.

This is a brief guide to the use of mathpac tags in marking up mathematical notation in documents conforming to MIL-PRF-28001. For additional examples and more detailed information concerning the mathpac tags, the reader should consult the following sources:

- a. The SGML comments preceding the declarations are from the math declaration set of appendix C in MIL-PRF-28001.

**MIL-HDBK-28001**

- b. The explanatory material in 8.1 through 8.6 of ISO/IEC TR 9573, as well as the sample markups in the SGML comments preceding declarations in 8.7 of TR 9573 element and entity definition.

However, the reader should note the following two points with regard to the sample markups provided in TR 9573. TR 9573 sometimes makes use of the shorttag SGML feature which is not allowed in the MIL-PRF-28001 SGML environment. The shorttag feature allows some minimization of notation in markup. Consequently, some TR 9573 mathpac examples will have to be translated into MIL-PRF-28001 SGML.

For example, the markup  $x_{i/2/}$  will not parse using the MIL-PRF-28001 SGML environment. The mark-up will have to be expanded to  $x_{i/2/}$  by expanding the null end tag notation as follows. The / after i and the / after 2 are null end tags (NET). The first / takes the place of  $</sub>$  and the second / takes the place of  $</sup>$ . However, before this can happen each NET must be preceded by a NET-enabling start tag. A NET-enabling start tag is a start tag where the concluding  $>$  is replaced with /.

Also tags such as  $\langle fence\ type=brace\ style=double \rangle \dots$  will have to be rewritten as  $\langle fence\ type="brace"\ style="double" \rangle \dots \langle /fence \rangle$  by delimiting attribute values. The shorttag feature allows delimiters to be omitted when the attribute value consists of name characters.

Moreover, the mathpac declarations in TR 9573 include shortref and their associated usemap declarations. Short references are basically user-defined macros; their use facilitates the markup of complex SGML constructs such as equations and tables. However, short reference usage is not allowed in the MIL-PRF-28001 SGML environment (SHORTREF is NONE on the MIL-PRF-28001 SGML Declaration) since FIPS PUB 152 prohibits it.

**8.2 Formulas.** Mathematical notation even though it does not contain an equal sign is tagged in much the same way as though it were an equation or formula. From now on, the term formula will be used to refer to formulas, equations, and mathematical notations.

At the outset it should be pointed out that  $a = b$ , and other simple mathematical notation used in the examples throughout this guide are used solely for the sake of simplicity in illustrating tag use. When such mathematical notation can be entered directly into the text from the keyboard, clearly there is no need to tag it in terms of mathpac.

All equations, formulas, and mathematical notations must be tagged as either an inline formula or a display formula as described in the following paragraphs.

**8.2.1 Inline formulas.** An inline formula is one that occurs in text. An example of an inline formula is:

... when  $a = b$ , the fraction is not defined.

**MIL-HDBK-28001**

Another example of an inline formula is:

The value of the denominator  $(a - b)$  is significant.

Inline formulas are tagged in terms of  $f$ . These inline formulas are tagged as follows:

... when  $a = b$ , the fraction is not defined.

and

The value of the denominator  $(a - b)$  is significant.

**8.2.2 Display formulas.** A display formula is one that is that is offset from the text: An example of a display formula is:

... the sum  

$$c = a + b$$
  
 is significant.

Another example of a display formula is:

The difference  

$$a - b$$
  
 is negligible.

Display formulas are tagged in terms of  $df$ . These display formulas are tagged as follows:

... the sum  $c = a + b$  is significant.

and

The difference  $a - b$  is negligible.

A group of display formulas is tagged in terms of  $dfg$ . Consider the following example:

The sum and difference  

$$(a + b)$$
  

$$(a - b)$$
  
 are significant and negligible respectively.

This example is tagged as:

The sum and difference  $(a + b)$   
 $(a - b)$  are significant and negligible respectively.

## MIL-HDBK-28001

Note that the markup for display formulas can appear on the same line in the text since the FOSI will assure that the display formula is offset in the formatted text.

**8.3 Tagging various formulas.** The preceding material gives a rough idea of what should be tagged, and what the two types of formulas are. Next consider the tagging of various mathematical constructs. The following examples for the various tagging procedures are admittedly very simple so as to illustrate the point in question.

The markup of certain mathpac tags may contain certain general tags, such as the text and operator tags. The text tags are italic, roman, and ov. The first two text tags refer to the respective type fonts. The ov tag is used to provide embellishments (such as ^, etc.) over tagged material. The operator tags are: mark, markref, break, sup, sub, sum, integral, product, plex, frac, diff, sqrt, root, square, power, pile, matrix, fence, middle, tensor, mfn, box, and vec. The use of the text and operator tags will be explained in the next section.

**8.3.1 Notation.** The following section describes the tagging of various mathematical notations ranging from the simple (subscripts, powers, roots, etc.) to the complex (derivatives, tensors, matrices, etc.).

**8.3.1.1 Superscripts and subscripts.** The inline formula ... if we set  $A_{ij}^2$  above ... is tagged as:

... if we set <f>A<sup>2</sup><sub>  
ij</sub> = 1</f> above ...

This example introduces the use of the sup and sub tags for superscripts and subscripts respectively. Both sub and sup markups may contain text and operator tags. However, not all tagged quantities may contain sup or sub tags. As noted later, a vector (see 8.3.1.9 ) may not have a superscript or a subscript nor may a function specified by an fname tag in mfn markup (see 8.3.1.13).

**8.3.1.2 Fractions.** The fraction  $\frac{a}{b}$  is tagged as:

<f><frac>a<over>b</frac></f>

The numerator appears between the frac and over tags, and the denominator between the over and </frac> tags. The markups of the numerator and denominator may contain text and operator tags. The value of the align attribute on the frac tag specifies the placement of the fraction. If it is not specified, the fraction will be centered.

**8.3.1.3 Powers.** The expression  $(a+b)^3$  is tagged as:

<f><power><degree>3</degree><of>(a + b)</power></f>

## MIL-HDBK-28001

The exponent appears between the power and the of tags and the quantity being raised to that power between the of and </power> tags. The markups of the exponent and the quantity being raised to that power may contain both text and operator tags.

The expression  $(a+b)^2$  may also be tagged as:

```
<f><square>(a + b)</square></f>
```

The quantity being squared appears between the square and </square> tags. The markup of the quantity being squared may contain text and operator tags.

8.3.1.4 Roots. The cube root of a  $\sqrt[3]{a}$  is tagged as:

```
<f><root><degree>3</degree><of>a</root></f>
```

The root appears between the root and of tags and the radicand between the of and </root> tags. The markups of the root and the radicand may contain text and operator tags.

The square root of a  $\sqrt{a}$  is tagged as:

```
<f><sqrt>a</sqrt></f>
```

The radicand appears between the sqrt and </sqrt> tags. The markup of the radicand may contain text and operator tags.

8.3.1.5 Derivatives. The derivative  $\frac{dy}{dx}$  is tagged as:

```
<f><diff>y<by>x</diff></f>
```

The quantity being differentiated appears between the diff and by tags, and the variable of differentiation between the by and </diff> tags. The markups of these two quantities may contain text and operator tags. The value of the type attribute on the diff tag specifies whether the derivative is ordinary or partial. If it is not specified, an ordinary derivative is assumed.

8.3.1.6 Integral. The integral  $\int_a^b y dx$  is tagged as:

```
<f><integral><from>a<to>b<of>
ydx</integral></f>
```

## MIL-HDBK-28001

The lower limit of integration appears between the from and to tags, the upper limit of integration between the to and of tags, and the integrand between the of and `</integral>` tags. The markups of these three quantities may contain text and operator tags.

**8.3.1.7 Summations.** The summation  $\sum_{i=1}^N a_i$  is tagged as:

```
<f><sum><from>i=1<to>N<of>a
<sub>i</sub></sum></f>
```

The lower limit of summation appears between the from and to tags, the upper limit of summation between the to and of tags, and the summand between the of and `</sum>` tags. The markups of these three quantities may contain text and operator tags.

**8.3.1.8 Products.** The treatment of products  $\prod_{i=1}^N a_i$  is analogous to that of summations

except that the product tag is used. It would be tagged as:

```
<f><product><from>i=1<to>N<of>a<sub>i
</sub></product></f>
```

**8.3.1.9 Vectors.** The vector  $V$  is tagged as `<f><vec>V</vec></f>`. Vectors may not have superscripts or subscripts.

**8.3.1.10 General operators.** A general operator is tagged in terms of a plex tag. For example, the union set operation on a collection of  $N$  sets

$$S_1, \bigcup_{i=1}^N S_i,$$

is tagged as:

```
<f><plex>&cup;<from>i=1<to>N<of>S<sub>i</sub></plex></f>
```

where the entity cup of the ISOtech entity set refers to the union operator symbol.

**8.3.1.11 Matrices.** Matrices are tagged in terms of columns, which may be referred to as piles (see 8.3.2.5). The matrix

$$\begin{vmatrix} 2 & 4 \\ 3 & 5 \end{vmatrix}$$

is tagged as:

```
<df><matrix><col>2<above>3</col><col>4<above>5</col></matrix></df>.
```

The markups of elements of a matrix may contain text or operator tags.

## MIL-HDBK-28001

8.3.1.12 Tensors. The markup of tensor notation is a bit complicated. The tensor tag has two attributes. The first attribute *posf* specifies whether the first character of the tensor's suffix is a superscript (*sup*) or subscript (*sub*); its default value is *sup*. The value of the second attribute *suffix* consists of the characters of the suffix in the order they occur in the suffix separated or not separated by a space according to the following rule: if a character in the suffix changes from superscript to subscript or vice versa, then that character is separated from the preceding character by a space at that point in the value of the *suffix* attribute.

Consider the following examples. The tensor  $A_j^{i k}$  is tagged as:

```
<f><tensor suffix="i j k">A</tensor>
</f>
```

Some explanation of this example will clarify the above rule for the *suffix* attribute. The first

character of the suffix  $j^{i k}$  is *i* which is clearly a superscript. Since the default value of the *posf* attribute is *sup*, there is no need to specify the *posf* attribute on the tensor tag. Also the first character of the value of the *suffix* attribute will be *i*. However, the second character of the suffix is *j* which is a subscript. This means that there must be a space between the *i* and *j* characters in the value of the *suffix* attribute. Now the third and last character of the suffix is *k* which is a superscript. This means there must be a space between *j* and *k* characters in the value of the *suffix* attribute. Accordingly, the value of the *suffix* attribute on the tensor tag is *i j k*.

For a second example, consider the tensor  $B_{mno}$ . It is tagged as

```
<f><tensor posf="sub" suffix="mno">B</tensor></f>.
```

In this case since all the characters of the suffix  $mno$  are subscripts, there are no intervening spaces in the *mno* value of the *suffix* attribute and the value of the *posf* attribute must be specified as *sub* on the tensor tag.

8.3.1.13 Functional notation. The *mfn* function allows functions to be tagged in two different ways described below. For more details, the reader is referred to the *mathpac* data dictionary in appendix C.

One way to tag a function is the following. If the name of the function is one of the following: *and*, *antilog*, *arc*, *arccos*, *arcsin*, *arctan*, *arg*, *colog*, *cos*, *cosh*, *cot*, *coth*, *csc*, *ctn*, *deg*, *det*, *dim*, *exp*, *for*, *gcd*, *glb*, *hom*, *if*, *im*, *ker*, *lg*, *lim*, *ln*, *log*, *lub*, *max*, *min*, *mod*, *re*, *sec*, *sin*, *sinh*, *tan*, and *tanh*, then the name of the function is specified as the value of the *type* attribute on the *mfn* tag. For example, the

## MIL-HDBK-28001

functions  $\cos(1 + x)$  and  $\max(x, y, z)$  can be tagged as `<f><mfnc type="cos">1 + x</mfnc></f>` and `<f><mfnc type="max">x, y, z</mfnc></f>` respectively.

Another way to tag a function is the following. The name of the function is specified by `fnc` and of tags. For example, the functions  $\tan(T)$  and  $\log(x, y, z, t)$  are tagged as `<f><mfnc><fnc>tanc</fnc><of>T</mfnc></f>` and `<f><mfnc><fnc>logc</fnc><of>x, y, z, t</mfnc></f>` respectively.

**8.3.2 Presentation and placement.** The following section describes the tagging of various mathematical notations with regard to their formatting (font, character embellishment, arrangement, placement, etc.).

**8.3.2.1 Fonts.** The `roman` and `italic` tags are used to provide their respective type font for those parts of a formula enclosed in these tags. Only alphanumeric text can be enclosed in these tags. The markups `<f><roman>A</roman></f>` and `<f><italic>A</italic></f>` will cause the mathematical notation A to be printed in roman or italic font respectively.

**8.3.2.2 Character embellishment.** The `ov` tag can be used to select twenty-one types of embellishment (dot through bar) to tagged quantities as specified by the `type` attribute on the `ov` tag with respect to the three values (above, below, or mid) of the `pos` (position) attribute and the six values (single, double, triple, dash, dots, or bold) of the `style` attribute on the `ov` tag.

Consider the following markups. The markup `<f><ov>A</ov></f>` will result in the  $\bar{A}$  symbol being formatted since the default values of the `type`, `pos`, and `style` attributes are bar, above, and single respectively. The markup `<f><ov pos="below">A</ov></f>` will result in the  $\underset{\sim}{A}$  symbol being formatted. The markup `<f><ov type="tilde">A</ov></f>` will result in  $\tilde{A}$  being formatted. The markup

`<f><ov type="acute"style="double">A</ov></f>`

will result in the  $A''$  symbol being formatted.

**8.3.2.3 Breaking formulas.** The `break` tag is used to divide formulas. For example, the display formula

$$\begin{array}{l} x + y + z = 25 + \\ X + Y + Z \end{array}$$

is tagged as `<df>x + y + z = 25 + <break>X + Y + Z</df>`.

## MIL-HDBK-28001

8.3.2.4 Aligning formulas. The mark tag and the markref tags are used to align formulas. For example, the system of simultaneous linear equations is tagged as:

$$\begin{array}{rcl} 2X + 43Y & = & 142 \\ 113X + 2Y & = & 26 \end{array}$$

```
<dfg>
<df>2X + 43Y<mark id="here">= 142</df>
<df>113X + 2Y <markref refid="here">= 26</df>
</dfg>
```

The same attribute value (which must begin with a letter) must be specified for the id attribute on the mark tag and the refid attribute on the markref tag.

8.3.2.5 Columns (piles). The column

a

b

c

is tagged as `<f><pile>a<above>b<above>c</pile></f>`. The align attribute of the pile tag specifies the alignment of the pile's elements. These elements may be right or left justified or centered (default).

8.3.2.6 Fences. Let us consider two examples to illustrate the fence concept. First suppose

the fraction  $\frac{3}{4}$  is to be enclosed in braces (curly bracket). The tagging for this would be

`<f><fence type="brace"><frac>3<over>4</frac></fence></f>`. An interval of the form

$(... ]$  is called a half open interval. The half open interval whose left (open) end is  $\frac{3}{4}$  and whose

right (closed) end is  $\frac{5}{6}$  is tagged as:

```
<f><fence open="(" close="]">
<frac>3<over>4</frac>,<frac>5
<over>6</frac></fence></f>
```

## MIL-HDBK-28001

In general the character used to fence off (enclose) a quantity is specified by either the paren (default), bracket, angbrack, brace, bar, or none value of the type attribute on the fence tag. This character can be further specified by the single (default), double, triple, dash, dots, or bold value of the style attribute on the fence tag. If different characters are used to fence off a quantity, the left most enclosing character is specified by the value of the open attribute on the fence tag and the right most enclosing character by the value of the close attribute on the fence tag. For examples of the use of different characters in fencing off a quantity, see the above half open interval markup.

A post (a vertical bar used to indicate choice) is entered using the middle tag and its style attribute similar to fence above. For example,

$$\left( \frac{a}{b} \mid 1 \right)$$

is tagged as:

```
<df><fence><frac>a<over>b</frac><middle>|</middle>1</fence></df>
```

**8.3.2.7 Boxes.** The markup `<f><box>a + b</box></f>` will result in  $a + b$  being enclosed in a box.

**8.4 Detailed examples.** Both of the following markups and their formatted hard copy are provided in 8.6 Complex examples of ISO/IEC TR 9573. The first example is that of a continued fraction. (The "..." indicates that the value of the fraction is the limit of an infinite sequence of converging fractions.)

$$a_0 + \frac{b_1}{a_1 + \frac{b_2}{a_2 + \frac{b_3}{a_3 + \frac{b_4}{a_4 + \dots}}}}$$

This example is tagged as follows:

```
<df>a<sub>0</sub>+<frac>b<sub>1</sub><over>a<sub>1</sub>+<frac>b<sub>2</sub><over>a<sub>2</sub>+<frac>b<sub>3</sub><over>a<sub>3</sub>+<frac>b<sub>4</sub><over>a<sub>4</sub>+...</frac></frac></frac></frac></df>
```

The second example is that of a product of an exponential and two fractions, one of which contains a partial derivative and the other an exponential.

It is tagged as follows:

```
<df>c<sub>&mu;</sub>(t)=<power>-<frac>i<over>h</frac>E<sub>&mu;</sub></sub>t<of>e</power>a<frac><fence><diff type=
```

## MIL-HDBK-28001

$$c_{\mu}(t) = e^{-\frac{i}{h} E_{\mu} t} a \frac{\left(\frac{\partial V}{\partial a}\right)_{\mu_0}}{E_{\mu} - E_0} \frac{e^{\frac{i}{h} (E_{\mu} - E_0) t} - 1}{\frac{i}{h} (E_{\mu} - E_0)}$$

```
"partial">V<by>a</diff></fence><sub>&mu;0</sub><over>E
<sub>&mu;</sub>-E<sub>0</sub></frac><frac><power>
<frac>i<over>h</frac>(E<sub>&mu;</sub>-E<sub>0
</sub>)t<of>e</power>-1<over><frac>i<over>h</frac>
(E<sub>&mu;</sub>-E<sub>0</sub></frac></df>
```

**8.5 Problems.** Two notable problems arise when using the mathpac tags. The first results from the mixed content in the content models of certain widely used mathpac elements. The second is the inability of mathpac tags to mark up certain commonly occurring mathematical notations.

**8.5.1 Mixed content problems.** Three mathpac elements operator, mfn, and middle have mixed content. This means that carriage return and spaces between tags may be considered as data under certain circumstances, instead of being ignored by a parser conforming to ISO 8879. This means that apparently acceptable markups of these elements may not parse.

**8.5.2 Mathpac tagging problems.** Unfortunately the mathpac tagset is incomplete and provides no way to tag some commonly used mathematical notations. The following list is based on the present limited investigation of the mathpac tags. There may well be other such deficiencies.

- a. The mathpac tagset does not support vectors with subscripts or superscripts. There seems to be no way to get around this.
- b. Function names tagged with the fname tag in mfn markups may not have subscripts or superscripts.
- c. There is no way of tagging second and higher order [partial]/[non-partial]/[both partial and non-partial] derivatives, other than tagging them as fractions. For example, the second derivative of y with respect to x must be tagged as:

```
<df><frac>d<sup>2</sup>y<over>dx
<sup>2</sup></frac></df>
```

Moreover, the tagging of such higher order partial derivatives requires the use of the ISOtech part entity to provide the partial differential character.

## MIL-HDBK-28001

## 9. PARTIAL DOCUMENTS

**9.1 Partial documents.** Partial document delivery is used to transmit SGML source data either as an interim deliverable or as an update package for a document that has been previously delivered. Its purpose is to minimize the retransmittal of unchanged data, or to indicate data that is incomplete. Partial document delivery is not intended to address the issues of page integrity or fidelity, nor is it intended to address change pages. However, delivery in this form does not preclude maintaining page integrity for change page delivery. The intent of this methodology is to allow the transmission of portions of a source document so the receiving system can identify the location of the information in the original document and perform the appropriate add, delete, or replace operation. The manner in which this is accomplished and the effect of the change on composition is up to the receiving system and should reflect the requirements called out in the controlling specifications.

**9.1.1 Concepts.** The method of reliable interchange of SGML source data relies on the concept of the document map. The document map is a document instance that contains the high-level element hierarchy of the document. It contains little actual text, other than perhaps the content of the identifying information. Elements of the hierarchy that are not being transmitted contain a special attribute to indicate that they are not being transmitted. Elements that do contain text to be updated are represented by references to external entities that contain the actual changed elements. Other attributes of the elements indicate the operation of add, replace, or delete. In order to facilitate the automated update of the information, the sender and receiver should agree to the elements that should be modularized as separate entities. It is best if this agreement is made part of the contract, but it is not mandatory. Using this methodology, a receiving system should be able to identify all information and map it to data in its system.

The use of the document map accomplishes two objectives: 1) the file may be parsed and validated for conformance to the required DTD without expecting errors, and 2) the map is used as a locator for the changed information in the original document. The file(s) containing the document map is (are together) known as the Transmittal Master for the document.

**9.1.2 Transmittal master.** The transmittal master file is a map of the document hierarchy. All significant elements of the DTD, as seen in the following example, have a content reference attribute stub with the sole declared value of STUB and a default value of #CONREF. When the stub attribute is specified on a tag for one of these elements, that element is considered to have EMPTY declared content with regard to that tag. No data is allowed as content to an element that has a CONREF attribute specified. For example, specifying the value for the attribute stub (since stub is a CONREF attribute) would indicate this portion of the document is unchanged, and is not being transmitted. Use the CONREF attribute at the highest level of the hierarchy at which no data is to be transmitted. The usual content of the element can be omitted and result in no parsing errors. In fact, one must not include any data in that hierarchy or a validation error will result. Only elements of the hierarchy which are required to locate a specific instance of an element, or which are required for the transmittal master to parse against the DTD, need be included in the transmittal master.

## MIL-HDBK-28001

This technique can be best described with a simple example. Suppose we take the memo document type declaration used as an example in section 6.1 and modify it to enable the delivery of partial documents by providing the para element with the stub content reference attribute as follows:

```
<!ATTLIST para stub (STUB) #CONREF >
```

The modified document type declaration is then:

```
<!--The document type declaration for a simple office memo modified for
partial document delivery-->
```

```
<!DOCTYPE memo (
<!ENTITY % mixed "#PCDATA | list" >
<!ELEMENT memo - - (header, body, sign) >
<!ATTLIST memo route ( internal | public) #IMPLIED
                release (confid | general) "general">
<!ELEMENT header - - (to+ & from & subj & date) >
<!ELEMENT (to | from | subj | date) - - (#PCDATA) >
<!ATTLIST to distrib (group | unique) "group" >
<!ATTLIST from origin (person | depart) "person" >
<!ATTLIST date day (mon | tue | wed | thu | fri | sat | sun ) "mon" >
<!ELEMENT body - - (para+) >
<!ELEMENT para - o (%mixed;)* >
<!ATTLIST para stub (STUB) #CONREF >
<!ELEMENT list - - (item+) >
<!ELEMENT item - o (%mixed;)* >
<!ELEMENT sign - - (name, title?, phonext) >
<!ELEMENT (name | title | phonext) - o (#PCDATA) >
]>
```

Let us take the case when we are delivering the text of the third paragraph of a multi-paragraph memo document. In the memo instance the para tags for those paragraphs that are not being delivered are stubbed, i.e., the stub attribute is specified on these para tags and these para tags have no content. Since the third paragraph is being delivered, the para tag for the third paragraph has content and is not stubbed. Its content will be provided by referencing an external general entity.

A transmittal master for our example as well as a file containing the replacement text for the external general entity are provided below. We will assume that "-//USA-DOD//DTD EXAMPLE HDBK//EN" is the formal public identifier for the above modified memo DTD, that the name of the entity providing the text of the third paragraph is para.3, and that memopara.3 file contains the replacement text of the para.3 entity.

```
<!-- This is the Transmittal Master -->
<!DOCTYPE memo PUBLIC "-//USA-DOD//DTD EXAMPLE HDBK//EN"
[<!ENTITY para.3 SYSTEM "memopara.3">]>
<memo route="internal" release="confid">
<header>
<to distrib="unique">Major Smith
<from origin="depart">General Jones
<subj>Press Release - CALS Standards
<date>16 August 1994
</header>
<body>
```

**MIL-HDBK-28001**

```

<!--This paragraph is NOT being delivered.      -->
<para stub="STUB">

<!--This paragraph is NOT being delivered.      -->
<para stub="STUB">

<!--This paragraph IS being delivered.          -->
<para>
&para.3;
</body>
<sign>
<name>Elmer Jones
<title>General
<phonext>23456
</sign>
</memo>

```

The following is an example of suitable replaceable text for the para.3 entity:

```

<!-- Replacement text for the para.3 entity -->

The following is for immediate release.  The DoD today announced that
MIL-HDBK-28001 is ready for Red Team review and will be forwarded within the
next two weeks.  The following organizations will be included in the review:

<list>
<item>the Air Force
<item>the Navy
<item>the Army
<item>JCALS
</list>

```

The above replacement text is contained in the memopara.3 file.

Stubbed elements have the following functions. First, they allow the transmittal master to be parsed and thus validate the material being delivered. Secondly, they allow the receiving system to properly situate the material being delivered with regard to previously delivered material. The example shown assumes interchange of data at the paragraph level.

References to graphics in the transmitted files must be resolved by defining the appropriate entities in the declaration subset. The transmittal master may include only the graphic entity declarations for graphics submitted as changed parts of the document.

The minimum requirements for partial delivery of SGML documents are:

- a. Delivery of the transmittal master file at each delivery. The complete content of module elements shall only be transmitted for those modules containing changes; all other module elements shall be nulled with stub attributes.
- b. Placement on transmittal media in accordance with MIL-STD-1840 applicable conventions.

**MIL-HDBK-28001****10. CREATING AND USING FORMATTING OUTPUT SPECIFICATION INSTANCES (FOSI)**

**10.1 Scope.** This section describes a method for interchanging formatting requirements for military technical documents whose source files are tagged according to DTDs developed in accordance with MIL-PRF-28001. Adherence to the rules described in this section allows for divergent receiving processing systems to unambiguously interpret the style and formatting intent of the sending system, such that by combining the document instance with the appropriate FOSI, the resulting publication will preserve the information content of the original with similar presentation.

The following functionality is not provided for in this version of the OS:

- a. Formatting mathematical elements.
- b. Using fragments of FOSIs to be merged with the baseline FOSIs.
- c. Supporting all functions necessary for producing change packages.
- d. Supporting all possible security classification markings.

**10.2 Statement of purpose, premises.** This section describes the rules for creating all FOSIs to be included in or delivered in accordance with MIL-PRF-28001, as well as the interchange format to be used. Throughout this document the terms Output Specification and OS refer to this section, and the terms Formatting Output Specification Instance and FOSI refer to a particular application of the rules and methods such as MIL-M-38784. The military functional specification, such as MIL-M-38784 determines the actual requirements. A specific DTD interprets the content and structural requirements of a particular functional specification, and a specific FOSI interprets the style and formatting requirements of the functional specification. The designer of the DTD and FOSI is responsible for assuring that they convey a consistent, unambiguous, and complete description of the pertinent areas of logical tagging and output presentation from the military specification. If a particular contract calls for an exception, or variant interpretation of a functional specification, then an unambiguous FOSI must be created.

**10.3 Media.** The media is the physical form on which information will be output, such as paper or electronic display screen. Although mainly concerned with paper media, this handbook does allow some limited electronic presentation capabilities. Future versions will deal with electronic display. Throughout this handbook the term page can be interpreted as a page of paper, an electronic screen, or the area inside the frame of a window on an electronic display.

**10.4 Design goals.** The overall goal is to allow for the interchange of style and formatting information between all types of publishing systems. This includes current batch and WYSIWYG systems, as well as future systems incorporating newer technology. This is accomplished by the interchange of style information, using the semantics described, to be used as input to the formatting system, whether human or computer.

**10.5 Page integrity and page fidelity.** Page integrity is defined to mean the ability to preserve the exact same information on each page in a document as it is exchanged between systems.

## MIL-HDBK-28001

This does not mean that the information will be presented exactly the same way, but only that it will appear between the same page boundaries. Page fidelity is defined to mean the ability to preserve the exact presentation characteristics *in addition to the same information on pages exchanged between systems*. Preserving page fidelity between systems is not technically feasible with current technology. Preserving page integrity may be possible to some degree with today's technology, but has the following consequences:

- a. The pages may look different.
- b. Additional cost may be associated with the effort to ensure that the pages are identical in content and presented in a readable fashion.
- c. In order to preserve page integrity as much as possible and still adhere to the required FOSI, the author of the FOSI needs to be sure to include enough flexibility in the specification of characteristic values to allow pages to appear differently.
- d. Complete page integrity may not be compatible with arbitrary SGML documents and/or certain FOSI specifications.

A FOSI cannot guarantee page integrity. Page integrity requirements should be carefully reviewed in the context of applicability, usability, and cost.

**10.6 Machine parseability.** FOSIs prepared in accordance with the OS DTD are machine parseable. Machine parseability is defined here to include the ability for a machine to automatically verify that a FOSI contains all the required characteristic values and is presented in the correct syntax.

**10.7 The Output Specification (OS) goal.** The goal of the OS is to allow for the interchange of style and formatting information between all types of publishing systems. It describes a method for interchanging formatting requirements for documents whose source files are tagged according to DTDs. This handbook contains specific guidance to create MIL-PRF-28001 compliant FOSIs.

**10.8 The Output Specification Document Type Definition (OS DTD).** The OS DTD describes the rules which must be followed in order to develop a FOSI. The FOSI specifies layout characteristics for page models, and style characteristics for graphics, tables, and all other elements.

**10.9 Organization of the Output Specification Document Type Definition (OS DTD) (categories of composition characteristics).** The categories of format and style characteristics are represented in the OS DTD as elements. Individual characteristics are represented as attributes on those elements.

### 10.10 Formatting Output Specification Instance (FOSI).

**10.10.1 Definition of a FOSI.** A FOSI is the set of characteristics and values chosen from the OS DTD to represent the formatting requirements for a particular type of document.

## MIL-HDBK-28001

Characteristics are descriptions of the format of a document, rather than commands that tell a formatting system what to do.

**10.10.2 When to use a FOSI.** A FOSI is intended to specify how a particular group of documents should be formatted. The FOSI for that class of documents should be used any time data needs to be presented. If the same information needs to be used for another purpose (it falls under a different class of documents) a different FOSI would be used to present the data in the appropriate format.

**10.10.3 Why use a FOSI.** The FOSI is used to accomplish the goal of interchange of formatting information across different processing systems.

**10.11 FOSI development guidance.** A FOSI author must have a background in typographic design and a working knowledge of SGML. The most important qualification, however, is an intimate familiarity with the requirements of the formatting and style specifications for the class of documents that are to be represented through the FOSI.

The following paragraphs describe a typical approach to creating or modifying a FOSI. Within each step of the approach, techniques for applying specific OS concepts are discussed. The novice FOSI author may find it useful to read this entire section in order to get an idea of the topics discussed.

Throughout this section, the characteristic names are used as they appear in MIL-PRF-28001. To help the reader relate constructs and characteristics to their actual encoding specified in MIL-PRF-28001, the names appearing in the DTD may also follow in parenthesis (generally shortened versions and uncapitalized). In referring to values of characteristics of the type toggle, the phrases turned on and turned off are used to mean non-zero and zero, respectively.

**10.11.1 Understanding the requirements.** The first step in creating a FOSI is understanding the requirements for the content/structure and formatting of the documents to be interchanged (see 5.1). The information about the content/structure of the document should already be rigorously described in a DTD. The DTD defines the element types, the possible content/structure the document can have using these element types, and the attributes that can be associated with each element type. Ideally, there is supporting documentation to describe the meaning and usage of each element type and its attributes, although if such documentation is not available, the burden may fall upon the FOSI author to determine this information.

Formatting information may be contained in a single functional specification or may appear in a combination of specifications or other documents. Again, the burden may fall upon the FOSI author to determine how the formatting information in the specifications relates to the element types defined in the DTD. In addition, the FOSI author must identify all relevant formatting information in the specifications that must be included in the FOSI.

There may be cases where an organization uses formatting practices in addition to formatting guidelines in the functional specifications. These common practices have been indirectly approved because of acceptance of delivered documents using these practices. An example is the common use of kerning in documents conforming to MIL-M-38784, where kerning is not mentioned in the

## MIL-HDBK-28001

specification (see 10.11.12.6). The FOSI author must be aware of these common practices, typically documented in the organization's style guidelines, so that all the relevant formatting information is included in the FOSI.

**10.11.2 Understanding OS concepts.** The FOSI author must be aware of some underlying principles of the OS in order to correctly communicate the formatting information. Following is a discussion of these basic principles.

**10.11.2.1 Purpose of a FOSI.** A FOSI is intended to specify, in general, how a particular class of documents should be formatted. It does not specify with precise fidelity how any particular document was actually formatted; this level of precision is not required and is beyond the purpose of a FOSI (see MIL-PRF-28001).

Characteristics are, in general, descriptions of the format of a document, rather than commands that tell a formatting system what to do. For example, if a FOSI has a value of 10 for the font size (size) characteristic for element type A, this should be interpreted as: however your formatting system works, make sure that font size 10 is used to process element A. The FOSI should not be interpreted as saying; when you see a start-tag for element A call a command that changes the font size and give it a value of 10. This is a subtle, but extremely important, distinction. The first interpretation allows any system (including a human) to create the desired end result, while the second interpretation allows for only systems with a specific command language to easily create the desired result. In this way, the OS does not presume to direct how a formatting system should behave in order to accomplish the desired result.

**10.11.2.2 Identification and treatment of source data.** The basic unit of data within the source document identified within a FOSI is an element (qualified by its context and occurrence). Additionally, attribute values associated with the element can be identified. Once identified, an element is treated as whole. The characteristics associated with the element through the FOSI apply to all the content of that element. There is no notion of start-tag and end-tag processing. In general, it is safe to think of the characteristics as going into effect as soon as processing of the element's content begins. Some characteristics, however, are designed to take effect after the element content has been processed and are so identified in this specification. An example is the end line (endl) characteristic. In addition, some characteristics specify whether other characteristics take effect before or after the element content is processed, for example, the placement (placemnt) characteristic of the puttext, putgraph, and usetext categories.

**10.11.2.3 Cognizance of source DTD.** Every relevant element and attribute in the source DTD should have an entry in the FOSI describing how it is to be formatted. In processing a FOSI, there should be no assumptions made about the source data. For example, an element type of fnote cannot be assumed to be a footnote. It must be identified as a footnote by positioning its description in the proper structure of the FOSI (the footnote description (ftndesc)). A footnote can have the element type xyz in the source DTD, but as long as it is described in the ftndesc of the FOSI, it will be treated as a footnote. Similarly, all attributes that affect formatting must be identified in the FOSI.

**10.11.2.4 Significant record ends.** Although the SGML standard (ISO 8879) carefully defines which record ends (carriage returns or line breaks) in the input source file are significant and

## MIL-HDBK-28001

which are to be ignored by the parser, it does not prescribe what the application should do with significant record ends.

Formatting applications in compliance with MIL-PRF-28001 should normally treat a significant record end as a space character; furthermore, consecutive multi-space characters (including record ends being treated as space characters) should normally be treated for the purpose of composition as a single space character. The exception is in the case of an Element in Context (e-i-c) with asis quadding. In this case, each space affects formatting and each significant record end causes a line break during formatting (consecutive record ends should produce multiple line breaks, i.e., produce a blank line in the composed output). In asis mode, it is an input error if the content given between two record ends will not fit on one output line. What happens in this case is left to the output system.

### 10.11.3 Organizing and documenting a FOSI.

**10.11.3.1 Specifying elements-in-context (e-i-c).** In a FOSI, characteristics must be specified for each element in every context in which a formatting system needs to treat the element differently. That is, before characteristics can be attached to elements, the elements have to be completely specified. This is accomplished by providing the following information:

Generic Identifier	This is a unique name that identifies an element. (It may exist in the source DTD or be defined as a pseudo-element.)
Context	The context characteristic gives the lineage of the GI. This specifies a context in which the element may appear (see 10.11.12.1).
Occurrence	The order of appearance of this e-i-c in relation to like elements. All occurrences of an e-i-c within its parent, regardless of intervening elements of a different type, are considered part of one e-i-c group. For example, when a list contains item(1), item(2), note(1), and item(3), item(1) is the first item and item(3) is the last item (see 10.11.12.1).

The following is an example of specifying an e-i-c:

```
<e-i-c gi = "title" context = "figure">
```

**10.11.3.1.1 Technique - e-i-c order in the FOSI.** The order of the e-i-c within the style, table, graphics, or footnote descriptions have no bearing on how the element is formatted. Some sensible ordering scheme is useful, however, for human readability and access. Ordering the e-i-cs to match the order of element types in the source DTD may be a good choice, as the reader of the FOSI is most likely familiar with the source DTD. Typically, the source DTD reflects the content/structure of the document and provides a progression from major divisions (system, chapter, subsystem, section, etc.), through data elements (test step, paragraph, etc.) down to inline-type elements (emphasis, part, etc.). If the source DTD does not provide this type of ordering, it may be useful to group FOSI entries into these types of functional areas as they may share common types of descriptions.

**MIL-HDBK-28001**

**10.11.3.2 Technique - documenting a FOSI.** It is good practice to include comments within a FOSI to explain techniques used. Comments cannot be used in place of encoding information through characteristics, but they can help the reader understand the intent of a particular encoding. In FOSIs, comments must be provided in the form of comment declarations (see 6.1.5). Comment declarations may be placed in or between element contents in FOSIs. Comments may not be inserted in FOSI tags and the tags of other instances as they may be in the declarations of a DTD.

**10.11.4 Setting up the resource description.** The resource description (rsrctdesc) gives document-wide hyphenation rules' (hyphrule), as well as descriptions of character fills (charfill), counters (counter), strings (stringdecl), and floats (floatloc) that will be used throughout the FOSI. The hyphenation rule category provides for setting various parameters for the hyphenation process that will be used throughout the document. The character fill category provides for describing literals that can be used to fill a space horizontally or vertically (see 10.11.12.19). The counter construct is used in the resource description to specify the properties of a counter that will be associated with one or more elements in context (e-i-cs) using the enumeration (enumerat) category (see 10.11.12.20). The string construct is used in the resource description to specify the properties of a text variable that will be associated with one or more e-i-cs using the savetext or usetext characteristic (see 10.11.12.24 and 10.11.12.25). The float construct is used in the resource description to specify that an e-i-c's content should float to some other place in the output instance than the next available location in the flowing text area.

**10.11.5 Setting up the security description.** In the security description (secdesc), the strings are established to be automatically generated for the security text identified in the header and footer. To set up these strings, the possible values for the security attribute in the source DTD must be known. First, identify the name of the attribute in the source that indicates security levels with the attspec characteristic. Then, through secorder indicate the priority of its values. This priority is used when computing the value that is to appear in the header or footer. Then, set up the string that is to appear for each value. Style and positioning characteristics are specified for the string through the sectext portion of the header and footer specification.

**10.11.6 Setting up page models.** A description (pagedesc) of how the pages are to look (the page model) can be set up independently of the content that goes on them. It describes the placement and relationship of the areas in which the content is to be placed. The FOSI author must analyze the formatting specifications for page layout to determine the sizes of these areas and specify the characteristics that control how and when these areas are created on the page.

**10.11.6.1 Technique - using page sets.** Page sets (pageset) provide the means to specify automatic relationships between recto, verso, recto with blank back, verso with blank front, and automatically generated blank pages. Typically, these relationships are useful when the document is to be printed two-sided in a book style. Each page layout can be described individually, but when only the recto page is described, all pages have the same layout. By turning on the recto/verso toggle, inner and outer margins are automatically reversed on recto and verso pages, usually to allow a wider margin for the bind edge. By turning on the blank page toggle, special actions can be taken when a blank page is automatically generated, for example, when the text for a new chapter starts on a recto page and the text for the previous chapter ends on a recto page.

## MIL-HDBK-28001

**10.11.6.2 Technique - setting page parameters.** The layout areas within the page can be thought of as a set of building blocks that must fit together. Because their relationships are defined in the OS, it is the responsibility of the FOSI author to ensure that the sizes specified are valid. One approach to defining the sizes of layout areas is to first define the width and depth of the page and then define the widths of areas. After defining the left and right margin, widths are next defined. The result of subtracting the left and right margin widths from the page width is used for the width of the flow text (though the flow text width is not explicitly specified). Next, for each different number of columns that are possible in the flow text (flowtext) of a given page specification (pagespec), the individual column widths for that number of columns is specified. If only one column exists within the flowing text area, its width is equal to that of the flowing text area. When more than one column exists, the width of each column is the same. The gutter area width is the area remaining between cumulative column widths and the width allowed for the flowtext areas. If more than two columns exist, the gutter widths are the same. Next, define the depth of the page. After defining the depth of the page, define the top and bottom margins. Next, determine appropriate nominal and maximum depths for headers and footers. Keep in mind that some header and footer information may be determined by the actual content of the page. The difference between the depth of the page and the top and bottom margins, header and footer nominal depths, and the appropriate space above and space below flowtext is used as the depth of the flowtext area.

The change mark (chgmark) area can be thought of as overlaid on the margins; its width and depth are specified independently of other size computations.

**10.11.6.2.1 Layout areas.** Each layout area is comprised of subordinate layout areas, unless it is a terminal layout area, in which case it is either empty or comprised of composition characteristics. A page model also defines related printing information for paper media such as the bind edge and necessary characteristics for one- or two-sided printing. Each layout area is described in the following sections with general rules that apply to all FOSIs. Further, rules that are specified below apply to all allowable page models. The following outline gives the structure of the page model layout areas.

### Page Set

#### Page Area

- Top Margin Area
- Bottom Margin Area
- Left Margin Area
- Right Margin Area
- Header Area
- Footer Area
- Flowing Text Area
  - Column
  - Footnote Area
  - Gutter Area

**10.11.6.2.2 Page set.** A page set (pageset) contains information that applies to a particular page area. A page area may cover recto pages, verso pages, recto pages with blank backs, verso pages

**MIL-HDBK-28001**

with blank fronts, and blank pages, and the information they have in common makes them part of the same page set. There can be any number of page sets for a document. Page sets are referenced by associated unique identifiers. Figure 1 gives a graphical representation of the page model layout area.

The model for a page set allows for one or more recto, verso, blank triples (plus header/footer redefinitions for blank front/back pages). Whenever processing switches to a different page set, the first page produced using that page set will use the appropriate entry (depending if the page is a recto, verso, or blank page) from the first triple of that page set; the second page will use the appropriate entry from the second triple of that page set; and so on with the last triple in the page set being used for all subsequent pages produced. This allows, for example, for opening pages of a page set to use a different page model than subsequent ones. The most common case will be a page set specification with only one recto, verso, blank triple. In the recto and verso elements, there is an optional page resource element in which one can include occurrences of the enumerate and savetext categories. This allows counters and string variables to be managed on a per page basis. For example, one would most likely increment the counter that is being used to count pages (e.g., as all or part of the page folio) in an enumerate in the page resource element of all pages.

**10.11.6.3 Technique - using page references.** Page references are a shortcut for specifying a page model when the only difference from another page model is the header and footer information. Each page model can be assigned a unique identifier through the page id (pgid) attribute. A page reference then refers to this page model through the page id reference (pgidref) attribute and any header and footer information additionally supplied overrides the header and footer information in the referenced page model. If no header and footer information is supplied, the page model uses the header and footer as defined in the referenced page model. For example, if the following had been placed in the pagedesc section of the FOSI,

```
<pagedesc>
  <pagespec pgid="idinfospec">
    <topmarg nomdepth="36pt">
      <botmarg nomdepth="36pt">
        <leftmarg width="72pt">
          <rtmarg width="36pt">
            <header nomdepth="36pt">
              <vquad verquad="top">
                <sectext>
                  <subchars></subchars>
                </sectext>

            <header></header>
            <footer nomdepth="36pt">
              <vquad verquad="top">
                <sectext>
                  <subchars></subchars>
                </sectext>
            </footer>
            <flowtext width="504pt" nomdepth="684pt"
              numcols="1" spabove="18pt" spabelow="6pt">
              <column width="504pt" mindepth="648pt" nomdepth="684pt"
                maxdepth="684pt">
              </flowtext>
            </pagespec>
```

MIL-HDBK-28001

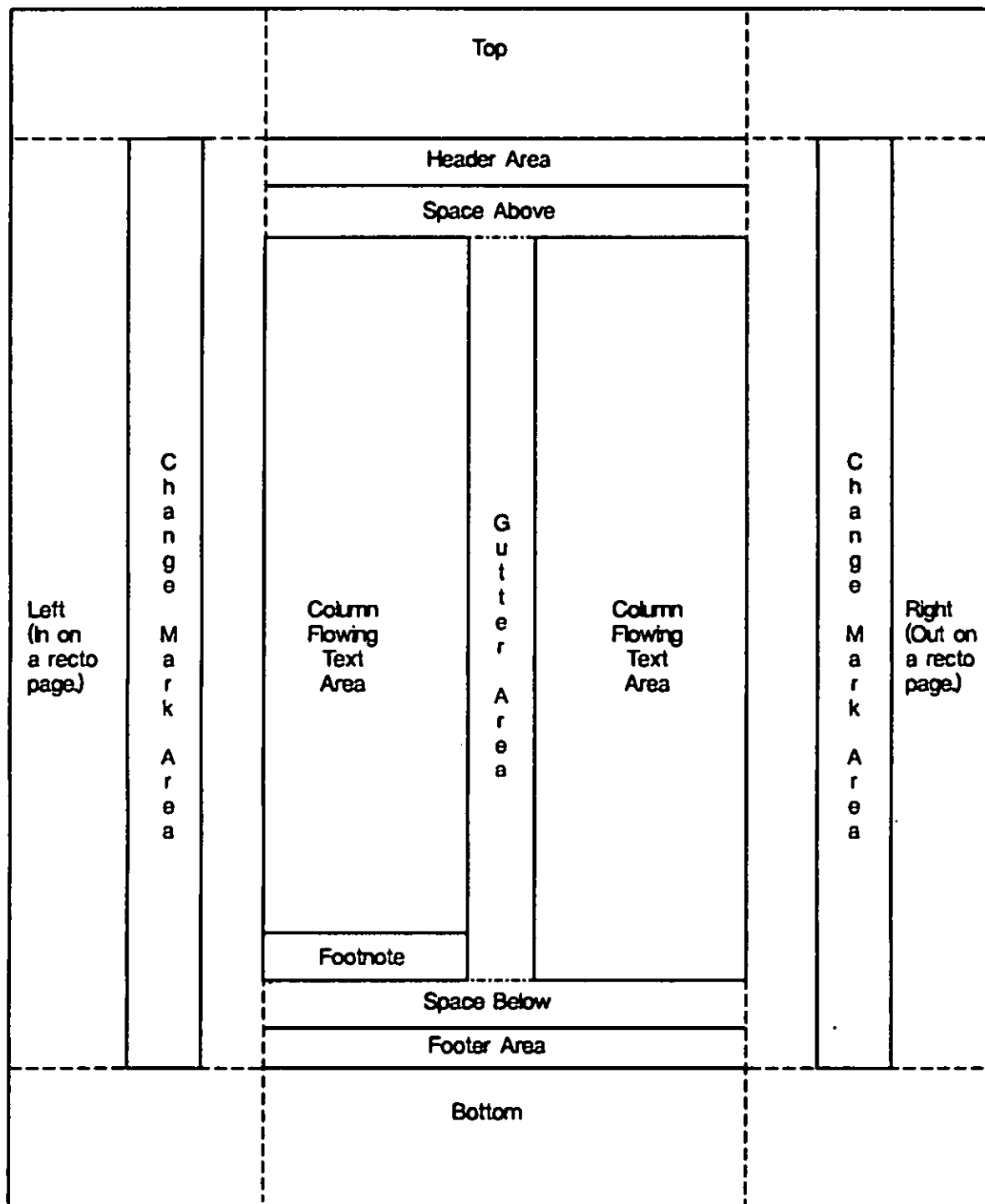


FIGURE 1. Page model layout.

**MIL-HDBK-28001**

to refer to this pagespec, you would invoke it as follows,

```
<pageref pgidref="idinfospec">
```

**10.11.6.4 Technique - setting up headers and footers.** Headers and footers typically contain text that is dependent on document content, such as the technical manual identification number and the chapter title. To specify text dependent on document content, use `usetext`, making sure to specify `savetext` on the appropriate e-i-c. The headers and footers may be specified to allow variable depth.

**10.11.6.5 Positioning techniques.** The style and positioning of text placed in the header and footers is determined by the subcharacteristics (`subchar`) specified for `puttext` and `usetext` and the vertical quadding (`vquad`) additionally specified within the header or footer. For simple cases, where the text is anticipated to be a single line, use the quadding values `right`, `left`, `center`, `in`, and `out` for horizontal positioning and the vertical quadding values of `top`, `middle`, and `bottom` for vertical positioning. This allows for nine positions relative to the header or footer area.

`Vquad` is allowed anywhere within the header and footer content. When it appears within a subcharacteristic list, it applies to the text specified by the `puttext` or `usetext` that is the parent of the list. When it appears outside the subcharacteristic list, it applies to the text specified preceding the `puttext` or `usetext`.

For more precise positioning and handling of multiple-line text, form a box using `prespace` (`presp`) to specify the distance from the top edge of the area, `postspace` (`postsp`) to specify the distance from the bottom edge of the area, `left indent` (`leftind`) to specify the distance from the left edge of the area, and `right indent` (`rightind`) to specify the distance from the right edge of the area. Quadding can then be used to specify how the lines are positioned within the box.

**10.11.6.6 Using security values.** Headers and footers may contain various types of security classifications, which may vary depending on the content of the page or sheet. This text is identified within the headers and footers as security text (`sectext`). This text is special because it is automatically generated based on the specifications within the security description (`secdesc`).

- a. In `secdesc`, use the `attspe` attribute to define the attribute in the source document that identifies the security. Use `secorder` to list the priority, from highest to lowest, of the values to be found in the source document security attribute.
- b. Use `sectoken` as many times as required to give the possible source document attribute values and the security string to be associated with each specific value.
- c. In the header and footer, use `sectext` to set the scope for consideration when determining the highest security. The subcharacteristics specify the style and positioning of the resulting text in the same way as other header and footer text.

**10.11.6.7 Using page numbers.** To generate page numbers, set up enumerate specification in the page resource (`pageres`).

**MIL-HDBK-28001**

**10.11.6.8 Technique - setting up footnote areas.** In setting up the footnote area, choose whether the footnotes will appear at the bottom of each column or span the flowing text area. Because the footnote area grows upward, think of the depth as the height. Specify the maximum amount of space that the footnotes can take up within the column and the fixed amount of space that should always appear between the text and the footnotes. Identify a rule separating the flowing text and footnotes by specifying its length and thickness. Specify whether footnotes can break across pages, and if so, specify what the rule and generated text look like (using subcharacteristics). Specify whether footnotes stay attached to the footer or the flowing text when a floating figure or table appears at the bottom of the page.

**10.11.6.9 Technique - controlling floating elements.** Certain elements, such as tables and figures can be thought of as floating elements because they may appear in the resulting formatted documents in a different position than where they occurred in the source documents. Proper control of floating elements greatly improves useability of information.

The following paragraphs describe the preferred FOSI techniques for controlling floating elements in the following situations:

- a. Floating figure - title at top of figure.
- b. Floating figure - title at bottom of figure.
- c. Multi-sheet figure - title at top of figure.
- d. Inline figure - title at bottom of figure.
- e. Figure on same page with associated text - title at top of figure.
- f. Figure on facing page with associated text - title at bottom of figure.
- g. Figure on same sheet with associated text - title at bottom of figure.
- h. All figures placed in a separate section.

**10.11.6.9.1 Floating figure - title at top of figure.** Figures and tables may be allowed to float to facilitate dense layout so that pages can be filled as completely as possible, thereby minimizing printing costs. Figures and tables are only accessed by the reader when needed.

```
<-- ref:float title_top -->
```

```
<-- Abbreviated instance and FOSI showing how float figures with title at
top of figure are handled.
```

**NOTE:** Only tags that show data flow (inheritance) or are relevant in presenting the concept of this example are shown. There are some tags, required and optional, that have obvious content but are not required for this example. These tags have been omitted for

## MIL-HDBK-28001

clarity. The example, shown below, will not work without all tags properly entered. -->

```

<--instance-->
  <para0>
    <title> TITLE1 .... (See figure <xref xrefid="f3t5"
      xidtype="figure">.)
    <figure id="f3t5"><title> TitleF1 ....
      <graphic boardno="figboard1">
    </figure>
    <para> Text1 ....
    <subpara1>
      <title> Title2 ....
    <step1><para> Text2 ....
    </step1>

    <--FOSI-->

<outspec foscite="float_title_top">
<rsrccdesc>

<floatloc floatid="float"
  floattyp="once"
  maxdepth="9in"
  minspace="0pt"
  nomspace="2pt"
  maxspace="4pt">

<pagedesc>
  <pageset id="floatfigure-titletop">

    <rectopg -- the rectopg is shown here because it is required content in
      the <pageset> tag. The remaining portion of the content model
      is optional.-->

      <pagespec>
        <flowtext topfloat="float"
          botfloat="float">

<styldesc>
  <e-i-c gi="figure">
    <charlist>
      <float flidref="float"
        scope="figure"
        pagetype="forward">
      <e-i-c gi="title"
        context="figure">
        <charlist>
          <suppress sup = "1">
          <savetext
            textid="figtitle"
            conrule="#CONTENT">

<grphdesc>
  <graphenv>
    <graphchars>
      <graphe-i-c>
        <e-i-c gi="graphic">

```

## MIL-HDBK-28001

```

        context="figure">
    <charlist>
        <keep    keep="1">
        <float    flidref="float"
                scope="figure"
                pagetype="forward">
        <usetext  source="figtitle"
                placemnt="before">

```

10.11.6.9.2 Floating figure - title at bottom of figure. This example is essentially the same as the example provided in paragraph 10.11.6.9.1 except that the figure title is placed at the bottom of the figure instead of at the top.

```
<-- ref:float title_bottom -->
```

```
<-- Abbreviated instance and FOSI showing how float figures with title at
bottom of figure are handled.
```

NOTE: Only tags that show data flow (inheritance) or are relevant in presenting the concept of this example are shown. There are some tags, required and optional, that have obvious content but are not required for this example. These tags have been omitted for clarity. The example, shown below, will not work without all tags properly entered. -->

```

<--instance-->
    <para0>
        <title> TITLE1 .... (See figure <xref xrefid="f3t6"
            xidtype="figure">.)
        <figure id="f3t6"><title> TitleF1 ....
        <graphic boardno="figboard1">
        </figure>
        <para> Text1 ....
        <subpara1>
            <title> Title2 ....
            <step1><para> Text2 ....
            </step1>

    <--FOSI-->

```

```
<outspec foscite="float_title_bottom">
```

```
<rsrccdesc>
```

```

<floatloc    floatid="float"
             floattyp="once"
             maxdepth="9in"
             minspace="0pt"
             nomspace="2pt"
             maxspace="4pt">

```

```
<pagedesc>
```

```

    <pageset id="floatfigure-titlebottom">
        <rectopg>
            <pagespec>
                <flowtext    topfloat="float"
                            botfloat="float">

```

## MIL-HDBK-28001

```

<styledesc>
  <e-i-c gi="figure">
    <charlist>
      <float    flidref="float"
              scope="figure"
              pagetype="forward">
    <e-i-c    gi="title"
              context="figure">
      <charlist><suppress sup = "1">
        <savetext    textid = "figtitle"
                    conrule = "#CONTENT">

<grphdesc>
  <graphenv>
    <graphchars>
      <graphe-i-c>
        <e-i-c    gi="graphic"
                  context="figure">
          <charlist>
            <keeps    keep="1">
            <float    flidref="float"
                    scope="figure"
                    pagetype="forward">
            <usetext  source = "figtitle"
                    placemnt = "after">

```

10.11.6.9.3 **Multi-sheet figure - title at top of figure.** Figures may have more than one sheet, in which case each sheet is separately titled and is separately floated. This example shows how examples in paragraphs 10.11.6.9.1 and 10.11.6.9.2 for single sheet figures can be extended to handle multisheet figures.

```
<-- ref:float multi-sheet-->
```

```
<-- Abbreviated instance and FOSI showing how multi-sheet float figures with
title at top of figure are handled.
```

NOTE: Only tags that show data flow (inheritance) or are relevant in presenting the concept of this example are shown. There are some tags, required and optional, that have obvious content but are not required for this example. These tags have been omitted for clarity. The example, shown below, will not work without all tags properly entered. -->

```

<--instance-->
<para0>
<title> TITLE1 .... (See figure <xref xrefid="f3t7"
  xidtype="figure">.)
<figure id="f3t7"><title> TitleF2 ....
<subfigure id=f3t7-1>
<graphic boardno="figboard1">
<subfigure id=f3t7-2>
<graphic boardno = "figboard2">
</figure>
<para> Text1 ....
<subpara1>
<title> Title2 ....
<step1><para> Text2 ....
</step1>

```

## MIL-HDBK-28001

```

<--FOSI-->

<outspec fositite="float_multi_sheet">
  <rsrctdesc>
    <floatloc floatid="float"
              floattyp="once"
              maxdepth="9in"
              minspace="0pt"
              nomspace="2pt"
              maxspace="4pt">

  <pagedesc>
    <pageset id="floatfigure-multisheet">
      <rectopg>
        <pagespec>
          <flowtext topfloat="float"
                    botfloat="float">

  <styldesc>
    <e-i-c gi="figure">
      <charlist>
        <float flidref="float"
              scope="figure"
              pagetype="forward">

    <e-i-c gi="title"
          context="figure">
      <charlist>
        <suppress sup="1">
        <savetext textid="figtitle"
              conrule="#CONTENT">
    <e-i-c gi="subfigure">
      <charlist>

  <grphdesc>
    <graphenv>
      <graphchars>
        <graphe-i-c>
          <e-i-c gi="graphic"
                context="figure">
            <charlist>
              <keeps keep="1">
              <float flidref="float"
                    scope="figure"
                    pagetype="forward"
              <usetext source="figtitle"
                    placement="before">

```

**10.11.6.9.4 Inline figure - title at bottom of figure.** An inline figure or table appears in the output in the same relative location as it is in the source document instance. It cannot be floated past other material to a place which might be more convenient in facilitating dense page layout. The following example shows the method for handling inline figures.

## MIL-HDBK-28001

```
<-- ref:inline -->
```

```
<-- Abbreviated instance and FOSI showing how inline figures with title at
bottom of figure are handled. (In most cases, in-line figures do not
have titles, but the following example shows one for explanatory
purposes.)
```

```
NOTE: Only tags that show data flow (inheritance) or are relevant in
presenting the concept of this example are shown. There are some
tags, required and optional, that have obvious content but are not
required for this example. These tags have been omitted for
clarity. The example, shown below, will not work without all tags
properly entered. -->
```

```
<--instance-->
<para0>
<title> TITLE1 ....
<para> Text1 ....
<subpara1>
<title> Title2 ....
<step1><para> Text2 ....
<figure id="f3t9"><title> TitleF1 ....
<graphic boardno="figboard9">
</step1>
<step1>
<para> Text3 ....
</step1>

<--FOSI-->

<outspec foscite="inline">

<rsrccdesc>

<-- "floatloc" does not apply to in-line figures and tables.-->

<pagedesc>
  <pageset id="inlinefigure-titlebottom">
    <rectopg>
      <pagespec>
        <flowtext>

<styledesc>
  <e-i-c gi="figure"
    context="step1">
    <charlist>
      <float scope="figure"
        pagetype="inline">
  <e-i-c gi="title"
    context="figure step1">
    <charlist>
      <suppress sup = "1">
      <savetext textid = "inlinefigtitle"
        conrule = "#CONTENT">

<grphdesc>
  <graphenv>
    <graphchars>
      <graphe-i-c>
```

## MIL-HDBK-28001

```

<e-i-c    gi="graphic"
          context="inlinefigure">
  <charlist>
    <keeps    keep="1">
    <float    scope="inlinefigure"
              pagetype="inline">
    <usetext  source = "inlinefigtitle"
              placemnt = "after">

```

10.11.6.9.5 Figure on same page with associated text - title at top of figure. Closely related information can be placed on the same or facing pages so the user can read it without flipping pages or searching through a document to find it.

In the source, text has attributes specified that indicate it is associated with figures or tables. For example, a figure might be associated with a particular paragraph with which it should be kept together for optimum readability.

```
<-- ref:same title_top -->
```

```
<-- Abbreviated instance and FOSI showing how figures are handled which
should be placed on the same page with the associated text with figure
title at top of figure. (Note: In most cases, associated figures do
not have titles, but the following example shows one for explanatory
purposes.)
```

NOTE: Only tags that show data flow (inheritance) or are relevant in presenting the concept of this example are shown. There are some tags, required and optional, that have obvious content but are not required for this example. These tags have been omitted for clarity. The example, shown below, will not work without all tags properly entered. -->

```

<--instance-->
<para0>
<title> TITLE1 ....
<figure id="f3t10"><title> TitleF1 ....
<graphic boardno="figboard10">
</figure>
<para> Text1 ....
<subpara1>
<title> Title2 ....
<para>Text2 ....
<step1 assocfig="f3t10"><para> Text3 ....
</step1>
<step1><para>Text4 ....
</step1>
<step1 assocfig= "f3t10"><para> Text5 ....
</step1>

<--FOSI-->

<outspec foscite="assocfig_samepage">

<rsrsrcdesc>

  <floatloc    floatid="same"

```

## MIL-HDBK-28001

```

        floattyp="multiref"
        maxdepth="9in"
        minspace="0pt"
        nomspace="2pt"
        maxspace="4pt">

<pagedesc>
  <pageset id="samepage-titletop">
    <rectopg>
      <pagespec>
        <flowtext botfloat="same">

<styldesc>
  <e-i-c gi="figure">
    <charlist>
      <suppress sup="1">
      <att>
        <fillval      attname="id"
                      fillcat="savetext"
                      fillchar="textid">
        <savetext conrule="#CONTENT">

    <e-i-c gi="title"
      context="figure para0">
      <charlist>
        <suppress sup = "1">
        <savetext textid="figtitle"
          conrule="#CONTENT">

    <e-i-c gi="step1">

      <att>
        <fillval attname="assocfig"
                  fillcat="usetext"
                  fillchar="source">

<grphdesc>
  <graphenv>
    <graphchars>
      <graphe-i-c>
        <e-i-c gi="graphic"
          context="figure para0">
          <charlist>
            <keeps keep="1">
            <float flidref="same"
                  scope="figure"
                  pagetype="same">
            <usetext source="figtitle"
                  placemnt="before">

```

**10.11.6.9.6 Figure on facing page with associated text - title at bottom of figure.** Placing closely related information on facing pages also greatly facilitates information useability. In this case, the text has attributes specified which indicate the associated figures or tables. The text stream is placed on one page and the associated figures and tables are floated to a facing page.

```
<-- ref:facing title_top -->
```

## MIL-HDBK-28001

`<--` Abbreviated instance and FOSI showing how figures are handled which should be placed on a page which faces the page containing the associated text. In this example, the figures are floated to a recto page. The figure title is placed at the bottom of the figure. (Note: In most cases, associated figures do not have titles, but the following example shows one for explanatory purposes.)

NOTE: Only tags that show data flow (inheritance) or are relevant in presenting the concept of this example are shown. There are some tags, required and optional, that have obvious content but are not required for this example. These tags have been omitted for clarity. The example, shown below, will not work without all tags properly entered. `-->`

```

<--instance-->
<para0>
<title> TITLE1 ....
<figure id="f3t11"><title> TitleF1 ....
<graphic boardno="figboard11">
</figure>
<para> Text1 ....
<subpara1>
<title> Title2 ....
<para>Text2 ....
<step1 assocfig="f3t11"><para> Text3 ....
</step1>
<step1><para>Text4 ....
</step1>
<step1 assocfig="f3t11"><para> Text5 ....
</step1>

<--FOSI-->

<outspec fositite="assocfig_facing_page">

<rsrctdesc>

    <floatloc    floatid="facing"
                floattyp="multiref"
                maxdepth="9in"
                minspace="0pt"
                nomspace="2pt"
                maxspace="4pt">

<pagedesc>
    <pageset id="facingpage-titlebottom">
        <rectopg>
            <pagespec>
                <flowtext botfloat="facing">

            <versopg>
                <pagespec>
                    <flowtext>

<styldesc>
    <e-i-c gi="figure">
        <charlist>
            <suppress sup="1">

```

**MIL-HDBK-28001**

```

<att>
  <fillval attname="id"
    fillcat="savetext"
    fillchar="textid">
  <savetext conrule="#CONTENT">

<e-i-c gi="title"
  context="figure para0">
<charlist>
  <suppress sup = "1">
  <savetext textid="figtitle"
    conrule="#CONTENT">

<e-i-c gi="step1">
  <att>
    <fillval attname="assocfig"
      fillcat="usetext"
      fillchar="source">

<grphdesc>
  <graphenv>
    <graphchars>
      <graphe-i-c>
        <e-i-c gi="graphic"
          context="figure para0">
          <charlist>
            <keeps keep="1">
            <float flidref="facing"
              scope="figure"
              pagetype="facing">
            <usetext source="figtitle"
              placemnt="after">

```

**10.11.6.9.7 Figure on same sheet with associated text - title at bottom of figure.** In some instances, related information is placed on the front and back of a single sheet, such as a work card.

```
<-- ref:same title_top -->
```

```
<-- Abbreviated instance and FOSI showing how figures which should be placed
on the same sheet as the associated text, with figure title at bottom of
figure, are handled. The text is placed on the front of the sheet
(rectopage) and the figures are placed on the back of the sheet
(versopage). (Note: In most cases, associated figures do not have
titles, but the following example shows one for explanatory purposes.)
```

NOTE: Only tags that show data flow (inheritance) or are relevant in presenting the concept of this example are shown. There are some tags, required and optional, that have obvious content but are not required for this example. These tags have been omitted for clarity. The example, shown below, will not work without all tags properly entered. -->

```

<--instance-->
<para0>
<title> TITLE1 ....
<figure id="f3t11"><title> TitleF1 ....
<graphic boardno="figboard11">
</figure>

```

## MIL-HDBK-28001

```

    <para> Text1 ....
    <subpara1>
    <title> Title2 ....
    <para>Text2 ....
    <step1 assocfig="f3t11"><para> Text3 ....
    </step1>
    <step1><para>Text4 ....
    </step1>
    <step1 assocfig= "f3t11"><para> Text5 ....
    </step1>

    <--FOSI-->

<outspec fosicite="sheet">

<rsrccdesc>

<floatloc    floatid="sheet"
              floattyp="multiref"
              maxdepth="9in"
              minspace="0pt"
              nomspace="2pt"
              maxspace="4pt">

<pagedesc>
  <pageset id="facingpage-titlebottom">
    <rectopg>
      <pagespec>
        <flowtext>

      <versopg>
        <pagespec>
          <flowtext botfloat="sheet">

<styldesc>
  <e-i-c gi="figure">
    <charlist>
      <suppress sup="1">
      <att>
        <fillval attname="id"
                  fillcat="savetext"
                  fillchar="textid">
        <savetext conrule="#CONTENT">

    <e-i-c    gi="title"
              context="figure para0">
      <charlist>
        <suppress    sup = "1">
        <savetext    textid="figtitle"
                    conrule="#CONTENT">

    <e-i-c    gi="step1"
              context="subpara1">
      <att>
        <fillval attname="assocfig"
                  fillcat="usetext"
                  fillchar="source">

<grphdesc>

```

## MIL-HDBK-28001

```

<graphenv>
  <graphchars>
    <graphe-i-c>
      <e-i-c    gi="graphic"
              context="figure para0">
        <charlist>
          <keeps    keep="1">
          <float    flidref="sheet"
                  scope="figure"
                  pagetype="sheet">
          <usetext  source="figtitle"
                  placemnt="after">

```

10.11.6.9.8 All figures placed in a separate section. Some types of documents place all figures and tables in separate sections at the end of the document or other special location. In this case, the figures or tables are suppressed, saved where they appear in the source instance, appended together, and then used later in the special location.

Figures and tables in the special location are only accessed by the reader when needed.

```
<-- ref:separate_section- title_top -->
```

```
<-- Abbreviated instance and FOSI showing how figures are placed in a
special section with title at top of figure.
```

NOTE: Only tags that show data flow (inheritance) or are relevant in presenting the concept of this example are shown. There are some tags, required and optional, that have obvious content but are not required for this example. These tags have been omitted for clarity. The example, shown below, will not work without all tags properly entered. -->

```

<--instance-->
<para0>
<title> TITLE1 .... (See figure <xref xrefid="f3t15" type="figure">..)
<figure id="f3t15"><title> TitleF1 ....
<graphic boardno="figboard15">
</figure>
<para> Text1 ....
<subpara1>
<title> Title2 .... (See figure<xref xrefid="f3t16" type="figure">..)
<step1><para> Text2 ....
</step1>
<figure id="f3t16"><title> TitleF1 ....
<graphic boardno="figboard16">
<figuresection>

<--FOSI-->

```

```
<outspec fositc="separate">
```

```
<rsrctdesc>
```

```

  <floatloc    floatid="separate"
              floattyp="once"
              maxdepth="9in"
              minspace="0pt"

```

## MIL-HDBK-28001

```

        nomspace="2pt"
        maxspace="4pt">

<pagedesc>
  <pageset id="separate">
    <rectopg>
      <pagspec>
        <flowtext    topfloat="separate"
                      botfloat="separate">

<styldesc>
  <e-i-c gi="figure">
    <charlist>
      <float    flidref="separate"
                scope="figure"
                pagetype="forward">
      <suppress sup=1>
      <savetext textid="separatesection" append="1">

      <e-i-c    gi="title"
                context="figure">
        <charlist>
          <suppress    sup = "1">
          <savetext    textid="figtitle"
                      conrule="#CONTENT">

      <e-i-c gi="figuresection">
        <charlist>
          <usetext source="separatesection">

<grphdesc>
  <graphenv>
    <graphchars>
      <graphe-i-c>
        <e-i-c    gi="graphic"
                  context="figure">
          <charlist>
            <keeps    keep="1">
            <float    flidref="separate"
                      scope="figure"
                      pagetype="forward">
            <usetext source="figtitle"
                      placemnt="before">

```

**10.11.7 Setting up the style description.** After setting up the page models, specify formatting characteristics for every element that may appear in the document. One approach is to look at the formatting specification and determine the overall general requirements. *These requirements can be specified for the document description, providing document-wide defaults.* Then, look for common sets of requirements, specifying them as named environments, for example, for front, body, and rear matter. Next, consider each element type and the requirements that must be specified for each that are different from the document or environment specifications. Determine whether there are unique requirements for element types in specific contexts and specify e-i-cs for each. Finally, determine how attributes in the source affect formatting and further specify the characteristics to handle them.

**10.11.8 Setting up the document default.** Since the values specified for the document description supply the ultimate default values when characteristics are left unspecified, great care

## MIL-HDBK-28001

should be exercised in selecting these values and consideration must be given to the ramification of leaving any specific characteristic unspecified in the document description. For example, determine whether most elements begin on a new line or not. If start line (startln) turned on is the document default, be sure that, for those elements that do not start on a new line, startln is set to turned off. Or, as the default, leave start line turned off and specifically turn it on for each relevant element. In general, choose defaults that will make the FOSI as brief as possible, yet allow flexibility to assign value for differing elements. Certain values, such as that for letterspace or wordspace, may be best optimized by the output system.

**10.11.9 Setting up environments.** Environments are useful when some set of characteristics is common to many elements. Environments can be referred to by any e-i-c and then only the differing characteristics for that e-i-c need to be specified. Care should be exercised in defining environments, using them to make the FOSI briefer and more comprehensible.

**10.11.9.1 Charsubsets - using characteristic list subsets.** If a group of characteristics (e.g., font, leading, wordspace) is used together often, one might want to define a characteristic list subset (charsubset) with the appropriate values and then refer to it by name in a charlist. This subset would merge with the rest of the charlist in the same manner as a charlist in a specval. Charlist subsets are a useful convenience for reducing specification throughout the FOSI. Note that the behavior of charlist subsets is different than that of environments. An environment is a complete set of characteristic assignments used for defaulting unspecified and uninherited characteristics. Charsubsets allow the definition of partial environment that gets merged with other charsubsets, the specifications in the charlist, and any default environment in effect.

**10.11.10 Determining element categories.** The first step in defining a specification for an element type is to determine the applicable functional area under which the associated GI should be defined. All elements described in the style description (styl Desc) are placed in the flowing text area in the order in which they occur in the source document. All elements to be treated as tables must be described in the table description (tab Desc) and those to be treated as graphics in the graphics description (grph Desc). All elements to be treated as footnotes must be described in the footnote description (ftndesc) and are placed in the footnote area.

**10.11.11 Describing elements.** If document and environment defaults have already been established, describing elements is a matter of determining which characteristics differ from those defaults and thus need to be specified.

**10.11.11.1 Technique - grouping elements.** When elements require the same set of characteristics and do not have unique contexts or attribute specifications, they can be grouped in a single FOSI entry. To group elements, specify the names in a list as the value for the GI of the e-i-c.

### 10.11.12 Other techniques.

**10.11.12.1 Using context and occurrence.** Analysis of the source DTD and formatting requirements may reveal that the author of the source DTD has chosen to use a single element type to identify a piece of information, but that element type may be used in many contexts throughout the document. For example, title identifies a title, but when used within a chapter, it identifies a chapter

## MIL-HDBK-28001

title, and when used within a section, it identifies a section title. It must be determined when different formatting characteristics apply and specify an e-i-c for each.

Be sure to analyze all levels of the document, especially areas where structural elements are optional. For example, a chapter may or may not have sections. Whether it actually does can have a profound impact on formatting, for example, numbering of paragraphs may be different. In these cases, it may be necessary to specify many lower-level elements twice: once in the context of section and once in the context of chapter.

Also, look carefully at elements that can appear in many places. For example, para can appear in any level of subparagraph or step. Adjust formatting characteristics for para, such as indents, depending on which subparagraph or step it is found.

```
<e-i-c gi="para" occur="last">
```

10.11.12.2 Using inheritance and defaulting. Inheritance and defaulting are the rules for determining the value of a characteristic when it is left unspecified for an e-i-c. The major consideration for choosing to use defaulting is brevity of the FOSI. Approaches to using defaulting have already been discussed previously.

In considering the use of inheritance, it is important to note that inheritance provides a dramatic reduction in the size of the FOSI for certain kinds of elements. For example, suppose that, for hypertext purposes, the tool element has been defined to identify information about a tool within a paragraph. In printed documents, this information should appear exactly the same as the rest of the paragraph text. By inheriting the paragraph characteristics, the tool text appears the same as the rest of the paragraph text.

Note that inheritance specifies that characteristics are picked up from those in effect for the parent at the point in the document where the e-i-c occurs. The FOSI author can be aware of all the possible places an e-i-c can occur in a document by examining the source DTD, but the actual parent and characteristics in effect can be determined only by looking at the actual source document. Inheritance can provide very important relief from having to specify all possible contexts of an e-i-c.

The material placed into the document by a usertext may contain complete elements from the source document (for example, in-line textual elements contained within #CONTENTS of a savetext string) as well as text. Both the text and the elements should determine their context and inheritance based on the environment into which the saved material is inserted by usertext. That is, the parent for all text and elements within a usertext fragment is the element instance whose e-i-c contains the usertext; the rest of the ancestry of the contents of the usertext would be the ancestry of this parent. This lineage not only determines the content for e-i-c computation but also forms the environment from which the formatting characteristics would be obtained.

```
<font inherit="1" style="sanserif" size="12pt" posture="upright"
weight="medium" smallcap="0" offset="0pt">
```

10.11.12.3 Determining the font. There are two ways to specify the style of the font for text. Use style to specify the general style of the font. Font styles are proportionally-spaced serif

## MIL-HDBK-28001

(serif), proportionally-spaced sans serif (sanserif), monospaced serif (monoser), and monospaced sans serif (monosans). In addition, it is possible to specify the name of an actual font (famname), for example, Century Schoolbook, which the formatting system can optionally select.

To specify characteristics of the font, use posture, weight, and proportionate width. To specify the normal font, use the values upright, medium, and regular, respectively. To specify the size of the font, supply a value for size, typically in points. How the formatting system actually chooses an actual font is based on the algorithms within the system. While the specification describes font as a style modified by characteristics, it is common for font libraries to include different actual fonts for upright and italic versions.

```
<font style="sanserif" size="12pt" posture="upright" weight="medium"
smallcap="0" offset="0pt">
```

**10.11.12.4 Specifying leading.** Leading is directly related to the font size. In this specification, leading is measured from text baseline to text baseline. Therefore the value for leading should be at least as large as the value for the font size, and is typically slightly larger. Always specify leading when specifying a font, as the inherited or defaulted value may not be appropriate. (This is a good place to make use of a charsubset that contains specifications for both the font and leading.)

```
<leading lead="12pt">
```

**10.11.12.5 Controlling hyphenation.** For each e-i-c, specify whether hyphenation should take place or not. If it should, the hyphenation characteristics specified in the document description apply. The only characteristic that can be overridden is the hyphenation zone. It may be desirable to disable hyphenation in text in large type, for example, chapter headings, or in table cells.

```
<hyphrule language="english" nobrkchr="123456789" minleft="3" minpush="3"
clbrkok="0" pgbkok="0">
```

**10.11.12.6 Specifying word spacing, letter spacing, and kerning.** Word spacing, letter spacing, and kerning are generally set up for the document description and are rarely changed for a particular element. Typically, word and letter spacing values are specified with oem spaces. The actual size of an em space is determined by the font in use. This allows word and letter spacing to vary with the font in use.

Kerning is to be used by the composition system to specify letter spacing. Pair kerning specifies an approach whereby kerning pairs are looked up in a kerning table. Track kerning is a methodology for placing the same amount of space between characters in a line. Sector kerning is an algorithmic approach for placing space between characters depending on the characters in the line. Combinations of pair and track kerning (pairtrk) and track and sector kerning (trksectr) may be specified. See letterspace in MIL-PRF-28001.

Specifying different values for minimum, nominal, and maximum gives the formatter freedom to adjust letters and words during justification. Choose values that allow enough latitude yet render the text readable.

## MIL-HDBK-28001

Note that many composition systems choose optimal values for word spacing, letter spacing, and kerning automatically based on the particular fonts in use; therefore, allowing these categories to default to the output system (by omitting an explicit specification everywhere in the FOSI including the docdesc) may produce the best results.

**10.11.12.7 Using indents.** Indents establish margins against which text can be positioned. Specify the indents relative to the column area boundaries, or specify indents relative to the text margins of the parent, for example, in the case of nested paragraphs. One typical use of indents is for hanging indents for lists and steps, where the first line contains a number, for example, to be outdented and the rest of the text appears as a block. To achieve this effect, specify the left margin for the block of text as a left indent (leftind) value on the e-i-c. Specify the first line indent (firstln) on the e-i-c to produce the outdent effect, and then follow the number with a padding specification (see the use of the @ syntax in the savetext construction rule in MIL-PRF-28001) equal to the outdent amount in the usetext used for outputting the number.

```
<indent leftind="0pt" rightind="0pt" firstln="">
<quadding inherit="0" quad="left" lastquad="relative">
```

**10.11.12.8 Boxing.** The drawing of a box.

- a. The integer value for outline and interior screen is specified as a percentage. For example a value of 80 means 80% of black.
- b. The offsets determine how far away from the boxed content area the rules will be drawn (positive values always mean away from the content).
- c. Note that boxing causes the boxed material to become an unbreakable unit (the entire region will behave as if keep were enabled for it); therefore a boxed region will not break over columns. The boxing category is also permitted in subchars; that is, a box can be put around material generated by a puttext or usetext.
- d. If startln="1" is in effect for the text being boxed, siderel of text means the width of the boxed content area is considered to be the full width to the margins currently in effect; siderel of col means the width of the boxed content area is considered to be the full width of the current column (regardless of margins); siderel of content means the width of the boxed content area is considered to be the width taken by the widest extent of text or ruling within the boxed content. When startln="1" is in effect, the leftgap and rightgap characteristics are ignored.
- e. If startln="1" is not in effect for the text being boxed, siderel is treated as equal to content and the box surrounds the in-line text. Note that the boxed text is unbreakable (i.e., cannot break over lines) and must therefore fit on one line in the output. When startln="1" is not in effect, the leftgap and rightgap characteristics give an amount of space to add (inside the box) preceding and following the actual content being boxed. (Values of zero would mean the left and right rule would touch the boxed text; positive values are, in both cases, away from the text.)

**MIL-HDBK-28001**

- f. A bold rule is a rule whose thickness is twice that of a single rule. A double rule is composed of two rules separated by a gap; the overall thickness of the double rule is equal to a bold rule (twice the thickness of a single rule). The thickness of each of the rules should be half the thickness of a single rule, and the thickness of the gap should be the same as the thickness of a single rule.

```
<boxing toffset="40pt"
  loffset="8pt"
  boffset="8pt"
  roffset="8pt"
  trel="top"
  brel="last"
  siderel="text"
  thick="2pt"
  ttype="tsingle"
  inpct="100"
  inclr="iblack"
  rtype="rsingle">
```

**10.11.12.9 Specifying quadding.** The quadding values of left, right, center, and justify represent typical typographic positioning techniques. The values in and out work exactly the same as left and right but leave the actual determination of which side to the formatter based on the bind edge. In and out are most commonly used in headers and footers. Asis instructs the formatter to try to put the same characters found on lines in the source (including spaces) on the same lines in the output (see 10.11.2.4). Typically, this is used when including computer program listings, example terminal screens, pseudo-graphics drawn with characters, and the like.

```
<quadding quad="center" lastquad="relative">
```

**10.11.12.10 Using highlighting.** Scoring, score weight (scorewt), and score offset (scoreoff) are used for underlining, overbars, and strike-throughs. Additionally, a score character (scorechr) can be overlaid to achieve a crossed out effect, for example, with an x. Reverse, color, and screens are used for special effects.

```
<highlt scoring="1" scorewt="1pt" scoreoff="2pt">
```

**10.11.12.11 Specifying change marks.** Specify that either a bar or literal string appears in the change mark area to denote changed text. Specify a bar by indicating a bar thickness (barthick). Specify a literal by providing the string to appear, for example rev1. Font and highlight characteristics can be specified for the string; there is no leading. The change mark literal must fit on one line with no line wrapping.

**10.11.12.12 Specifying prespace and postspace.** Prespace and postspace specify the space before and after an element, respectively. Note that consecutive prespaces and/or postspaces generally combine in such a fashion that only the highest priority spacing (or, in the case of equal priorities, the largest dimension) in a series takes effect. Some planning for prespace and postspace values will ensure that the FOSI reflects the formatting requirements. In general, first specify prespace and postspace where formatting requirements absolutely require it and place an adjacency priority of high with them. Then, analyze the general requirements for spacing between elements and try to specify the space as either prespace or postspace. In this way, conflicts will be avoided. When it is not

**MIL-HDBK-28001**

possible to specify all of the spacing as either prespace or postspace, use priorities that will allow the formatter to make intelligent choices about how to distribute the space.

Prespace and postspace are additive when the value of priority attribute for both elements equals force. Otherwise, resolution is made on the basis of various attribute values in the FOSI (see MIL-PRF-28001 for additional information).

Specifying different values for minimum, nominal, and maximum gives the formatter freedom to adjust prespace and postspace during vertical justification. Choose values that allow enough latitude yet keep within the bounds of the formatting specification.

```
<presp minimum="10pt" nominal="12pt" maximum="14pt">
<postsp minimum="10pt" nominal="12pt" maximum="14pt">
```

**10.11.12.13 Specifying keeps.** Keeps should be used with discretion because it is very easy to specify unrealistic expectations for the formatter. For example, turning on keep for a chapter indicates the entire chapter should be kept on a single page. As the content of the document is unknown, such a requirement may be impossible to fulfill. Turn on keep for elements that fit on a single page. Note that tables and footnotes have special characteristics for controlling breaking across pages and graphics are inherently kept on a single page. A more common formatting requirement is to keep some pieces of adjoining elements together on a page, for example, to keep a title with the first two lines of the following paragraph. This type of requirement can be specified with *keep next*, *keep previous* (*prev*), *widow count* (*widowct*), and *orphan count* (*orphanct*). Set the scope to indicate whether an element may break across columns or pages.

```
<keeps keep="1" scope="page">
```

**10.11.12.14 Specifying vertical justification parameters.** In general, a formatter should attempt to honor all specifications for an element's prespace, postspace, and keeps. However, in performing vertical justification, the formatter may need to make some adjustments in order to balance the text in the columns. Indicate for each e-i-c which values are most important by setting a vertical justification (*vjinfo*) priority on each. If such individual attention is not necessary, then set these priorities for the document description and they will become the default for all e-i-c.

**10.11.12.15 Specifying text breaks (*textbrk*).** Typically a formatting specification explicitly states requirements for starting text on new columns or pages, such as, starting chapters on a new page. This can be specified with *start column* (*startcol*) and *start page* (*startpg*). In addition, the page model (e.g., foldout pages) to be used when starting on a new page can be specified. The OS makes no assumptions about whether elements start on a new line or not. This information must be explicitly specified for each element that starts on a new line. If most elements start on a new line, then turn on *start line* as the document default and explicitly turn it off for those elements that do not start on a new line.

Turning on *end line* is useful for situations where it cannot be predicted what element will follow, but it is known that it must begin on a new line. For example, the paragraph following the title in a primary paragraph is usually run-in, that is, it does not start on a new line. However, if a warning

**MIL-HDBK-28001**

intercedes between the title and the paragraph, the paragraph must start on a new line (or else it would be run-in with the warning). The way to specify this is to turn on end line for the warning.

```
<textbrk startln="0" endln="0">
```

**10.11.12.16 Using spans.** Span is used to specify that text normally placed within a single column in the flowing text area should span all the columns. Note that tables, footnotes, and graphics have special characteristics for specifying their width.

```
<span span="1">
```

**10.11.12.17 Using borders.** Borders that always appear on a particular type of page should be specified in the page specification (pagespec). In addition, the occurrence of certain elements on a page may trigger the appearance of a border, for example, emergency information. Border patterns are specified with a name, which is described in the declaration subset of the FOSI.

**10.11.12.18 Using rules.** Rules are used for inline rules within paragraph text, for example, a signature line. Multiple rules can be specified on a single e-i-c; each specification draws one rule.

**10.11.12.19 Using character fills.** The most common use of character fill (charfill) is for leader dots. Character fill patterns are specified in the resource description to set up how the fill string looks and to assign the string a unique name. To specify where the string appears in the output, use the unique name in the source of a usetext specification.

```
<charfill literal="." spbefore="8pt" spafter="8pt" cfid="dots">
```

**10.11.12.20 Using automatic numbering.** Automatic numbering is typically specified for structural elements such as chapters, sections, paragraphs, and steps, as well as tables, figures, and footnotes. In a FOSI, set up counters that can be referenced by an e-i-c such that the actual value is maintained by the formatter. The counter element in the resource description (rsrdesc) is used to set up each counter. Specify the style of the number when it is printed. Choose a numbering sequence, such as Arabic, or set up a specific ordered sequence. An example of a specific ordered sequence is daggers, double daggers, and so forth, for footnotes. Counters are assigned a unique identifier that can be referenced so that compound numbers can be created. For example, the paragraph number might include the chapter number.

Specify how these counters are incremented and reset using the enumeration (enumerat) category of an e-i-c. The enumeration characteristics only set up how the counter is to be handled when the e-i-c it is associated with is encountered. To specify where the number appears in the output, the unique identifier must be included in the source of a usetext specification. Alternatively, the counter may have already been specified as part of the data in the construction rule of a savetext specification and use of the savetext identifier as the source of the usetext then includes the counter.

```
<counter initial="0" style="ARABIC" enumid="pgcnt">
<enumerat increm="1" enumid="pgcnt">
```

**10.11.12.21 Suppressing text.** Occasionally, text is marked up in the source document that is intended to be used for some purpose other than in the normal text flow. For example, a short title is

## MIL-HDBK-28001

provided for use in the table of contents or running header but does not appear as part of the title. In this case, it is necessary to indicate that the content of the short title does not appear in the normal text flow by turning on the suppress characteristic. Typically, the text is saved with a savetext so that it can be used elsewhere.

**10.11.12.22 Puttext - using generated text.** Often, the formatting specification requires the generation of a standard piece of text with each occurrence of an element, for example, the note heading for a note. This text does not appear in the source document itself, however, by specifying it in the FOSI, it can be ensured that it is consistent throughout the document. Puttext allows for specifying a text string that is to appear before or after the element's content. Subcharacteristics can be used to describe the style and positioning of the text string. (See putgraph).

```
<puttext literal="NOTE" placemnt="after">
```

**10.11.12.23 Putgraph - using pre-defined graphics.** There are cases where graphics appear in the document that are not part of figures, such as the DoD seal. Putgraph allows identification of these graphics and specifies how they appear in running text. In addition, putgraph can be used for some specific generated text instead of usetext, for example, warning heads in 3D boxes. These graphics are to be treated as a single character for positioning by the formatter. Specifying the width and depth of the graphic is a requirement, as the formatter does not have font metric information available to use to position the graphic.

Putgraph is used to place a graphic file into the document instance. To place the graphic, the graphic name (graphname) must be supplied. However, it is recommended that the following attributes also be specified: the width of the graphic, the depth of the graphic, the placement of the graphic (before or after the content of the element), the scale factor, the horizontal scaling, the vertical scaling, the horizontal offset, the vertical offset, and the rotation. See MIL-PRF-28001 for characteristics and values.

**10.11.12.24 Savetext - saving text for multiple uses.** Savetext allows for saving the content of an element for use elsewhere in the document, for example in the header, footer, or table of contents. The content also still appears in the flowing text area in its normal sequence (unless inhibited by use of the suppress category). Savetext can be used to save combinations of other saved text, saved counters, pseudo-elements, and literals. Usetext is used to retrieve the saved text and specify how it is used.

Ordinarily, when a subsequent savetext is done with the same savetext textid as a previous savetext, the subsequent one replaces the previous one. This is the behavior when the append characteristic has the value turned off (its default). However, if the value of the append characteristic is turned on, the value in the conrule of the savetext would be appended to the text already saved to this savetext textid. Consecutive savetexts with the same savetext name and append="turned on" would cause all the saved text to accumulate, and a subsequent usetext would retrieve the entire accumulated set.

The position of the element's content within the saved text is denoted by #CONTENT in the construction rule. If #CONTENT does not appear, no content is saved. When #CONTENT does appear, the entire content of the element, including any children it may have, is saved, so that it can

**MIL-HDBK-28001**

have an effect when the saved content is used. These tags (whether elements or pseudo-elements) will have formatting characteristics associated with them via an e-i-c entry in the FOSI. This makes it possible to save text, and allows it to be formatted differently each time it is used. For example, an element may contain a date separated by / characters, such as 01/02/1990. It may be useful to extract this date into its component parts. One way to do this would be using the start position and end position characteristics. The day could be extracted by saving positions 1 and 2, the month by saving positions 4 and 5 and the year by saving positions 7 through 10. Another method would be to use the delimiter and occurrence characteristics. Using this method, the day could be extracted by saving the text up to the first delimiter (the /), the month could be extracted by saving the text between the first and second delimiters, and the year could be extracted by saving the text after the second delimiter.

```
<savetext textid="hold-area"
  conrule="\The answer is\, 10pt, #CONTENT">
```

**10.11.12.25 Usetext - using saved text.** Usetext specifies how any text saved via savetext is used. Subcharacteristics can be used to specify the style and positioning of the retrieved text. The usage rule specifies any additional information needed to process the text. Note that usage rules are system-dependent.

- a. This category is used to output saved data. Specifically, it uses data saved with the savetext, enumerate, and character fill characteristics. Saved data is identified in the source by unique names assigned to the saved data. A usage rule may be applied to the data before placing the result in the output text stream.
- b. The syntax for the source is the same as that for the savetext construction rule. This allows for specification of multiple pieces of saved data in conjunction with literal text.
- c. When usetext is specified multiple times, or is specified along with ruling, puttext, and putgraph, the generated data in the output stream should appear in the order in which the characteristics are specified for the e-i-c (both before and after the content).
- d. The usetext content is placed immediately before or after the contents of the element (properly mixed with other puttext, putgraph, or usetext generated data); any prespace or postspace of the element or any effects of other characteristics of the element (such as those due to textbrk, keeps, etc.) occur previous to before generated text and subsequent to after generated text.
- e. If any userules are used which are not supported by this specification, they will be specified by a negative integer.
- f. The usage parameter characteristic is used to provide parameters to the specified userule. This characteristic is ignored unless a non-zero userule is specified. The form of this characteristic is the same as that of the puttext literal characteristic. The syntax and semantics of the string specified for this parameter are determined by the specific userule with which it is used.

**MIL-HDBK-28001**

**<usetext source="hold-area" placement="after">**

**10.11.12.25.1 Sort userule - userule #1.** Sorting is performed on data that appears in the source of a usetext. Data shall be specified as records. Within each record will be one and only one key. Both the records and keys shall be scoped by pseudo-elements. Sorting will be performed on the textual data within each key.

**Parameters:** Parameters shall be separated by commas. Values for parameters one through four should be provided. Values for parameters five and six are optional.

- Parameter 1:** One of the following values determines the sort to be performed: alphabetic, numeric, refdes, tonumber, or partnum.
- Parameter 2:** Either an a or a d to indicate ascending or descending.
- Parameter 3:** Name of the pseudo-element that serves as the record indicator.
- Parameter 4:** Name of the pseudo-element that serves as the key field for sorting.
- Parameter 5:** Names of elements, separated by whitespace, whose content is ignored when sorting.
- Parameter 6:** Valid for alphabetic sorts only. Alphabetic sorts are case-insensitive unless this parameter is specified. If a cs appears in this position and an alphabetic sort is specified in parameter 1, the sort will be performed case-sensitive. For case sensitive sorts, the letters A-Z are considered as occurring before the letters a-z.

**Result:** The text that will appear in the output stream after the userule has been processed will be determined as follows:

- a. Records will be reordered in the proper sequence depending on the setting of the first parameter. All sequences of whitespace are treated as a single space. Leading and trailing whitespace is ignored. In addition, keys that evaluate equally when sorted with an ascending sort shall retain their relative order in the sorted list. Keys that evaluate equally when sorted with a descending sort should appear in the sorted list in opposite order as they appeared in the original list.
- b. The sorting orders are as follows:
  - 1. alphabetic - the content of this pseudo-element is sorted in alphabetical order on a character by character basis. The collating sequence shall be ASCII.
  - 2. numeric - the content of this pseudo-element is sorted in numeric order. Numeric is defined as applying to non-negative integers only. In addition, leading 0s shall have no affect on sorting order, (i.e. 02 is equivalent to 2.)

**MIL-HDBK-28001**

3. reference designation (refdes) - a reference designation takes the following form: optional sequence of numbers, followed by an optional group consisting of a sequence of letters followed by a sequence of numbers, which may occur one or more times. Each sequence of letters shall be sorted in alphabetical order, while each sequence of numbers shall be sorted in numeric order. Lower case letters are disallowed. The leftmost sequence is therefore significant. All entries without the initial numeric sequence shall appear before all entries with the initial numeric sequence. An example of a sorted reference designation sequence is:

A1B2  
 A1B3  
 A1B10  
 A1B11  
 A2C2  
 A2C2D3  
 AB1C2D3  
 11A1D2  
 12A1D2  
 12A1D3  
 100P  
 111A

4. TO number (tonumber) - the content of this pseudo-element is sorted as follows:

Each TO number is broken into fields separated by hyphens. The leftmost field is the most significant, while the rightmost field is least significant. A refdes sort shall be applied to each field. In addition, parentheses (the characters "(" and ")") shall be ignored by the sort process. Special characters shall be considered as occurring after A-Z. In addition, all like special characters shall be grouped together, although the order is implementation dependent. For example, AB shall occur before A/, B/, and A#. In addition, the last three sequences can appear as A/, B/, A# or A#, A/, B/. An example list of sorted TO Numbers follows:

00-25-234  
 1-1A-14  
 1-1(B)-14  
 1-1C-14  
 1-1/-14  
 1F-16A-2-36JG-00-1  
 1F-16A-2-36JG-22-1  
 2B-22-33A  
 11B10-ARN15-2  
 11B11-TWP-2-1  
 100B2-C11  
 100#2-C11

**MIL-HDBK-28001**

5. part number (partnum) - part number arrangement shall begin at the extreme left and continue from left to right, one position at a time. For the first character of the part number, the letters A through N and P through Z take precedence over the numerals 0 through 9. (Alphabetic Os are considered numeric zeros. It is valid to make this substitution before sorting, and to retain the substituted value in the subsequent presentations.) For the second and succeeding characters of a part number, precedence is as follows: (1) diagonal (/), (2) period (.), (3) dash (-), (4) letters A through N and P through Z, (5) numerals 0 through 9. Lower case letters are disallowed. The following is a sample arrangement:

AN931-4-13  
 A2460  
 A317  
 A32  
 B/1  
 B.1  
 B-1  
 B12  
 B2  
 S/1  
 1140  
 121873  
 128  
 16.W2  
 16W060  
 32P010-1  
 32P0101  
 39A46

Sample usage:

Consider the following FOSI fragment.

```
<savetext
  textid="sortdata"
  conrule="<record>,
    <item>,\SGML\,</item>,
    <def>,\Standard Generalized Markup Language\,</def>,
  </record>,
  <record>,
    <item>,\FOSI\,</item>,
    <def>,\Formatting Output Specification
      Instance\,</def>,
  </record>,
  <record>,
    <item>,\DTD\,</item>,
    <def>,\Document Type Definition\,</def>,
  </record>,
  <record>,
    <item>,\DTD\,</item>,"
```

**MIL-HDBK-28001**

```

    <def>,\Sometimes wrongly used as Document Type
      Declaration\,</def>,
  </record>">

```

```

<usetext
  source="sortdata"
  userule="1"
  useparm="alphabetic,a,<record>,<item>">

```

After the userule is applied, the equivalent usetext would be:

```

<usetext
  source="<record>,
    <item>,\DTD\,</item>,
    <def>,\Document Type Definition\,</def>,
  </record>,
  <record>,
    <item>,\DTD\,</item>,
    <def>,\Sometimes wrongly used as Document
      Type Declaration\,</def>,
  </record>,
  <record>,
    <item>,\FOSI\,</item>,
    <def>,\Formatting Output Specification
      Instance\,</def>,
  </record>,
  <record>,
    <item>,\SGML\,</item>,
    <def>,\Standard Generalized Markup
      Language\,</def>,
  </record>">

```

Note that the relative order of both DTD entries is preserved in the equivalent usetext.

If case-sensitivity is desired, the following would be a sample usetext:

```

<usetext
  source="sortdata"
  userule="1"
  useparm="alphabetic,a,<record>,<item>,,cs">

```

If the key field contained an element named footnote whose content should be ignored by the sorting process, the following would be a sample usetext:

```

<usetext
  source="sortdata"
  userule="1"
  useparm="alphabetic,a,<record>,<item>,footnote,cs">

```

Note that parameter 5 applies to elements that occur in the source document, not pseudo elements, therefore the syntax is slightly different.

**10.11.12.25.2 System/Subsystem/Subject Number (SSSN) userule - userule #2.** To determine most general system/subsystem/subject (SSSN) number. The most general SSSN number is determined from data contained in the source of the usetext. The source of the usetext shall contain a

**MIL-HDBK-28001**

whitespace separated list of SSSN numbers. A SSSN number is defined as being in the form DD-DD-DD where D is a digit from 0-9.

Parameters: None.

Result: The text that will appear in the output stream after the userule has been processed will be determined as follows:

- a. Consider the first character as being the current character.
- b. Consider the current character of each of the SSSN numbers that appear in the savetext.
  1. If all of the characters are identical, place the character in the output stream. If not, go to step 3.
  2. If all the characters in each SSSN number have been considered, end this algorithm.
  3. If all the characters in each SSSN number have not been considered, consider the next character as being the current character. Go to step b.
- c. If each of the characters are not identical, place a 0 in the output stream. In addition, fill all remaining numeric positions with a 0, and fill all hyphenated positions with hyphens. End this algorithm.

Sample usage:

Consider a savetext string with savetext name of SSSNlist. SSSNlist contains the following: 20-21-30 20-22-30 20-22-32 20-22-43. After applying the above userule (<usetext source="SSSNlist" userule="2">), 20-20-00 would be placed into the output stream.

**10.11.12.26 Interaction of puttext, putgraph, ruling, and usetext.** When more than one puttext, putgraph, ruling, and/or usetext is specified on an e-i-c, the order of their occurrence in the FOSI is significant. That is, the text or graphics should appear in the same order in the output as specified in the FOSI, followed by the content of the element, followed by all text or graphics specified with placement of after.

**10.11.12.27 Using string.** This construct is used to specify the properties of a string that will be associated with one or more e-i-cs using the savetext or usetext category. All variables which are time independent (i.e. have the same value regardless of when they are used), must be specified with their time status attribute set to "1". The scope attribute can be used to indicate an element name after which all uses of the string should be resolved. An example would be the string which contains the collective information to produce a table of contents. Such a string would be specified as time independent, with its scope attribute set to doc.

**MIL-HDBK-28001**

- a. The savetext name follows the same rules as that for the savetext category; the literal follows the same rules as that for the puttext category.
- b. A time status of zero (disabled) indicates that the variable named by the savetext name is time dependent. That is, the value of the variable will be evaluated at the time it is referenced in a savetext construction rule or usetext source. A time status of one (enabled) indicates that the variable is time independent; that is, the value that is used to replace this savetext name everywhere this variable appears is the value this variable would have before the end of its scope. All time independent variables must be declared using the string construct; any variable not declared will be time dependent.
- c. In the case of time independent variables, the contents of the variable named by the savetext name will be resolved after the element specified by the scope characteristic has ended and been completely processed. The element specified by the scope characteristic must be an ancestor of the e-i-c instance in which this instance occurs; if not, the variable will be resolved at the end of the document element.

**10.11.12.28 Using cross references.** Frequently you want to direct the readers attention to a different part of the document with a cross reference. This is done typically with a, see section x, page XX, type reference (though in many style guides, referencing via page number is discouraged).

For purposes of discussion, we'll set up a simple cross reference scheme. The DTD designer will add a topic to each subhead to establish an anchor point for future cross referencing. The DTD segment might look like:

```
<!ELEMENT subhd - o (#PCDATA) >
<!ATTLIST subhd topic ID #REQUIRED>

<!ELEMENT p - o (#PCDATA) +(refer)>

<!ELEMENT refer - o EMPTY >
<!ATTLIST refer to IDREF #REQUIRED>
```

The author would start a section off with:

```
<subhd topic="overview">Getting Started</subhd>
```

Later in the instance, to refer back to another section, the author would insert:

```
For more information, <refer to="overview">.
```

After composition, the page number would be resolved resulting in:

For more information, refer to *Getting Started*, on page 2.

**10.11.12.28.1 How to set up the FOSI.** Two elements need to be looked at in the FOSI, the refer element, and the subhd element. In addition, because the name of the section or subhead is to be in italics, a pseudo element will need to be created. In the example, it will be citation.

**MIL-HDBK-28001**

First, when the subhead name is entered, it will be entered into a variable and indexed by its topic (ID):

```
<e-i-c gi="subhd">
  <charlist>
    <font ...
      <leading ...
      <quadding ...
    </charlist>
  <att>
    <fillval attname="topic" fillcat="savetext" fillchar="textid">
    <charsubset>
      <savetext conrule="\ refer to \,
        <citation>, #CONTENT, </citation>, \, on page \.pagecnt">
    </charsubset>
  </att>
```

Next the pseudo element is created:

```
<e-i-c gi="citation">
  <charlist>
    <font inherit="1" posture="italic">
  </charlist>
</e-i-c>
```

Finally, the refer element is defined, that will pull the correct corresponding phrase from the appropriate variable:

```
<e-i-c gi="refer">
  <charlist>
    <font inherit="1">
  </charlist>
  <att>
    <fillval attname="to" fillcat="usetext" fillchar="source">
  </att>
```

10.11.12.29 Defining leader dots. Leader dots are defined as a character fill in the resource description section at the beginning of the FOSI. As an example:

```
<charfill literal="." padding="6pt" spbefore="8pt" spafter="8pt" cfid="dots">
```

The parameters used are:

literal	Indicates the character(s) used for the leaders.
spbefore	The minimum horizontal space to insert before starting the leaders.
spafter	The minimum horizontal space to insert after the leaders.
padding	Indicates the spacing to insert between successive occurrences of the literal.
cfid	The ID name of the character fill. It will be used to later reference the leader dots.

**MIL-HDBK-28001**

**10.11.12.30 Building a Table of Contents (TOC).** To create a table of contents, two e-i-cs are established. In the following example, an element called toc will be defined in the DTD (as an empty element) and the tag is inserted in the document instance where the table of contents should be placed.

A pseudo element called tocentry will establish the font and point size for each line in the table of contents, as well as starting each entry on a new line.

When a new title for either a chapter or section is encountered, its content as well as the current page number is saved to the variable tocmatter. Append is turned on, so that it is added to tocmatter rather than written over it.

```
<e-i-c gi="toc">
  <charlist>
    <font style="serif" size="12pt" weight="bold">
      <leading lead="12pt">
        <quadding quad="left" lastquad="relative">
          <presp minimum="2in" nominal="2in" maximum="2in">
            <textbrk startpg="recto" pageid="tocpages" newpgmdl="global">
              <usetext source="tocmatter">
            </usetext>
          </charlist>
        </e-i-c>

<e-i-c gi="tocentry">
  <charlist>
    <font style="serif" size="12pt" weight="medium">
      <textbrk startln="1" endln="0">
    </charlist>
  </e-i-c>

<e-i-c gi="title" context="chapter">
  <charlist>
    <font style="sanserif" size="36pt" posture="upright" weight="bold">
      <leading lead="40pt">
        <postsp minimum="16pt" nominal="20pt" maximum="24pt">
          <textbrk startln="0" endln="1">
            <savetext
              textid="tocmatter"
              conrule="<tocentry>, #CONTENT, dots, pagenumstr[bo], </tocentry>"
              append="1">
            </charlist>
          </e-i-c>

<e-i-c gi="title" context="section">
  <charlist>
    <font style="serif" size="18pt" posture="upright" weight="bold">
      <leading lead="22pt">
        <postsp minimum="10pt" nominal="12pt" maximum="14pt">
          <keeps next="1">
            <textbrk startln="0" endln="1">
              <savetext textid="tocmatter"
                conrule="<tocentry>,@5pi, #CONTENT,
                  dots, pagenumstr[BO], </tocentry>"
                append="1">
```

**MIL-HDBK-28001**

```
</charlist>
</e-i-c>
```

Notice that titles from sections will be indented 5 picas in the table of contents. Also, a special pageset is established in the pagedesc section called tocpages.

**10.11.12.31 Pseudo-elements.** Generally the GI of an e-i-c entry in the FOSI identifies an element in the source DTD. However, the GI may also name a pseudo-element that does not conflict with the name of any element in the source DTD. Pseudo-elements can be included in the construction rule of a savetext or the source of a usetext. Whenever a pseudo-element appears in a usetext source or a savetext construction rule, both the start and end tags of the pseudo element must be present. The formatting characteristics would take effect when the pseudo-element is used via the usetext category. Note that a pseudo-element does not appear in the source DTD, but only in the FOSI. The pseudo-element is a construct that can act as a reference to an e-i-c entry in the FOSI. There are a number of important exceptions to the ability of pseudo-elements to act like elements from the document instance. Pseudo-elements have no content model. They do not have attributes and therefore cannot appear in a specval or fillval attloc. Use of specvals and fillvals within pseudo-element e-i-c's is allowed, although only specvals and fillvals with attloc set to #FOSI will be evaluated. This allows combining pseudo elements and elements from the source document in the same e-i-c, for example `<e-i-c gi="warning warning.pseudo">`. Pseudo elements shall not appear in the context specification of an e-i-c. The only valid specification for the occurrence attribute on a pseudo-element e-i-c's is all. Pseudo elements cannot be used to scope floating material or be used as scopes for time independent strings. The suppress category has no meaning in the charlist of a pseudo-element and is ignored if specified. All other charlist categories work with pseudo-elements in the same manner as they do with elements from the document instance. The restrictions on pseudo-elements prevent their being used to create programming constructs. Pseudo-elements simply serve as a way of marking text and specifying the formatting of that text.

**10.11.13 Handling source attributes.**

**10.11.13.1 Technique - identifying source attributes.** Source attributes that affect formatting must be identified in the FOSI with each e-i-c for which that attribute can be specified. An attribute is identified through the attribute name (attname). The value should be the name of the attribute, exactly as it appears in the source DTD. In addition, it is possible to refer to attributes that were specified for ancestors of the e-i-c through the attribute location (attloc). The value of attloc should be the name of an ancestor of the e-i-c and is assumed to be the most immediate ancestor of that element type. For example, when specifying the characteristics for a section title (gi = "title", context = "section"), it may be necessary to know whether the tocentry attribute was set on the section in order to determine whether to save the contents of the title for use in the table of contents. To specify this in the att section of the FOSI entry for the title, specify a value of tocentry for attname and section for attloc. When no value is supplied for attloc, the attribute is assumed to have been specified for the e-i-c being described. When the value #FOSI is supplied for attloc, the attribute indicated in attname is an internal FOSI variable.

**10.11.13.2 Technique - specifying the use of the attribute.** There are two ways of specifying how an attribute value is to be used: specval and fillval. For specval, one possible value of

**MIL-HDBK-28001**

the attribute is identified through attval. When that value actually appears in the source document for the e-i-c, the characteristics specified in the characteristic subset for the specval take effect. The string #NONZERO can be used to indicate any non-zero value, the string #NONE may be supplied for attributes whose declared value is #IMPLIED to indicate that no assignment was made to that attribute in the source document, and the string #ANY indicates any value. Furthermore, for attributes whose declared value is a number, an attval string may be of the form #xx#yyy where xx is one of the letter pairs LT, GT, EQ, LE, GE, NE and yyy is a numeric constant. This allows some simple arithmetic conditions to be tested using specval.

In addition, the prefix #ITEM# is used to indicate that the value is one out of a list. There will then be multiple specvals with the same attname, with each possible value in the list given as an attval. The characteristics specified for these list items are cumulative. That is, for all the values that actually occur in the list, all the corresponding characteristics take effect. For example, the declared value for the emph attribute on the emphasis element is NAMES (a list of names). For each possible value, (e.g., bold, italic, and underline), provide a specval with a list of characteristics for each. Then, when the actual attribute value is bold underline, the characteristics specified for bold and underline take effect.

For fillval, the value of the attribute is actually used as the value for a characteristic identified by its category (fillcat) and characteristic name (fillchar). The fillcat and fillchar values must be names of elements defined in the OS DTD. Use the characteristic subset to specify the values of the other characteristics of the category used in the fillcat. Fillval is useful when the value for a characteristic is determined by the author of the source document as opposed to the author of the FOSI, for example, the number of columns in a table. If the declared value of the attribute is CDATA, the comparison will be case sensitive, otherwise it will not be case-sensitive.

**10.11.13.3 Technique - interaction of attributes and other values.** In general, set up the characteristics for an element without regard to the possible attributes that may affect it. Then, in the attribute specification (att), override the typical values through the characteristics specified in the characteristic lists (charlist) specified for the attributes. In some cases, the value of an attribute is so integral to the nature of the element that it may not be possible to set up typical characteristics without considering the attribute values. In this case, there is no need to set up characteristics outside of the attribute specification.

**10.11.14 Describing graphics.** The characteristics described in the following paragraphs allow a formatting system to insert graphic entities into a document. Special sizing and placement characteristics are available for graphic elements. A graphic is a single illustration with no other text associated with it. A macrographic is a collection of graphics that are overlaid in a single reproduction area.

**10.11.14.1 Sources of graphics information.** A graphic element is either a non-SGML external entity (usually a graphics-encoded file) or an SGML marked-up element. Non-SGML graphic entities are usually associated with an empty element in the source DTD.

In the case of non-SGML entities, it is assumed the graphic object has an inherent bounding box that can be used to size and place the graphic within the reproduction window. In the case of SGML marked-up

## MIL-HDBK-28001

elements, such a bounding box must be defined for each block of text (element). Sizing capabilities, such as scaling, are not available for text elements. The definition of the bounding box includes its width and depth and a specification of which corner is to be used as the reference point for placement purposes. The bounding box itself is placed relative to the reproduction area. Within the bounding box, composition characteristics are used to compose the text and are relative to the bounding box. In other words, the bounding box can be thought of as the current flowing text area.

Graphic elements often have associated information about how to display the graphic. This information can reside in several places:

- a. Specified in a FOSI.
- b. Specified in source document attributes.
- c. Data attributes specified for the data content notation.
- d. System-specific graphics parameters.

The OS addresses only a and b. For any graphics characteristic, if no value is specified in the FOSI or source, it is left to the system to determine the value from other sources.

**10.11.14.2 Interaction of the reproduction window, sizing, and placement.** Specifying information for a graphic element display can be thought of as a three-step process:

- a. Defining a reproduction window into which the graphic element is to be placed.
- b. Determining which portion of the graphic is to be displayed or determining a bounding box for text.
- c. Specifying how the graphic is to be scaled and how the graphic or text block is to be positioned in the repro window.

A reproduction window can be associated with a single graphic element or with a non-graphic element that is simply a container for one or more graphic elements. In the case of a container, no sizing or placement information is specified. In the case of graphics or text blocks within the container, no repro window is specified, but sizing (for graphics) and placement information is. The graphic or text block is placed into the repro window associated with its nearest ancestor in the source document.

**10.11.14.3 World coordinates.** The world coordinate system is used to describe the two-dimensional space in which the graphic is defined and placed. The point of origin is the lower left corner of the graphic and has the coordinates (0,0). The upper right corner has the coordinates (10000,10000).

**10.11.14.4 Assumptions.** The OS assumes that there is no conflict between the information provided within the graphic entity and the SGML source provided by the author.

**MIL-HDBK-28001**

**10.11.14.5 Available space.** The available space for placement of graphics is constrained by the FOSI page models. That is, graphics must be able to fit, after allowed cropping and scaling, into one of the layout areas specified by the FOSI.

**10.11.15 Graphic characteristics.**

**10.11.15.1 Reproduction area dimensions.** Information about the size of the reproduction area (the area on the presentation media) in which the graphic is to be placed.

Characteristics:	Values:
Repro Area Width	Size/Distance
Repro Area Depth	Size/Distance

**10.11.15.2 Graphic sizing (sizing).** Information concerning constraints on how to modify the size or view of graphics to be placed in the reproduction area.

Characteristics:	Values:
Graphic Name	Pointer
Horizontal Scaling	Integer
Vertical Scaling	Integer
Scale to Fit	Toggle
Lower Left Coordinates	String
Upper Right Coordinates	String

**Notes:**

- a. The graphic name (graphname) must be the name of an entity declared in the FOSI subset.
- b. Scaling values are interpreted as percentages.
- c. Scaling is disallowed by supplying a value of 0 for the horizontal (hscale) and vertical scaling (yscale) characteristics as well as the scale to fit (scalefit) characteristic.
- d. The scale to fit characteristic allows the graphic to be scaled as needed to fit the size of the reproduction area. Scaling is by the same factor in both directions. Horizontal and vertical scaling characteristics have precedence when they have values other than 0.
- e. The sizing coordinates define a section (window) of the graphic. This allows both general cropping functionality as well as the ability to designate a particular portion of the graphic to be used for an illustration.

**MIL-HDBK-28001**

- f. The syntax to be used for the lower left coordinate (llcordra) string is `integer, integer` where the first integer refers to the starting position of the graphic window along the horizontal axis and the second integer refers to the starting position of the graphic window along the vertical axis. Thus a start coordinate value of 0,0 indicates that the desired graphic window begins at the lower left point of the graphic board.
- g. The syntax to be used for the upper right coordinate (urcordra) string is `integer, integer` where the first integer refers to the ending position of the graphic window along the horizontal axis and the second integer refers to the ending position of the graphic window along the vertical axis. Thus an end coordinate value of 10000,10000 indicates that the desired graphic window ends at the upper right point of the graphic board. A specification of 0,0 for the lower left and 10000,10000 for the upper right designates that the portion of the graphic to be used includes the entire graphic board, whereas 5000,5000 for the lower left and 10000,10000 for the upper right would signify that the upper right quadrant of the graphic would be used for the illustration.

**10.11.15.3 Text block.** Information concerning the size and reference point of a text block (textblock).

Characteristics:	Values:
Text Block Width	Size/distance
Text Block Depth	Size/distance
Horizontal reference point	List (Left, Right)
Vertical reference point	List (Top, Bottom)

**Notes:**

- a. When text block depth (tbdepth) is not specified or set to zero, the text block has variable depth according to its content, much like a table cell.
- b. The reference points specify the fixed corner of the text block. It is the corner placed at coordinate start and end in the placement category. In the case of variable depth text blocks, the text block depth grows away from the fixed corner, that is, downward if the reference point is top and upward if the reference point is bottom.

**10.11.15.3.1 Placement (placemnt).** Information concerning constraints on where and how to place graphics or text blocks with respect to the reproduction area.

Characteristics:	Values:
Horizontal Placement	List (Left, Center, Right, None)
Vertical Placement	List (Top, Middle, Bottom, None)

**MIL-HDBK-28001**

Start Coordinates	String
End Coordinates	String
Rotation	Toggle

**Notes:**

- a. Horizontal (hplace) and vertical placement (vrplace) are used to place graphics within a repro area. Either relative (horizontal and vertical) or specific (coordinates) positioning should be specified. That is, use of the placement and coordinate characteristics is mutually exclusive. When start and end coordinates are supplied, the values for horizontal and vertical placement are automatically set to none.
- b. The start (coordst) and end (coordend) coordinate characteristics may be used for placing multiple graphics or portions of graphics in a single reproduction area. When these characteristics are used and scale to fit is turned on, then the graphic is scaled to fit within the bounds of the coordinates specified instead of the entire reproduction area. If scale to fit is turned off and the graphic will not fit in the window specified by the start and end coordinates, then the start coordinate is to be used as an origin and as much of the graphic as will fit in the window is displayed. End coordinate is ignored for text blocks, as they cannot be scaled.
- c. The syntax to be used for the start coordinate string is `integer, integer` where the first integer refers to the starting position of the graphic along the horizontal axis and the second integer refers to the starting position of the graphic along the vertical axis. Thus a start coordinate value of 0,0 indicates that the lower left point of the graphic is to be placed starting at the lower left corner (point of origin) of the repro area.
- d. The syntax to be used for the end coordinate string is `integer, integer` where the first integer refers to the ending position of the graphic along the horizontal axis and the second integer refers to the ending position of the graphic along the vertical axis. Thus an end coordinate value of 10000,10000 indicates that the graphic is to be placed such that the upper right point of the graphic coincides with the upper right corner of the repro area.
- e. Rotation is only allowed for figures, charts, graphics, and tables taken in their entireties.
- f. When the value for rotation is off, the graphic is placed in the same orientation in which it has been received. When the value is on, the graphic is rotated 90 degrees counterclockwise.

**10.11.15.4 Elements and graphics attributes.** It should be noted that in order for a graphic to be processed in accordance to the specifications provided by the graphic attributes, the FOSI must define the appropriate formatting characteristics and values for all of the graphic attributes. The

## MIL-HDBK-28001

graphic element and its attributes are to be used in markup of documents whose DTD is in conformance with MIL-PRF-28001.

- a. **Size/distance.** This value is used to express measurement. The syntax for describing a size/distance value is number followed by the unit. For instance, 6pt means 6 points. Numbers can be positive or negative. The numbers can also express precision in tenths or hundredths with the use of decimal points. For instance, 6.4pt means 6 and 4/10ths of a point. Combinations of units are allowed, but must be completely specified. For example, 5pi 3pt is the correct syntax for specifying 5 picas and 3 points.

**10.11.16 Important terms related to graphics.** Terms used throughout this section are defined as follows:

- a. **Source graphic.** A source graphic is a graphic file that can be retrieved using the name provided by the boardno attribute. This source graphic can be cropped, scaled, and positioned within a reproduction area.
- b. **Reproduction area.** A reproduction area is a window in which a graphic will be placed. Specifications can be provided (via graphic attributes) for the positioning of the graphic within the reproduction area.
- c. **World coordinates.** The world coordinate system is used to describe the two-dimensional space in which the graphic is defined and placed. The point of origin is the lower left corner of the graphic and has the coordinates (0,0). The upper right corner has the coordinates (10000,10000).

**10.11.17 FOSI graphic environment.** In order to accommodate the requirement for uniform graphic styles and to ease the burden of having the author specify all graphic attribute values, a FOSI can provide different sets of graphic characteristic values which can be accessible by the graphsty ID. However, every FOSI must define a default graphic environment. This default set of characteristic values allows the author to leave out attribute values that are supplied in the FOSI and does not require the graphsty attribute to be specified.

For the examples in this document, the graphic characteristic values specified below will be used as the default graphic environment.

```
<grphdesc>
<graphenv
  graphenvid="default">
<graphchars>
<repro
  reprowid="4.5in"
  reprodep="3.5in">
<sizing
  hscale="0"
  vscale="0"
  scalefit="yes"
```

**MIL-HDBK-28001**

```

      llcordra="0,0"
      urcordra="10000,10000">
    <placemnt
      hplace="center"
      vrplace="middle"
      coordst="0,0"
      coordend="10000,10000"
      rotation="0">

```

**10.11.17.1 Default characteristic values.** The default attribute values provided by the default FOSI graphic environment are provided below.

- a. Reproduction area width (reprowid) & reproductions area depth (reprodep): Default values are 4.5 inches width and 3.5 inches depth. Default values should be set up to satisfy the majority of graphics being placed in a document, if possible.
- b. Lower left coordinate (llcordra) & upper right coordinate (urcordra): Default values should indicate that entire graphic be scaled into the reproduction area. The syntax to be used for the lower left coordinate string is *integer, integer* where the first integer refers to the starting position of the graphic window along the horizontal axis and the second integer refers to the starting position of the graphic window along the vertical axis. Thus a start coordinate value of 0,0 indicates that the desired graphic window begins at the lower left point of the graphic board. Note that the urcordra attribute of the graphic tag in the example source DTD logically corresponds to the urcordra characteristic of the sizing element of the OS.
- c. The syntax to be used for the upper right coordinate string is *integer, integer* where the first integer refers to the ending position of the graphic window along the horizontal axis and the second integer refers to the ending position of the graphic window along the vertical axis. Thus an end coordinate value of 10000,10000 indicates that the desired graphic window ends at the upper right point of the graphic board. A specification of 0,0 for the lower left and 10000,10000 for the upper right designates that the portion of the graphic to be used includes the entire graphic board, whereas 5000,5000 for the lower left and 10000,10000 for the upper right would signify that the upper right quadrant of the graphic would be used for the illustration.
- d. Horizontal scale (hscale) & vertical scale (vscale): Default values should specify no horizontal or vertical scaling.
- e. Scale to fit (scalefit): Default value should be yes. However, if values for hscale & vscale are provided in an instance, they will take precedence over scalefit.
- f. Horizontal placement (hplace) & vertical placement (vrplace): Default value of hplace should be center; vrplace should be middle. However, this is appropriate only if the reproduction area is larger than the size of the source graphic and scalefit is not specified. Note that the vrplace attribute of the graphic tag in the example source DTD logically corresponds to the vrplace characteristic of the placemnt element of the OS.

**MIL-HDBK-28001**

- g. Coordinate start (coordst) & coordinate end (coordend): Default values should indicate that the entire reproduction area is used.
- h. Rotation (rotation): Default value should indicate no rotation.

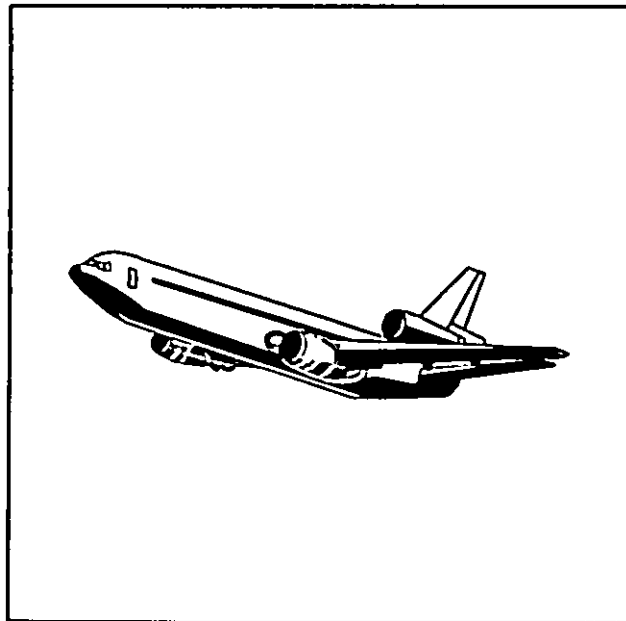
As is always the case where an author specifies attribute values that affect formatting, values supplied in an instance override the FOSI. This allows authors to make an exception where necessary, and also frees them from having to specify many attributes when they are already provided in the FOSI.

**10.11.18 Graphic.** This section describes how to use the graphic element in the document instance, how specification of its attributes affect the FOSI, and how the instance interacts with the FOSI to produce a graphic display. A graphic can be either stored as vector (MIL-PRF-28000 (IGES) or MIL-PRF-28003 (CGM)) or raster (MIL-PRF-28002) data and is used as the source for an illustration in a document. Figure 2, Sample graphic 1, will be used as the source graphic for all examples in this document. This graphic is 4.477 inches wide and 3.39 inches high. It is being displayed in a reproduction area which is 4.5 inches wide and 3.5 inches high. The graphic file should be stored on the processing system and accessible by the boardno attribute. For the examples in this document, the source graphic is accessible using the boardno name, plane.

**10.11.18.1 Required attribute.**

**10.11.18.1.1 Boardno.** The boardno attribute is a required attribute. To incorporate the source graphic used in this handbook, the SGML tagging is:

```
<graphic boardno="plane">
```



**FIGURE 2.** Sample graphic 1.

**MIL-HDBK-28001**

The purpose of this attribute is for entering the unique graphic identifier, and it acts as a pointer to reference an external file containing the source graphic. The declared value is ENTITY. Therefore, at the beginning of the SGML tagged document in the document type declaration, there must be an ENTITY declaration defining the boardno name. For example:

```
<!ENTITY plane SYSTEM "airplane.g4" NDATA fax>
```

where plane is the boardno attribute value and airplane.g4 is the filename containing the graphic. NDATA is non-SGML data containing codes outside the set declared to be valid SGML characters for the document. Fax is the required notation for raster data.

**10.11.18.2 Optional (implied) attributes.** The following is an overview of all the optional graphic attributes of the graphic element in the example source DTD. The repropwid and reprodep attributes specify a reproduction area in which to place the graphic (see 10.11.18.2.1). The llcordra and rucordra attributes provide coordinates to identify a portion of the source graphic to be used (see 10.11.18.3.1). The hscale and vscale attributes allow the horizontal and vertical scaling of a graphic (see 10.11.18.4.1). The scalefit attribute can be used to automatically scale a graphic to fit a reproduction area (see 10.11.18.5.1). The hplace and vplace attributes can be used to indicate a general placement of a graphic within a reproduction area (see 10.11.18.6.1). The coordst and coordend attributes can be used to indicate an exact placement of a graphic within a reproduction area through use of coordinates (see 10.11.18.7.1). The rotation attribute can be used to specify the rotation of a graphic when being placed in the reproduction area (see 10.11.18.8.1). Finally, the graphic style (graphsty) attribute can be used to access values for any or all of these attributes. These default values must be defined in the FOSI (see 10.11.18.10) and accessible by the graphsty id. If no graphsty id is provided for a graphic, the FOSI uses the default values.

**10.11.18.2.1 Reprowid & reprodep.** Reprowid is used to specify the desired reproduction area width (size/distance). Reprodep is used to specify the desired reproduction area depth (size/distance). If specifications are not provided for either attribute, the values specified in a macrograph tag should be used. If there is no macrograph tag or if there were no values specified in the macrograph tag, the FOSI provides the values.

## MIL-HDBK-28001

Example (see figure 3):

```
<graphic boardno="plane" reprowid="4.5in"
  reprodep="3.5in">
```

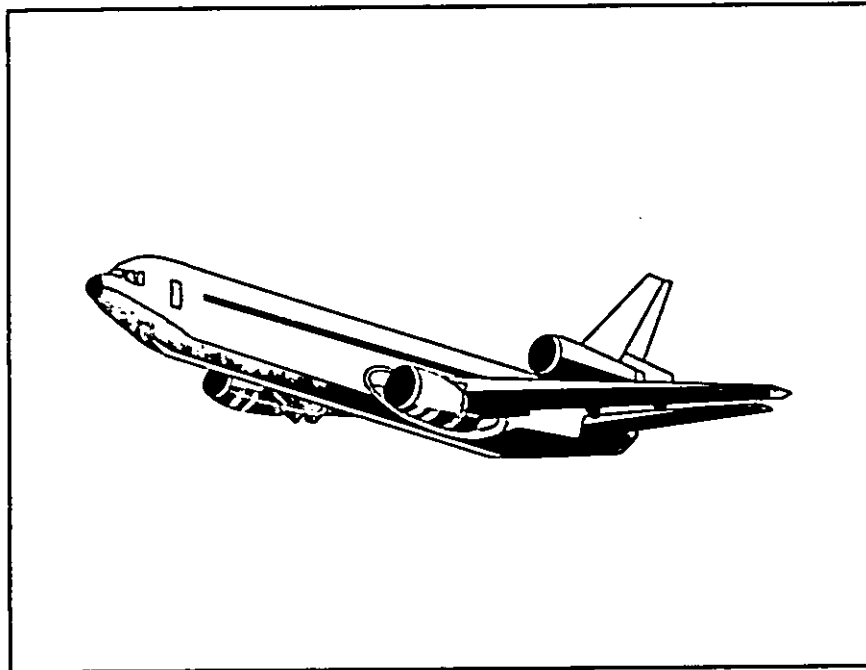


FIGURE 3. Sample graphic 2.

**10.11.18.3 Implied values of remaining attributes:** The attribute values that have not been specified take on the default values as specified in the FOSI. Default *llcordra* & *rucordra* specify that the entire graphic be used. Default *hscale* & *vscale* specify no scaling of the graphic. *Scalefit* specifies the graphic be scaled to fit the reproduction area. Default *hplace* & *vplace* specifies that *hplace* be center; *vplace* be middle. Note that this is not appropriate since the graphic is being scaled to fit the reproduction area. Default *coordst* & *coordend* specifies that the entire reproduction area be used. Default rotation is no rotation.

**10.11.18.3.1 *llcordra* & *rucordra*.** *llcordra* is used to specify the left lower coordinate pair of the portion of a graphic to be placed in the reproduction area. The coordinates are separated by a comma. *rucordra* is used to specify the upper right coordinate pair of the portion of a graphic to be placed in the reproduction area. The coordinates are separated by a comma. These attributes allow the ability to designate a particular portion of the graphic to be used for an illustration and employs world coordinates.

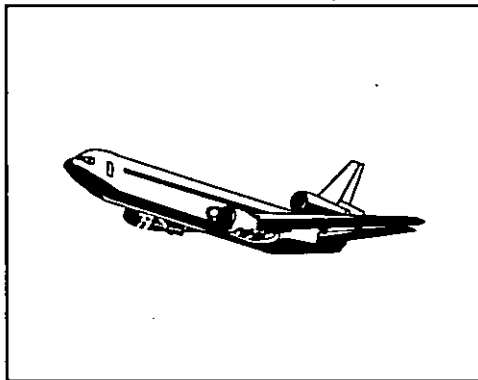
The syntax for the *llcordra* attribute (left lower) is *integer, integer*. The first integer is the starting position of the graphic window along the horizontal axis. The second integer is the starting position of the graphic window along the vertical axis. For instance, a *llcordra* value of 0,0 means that the graphic window starts at the lower left point of the graphic.

**MIL-HDBK-28001**

The syntax for the `rucordra` attribute (right upper) is `integer, integer`. The first integer is the ending position of the graphic window along the horizontal axis. The second integer is the ending position of the graphic window along the vertical axis. For instance, a `rucordra` value of 10000,10000 means that the graphic window ends at the upper right point of the graphic. `lrcordra` value of 0,0 and `rucordra` value of 10000,10000 specifies the entire graphic be used for the illustration.

Example (see figure 4):

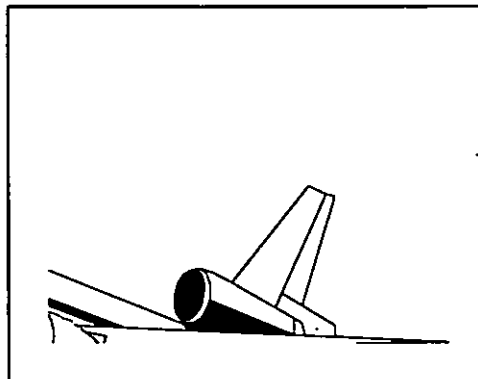
```
<graphic boardno="plane" lrcordra="0,0" rucordra="10000, 10000"
reprowid="2.5in" reprodep="2in">
```



**FIGURE 4. Sample graphic 3.**

Example (see figure 5):

```
<graphic boardno="plane" lrcordra="5000, 5000"
rucordra="10000, 10000" reprowid="2.5in" reprodep="2in">
```



**FIGURE 5. Sample graphic 4.**

`lrcordra` value of 5000,5000 and `rucordra` value of 10000,10000 specifies the upper right quadrant of the graphic be used for the illustration.

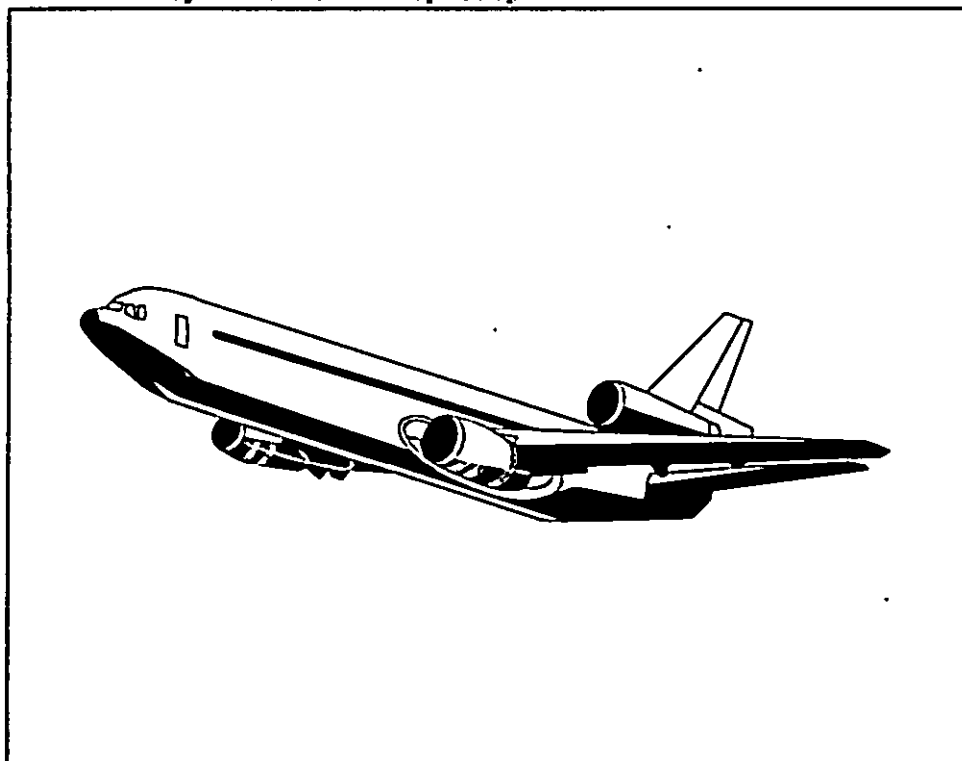
**MIL-HDBK-28001**

**10.11.18.4 Implied values of remaining attributes:** The attribute values that have not been specified take on the default values as specified in the FOSI. Default hscale & vscale provide no specific scaling of the graphic. Default scalefit specifies the graphic be scaled to fit the reproduction area. Default hplace & vplace specifies that hplace be center; vplace be middle. Note that this is not appropriate since the graphic is being scaled to fit the reproduction area. Default coordst & coordend specifies that the entire reproduction area be used. Default rotation is no rotation.

**10.11.18.4.1 Hscale & vscale.** The hscale and vscale properties can be changed to scale the graphic to whatever size desired (either larger or smaller than the original). The scaling values are expressed in the form of percentages.

Example (see figure 6):

```
<graphic boardno="plane" hscale="100" vscale="100"  
reprowid="5in" reprodep="4in">
```



**FIGURE 6. Sample graphic 5.**

## MIL-HDBK-28001

Example (see figure 7):

```
<graphic boardno="plane" hscale="50" vscale="50"
reprowid="2.5in" reprodep="2in">
```

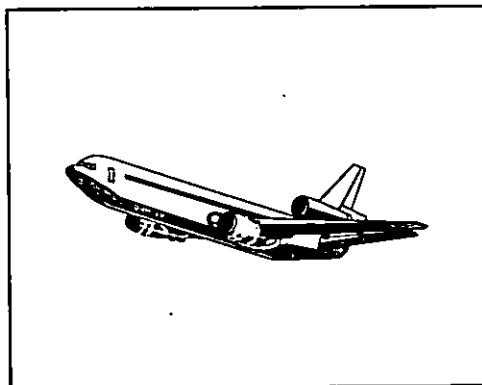


FIGURE 7. Sample graphic 6.

Example (see figure 8):

```
<graphic boardno="plane" hscale="25" vscale="25"
reprowid="1.25in" reprodep="1.25in">
```

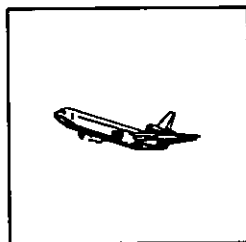


FIGURE 8. Sample graphic 7.

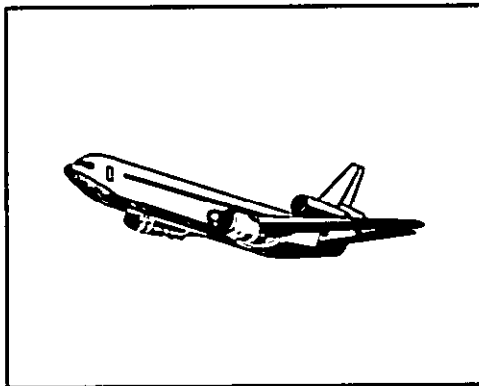
**10.11.18.5 Implied values of remaining attributes:** The attribute values that have not been specified take on the default values as specified in the FOSI. Default `llcordra` & `rucordra` specify that the entire graphic be used. Default `scalefit` specifies the graphic be scaled to fit the reproduction area. However, since `hscale` and `vscale` are being specified in the examples, these take precedence over `scalefit`. Default `hplace` and `vplace` specifies that `hplace` be center; `vplace` be middle. Default `coordst` & `coordend` specifies that the entire reproduction area be used. The default rotation is no rotation.

**10.11.18.5.1 Scalefit.** This attribute allows the graphic to be scaled as needed to fit the size of the reproduction area. The factor for scaling is the same in both directions. If the scaling attributes (`hscale` and `vscale`) have values other than 0, these take precedence over the `scalefit` attribute. The declared value is `%yesorno`; 1 for yes and 0 for no.

**MIL-HDBK-28001**

Example (see figure 9):

```
<graphic boardno="plane" reprodw="2.5in" reprodep="2in" scalefit="1">
```



**FIGURE 9.** Sample graphic 8.

**10.11.18.6 Implied values of remaining attributes:** The attribute values that have not been specified take on the default values as specified in the FOSI. Default llcordra & rucordra specify that the entire graphic be used. Default hscale & vscale provide no specific scaling of the graphic. Default hplace & vplace specifies that hplace be center; vplace be middle. Note that this is not appropriate since the graphic is being scaled to fit the reproduction area. Default coordst & coordend specifies that the entire reproduction area be used. The default rotation is no rotation.

**10.11.18.6.1 Hplace & vplace.** The hplace attribute is used to specify a horizontal placement of the graphic within the reproduction area. With the use of the hplace attribute, a graphic can be placed to the left, right, or center within the reproduction area.

The vplace attribute is used to specify the vertical placement of the graphics within the reproduction area. With the use of the vplace attribute, the graphics can be placed to the top, bottom, or middle of the reproduction area.

## MIL-HDBK-28001

Example (see figure 10):

```
<graphic boardno="plane" hplace="left" reprojid="2.25in"  
reprodep="2.25in" hscale="50" vscale="50">
```

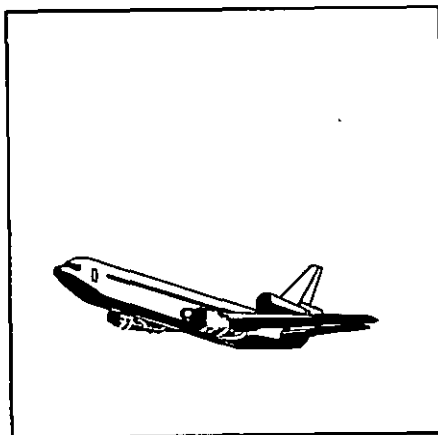


FIGURE 10. Sample graphic 9.

Example (see figure 11):

```
<graphic boardno="plane" vplace="top" reprojid="2.25in"  
reprodep="2.25in" hscale="50" vscale="50">
```

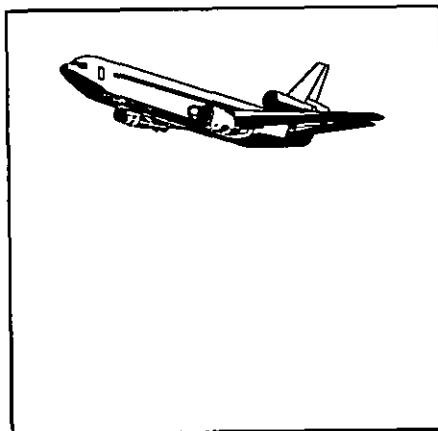


FIGURE 11. Sample graphic 10.

## MIL-HDBK-28001

Example (see figure 12):

```
<graphic boardno="plane" hplace="right" vplace="bottom"
reprowid="2.25in" reprodep="2.25in" hscale="50" vscale="50">
```

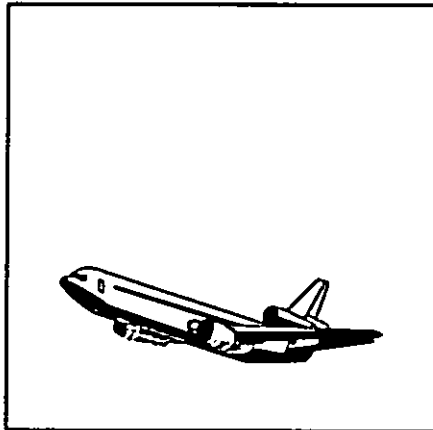


FIGURE 12. Sample graphic 11.

**10.11.18.7 Implied values of remaining attributes:** The attribute values that have not been specified take on the default values as specified in the FOSI. Default llcordra & rucordra specify that the entire graphic be used. Default scalefit specifies the graphic be scaled to fit the reproduction area. However, since hscale and vscale are being specified in the examples, these take precedence over scalefit. Default coordst & coordend specifies that the entire reproduction area be used. The default rotation is no rotation.

**10.11.18.7.1 Coordst & coordend.** The coordst attribute provides the left lower coordinate pair which specifies the start position in the reproduction area for placing the graphic. The coordend attribute provides the right upper coordinate pair. This is the end position in the reproduction area for placing the graphic. Both attributes employ the world coordinates.

The syntax for the coordst attribute (left lower) is integer, integer. The first integer is the starting position of the reproduction area along the horizontal axis. The second integer is the starting position of the reproduction area along the vertical axis. For instance, a coordst value of 0,0 means that the lower left corner of the graphic would be placed to the lower left of the reproduction area.

The syntax for the coordend attribute (right upper) is integer, integer. The first integer is the ending position of the reproduction area along the horizontal axis. The second integer is the ending position of the reproduction area along the vertical axis. For instance, a coordend value of 10000,10000 means that the upper right corner of the graphic would be placed at the upper right point of the reproduction area.

# MIL-HDBK-28001

Example (see figure 13):

```
<graphic boardno="plane" coordst="0,0" coordend="10000,10000"  
reprowid="2.25in" reprodep="2.25in" hscale="50" vscale="50"> .
```

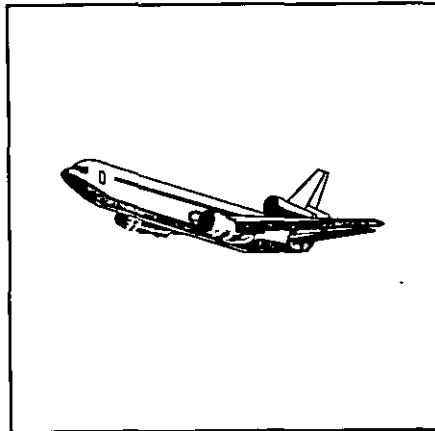


FIGURE 13. Sample graphic 12.

## MIL-HDBK-28001

Example (see figure 14):

```
<graphic boardno="plane" coordst="5000,5000"  
coordend="10000,10000" hscale="25" vscale="25" reprowid="2.25in"  
reprodep="2.25in">
```

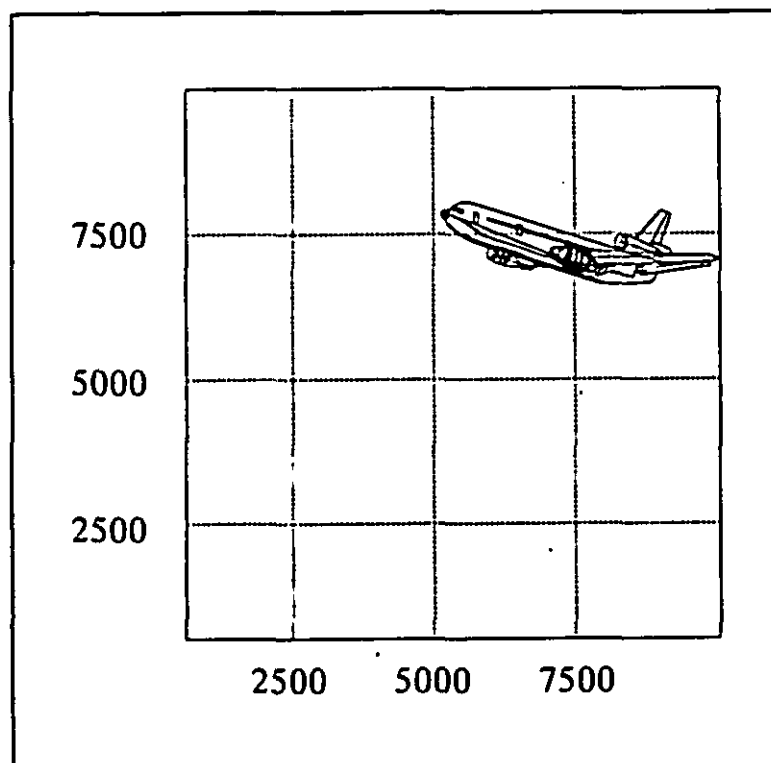


FIGURE 14. Sample graphic 13.

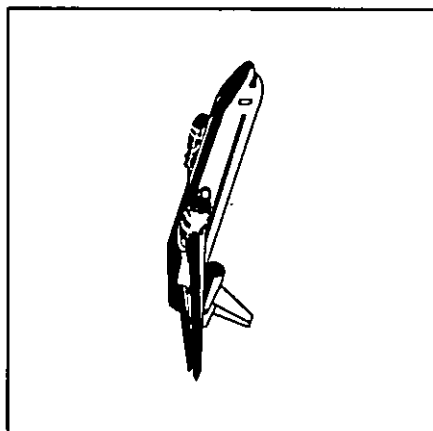
**10.11.18.8 Implied values of remaining attributes:** The attribute values that have not been specified take on the default values as specified in the FOSI. Default llcordra and rucordra specify that the entire graphic be used. Default scalefit specifies the graphic be scaled to fit the reproduction area specified by coordend and coordst. However, since hscale and vscale are being specified in these examples, these take precedence over scalefit. The default rotation is no rotation.

**10.11.18.8.1 Rotation.** The rotation attribute value gives the degree of rotation on the graphics.

**MIL-HDBK-28001**

Example (see figure 15):

```
<graphic boardno="plane" reprodep="2.25" reprowid="2.25"
hscale="50" vscale="50" rotation="90">
```



**FIGURE 15.** Sample graphic 14.

**10.11.18.9 Implied values of remaining attributes:** The attribute values that have not been specified take on the default values as specified in the FOSI. Default llcordra & urcordra specify that the entire graphic be used. Default scalefit specifies the graphic be scaled to fit the reproduction area. However, since hscale and vscale are being specified in the examples, these take precedence over scalefit. Default coordst & coordend specifies that the entire reproduction area be used.

**10.11.18.10 Graphsty.** The graphic style (graphsty) attribute is a feature which allows access to graphic environments that have been pre-defined in a FOSI. Attribute values are defined in the FOSI and are accessible by the graphsty attribute id. In every FOSI there must be one graphic environment which has the name default. This default set of characteristic values allows the author to leave out attribute values that are supplied in the FOSI. If the graphsty attribute is not included in an instance, the default graphic environment is used. As is always the case where an author can specify attribute values that affect formatting, the author's values override the FOSI characteristics. This allows for authors to make an exception where necessary, but also frees them from having to specify many attributes when they are already provided in the FOSI.

**10.11.18.11 Describing tables.** This section describes the characteristics necessary for the composition of tables. Tables are treated separately in this handbook because they have unique formatting characteristics. Tables are important in technical manuals because they contain and present a large amount of data in a format that shows relationships among the data. The characteristics for tables described below allow for robust and discretionary access and manipulation of data contained within tables, and facilitate exchange with, and use within, databases.

Within a FOSI, the table description is used to describe the organization and formatting of an actual table, that is, data organized into a two-dimensional grid. Any associated information, such as title, is described in the style description.

**MIL-HDBK-28001**

**10.11.18.11.1 Tables as graphics.** While it is recognized that tables are often created and treated as graphic illustrations, it is essential for automated processing of source data that the markup of tabular data be developed in such a way that allows the elements within a table to be mapped to the FOSI table description and that automated publishing systems be able to manipulate and reproduce tabular data through the use of a FOSI. Treating tables as graphics is not precluded, but should be carefully evaluated against requirements for information use. This section deals only with tabular data appropriately tagged.

**10.11.18.11.2 The structure of tables.** Tables have two structures. The source structure describes the organization of table data within the source document. The output structure describes how that data actually appears in a two-dimensional format in the output. The purpose of the FOSI entry for a table is to describe how those two structures are related.

**10.11.18.11.3 Source structure.** The source structure of a table is defined in the source DTD. This structure should reflect the logical organization of the data as it relates to its purpose. Typically, however, this structure also reflects the two-dimensional output structure. The following is the source structure for a table.

**Table**

```

Title
Table Groups
    Column Specifications
    Table Head
        Column Specifications
        Rows
            Entries
            Entry Tables
    Table Foot
        Column Specification
        Rows
            Entries
            Entry Tables
    Table Body
        Rows
            Entries
            Entry Tables

```

Note that this source structure is generic, that is, the elements are general and do not relate to any specific type of information that might go into the table. Other source structures for tables could be defined that more specifically detail the table content. For example, a source structure for a special tools list table could look like this:

**MIL-HDBK-28001****Special Tools List****Tools**

Description

Part Number

Reference

In this case, the logical elements in the source structure are more closely related to the data in the table.

**10.11.18.11.4 Output structure and table objects.** A table is a rectangular two-dimensional grid. Its horizontal and vertical dimensions may or may not have uniform measures; they may be determined from the source document instance, the FOSI, or both. The objects within a table are table subset groups, table subsets, columns, rows, and cells. A table is the entire rectangle that takes up space in the flowing text area. Characteristics of the table control the frame. A table subset group (tgroup) is a set of table subsets containing an optional heading subset, an optional footing subset, and one or more body subsets. A table subset is a set of contiguous rows within a table such as the header, footer, or body of a tgroup. There is no space between table subsets in a table. There are three types of table subsets - heading, footing, and body subsets.

The rows of heading or footing subsets are repeated above and below (respectively) each contiguous (unbroken) block of rows from any of the table body subsets in the same table subset group. Thus, if the table body subsets break across pages or columns, the heading and footing subsets for that table subset group are repeated.

Table subsets may have different numbers of columns, but must be the same width as the table. A column is a vertical collection of cells. A column has some specified width and is the depth of the table subset. A row is a horizontal collection of cells. The width of a row is the width of the table and the depth of a row is the depth of the deepest cell. A cell is the intersection of a column and row and forms the basic area into which table content is placed. Characteristics of the cell control the column and row separators, margins, and alignment. Table objects have the following hierarchy:

**Table**

Table Subset Groups (1 or more)

Column specifications (0 or more)

Table Subsets (1 or more)

Column Specifications (0 or more)

Rows (1 or more)

Cells (1 or more)

In other words, tables are made up of table subset groups, (typically a single subset group), which contain table subsets, which are made up of columns and rows, which intersect to form cells.

**10.11.18.12 Mapping source structure to output structure.** Through the table description (tabdesc) of a FOSI, the source structure of a table in the source DTD is mapped into the output structure defined in this section. Each source element is mapped into one or more table object(s). For example, the table source structure in appendix A has the following mapping:

**MIL-HDBK-28001**

Table	Table
Tgroup	Table Subset Group
Colspec	Column
Spanspec	Column
Thead	Table Subset
Tfoot	Table Subset
Tbody	Table Subset
Row	Row
Entry	Cell
Entrytbl	Table subset

Note that the table title in the source structure does not have a corresponding object in the output structure. Formatting of the table title is specified in the style description.

Other table structures may be mapped to the table output structure differently. For example, the source structure for a special tools list made up of description, part number, and reference might be represented with the following simple DTD fragment:

```
<!ELEMENT tools - - (desc, partnum, ref)+>
<!ELEMENT (desc, partnum, ref) - o (#PCDATA)>
```

Here tools is mapped to a table in the FOSI, and desc, partnum, and ref are mapped to cells. To fully specify the output of the table, however, may require specifying characteristics for columns and rows, which do not have corresponding elements in the source.

This table, including an implied heading, may be mapped to the table object output structure as follows:

Tools	Table
	Table Subset Group
	Columns (3)
	Table Subset (heading)
	Row
	Cells (3)
	Table Subset (body)
Desc	Row
	Cell
Partnum	Cell
Ref	Cell

Note that each element may be mapped to a contiguous range of objects in the table object hierarchy, with specification of appropriate characteristics for each. This mapping is accomplished by specifying the table output objects and their characteristics in the charlist of the e-i-c of each relevant element. Each table output object has a category in the output specification, with associated characteristics.

**MIL-HDBK-28001**

The content of a source element must be contained within the lowest level table object specified in the charlist for that source element. Thus, an error would occur if a source element were mapped into a cell, and one of its sub-elements were mapped into a table subset.

The range of table objects into which a source element is mapped must be contiguous in the table object hierarchy. It is an error for an element to be mapped to a discontinuous range of elements. Thus, an error would occur if a FOSI attempted to map an element into a table subset and a cell, but not into a row.

To place text or graphics directly from the FOSI into the output table objects, as shown in the above mapping, use the appropriate table object categories in the subchars specification of a usetext, puttext, or putgraph.

The e-i-cs for source elements to be mapped into table objects may use the att and specval or fillval constructs to create table specifications which are dependent on attributes of the document instance elements. This supports the implementation of named table and table group styles.

**10.11.18.13 Text flow rules of tables.** Text flows within tables from left to right and from top to bottom. That is, the first column of the first row (the first cell) is filled with the necessary text; then the second column of the first row (the second cell) is filled. After the last column of the first row is filled, text flows to the first column of the second row, and so on. Whenever two or more cells are spanned, they are treated as a single cell with respect to the flow of text and filled with its content in the upper left cell only.

Columns are implicitly numbered starting with 1 at the left most column and incrementing by 1 to the right. Rows in each table subset are implicitly numbered starting with 1 at the top row and incrementing by 1 to the bottom. A cell is uniquely identified by its position within a row and column.

Spanning is the creation of cells that are larger than a single grid location. Span width indicates the number of columns covered horizontally by a cell. Span depth indicates the number of rows covered vertically by a cell. Spanning occurs in the direction of the flow of cell filling, that is, for horizontal spanning, to the right of the cell where spanning is specified, and for vertical spanning, towards the bottom from the cell where spanning is specified. A spanned cell is considered to be part of the first row and first column in which it occurs.

The source markup for a table entry can explicitly specify the name of the column in which the cell contents is to occur. When this information is not supplied, a flow rule is used to determine the cell location. Entries are placed in the current column, which, if not previously specified is column 1. After a column is filled, the current column becomes the column with the number current column number + span width (where no spanning means span width is 1).

When fewer entries occur than cells in a row, the remaining cells are treated as if they were each filled by an entry with empty content; that is, row and column separators appear for those cells that would have them had the cell been filled.

**MIL-HDBK-28001**

**10.11.18.14 Specifying the style of a table.** Two aspects to specifying the style of a table are geometric and text composition. The geometric aspect includes the number of columns, rulings, and margins. The text composition aspect includes font, positioning of text within cells, and generally those characteristics that can be applied to text. Both of these kinds of characteristics can be specified in the FOSI. Special table characteristics are provided to control the style of the table itself. Composition characteristics are used to specify the style of the content.

In some cases, such as in the DTD in appendix A, style parameters can be controlled through attributes in the source document. In these cases, these values are passed through to the FOSI through the fillval construct. It is a matter of agreement between the source DTD author and FOSI author which values are fixed in the FOSI and which values are actually supplied by the author in the source document.

**10.11.18.14.1 Table characteristics.** When an e-i-c is mapped to the table output object by the occurrence of the table attributes (tabatts) category in its charlist, unique characteristics for that table as a whole are set up, such as the width, frame style (framestyle) and frame thickness (thick).

**10.11.18.14.2 Table subset group and table subset characteristics.** When an e-i-c is mapped to a table subset group output object, or a table subset output object by the occurrence of the tgroupatts or subsetatts category in its charlist, unique characteristics for that table subset group or subset are set up, including the number of columns in the table, as well as characteristics for columns, rows, and cells. Columns are implicitly numbered beginning with 1 for the leftmost column and incrementing by 1 for each column. Column names can be associated with each implicitly numbered column. Span names can be established to refer to a horizontal span of columns indicated by the name of the column in which to start and the column in which to end.

When no table subset characteristics (e.g., number of columns) or column characteristics (e.g., column widths) are specified for a table subset object, the characteristics specified for the nearest ancestor in the source document that is mapped to a table subset group are used. If no such element exists, it is an error.

In addition, composition characteristics can be specified for each e-i-c to specify the style and positioning of the content within the elements. These composition characteristics override the table characteristics already set up for the table objects. Indents specified for the content are additive to the margins established for the cell.

**10.11.18.14.3 Scope of table characteristics.** In general, there is a unique set of characteristics for each table object. In addition, there is a set of standard cell characteristics (stdcellatts) that control the characteristics of a cell but can be specified on any table object. These characteristics include column and row separators, margins, and alignment. When specified on a table object, these characteristics apply to the cells within the scope of that object. For example, when standard cell characteristics are specified on a column, those values apply to all the cells in the column. When specified on more than one table object, the values for objects lower in the hierarchy override the others for the scope of that object; that is, the order of precedence is cell, row, column, table subset, table subset group, and table.

## MIL-HDBK-28001

The table description categories (tabatts, tgroupatts, subsetatts, colatts, rowatts, cellatts, stdcellatts), in the charlist are intended only for the specification of table output object characteristics. The categories may be used in environments and charsubsets in the style description section of a FOSI to define environments and charsubsets which may be used in the table description section of the FOSI, as part of a table description. The use of any of these categories in an e-i-c in the style description section of a FOSI, either directly or by reference to an environment of charsubset is an error.

**10.11.18.14.4 The content of tables.** While it may be sufficient to specify the formatting of simple text (#PCDATA) within a cell by supplying cell characteristics and composition characteristics for the cell, there may be a need to specify additional composition characteristics when the cell contains more complex element content. For example, lists within table entries may differ from lists in flowing text. These e-i-cs (that are children of the source table entry) must also be included in the style description of the FOSI and the appropriate composition characteristics specified. The following composition characteristics do NOT apply within tables: highlight reverse, background color and background screen (cell reverse, color, and shading take precedence); keeps (row breaking takes precedence); vertical justification; textbreak, start column, start page, page model id, and new page model; span (cell spanning takes precedence).

**10.11.19 Overriding FOSI values through the source markup.** For any particular source DTD, the values of many source attributes may map directly to table characteristic values. Where this occurs, source attribute values can override any value supplied in the corresponding table characteristic. A FOSI entry may be constructed such that values are supplied for characteristics but any values supplied in the source override the specified value by using the fillval construct.

Authors need not specify all the attributes available to them. The attributes allow authors to create an exception to a particular table style. For example, an author may wish to specify that a particular column should have its contents center-aligned when referencing a table style that specifies that all columns are left-aligned. The author then needs only to specify a single attribute to get the desired result, as the center value the author specifies for the align attribute can override the left value specified in the FOSI. The left value remains in the FOSI after the exception is executed.

**10.11.19.1 Table characteristics.** The characteristics listed below are unique to tables, and may be used in conjunction with composition characteristics to fully specify the output of tables. Characteristics that apply specifically to the table element.

Characteristics:	Values:
Table Style	ID
Width Type	List (Specific, Relative)
Specific Width	Size/Distance
Relative Width	Percentage
Frame List	(All, Top, Bottom, Top&Bottom, Sides, None)
Frame Thickness	Size/Distance
Frame Style	List (None, Blank, Single, Bold, Double)
Continue with Row Separator	Toggle

**MIL-HDBK-28001****EXPLANATION:**

- a. FOSIs have a minimum of one table style specified as a result of supplying a default value for each table characteristic.
- b. Width type (widthtype) indicates whether the width of the table is a particular specified value or is relative to the column or page in which it is placed. When specific is specified, a value should be supplied for the specific width (specwidth) characteristic. When relative is specified, a value for the relative width (relwidth) characteristic should be supplied to indicate the percentage of the column or page the table is to fill.
- c. The frame thickness (thick) is measured from the outside edge of the table boundary inwards. Notice that this differs from the other rule separators.
- d. When the continue with row separator (consep) characteristic is turned on, a table row separator appears under the last table row on a column or page and above the first table row on the succeeding column or page. When turned off, both row separators are omitted.

**10.11.20 Standard cell characteristics.** These characteristics apply to cells. They may be specified on any table object and apply to the cells within the scope of that object.

Characteristics:	Values:
Column Separator On	Toggle
Row Separator On	Toggle
Column Separator Width	Size/Distance
Row Separator Width	Size/Distance
Column Separator Style	List (Blank, Single, Bold, Double, Dashed)
Row Separator Style	List (Blank, Single, Bold, Double, Dashed)
Left Margin	Size/Distance
Right Margin	Size/Distance
Top Margin	Size/Distance
Bottom Margin	Size/Distance
Horizontal Alignment	List (Right - Flush Right/Ragged left, Left - Flush Left/Ragged right Center - Centered Justify - Flush Left & Flush Right, Char - A character to align to the left of)
Vertical Alignment	List (Top, Middle, Bottom)
Alignment Character	String
Alignment Character Offset	Integer (percentage)
Reverse	Toggle
Background Color	List (black, white, red, orange, yellow, green, blue, violet, brown, gray)
Shading	Percentage

**MIL-HDBK-28001**

Rotation  
Text Width

Toggle  
Size/Distance

**EXPLANATION:**

- a. When column (colsepon) or row (rowsepon) separators are turned on, they have no effect on the table frame. When column or row separators are turned on and frame is turned off, no rule appears around the outside edges of the table.
- b. Column (colsep) and row (rowsep) separators appear centered over the right boundary and bottom boundary respectively. Thus they always appear between columns or rows. The separators shall appear centered between two columns or rows, such that when a column separator is 1 point thick, .5 points impinge on the width of the column to the left of the separator and .5 points impinge on the width of the column to the right of the separator.
- c. The values for row and column margins for a cell must be greater than half the thickness of any column or row separators that apply to that cell.
- d. Any highlighting characteristics apply to the full cell area, extending through the margins to the cell separators.
- e. For shading, 0 is white, 100 is full color saturation.
- f. When the value for rotation is turned off, the contents of the cell are placed in the same orientation as the table. When the value is turned on, the content of the cell is rotated 90 degrees.
- g. The textwidth characteristic is used only when rotation is enabled. Textwidth specifies the width to which the text is formatted. When unspecified, textwidth is the column width minus the left and right cell margins.
- h. For cells with rotation specified, the cell contents are formatted to the width specified by textwidth using any applicable composition characteristics; the cell contents are rotated; then cell characteristics are applied.
- i. No row with cells containing any graphic or rotated text can be split.

**10.11.20.1 Table subset group characteristics.** These characteristics apply specifically to table subset groups.

Characteristics:

Values:

Number of Columns

Integer

**MIL-HDBK-28001****EXPLANATION:**

- a. The value for number of columns may be obtained from the source document via the fillval characteristic.

**10.11.20.1.1 Table subset characteristics.** These characteristics apply specifically to table subsets.

Characteristics:	Values:
Number of Columns	Integer
Keep - Keep the whole subset together within the specified boundary	List (Column, Page, None)
Subset Type	List (Heading, Footing, Body)

**EXPLANATION:**

- a. The value for number of columns may be obtained from the source document instance via the fillval characteristic.
- b. The keep characteristic indicates the boundaries across which the subset cannot break either a column or page boundary. Specifying column implies also keeping within the page.
- c. The subset type characteristic indicates to which of the three table subset types to map this element.

**10.11.20.1.2 Column characteristics.** These characteristics apply specifically to table columns.

Characteristics:	Values:
Column Width	String
Column Number	String
Column Name	String
Span Name	String
Start Column Name	String
End Column Name	String

**EXPLANATION:**

- a. Column characteristics (colatts) are used either to associate a column width (colwidth) and optional column name (colname) with an implicitly numbered column (colnum) or to associate the starting (namest) and ending (nameend) column names with a span name (spanname). Either column width with column number and option column name

**MIL-HDBK-28001**

should be specified or span name, start column name, and end column name should be specified. The width of span is the sum of the widths of its contained columns.

- b. The value of column number refers to the number of the column in the output structure. It can either be specified in the FOSI or obtained from the source. In a FOSI, a number refers to the implicit column number of a column in the output structure. The keyword #LAST is used to refer to the last (rightmost) column regardless of its implicit number. The keyword #DEFAULT can be used to specify a set of characteristics to be used when no column characteristics are specified for a column, either explicitly in a FOSI or obtained from the source.

The value of column number can also be obtained from an attribute value specified for the *source element that is being mapped to a column* (via a fillval). When this attribute has no value specified in the source, the column number is derived as one greater than the last column number specified (derived or explicitly) within the same table group. If no column number has yet been specified for the current table group, column number is 1.

For example, suppose the colspec source element is mapped to a column and its colnum attribute indicates the column number. If the first colspec element has no value for colnum specified, column number is assumed to be 1. If the next colspec element's colnum attribute has the value 3, column number would be 3 and the column characteristics for column 2 would have to be derived from a column characteristic using the #DEFAULT keyword. If the next colspec element had no value supplied for its colnum attribute, column number would be 4, and so on.

- c. The column width is specified in terms of units of proportional measure plus amount of fixed measure. The specification for the amount of proportional measure is of the form number\*. The fixed measure is specified exactly as any other width, size or distance in the FOSI. The column width specification 5\*+3pt is interpreted as 5 units of proportional measure, plus 3 points. The *proportional measure number* must be an integer.

The unit of proportional measure is calculated by subtracting the sum of all fixed measures for all columns in a table from the width, and dividing the remainder by the sum of the number of proportional units for all columns in the table.

- d. It is an error to fail to provide a column width for all columns in the output structure, either through implicit or explicit reference.
- e. Start column name and end column name refer to names assigned to columns in the output structure via these column characteristics. The span name is associated with this span of columns.
- f. It is an error to specify more than one set of column characteristics for a given column, either through implicit or explicit reference.

**MIL-HDBK-28001**

**10.11.20.1.3 Row characteristics (rowatts).** These characteristics apply specifically to table rows.

Characteristic:	Values:
Break Row	Toggle

**EXPLANATION:**

- a. When tables are allowed to continue across column or page boundaries, they shall normally break between rows. When the break row characteristic is turned on, then a break across a column or page boundary may occur within a row.

**10.11.20.1.4 Cell characteristics (cellatts).** These characteristics apply specifically to table cells.

Characteristics:	Values:
Column Name	String
Span Name	String
Span Depth	Integer

**EXPLANATION:**

- a. Either a column name or span name should be specified. The column name and span name refer to the names assigned to a column or span of columns via the column characteristics. In the case where both are specified, the column name is used. In the case where neither is specified, the text flow rules determine the column in which the cell contents is placed.
- b. The integer value for the span depth (spandep) characteristic designates the number of rows the entry is to straddle.

**10.11.21 Describing footnotes.** There are several aspects to describing footnotes that must be coordinated. The footnote area element, subordinate to the flowing text area in the page description section of the FOSI, has various attributes that describe characteristics of the footnote area such as footnote placement and separators. This footnote area element has subcharacteristics to specify further the formatting of the various footnote separators. Note that the subcharacteristics in the footnote area are not used to affect the formatting of the text of the footnotes themselves.

In the footnote description (ftndesc) section of the FOSI, one gives the e-i-c's that describe the elements (or pseudo-elements) that will cause their contents to be placed in the footnote area of the page on which these elements are used. Associated with each e-i-c in the ftndesc is a fnatts which contains a charlist (minus keeps and span) that specify the formatting of the footnote content itself. More precisely, the contents (e.g., the charlist) of the e-i-c in the ftndesc determines the characteristics for what gets placed in the flowing text area when this e-i-c instance is encountered; the contents (i.e.,

**MIL-HDBK-28001**

the charlist) of the ftnatts associated with this e-i-c determines the characteristics for what gets placed in the footnote area when this e-i-c is encountered. The contents of this e-i-c instance (that is, the source document element's content) is placed into the footnote area under control of the characteristics defined by the ftnatts (unless the ftnatts' charlist uses the suppress characteristic). (The ftnatts' charlist may specify a default environment [envname]; however, the use of inheritance in the ftnatt's charlist or any of its categories will be ignored.)

In the simpler case where the source DTD has defined an element whose content is the footnote text and whose location in the document instance determines where the footnote callout is to be placed, this element would be described in the ftnDESC and no other element is required. Assuming such a DTD element called ftn, the ftnDESC might look like:

```
<ftnDESC>
  <e-i-c gi="ftn">
    <!-- typeset callout in flowtext -->
    <charlist inherit="1">
      <!-- we assume ftnctr has an appropriate 'counter'
           entry in the rsrcDESC -->
      <enumerat
        increm="1"
        enumid="ftnctr">
      <usetext
        source="ftnctr">
      <subchars>
        <font
          inherit="1"
          size="6pt"
          offset="4pt">
        </subchars>
      </e-i-c>
    <ftnatts>
      <!-- typeset superscript and footnote text in footnote area -->
      <charlist>
        <font
          size="8pt">
        <leading
          lead="9pt">
        <!-- we assume the default (docDESC) indent and quadding is
             appropriate for the footnotes -->
        <presp
          minimum="2pt"
          nominal="3pt"
          maximum="4pt">
        <textbrk
          startln="1">
        <usetext
          source="ftnctr"
          placemnt="before">
        <subchars>
          <font
            size="5pt"
            offset="3pt">
          </subchars>
        </usetext>
      </ftnatts>
    </ftnDESC>
```

## MIL-HDBK-28001

A source document fragment such as:

```
<para>This is a paragraph<ftnt>This is a footnote.</ftnt> with
a footnote in it.</para>
```

would cause a superscript number to be placed after the word paragraph in the paragraph and a footnote with an initial superscript number and the text "This is a footnote." to be placed in the footnote area of the current page.

In the more complex situation used in the example DTD (see appendix A), there is one element (ftnote) whose content is the footnote text but whose position in the document instance is irrelevant and another element (ftnref) with no content but whose position determines (1) the location of the callout in the flowing text and (2) the page in whose footnote area the footnote text will be placed. The ftnref element has an IDREF attribute that refers to the ftnote element's ID attribute to allow the ftnref to reference the appropriate footnote text. To accomplish this, the ftnote e-i-c must use the savetext category to save its contents for later use by the ftnref e-i-c. Neither the ftnote e-i-c nor the ftnref e-i-c are special in that they both appear in the styldesc part of the FOSI. The ftnref element must (1) cause the footnote callout number to appear in the flowing text and (2) cause the previously saved footnote text to be placed in the footnote area of the current page. Both effects are accomplished by using (via usetext) the saved footnote text surrounded by the tags of the element (in this case, a pseudo-element) defined in the findesc. Note that this case could use the same findesc as described above (except that, in this case, the ftn element is now a pseudo-element). Furthermore, there would be e-i-c entries in the styldesc for ftnote and ftnref as follows:

```
<e-i-c gi="ftnote">
  <!-- don't typeset anything, but save this element's contents
        as the contents of a <ftnt>...</ftnt> element -->
  <charlist>
    <suppress
      sup="1">
  </charlist>
  <att>
    <fillval
      attname="id"
      fillcat="savetext"
      fillchar="textid">
    <savetext
      conrule="<ftnt>,#CONTENT,</ftnt>"
    </filllist>
  </att>
</e-i-c>
<e-i-c gi="ftnref">
  <!-- reference (via usetext) the previously saved footnote text -->
  <charlist>
</charlist>
  <att>
    <fillval
      attname="xrefid"
      fillcat="usetext"
      fillchar="source">
  </att>
</e-i-c>
```

**MIL-HDBK-28001**

A source document fragment such as:

```
<ftnote id="xyz">This is a footnote.</ftnote>  
<para>This is a paragraph<ftnref xrefid="xyz"> with  
a footnote in it.</para>
```

would cause the same output as in the previous case.

**10.12 Parsing the FOSI.** When the FOSI is completed for the entire document, verify that the encoding conforms to the OS DTD by running an SGML parser. The parser confirms that the SGML syntax is correct and, in the case where characteristic values are a finite list, verifies that correct values have been specified. The SGML parser cannot verify that other values make sense. It is the FOSI author's responsibility to ensure that the FOSI accurately reflects the formatting specification and that all required values have been specified.

## MIL-HDBK-28001

### 11. ELECTRONIC REVIEW

**11.1 Electronic review.** The electronic review declaration set provides the required SGML structures for the review of SGML text documents electronically using SGML for the comments. (Note that this includes commenting on graphics and other entities at the level of the entire entity.) The capability supported by these structures enables reviewers located in diverse environments to make and exchange comments to multiple copies of a document file over a network. The comments may then be sorted, processed, and incorporated into the document by the owner system.

**11.1.1 Electronic review process improvement.** This capability represents a process improvement over the traditional document development process:

- a. The use of SGML throughout the entire document development cycle eliminates redline paper copies and costly conversion cycles that typically occur during document development.
- b. The benefits of the SGML intelligent markup remain accessible throughout.
- c. Time required for delivery and review cycle is reduced.
- d. Text and graphics are maintained in discrete files under their originating processes, facilitating reuse.

**11.1.2 Electronic review comments.** Electronic review comments using SGML are distinguished from comments made using proprietary vendor annotation capabilities by the following characteristics:

- a. The comments may be associated with elements at any level in the document structure.
- b. The comments may be shared between different proprietary platforms.
- c. The comments are machine-processable for a variety of purposes, some of which may be user-defined.

**11.1.3 Electronic review declaration set overview.** The electronic review declaration set consists of a portable electronic review toolkit suitable for incorporation into any document type declaration, for use in review of any document instance of that type. The structures have been defined as generically as possible, in order to take many kinds of review into account: e.g., internal contractor reviews, Government reviews, contractor/Government reviews, and specification reviews (see appendix D). Provision has been made for the definition of user-defined values for categories of review information. Additional thought has been given to processing scenarios involving use of the declaration set on a range of platforms, with automation of functions supported by the SGML structures varying to great degree. Consequently, the declaration set has been designed to support the preparation of comments (modification requests, or modreqs ) that may be stored in either of the following ways:

**MIL-HDBK-28001**

- a. Within a modification request-only document that references the base document instance.
- b. Within the base document instance to which the modification request(s) refers.

**11.2 Electronic review declaration set.** The following lists the declaration set defining the required SGML structures for the review of SGML text documents electronically using SGML for the comments (modification requests). This declaration set may optionally be used in the review of DoD publications conforming to this specification.

<!--The following declarations may be referred to using a public entity as follows:

```
<!ENTITY % ereview
    PUBLIC "-//USA-DOD//DTD SUP MIL-PRF-28001 EREVIEW REV B//EN">
```

-->

<!-- The %mrinfo entity is required for support of the electronic review declaration set. Note that this entity matches an identical set of elements in the base document being reviewed, and may therefore require tailoring accordingly. Using the example DTD in appendix A, the %mrinfo entity is declared as follows:

```
<!ENTITY % mrinfo      "(pubno+, (revnum|(chgnum, chgdate)|pubdate))" >
```

-->

<!-- The %mrtext entity indicates what elements from the base DTD can occur in the textual (i.e., mrpara and mritem) elements in a modreq. Using the example DTD in appendix A, the %mrtext entity is declared as follows:

```
<!ENTITY % mrtext      "#PCDATA | symbol" >
```

-->

<!-- The %mrelems entity indicates what elements from the base DTD can occur within mrreason, mrinstr, mrgeomod, and mrrespns in a modreq (the mrpara and mrlist elements are generally meant to be included). Using the example DTD in appendix A, the %mrelems entity is declared as follows:

```
<!ENTITY % mrelems     "mrpara | mrlist | graphic" >
```

-->

<!-- Generic default definitions of %mrinfo, %mrtext, and %mrelems are given below. These are to be replaced by a definition appropriate to the document being reviewed: -->

```
<!ENTITY % mrinfo      "ANY" >
```

```
<!ENTITY % mrtext      "#PCDATA" >
```

```
<!ENTITY % mrelems     " mrpara | mrlist " >
```

<!-- Beginning of modification request declaration set -->

## MIL-HDBK-28001

```

<!ELEMENT modreq      - o      (mrinfo?, mrmmod, mrrespns?) >
<!ATTLIST modreq      id        ID                #REQUIRED
                        xref      NMTOKEN           #IMPLIED
                        refpos    (prexref|postxref|xref) "xref"
                        by        CDATA             #REQUIRED
                        date      CDATA             #REQUIRED
                        organiz    NMTOKEN           #IMPLIED
                        orgcat     NMTOKEN           #IMPLIED
                        cmntrcat   NMTOKEN           #IMPLIED
                        priority    (1|2|3|4|5)       #IMPLIED
                        category    NMTOKEN           #IMPLIED
                        topic      CDATA             #IMPLIED
>
<!ELEMENT mrinfo      - o      %mrinfo; >
<!ELEMENT mrmmod      - -      (mrreason?, (mrngenmod|(mrinstr?, mrchgtxt))) >
<!ELEMENT (mrreason|mrinstr|mrngenmod) - o      (%mrelems;)+ >
<!ELEMENT mrchgtxt    - -      ANY >
<!ATTLIST mrchgtxt     chgloc     NUMBER           #IMPLIED
                        chglen     NUMBER           #IMPLIED
                        action      (insert|delete|replace) "replace" >
<!ELEMENT mrrespns    - -      (%mrelems;)* >
<!ATTLIST mrrespns     disposn    NMTOKEN           #IMPLIED
                        status     NMTOKEN           #IMPLIED >
<!ELEMENT (mrpara|mritem) - o      (%mrtext;) >
<!ELEMENT mrlist      - -      (mritem+)* >

<!-- End of modification request declaration set -->

```

11.3 Using the electronic review toolkit. The electronic review declaration set consists of a stand-alone toolkit that may be referenced as a public entity for use with a given document type declaration, with little or no change to that document type declaration.

Use of the toolkit requires:

- a. Unique identifiers on elements (at least to the level of granularity included in the review).
- b. Redefinition of the document level element via parameterization.

An example of referencing the electronic review toolkit for use with the example DTD in appendix A, without change to the publicly registered DTD, is as follows:

```

<!DOCTYPE doc PUBLIC "-//USA-DOD//DTD EXAMPLE MIL-HDBK-28001//EN"
[
<!-- The following parameter entity contains information identifying the
review document. This information matches the identical set of elements in
the review document. -->
<!ENTITY % mrinfo "(pubno+, (revnum | (chgnum, chgdate) | pubdate))*">

```

**MIL-HDBK-28001**

<!-- The following parameter entity indicates what elements from the base DTD can occur in the "textual" (i.e., mrpara and mritem) elements in a modreq. -->

```
<!ENTITY % mrtext "#PCDATA | symbol" >
```

<!-- The following parameter entity indicates what elements from the base DTD can occur along with the mrpara and mrlist elements (e.g., within mrreason, mrinstr, mrngenmod, and mrrespns) in a modreq. -->

```
<!ENTITY % mrelems "mrpara | mrlist | graphic" >
```

```
<!ENTITY % review
```

```
    PUBLIC "-//USA-DOD//DTD SUP MIL-PRF-28001 EREVIEW REV B//EN">
%review;
```

<!-- DOC CONTENT MODEL REDEFINITION -->

```
<!ENTITY % doc "((mrinfo, modreq*) | (volume+ | docpart+ | (front?, body?, rear?)))">
```

```
<!ENTITY % docexpt "ftnote | pgbrk | brk | arbtext | hrule | modreq">
```

```
]>
```

Cases in which a fine-grained review is required may necessitate adding more id attributes to a publicly registered DTD. It is presumed that any such changes would be made to a working version, however, without impact on final deliveries.

In cases involving very large documents, it may be necessary to have a temporary SGML declaration that accompanies the temporary working DTD, in order to accommodate the ID Capacity (IDCAP) and/or the IDREF Capacity (IDREFCAP).

**11.4 Electronic review functionality.** Examples of functionality that can be supported by the modification request structures are as follows:

- a. Sorting of all comments to a given text element. (Note that this processing adds a requirement to the development of the base document instance: the assignment of unique identifiers to all document elements with which reviewers may be required to associate comments.)
- b. Sorting the comments on the basis of comment ID information: e.g., reviewer, date, organization, priority, classification, and category.
- c. Sorting comments related to a particular user-defined topic or concept.
- d. Tracking the comments on the basis of configuration/version information about the comments, and compiling a history file of comments (including disposition and status) from a given review, or from all reviews of a given document.
- e. Tracking comments on comments.

**MIL-HDBK-28001**

- f. Tracking the disposition determined as appropriate for each comment by the responsible organization.
- g. In the case of new text, replacement text, or a deletion being proposed by a reviewer, supporting the following:
  - 1) Marking in some manner (e.g., highlighting) the precise portion of the identified element that is to be affected by the change, and
  - 2) Tracking the location information in the modreq. In the case of a modreq proposing new text, replacement text, or a deletion that is being evaluated by a document owner, displaying the precise area affected by the change on the receiving system, on the basis of the location information in the modreq.
- h. In the case of general modifications to a document being proposed by reviewers (any proposed changes other than ones in which specific proposed new text or replacement text is suggested), sorting the general modifications for evaluation and manual processing by the responsible organization.
- i. Tracking the status of the owner organization's response to the comment.

**11.5 Electronic review user interface.** While the modreq declaration set may be used with only minimal automation, it can support user interfaces providing a high degree of automation of the electronic review functions. For example, in the case where modreqs are written to a separate modreq-only file, the declaration set could support a window application which prompts the user to fill in a template with information for a particular modreq, hiding the tags. Similarly, attribute values for the modreq might be set via menu selections or use of a dialogue box. In the case where modreqs are stored in the document instance, the declaration set might support automated suppression/deletion of the modreqs prior to final CDRL delivery. In either case, features for configuration control of the modreqs could include a data base of modreqs which could be compiled into a history file for a given review or for all reviews of the document.

**11.6 Electronic review process.** While the details of the electronic review process are not specified, implementors will find it helpful to keep in mind a typical process for the review of an SGML text document over an electronic network using SGML for the comments, as described below. Notice that clear distinction between the two required roles of document reviewer and document owner is critical to the smooth conduct of electronic review. Also note that extremely fine details of timing offer numerous possibilities, to be determined by the implementation.

- a. The owner of the review document (e.g., organization or author) establishes those attribute values for a modreq that are designed to be user-defined, then documents and distributes these values as the operative attribute values for a given review or for general use, along with any other review procedures required.
- b. The owner of the review document instance assigns unique identifiers to every element (or to every element included in the review, since use of the SGML electronic review

**MIL-HDBK-28001**

capability requires unique identifiers on elements). This assignment of unique identifiers may be automated or semi-automated by implementations of vendor software.

- c. A copy of the review document is transmitted electronically over a network, or made accessible via a distributed file system, to the receiving organization(s) or individual(s).
- d. The reviewer brings up the document on the receiving system, most likely in read-only form, and begins the process of evaluation.
- e. The reviewer writes modreqs, and optionally provides reasons for the following types of proposed changes to the review document:
  - 1) Change text, which consists of new text, replacement text, or a text element deletion, or
  - 2) A general modification to the document (i.e., any proposed change other than one in which specific new text, replacement text, or a deletion is suggested).

In either case, a unique identifier is assigned to each modreq, and the appropriate attribute values are set for the review information for each modreq. Where the implementation allows, the modreqs are written to a separate modreq-only file. Optionally, the modreqs may also be stored within the review document instance (the actual content of the review document that contains the modreqs is not changed).

- f. The receiving system transmits electronically (or makes accessible via a distributed file system) to the owner system the modreq-only file, or, optionally, the review document in which modreqs have been embedded.
- g. In the case of modreqs transmitted by multiple reviewers, the owner system performs ID resolution of the various types of IDs assigned to the modreqs. (Alternatively, a common convention for the creation of modreq IDs should be promulgated by the owner organization prior to commencement of the review.)
- h. The owner evaluates the modreqs, determining
  - 1) The disposition for each, and
  - 2) The initial status of the response to each.

The corresponding attribute values are then set.

- i. The owner organization incorporates the comments as appropriate, thereby creating the new version of the base document. This is accomplished either by automated incorporation of new text, replacement text, or a text element deletion for which an

## **MIL-HDBK-28001**

approved status is indicated, or by manual incorporation of changes resulting from an approved general modification.

- j. The history file for the modreqs is updated with the final status of each modreq from the review.

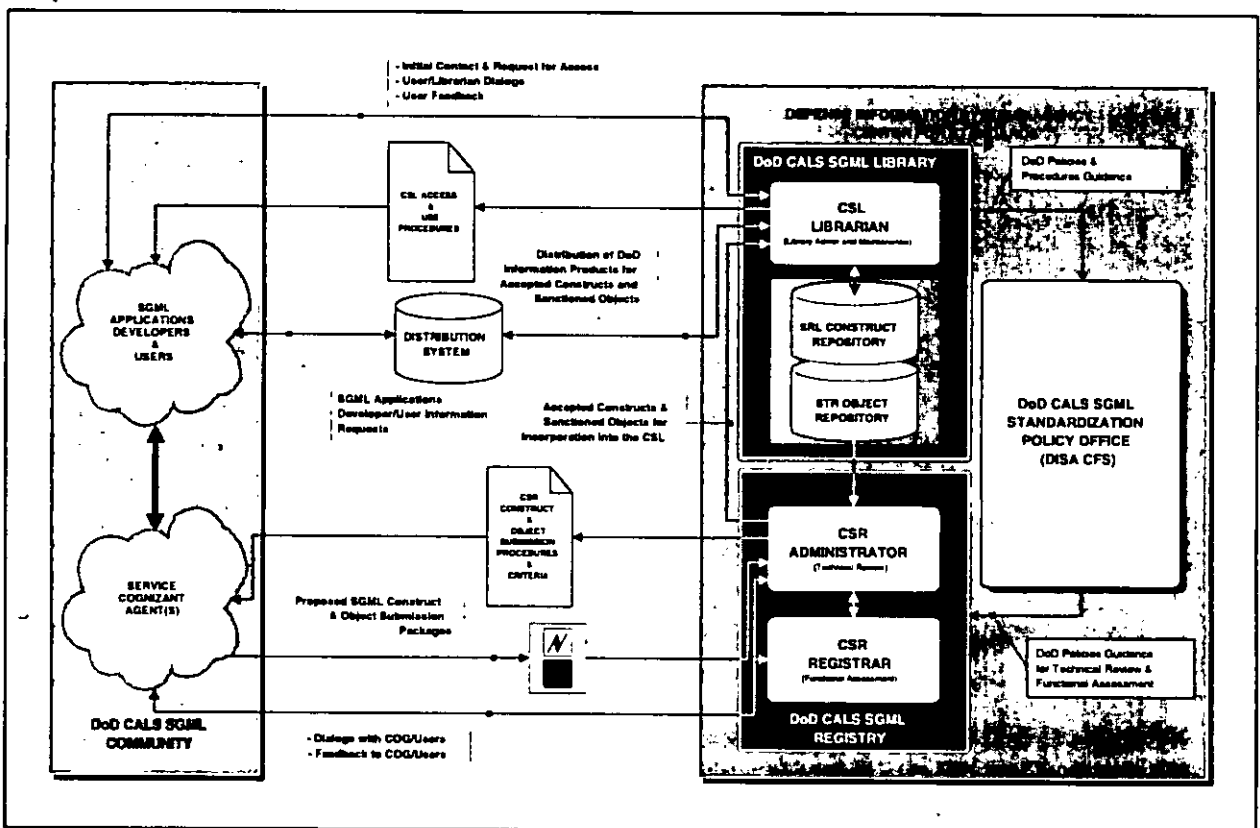
**11.7 Modreq tag set description. See appendix D.**

## MIL-HDBK-28001

## 12. SGML REUSE LIBRARY (SRL) AND SGML TAGSET REGISTRY (STR)

**12.1 Background.** The SGML Tagset Registry and the SRL are being created by Defense Information Systems Agency (DISA) Center For Standards (CFS) to promote the efficient use of SGML constructs and objects in the DoD. DoD wide standardization of SGML constructs (DTDs, FOSIs, and their fragments) and objects (elements, entities, attributes of elements, public identifiers, notations, and standard tagging schemes) facilitates reuse and saves DTD development time and source file tagging time by using familiar markup rules.

**12.2 DoD SGML standardization and reuse program.** The DoD SGML Standardization and Reuse Program establishes and implements the STR and the SRL. This program is organized as pictured in figure 16. The following paragraphs describe the principal functions unique to the program.



**FIGURE 16.** DoD SGML standardization and reuse program organization.

- a. CALS SGML Standardization Policy Office (CSSPO). The CSSPO is a function of DISA CFS responsible for implementation and management of the DoD SGML standardization and reuse program. As such the CSSPO develops, codifies, and disseminates DoD policies and procedures to facilitate the program objectives. For

## MIL-HDBK-28001

more information about the SGML standardization and reuse program, contact DISA CFS at the address below:

ATTN: CALS Digital Standards Office  
DISA, Center For Standards  
Code: JIEO/JEBEB  
10701 Parkridge Blvd.  
Reston, VA 22091-4398

PHONE Number: (703)735-3568 DSN: 653-3568  
FAX Number: (703)735-3257 DSN: 653-3257

- b. **CALS SGML Registry (CSR).** The CSR is the sanctioning authority for all DoD standard SGML constructs and objects. The CSR consists of two functional areas: the *CSR Administrator* and the *CSR Registrar*.
- c. **CSR Administrator.** The CSR Administrator reviews proposed SGML constructs and objects to ensure compliance with submission criteria and that they are technically correct in accordance with MIL-PRF-28001. The CSR Administrator distributes proposed SGML objects to the membership of the CSR Registrar for the functional assessment and coordination.
- d. **CSR Registrar.** The CSR Registrar conducts functional assessment of proposed SGML objects. CSR Registrar membership includes the DoD service and agency custodians for MIL-PRF-28001 and other military standardization documents specifying SGML constructs and objects, other DoD functional areas developing and utilizing SGML applications, and DISA CFS.
- e. **CALS SGML Library (CSL).** (See 12.3).

**12.3 CALS SGML Library (CSL).** The CSL is the central SGML data repository for the DoD SGML standardization and reuse program. The CSL provides for the application of CSSPO policies and procedures, the maintenance of data and information for the DoD SGML standardization and reuse program, the technical support of the CSR function, and the distribution of SGML constructs and objects to SGML applications developers and users. The CSL consists of three sub functions, the *CSL Librarian*, *SGML Reuse Library (SRL)*, and *SGML Tagset Registry (STR)*.

**12.3.1 CSL Librarian.** The CSL librarian is the focal point for operation of the CSL, STR, and SRL. Librarian functions provide management for the STR and SRL repositories and the supporting communications infrastructure. Librarian functions include:

- a. Implementing the policies and procedures established by the CSSPO for the operation and maintenance of the CSL repositories.
- b. Developing and distributing internal procedures and instructions for the operation and use of the CSL repositories.

**MIL-HDBK-28001**

- c. Developing, distributing, and maintaining application software for the operation and maintenance of the CSL repositories.
- d. Developing, distributing, and maintaining application software for the use of CSL information products.
- e. Providing configuration management for all application software, software documentation, data base documentation, data base structure, and construct and object data within the scope of the CSL repositories.

**12.3.2 SGML Reuse Library (SRL).** The SRL is the single-point source for DoD accepted SGML DTDs, FOSIs, and their fragments. The SRL is comprised of a repository and communications infrastructure, operating procedures, and user procedures.

- a. The SRL repository and communications infrastructure resides within the Asset Source for Software Engineering Technology (ASSET) Reuse Library (ARL). The ARL is an Advanced Research Projects Agency (ARPA) funded program to promote the software reuse. ARL users have the capability to search, view, and download SGML constructs through commercial and Government on-line telecommunications services. Access to the SRL repository requires the user to apply through the ARL.
- b. The CSL Librarian develops and implements operating procedures for the SRL in accordance with the policies and requirements established by the CSSPO. These procedures detail the internal operation of the SRL and the maintenance of the SRL repository.
- c. CSL use and access procedures provide users with information to obtain access to the SRL, to use the SRL automated functions, and to submit SGML constructs for inclusion in the SRL.

**12.3.3 SGML Tagset Registry (STR).** The STR is the single-point source for DoD standard SGML objects. Composition of the STR includes the repository infrastructure, operating procedures, and use procedures.

- a. The STR infrastructure consists of a SGML object repository and communications capability.
  - 1. The STR repository is a digital system employing a relational data base management system. This repository consists of the SGML data structures and the supporting data base applications. SGML object data structures house the characteristic information necessary for each registered SGML object. Support software applications provides the capability to use and maintain the information entrusted to the repository. Users access the content of the repository through an information product available through the National Technology Information Service FedWorld Bulletin Board System and File Transfer Protocol server. This product contains the STR data base and the user application to navigate the data.

**MIL-HDBK-28001**

2. The STR communications capability provides the user access to DoD standard SGML objects from Government and commercial on-line services accessible through existing telecommunications capabilities. The use of existing services and telecommunications capabilities ensures STR users will not require new or enhanced communications capabilities.
- b. The CSL librarian develops and documents the STR operating procedures and implements those policies established by the CSSPO for the STR. These procedures detail the internal operation of the STR and the maintenance of the STR repository.
  - c. Use of the STR is supported through the following:
    1. Instructions and procedures to gain access to the STR repository and to locate and retrieve sanctioned SGML object data and the supporting user applications.
    2. Instructions for using SGML object data and the supporting user applications, obtaining help in using an information product, and reporting deficiencies in an information product.
    3. Instructions, procedures, and requirements for submitting new SGML objects to be considered for incorporation as DoD standard SGML objects.

**12.4 Summary of operations.** Figure 16 also shows how the DoD SGML standardization and reuse program works. The following paragraphs describe two possible operational scenarios.

**12.4.1 Scenario 1, direct reuse.** The Direct Reuse scenario demonstrates how an organization uses the CSL to develop a document instance. This scenario begins with the assumption that the document class and the SGML Public Identifier of an existing DTD are known. The document developer accesses the SRL and locates information about the DTD by searching on the DTD's Public Identifier. The developer downloads the information and the DTD from the SRL to use at the developer's work site. The developer accesses the STR to obtain the definitions of the objects declared in the DTD. The developer uses the DTD and the object definitions to create the SGML document instance.

**12.4.2 Scenario 2, modified reuse.** The Modified Reuse scenario demonstrates how an organization uses the SRL and the STR to develop a DTD for a new class of documents. This scenario begins with the assumption that the developer has identified the generic structure and content requirements for this new document class. The developer access the SRL to determine if an existing DTD has features in common with the new document class. To accomplish this, the developer searches the SRL for DTDs by title or key words and reviews DTD descriptive information. The developer may be able to reuse portions of existing DTDs that have features in common with the new document class. The developer then downloads the DTD reuse candidates for further analysis. The developer also uses the STR to obtain the object definitions declared in the candidate DTDs, for use during the analysis.

## **MIL-HDBK-28001**

The developer accomplishes the following activities to further define the DTD. The appropriate portions of candidate DTDs are selected to describe the new document class. Document structure and content requirements that can not be satisfied using the existing DTDs are then identified. The STR is used to access descriptions of standard objects. These descriptions are analyzed to identify standard objects which satisfy some or all of the requirements. Requirements which can not be satisfied by standard objects are defined in detail. These newly defined objects are submitted to the CSR for review and coordination.

The CSR review and coordination process consists of a technical review by CSR Administrator and coordinated through the CSR Registrar for functional issues. This process identifies any technical problems and functional issues for new constructs and objects. This review process should take no longer than 30 days. New constructs and objects are registered only when all technical problems have been resolved and any functional issues are reconciled.

The developer completes the DTD development process using DoD standard SGML objects and reusable DTD constructs. The developer ensures the DTD meets technical and functional requirements of the contract or work agreement, and also ensures that the DTD complies with MIL-PRF-28001. The developer submit the validated and verified DTD to the CSR Administrator for access through the SRL.

The CSR Administrator checks the DTD to ensure that only standard objects are used. The administrator also verifies that the DTD developer has certified that the DTD meets the technical and functional requirements of the contract or work agreement, and that the DTD is MIL-PRF-28001 compliant. Once the DTD information is validated by the administrator, the DTD is made available for access through the SRL.

## MIL-HDBK-28001

## APPENDIX A

## EXAMPLE DOCUMENT TYPE DEFINITION

## 10. SCOPE

10.1 Scope. This appendix contains an example document type definition and two tables containing descriptions of its elements, attributes and attribute sets.

10.2 Example DTD. This is an example only and should not be used for acquisition purposes.

```
<!-- Start of Example DTD -->
```

```
<!-- The following set of declarations may be referred to using a public
entity as follows: -->
```

```
<!-- Depending on the software being used, the user may need to add the CALS
SGML Declaration to this DTD or refer to it in an external reference. -->
```

```
<!-- The following set of declarations may be referred to using a public
entity as follows:
```

```
<!DOCTYPE doc PUBLIC "-//USA-DOD//DTD EXAMPLE MIL-HDBK-28001//EN" > -->
```

```
<!-- NOTE: In order to parse the following Document Type Declaration Subset
alone, insert "<!DOCTYPE doc [" at the beginning of the file and "]">" at the
end of the file. -->
```

```
<!-- ENTITY DECLARATIONS -->
```

```
<!ENTITY % doc "volume+ | docpart+ | (front?, body?, rear?)" >
```

```
<!ENTITY % docexpt "ftnote | pgbrk | brk | arbtext | hrule" >
```

```
<!-- %service is the declared value of the service attribute of <doc>. It may
be redefined to include additional or remove existing acceptable values for
this attribute. -->
```

```
<!ENTITY % service "af | navy | army | mc | dla | cg" >
```

```
<!-- The %docatt entity may be redefined to provide additional attributes for
the document level element as required.-->
```

<!ENTITY % docatt	'service	(&service;)	#REQUIRED
	docid	CDATA	#IMPLIED
	version	(basic   revision   change)	"basic"
	docstat	(prelim   draft   formal)	"formal"
	mantype	(standard   card   decal)	"standard" ' >

```
<!-- MATH PACKAGE INCLUSION: To include the standard math package in a
document, include in the document's document type declaration subset the
following declaration:
```

```
<!ENTITY % math "INCLUDE" > -->
```

## MIL-HDBK-28001

## APPENDIX A

```

<!ENTITY % math "IGNORE" >

<![ %math; [

<!ENTITY % mathpac PUBLIC
"-//USA-DOD//DTD SUP MIL-HDBK-28001 MATHPAC 911001//EN" >

%mathpac;]]>

<!-- NOTE: Additionally required character sets must be explicitly designated
in the document's document type declaration subset. -->

<!-- The following entity is referenced in %text; -->

<![ %math; [

<!ENTITY % mathtxt " | dfref | f " >

<!-- only if %math; is "include" -->

]]>

<!ENTITY % mathtxt "" >

<!-- otherwise -->

<!-- The following entity is referenced in %paracon; -->

<![ %math; [

<!ENTITY % mathcon " | df | dfg " >

<!-- only if %math; is "include" -->

]]>

<!ENTITY % mathcon "" >

<!-- ATTRIBUTE DEFINITION COLLECTIONS AND PARTS THEREOF -->

<!-- Many attributes have a Boolean value. They are uniformly given the
declared value "%yesorno;" rather than NUMBER to indicate this intent. 0 is
interpreted as false; all other numbers as true. -->

<!ENTITY % yesorno "NUMBER" >

<!-- The itemid attribute group provides the ability to describe the text to
which the attribute group pertains by the identifiers associated with the part
to which the text refers. This group is also used within the standard body
attributes (described below). -->

<!ENTITY    % itemid    "sssn    CDATA          #IMPLIED
                        unit     CDATA          #IMPLIED
                        module    CDATA          #IMPLIED
                        lru       CDATA          #IMPLIED
                        assem     CDATA          #IMPLIED
                        subassem  CDATA          #IMPLIED

```

## MIL-HDBK-28001

## APPENDIX A

ssubassm	CDATA	#IMPLIED
compon	CDATA	#IMPLIED
partno	CDATA	#IMPLIED
refdes	CDATA	#IMPLIED" >

<!-- The content attribute group provides the ability to describe the text to which the attribute group pertains by the type of content, applicability, skilltrack, figures, and tables associated with the text. This group is also used within the standard body attributes (described below). -->

<!ENTITY	% content	"texttype	NUMBER	#IMPLIED
		applictype	IDREFS	#IMPLIED
		applicrefid	IDREFS	#IMPLIED
		skilltrk	NMTOKENS	#IMPLIED
		contyp	(desc   proc)	#IMPLIED
		assocfig	IDREFS	#IMPLIED
		assoctab	IDREFS	#IMPLIED" >

<!-- Some elements get a collection of attributes known collectively as body attributes. The %bodyatt entity contains all of the appropriate attribute definitions. -->

<!ENTITY	% bodyatt	'id	ID	#IMPLIED
		inschlvl	NUTOKENS	#IMPLIED
		delchlvl	NUTOKENS	#IMPLIED
		label	CDATA	#IMPLIED
		hcp	%yesorno;	"0"
		stub	(STUB)	#CONREF
		%itemid;		
		%content;	' >	

<!-- Many elements get a security-related collection of attributes. The %secur entity contains all of the appropriate attribute definitions. -->

<!ENTITY	% secur	'security	(u   c   s   ts)	#IMPLIED
		restrict	NMTOKENS	#IMPLIED
		release	NMTOKENS	#IMPLIED
		codeword	NMTOKENS	#IMPLIED
		scilevel	%yesorno;	"0"
		diglyph	NMTOKENS	#IMPLIED' >

<!-- %erptype is the declared value of the erptype attribute of <errpt>. It may be redefined to include additional or remove existing acceptable values for this attribute. -->

<!ENTITY % erptype "tpdr | afto22 | da2028" >

<!-- %notctype is the declared value of the notctype attribute of <notice>. It may be redefined to include additional or remove existing acceptable values for this attribute. -->

<!ENTITY % notctype "dist | auth | fouo | branch | pgclass | disclos | supersed | effdate | suppl | nopg | noclaspg | warning | destr | safesup | opersup | maintsup" >

## MIL-HDBK-28001

## APPENDIX A

<!-- %sigtype is the declared value of the sigtype attribute of <sigblk>. It may be redefined to include additional or remove existing acceptable values for this attribute. -->

<!ENTITY % sigtype "preparer | approval | review | concur | other" >

<!-- ELEMENT TYPE COLLECTIONS AND MODEL GROUPS -->

<!-- TITLES -->

<!-- Some elements which have either required or optional titles may at times also require shortened forms of the title. If shortened titles are to be allowed in the instance then the parameter entity %shortitleuse; should be redefined as "include". -->

<!ENTITY % shortitleuse 'IGNORE' >

<![ %shortitleuse; [

<!ENTITY % shortitle ", shortttile?">]]>

<!ENTITY % shortitle "" >

<!-- RUNNING TEXT -->

<!-- Various numbers embedded in running text are tagged to permit easy identification for database work. They generally have no special display formatting requirements. -->

<!ENTITY % nums "(partno | serno | partdesc | smrcode | nsn | modelno | sssn | refdes | docno | figindex | lin | faultcode)" >

<!-- NOTE: The following parameter entity declaration contains one or more parameter entity references. This entity declaration must be preceded by the entity declaration for these referenced entities. -->

<!ENTITY % text "(#PCDATA | ftnref | xref | indxflag | verbatim | emergency | change | emphasis | applicabil | symbol | subscript | superscript | %nums; | tool | testeq | material | torqueval | extref | dataiden %mathxt;)+>" >

<!-- PARAGRAPH CONTENT -->

<!-- Various types of lists can occur within the body of a paragraph, and generally where one can occur, so can any other type. -->

<!ENTITY % list "(seqlist | randlist | deflist)" >

<!-- Unnumbered paragraph content consists of text, with optionally intermixed lists, applicability definitions (and math displays, if the math package is included). -->

<!-- NOTE: The following parameter entity declaration contains one or more parameter entity references. This entity declaration must be preceded by the entity declaration for these referenced entities. -->

<!ENTITY % paracon "(%text; | %list; | applicdef %mathcon;)+>" >

## MIL-HDBK-28001

## APPENDIX A

<!-- (UNNUMBERED) PARAGRAPHS AND PARAGRAPH-LIKE ELEMENTS -->

<!-- Special paragraphs usually are just an appropriately labelled paragraph, but in certain cases they can have more than one paragraph within them. -->

<!ENTITY % spcpara "(warning | caution | note)" >

<!-- NUMBERED/TITLED "PARAGRAPHS" AND OTHER SUBSECTION-LIKE ELEMENTS -->

<!-- Step content consists of optional warnings, cautions, and notes (in that order, and applying to the following paragraphs), and then an unnumbered paragraph, followed optionally by notes. Numbered paragraph content consists of a title, the same special and unnumbered paragraphs followed optionally by notes as are in step content, and finally optional steps. -->

<!ENTITY % stepcon "(specpara | para)+>" >

<!ENTITY % titles "(title %shorttitle;)" >

<!-- NOTE: The following parameter entity declaration contains one or more parameter entity references. This entity declaration must be preceded by the entity declaration for these referenced entities. -->

<!ENTITY % nparcon "(%titles;, (specpara | para)+)" >

<!-- FRONT, BODY, REAR MATTER ELEMENTS -->

<!-- The content models for <front>, <idinfo>, <section>, and <rear> are entities so that they can be redefined. -->

<!ENTITY % frnt "((idinfo | warnsum | chgsheet | lep | promul | chgrec | foreword | preface | intro | contents | illuslist | tablelist | safesum | howtouse )+)" >

<!ENTITY % idinf "((pubno | prepubno | nsn | chgnum | revnum | titleblk | mfr | contractno | docmfr | seal | notice | downgrd | pubdate | chgdate)+)" >

<!-- %chgsht is a parameter entity reference for the content model of the element type chgsheet. It is used as is or it may be changed for use with a specific class of documents. An example of how it may be changed would be if the system were to generate the change sheet. Then the content model would be changed to a declared content of EMPTY. -->

<!ENTITY % chgsht "(chgnum, address, date, prttitle, para?, chglist)" >

<!-- NOTE: The following parameter entity declaration contains one or more parameter entity references. This entity declaration must be preceded by the entity declaration for these referenced entities. -->

<!ENTITY % sect "(%titles;, para0)" >

<!ENTITY % rr "((appendix | glossary | index | errpt | foldsect)+)" >

<!-- MISCELLANEOUS -->

## MIL-HDBK-28001

## APPENDIX A

<!-- SPECIAL CHARACTER SETS -->

<!-- The following public character entity sets are required to meet the general requirements of most service applications. Exceptional character requirements may be met by selecting additional public character entity sets from MIL-PRF-28001 appendix C. Those exceptional requirements must be separately specified in the contract. -->

<!ENTITY % ISOLat1 PUBLIC "ISO 8879:1986//ENTITIES Added Latin 1//EN">

<!ENTITY % ISOpub PUBLIC "ISO 8879:1986//ENTITIES Publishing//EN">

<!ENTITY % ISOgrk3 PUBLIC "ISO 8879:1986//ENTITIES Greek Symbols//EN">

<!ENTITY % ISOnum PUBLIC  
"ISO 8879:1986//ENTITIES Numeric and Special Graphic//EN" >

<!ENTITY % ISotech PUBLIC  
"ISO 8879:1986//ENTITIES General Technical//EN" >

%ISOLat1; %ISOpub; %ISOgrk3; %ISOnum; %ISotech;

<!-- ELEMENT DEFINITIONS -->

<!-- BIG ELEMENTS (BIGGER THAN FRONT MATTER, BODY, OR REAR MATTER) -->

<!-- A document contains volumes, a volume contains parts, a part has front matter, body, and rear matter. -->

<!ELEMENT doc - - (%doc;) +(%docexpt;) >

<!ATTLIST doc %docatt; %secur; >

<!ELEMENT volume - - ((docpart, docpart+) | rear?) >

<!ATTLIST volume tocentry %yesorno; "1" %bodyatt; %secur; >

<!ELEMENT docpart - - (front?, body?, rear?) >

<!ATTLIST docpart %bodyatt; %secur; >

<!-- FRONT MATTER AND ELEMENTS PECULIAR THERETO -->

<!-- Front matter contains identifying information for the document: title and cover pages, foreword, various lists, and various special-purpose types of information interspersed. The %frnt; entity permits specialization to a particular variant DOCTYPE. -->

<!-- entity % frnt    "(idinfo | warnsum | chgsheet | lep | promul | chgrec |  
                      foreword | preface | intro | contents | illuslist |  
                      tablelist | safesum | howtouse )+" -->

<!ELEMENT front - - (%frnt;) >

<!ATTLIST front %secur; >

## MIL-HDBK-28001

## APPENDIX A

```

<!-- entity % idinf  "(pubno | prepubno | nsn | chgnum | revnum | titleblk |
                      mfr | contractno | docmfr | seal | notice | downgrd |
                      pubdate | chgdate)+" -->

<!ELEMENT idinfo - - (%idinf;) >
<!ATTLIST idinfo %secur; >

<!ELEMENT (pubno | prepubno) - o (user?, docno)+ >
<!ATTLIST (pubno | prepubno) %secur; >

<!ELEMENT user - - (%text;) >
<!ATTLIST user %secur; >

<!ELEMENT titleblk - - (volnum?, docpartn?, revnum?, doctype, maintlvl*,
                        prttitle, stitle?) >

<!ATTLIST titleblk %secur; >

<!ELEMENT (volnum | docpartn | revnum | doctype | maintlvl | chgnum) - o
          (%text;) >

<!ATTLIST (volnum | docpartn | revnum | doctype | maintlvl | chgnum)
          %secur;>

<!ELEMENT prttitle - - (nomen, eqpttype?, (pslist | partno | serno | modelno
                        | nsn)*, subject?) >

<!ATTLIST prttitle %secur; >

<!ELEMENT nomen - - (%text;) >
<!ATTLIST nomen %secur; >

<!ELEMENT eqpttype - - (%text;) >
<!ATTLIST eqpttype %secur; >

<!ELEMENT pslist - - (partno, serno)+ >
<!ATTLIST pslist %secur; >

<!-- partno, serno, modelno, nsn found in %nums under "TEXT". -->

<!ELEMENT subject - - (%text;) >
<!ATTLIST subject %secur; >

<!-- end <prttitle>, continue <titleblk> -->

<!ELEMENT stitle - o (%text;) >
<!ATTLIST stitle %secur; >

<!-- end <titleblk>, continue <idinfo> -->

```

## MIL-HDBK-28001

## APPENDIX A

```

<!ELEMENT (mfr | contractno | docmfr) - o (%text;) >

<!ATTLIST (mfr | contractno | docmfr) %secur; >

<!ELEMENT seal - o (graphic) >

<!-- A <notice> usually contains standard text as defined in entity
declarations in MIL-PRF-28001, e.g., <notice
notctype="dist">&distrib;</notice>. If a notice containing other text is
needed, the text should be directly included as the content of the <notice>
element, e.g.: <notice notctype="auth">Published by Authority of the
Secretary of Defense</notice>. -->

<!ELEMENT notice - o (para+) +(table) >

<!ATTLIST notice notctype (%notctype;) #IMPLIED %secur; >

<!ELEMENT downgrd - o (authrty | phrase | oadr | date)+ >

<!ATTLIST downgrd %secur; >

<!ELEMENT authrty - o (%text;) >

<!ATTLIST authrty %secur; >

<!ELEMENT (phrase | oadr) - o (%text;) >

<!ATTLIST (phrase | oadr) %secur; >

<!-- end <downgrd>, continue <idinfo> -->

<!ELEMENT (pubdate | chgdate) - o (%text;) >

<!ATTLIST (pubdate | chgdate) %secur; >

<!-- end <idinfo>, continue <front> -->

<!ELEMENT lep - o EMPTY >

<!ELEMENT warnsum - o (para0 | para | warning)+ >

<!ATTLIST warnsum inschlvl      NUTOKENS      #IMPLIED
                  delchlvl      NUTOKENS      #IMPLIED
                  tocentry      %yesorno      "0"
                  %secur; >

<!ELEMENT chgsheet - o (%chgsht;) >

<!ATTLIST chgsheet %secur; >

<!ELEMENT chglist - o (removepg, insertpg)+ >

<!ATTLIST chglist %secur; >

<!ELEMENT (removepg | insertpg) - o (#PCDATA) >

<!ATTLIST (removepg | insertpg) %secur; >

```

## MIL-HDBK-28001

## APPENDIX A

```

<!ELEMENT promul - - (title?, para*, (sigblk | graphic)*) >
<!ATTLIST promul %bodyatt; %secur; >

<!ELEMENT chgrec - o (table | graphic) >
<!ATTLIST chgrec %bodyatt; %secur; >

<!ELEMENT (foreword| preface | intro) - o (para0 | para | symbsect | abbrsect
| %spcpara; | internatlstd |
sigblk)+ +(figure | table | chart) >

<!ATTLIST (foreword| preface | intro) tocentry %yesorno; "1"
%bodyatt;
%secur; >

<!ELEMENT (symbsect| abbrsect) - o (deflist)+ >
<!ATTLIST (symbsect| abbrsect) %bodyatt %secur; >

<!ELEMENT internatlstd - o (para+) >
<!ATTLIST internatlstd %secur; >

<!ELEMENT sigblk - o (purpose | graphic | signer | position | organiz |
address | date)+ >

<!ATTLIST sigblk sigtype (%sigtype;) #IMPLIED %secur; >

<!ELEMENT purpose - o (%text;) >
<!ATTLIST purpose %secur; >

<!ELEMENT signer - o (%text;) >
<!ATTLIST signer %secur; >

<!ELEMENT (position| organiz | address) - o (%text;) >
<!ATTLIST (position | organiz | address) %secur; >

<!-- end <sigblk> and <foreword>, continue <front> -->

<!ELEMENT howtouse - o ( %sect; | para0 | para+) >
<!ATTLIST howtouse tocentry %yesorno; "0"
shortentry %yesorno; "0"
%bodyatt;
%secur; >

<!ELEMENT contents - o EMPTY >
<!ATTLIST contents shortentry %yesorno; "0" >

<!ELEMENT (illuslist| tablelist) - o EMPTY >
<!ATTLIST (illuslist | tablelist ) tocentry %yesorno; "1"

```

## MIL-HDBK-28001

## APPENDIX A

```

shortentry %yesorno; "0" >

<!ELEMENT safesum - o (para | precaut | warning | caution)+ >

<!ATTLIST safesum      tocentry          %yesorno;    "0"
                       shortentry        %yesorno;    "0"
                       %bodyatt;
                       %secur; >

<!ELEMENT precaut - o (%text;) >

<!ATTLIST precaut %secur; >

<!-- BODY AND ELEMENTS PECULIAR THERETO -->

<!-- The body contains two or more chapters, a chapter contains two or more
sections, a section contains numbered paragraphs. If there is only one, its
content may be used as the content of the next higher level, except that a
one-chapter body cannot have sections. -->

<!ELEMENT body - - ( chapter | section | ftnsec | para0 | ddunit)+ >

<!ATTLIST body %secur; >

<!ELEMENT chapter - - (%titles;,(section | ftnsec)+ | para0+)) >

<!ATTLIST chapter tocentry %yesorno;    "1"
                  shortentry %yesorno;  "0"
                  %bodyatt;
                  %secur; >

<!--ENTITY % sect "(%titles;, para0+)"> -->

<!ELEMENT (section|ftnsec) - - (%sect;) >

<!ATTLIST (section| ftnsec) tocentry %yesorno;    "1"
                           shortentry %yesorno;  "0"
                           %bodyatt;
                           %secur; >

<!ELEMENT ddunit - - (ddintro, ddsheet+) >

<!ATTLIST ddunit portion (section | chapter) #IMPLIED %secur; >

<!ELEMENT ddintro - o (title | dddesc | ddindex)+ >

<!ATTLIST ddintro %secur; >

<!ELEMENT dddesc - o ((para+, para0*) | para0+) >

<!ATTLIST dddesc %secur; >

<!ELEMENT ddindex - o (((para+, para0*) | para0+) | ddlist) >

<!ATTLIST ddindex %secur; >

<!ELEMENT ddlist - o (partno | pgno)+ >

```

## MIL-HDBK-28001

## APPENDIX A

```

<!ATTLIST ddlist %secur; >
<!ELEMENT pgno - - (%text;) >
<!ATTLIST pgno %secur; >
<!ELEMENT ddsheet - - (partname, ( partno | modelno | serno | eqpttype),
                        ((para+, para0*) | para0+)) >
<!ATTLIST ddsheet %secur; >
<!ELEMENT partname - - (%text;) >
<!ATTLIST partname %secur; >
<!-- REAR MATTER AND ELEMENTS PECULIAR THERETO -->
<!-- entity % rr "( appendix | glossary | index | errpt | foldsect)+ " -->
<!ELEMENT rear - - (%rr;) >
<!ATTLIST rear %secur; >
<!ELEMENT appendix - - (%titles;, ((section | ftnsec)+ | para0+)) >
<!ATTLIST appendix  tocentry    %yesorno;    "1"
                    shortentry  %yesorno;    "0"
                    %bodyatt;
                    %secur; >
<!ELEMENT glossary - - (para?, (title, deflist)+) >
<!ATTLIST glossary  tocentry    %yesorno;    "1"
                    shortentry  %yesorno;    "0"
                    %secur; >
<!ELEMENT index - o EMPTY >
<!ATTLIST index      shortentry  %yesorno;    "0" >
<!ELEMENT errpt - o EMPTY >
<!ATTLIST errpt      erptype     (%erptype;)    #REQUIRED %secur; >
<!ELEMENT foldsect - - (foldout+) >
<!ATTLIST foldsect %secur; >
<!ELEMENT foldout - o (figure | table | chart) >
<!ATTLIST foldout    pgstyle     NUMBER        #IMPLIED %secur; >
<!--NUMBERED/TITLED PARAGRAPHS AND OTHER SUBSECTION-LIKE ELEMENTS-->
<!--<!ENTITY % nparcon "(%titles;, (specpara | para)+, (step1, step1+)?)" -->
<!ELEMENT specpara - - (warning*, caution*, note*, para, note*) >

```

## MIL-HDBK-28001

## APPENDIX A

```

<!ATTLIST specpara %secur; >

<!ELEMENT para0 - o (%nparcon; ,step1*,subpara1*) +(figure | chart | table) >

<!ATTLIST para0    tocentry    %yesorno;    "1"
                   shortentry  %yesorno;    "0"
                   %bodyatt;
                   %secur; >

<!ELEMENT subpara1 - o (%nparcon; ,(step1+ | step2+)?, subpara2*) >

<!ATTLIST    (subpara1 | subpara2 | subpara3 | subpara4 | subpara5 | subpara6
              | subpara7 | subpara8) tocentry %yesorno; "0" shortentry %yesorno;
              "0" %bodyatt; %secur; >

<!ELEMENT subpara2 - o (%nparcon; ,( step1+ | step3+)?, subpara3*) >

<!-- See above for attribute list. -->

<!ELEMENT subpara3 - o (%nparcon; ,(step1+ | step4+)?, subpara4*) >

<!-- See above for attribute list. -->

<!ELEMENT subpara4 - o (%nparcon; ,(step1+ | step5+)?, subpara5*) >

<!-- See above for attribute list. -->

<!ELEMENT subpara5 - o (%nparcon; ,(step1+ | step6+)?, subpara6*) >

<!-- See above for attribute list. -->

<!ELEMENT subpara6 - o (%nparcon; ,(step1+ | step7+)?, subpara7*) >

<!-- See above for attribute list. -->

<!ELEMENT subpara7 - o (%nparcon; ,(step1+ | step8+)?, subpara8*) >

<!-- See above for attribute list. -->

<!ELEMENT subpara8 - o (%nparcon; ,step1* ) >

<!-- See above for attribute list. -->

<!-- ENTITY % stepcon "(specpara | para)+" -->

<!ELEMENT step1 - o (%stepcon; ,step2*) >

<!ATTLIST    (step1 | step2 | step3 | step4 | step5 | step6 | step7 | step8)
              %bodyatt; %secur; >

<!ELEMENT step2 - o (%stepcon; ,step3*) >

<!-- See above for attribute list. -->

<!ELEMENT step3 - o (%stepcon; ,step4*) >

<!-- See above for attribute list. -->

```

.

4

4

## MIL-HDBK-28001

## APPENDIX A

```

<!ELEMENT term - o (%text;) >

<!ATTLIST
    term %secur; >

<!ELEMENT def - o (para+) +(table) >

<!ATTLIST
    def %secur; >

<!ELEMENT applicdef - - (title?, applichd, applicid+) >

<!ATTLIST
    applicdef id ID #REQUIRED %secur; >

<!ELEMENT applichd - o (term, def) >

<!ATTLIST
    applichd %secur; >

<!ELEMENT applicid - o (term, def) >

<!ATTLIST
    applicid id ID #REQUIRED %secur; >

<!-- SPECIAL PARAGRAPH ELEMENTS -->

<!--entity % spcpara "(warning | caution | note)" -->

<!ELEMENT (warning | caution | note) - - (graphic | para | %list;)+
    -(figure | table | chart) >

<!ATTLIST
    warning type CDATA #IMPLIED
    xrefid IDREF #IMPLIED
    vital %yesorno; "0"
    %secur; >

<!ATTLIST
    (caution | note) type CDATA #IMPLIED
    xrefid IDREF #IMPLIED
    %secur; >

<!-- RUNNING TEXT -->

<!-- Various numbers embedded in running text are tagged to permit easy
identification for database work. They generally have no special display
formatting requirements. -->

<!ELEMENT xref - o EMPTY >

<!ATTLIST xref xrefid IDREF #REQUIRED
    xidtype (text | figure | table) #REQUIRED
    pretext CDATA #IMPLIED
    posttext CDATA #IMPLIED
    %secur; >

<!ELEMENT extref - o EMPTY >

<!ATTLIST extref docno CDATA #IMPLIED
    pretext CDATA #IMPLIED
    posttext CDATA #IMPLIED
    %secur; >

```

## MIL-HDBK-28001

## APPENDIX A

<!ELEMENT graphic - o EMPTY >

<!ATTLIST graphic	boardno	ENTITY	#REQUIRED	
	graphsty	NMTOKEN	#IMPLIED	
	llcordra	CDATA	#IMPLIED	
	urcordra	CDATA	#IMPLIED	
	reprowid	NUTOKEN	#IMPLIED	
	reprodep	NUTOKEN	#IMPLIED	
	hscale	NUTOKEN	#IMPLIED	
	vscale	NUTOKEN	#IMPLIED	
	scalefit	%yesorno;	#IMPLIED	
	hplace	(left   right   center   none)	#IMPLIED	
	vplace	(top   middle   bottom   non)	#IMPLIED	
	coordst	CDATA		#IMPLIED
	coordend	CDATA		#IMPLIED
	rotation	NUMBER		#IMPLIED
	%secur;			

<!ELEMENT symbol - o EMPTY >

<!ATTLIST symbol	boardno	ENTITY	#REQUIRED
	reprowid	NUTOKEN	#IMPLIED
	reprodep	NUTOKEN	#IMPLIED
	hscale	NUTOKEN	#IMPLIED
	vscale	NUTOKEN	#IMPLIED
	scalefit	%yesorno;	#IMPLIED
	offset	NUTOKEN	#IMPLIED
	rotation	NUMBER	#IMPLIED
	%secur;		

<!ELEMENT (subscript | superscript) - - RCDATA >

<!ATTLIST (subscript | superscript) %secur; >

<!ELEMENT (tool | testeq | material | torqueval) - - (%text;) >

<!ATTLIST (tool | testeq | material | torqueval) %content; %secur; >

<!ELEMENT dataiden - - (%text;) >

<!ATTLIST dataiden %bodyatt; %secur; >

<!ELEMENT ftnref - o EMPTY >

<!ATTLIST ftnref	xrefid	IDREF	#REQUIRED >
------------------	--------	-------	-------------

<!ELEMENT indxflag - o EMPTY >

<!ATTLIST indxflag	ref1	CDATA	#IMPLIED
	ref2	CDATA	#IMPLIED
	ref3	CDATA	#IMPLIED
	ref4	CDATA	#IMPLIED
	%secur;		

<!ELEMENT verbatim - - RCDATA >

## MIL-HDBK-28001

## APPENDIX A

```

<!ATTLIST verbatim    allowbrk %yesorno;           "1", %secur; >

<!ELEMENT emergency - - (%text;) >

<!ELEMENT change - - (%text;) >

<!ATTLIST change      level      NUMBER            #IMPLIED
                    change (add | delete)          #IMPLIED
                    mar         %yesorno;          #IMPLIED
                    %secur; >

<!ELEMENT emphasis - - (%text;) >

<!ATTLIST emphasis    emph      NAMES              #REQUIRED >

<!ELEMENT applicabil - - (%text;) >

<!ATTLIST applicabil  applicrefid IDREFS            #REQUIRED
                    applictype  IDREFS              #IMPLIED
                    %secur; >

<!-- Various numbers embedded in running text are tagged to permit easy
identification for database work. They generally have no special display
formatting requirements. -->

<!-- entity % nums      "(partno | serno | modelno | nsn | partdesc | smrcode |
                        ssn | refdes | lin | docno | faultcode | figindex)" -->

<!ELEMENT      (partno | serno | modelno | nsn | partdesc | smrcode | ssn |
                refdes | lin | docno) - - (%text;) >

<!ATTLIST      (partno | serno | modelno | nsn | partdesc | smrcode | ssn |
                refdes | lin | docno) %secur; >

<!ELEMENT faultcode - - (%text;) >

<!ATTLIST faultcode  %content; %secur; >

<!ELEMENT figindex - o (xref, callout) >

<!ATTLIST figindex %secur; >

<!ELEMENT callout - - (%text;) >

<!ATTLIST callout assocfig IDREF              #IMPLIED >

<!-- MISCELLANEOUS ELEMENTS -->

<!-- <pgbrk>, <brk>, <arbttext>, and <hrule>, are similar to various elements
in %text;, but are permitted more universally. <date> and <title> are special
purpose but often- used elements that occur in numerous content models. -->

<!ELEMENT pgbrk - o EMPTY >

<!ATTLIST pgbrk      pgnumber CDATA              #IMPLIED
                    chgleve  NUMBER              #IMPLIED >

```

## MIL-HDBK-28001

## APPENDIX A

```

<!ELEMENT brk - o EMPTY >

<!ATTLIST brk type (col | line | epq | opg | npq) 'line' >

<!ELEMENT arbtext - - RCDATA >

<!ATTLIST arbtext arbtype NUMBER #IMPLIED >

<!ELEMENT hrule - o EMPTY >

<!ATTLIST hrule      thick      NUTOKEN      #REQUIRED
                    offset      NUTOKEN      #REQUIRED
                    length      NMTOKEN      #REQUIRED >

<!ELEMENT date - o (%text;) >

<!ATTLIST date %secur; >

<!ELEMENT (title | shorttitle) - o (%text;) -(table | chart | figure) >

<!ATTLIST (title | shorttitle) %secur; >

<!-- FLOATING ELEMENTS -->

<!-- Floating elements are only loosely attached to a particular point in the
text. They are printed/displayed somewhere nearby their "attachment point";
just where is prescribed by the FOSI. <figure>s, <table>s, and <chart>s have
their "attachment point" at the point where they occur in the text. The
location of the body of a <ftnote> is independent of its "attachment point";
each <ftnote> is identified by an ID value, and the "attachment point" is the
(first occurring) <ftnref> that references that ID. -->

<!ELEMENT figure - - (((%titles;)?, (subfig+ | ((graphic | macrograph) &
tgroup? & legend?))) | (verbatim, (%titles;)?))
-(figure|table|chart)>

<!ATTLIST figure tocentry %yesorno; "1"
                shortentry %yesorno; "0"
                orient (port | land) "port"
                %bodyatt;
                %secur; >

<!ELEMENT subfig - - ((graphic | macrograph) & tgroup? & legend?) >

<!ELEMENT macrograph - - (graphic+) >

<!ATTLIST macrograph reprowid NUTOKEN #IMPLIED
                    reprodep NUTOKEN #IMPLIED
                    graphstyleid NMTOKEN #IMPLIED >

<!ELEMENT legend - o (callout, def)+ >

<!ATTLIST legend assocfig IDREF #IMPLIED %secur; >

<!ELEMENT (table | chart) - - (((%titles;,tgroup+) | graphic+) -(table |
chart | figure) >

```

## MIL-HDBK-28001

## APPENDIX A

```

<!ATTLIST (table | chart)  tabstyle      NMTOKEN          #IMPLIED
                           tocentry      %yesorno;         "1"
                           shortentry    %yesorno;         #IMPLIED
                           frame         (top | bottom | topbot | all |
                           sides | none)  #IMPLIED
                           colsep        %yesorno;         #IMPLIED
                           rowsep        %yesorno;         #IMPLIED
                           orient        (port | land)      #IMPLIED
                           pgwide        %yesorno;         #IMPLIED
                           %bodyatt;
                           %secur; >

<!ELEMENT tgroup - o (colspec*, spanspec*, thead?, tfoot?, tbody) >

<!ATTLIST tgroup  cols          NUMBER          #REQUIRED
                  tgroupstyle  NMTOKEN          #IMPLIED
                  colsep        %yesorno;         #IMPLIED
                  rowsep        %yesorno;         #IMPLIED
                  align         (left | right | center | justify | char) "left"
                  charoff       NUTOKEN          "50"
                  char          CDATA            ""
                  %secur; >

<!ELEMENT colspec - o EMPTY >

<!ATTLIST colspec  colnum       NUMBER          #IMPLIED
                  colname      NMTOKEN          #IMPLIED
                  align        (left | right | center | justify | char) #IMPLIED
                  charoff      NUTOKEN          #IMPLIED
                  char          CDATA            #IMPLIED
                  colwidth     CDATA            #IMPLIED
                  colse        %yesorno;         #IMPLIED
                  rowsep        %yesorno;         #IMPLIED >

<!ELEMENT spanspec - o EMPTY >

<!ATTLIST spanspec  namest      NMTOKEN          #REQUIRED
                  nameend     NMTOKEN          #REQUIRED
                  spanname    NMTOKEN          #REQUIRED
                  align       (left | right | center | justify | char )
                  "center"
                  charoff     NUTOKEN          #IMPLIED
                  char        CDATA            #IMPLIED
                  colsep      %yesorno;         #IMPLIED
                  rowsep      %yesorno;         #IMPLIED >

<!ELEMENT (thead | tfoot) - o (colspec*, row+) -(entrytbl) >

<!ATTLIST thead      valign (top | middle | bottom) "bottom" %secur; >

<!ATTLIST tfoot      valign (top | middle | bottom) "top" %secur; >

<!ELEMENT tbody - o (row+) >

<!ATTLIST tbody      valign (top | middle | bottom) "top" %secur; >

<!ELEMENT row - o (entry | entrytbl)+ >

```

## MIL-HDBK-28001

## APPENDIX A

```

<!ATTLIST row      rowsep      %yesorno;                #IMPLIED
                  valign      (top | bottom | middle) #IMPLIED
                  %secur; >

<!ELEMENT entry - o ((para | warning | caution | note | legend) | %paracon;)+>

<!ATTLIST entry    colname      NMTOKEN                  #IMPLIED
                  namest       NMTOKEN                  #IMPLIED
                  nameend      NMTOKEN                  #IMPLIED
                  spanname     NMTOKEN                  #IMPLIED
                  morerows     NUMBER                    "0"
                  colsep       %yesorno;                #IMPLIED
                  rowsep       %yesorno;                #IMPLIED
                  rotate       %yesorno;                "0"
                  valign       (top | bottom | middle)   #IMPLIED
                  align        (left | right | center | justify |
                                char )                  #IMPLIED
                  charof       NUTOKEN                  #IMPLIED
                  char          CDATA                    #IMPLIED
                  %secur; >

<!ELEMENT entrytbl - - (colspec*, spanspec*, thead?, tbody)+ - (entrytbl) >
<!ATTLIST entrytbl cols         NUMBER                  #REQUIRED
                  namest       NMTOKEN                  #IMPLIED
                  nameend      NMTOKEN                  #IMPLIED
                  tgroupstyle  NMTOKEN                  #IMPLIED
                  colname      NMTOKEN                  #IMPLIED
                  spanname     NMTOKEN                  #IMPLIED
                  colsep       %yesorno;                #IMPLIED
                  rowsep       %yesorno;                #IMPLIED
                  align        (left | right | center | justify |
                                char )                  #IMPLIED
                  charoff      NUTOKEN                  #IMPLIED
                  char          CDATA                    #IMPLIED
                  %secur; >

<!ELEMENT ftnote - - (para+) -(ftnote | ftnref) +(table) >

<!ATTLIST ftnote   id          ID                      #REQUIRED
                  mark        (ctr | sym)              "ctr"
                  label       CDATA                    #IMPLIED
                  %secur; >

<!-- ELEMENT TYPES WHOSE USE IS NOT ILLUSTRATED IN THIS DECLARATION SET -->

<!ELEMENT contassurpg - o EMPTY >

<!ATTLIST contassurpg content ENTITY #REQUIRED >

<!ELEMENT reldoc - o (docno+) >

<!ELEMENT cfgpge - o EMPTY >

<!ATTLIST cfgpge name ENTITY #REQUIRED >

<!ELEMENT coverindex - o EMPTY >

```

## MIL-HDBK-28001

## APPENDIX A

```

<!ELEMENT staloc - o EMPTY >
<!ATTLIST staloc name ENTITY #REQUIRED >
<!ELEMENT testcode - o (%text;) >
<!ATTLIST testcode codetype (major | minor | sec) "major" %content; >
<!-- End of Example DTD -->

```

### 10.3 Alphabetical listing of tag descriptions.

**10.3.1 Text types.** The list of text types and their associated numbers have not yet been determined. The list will specify the legal values for the texttype attribute cited in the body attribute set. They may be listed here, as in previous publications, or listed externally.

**10.3.2 Element and attribute set descriptions.** Table I provides an alphabetically sorted, semantic description of each element type used in the example DTD.

Element types are to be used with their respective attribute definitions. Semantic descriptions of each element type are provided so that users may choose the most appropriate element type when creating a document type declaration for their application. Content models of element types may vary from document type declaration to document type declaration.

**10.3.3 Table components.** There are three columns in the table. Each is described below.

**10.3.3.1 Element type/attribute column.** The first column gives the name of the element type and a listing of each of its attributes (or attribute sets). No values are given for the attributes in this column. Attribute sets are not resolved.

**10.3.3.2 Full-name column.** The second column is the natural language name of the element type.

**10.3.3.3 Description column.** The third column provides several descriptions. First, there is a description of the element type itself. Then each attribute is listed, divided into groupings of required or optional. If the declared value of an attribute can be a list of tokens, they are provided; otherwise the keyword is given. The default value of the attribute is also given. If the default value is #IMPLIED, it signifies that the application may imply the value if it is not explicitly provided in the element type's usage. If a null value is to be assumed when a value is not specified for an attribute with an IMPLIED default, then (NULL) will follow the word IMPLIED. Otherwise the implication shall be explained.

**10.4 Attribute set descriptions.** There are also attribute set descriptions provided in this table, alphabetically with the element types. These attribute sets are not element types themselves, but rather are incorporated by reference with element types. These attribute sets are detailed in table II.

## MIL-HDBK-28001

## APPENDIX A

TABLE 1. <u>Element and attribute descriptions.</u>		
Element/Attribute	Full Name	Description
<b>&lt;abbrsect</b>  <b>%bodyatt;</b>  <b>%secur;&gt;</b>	Abbreviation Section	Identifies an abbreviation section.  Optional Attribute(s)  <b>%BODYATT::</b> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  <b>%SECUR::</b> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<b>&lt;address</b>  <b>%secur;&gt;</b>	Address	Identifies address information.  Optional Attribute(s)  <b>%SECUR::</b> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<b>&lt;appendix</b>  <b>tocentry =x</b>  <b>shortentry =x</b>  <b>%bodyatt;</b>  <b>%secur; &gt;</b>	Appendix	Identifies an appendix of the document.  Optional Attribute(s)  <b>TOCENTRY:</b> If the value consists only of zeros, the element's number and title are not used in the contents. Any other numeric value triggers the appendix title's use in the contents listing. Declared Value = %yesorno;(NUMBER) Default = "1"  <b>SHORTENTRY:</b> If the value consists only of zeros, the element's <shorttitle> (or <title>, if no short title is given) is not used in the <coverindex> or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno;(NUMBER) Default = "0"  <b>%BODYATT::</b> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  <b>%SECUR::</b> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.



## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%secur;>		<p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each element in the set.</p>
<applichd	Applicability Definition Heading	<p>Each applicability is known by a unique identifier which in turn refers to a coupling of a term and a definition. Various classes of terms may be used relevant to applicability. This heading classifies the type of applicability being defined.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each element in the set.</p>
id = x	Applicability Identifier	<p>Each applicability identifier, typically referring to a coupling of a term and a definition, is known by a unique identifier. It is with this element type that a symbol or term is associated with its definition. Examples of applicability identifiers might be particular aircraft tail numbers.</p> <p>Required Attribute(s)</p> <p>ID: The unique identifier of the applicability identifier. Declared Value = ID</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each element in the set.</p>
%secur;>		
<arbitrary	Arbitrary Text	<p>Identifies arbitrary information determined by the corresponding definition in the Output Specification. This text is typically used for manually putting in text to be used as headers or footers. It does not imply any particular processing by default. Rather this information is determined by the usages outlined in the FOSI.</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
arctype =x>		<p>Optional Attribute(s)</p> <p>ARCTYPE: Identifies different types of arbitrary text defined in the OS. A number that corresponds to a definition in the OS (i.e., header or footer information). Declared Value = NUMBER Default = IMPLIED (NULL)</p>
<authrty  %secur;>	Classification Authority	<p>Identifies the classification authority for a downgrade.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<body  %secur;>	Body Matter	<p>Identifies the body of the document.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<brk  type =x>	User Created Break	<p>User-created break to handle column, line, even/odd, and next page breaks. This element type is used to indicate that a break should be forced at its occurrence during processing of the information. There is no implication that there are any other breaks within the document, and, therefore, no implication that this type of break supports page integrity.</p> <p>Optional Attribute(s)</p> <p>TYPE: Type of break to be used. Declared Value = col (break to new column on same or next page), line (break line), epq (break to next even page), opq (break to next odd page), or npq (break to next page). Default = "line"</p>
<callout	Callout	Identifies a graphic callout identifier in text that correlates to a graphic callout in a figure used within the document.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
assocfig=x>		<p>Optional Attribute(s)</p> <p>ASSOCFIG: Reference to the unique identifier of the figure with which the callout is associated. Declared Value = IDREF Default = IMPLIED (NULL)</p>
<caution  type =x  xrefid =x  %\$secur:>	Caution	<p>Identifies a caution.</p> <p>Optional Attribute(s)</p> <p>TYPE: This specifies the type of caution. This type may be used as the title of the caution. Declared Value = CDATA Default = IMPLIED (NULL)</p> <p>XREFID: This specifies a cross reference to the unique identifier of a corresponding element. Declared Value = IDREF Default = IMPLIED (NULL)</p> <p>%\$SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<cfgpgc      name =x>	Configuration Page	<p>Defines an external configuration page. The location of the external entity is defined with the value of the content attribute. This will be an NDATA external entity in a content data notation defined within the appropriate document type declaration. The external entity is assumed to be a full- page graphic.</p> <p>Required Attribute(s)</p> <p>NAME: Value is the name of an external entity. Declared Value = ENTITY</p>
<change  level =x  change =x	Change Information	<p>Indicates the scope of changed information.</p> <p>Optional Attribute(s)</p> <p>LEVEL: Identifies level of change. Declared Value = NUMBER Default = IMPLIED (NULL)</p> <p>CHANGE: Type of change Declared Value = add or delete. Default = IMPLIED (NULL)</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
mark =x  %secur;>		<p>MARK: If the value consists only of zeros, no sidemark is used; if the value is any other numeral, a sidemark is used. If left to default, a sidemark is used only if the value of the level attribute is equal to the value of the change level of the document. Declared Value = %yesorno;(NUMBER) Default = IMPLIED (NULL)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<chapter  tocentry =x>  shortentry =x  %bodyatt;  %secur;>	Chapter	<p>Identifies a chapter of the document.</p> <p>Optional Attribute(s)</p> <p>TOCENTRY: If the value consists only of zeros, the element's number and title are not used in the contents. Any other numeric value triggers the Chapter title's use in the contents listing. Declared Value = %yesorno;(NUMBER) Default = "1"</p> <p>SHORTENTRY: If the value consists only of zeros, the element's &lt;shorttitle&gt; (or &lt;title&gt;, if no short title is given) is not used in the &lt;coverindex&gt; or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno;(NUMBER) Default = "0"</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<chart  tabstyle =x	Chart	<p>Identifies a chart, typically tabular information that contributes to the List of Illustrations.</p> <p>Optional Attribute(s)</p> <p>TABSTYLE: A unique chart style defined in the FOSI. Declared Value = NMTOKEN Default = IMPLIED (NULL)</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
tocentry =x		TOCENTRY: If other than zeros, and the title is present, this chart title should be included in the list of illustrations. (Ignore if the optional title is omitted). Declared Value = %yesorno;(NUMBER) Default = "1"
shortentry =x		SHORTENTRY: If the value consists only of zeros, the element's <shorttitle> (or <title>, if no short title is given) is not used in the <coverindex> or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno;(NUMBER) Default = IMPLIED (NULL)
frame =x		FRAME: Describes position of outer rulings. Declared Value = sides (left and right), top (below title), bottom (after last row possibly of tfoot material), topbot (both top and bottom), all (all of above), or none (none of above). Default = IMPLIED (NULL implies value from tabstyle in FOSI, if available, NULL if not)
colsep =x		COLSEP: Default for all items in this chart. If other than zeros, display the internal column rulings to the right of each item; if only zeros, do not display it. Ignored for the last column, where the frame sides setting applies. Declared Values = %yesorno;(NUMBER) Default = IMPLIED (NULL implies value from tabstyle in FOSI, if available, NULL if not)
rowsep =x		ROWSEP: Default for all items in this chart. If other than zeros, display the internal vertical row ruling below each item. If only zeros, do not display it. Ignored for the last row of the chart. Declared Value = %yesorno;(NUMBER) Default = IMPLIED (NULL implies value from tabstyle in FOSI, if available, NULL if not)
orient =x		ORIENT: Orientation of the entire chart. Declared Value = port (table writing direction, along rows, is the same as marginal text), or land (table writing direction is 90 degrees counterclockwise to marginal text). Default = IMPLIED (NULL implies value from tabstyle in FOSI, if available, NULL if not)

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
<p>pgwide =x</p> <p>%bodyatt;</p> <p>%secur;&gt;</p>		<p>PGWIDE: If other than zeros, the chart runs across the page. If only zeros, the chart runs across just the (galley) width of the current column of the page (regardless of orient). If the value is only zeros, it has no meaning for orient=land.</p> <p>Declared Value = %yesorno;(NUMBER)</p> <p>Default = IMPLIED (NULL implies value from tabstyle in FOSI, if available, NULL if not)</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p>
<p>&lt;chgdate</p> <p>%secur;&gt;</p>	Change Date	<p>Identifies the document's publication change date.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p>
<p>&lt;chglist</p> <p>%secur;&gt;</p>	Change List	<p>The change list is associated with the change sheet. It lists which pages are to be removed and which are to be inserted relative to the current change.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p>
<p>&lt;chgnum</p> <p>%secur;&gt;</p>	Change Number	<p>Identifies the current change level of the document.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p>
<p>&lt;chgrec</p> <p>%bodyatt;</p>	Change Record	<p>Identifies the change record information.</p> <p>Optional Attribute(s)</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%\$ secur;>		%\$SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<chgsheet	Change Sheet	Identifies a change sheet in the document. This sheet may be made up of elements explicitly placed in the document or it may be generated by the system. The purpose of the change sheet is to list the reason for the change to the data and also to provide a table designating which pages are to be removed and which are to be inserted.  Optional Attribute(s)  %\$SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
%\$ secur;>		%\$SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<col	Matrix Column	Identifies a column in a matrix.  Optional Attribute(s)  ALIGN: Horizontal alignment of the characters within a matrix as defined in the "f.align" entity. Declared Value = center, left, or right. Default = "center"
align =x>		
<colspec	Column Specification	Specifies a column, a vertical portion of a <table>, <chart>, or <entrytbl>. The default values come from the <tgroup>, <thead>, or <tfoot> starting the current (enclosing) group. Each <colspec> is for a single column, so it properly has a column number, colnum, implicitly in order starting from 1, and an optional colname by which it is known when used in any <spanspec> or in <entry>. A <colspec> set on <thead> or <tfoot> should be complete for all columns. It overrides those on the containing <tgroup> and applies to just the <thead> or <tfoot>. If there is no <colspec> used within <thead> or <tfoot>, then the <colspec> of the containing <tgroup> (or the prior <tgroup>) is used. <Colspec>s from the containing <tgroup> apply to <tbody>.  Optional Attribute(s)  COLNUM: Number of column, counting from 1 at left of the chart or table. Declared Value = NUMBER Default = IMPLIED (NULL)  COLNAME: Name of column, used to specify the position in a row, or the start or end of a horizontal span of columns (<spanspec>). Declared Value = NMTOKEN Default = IMPLIED (NULL)
colnum =x		
colname =x		

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
align =x		ALIGN: Text horizontal position within the column. Declared Value = left (quad flush left), center (centered), right (quad flush right), justify (both quad left and quad right), or char (align on leftmost of char, positioned by charoff). Default = IMPLIED Note: If no value given, obtain from FOSI.
charoff =x		CHAROFF: For align="char", horizontal character offset is the percent of the current column width to the left of the (left edge of the) alignment character. Declared Value = NUTOKEN Default = IMPLIED, from enclosing table group
char =x		CHAR: If align = "char", the value is the single alignment character on which the first to occur of this character in the entry is aligned. Entries not containing this character are aligned to the left of this position. Declared Value = CDATA Default = IMPLIED, from enclosing table group
colwidth =x		COLWIDTH: Either proportional measure of the form number*, i.e., "5*" for 5 times the proportion, or "*" (= "1*"); fixed measure, i.e., 2pt for 2 point, 3pi for 3 pica; or mixed measure, i.e., 2*+3pt. Coefficients are positive numbers with up to 2 decimal places. Declared Value = CDATA Default = IMPLIED Note: If no value is given, then obtain value from the FOSI or, if there is no FOSI value, then assume a proportion of 1.
colsep =x		COLSEP: Default for all items in this column (within the closing group) of the table or chart. If other than zeros, display the internal column ruling to the right of each item; if only zeros, do not display it. Ignored for the last column, where the frame setting applies. Declared Value = %yesorno;(NUMBER) Default = IMPLIED, from enclosing table group.
rowsep =x>		ROWSEP: Default for all items in this column (within the enclosing group) of the table or chart. If other than zeros, display the internal horizontal row ruling below each item. If only zeros, do not display it. Ignored for the last row of the table, since overridden by the frame setting. Declared Value = %yesorno;(NUMBER) Default = IMPLIED, from enclosing table group.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.		
Element/Attribute	Full Name	Description
<contassurpg  content =x>	Content Assurance Page	Defines an external content assurance page. The location of the external entity is defined with the value of the content attribute.  Required Attribute(s)  CONTENT: The name of the entity that defines the content assurance page should be the value of this attribute. Declared Value = ENTITY
<contents  shortentry =x>	Generated Table of Contents	Identifies element that refers to a table of contents generated by the receiving system.  Optional Attribute(s)  SHORTEXTY: If the value consists only of zeros, the element's <shorttitle> (or <title>, if no short title is given) is not used in the <coverindex> or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno;(NUMBER) Default = "0"
<contractno  %secur>	Contract Number	Identifies a contract number.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<coverindex>	Generated Cover Index	Identifies the element that refers to a front-cover index generated by the receiving system. This index is typically in a table of contents order and uses the <shorttitle> content, if present (<title>, if not). It usually appears on the cover of documents conforming to MIL-M-63036 and MIL-M-63038. Not all instances of element types that support the "shortentry" attribute need be extracted for any one publication.
<dataiden  %bodyatt:	Data Identification Number	Identifies information that has a different "texttype" from that of the surrounding text within a sentence, paragraph, step, note, table entry, etc.  Optional Attribute(s)  %BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<date	Date	Identifies a date of the form yyyy/mm/dd. However, it may be formatted for presentation as appropriate to the application.
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<dddesc	Difference Data Description	Identifies a difference data description.
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<ddindex	Difference Data Index	Identifies the difference data index.
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<ddintro	Difference Data Introduction	Identifies the difference data introduction.
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<ddlist	Difference Data List	Identifies a difference data list.
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<ddsheets	Difference Data Sheet	Identifies a difference data sheet.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<ddunit portion = x %secur;>	Difference Data Unit	Identifies a difference data unit. Optional Attribute(s) PORTION: Specifies to what major document structure the difference data pertains. Declared Value = section or chapter. Default = IMPLIED (NULL implies "chapter" if <section> is not used in instance) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<def %secur;>	Definition	Identifies a definition. Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<deflist %secur;>	Definition List	Identifies a definition list. Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<df id = x align = x	Display Formula	Identifies a display formula. Optional Attribute(s) ID: Unique identifier of the display formula. Declared Value = ID Default = IMPLIED (NULL) ALIGN: Horizontal alignment of the display formula as declared in the "f.align" entity. Declared Value = center, left, or right. Default = "left"

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
num =x>		NUM: Specifies an explicit formula number; if omitted, sequential numbering of the formulae would normally be performed by the text formatter. Declared Value = CDATA Default = IMPLIED (NULL)
<dfg  id =x  align =x  num =x>	Display Formula Group	Identifies a formula group whose content is display formulae.  Optional Attribute(s)  ID: Unique identifier of the display formula. Declared Value = ID Default = IMPLIED (NULL)  ALIGN: Horizontal alignment of the display formula as declared in the "f.align" entity. Declared Value = center, left, or right. Default = "left"  NUM: Specifies an explicit formula group number; if omitted, sequential numbering of the formulae would normally be performed by the text formatter. Declared Value = CDATA Default = IMPLIED (NULL)
<dref  refid =x>  page =x	Formula Reference	Identifies a reference to a formula or formula group.  Required Attribute(s)  REFID: Reference to the unique identifier of a <df>. Declared Value = IDREF  Optional Attribute(s)  PAGE: Page number is added (default) to the reference of the unique identifier of a <df>. Declared Value = yes or no. Default = "yes"
<doc  service =x	Document Level Element	Identifies the start of the data of a technical document.  Required Attribute(s)  SERVICE: Specifies the military service of the procuring activity (service primarily responsible for the document). Declared Value = af (Air Force), navy (Navy), army (Army), mc (Marine Corps), dla (Defense Logistics Agency), cg (Coast Guard), or %service; (contractor defined).

## MIL-HDBK-28001

## APPENDIX A

TABLE 1. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
docid =x		Optional Attribute(s)  DOCID: Unique identifier of the document, which can be used to perform interdocument cross references. However, it should be noted that this is a particular of the application and is not an SGML construct that is validated by the parser. Declared Value = CDATA Default = IMPLIED (NULL)
docstat =x		DOCSTAT: Specifies the current status of the document publication. Declared Value = revision (newly revised), change (newly changed), draft (draft), prelim (preliminary title), or formal (formal version). Default = "formal"
mantype =x>		MANTYPE: Designates the manual type of the document. Declared Value = standard (standard manual), card, or decal. NOTE: a card type manual is normally a set of 5" x 8" index cards used within the Air Force for purposes such as periodic inspections (pre/post flight inspections). "CARD" and "decal" are for use with documents conforming to MIL-M-63004. Default = "standard"
<docmfr	Document Manufacturer	Identifies the manufacturer of the document.
%secur:>		Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<docno	Document Number	Identifies a document number.
%secur:>		Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<docpart	Document Part	Identifies a part in technical manuals, all of which have the same publication number.
%bodyatt;		Optional Attribute(s)  %BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.

## MIL-HDBK-28001

## APPENDIX A

TABLE 1. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<docpartn  %secur;>	Document Part Number	Identifies the document part number.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<doctype  %secur;>	Document Type	Identifies the type of publication.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<downgrd  %secur;>	Downgrade Notice	Identifies the downgrade notice.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<emergency>	Emergency Information	Indicates the scope of emergency information.
<emphasis  emph =x>	Emphasis	Indicates the scope of emphasized information. The emphasis types available will be listed in the Output Specification and associated with a named value.  Required Attribute(s)  EMPH: Names identifying the types of emphasis. More than one name can be used, meaning that each of the names will produce a form of emphasis. Emphasis types are identified in the Output Specification and FOSI for the document. Declared Value = NAMES
<entry	Entry	Identifies an entry in a table or chart. Default values come from the table or chart, tgroup, colspec, spanspec, thead, tfoot, tbody, or row attlist values for like-named attributes. An entry not specified by a spanspec gets its defaults from its starting column.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
colname =x		Optional Attribute(s) COLNAME: Column name of entry. Omit if spanname is present. Declared Value = NMTOKEN Default = IMPLIED (NULL, implies the next column after the end of the prior entry or entrytbl, else the first column of the row).
namest =x		NAMEST: Name of leftmost column of span. Names are identified in colspec of the current tgroup. Declared value = NMTOKEN
nameend=x		NAMEEND: Name of rightmost column of span. Names are identified in colspec of the current tgroup. Declared value = NMTOKEN
spanname =x		SPANNAME: Name of a horizontal span. Declared Value = NMTOKEN Default = IMPLIED (NULL)
morerows =x		MOREROWS: Number of additional rows in a vertical straddle. Declared Value = NUMBER Default = "0"
colsep =x		COLSEP: If other than zeros, display the internal vertical column ruling at the right of the entry; if only zeros, do not display it. Ignored for the last column, where the frame setting applies. Declared Value = %yesorno;(NUMBER) Default = IMPLIED, from colspec or spanspec.
rowsep =x		ROWSEP: If other than zeros, display the internal horizontal row rulings below the entry; if only zeros, do not display it. Ignored for the last row where the frame setting applies. Declared Value = %yesorno;(NUMBER) Default = IMPLIED, from the row.
rotate =x		ROTATE: Rotations are not additive to those specified in FOSI. Content is either in the orientation of the table (value is one or more zeros) or 90 degrees counterclockwise to table orientation (value is other than zeros). Declared Value = %yesorno;(NUMBER) Default = "0"

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
valign =x		VALIGN: Text vertical positioning within the entries. Declared Value = top, middle (vertically centered), or bottom. Default = IMPLIED
align =x		ALIGN: Text horizontal position within the column. Declared Value = left (quad flush left), center (centered), right (quad flush right), justify (both quad left and right), or char (align on leftmost of char position by charoff). Default = IMPLIED, from colspec or spanspec.
char =x		CHAR: If align = "char", the value is the single alignment character on which the first to occur of this character in the entry is aligned. If that character does not occur in the entry, the entry aligns to the right of that character offset. Declared Value = CDATA Default = IMPLIED (NULL implies there is no aligning character).
charoff =x		CHAROFF: For align = "char", percent of the current width to the left of the (left edge of) the character. Declared Value = NUTOKEN Default = IMPLIED, from colspec or spanspec.
%secur>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<entrytbl	Entry Table	An entrytbl takes the place of an entry, but fits into a single row. It permits subdivisions of an entry or horizontal span of entries into possibly different columns and (sub)rows. Note that several entrytbls may occur in the same body row, and these could have a different number of (sub)rows. There is no implication of alignment of (sub)rows in different entrytbls. Default values come from the table or chart, tgroup, colspec, spanspec, thead, tfoot, tbody, or row attlist values for like-named attributes, in the same manner as an entry.  Required Attribute(s)  COLS: Number of columns in the table or chart. Declared Value = NUMBER  Optional Attribute(s)  TGROUPESTYLE: A unique table group style defined in the FOSI. Declared Value = NMTOKEN Default = IMPLIED (NULL)
cols =x		
tgroupstyle =x		

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
colname =x		COLNAME: Leftmost column of entrytbl in the row of the table or chart. Declared Value = NMTOKEN Default = IMPLIED (NULL implies next column).
spanname =x		SPANNAME: Name of a horizontal span. Declared Value = NMTOKEN Default = IMPLIED (NULL)
namest =x		NAMEST: Name of leftmost column of span. Names are identified in colspec of the current tgroup. Declared value = NMTOKEN
nameend=x		NAMEEND: Name of rightmost column of span. Names are identified in colspec of the current tgroup. Declared value = NMTOKEN
colsep=x		COLSEP: If other than zeros, display the internal vertical column ruling at the right of the entrytbl, (but not for the last table column, where the siderule setting applies). If only zeros, do not display. Declared Value = %yesorno;(NUMBER) Default = IMPLIED, from enclosing table group.
rowsep =x		ROWSEP: If other than zeros, display by default the horizontal rulings below entrytbl (but not for entries in the last row of the table or chart). If only zeros, do not display ruling. Declared Value = %yesorno;(NUMBER) Default = IMPLIED, from enclosing table group.
align=x		ALIGN: Text horizontal position within the column. Declared Value = left (quad flush left), center (centered), right (quad flush right), justify (both quad left and right), or char (align on leftmost of char position by charoff). Default = IMPLIED, from colspec or spanspec.
charoff =x		CHAR: If align = "char", the value is the single alignment character on which the first to occur of this character in the entry is aligned. If that character does not occur in the entry, the entry aligns to the right of that character offset. Declared Value = CDATA Default = IMPLIED (NULL implies there is no aligning character).
char=x		CHAROFF: For align = "char", percent of the current width to the left of the (left edge of) the character. Declared Value = NUTOKEN

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<eqpttype	Equipment Type	Identifies the equipment type associated with the document.  Optional Attribute(s)
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<errpt	Error Report	System-generated form for reporting errors in the document.  Required Attribute(s)
erptype =x		ERPTYPE: Type of error report form to be used. Declared Value = tpdrr (Navy technical manuals deficiency report), afo22 (Air Force report), or da2028 (Army report). Optionally, other report types may be defined in the replacement text of %erptype;.
%secur;>		Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<extref	Cross Reference	This is a cross reference to another document.  Optional Attribute(s)
docno =x		DOCNO: Value is the identifier of an external document. Declared Value = CDATA Default = IMPLIED (NULL)
pretext =x		PRETEXT: Text that will precede the cross reference when resolved for display. Declared Value = CDATA Default = IMPLIED (NULL)
posttext =x		POSTTEXT: Text that will follow the cross reference when resolved for display. Declared Value = CDATA Default = IMPLIED (NULL)

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.		
Element/Attribute	Full Name	Description
% secur;		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<f>	Inline Formula	Identifies an inline formula.
<faultcode  % content;  % secur;	Fault Code	Identifies a fault code.  Optional Attribute(s)  %CONTENT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<figindex  % secur;	Figure Index Number	Identifies a figure index number.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<figure  tocentry = x  shortentry = x  orient = x	Figure	Identifies a figure in the document.  Optional Attribute(s)  TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illustlist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "1"  SHORTENTRY: If the value consists only of zeros, the element's <shorttitle> (or <title>, if no short title is given) is not used in the <coverindex> or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"  ORIENT: Specifies the orientation of the figure on the page, together with its title. (Note: Rotations are additive.) Declared Value = port or land. Default = "port"

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%bodyatt;  %secur;>		<p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<foldout  pgstyle =x  %secur;>	Foldout	<p>Identifies a foldout within the foldout section. It can consist of a table, figure, or chart.</p> <p>Optional Attribute(s)</p> <p>PGSTYLE: Refers to page style defined in the corresponding FOSI. Declared Value = NUMBER Default = IMPLIED (NULL)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<foldsect  %secur;>	Foldout Section	<p>Identifies the foldout section of the document.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<foreword  tocentry =x  %bodyatt;	Foreword	<p>Identifies the foreword material of the document. The foreword, when included in a volume or part of a manual, shall contain the purpose and scope of the manual plus any other information required by the technical content specification. It may define new abbreviations and symbols.</p> <p>Optional Attribute(s)</p> <p>TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "1"</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<front  %secur;>	Front Matter	Identifies the front matter.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<fnote  id =x  mark =x  label =x  %secur;>	Footnote	Identifies the body of a footnote in the document.  Required Attribute(s)  ID: Unique identifier of the footnote. Declared Value = ID  Optional Attribute(s)  MARK: If symbol is chosen, they will be assigned in the order specified in the GPO Manual of Style. Declared Value = ctr (counter) or sym (symbol). Default = "ctr"  LABEL: If used, it specifies the number or symbol of the fnote and overrides autogeneration of the number or symbol by the processing system. Declared Value = CDATA Default = IMPLIED (NULL)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<fnref  xrefid =x>	Footnote Reference	Indicates a footnote reference.  Required Attribute(s)  XREFID: Unique identifier of the footnote being referenced. Declared Value = IDREF
<fnsec	Footnote Section	Identifies a section that is used to list footnotes.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
tocentry =x		Optional Attribute(s) TOCENTRY: If the value consists only of zeros, the element's number and title are not used in table of contents. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "1"
shortentry =x		SHORTENTRY: If the value consists only of zeros, the element's <shorttitle> (or <title>, if no short title is given) is not used in the <coverindex> or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"
%bodyatt;		%BODYATT: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
%secur;>		%SECUR: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<glossary	Glossary	Identifies the glossary information of a document. Optional Attribute(s)
tocentry =x		TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "1"
shortentry =x		SHORTENTRY: If the value consists only of zeros, the element's <shorttitle> (or <title>, if no short title is given) is not used in the <coverindex> or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"
%secur;>		%SECUR: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<graphic	Graphic	Identifies a graphic. A graphic is stored either as vector (MIL-PRF-28000 or MIL-PRF-28003) or raster (MIL-PRF-28002) data and is used as an illustration in the document.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
boardno =x		<p>Required Attribute(s)</p> <p>BOARDNO: Enter unique graphic identifier. Declared Value = ENTITY</p> <p>Optional Attribute(s)</p>
graphsty =x		<p>GRAPHSTY: Characteristic provided to allow for cases where the "grphstyl" ID is to be used. Declared Value = NMTOKEN Default = IMPLIED (NULL implies only one style available).</p>
llcordra =x		<p>LLCORDRA: Left lower coordinate pair of the portion of the graphic to be placed in the portion of the repro area, separated by comma. Declared Value = CDATA Default = IMPLIED (NULL)</p>
urcordra =x		<p>URCORDRA: Right upper coordinate pair of the portion of the graphic to be placed in the portion of the repro area, separated by comma. Declared Value = CDATA Default = IMPLIED (NULL)</p>
reprowid =x		<p>REPROWID: Repro area width. Declared Value = NUTOKEN Default = IMPLIED (NULL implies value from &lt;macrograph&gt;, if available, NULL if not)</p>
reprodep =x		<p>REPRODEP: Repro area depth. Declared Value = NUTOKEN Default = IMPLIED (NULL implies value from &lt;macrograph&gt;, if available, NULL if not)</p>
hscale =x		<p>HSCALE: Horizontal scaling. Declared Value = NUTOKEN Default = IMPLIED (NULL)</p>
vscale =x		<p>VSCALE: Vertical scaling. Declared Value = NUTOKEN Default = IMPLIED (NULL)</p>
scalefit = x		<p>SCALEFIT: Characteristic allows the graphic to be scaled as needed to fit the size of the reproduction area. Declared Value = %yesorno; (NUMBER) Default = IMPLIED (NULL)</p>
hplace =x		<p>HPLACE: Horizontal placement in the available repro area. Declared Value = left, right, center, or none (equivalent to a null value which defaults to implied by the graphstyle). Default = IMPLIED (NULL)</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
vplace =x		VPLACE: Vertical placement in the available repro area. Declared Value = top, middle, bottom, or non (equivalent to a null value which defaults to implied by the graphstyle). Default = IMPLIED (NULL)
coordst =x		COORDST: Left lower coordinate pair, separated by comma. Start position in repro area for placement of the portion of the graphic specified by llicordra and urcordra. Declared Value = CDATA Default = IMPLIED (NULL)
coordend =x		COORDEND: Right upper coordinate pair, separated by comma, end position in repro area for placement of portion of the graphic. Declared Value = CDATA Default = IMPLIED (NULL)
rotation =x		ROTATION: Degree of rotation of the graphic. Declared Value = NUMBER Default = IMPLIED (NULL)
%bodyatt;		%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<howtouse	How To Use	Identifies "how to use" information.  Optional Attribute(s)
tocentry =x		TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "0"
shortentry =x		SHORTENTRY: If the value consists only of zeros, the element's <shorttitle> (or <title>, if no short title is given) is not used in the <coverindex> or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"
%bodyatt;		%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<hrule  thick =x  offset =x  length =x>	Horizontal Rule	Ability to place a horizontal rule (line).  Required Attribute(s)  THICK: Rule Thickness Declared value = NUTOKEN  OFFSET: Offset from the baseline. Declared value = NUTOKEN  LENGTH: Rule Length Declared value = NMTOKEN
<idinfo  %secur;>	Identification Information	Identifies information which will be used to produce the title page and cover.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<illuslist  tocentry =x    shortentry =x>	Generated List of Illustrations	Identifies element that refers to a list of illustrations generated by the receiving system.  Optional Attribute(s)  TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "1"  SHORTENTRY: If the value consists only of zeros, the element's <shorttitle> (or <title>, if no short title is given) is not used in the <coverindex> or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"
<index	Index	Identifies element that refers to an index generated by the receiving system.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
shortentry=x>		<p>Optional Attribute(s)</p> <p>SHORTENTRY: If the value consists only of zeros, the Index is not listed in the &lt;coverindex&gt; or any other type of compiled listing. Any other numeric value triggers its listing. Declared Value = %yesorno; (NUMBER) Default = "0"</p>
<indxflag	Index Entry Flag	<p>Identifies text of an item to be extracted for the index. The four attributes, ref1-ref4, refer to the level of the index entry. If the item is a ref2 entry, then the ref1 attribute must be used. If the item is a ref3 entry, then both the ref1 and ref2 attributes must be used and so forth.</p> <p>Optional Attribute(s)</p> <p>ref1 =x REF1: First level reference on an index flag. Declared Value = CDATA Default = IMPLIED (NULL)</p> <p>ref2 =x REF2: Second level reference on an index flag. Declared Value = CDATA Default = IMPLIED (NULL)</p> <p>ref3 =x REF3: Third level reference on an index flag. Declared Value = CDATA Default = IMPLIED (NULL)</p> <p>ref4 =x REF4: Fourth level reference on an index flag. Declared Value = CDATA Default = IMPLIED (NULL)</p> <p>%secur;&gt; %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<insertpg	Insert Page	<p>Identifies the page number of a page to be inserted relative to the current change level. It is usually associated with a change list and change sheet.</p> <p>Optional Attribute(s)</p> <p>%secur;&gt; %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<intlstd	International Standard Information	Identifies information regarding compliance with international standards.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
<code>&amp;secur;&gt;</code>		Optional Attribute(s) <code>&amp;SECUR::</code> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<code>&lt;intro</code>  <code>tocentry =x</code>  <code>&amp;bodyatt;</code>  <code>&amp;secur;&gt;</code>	Introduction	Identifies the introductory material of the document.  Optional Attribute(s)  <code>TOCENTRY:</code> If the value consists only of zeros, the element's number and title are not used in contents, <code>illuslist</code> , or <code>tablelist</code> . Any other numeric value triggers their use in the appropriate listing. Declared Value = <code>&amp;yesorno;</code> (NUMBER) Default = "1"  <code>&amp;BODYATT::</code> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  <code>&amp;SECUR::</code> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<code>&lt;item</code>  <code>id =x</code>  <code>label =x</code>  <code>&amp;secur;&gt;</code>	Item	Identifies an item typically occurring within a type of list.  Optional Attribute(s)  <code>ID:</code> Unique identifier of the item. Declared Value = <code>ID</code> Default = IMPLIED (NULL)  <code>LABEL:</code> If used, it specifies the number or symbol of the item and it will override the numbering that would be generated based on the type of the list. Declared Value = <code>CDATA</code> Default = IMPLIED (NULL)  <code>&amp;SECUR::</code> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<code>&lt;legend</code>  <code>assocfig =x</code>	Legend	Identifies a legend.  Optional Attribute(s)  <code>ASSOCFIG:</code> Reference to the unique identifier of the figure with which the legend is associated. Declared Value = <code>IDREF</code> Default = IMPLIED (NULL)

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<lep>	Generated List of Effective Pages	Identifies element that refers to a list of effective pages generated by the receiving system.
<lin	Line Item Number	Identifies a line item number.
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<macrograph>	Macrographic	Identifies a group of graphics that are used together to form an illustration. All graphics in each macrograph share the same repro area. If repropwid and/or reprodep are specified on a graphic within the context of a macrograph, they are ignored.
reprowid=x		Optional Attribute(s) REPROWID: Width of the repro area. Declared Value = NUTOKEN Default = IMPLIED (NULL)
reprodep=x>		REPRODEP: Depth of the repro area. Declared Value = NUTOKEN Default = IMPLIED (NULL)
<maintlvl	Maintenance Level	Identifies the maintenance level of the manual.
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<material	Material	Identifies a material.
%content;		Optional Attribute(s) %CONTENT; Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<mfr	Manufacturer	Identifies the equipment manufacturer.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<modelno %secur;>	Equipment Model Number	Identifies an equipment model number.  Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<nomen %secur;>	Equipment Nomenclature	Identifies the equipment nomenclature of the document.  Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<note  type =x  xrefid =x  %secur;>	Note	Identifies a note.  Optional Attribute(s) TYPE: This specifies the type of note. This type may be used as the title of the note. Declared Value = CDATA Default = IMPLIED (NULL) XREFID: This specifies a cross reference to the unique identifier of a corresponding element. Declared Value = IDREF Default = IMPLIED (NULL) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<notice	Notice	Identifies a notice.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
<p>notctype =x</p> <p>%secur;&gt;</p>		<p><b>Optional Attribute(s)</b></p> <p>NOTCTYPE: Specifies the type of notice to be inserted on the page. Declared Value = dist (distribution), auth (authority), fouo (For Official Use Only), branch (service branch), pgclass (This page is unclassified notice), disclos (disclosure), supersed (supersedure), effdate (effective date), suppl (supplement notice), nopg (number of notice pages in a secured document), noclaspg (number of classified pages in a secured document), warning (warning notice), destr (destruction), safesup (safety supplement), opersup (operational supplement), or maintsup (maintenance supplement). The parameter entity %notctype may also be defined for the document to extend the possible choices. The text of notices is either explicitly present or implicitly present through the use of entity references. Default = IMPLIED (NULL)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<p>&lt;nsn</p> <p>%secur;&gt;</p>	National Stock Number	<p>Identifies a national stock number.</p> <p><b>Optional Attribute(s)</b></p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<p>&lt;oadr</p> <p>%secur;&gt;</p>	Official Authority Downgrade Review	<p>Identifies text instructing confirmation of downgrade notice with official authority.</p> <p><b>Optional Attribute(s)</b></p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<p>&lt;organiz</p> <p>%secur;&gt;</p>	Organization	<p>Identifies an agency, organization, or company.</p> <p><b>Optional Attribute(s)</b></p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<para	Paragraph Text	Identifies text within a paragraph.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%bodyatt;  %secur;>		Optional Attribute(s)  %BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<para0  tocentry =x  shortentry =x  %bodyatt;  %secur;>	Primary Paragraph	Identifies a primary numbered paragraph in the document's structure.  Optional Attribute(s)  TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illustlist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "1"  SHORTENTRY: If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"  %BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<partdesc  %secur;>	Part Description	Identifies an equipment part description.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<partname	Part Name	Identifies an equipment part name.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<partno %secur;>	Equipment Part Number	Identifies an equipment part number.  Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<pgbrk  pgnumber =x  chglevel =x>	Page Break	User- or system-inserted page break to preserve page integrity along with the capability to specify the change level of the delimited page. This element type is used to designate the limits of a page. Further, through attributes, it provides the ability to designate the folio and current change level of the delimited information. If this element type is used, it is assumed it will be used consistently throughout the document to designate all page breaks.  Optional Attribute(s) PGNUMBER: The page number of the information preceding the <pgbrk>. Declared Value = CDATA Default = IMPLIED, the next page in the document. CHGLEVEL: The change level of the information preceding the <pgbrk>. Declared Value = NUMBER Default = IMPLIED (NULL)
<pgno %secur;>	Page Number	Identifies a page number.  Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<phrase %secur;>	Downgrade Phrase	Identifies the downgrade notice text.  Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
<p>&lt;position</p> <p>%secur;&gt;</p>	Position	<p>Identifies position or rank.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<p>&lt;precaut</p> <p>%secur;&gt;</p>	Precaution	<p>Identifies a precaution.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<p>&lt;preface</p> <p>tocentry =x</p> <p>%bodyatt;</p> <p>%secur;&gt;</p>	Preface	<p>Identifies the preface material of the document. The preface, when included in a volume or part of a manual, shall contain the purpose and scope of the manual plus any other information required by the technical content specification. It may define new abbreviations and symbols.</p> <p>Optional Attribute(s)</p> <p>TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illust, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesomo; (NUMBER) Default = "1"</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<p>&lt;prepubno</p> <p>%secur;&gt;</p>	Previous Publication Number	<p>Identifies the previous publication number of the document.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<promul	Promulgation Information	Identifies the promulgation information.

## MIL-HDBK-28001

## APPENDIX A

TABLE 1. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
%bodyatt;  %secur;>		Optional Attribute(s)  %BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<prtitle  %secur;>	Prime Title	Identifies the prime title block of the document. A graphic may be used before, after, or within the nomenclature, type, or the subject.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<pslist  %secur;>	Part Number / Serial Number List	Identifies a list of equipment part and serial numbers.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<pubdate  %secur;>	Publication Date	Identifies the base publication date.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<pubno  %secur;>	Publication Number	Identifies the publication number of the document.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<purpose	Signature Purpose	Identifies the purpose of the signature (i.e., Validated By, Verified By, etc.)

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.		
Element/Attribute	Full Name	Description
%secur;>		Optional Attribute(s) %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<randlist  prefix =x  %secur;>	Random List	Identifies a random list.  Optional Attribute(s)  PREFIX: Identifies the prefix desired on items in the list such as a symbol (e.g., bullet). Declared Value = CDATA Default = IMPLIED (NULL)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<rear  %secur;>	Rear Matter	Identifies the rear matter.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<refdes  %secur;>	Reference Designation Index Number	Identifies the reference designation index number of an equipment part.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<refdoc>	Referenced Documents	Identifies the documents referenced by the manual.
<removepg  %secur;>	Remove Page	Identifies the page number of the page to be removed relative to the current change level. It is usually associated with a change list and change sheet.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<revnum	Revision Number	Identifies revision number of the document.



## MIL-HDBK-28001

## APPENDIX A

TABLE 1. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
%service;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<seal>	Seal	Identifies the seal, if any, associated with the document.
<section  tocentry =x  shortentry =x  %bodyatt;  %secur;>	Section	Identifies a section of a document.  Optional Attribute(s)  TOCENTRY: If the value consists only of zeros, the element's number and title are not used in table of contents. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesomo; (NUMBER) Default = "1"  SHORTENTRY: If the value consists only of zeros, the element's <shorttitle> (or <title>, if no short title is given) is not used in the <coverindex> or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesomo; (NUMBER) Default = "0"  %BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<seqlist  prefix =x	Sequential List	Identifies a sequential list. It is composed of an optional title and one or more items.  Optional Attribute(s)  PREFIX: The user may specify the prefix desired on the items in the list, such as "STEP." Declared Value = CDATA Default = IMPLIED (NULL)

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.		
Element/Attribute	Full Name	Description
numstyle =x  %bodyatt;  %secur;>		<p>NUMSTYLE: Enumeration style to be used on items within the &lt;seqlist&gt;.</p> <p>Declared Value = arabic, romanuc (upper case roman), romanlc (lower case roman), alphauc (upper case alpha), or alphalc (lower case alpha).</p> <p>Default = IMPLIED (NULL implies compliance with GPO style).</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p>
<serno  %secur;>	Equipment Serial Number	<p>Identifies an equipment serial number.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p>
<shorttitle  %secur;>	Short Title	<p>Identifies text for a short title. Short titles may be used to capture a shortened form of a title. This shortened form may then be used in a variety of ways. Examples of how it may be used include extraction for use in compiled data and capture of text for use in a running header.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p>
<sigblk  sigtype =x  %secur;>	Signature Block	<p>Identifies a signature block.</p> <p>Optional Attribute(s)</p> <p>SIGTYPE: Type of signature block as defined by the "%sigtype;" entity.</p> <p>Declared Value = preparer, approval, review, concur, or other.</p> <p>Default = IMPLIED</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element.</p> <p>Default = As appropriate for each attribute in the set.</p>
<signer	Signer	Identifies the signer of a particular signature block.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
<code>%\$secur;&gt;</code>		<p>Optional Attribute(s)</p> <p><code>%\$SECUR::</code> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<code>&lt;smrcode</code>	SMR Code	<p>Identifies the SMR code of an equipment part.</p> <p>Optional Attribute(s)</p> <p><code>%\$SECUR::</code> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<code>&lt;spanspec</code>	Spanned Column Specification	<p>Identifies a horizontal span of columns and associated attributes that can subsequently be referenced by its spanname to provide attributes repeatedly used in the entries or entry tables in several rows of the table group controlled by the group colspec, or within the specific thead, tfoot, or tbody context in which the spanspec occurs. Namest and nameend identify the first and last columns in increasing order that identify the span. The reason colname is used rather than colnum in identifying spanspec is that the names are independent of revisions that may change the number of inserted/deleted columns, as long as namest remains to the left of (has a smaller colnum than) nameend. Spanspec set on thead or tfoot override those on the containing tgroup and apply to just the thead or tfoot. Spanspec from the containing tgroup apply to tbody.</p> <p>Required Attribute(s)</p> <p><code>NAMEST:</code> Name of leftmost column of span. Names are identified in colspec of the current tgroup. Declared value = NMTOKEN</p> <p><code>NAMEEND:</code> Name of rightmost column of span. Names are identified in colspec of the current tgroup. Declared value = NMTOKEN</p> <p><code>SPANNAME:</code> Name of the horizontal span. Declared value = NMTOKEN</p> <p>Optional Attribute(s)</p> <p><code>ALIGN:</code> Text horizontal position within the column. Declared Value = left (quad flush left), center (centered), right (quad flush right), justify (both quad left and quad right), or char (align on leftmost of char, positioned by charoff). Default = "center"</p>
<code>namest =x</code>		
<code>nameend =x</code>		
<code>spanname =x</code>		
<code>align =x</code>		

## MIL-HDBK-28001

## APPENDIX A

TABLE 1. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
charoff =x		CHAROFF: For align="char", the percent of the width of the current span of columns to the left of the (left edge of) the alignment character. Declared Value = NUTOKEN Default = IMPLIED, from namest column's colspec.
char =x		CHAR: If align="char", the value is the single alignment character on which the first to occur of this character in the entry is aligned. Entries not containing this character are aligned to the left of this position. Declared Value = CDATA Default = IMPLIED, from namest column's colspec.
colsep =x		COLSEP: Default for all items in this column (within the closing group) of the table or chart. If other than zeros, display the internal column ruling to the right of the entry; if only zeros, do not display it. This permits an override of the ruling that the nameend column already provides for ruling to the right. Ignored for the last column, where the siderule setting applies. Declared Value = %yesorno; (NUMBER) Default = IMPLIED, from namest column's colspec.
rowsep =x>		ROWSEP: Default for all items in this span of columns (within the enclosing group) of the table or chart. If other than zeros, display the internal horizontal row ruling below each entry. If only zeros, do not display it. Ignored for the last row of the table, where the frame specification determines the ruling. Declared Value = %yesorno; (NUMBER) Default = IMPLIED, from namest column's colspec.
<specpara	Special Content Paragraph	Identifies a paragraph that may be used with special paragraph content, such as warnings, cautions, or notes.  Optional Attribute(s)
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<sssn	SSSN Number	Identifies system/subsystem/subject number (MIDAS) of an equipment part.  Optional Attribute(s)
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<staloc	Station Locator Diagram	Defines a station locator diagram using an external entity.

## MIL-HDBK-28001

## APPENDIX A

TABLE 1. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
name =x>		<p>Required Attribute(s)</p> <p>NAME: The value is the name of the external entity. Declared Value = ENTITY</p>
<step1  %bodyatt;  %secur;>	First Level Procedural Step	<p>Identifies a first level procedural step.</p> <p>Optional Attribute(s)</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<step2  %bodyatt;  %secur;>	Second Level Procedural Step	<p>Identifies a second level procedural step.</p> <p>Optional Attribute(s)</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<step3  %bodyatt;  %secur;>	Third Level Procedural Step	<p>Identifies a third level procedural step.</p> <p>Optional Attribute(s)</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<step4  %bodyatt;	Fourth Level Procedural Step	<p>Identifies a fourth level procedural step.</p> <p>Optional Attribute(s)</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE 1. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<step5  %bodyatt;  %secur;>	Fifth Level Procedural Step	Identifies a fifth level procedural step.  Optional Attribute(s)  %BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<step6  %bodyatt;  %secur;>	Sixth Level Procedural Step	Identifies a sixth level procedural step.  Optional Attribute(s)  %BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<step7  %bodyatt;  %secur;>	Seventh Level Procedural Step	Identifies a seventh level procedural step.  Optional Attribute(s)  %BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<step8  %bodyatt;	Eighth Level Procedural Step	Identifies an eighth level procedural step.  Optional Attribute(s)  %BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.

## MIL-HDBK-28001

## APPENDIX A

TABLE 1. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<stitle	Document Subtitle	Identifies the subtitle of the document, typically serves as the volume title.  Optional Attribute(s)
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<subfig>	Subfigures	Each subfig is one or more graphics or macrographics and implies that all the data within the subfig remains together on one display surface, such as a page. The use of subfig makes it possible to have multiple sheet figures. Many graphics, across multiple pages are within the scope of one numbered and titled figure.
<subject	Document Subject Matter	Identifies the document's subject matter.  Optional Attribute(s)
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<subpara1	First Level Subordinate Paragraph	Identifies a first level subordinate paragraph.  Optional Attribute(s)
tocentry =x		TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "0"
shortentry =x		SHORTENTRY: If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"
%bodyatt;		%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<subpara2	Second Level Subordinate Paragraph	Identifies a second level subordinate paragraph.
tocentry =x		Optional Attribute(s)  TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesomo; (NUMBER) Default = "0"
shortentry =x		SHORTENTRY: If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesomo; (NUMBER) Default = "0"
%bodyatt;		%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<subpara3	Third Level Subordinate Paragraph	Identifies a third level subordinate paragraph.
tocentry =x		Optional Attribute(s)  TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesomo; (NUMBER) Default = "0"
shortentry =x		SHORTENTRY: If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesomo; (NUMBER) Default = "0"

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.		
Element/Attribute	Full Name	Description
%bodyatt;  %secur;>		<p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<p>&lt;subpara4</p> <p>tocentry =x</p> <p>shortentry =x</p> <p>%bodyatt;</p> <p>%secur;&gt;</p>	Fourth Level Subordinate Paragraph	<p>Identifies a fourth level subordinate paragraph.</p> <p>Optional Attribute(s)</p> <p>TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "0"</p> <p>SHORTENTRY: If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<p>&lt;subpara5</p> <p>tocentry =x</p>	Fifth Level Subordinate Paragraph	<p>Identifies a fifth level subordinate paragraph.</p> <p>Optional Attribute(s)</p> <p>TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "0"</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
shortentry =x  %bodyatt;  %secur;>		<p><b>SHORTENTRY:</b> If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"</p> <p><b>%BODYATT::</b> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p><b>%SECUR::</b> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<subpara6  tocentry =x  shortentry =x  %bodyatt;  %secur;>	Sixth Level Subordinate Paragraph	<p>Identifies a sixth level subordinate paragraph.</p> <p>Optional Attribute(s)</p> <p><b>TOCENTRY:</b> If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "0"</p> <p><b>SHORTENTRY:</b> If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"</p> <p><b>%BODYATT::</b> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p><b>%SECUR::</b> Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<subpara7	Seventh Level Subordinate Paragraph	Identifies a seventh level subordinate paragraph.

## MIL-HDBK-28001

## APPENDIX A

TABLE 1. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
<p>tocentry =x</p> <p>shortentry =x</p> <p>%bodyatt:</p> <p>%secur&gt;</p>		<p>Optional Attribute(s)</p> <p>TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "0"</p> <p>SHORTENTRY: If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<p>&lt;subpara8</p> <p>tocentry =x</p> <p>shortentry =x</p> <p>%bodyatt:</p>	Eighth Level Subordinate Paragraph	<p>Identifies an eighth level subordinate paragraph.</p> <p>Optional Attribute(s)</p> <p>TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "0"</p> <p>SHORTENTRY: If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.		
Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<subscript	Subscript	Indicates a subscript. For use external to mathematical formulae.  Optional Attribute(s)
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<supscript	Superscript	Indicates a superscript. For use external to mathematical formulae.  Optional Attribute(s)
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<symbol	Symbol	Identifies a unique symbol not found in character sets, is used as a graphic in text. A symbol is stored as a graphic either as vector (MIL-PRF-28000 or MIL-PRF-28003) or raster (MIL-PRF-28002) data and is used as a symbol in the text of a document.  Required Attribute(s)
boardno =x		BOARDNO: Enter unique symbol identifier. Declared Value= ENTITY
reprowid =x		Optional Attribute(s)  REPROWID: Repro area width. Declared Value = NUTOKEN Default = IMPLIED (NULL)
reprodep =x		REPRODEP: Repro area depth. Declared Value = NUTOKEN Default = IMPLIED (NULL implies value from <macrograph>, if available, NULL if not)
hscale =x		HSCALE: Horizontal scaling. Declared Value = NUTOKEN Default = IMPLIED (NULL)

## MIL-HDBK-28001

## APPENDIX A

TABLE 1. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
vscale =x  scalefit =x  offset =x  rotation =x  %secur;>		<p>VSCALE: Vertical scaling. Declared Value = NUTOKEN Default = IMPLIED (NULL)</p> <p>SCALEFIT: Characteristic allows the symbol to be scaled as needed to fit the size of the reproduction area. Declared Value=%yesorno; (NUMBER) Default = IMPLIED (NULL)</p> <p>OFFSET: Vertical relationship to type baseline of text. A positive value will move the symbol below the baseline and a negative value will place the symbol above the baseline. Units of measure are in points. Declared Value = NMTOKEN Default = IMPLIED (NULL) (Zero points offset)</p> <p>ROTATION: Degree of rotation of the symbol. Declared Value = NUMBER Default= IMPLIED (NULL)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default: As appropriate for each attribute in the set.</p>
<ymbsect  %bodyatt;  %secur;>	Symbol Section	<p>Identifies a symbol section.</p> <p>Optional Attribute(s)</p> <p>%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<table  tabstyle =x  tocentry =x	Table	<p>Identifies a table.</p> <p>Optional Attribute(s)</p> <p>TABSTYLE: A unique table style defined in the FOSI. Declared Value = NMTOKEN Default = IMPLIED (NULL)</p> <p>TOCENTRY: If other than zeros, and the title is present, this table title should be included in the list of tables. (Ignore if the optional title is omitted). Declared Value = %yesorno; (NUMBER) Default = "1"</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
shortentry =x		<p>SHORTENTRY: If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle.</p> <p>Declared Value = %yesorno; (NUMBER)</p> <p>Default = IMPLIED (NULL)</p>
frame =x		<p>FRAME: Describes position of outer rulings.</p> <p>Declared Value = sides (left and right), top (below title), bottom (after last row possibly of tfoot material), topbot (both top and bottom), all (all of above), or none (none of above).</p> <p>Default = IMPLIED (NULL implies value from tabstyle in FOSI if available, NULL if not).</p>
colsep =x		<p>COLSEP: Default for all items in this table. If other than zeros, display the internal column rulings to the right of each item; if only zeros, do not display it. Ignored for the last column, where the frame setting applies.</p> <p>Declared Values = %yesorno; (NUMBER)</p> <p>Default = IMPLIED (NULL implies value from tabstyle in FOSI if available, NULL if not).</p>
rowsep =x		<p>ROWSEP: Default for all items in this table. If other than zeros, display the internal horizontal row ruling below each item. If only zeros, do not display it. Ignored for the last row of the table, where the frame value applies.</p> <p>Declared Value = %yesorno; (NUMBER)</p> <p>Default = IMPLIED (from tabstyle if used, NULL if not).</p>
orient =x		<p>ORIENT: Orientation of the entire chart.</p> <p>Declared Value = port (table writing direction, along rows, is the same as marginal text), or land (table writing direction is 90 degrees counterclockwise to marginal text).</p> <p>Default = IMPLIED (NULL implies value from tabstyle in FOSI, if available, NULL if not).</p>
pgwide =x		<p>PGWIDE: If other than zeros, the table runs across the page. If only zeros, the table runs across just the (galley) width of the current column of the page (regardless of orient). If the value is only zeros, it has no meaning for orient=land.</p> <p>Declared Value = %yesorno; (NUMBER)</p> <p>Default = IMPLIED (NULL implies value from tabstyle in FOSI, if available, NULL if not).</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.		
Element/Attribute	Full Name	Description
%bodyatt;  %secur;>		%BODYATT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element.
<tablelist  tocentry =x  shortentry =x>	Generated List of Tables	Identifies element that refers to a list of tables generated by the receiving system.  Optional Attribute(s)  TOCENTRY: If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "1"  SHORTENTRY: If the value consists only of zeros, the element's shorttitle (or title, if no short title is given) is not used in the coverindex or any other type of compiled listing. Any other numeric value triggers the use of the shorttitle. Declared Value = %yesorno; (NUMBER) Default = "0"
<tbody  valign =x  %secur;>	Table Body	Identifies the body of the <table>.  Optional Attribute(s)  VALIGN: Text vertical positioning within the entries. Declared Value = top, middle (approximately vertically centered), or bottom. Default = "top"  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<term  %secur;>	Term	Identifies a term, symbol, or abbreviation.  Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
<testcode  codetype =x  %content;>	Test Code	Identifies the test code for a specific test point.  Optional Attribute(s)  CODETYPE: Specifies type of testcode. Declared Value = major (major test point), minor (minor test point), or sec (secondary test point). Default = "major"  %CONTENT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<testeq  %content:  %secur;>	Test Equipment	Identifies a piece of test equipment.  Optional Attribute(s)  %CONTENT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<tfoot  valign =x  %secur;>	Table Foot	Identifies the footer information in a table or chart displayed after the tbody and also at the bottom of any tbody rows before a physical break.  Optional Attribute(s)  VALIGN: Text vertical positioning within the entries. Declared Value = top, middle (approximately vertically centered), or bottom. Default = "top"  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.		
Element/Attribute	Full Name	Description
<tgroup	Table Group	<p>Each tgroup effectively identifies a new portion of a table or chart. If a new colspec is provided, it replaces a previous one. If both colspec and spanspec are new, that spanspec should refer to columns in the most recent colspec. If only a new spanspec is provided, it should refer to columns defined by the (most immediately prior) colspec in a tgroup of the table or chart. A new colspec to either a tthead or ttfoot replaces all prior column definitions.</p> <p style="text-align: center;">Required Attribute(s)</p> <p>COLS: Number of columns in the table or chart. Declared Value = NUMBER</p> <p style="text-align: center;">Optional Attribute(s)</p> <p>TGROUPSTYLE: A unique table group style defined in the FOSI. Declared Value = NMTOKEN Default = IMPLIED (NULL)</p> <p>COLSEP: Default for items in this table group. If other than zeros, display the internal column rulings to the right of each item; if only zeros, do not display it. Ignored for the last column, where the frame setting applies. Declared Value = %yesorno; (NUMBER) Default = IMPLIED, from tgroupstyle if used, NULL if not.</p> <p>ROWSEP: Default for items in this table group. If other than zeros, display the internal horizontal row ruling below each item. If only zeros, do not display it. Ignored for the last row of the table, where the frame value applies. Declared Value = %yesorno; (NUMBER) Default = IMPLIED, from tgroupstyle if used, NULL if not.</p> <p>ALIGN: Text horizontal position within the column (or portion of it controlled by this colspec). Declared Value = left (quad flush left), center (centered), right (quad flush right), justify (both quad left and right), or char (align on leftmost of char, position by charoff). Default = "left" (unless overridden by tgroupstyle)</p>
cols =x		
tgroupstyle =x		
colsep =x		
rowsep =x		
align =x		

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
charoff =x		CHAROFF: For align = "char", percent of the current width to the left of the (left edge of) character. Declared Value = NUTOKEN Default = "50" (unless overridden by tgroupstyle)
char =x		CHAR: If align = "char", the value is the single alignment character on which the first to occur of this character in the entry is aligned. If that character does not occur in the entry, the entry aligns to the right of that character offset. Declared Value = CDATA Default = "" (unless overridden by tgroupstyle)
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<thead	Table Head	Identifies the heading information in a table or chart, displayed at the top of the table and again at the top of any continuation after a physical break between rows in tbody.
valign =x		Optional Attribute(s)  VALIGN: Text vertical positioning within the entries. Declared Value = top, middle (vertically centered), or bottom. Default = "bottom"
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<title	Title	Identifies text for a title.
%secur;>		Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<titleblk	Title Block Matter	Identifies title block material.
%secur;>		Optional Attribute(s)  %SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
<tool	Tool	Identifies a tool.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
%content;  %secur;>		<p>Optional Attribute(s)</p> <p>%CONTENT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<torqueval %secur;>	Torque Value	<p>Identifies a torque value.</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<verbatim  allowbrk=x  %secur;>	Verbatim Text	<p>Used to indicate if the text is to be picked up and laid down as it is. Typically, it implies the usage of a monospace font and the designated point size. All record ends are retained. The use of tabs in verbatim text may cause unexpected results and should therefore be avoided.</p> <p>Optional Attribute(s)</p> <p>ALLOWBRK: Specifies if the verbatim information can be broken over a page boundary. The values are numeric. A value consisting only of zeros specifies that a break is not allowed; any other number specifies that a break is allowed. Declared Value = %yesorno; (NUMBER) Default = "1"</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<volnum  %secur;>	Volume Number	<p>Identifies the number of the volume.</p> <p>Optional Attribute(s)</p> <p>%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<volume	Volume	Separates a volume of a technical manual. Each volume of a technical manual has its own subtitle, but retains the same publication number as the other volumes.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
tocentry =x  %bodyatt;  %secur;>		<p>Optional Attribute(s)</p> <p>TOCENTRY: A non-zero number causes the volume and its number to be included in the contents, illulist, and tablelist for the volume, and to propagate to any document-wide ones as well. If the value contains only zeros, the volume is not included. Declared Value = %yesomo; (NUMBER) Default = "1"</p> <p>%BODYATT: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p> <p>%SECUR: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<warning  type =x  xrefid =x  vital =x  %secur;>	Warning	<p>Identifies a warning.</p> <p>Optional Attribute(s)</p> <p>TYPE: This specifies the type of warning. This type may be used as the title of the warning. Declared Value = CDATA Default = IMPLIED (NULL)</p> <p>XREFID: This specifies a cross reference to the unique identifier of a corresponding element. Declared Value = IDREF Default = IMPLIED (NULL)</p> <p>VITAL: If the warning is not vital, use a value consisting only of zeros. Any other digit will imply the warning is vital. Vital warnings are used in the WARNSUM. Declared Value = %yesomo;(NUMBER) Default = "0"</p> <p>%SECUR: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.</p>
<warnsum	Warning Summary	Identifies the warning summary information of the document. The warning summary consists of warnings extracted from the text based on their "vital" status or warnings which apply to the document generally.

## MIL-HDBK-28001

## APPENDIX A

TABLE I. Element and attribute descriptions - Continued.

Element/Attribute	Full Name	Description
<p>inschlvl =x</p> <p>delchlvl =x</p> <p>tocentry =x</p> <p>%\$secur;&gt;</p>		<p><b>Optional Attribute(s)</b></p> <p><b>INSCHLVL:</b> Specifies the change level(s) at which information was inserted. An audit trail can be maintained by listing multiple change levels separated by spaces. Declared Value = NUTOKENS Default = IMPLIED (NULL)</p> <p><b>DELCHLVL:</b> Specifies the change level(s) at which information was deleted. An audit trail can be maintained by listing multiple change levels separated by spaces. Declared Value = NUTOKENS Default = IMPLIED (NULL)</p> <p><b>TOCENTRY:</b> If the value consists only of zeros, the element's number and title are not used in contents, illuslist, or tablelist. Any other numeric value triggers their use in the appropriate listing. Declared Value = %yesorno; (NUMBER) Default = "0"</p> <p><b>%\$SECUR::</b> Any of the attributes in the associated Attribute Set may be used with this element Default = As appropriate for each attribute in the set.</p>
<p>&lt;xref</p> <p>xrefid =x</p> <p>xidtype =x</p> <p>pretext =x</p> <p>posttext =x</p>	Cross Reference	<p>Value is the "id" of the element being referenced. The element's "id" value is replaced with the automatically assigned enumeration at the time of output. Example: step2 See figure xref xrefid="abc". At output, the processing system will insert the correct enumeration, such as: See figure 4-29.</p> <p><b>Required Attribute(s)</b></p> <p><b>XREFID:</b> Value is that of a unique identifier for cross referencing of information. Declared Value = IDREF</p> <p><b>XIDTYPE:</b> Value is that of a string for identifying the nature of the information being cross referenced. Declared Value = text, table, or figure.</p> <p><b>Optional Attribute(s)</b></p> <p><b>PRETEXT:</b> Text that will precede the cross reference when resolved for display. Declared Value = CDATA Default = IMPLIED (NULL)</p> <p><b>POSTTEXT:</b> Text that will follow the cross reference when resolved for display. Declared Value = CDATA Default = IMPLIED (NULL)</p>

**MIL-HDBK-28001****APPENDIX A**

TABLE I. <u>Element and attribute descriptions</u> - Continued.		
Element/Attribute	Full Name	Description
%secur;>		%SECUR:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set

## MIL-HDBK-28001

## APPENDIX A

## 20. ALPHABETICAL LISTING OF ATTRIBUTE DESCRIPTIONS

20.1 Attribute set descriptions. The following table describes each of the attribute types in the DTD in this appendix. The purpose of this table is to provide an alphabetically sorted, semantic description of each attribute.

20.1.1 Attribute column. The first column gives the name of the attribute. No values are given for the attributes in this column. Attribute sets are not resolved.

20.1.2 Full-name column. The second column is the natural language name of the attribute.

20.1.3 Description column. The third column provides attribute set descriptions which are listed alphabetically.

TABLE II. <u>Attribute set descriptions.</u>		
Attribute	Full Name	Description
%bodyatt;	Body Attribute Set	These attributes may be used with any element type that references this attribute set (%bodyatt;) in the document type declaration.
id = x		<p>Optional Attribute(s)</p> <p>ID: An identifier of the element which is assigned at origination and which remains unchanged as the document is revised or updated even though the automatically assigned enumeration or manually-assigned "labels" change (in some cases many times). The value of the "id" is used when making references to the element from other portions of the document. If no ID is given, none will be maintained and the element can then not be cross-referenced to by means of an IDREF on another element or with &lt;ref&gt;.</p> <p>Declared Value = ID</p> <p>Default = IMPLIED (NULL)</p>
inschlvl = x		<p>INSCHLVL: Specifies the change level(s) at which information was inserted. An audit trail can be maintained by listing multiple change levels separated by spaces.</p> <p>Declared Value = NUTOKENS</p> <p>Default = IMPLIED (NULL)</p>
delchlvl = x		<p>DELCHLVL: Specifies the change level(s) at which information was deleted. An audit trail can be maintained by listing multiple change levels separated by spaces.</p> <p>Declared Value = NUTOKENS</p> <p>Default = IMPLIED (NULL)</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE II. Attribute set descriptions - Continued-

Attribute	Full Name	Description
label =x		LABEL: Label associated with paragraph, figure, or table (i.e., chapter number). Label is only appropriate for manually enumerated documents. Typically, the rendering system will autoenumerate the elements requiring it, in which case the label attribute is omitted. Declared Value = CDATA Default = IMPLIED (NULL)
hcp =x		HCP: Hardness Critical Process - If the value consists only of zeros, there is no hardness critical information. If another value is given, the element contains hardness critical information. Declared Value = %yesorno; (NUMBER) Default = "0"
stub =x		STUB: Partial document #CONREF attribute declared value (STUB) if present the content mode for this element is EMPTY. If absent, that content model is as indicated. Used in the transmittal master for partial documents to provide placeholders for missing or unchanged GIs Declared Value = STUB Default = #CONREF
%content;		%CONTENT:: Any of the attributes in the associated Attribute Set may be used with this element. Default = As appropriate for each attribute in the set.
%content;	Content Attribute Set	These attributes may be used with any element type that references this attribute set (%content;) or the body attribute set (%bodyatt;) in the document type declaration.  Optional Attribute(s)
texttype =x		TEXTTYPE: (pending information from OSD) Declared Value = NUMBER Default = IMPLIED (NULL)
applictype =x		APPLICTYPE: This attribute references unique identifier(s) assigned to applicability definition(s) (<applicdef id='xxx'>). An example might be that the type of applicability would be aircraft tail numbers. Although it is possible to derive the applicability type from the applicability reference identifier, it may be explicitly stated with this attribute. Declared Value = IDREFS Default = IMPLIED (NULL)
applicrefid =x		APPLICREFID: References unique identifier(s) assigned to applicability identifier(s) (<applicid id='xxx'>). An example might be a particular aircraft tail number(s). Declared Value = IDREFS Default = IMPLIED (NULL)

## MIL-HDBK-28001

## APPENDIX A

TABLE II. <u>Attribute set descriptions</u> - Continued.		
Attribute	Full Name	Description
skilltrk =x		SKILLTRK: Designation of the skill level of the user at which the current element of information is aimed. A particular set of values common to all documents has not been created. Currently, the relevant values are set by contract. Declared Value = NMTOKENS Default = IMPLIED (NULL)
contype =x		CONTYPE: Identifies the content type. When used with steps, the implied value is procedural. When used with all other element types, the implied value is descriptive. Declared Value = desc (Descriptive) or proc (procedural block format). Default = IMPLIED
assocfig =x		ASSOCFIG: Identifies associated figure(s) with the text data through the use of the "id" attribute in the <figure> tag. Declared Value = IDREFS Default = IMPLIED (NULL)
assoctab =x		ASSOCTAB: Identifies associated table(s) with the text data. Declared Value = IDREFS Default = IMPLIED (NULL)
%f.style;	Style	Defines a mathematical style to be used with a formula.  Optional Attribute(s)  STYLE: Style of rules declared in the "f.style" entity. Declared Value = single, double, triple, dash, dots, or bold.
%f.align;	Mathematical Formula Alignment	Defines the mathematical alignment position within a formula.  Optional Attribute(s)  ALIGN: Defines the values declared in the "f.align" entity. Declared Value = center, left, or right.
%f.diff;	Differential	Differential tag that identifies a derivative.  Optional Attribute(s)  TYPE: Type of differential as declared in the "f.diff" entity. Declared Value = ordinary or partial
%f.type;	Type of Fence	Identifies the character that will be used as a fence (e.g., open bracket).  Optional Attribute(s)  TYPE: Identifies the character to be used for the fence. Declared Value = paren, bracket, angbrack, brace, bar, or none.

## MIL-HDBK-28001

## APPENDIX A

TABLE II. <u>Attribute set descriptions</u> - Continued.		
Attribute	Full Name	Description
%f.func;	Mathematical Function	<p>Identifies a mathematical function and its argument. By convention, most characters except numeral digits and syntax elements are set in italics in mathematical formulae. Exceptions to this convention include names of functions.</p> <p>Optional Attribute(s)</p> <p>TYPE: Defines the type of mathematical function whose values are declared in the "f.func" entity.</p> <p>Declared Value = and, antilog, arc, arccos, arcsin, arctan, arg, colog, cos, cosh, cot, coth, csc, ctn, deg, det, dim, exp, for, gcd, glb, hom, if, im, ker, lg, lim, ln, log, lub, max, min, mod, re, sec, sin, sinh, tan, tanh.</p>
%f.ov;	"Over" Embellishments	<p>Identifies parts of a formula over which special accents or diacritical marks are to be placed. (Over-Character Tag)</p> <p>Optional Attribute(s)</p> <p>TYPE: Defines the type of mark whose values are declared in the "f.ov" entity.</p> <p>Declared Value = dot, dotdot, dot3, dot4, tie, tiebrace, hat, haczeck, acute, grave, cedil, ring, macron, ogonek, dblac, breve, tilde, vec, rvec, dyad, or bar.</p>
%f.oper;	Operator	<p>Identifies an operator within a formula. Defined by the values declared in the "f.oper" entity.</p> <p>Valid operators are: = mark, markref, break, sup, sub, sum, integral, product, plex, frac, diff, sqrt, root, square, power, pile, matrix, middle, tensor, mfn, box, fence, or vec.</p>
%f.pos;	Position of Elements	<p>Identifies position of elements within a formula containing superscripts and subscripts.</p> <p>Optional Attribute(s)</p> <p>TYPE: Defines the position of elements whose values are declared in the "f.pos" entity.</p> <p>Declared Value = pre, mid, or post.</p>
%f.text;	Type of Text	<p>Identifies the type of text to be used. Defines the type of text whose values are declared in the "f.text" entity.</p> <p>Valid values are: = #PCDATA, roman, italic, or ov.</p>
%itemid;	Item Identifier Attribute Set	<p>These attributes may be used with any element type that references this attribute set (%itemid;) or the body attribute set (%bodyatt;) in the document type declaration.</p>

## MIL-HDBK-28001

## APPENDIX A

TABLE II. <u>Attribute set descriptions</u> - Continued.		
Attribute	Full Name	Description
		Optional Attribute(s)
sssn =x		SSSN: The value of this attribute would be the appropriate system/subsystem/subject number (sometimes referred to as a MIDAS number) of the equipment/part described in the text of the element type. Declared Value = CDATA Default = IMPLIED (NULL)
unit =x		UNIT: The value of this attribute would be the appropriate unit number of the equipment/part described in the text of the element type. Declared Value = CDATA Default = IMPLIED (NULL)
module =x		MODULE: The value of this attribute would be the appropriate module number of the equipment/part described in the text of the element type. Declared Value = CDATA Default = IMPLIED (NULL)
lru =x		LRU: The value of this attribute would be the appropriate line replaceable unit number of the equipment/part described in the text of the element type. Declared Value = CDATA Default = IMPLIED (NULL)
assem =x		ASSEM: The value of this attribute would be the appropriate assembly name of the equipment/part described in the text of the element type. Declared Value = CDATA Default = IMPLIED (NULL)
subassem =x		SUBASSEM: The value of this attribute would be the appropriate subassembly name of the equipment/part described in the text of the element type. Declared Value = CDATA Default = IMPLIED (NULL)
ssubassm =x		SSUBASSM: The value of this attribute would be the appropriate sub-subassembly name of the equipment/part described in the text of the element type. Declared Value = CDATA Default = IMPLIED (NULL)
compon =x		COMPON: The value of this attribute would be the appropriate component name of the equipment/part described in the text of the element type. Declared Value = CDATA Default = IMPLIED (NULL)

## MIL-HDBK-28001

## APPENDIX A

TABLE II. Attribute set descriptions - Continued.

Attribute	Full Name	Description
partno =x		PARTNO: The value of this attribute would be the appropriate part number of the equipment/part described in the text of the element type. Declared Value = CDATA Default = IMPLIED (NULL)
refdes =x		REFDES: The value of this attribute would be the appropriate reference designator of the equipment/part described in the text of the element type. Declared Value = CDATA Default = IMPLIED (NULL)
%secur;	Security Attribute Set	These attributes may be used with any element type that references this attribute set (%secur;) in the document type declaration.
		Optional Attribute(s)
security =x		SECURITY: Security classification of the element. Declared Value = u (unclassified), c (confidential), s (secret), ts (top secret) Default = IMPLIED (NULL)
restrict =x		RESTRICT: Restrictions - These might include: noforeign, nato, nocontract, proprietary, fouo, etc. Declared Value = NMTOKENS Default = IMPLIED (NULL)
release =x		RELEASE: Release Specification - Countries to which document may be released. Declared Value = NMTOKENS Default = IMPLIED (NULL)
codeword =x		CODEWORD: Associated code words Declared Value = NMTOKENS Default = IMPLIED (NULL)
scilevel =x		SCILEVEL: Flag to indicate if element has a Special Compartmentalized Information level. Declared Value = %yesorno; (NUMBER) Default = "0"
diglyph =x		DIGLYPH: One or more two-letter codes defining the classification of the element. Values are determined by contract. Declared Value = NMTOKENS Default = IMPLIED (NULL)
%yesorno;	Yes or No Attribute Set	The value of this attribute is a number. If only zeros are used, the response is "no." If any other numeric value is used, the response is "yes."

## MIL-HDBK-28001

## APPENDIX B

## COORDINATED SAMPLE DOCUMENT

## 10. SCOPE

10.1 Scope. This appendix contains the coordinated sample document type declaration, sample document instance, sample FOSI, and printed output.

## 20. SAMPLE COORDINATED DOCUMENT TYPE DECLARATION

```
<!-- Example Document Type Declaration -->
<!-- This Document Type Declaration is for illustration purposes -->
<!DOCTYPE docletter [
  <!ELEMENT docletter - - (header, salut, body, close)>
  <!ELEMENT header - o (name, city, slogan)>
  <!ELEMENT name - o (#PCDATA) >
  <!ELEMENT city - o (#PCDATA) >
  <!ELEMENT slogan - o (#PCDATA) >
  <!ELEMENT salut - o (#PCDATA) >
  <!ELEMENT body - o (title?, para)+>
  <!ELEMENT title - o (#PCDATA)>
  <!ATTLIST title status (final | draft) #IMPLIED >
  <!ELEMENT para - o (#PCDATA)>
  <!ATTLIST para status (final | draft) #IMPLIED >
  <!ELEMENT close - o (closing, name) >
  <!ELEMENT closing - o (#PCDATA) >
]>
<!-- End of Example Document Type Declaration -->
```

## 30. SAMPLE COORDINATED DOCUMENT INSTANCE

```
<!-- Example document instance -->
<docletter>
  <header>
    <name>John Q. CALS</name>
    <city>Washington D.C.</city>
    <slogan>CALS is the future!</slogan>
  </header>
  <salut>Dear Friends;</salut>

  <body>
    <para>Part of the CALS effort involves the use of the Standard Generalized
    Markup Language (SGML). SGML will provide the Department of Defense with data
```

## MIL-HDBK-28001

## APPENDIX B

that is transportable over multiple platforms. This data will be easier to maintain and be more accessible. The three components which are needed to create a technical manual in CALS with SGML are a Document Type Definition (DTD), a FOSI and a tagged instance.

</para>

<title>Document Type Definition (DTD)</title>

<para>A DTD defines the structure of the document. It serves as an electronic outline. A DTD defines tags or ELEMENTS for parts of the document as well as the content of those elements. As an example it might define the ELEMENT para to contain the content of paragraphs. Whenever a document is created using the DTD a parse can be done to ensure that the document conforms to the DTD. Many editors exist which ensure that the document being created conforms to the DTD.</para>

<title>Formatting Output Specification Instance (FOSI)</title>

<para>A FOSI controls the format of the document. If the DTD contains an ELEMENT para, the FOSI specifies how the content of the para is to be presented. A FOSI is written according to the output specification DTD of MIL-PRF-28001.</para>

<title>Tagged Instance</title>

<para>The tagged instance contains the content of the document. The content will be wrapped in the tags of the DTD. It will define pointers to graphics. A DTD and FOSI can be used to process an infinite number of tagged instances. As an example, any letter created using the tags defined in this DTD can be presented using the FOSI.</para>

<para status="draft"> One of the things to note in this example is how ATTRIBUTES are used to control the output. All titles and paragraphs can be marked as either draft or final. Those paragraphs which are draft appear in a different font. As an example this paragraph is marked as draft and is displayed using a serif font.</para>

</body>

<close>

<closing>Enjoy your work in CALS,</closing>

<name>John Q. CALS</name>

</close>

</docletter>

<!-- End of Example document instance -->

## MIL-HDBK-28001

## APPENDIX B

## 40. SAMPLE COORDINATED FOSI

```

<!-- Example FOSI -->
<outspec>
<rsrccdesc>
<hyphrule language="english" nobrkchr="123456789" minleft="3" minpush="3"
clbrkok="0" pgbrkok="0">
</rsrccdesc>
<pagedesc>
<pageset id="example.pgset">
<rectopg width="8.5in" nomdepth="11in">
<pagespec pgid="example.pgset.rspect">
<topmarg nomdepth="1in">
<botmarg nomdepth="1in">
<leftmarg width="1in">
<rtmarg width="1in">
<header nomdepth="24pt">
</header>
<footer></footer>
<flowtext numcols="1">
<column width="6.5in"></flowtext>
</pagespec>
</rectopg>
</pageset>
</pagedesc>
<styldesc>
<docdesc>
<charlist>
<font inherit="1" style="sanserif" size="12pt" posture="upright"
weight="medium"
smallcap="0" offset="0pt">
<leading inherit="0" lead="12pt">
<hyphen>
<wordsp>
<lettersp>
<indent inherit="1" leftind="0pt" rightind="0pt" firstln="">
<quadding inherit="0" quad="left" lastquad="relative">
<highlt>
<chgmark></chgmark>
<presp minimum="12pt" nominal="12pt" maximum="12pt">
<postsp minimum="12pt" nominal="12pt" maximum="12pt">
<keeps>
<vjinfo>
<textbrk startln="1">
<span>
<border>
<float>
<algroup>
<suppress sup="1">
<boxing>
<enumerat increm="1"></charlist>
</docdesc>
<e-i-c gi="docletter">
<charlist>
<textbrk startpg="recto" pageid="example.pgset" newpgmdl="global" startln="1">
</charlist>
</e-i-c>
<e-i-c gi="header">
<charlist>
<quadding quad="center" lastquad="lleft"></charlist>

```

## MIL-HDBK-28001

## APPENDIX B

```

</e-i-c>
<e-i-c gi="city">
<charlist inherit="1">
<quadding quad="center" lastquad="lcenter">
<textbrk startln="1"></charlist>
</e-i-c>
<e-i-c gi="slogan">
<charlist inherit="1">
<quadding quad="center" lastquad="lcenter">
<textbrk startln="1"></charlist>
</e-i-c>
<e-i-c gi="salut">
<charlist>
<presp minimum="18pt" nominal="18pt" maximum="18pt">
<postsp>
<textbrk startln="1"></charlist>
</e-i-c>
<e-i-c gi="body">
<charlist></charlist>
</e-i-c>
<e-i-c gi="title">
<charlist>
<quadding quad="center" lastquad="lcenter">
<textbrk startln="1" endl="1"></charlist>
<att>
<specval attname="status" attval="draft">
<charsubset>
<font style="serif"></charsubset>
</att>
</e-i-c>
<e-i-c gi="para">
<charlist>
<indent leftind="0pt" firstln="3em">
<presp>
<textbrk startln="1"></charlist>
<att>
<specval attname="status" attval="draft">
<charsubset>
<font style="serif"></charsubset>
</att>
</e-i-c>
<e-i-c gi="close">
<charlist></charlist>
</e-i-c>
<e-i-c gi="closing">
<charlist>
<quadding quad="center">
<presp>
<textbrk startln="1"></charlist>
</e-i-c>
<e-i-c gi="name">
<charlist>
<quadding quad="center">
<presp>
<textbrk startln="1"></charlist>
</e-i-c>
</stylDESC>
</outspec>

<!-- End of Example FOSI -->

```

**MIL-HDBK-28001****APPENDIX B****50. SAMPLE COORDINATED OUTPUT**

John Q. CALS

Washington D.C.

CALS is the future!

Dear Friends:

Part of the CALS effort involves the use of the Standard Generalized Markup Language (SGML). SGML will provide the Department of Defense with data that is transportable over multiple platforms. This data will be easier to maintain and be more accessible. The three components which are needed to create a technical manual in CALS with SGML are a Document Type Definition (DTD), a FOSI, and a tagged instance.

**Document Type Definition (DTD)**

A DTD defines the structure of the document. It serves as an electronic outline. A DTD defines tags or ELEMENTS for parts of the document as well as the content of those elements. As an example it might define the ELEMENT para to contain the content of paragraphs. Whenever a document is created using the DTD a parse can be done to ensure that the document conforms to the DTD. Many editors exist which ensure that the document being created conforms to the DTD.

**Formatting Output Specification Instance (FOSI)**

A FOSI controls the format of the document. If the DTD contains an ELEMENT para, the FOSI specifies how the content of the para is to be presented. A FOSI is written according to the output specification DTD of MIL-PRF-28001.

**Tagged Instance**

The tagged instance contains the content of the document. The content will be wrapped in the tags of the DTD. It will define pointers to graphics. A DTD and FOSI can be used to process an infinite number of tagged instances. As an example, any letter created using the tags defined in this DTD can be presented using the FOSI.

One of the things to note in this example is how ATTRIBUTES are used to control the output. All titles and paragraphs can be marked as either draft or final. Those paragraphs which are draft appear in a different font. As an example, this paragraph is marked as draft and is displayed using a serif font.

Enjoy your work in CALS.

John Q. CALS

**FIGURE 17. Sample output.**

## MIL-HDBK-28001

## APPENDIX C

## MATHPAC TAG SET

## 10. MATHPAC TAG SET DESCRIPTIONS

**10.1 Scope.** This appendix contains the mathpac tag set descriptions. See section 8 for guidance in applying and tagging mathpac elements.

**10.2 Mathematical element type descriptions.** Descriptions of the element types used within the &mathpac; PUBLIC entity are reproduced with the permission of the International Organization for Standardization (ISO), from ISO/IEC/TR 9573:1988 Information processing - SGML support facilities - Techniques for using SGML.

TABLE III. <u>Mathpac tag set descriptions.</u>		
Element/ Attribute	Full Name	Description
<above>	Above	Identifies the next element in a vertical list.
<above1>	Above1	Identifies the top element where a start and end tag may be omitted. This is followed by one or more above elements, where the start tag is required.
<box style=x>	Box	Identifies parts of a formula to be placed inside a box.  Optional Attribute(s)  STYLE: Style of the box's rules declared in the "f.style" entity. Declared Value = single, double, triple, dash, dots, or bold. Default = "single"
<break type =x>	Formula Breakpoint	Identifies a division point (for horizontal alignment break) within a formula.  Optional Attribute(s)  TYPE: Specifies whether formula breakpoint is required or not. Declared Value = required or optional. Default = "required"
<by>	Derivative Denominator	Identifies the denominator of a derivative.
<col align =x>	Matrix Column	Identifies a column of a matrix.  Optional Attribute(s)  ALIGN: Horizontal alignment of the characters within a matrix as defined in the "f.align" entity. Declared Value = center, left, or right. Default = "center"

## MIL-HDBK-28001

## APPENDIX C

TABLE III. Mathpac tag set descriptions - Continued.		
Element/ Attribute	Full Name	Description
<degree>	Degree	Identifies the power to which a quantity is to be raised or the root to be taken of a quantity. The content of the "degree" tag would be "3" for a cubed quantity (tagged in terms of "power") and "4" for the fourth root of a quantity (tagged in terms of "root").
<df  id =x  align =x  num =x>	Display Formula	Identifies a display formula.  Optional Attribute(s)  ID: Unique identifier of the display formula. Declared Value = ID Default = IMPLIED (NULL)  ALIGN: Horizontal alignment of the display formula as declared in the f.align entity. Declared Value = center, left, or right. Default = left  NUM: Specifies an explicit formula number; if omitted, sequential numbering of the formulae would normally be performed by the text formatter. Declared Value = CDATA Default = IMPLIED (NULL)
<dfg  id =x  align =x  num =x>	Display Formula Group	Identifies a formula group whose content is display formulae.  Optional Attribute(s)  ID: Unique identifier of the display formula. Declared Value = ID Default = IMPLIED (NULL)  ALIGN: Horizontal alignment of the display formula as declared in the f.align entity. Declared Value = center, left, or right. Default = left  NUM: Specifies an explicit formula group number; if omitted, sequential numbering of the formulae would normally be performed by the text formatter. Declared Value = CDATA Default = IMPLIED (NULL)
<dfref  refid =x>	Formula Reference	Identifies a reference to a formula or formula group.  Required Attribute(s)  REFID: Reference to the unique identifier of a <df>. Declared Value = IDREF

## MIL-HDBK-28001

## APPENDIX C

TABLE III. Mathpac tag set descriptions - Continued.		
Element/ Attribute	Full Name	Description
page =x		<p>Optional Attribute(s)</p> <p>PAGE: Page number is added (default) to the reference of the unique identifier of a &lt;df&gt;.</p> <p>Declared Value = yes or no.</p> <p>Default = "yes"</p>
<diff	Differential	Differential tag that identifies a derivative.
type =x>		<p>Optional Attribute(s)</p> <p>TYPE: Type of differential as declared in the "f.diff" entity.</p> <p>Declared Value = ordinary or partial.</p> <p>Default = "ordinary"</p>
<diffof>	Derivative Numerator	Identifies the numerator of the derivative.
<f>	Inline Formula	Identifies an inline formula.
<fence	Fence	Identifies a fence (e.g., open bracket).
type =x		<p>Optional Attribute(s)</p> <p>TYPE: Identifies the character to be used for the fence.</p> <p>Declared Value = paren, bracket, angbrack, brace, bar, or none.</p> <p>Default = paren</p>
open =x		<p>OPEN: Unpaired fences, also single occurrences, are entered using the fence element and specifying the open and close attributes. The value of these attributes is the fence character to be used.</p> <p>Declared Value = CDATA</p> <p>Default = IMPLIED (NULL)</p>
close =x		<p>CLOSE: Unpaired fences, also single occurrences, are entered using the fence element and specifying the open and close attributes. The value of these attributes is the fence character to be used.</p> <p>Declared Value = CDATA</p> <p>Default = IMPLIED (NULL)</p>
style =x>		<p>STYLE: Defines the style of the fence whose values are declared in the f.style entity.</p> <p>Declared Value = single, double, triple, dash, dots, or bold.</p> <p>Default = single</p>
<fname>	Function Name	Identifies a function name associated with the "<mfn>" tag.
<frac	Fraction	Identifies a fraction.

## MIL-HDBK-28001

## APPENDIX C

TABLE III. Mathpac tag set descriptions - Continued.		
Element/ Attribute	Full Name	Description
$\text{align} = x >$		<p>Optional Attribute(s)</p> <p>ALIGN: Horizontal alignment of the fraction as declared in the f.align entity. Declared Value = center, left, or right. Default = center</p>
$\langle \text{from} \rangle$	From	Identifies mathematical lower limit.
$\langle \text{integral} \rangle$	Integral	Identifies an integral (special case of the general plex).
$\langle \text{italic} \rangle$	Italic	By convention, most characters except numeral digits and syntax elements are set in italics in mathematical formulae. Exceptions to this convention include names of functions.
$\langle \text{mark} \rangle$	Mark	<p>Identifies a division point (for horizontal alignment break) within a formula.</p> <p>Required Attribute(s)</p> <p>ID: Used when referring to a mark and must be a unique reference in the document. Declared Value = ID</p>
$\langle \text{markref} \rangle$	Mark Reference	<p>Identifies a division point (for horizontal alignment break) within a formula. It causes the system to align the formula horizontally on that mark.</p> <p>Required Attribute(s)</p> <p>REFID: Value is a valid identifier to which the markref refers. Declared Value = IDREF</p>
$\langle \text{matrix} \rangle$	Matrix	Identifies a matrix. They are different from piles in that row elements of the matrix will always align horizontally rather than just occupy the required vertical space in the column.
$\langle \text{mfn} \rangle$	Mathematical Function	<p>Identifies a mathematical function and its argument. By convention, most characters except numeral digits and syntax elements are set in italics in mathematical formulae. Exceptions to this convention include names of functions.</p> <p>Optional Attribute(s)</p> <p>TYPE: Type of mathematical function whose values are declared in the f.func entity. Declared Value = and, antilog, arc, arccos, arcsin, arctan, arg, colog, cos, cosh, cot, coth, csc, ctn, deg, det, dim, exp, for, gcd, glb, hom, if, im, key, lg, lim, ln, log, lub, max, min, mod, re, sec, sin, sinh, tan, or tanh. Default = IMPLIED (NULL)</p>
$\langle \text{middle} \rangle$	Middle	Identifies a post.

## MIL-HDBK-28001

## APPENDIX C

TABLE III. Mathpac tag set descriptions - Continued.		
Element/ Attribute	Full Name	Description
style =x>		Optional Attributes  STYLE: Style of the post as declared in the "f.style" entity. Declared Value = single, double, triple, dash, dots, or bold. Default = single
<numer>	Numerator	Identifies the numerator of the fraction.
<of>	Limit Of Plex	Identifies limit for a plex. Used with from and to to define mathematical upper and lower limits.
<operator>	Operator	Identifies an mathematical operator.
<over>	Denominator	Identifies the denominator of a fraction.
<ov>	"Over" Embellishments	Identifies parts of a formula over which special accents or diacritical marks are to be placed. (Over-Character Tag)  Optional Attribute(s)  TYPE: Defines type of mark whose values are declared in the f.ov entity. Declared Value = dot, doidot, dot3, dot4, tie, tiebrace, hat, caron, acute, grave, cedil, ring, macron, ogonek, dblac, breve, tilde, vec, rvec, dyad, or bar. Default = bar  POS: Defines position of the mark. Declared Value = above, below, or mid. Default = above  STYLE: Defines style of the mark as declared in the f.style entity. Declared Value = single, double, triple, dash, dots, or bold. Default = single
<pile>	Pile	Identifies a vertical grouping of segments of a formula. Commonly used for arranging equations in columns (i.e., fibonacci sequence).  Optional Attribute(s)  ALIGN: Horizontal alignment of the pile as declared in the f.align entity. Declared Value = center, left, or right. Default = center
align =x>		
<plex>	Plex	Identifies a general limits operator.
<power>	Power	Identifies a power. The number of times as indicated by an exponent that a number occurs as a factor in a product.

## MIL-HDBK-28001

## APPENDIX C

TABLE III. Mathpac tag set descriptions - Continued.		
Element/ Attribute	Full Name	Description
<product>	Product	Identifies a product (special case of the general plex).
<roman>	Roman	Identifies the rare case where parts of a formula should be set in a roman font, contrary to common practice and not being a function name.
<root>	Root	Identifies a root. (The nth root of the number X, is defined to be a number Y such that Y to the N equals X).
<sqr>	Square Root	Identifies a square root. (The square root of the number X, is defined to be a number Y such that Y multiplied by itself equals X.)
<square>	Square	Identifies a square. The product of a number multiplied by itself.
<sub>  pos =x>	Subscript	Indicates a subscript. For use only within mathematical formulae. NOTE: Within text use subscript instead.  Optional Attribute(s)  POS: Position of subscript. Declared Value = pre, mid, or post. Default = post
<sum>	Sum	Identifies a sum (special case of the general plex).
<sup>  pos =x>	Superscript	Indicates a superscript. For use only within mathematical formulae. NOTE: Within text use supscrpt instead.  Optional Attribute(s)  POS: Position of superscript. Declared Value = pre, mid, or post. Default = post
<tensor>  posf =x  suffix =x>	Tensor	Identifies a tensor. A generalized vector with more than three components each of which is a function of the coordinates of an arbitrary point in space of an appropriate number of dimensions.  Optional Attribute(s)  POSF: The value of the position of the first suffix attribute. Declared Value = sub (subscript) or sup (superscript). Default = sup  Required Attribute(s)  SUFFIX: The suffix attribute takes the suffixes of the tensor; a space indicating a switch from superscript to subscript or subscript to superscript. Declared Value = CDATA
<to>	To	Identifies mathematical upper limit.

## MIL-HDBK-28001

## APPENDIX C

TABLE III. <u>Mathpac tag set descriptions</u> - Continued.		
Element/ Attribute	Full Name	Description
<vec>	Vector	Identifies a vector in a formula. By convention these are set in bold roman or medium italic with an arrow above.

## MIL-HDBK-28001

## APPENDIX C

TABLE IV. <u>Mathpac attribute set descriptions.</u>		
Attribute	Full Name	Description
%f.align;	Mathematical Formula Alignment	<p>Defines the mathematical alignment position within a formula.</p> <p>Optional Attribute(s)</p> <p>ALIGN: Defines the values declared in the f.align entity. Declared Value = center, left, or right.</p>
%f.diff;	Differential	<p>Differential tag that identifies a derivative.</p> <p>Optional Attribute(s)</p> <p>TYPE: Type of differential as declared in the f.diff entity. Declared Value = ordinary or partial</p>
%f.func;	Mathematical Function	<p>Identifies a mathematical function and its argument. By convention, most characters except numeral digits and syntax elements are set in italics in mathematical formulae. Exceptions to this convention include names of functions.</p> <p>Optional Attribute(s)</p> <p>TYPE: Defines the type of mathematical function whose values are declared in the f.func entity. Declared Value = and, antilog, arc, arccos, arcsin, arctan, arg, colog, cos, cosh, cot, coth, csc, ctn, deg, det, dim, exp, for, gcd, glb, hom, if, im, ker, lglim, ln, log, lub, max, min, mod, re, sec, sin, sinh, tan, or tanh.</p>
%f.oper;	Operator	<p>Identifies an operator within a formula. Defined by the values declared in the f.oper entity.</p> <p>Valid operators are: = mark, markref, break, sup, sub, sum, integral, product, plex, frac, diff, sqrt, root, square, power, pile, matrix, fence, middle, tensor, mfn, box, or vec.</p>
%f.ov;	"Over" Embellishments	<p>Identifies parts of a formula over which special accents or diacritical marks are to be placed. (Over-Character Tag)</p> <p>Optional Attribute(s)</p> <p>TYPE: Defines the type of mark whose values are declared in the f.ov entity.</p> <p>Declared Value = dot, dotdot, dot3, dot4, tie, tiebrace, hat, haczeck, acute, grave, cedil, ring, macron, ogonek, dblac, breve, tilde, vec, rvec, dyad, or bar.</p>
%f.pos;	Position of Elements	<p>Identifies position of elements within a formula containing superscripts and subscripts.</p>

## MIL-HDBK-28001

## APPENDIX C

TABLE IV. Mathpac attribute set descriptions - Continued.		
Attribute	Full Name	Description
		<p>Optional Attribute(s)</p> <p>TYPE: Defines the position of elements whose values are declared in the f.pos entity. Declared Value = pre, mid, or post.</p>
%f.style:	Style	<p>Defines a mathematical style to be used with a formula.</p> <p>Optional Attribute(s)</p> <p>STYLE: Style of rules declared in the f.style entity. Declared Value = single, double, triple, dash, dots, or bold.</p>
%f.text:	Type of Text	<p>Identifies the type of text to be used. Defines the type of text whose values are declared in the f.text entity. Valid values are: = #PCDATA, roman, italic, or ov.</p>
%f.type:		<p>Identifies the character that will be used as a fence (e.g., open bracket).</p> <p>Optional Attribute(s)</p> <p>TYPE: Identifies the character to be used for the fence. Declared Value = paren, bracket, angbrack, brace, bar, or none.</p>

## MIL-HDBK-28001

## APPENDIX D

## MODREQ TAG SET

## 10. MODREQ TAG SET DESCRIPTIONS

10.1 Scope. This appendix contains the modreq tag set descriptions. See section 9 for guidance in applying and using the modification request (electronic review) elements.

TABLE V. <u>Modreq tag set descriptions.</u>		
Element/Attribute	Full Name	Description
<modreq	Modification Request	Identifies a request made by a reviewer to add new text to a document or to modify a document element or a portion of a document element  Required Attribute(s)
id=x		ID: Unique identifier for the modification request. Declared Value = ID
by=x		BY: Value is the name of the reviewer requesting the modification to the document. Declared Value = CDATA
date=x		DATE: Value is the date of modification request. Declared Value = CDATA
		Optional Attribute(s)
xref=x		XREF: Value is the id of the element being referenced (i.e., the element with which the modification request is associated). Declared Value = NMTOKEN Default=IMPLIED (NULL)
refpos=x		REFPOS: Indicates the location of proposed change in relation to the element being cross-referenced. Declared Value = prexref (immediately preceding the cross-referenced by the modreq), postxref (immediately following the element cross-referenced), or xref (the cross-referenced element). Default = xref
organiz=x		ORGANIZ: Value is the identification number assigned to the organization to which the reviewer submitting the modification request belongs. Examples of organizations are Army, Navy, AF, DLA, OtherDod, CivAgency, Labs, etc. Declared Value = NMTOKEN Default - IMPLIED (NULL)

## MIL-HDBK-28001

## APPENDIX D

TABLE V. Modreq tag set descriptions - Continued.

Element/Attribute	Full Name	Description
orgcat=x		<p>ORGCAT: Value is the identification number assigned to the organization category to which the reviewer submitting the modification request belongs. Examples of organization categories are Public and Private.</p> <p>Declared Value = NMTOKEN</p> <p>Default = IMPLIED (NULL)</p>
cmntrcat=x		<p>CMNTRCAT: Value is the identification number assigned to the commentor category to which the reviewer belongs. Examples of the commentor categories within an organization in the Public Sector are Document Custodian, Reviewer, and User.</p> <p>Declared Value = NMTOKEN</p> <p>Default = IMPLIED (NULL)</p>
priority=x		<p>PRIORITY: Weighing by the reviewer of the criticality of the modification request, based on a numeric scale of one thru five, where values associated with the numbers are assigned by the responsible organization. Examples of values are Top Priority as 1, and Low Priority as 5.</p> <p>Declared Value = 1, 2, 3, 4, or 5</p> <p>Default = IMPLIED (NULL)</p>
category=x		<p>CATEGORY: Value is the identification number associated with a comment category defined by the responsible organization. Sample comment categories that might be associated with numbers are General as 1, Specific as 2, Editorial as 3, and Technical as 4, etc.</p> <p>Declared Value = NMTOKEN</p> <p>Default = IMPLIED (NULL)</p>
topic=x>		<p>TOPIC: Value is the name of a topic or concept (i.e., a sort key) to which the modification request relates. The formate and semantics of the topic field are left up to the application.</p> <p>Declared Value = CDATA</p> <p>Default = IMPLIED (NULL)</p>

## MIL-HDBK-28001

## APPENDIX D

TABLE V. Modreq tag set descriptions - Continued.

Element/Attribute	Full Name	Description
<mrchgtxt	Modification Request Change Text	Identifies 1) the addition of new text (containing markup) proposed by the reviewer, 2) replacement text (containing markup) proposed by the reviewer for the element identified by this modreq's xref attribute, or 3) a text element deletion proposed by the reviewer. In all cases, the contents (including markup) of the mrchgtxt element -- or lack of contents, in the case of a deletion -- should result in a valid document. If a valid document cannot be achieved through the use of the mrchgtxt element, then a general modification should be used.
chgloc=x		<p>Optional Attribute(s)</p> <p>CHGLOC: Value is the offset from the start tag of the element cross-referenced by the modreq to the start of the proposed change. The offset "unit" is a character, entity, empty tag, start tag, end tag, or processing instruction (the use of a normalized document is assumed).</p> <p>Note that the chgloc is intended to assist the document owner in locating the start of a proposed change within the element cross-referenced by the modreq (e.g., through highlighting, in an electronic display). Chgloc is not intended to support the actual incorporation of a proposed change. The actions of insert, replace, and delete are carried out on the entire contents of mrchgtxt (i.e., on the entire element cross-referenced by the modreq).</p> <p>Declared Value = NUMBER Default = IMPLIED (NULL)</p>
chglen=x		<p>CHGLEN: Value is the offset from the chgloc to the end of the proposed change, using the same offset unit defined for chgloc, above.</p> <p>Note that the chglen is intended to assist the document owner in locating the end of a proposed change within the element cross-referenced by the modreq, in a manner similar to that described for chgloc, above.</p> <p>Declared Value = NUMBER Default = IMPLIED (NULL)</p>

## MIL-HDBK-28001

## APPENDIX D

TABLE V. Modreq tag set descriptions - Continued.

Element/Attribute	Full Name	Description
action=x>		<b>ACTION:</b> Value is the specific action required to support incorporation of the proposed change text, in the event that it has been approved by the document owner. Declared Value = insert, delete, or replace. Default = replace
<mrngenmod>	Modification Request General Modification	Identifies a general modification to a document proposed by a reviewer. This element is to be used for any proposed change other than one in which specific new text or replacement text, or a specific element to be deleted, is suggested (in which case, the mrchgtxt element is to be used).
<mrinfo>	Modification Request Information	Contains information identifying the base document instance (including revision or change level) to which the modreq refers. This information matches the identical set of elements in the base document. It may be omitted for a given modreq in a case where 1) the modreq is contained within the referenced document, or 2) the modreq is contained within a modreq-only document with reference to the base document via the mrinfo element at the top of the file.
<mrinstr>	Modification Request Instruction	Identifies instructions for incorporating the proposed replacement text (see the mrchgtxt element), in the event that the modification is accepted by the responsible organization.
<mritem>	Modification Request Item	Identifies an item in an mrlist contained in a modification request.
<mrlist>	Modification Request List	Identifies a list within the explanatory components of a modification request.
<mrmod>	Modification Request Modification	Identifies a modification that a reviewer is requesting to a document.
<mrpara>	Modification Request Paragraph	Identifies a paragraph within the explanatory components of a modification request.
<mrreason>	Modification Request Reason	Identifies a reviewer's reason for requesting a modification to a document.
<mrrespns>	Modification Request Response	Identifies a response (by the organization responsible for coordinating the review) to a proposed modification request to a document.

## MIL-HDBK-28001

## APPENDIX D

TABLE V. Modreq tag set descriptions - Continued.		
Element/Attribute	Full Name	Description
disposn = x		<p><b>Optional Attribute(s)</b></p> <p><b>DISPOSN:</b> Value is the identification number assigned to the disposition selected by the responsible review organization as appropriate for the modification request. Sample dispositions that might be associated with numbers are Accept as 1, Reject as 2, Modify as 3, Defer as 4, Withdraw as 5 and Duplicate as 6.  <b>Declared Value = NMTOKEN</b>  <b>Default = IMPLIED (NULL)</b></p>
status = x>		<p><b>STATUS:</b> Value is the identification number assigned to the <i>current status of the response</i> to the modification request. Sample statuses that might be associated with numbers are Complete as 1, and Pending as 2.  <b>Declared Value = NMTOKEN</b>  <b>Default=IMPLIED (NULL)</b></p>

**MIL-HDBK-28001**

**APPENDIX E**

**MODULAR DTD**

**10. MODULAR DTD**

**10.1 Scope.** This appendix is reserved for a description of modular DTDs.

## MIL-HDBK-28001

## APPENDIX F

## ELEMENT AND ATTRIBUTE INDEX

## 10. ELEMENT INDEX

10.1 Scope. This appendix contains the index of elements and attributes in the handbook.

Above	42, 44, 45
Appendix	25, 27
Applicabil	23
Arbtext	128
Assem	34
Assembly	34
Att	92
Attloc	91
Attname	91
Attspec	56
Attval	92
Author	25, 26
Barthick	78
Below	44
Biblio	25, 27
Boardno	62, 99
Body	20, 21, 25-27, 34, 49, 50, 128
Boffset	78
Bold	44
Book	25, 26
Botfloat	62
Botmarg	58
Boxing	78
Break	44
Brel	78
Brk	128
By	41, 127
Category	127
Cellatts	116
Cfid	80
Change	23, 29
Chapter	20, 25, 34, 55
Chapters	26
Charfill	56, 80
Charlist	62, 92, 121
Charsubset	74, 76
Checklist	34
Chgdate	126, 127
Chgmark	57
Chgnum	126, 127

## MIL-HDBK-28001

## APPENDIX F

Clbrkok	76
Cmntcat	127
Col	42
Colatts	116
Colnum	120
Colspec	120
Concluding material	34
Conrule	62, 82, 85
Contents	27
Context	62
COORDEND	99, 100, 103-105, 107, 110
COORDST	99, 100, 103-105, 107, 110
Counter	56, 80
Dash	44
Dataiden	23
Date	21, 49, 127
Day	28
Def	85, 86
Desc	113
Description	20
Df	39, 42, 44-47
Dfg	39, 45
Diff	41
Doc	27, 29
Docpart	128
Dots	44
Double	44
E-i-c	55
Emergency	23
Emphasis	23, 55
Endln	54
Enumerat	56, 80
Enumid	80
F	39-45
Famname	76
Fence	45
Figure	62, 64, 67, 69, 70, 72
Fillcat	92
Fillchar	92
Fillval	91, 116
Firstln	77
Flidref	62
Float	62
Floatid	62
Floatloc	56, 62
Floattyp	62

## MIL-HDBK-28001

## APPENDIX F

Flowtext	57, 58, 62
Fname	47
Font	75
Footer	58
Footnotes	121
Frac	40, 45-47
Frame	118
From	21, 41, 42, 49
Front	25-27, 128
Ftnatts	121
Findesc	54, 74, 121
Finote	24, 28, 54, 124, 128
Finref	23, 123, 124
Fint	123
Glossary	25, 27
Graphchars	62, 97
Graphenv	62, 97
Graphic	23, 62, 64, 66, 67, 69, 70, 72, 99, 101, 102, 104-108, 110, 126, 128
Graphname	81
Graphsty	100, 110
Graphsty id	100
Grphdesc	62, 74, 97
Header	20, 49, 58
Hplace	98, 100, 103-105, 107
Hrule	23, 128
Hscale	98, 100, 103-105, 107-110
Hyphrule	56, 76
Id	45, 62, 127
Inclr	78
Increment	80
Indent	77
Index	25, 27
Indxflag	23
Inherit	77
Inpct	78
Integral	41
Introduction	34
Italic	44
Item	21, 27, 49, 85, 86
Keep	63
Keeps	63
Language	76
Lastquad	77
Lead	76
Leading	76
Lefigap	77

## MIL-HDBK-28001

## APPENDIX F

Leftind	60, 77
Leftmarg	58
List	20, 21, 27, 28
Llcordra	98, 100-102, 104, 105, 107, 109, 110
Loffset	78
MACROGRAPH	100
Maint	34
Maintenance	20
Maintenance,	34
Mark	45
Markref	45
Material	23
Matrix	42
Maxdepth	62
Maxspace	62
Memo	20, 49
Mfn	43, 47
Mid	44
Middle	47
Minleft	76
Minpush	76
Minspace	62
Modreq	1, 125, 127, 128
Month	28
Mrchgtxt	127
Mrgenmod	127
Mrinfo	127, 128
Mrinstr	127
Mritem	127
Mrlist	126, 128
Mrmod	127
Mrpara	126-128
Mrreason	127
Mrrespns	127
Name	21, 22, 49, 50
Nobrkchr	76
Nomspace	62
Number	97
Occur	75
Of	41, 42
Operation	20
Operator	47
Organiz	127
Orgcat	127
Ov	44
Over	40, 45-47

## MIL-HDBK-28001

## APPENDIX F

Pagedesc	56, 58, 62, 91
Pageres	60
Pageset	56, 57, 62, 91
Pagespec	57, 58, 80
Pagetype	62
Pairtrk	76
Para	20, 21, 24, 27, 28, 34, 49, 50, 62-64, 66, 67, 69, 71, 72, 75, 123, 124
Para0	34, 62-64, 66, 67, 69, 70, 72
Paragraph	55
Part	55
Part identifier	34
Partnum	113
Pgbrk	128
Pgbrkok	76
Pgid	58
Pgidref	58
Phonext	21, 49
Pile	45
Placement	81
Placemnt	54, 63, 98
Plex	42
Pos	44
Postsp	60, 79
Power	40
Presp	60, 79
Priority	127
Product	42
Pseudo-elements	91
Pubdate	126, 127
Pubno	126, 127
Putgraph	81
Puttext	60, 81
Quad	77
Quadding	77
Rear	25-27, 128
Recto	56
Ref	113
Refpos	127
Release	21, 49
Remove	34
Repair	20
Replace	34
Repro	97
Reprodep	98, 100, 104-108, 110
Reprowid	98, 100, 104-108, 110
Revnum	126, 127

## MIL-HDBK-28001

## APPENDIX F

Rightgap	77
Rightind	60, 77
Roffset	78
Roman	44
Root	41
Rotation	99, 101, 103-105, 107, 109
Route	21, 49
Rowatts	116
Rsrcdesc	56, 62, 80
Rtmarg	58
Rtype	78
Rucordra	101
Safesum	27
Savetext	62, 82, 88
Scalefit	97, 98, 100, 103, 104, 109, 110
Scope	62, 87
Scorechr	78
Scoreoff	78
Scorewt	78
Secdsc	56, 60
Sectext	58, 60
Section	20, 25, 34, 55, 91
Sections	26
Sectoken	60
Secur	34
Security	20, 29, 34, 35, 60
Service	29
Siderel	77, 78
Sign	20, 49
Significant record ends	54
Single	44
Size	54
Sizing	97
Source	63
Spafter	80
Span	80
Spbefore	80
Specval	91
Sqrt	41
Square	41
Startln	74, 77
Stdcellatts	116
Step1	34, 62-64, 66, 67, 69, 71, 72
String	87
Stringdecl	56
Stub	49, 50

## MIL-HDBK-28001

## APPENDIX F

Styldesc	62, 74
Sub	40, 42, 46
Subchars	58
Subfigure	64
Subhd	88
Subj	21, 49
Subparal	62-64, 66, 67, 69, 71, 72
Subscript	23
Subsect	25
Subsection	34
Subsetatts	115, 116
Subsystem	55
Sum	42
Sup	40, 47, 62
Suppress	62
Supscript	23
Symbol	28, 126, 128
System	55
Tabatts	116
Tabdesc	74
Table	28, 35
Task	20
Tensor	43
Test	20
Test step	55
Testeq	23
Textbrk	79, 80
Textid	62, 82
Textwidth	118
Tgroup	112
Tgroupatts	115, 116
Thick	78
Title	20, 21, 25, 26, 34, 49, 62-64, 66, 67, 69, 70, 72, 91
Titlepg	27
To	21, 41, 42, 49
Toffset	78
Tool	23
Tools	113
Topfloat	62
Topic	88, 127
Topmarg	58
Torqueval	23
Trel	78
Triple	44
Trksectr	76
Ttype	78

**MIL-HDBK-28001****APPENDIX F**

Urcordra .....	98, 104, 107, 109, 110
Usemap .....	38
Userule .....	86, 87
Usetext .....	60, 63
Verbatim .....	23
Verso .....	56
Vjinfo .....	79
Volume .....	128
Vplace .....	100, 103-105, 107
Vquad .....	58, 60
Vrplace .....	98
Vscale .....	98, 100, 103-105, 107-110
Warning .....	34
Xref .....	23, 62, 127
Xrefid .....	62
Year .....	28

**MIL-HDBK-28001**

**CONCLUDING MATERIAL**

**Custodians:**

Army - TM  
Navy - CH  
Air Force - 16  
DLA - DH

**Preparing Activity:**

DISA - DC-5  
(Project IPSC-0316)

**Review Activities:**

Army - AC, AL, AR, AT, AV, CR-1, CU, ER, GL, MI, ME, MR, SC, SM, TE  
Navy - AS, EC, OS, SA, YD-2  
Air Force - 01, 02, 11, 13, 17, 19, 68, 79, 99  
NSA - NS  
DIA - DI  
DLA - CS, ES, GS, IS  
Coast Guard - CG  
NASA - NA  
Other - DOE, NCS  
OASD - DO, IR

# STANDARDIZATION DOCUMENT IMPROVEMENT PROPOSAL

## INSTRUCTIONS

1. The preparing activity must complete blocks 1, 2, 3, and 8. In block 1, both the document number and revision letter should be given.
2. The submitter of this form must complete blocks 4, 5, 6, and 7.
3. The preparing activity must provide a reply within 30 days from receipt of the form.

NOTE: This form may not be used to request copies of documents, nor to request waivers, or clarification of requirements on current contracts. Comments submitted on this form do not constitute or imply authorization to waive any portion of the referenced document(s) or to amend contractual requirements.

I RECOMMEND A CHANGE:	1. DOCUMENT NUMBER	2. DOCUMENT DATE (YYMMDD)
	MIL-HDBK-28001	950630

### 3. DOCUMENT TITLE

Application of MIL-PRF-28001 Using Standard Generalized Markup Language (SGML)

4. NATURE OF CHANGE (Identify paragraph number and include proposed rewrite, if possible. Attach extra sheets as needed.)

### 5. REASON FOR RECOMMENDATION

### 6. SUBMITTER

a. NAME (Last, First, Middle Initial)

b. ORGANIZATION

c. ADDRESS (Include Zip Code)

d. TELEPHONE (Include Area Code)  
(1) Commercial

7. DATE SUBMITTED  
(YYMMDD)

(2) DSN  
(If applicable)

### 8. PREPARING ACTIVITY

a. NAME

Mr. Alan Peltzman

b. TELEPHONE (Include Area Code)

(1) Commercial  
(703) 735-3568

(2) DSN  
653-3568

c. ADDRESS (Include Zip Code)

ATTN: CALS Digital Standards Office  
DISA, CFS  
Code: JEQ/JEBEB  
10701 Parkridge Blvd  
Reston, VA 22091-4398

IF YOU DO NOT RECEIVE A REPLY WITHIN 45 DAYS, CONTACT:

Defense Quality and Standardization Office  
5203 Leesburg Pike, Suite 1403, Falls Church, VA 22041-3466  
Telephone (703) 756-2340 DSN 289-2340