

# **CDA Reference Manual: Volume 1**

Order Number: Volume 1: AA-PBD3A-TE

**June 1990**

This manual provides reference material for using the CDA Toolkit to create compound document applications, converters, and viewers.

**Revision/Update Information:** This revised manual supersedes the *CDA Reference Manual* Part I (Order No. AA-PABUA-TE) and Part II (Order No. AA-PABVA-TE).

**Software Version:** VMS Version 5.4  
ULTRIX Version 4.0

**digital equipment corporation  
maynard, massachusetts**

---

**First Printing, December 1988**  
**Revised, October 1989**  
**Revised, June 1990**

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.


Restricted Rights: Use, duplication, or disclosure by the U.S. Government is subject to restrictions as set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013.

© Digital Equipment Corporation 1990.

All Rights Reserved.  
Printed in U.S.A.

The postpaid Reader's Comments forms at the end of this document request your critical evaluation to assist in preparing future documentation.

The following are trademarks of Digital Equipment Corporation:

|            |             |  |
|------------|-------------|--|
| CDA        | LiveLink    | VAXcluster   |
| DDIF       | LN03        | VAX RMS  |
| DDIS       | MASSBUS     | VAXstation   |
| DEC        | PrintServer | VT   |
| DECnet     | Q-bus       | XUI  |
| DECUS      | ReGIS       |  |
| DECwindows | ULTRIX      |  |
| DIGITAL    | UNIBUS      |  |
| DTIF       | VAX         |  |

The following are third-party trademarks:

PostScript is a registered trademark of Adobe Systems, Inc.

Lotus 1-2-3 is a registered trademark of Lotus Development Corp.

ZK5428

This document was prepared using VAX DOCUMENT, Version 1.2

# Contents

---

|                      |      |
|----------------------|------|
| <b>Preface</b> ..... | xvii |
|----------------------|------|

---

|                  |  |      |
|------------------|--|------|
| <b>Chapter 1</b> | <b>Introduction</b>                                    |      |
| 1.1              | <b>CDA Item Data Types</b> .....                       | 1-1  |
| 1.2              | <b>CDA Default Values</b> .....                        | 1-4  |
| 1.3              | <b>DDIF Standard Aggregates</b> .....                  | 1-5  |
| 1.4              | <b>DDIF Tags</b> .....                                 | 1-7  |
| 1.5              | <b>DDIF Initial Values</b> .....                       | 1-9  |
| 1.6              | <b>DDIF Processing Options</b> .....                   | 1-13 |
| 1.6.1            | Inherit Attributes Processing Option .....             | 1-14 |
| 1.6.1.1          | Type References and Definitions .....                  | 1-14 |
| 1.6.1.2          | Style Guides .....                                     | 1-15 |
| 1.6.2            | Retain Definitions Processing Option .....             | 1-15 |
| 1.6.3            | Evaluate Content Processing Option .....               | 1-15 |
| 1.6.3.1          | Content References and Definitions .....               | 1-16 |
| 1.6.3.2          | Computed Segment Content and External References ..... | 1-16 |
| 1.6.4            | Discard Segments Processing Options .....              | 1-17 |
| 1.7              | <b>DTIF Document Aggregate Hierarchy</b> .....         | 1-17 |
| 1.8              | <b>CFE Aggregate Hierarchy</b> .....                   | 1-18 |
| 1.9              | <b>ESF Aggregate Hierarchy</b> .....                   | 1-19 |

---

|                  |   |     |
|------------------|---|-----|
| <b>Chapter 2</b> | <b>Bundled Converter Formats and Commands</b>       |     |
| 2.1              | <b>Base System Front Ends (Input Formats)</b> ..... | 2-1 |
| 2.1.1            | DDIF Front End .....                                | 2-1 |
| 2.1.1.1          | Data Mapping .....                                  | 2-1 |
| 2.1.1.2          | Conversion Restrictions .....                       | 2-1 |
| 2.1.1.3          | External File References .....                      | 2-1 |
| 2.1.1.4          | Document Syntax Errors .....                        | 2-1 |
| 2.1.2            | DTIF Front End .....                                | 2-2 |
| 2.1.2.1          | Data Mapping .....                                  | 2-2 |
| 2.1.2.2          | Conversion Restrictions .....                       | 2-2 |
| 2.1.2.3          | External File References .....                      | 2-2 |
| 2.1.2.4          | DTIF Syntax Errors .....                            | 2-2 |

|            |   |            |
|------------|---|------------|
| 2.1.3      | Text Front End                                | 2-2        |
| 2.1.3.1    | Data Mapping                                  | 2-2        |
| 2.1.3.2    | Conversion Restrictions                       | 2-3        |
| 2.1.3.3    | External File References                      | 2-3        |
| 2.1.3.4    | Document Syntax Errors                        | 2-3        |
| <b>2.2</b> | <b>Base System Back Ends (Output Formats)</b> | <b>2-3</b> |
| 2.2.1      | DDIF Back End                                 | 2-3        |
| 2.2.1.1    | Data Mapping                                  | 2-3        |
| 2.2.1.2    | Conversion Restrictions                       | 2-3        |
| 2.2.2      | DTIF Back End                                 | 2-3        |
| 2.2.2.1    | Data Mapping                                  | 2-3        |
| 2.2.2.2    | Conversion Restrictions                       | 2-4        |
| 2.2.2.3    | External File References                      | 2-4        |
| 2.2.3      | Text Back End                                 | 2-4        |
| 2.2.3.1    | Data Mapping                                  | 2-4        |
| 2.2.3.2    | Conversion Restrictions                       | 2-4        |
| 2.2.3.3    | Processing Options                            | 2-4        |
| 2.2.4      | PostScript Back End                           | 2-5        |
| 2.2.4.1    | Data Mapping                                  | 2-5        |
| 2.2.4.2    | Conversion Restrictions                       | 2-5        |
| 2.2.4.3    | Processing Options                            | 2-6        |
| 2.2.4.4    | Paper Size Processing Option                  | 2-6        |
| 2.2.4.5    | Paper Height Processing Option                | 2-7        |
| 2.2.4.6    | Paper Width Processing Option                 | 2-7        |
| 2.2.4.7    | Top Margin Processing Option                  | 2-7        |
| 2.2.4.8    | Bottom Margin Processing Option               | 2-7        |
| 2.2.4.9    | Left Margin Processing Option                 | 2-7        |
| 2.2.4.10   | Right Margin Processing Option                | 2-7        |
| 2.2.4.11   | Paper Orientation Processing Option           | 2-7        |
| 2.2.4.12   | Eight Bit Output Processing Option            | 2-8        |
| 2.2.4.13   | Output Buffer Size Processing Option          | 2-8        |
| 2.2.4.14   | Soft Directives Processing Option             | 2-8        |
| 2.2.4.15   | Word Wrap Processing Option                   | 2-8        |
| 2.2.4.16   | Page Wrap Processing Option                   | 2-8        |
| 2.2.4.17   | Layout Processing Option                      | 2-8        |
| 2.2.5      | Analysis Back End                             | 2-8        |
| 2.2.6      | Processing Options for Domain Conversion      | 2-9        |
|            | CONVERT/DOCUMENT                              | 2-11       |
|            | VIEW  | 2-13       |
|            | CDOC  | 2-16       |
|            | DXVDOC  | 2-18       |
|            | VDOC  | 2-20       |

---

## Chapter 3 Transferring CDA Documents

|     |   |     |
|-----|---|-----|
| 3.1 | Sending CDA Documents on a VMS System       | 3-1 |
| 3.2 | Receiving CDA Documents on a VMS System     | 3-2 |
| 3.3 | Sending CDA Documents on an ULTRIX System   | 3-2 |
| 3.4 | Receiving CDA Documents on an ULTRIX System | 3-3 |
| 3.5 | Copying CDA Documents on a VMS System       | 3-3 |
| 3.6 | Copying CDA Documents on an ULTRIX System   | 3-3 |

---

**Chapter 4 DDIF Structures**

|            |   |            |
|------------|---|------------|
| <b>4.1</b> | <b>DDIF Document Structure Overview</b> | <b>4-1</b> |
| <b>4.2</b> | <b>Generic Aggregate Items</b>          | <b>4-1</b> |
|            | DDIF\$_ARC                              | 4-2        |
|            | DDIF\$_BEZ                              | 4-5        |
|            | DDIF\$_CRF                              | 4-7        |
|            | DDIF\$_CTD                              | 4-8        |
|            | DDIF\$_CTS                              | 4-10       |
|            | DDIF\$_DDF                              | 4-12       |
|            | DDIF\$_DHD                              | 4-13       |
|            | DDIF\$_DSC                              | 4-16       |
|            | DDIF\$_ERF                              | 4-18       |
|            | DDIF\$_EXT                              | 4-21       |
|            | DDIF\$_FAS                              | 4-23       |
|            | DDIF\$_FTD                              | 4-25       |
|            | DDIF\$_GLA                              | 4-26       |
|            | DDIF\$_GLY                              | 4-28       |
|            | DDIF\$_GTX                              | 4-32       |
|            | DDIF\$_HRD                              | 4-34       |
|            | DDIF\$_HRV                              | 4-36       |
|            | DDIF\$_IDU                              | 4-38       |
|            | DDIF\$_IMG                              | 4-41       |
|            | DDIF\$_LG1                              | 4-42       |
|            | DDIF\$_LIN                              | 4-43       |
|            | DDIF\$_LL1                              | 4-45       |
|            | DDIF\$_LS1                              | 4-50       |
|            | DDIF\$_LSD                              | 4-52       |
|            | DDIF\$_LW1                              | 4-54       |
|            | DDIF\$_OCC                              | 4-57       |
|            | DDIF\$_PGD                              | 4-59       |
|            | DDIF\$_PGL                              | 4-61       |
|            | DDIF\$_PGS                              | 4-64       |
|            | DDIF\$_PHD                              | 4-66       |
|            | DDIF\$_PTD                              | 4-67       |
|            | DDIF\$_PTH                              | 4-70       |
|            | DDIF\$_PVT                              | 4-74       |
|            | DDIF\$_RCD                              | 4-76       |
|            | DDIF\$_RGB                              | 4-77       |
|            | DDIF\$_SEG                              | 4-78       |
|            | DDIF\$_SFT                              | 4-81       |
|            | DDIF\$_SFV                              | 4-83       |
|            | DDIF\$_SGA                              | 4-85       |
|            | GENERAL SEGMENT ATTRIBUTES              | 4-90       |
|            | COMPUTED CONTENT ATTRIBUTES             | 4-92       |
|            | COPIED AND REMOTE COMPUTED CONTENT      | 4-93       |
|            | VARIABLE COMPUTED CONTENT               | 4-94       |
|            | CROSS-REFERENCE COMPUTED CONTENT        | 4-95       |
|            | FUNCTION COMPUTED CONTENT               | 4-96       |
|            | STRUCTURE DESCRIPTION                   | 4-97       |
|            | LANGUAGE ATTRIBUTES                     | 4-98       |
|            | LEGEND ATTRIBUTES                       | 4-99       |
|            | MEASUREMENT ATTRIBUTES                  | 4-100      |
|            | ALTERNATE PRESENTATION ATTRIBUTE        | 4-101      |
|            | LAYOUT ATTRIBUTES                       | 4-102      |
|            | GALLEY-BASED LAYOUT                     | 4-104      |

|                                  |       |
|----------------------------------|-------|
| PATH-BASED LAYOUT                | 4-105 |
| POSITION-RELATIVE LAYOUT         | 4-109 |
| TEXT POSITION LAYOUT             | 4-111 |
| FONT DEFINITIONS                 | 4-112 |
| PATTERN DEFINITIONS              | 4-113 |
| PATH DEFINITIONS                 | 4-114 |
| LINE-STYLE DEFINITIONS           | 4-115 |
| CONTENT DEFINITIONS              | 4-116 |
| TYPE DEFINITIONS                 | 4-117 |
| TEXT ATTRIBUTES                  | 4-118 |
| TEXT MASK PATTERN                | 4-119 |
| TEXT FONT                        | 4-120 |
| TEXT RENDITION                   | 4-121 |
| TEXT SIZE                        | 4-123 |
| TEXT DIRECTION                   | 4-124 |
| TEXT CHARACTER DECIMAL ALIGNMENT | 4-125 |
| TEXT LEADER                      | 4-126 |
| TEXT KERNING                     | 4-127 |
| LINE ATTRIBUTES                  | 4-128 |
| MARKER ATTRIBUTES                | 4-132 |
| GALLEY ATTRIBUTES                | 4-133 |
| IMAGE ATTRIBUTES                 | 4-134 |
| IMAGE COMPONENT SPACE            | 4-137 |
| FRAME PARAMETERS                 | 4-139 |
| FRAME FLAGS                      | 4-140 |
| FRAME BOUNDING BOX               | 4-141 |
| FRAME OUTLINE                    | 4-143 |
| FRAME CLIPPING                   | 4-144 |
| FRAME POSITION                   | 4-145 |
| FIXED FRAME                      | 4-147 |
| INLINE FRAME                     | 4-148 |
| GALLEY FRAME                     | 4-149 |
| MARGIN FRAME                     | 4-150 |
| FRAME CONTENT TRANSFORMATION     | 4-152 |
| ITEM CHANGE LIST                 | 4-153 |
| DDIF\$_SGB                       | 4-154 |
| COUNTER VARIABLE VALUES          | 4-156 |
| COMPUTED VARIABLE VALUES         | 4-158 |
| LIST VARIABLE VALUES             | 4-159 |
| DDIF\$_TBS                       | 4-160 |
| DDIF\$_TRN                       | 4-162 |
| DDIF\$_TXT                       | 4-164 |
| DDIF\$_TYD                       | 4-165 |

---

## Chapter 5 DTIF Structures

|     |                                  |      |
|-----|----------------------------------|------|
| 5.1 | DTIF Document Structure Overview | 5-1  |
| 5.2 | Generic Aggregate Items          | 5-1  |
|     | DTIF\$_ARD                       | 5-2  |
|     | DTIF\$_CAT                       | 5-4  |
|     | DTIF\$_CCD                       | 5-10 |
|     | DTIF\$_CFT                       | 5-12 |
|     | DTIF\$_CLD                       | 5-13 |
|     | DTIF\$_CLR                       | 5-18 |
|     | DTIF\$_COR                       | 5-19 |

|            |      |
|------------|------|
| DTIF\$_DAT | 5-20 |
| DTIF\$_DSC | 5-22 |
| DTIF\$_DTF | 5-24 |
| DTIF\$_ERF | 5-25 |
| DTIF\$_EXT | 5-27 |
| DTIF\$_FMI | 5-29 |
| DTIF\$_HDR | 5-43 |
| DTIF\$_LPT | 5-46 |
| DTIF\$_NES | 5-51 |
| DTIF\$_NMR | 5-52 |
| DTIF\$_NVL | 5-53 |
| DTIF\$_RNG | 5-55 |
| DTIF\$_ROW | 5-57 |
| DTIF\$_RWR | 5-59 |
| DTIF\$_TBL | 5-60 |
| DTIF\$_TMD | 5-62 |
| DTIF\$_VTX | 5-65 |
| DTIF\$_WND | 5-66 |

---

**Chapter 6 CFE Structures**

|  |      |
|--|------|
| <b>6.1 CFE Generic Aggregate Items</b> | 6-1  |
| CFE\$_CCD                              | 6-2  |
| CFE\$_CFT                              | 6-4  |
| CFE\$_CLR                              | 6-5  |
| CFE\$_COR                              | 6-6  |
| CFE\$_DAT                              | 6-7  |
| CFE\$_EXL                              | 6-9  |
| CFE\$_EXP                              | 6-53 |
| CFE\$_FRF                              | 6-55 |
| CFE\$_NPM                              | 6-56 |
| CFE\$_PEX                              | 6-57 |
| CFE\$_PFE                              | 6-58 |
| CFE\$_RWR                              | 6-60 |
| CFE\$_SLL                              | 6-61 |
| CFE\$_STF                              | 6-62 |
| CFE\$_STP                              | 6-63 |
| CFE\$_TXC                              | 6-64 |
| CFE\$_VTX                              | 6-65 |

---

**Chapter 7 ESF Structures**

|  |      |
|--|------|
| <b>7.1 ESF Generic Aggregate Items</b> | 7-1  |
| ESF\$_DAT                              | 7-2  |
| ESF\$_EDS                              | 7-4  |
| ESF\$_EXT                              | 7-20 |
| ESF\$_NVL                              | 7-22 |
| ESF\$_RPT                              | 7-24 |
| ESF\$_TXS                              | 7-27 |

---

**Chapter 8 CDA Toolkit Routines**

|            |   |            |
|------------|---|------------|
| <b>8.1</b> | <b>Compile and Link Procedures for Applications</b> | <b>8-1</b> |
| 8.1.1      | VMS Link Procedure                                  | 8-2        |
| 8.1.2      | ULTRIX Link Procedure                               | 8-2        |
|            | AGGREGATE TYPE TO OBJECT ID                         | 8-3        |
|            | CLOSE FILE  | 8-6        |
|            | CLOSE STREAM  | 8-9        |
|            | CLOSE TEXT FILE                                     | 8-11       |
|            | CONVERT   | 8-13       |
|            | CONVERT AGGREGATE                                   | 8-24       |
|            | CONVERT DOCUMENT                                    | 8-28       |
|            | CONVERT POSITION                                    | 8-31       |
|            | COPY AGGREGATE                                      | 8-33       |
|            | CREATE AGGREGATE                                    | 8-36       |
|            | CREATE FILE   | 8-39       |
|            | CREATE ROOT AGGREGATE                               | 8-45       |
|            | CREATE STREAM                                       | 8-50       |
|            | CREATE TEXT FILE                                    | 8-54       |
|            | DELETE AGGREGATE                                    | 8-58       |
|            | DELETE ROOT AGGREGATE                               | 8-60       |
|            | ENTER SCOPE   | 8-62       |
|            | ERASE ITEM  | 8-76       |
|            | FIND DEFINITION                                     | 8-79       |
|            | FIND TRANSFORMATION                                 | 8-83       |
|            | FLUSH STREAM  | 8-86       |
|            | GET AGGREGATE                                       | 8-88       |
|            | GET ARRAY SIZE                                      | 8-94       |
|            | GET DOCUMENT  | 8-97       |
|            | GET EXTERNAL ENCODING                               | 8-100      |
|            | GET STREAM POSITION                                 | 8-103      |
|            | GET TEXT POSITION                                   | 8-106      |
|            | INSERT AGGREGATE                                    | 8-108      |
|            | LEAVE SCOPE   | 8-112      |
|            | LOCATE ITEM   | 8-115      |
|            | NEXT AGGREGATE                                      | 8-120      |
|            | OBJECT ID TO AGGREGATE TYPE                         | 8-123      |
|            | OPEN CONVERTER                                      | 8-126      |
|            | OPEN FILE   | 8-130      |
|            | OPEN STREAM   | 8-138      |
|            | OPEN TEXT FILE                                      | 8-141      |
|            | PRUNE AGGREGATE                                     | 8-145      |
|            | PRUNE POSITION                                      | 8-148      |
|            | PUT AGGREGATE                                       | 8-150      |
|            | PUT DOCUMENT  | 8-153      |
|            | READ TEXT FILE                                      | 8-156      |
|            | REMOVE AGGREGATE                                    | 8-158      |
|            | STORE ITEM  | 8-160      |
|            | WRITE TEXT FILE                                     | 8-168      |



|                   |   |       |
|-------------------|---|-------|
| <b>Chapter 9</b>  | <b>User-Defined Routines</b>                                  |       |
|                   | <i>ALLOCATE/DEALLOCATE</i> ROUTINES .....                     | 9-2   |
|                   | <i>FLUSH</i> ROUTINE .....                                    | 9-4   |
|                   | <i>GET</i> ROUTINE .....                                      | 9-6   |
|                   | <i>GET-POSITION</i> ROUTINE .....                             | 9-8   |
|                   | <i>PUT</i> ROUTINE .....                                      | 9-10  |
| <hr/>             |   |       |
| <b>Chapter 10</b> | <b>CDA Toolkit Example Program</b>                            |       |
| <hr/>             |   |       |
| <b>Chapter 11</b> | <b>CDA Converter Routines</b>                                 |       |
| 11.1              | <b>Compile and Link Procedures for Converter Images</b> ..... | 11-1  |
| 11.1.1            | VMS Compile and Link Procedure .....                          | 11-2  |
| 11.1.2            | ULTRIX Compile and Link Procedure .....                       | 11-2  |
|                   | <i>CLOSE</i> ENTRY POINT .....                                | 11-3  |
|                   | <i>GET-AGGREGATE</i> ENTRY POINT .....                        | 11-5  |
|                   | <i>GET-POSITION</i> ENTRY POINT .....                         | 11-8  |
|                   | <i>DOMAIN\$READ_FORMAT</i> ENTRY POINT .....                  | 11-10 |
|                   | <i>DOMAIN\$WRITE_FORMAT</i> ENTRY POINT .....                 | 11-15 |
| <hr/>             |   |       |
| <b>Chapter 12</b> | <b>Text Front End Source File</b>                             |       |
| <hr/>             |   |       |
| <b>Chapter 13</b> | <b>CDA Viewer Routines</b>                                    |       |
| 13.1              | <b>CDA Viewer Support of Adobe Font Metrics</b> .....         | 13-1  |
| 13.2              | <b>Compile and Link Procedures for Viewer Images</b> .....    | 13-2  |
| 13.2.1            | VMS Link Procedure .....                                      | 13-2  |
| 13.2.2            | ULTRIX Link Procedures .....                                  | 13-2  |
|                   | <i>CC DELETE PAGE</i> .....                                   | 13-4  |
|                   | <i>CC END</i> .....   | 13-6  |
|                   | <i>CC GET PAGE</i> .....                                      | 13-8  |
|                   | <i>CC INITIALIZE</i> .....                                    | 13-10 |
|                   | <i>BOTTOM DOCUMENT</i> .....                                  | 13-13 |
|                   | <i>CLOSE FILE</i> .....                                       | 13-15 |
|                   | <i>DOCUMENT INFO</i> .....                                    | 13-16 |
|                   | <i>GOTO PAGE</i> .....  | 13-18 |
|                   | <i>NEXT PAGE</i> .....  | 13-20 |
|                   | <i>PREVIOUS PAGE</i> .....                                    | 13-22 |
|                   | <i>REGISTER CLASS</i> .....                                   | 13-24 |
|                   | <i>TOP DOCUMENT</i> .....                                     | 13-25 |
|                   | <i>VIEWER</i> .....   | 13-27 |
|                   | <i>VIEWER CREATE</i> .....                                    | 13-31 |
|                   | <i>VIEWER FILE</i> .....                                      | 13-34 |

---

**Appendix A DDIF Fill Patterns**

---

**Appendix B DDIF Syntax Diagrams**

|            |                                       |            |
|------------|---------------------------------------|------------|
| <b>B.1</b> | <b>DDIS Built-In Data Types</b> ..... | <b>B-1</b> |
| <b>B.2</b> | <b>Built-In Operators</b> .....       | <b>B-3</b> |
| <b>B.3</b> | <b>DDIS Defined Types</b> .....       | <b>B-3</b> |
| <b>B.4</b> | <b>DDIF Syntax Diagrams</b> .....     | <b>B-5</b> |

---

**Appendix C DTIF Syntax Diagrams**

---

**Appendix D CFE Syntax Diagrams**

---

**Appendix E ESF Syntax Diagrams**

---

---

**Appendix F VMS Support for CDA in DECwindows**

|            |  |             |
|------------|--|-------------|
| <b>F.1</b> | <b>VMS Commands and Utilities</b> .....                            | <b>F-1</b>  |
| F.1.1      | Displaying RMS File Tags .....                                     | F-2         |
| F.1.1.1    | DIRECTORY/FULL .....   | F-2         |
| F.1.1.2    | ANALYZE/RMS_FILE .....   | F-2         |
| F.1.2      | Creating RMS File Tags .....                                       | F-3         |
| F.1.3      | Preserving RMS File Tags and DDIF Semantics .....                  | F-4         |
| F.1.3.1    | COPY Command .....   | F-4         |
| F.1.3.2    | VMS Mail Utility .....   | F-4         |
| F.1.4      | APPEND Command .....   | F-5         |
| <b>F.2</b> | <b>DDIF Support in a Heterogeneous Environment</b> .....           | <b>F-5</b>  |
| F.2.1      | EXCHANGE/NETWORK Command .....                                     | F-5         |
| F.2.2      | Using the COPY Command in a Heterogeneous Environment .....        | F-6         |
| F.2.3      | VMS Mail Utility in a Heterogeneous Environment .....              | F-6         |
| <b>F.3</b> | <b>VMS RMS Interface Changes</b> .....                             | <b>F-6</b>  |
| F.3.1      | Programming Interface for File Tagging .....                       | F-6         |
| F.3.2      | Accessing a Tagged File .....                                      | F-10        |
| F.3.2.1    | File Accesses That Do Not Sense Tags .....                         | F-11        |
| F.3.2.2    | File Accesses That Sense Tags .....                                | F-11        |
| F.3.3      | Preserving Tags .....  | F-13        |
| <b>F.4</b> | <b>Distributed File System Support for DDIF Tagged Files</b> ..... | <b>F-14</b> |
| <b>F.5</b> | <b>VMS RMS Errors</b> .....  | <b>F-14</b> |

---

## Appendix G CDA\$ Facility Messages

---

### Glossary of Terms

---

### Index

---

#### Examples

|      |  |       |
|------|--|-------|
| 1-1  | Analysis Output of DDIF Initial Segment Attributes . . . . . | 1-10  |
| 10-1 | Sample CDA Toolkit Program . . . . .                         | 10-1  |
| 10-2 | Analysis Output of DDIF File . . . . .                       | 10-33 |
| F-1  | Tagging a File . . . . .                                     | F-9   |
| F-2  | Accessing a Tagged File . . . . .                            | F-12  |

---

#### Figures

|      |   |       |
|------|---|-------|
| 1-1  | DDIF Document Aggregate Hierarchy . . . . .       | 1-7   |
| 1-2  | DTIF Document Aggregate Hierarchy . . . . .       | 1-18  |
| 1-3  | CFE Aggregate Hierarchy . . . . .                 | 1-19  |
| 1-4  | ESF Aggregate Hierarchy . . . . .                 | 1-20  |
| 4-1  | Character Orientation . . . . .                   | 4-108 |
| 8-1  | Example Document . . . . .                        | 8-92  |
| A-1  | CDA Fill Patterns . . . . .                       | A-6   |
| B-1  | Object Descriptor Syntax Diagram . . . . .        | B-4   |
| B-2  | Latin1 String Syntax Diagram . . . . .            | B-4   |
| B-3  | Text String Syntax Diagram . . . . .              | B-5   |
| B-4  | Character String Syntax Diagram . . . . .         | B-5   |
| B-5  | Application Private Data Syntax Diagram . . . . . | B-5   |
| B-6  | DDIF Document Syntax Diagram . . . . .            | B-5   |
| B-7  | Document Descriptor Syntax Diagram . . . . .      | B-6   |
| B-8  | Document Header Syntax Diagram . . . . .          | B-6   |
| B-9  | Document Root Segment . . . . .                   | B-6   |
| B-10 | Segment Primitive Syntax Diagram . . . . .        | B-7   |
| B-11 | Begin-Segment Syntax Diagram . . . . .            | B-7   |
| B-12 | Text Primitive Syntax Diagram . . . . .           | B-7   |
| B-13 | Text Attributes Syntax Diagram . . . . .          | B-8   |
| B-14 | Rendition Code Syntax Diagram . . . . .           | B-8   |
| B-15 | Leader Style Syntax Diagram . . . . .             | B-9   |
| B-16 | Text Layout Syntax Diagram . . . . .              | B-9   |
| B-17 | Text String Layout Syntax Diagram . . . . .       | B-10  |
| B-18 | Formatting Primitive Syntax Diagram . . . . .     | B-10  |
| B-19 | Value Directive Syntax Diagram . . . . .          | B-10  |
| B-20 | Directive Syntax Diagram . . . . .                | B-11  |
| B-21 | Escapement Directive Syntax Diagram . . . . .     | B-11  |
| B-22 | Variable Reset Syntax Diagram . . . . .           | B-11  |
| B-23 | Graphics Primitive Syntax Diagram . . . . .       | B-11  |
| B-24 | Polyline Syntax Diagram . . . . .                 | B-12  |

|      |   |      |
|------|---|------|
| B-25 | Cubic Bézier Syntax Diagram .....                     | B-12 |
| B-26 | Arc Syntax Diagram .....                              | B-12 |
| B-27 | Fill Area Set Syntax Diagram .....                    | B-13 |
| B-28 | Line Attributes Syntax Diagram .....                  | B-14 |
| B-29 | Line Style Number Syntax Diagram .....                | B-14 |
| B-30 | Line End Number Syntax Diagram .....                  | B-14 |
| B-31 | Line Join Syntax Diagram .....                        | B-14 |
| B-32 | Marker Attributes Syntax Diagram .....                | B-15 |
| B-33 | Marker Number Syntax Diagram .....                    | B-15 |
| B-34 | Image Primitive Syntax Diagram .....                  | B-15 |
| B-35 | Image Coding Attributes Syntax Diagram .....          | B-16 |
| B-36 | Image Attributes Syntax Diagram .....                 | B-16 |
| B-37 | Image Lookup Table Data Syntax Diagram .....          | B-17 |
| B-38 | Image Component Space Attributes Syntax Diagram ..... | B-17 |
| B-39 | Restricted Content Syntax Diagram .....               | B-18 |
| B-40 | Content Reference Primitive Syntax Diagram .....      | B-18 |
| B-41 | Content Reference Syntax Diagram .....                | B-18 |
| B-42 | Bounding Box Syntax Diagram .....                     | B-19 |
| B-43 | Color Syntax Diagram .....                            | B-19 |
| B-44 | Red/Green/Blue Syntax Diagram .....                   | B-19 |
| B-45 | Compute Definition Syntax Diagram .....               | B-20 |
| B-46 | Cross-Reference Syntax Diagram .....                  | B-20 |
| B-47 | Escapement Syntax Diagram .....                       | B-20 |
| B-48 | External Reference Syntax Diagram .....               | B-20 |
| B-49 | Font Definition Syntax Diagram .....                  | B-21 |
| B-50 | Format Syntax Diagram .....                           | B-21 |
| B-51 | Frame Parameters Syntax Diagram .....                 | B-21 |
| B-52 | Inline Frame Parameters Syntax Diagram .....          | B-22 |
| B-53 | Galley Frame Parameters Syntax Diagram .....          | B-22 |
| B-54 | Galley Vertical Position Syntax Diagram .....         | B-22 |
| B-55 | Margin Frame Parameters Syntax Diagram .....          | B-23 |
| B-56 | Margin Horizontal Position Syntax Diagram .....       | B-23 |
| B-57 | Function Link Syntax Diagram .....                    | B-23 |
| B-58 | External Reference Index Syntax Diagram .....         | B-24 |
| B-59 | Language Index Syntax Diagram .....                   | B-24 |
| B-60 | Content Definition Syntax Diagram .....               | B-24 |
| B-61 | Label Types Syntax Diagram .....                      | B-24 |
| B-62 | Label Syntax Diagram .....                            | B-25 |
| B-63 | ASCII String Syntax Diagram .....                     | B-25 |
| B-64 | Variable Label Syntax Diagram .....                   | B-25 |
| B-65 | Legend Units Syntax Diagram .....                     | B-25 |
| B-66 | Angle Syntax Diagram .....                            | B-25 |
| B-67 | AngleRef Syntax Diagram .....                         | B-26 |
| B-68 | Measurement Syntax Diagram .....                      | B-26 |
| B-69 | Position Syntax Diagram .....                         | B-26 |
| B-70 | Ratio Syntax Diagram .....                            | B-26 |
| B-71 | Right Angle Syntax Diagram .....                      | B-27 |
| B-72 | Size Syntax Diagram .....                             | B-27 |
| B-73 | X-Coordinate Syntax Diagram .....                     | B-28 |

|       |  |      |
|-------|--|------|
| B-74  | Y-Coordinate Syntax Diagram . . . . .            | B-28 |
| B-75  | Measurement Units Syntax Diagram . . . . .       | B-28 |
| B-76  | Named Value Syntax Diagram . . . . .             | B-28 |
| B-77  | Value Data Syntax Diagram . . . . .              | B-29 |
| B-78  | Named Value List Syntax Diagram . . . . .        | B-29 |
| B-79  | Font Number Syntax Diagram . . . . .             | B-29 |
| B-80  | Marker Number Syntax Diagram . . . . .           | B-29 |
| B-81  | Path Number Syntax Diagram . . . . .             | B-30 |
| B-82  | Pattern Number Syntax Diagram . . . . .          | B-30 |
| B-83  | Path Definition Syntax Diagram . . . . .         | B-30 |
| B-84  | Composite Path Syntax Diagram . . . . .          | B-30 |
| B-85  | Arc Path Syntax Diagram . . . . .                | B-31 |
| B-86  | Cubic Bézier Path Syntax Diagram . . . . .       | B-31 |
| B-87  | Line Definition Syntax Diagram . . . . .         | B-31 |
| B-88  | Polyline Path Syntax Diagram . . . . .           | B-32 |
| B-89  | Pattern Definition Syntax Diagram . . . . .      | B-32 |
| B-90  | Standard Pattern Syntax Diagram . . . . .        | B-32 |
| B-91  | Reference Syntax Diagram . . . . .               | B-33 |
| B-92  | Segment Attributes Syntax Diagram . . . . .      | B-33 |
| B-93  | Segment Type Definition Syntax Diagram . . . . . | B-34 |
| B-94  | Structure Definition Syntax Diagram . . . . .    | B-34 |
| B-95  | Occurrence Definition Syntax Diagram . . . . .   | B-34 |
| B-96  | Structure Element Syntax Diagram . . . . .       | B-35 |
| B-97  | Tag Syntax Diagram . . . . .                     | B-35 |
| B-98  | Category Tag Syntax Diagram . . . . .            | B-35 |
| B-99  | Conformance Tag Syntax Diagram . . . . .         | B-35 |
| B-100 | Named Value Tag Syntax Diagram . . . . .         | B-35 |
| B-101 | Segment Tag Syntax Diagram . . . . .             | B-36 |
| B-102 | Storage System Tag Syntax Diagram . . . . .      | B-36 |
| B-103 | Stream Tag Syntax Diagram . . . . .              | B-36 |
| B-104 | Transformation Syntax Diagram . . . . .          | B-36 |
| B-105 | Variable Binding Syntax Diagram . . . . .        | B-37 |
| B-106 | Counter Definition Syntax Diagram . . . . .      | B-37 |
| B-107 | Layout Object Type Syntax Diagram . . . . .      | B-37 |
| B-108 | Expression Syntax Diagram . . . . .              | B-38 |
| B-109 | Counter Style Syntax Diagram . . . . .           | B-38 |
| B-110 | String Expression Syntax Diagram . . . . .       | B-38 |
| B-111 | Record List Syntax Diagram . . . . .             | B-39 |
| B-112 | Record Definition Syntax Diagram . . . . .       | B-39 |
| B-113 | Generic Layout Syntax Diagram . . . . .          | B-39 |
| B-114 | Page Description Syntax Diagram . . . . .        | B-39 |
| B-115 | Page Set Syntax Diagram . . . . .                | B-40 |
| B-116 | Page Layout Syntax Diagram . . . . .             | B-40 |
| B-117 | Layout Primitive Syntax Diagram . . . . .        | B-40 |
| B-118 | Layout Galley Syntax Diagram . . . . .           | B-41 |
| B-119 | Galley Attributes Syntax Diagram . . . . .       | B-41 |
| B-120 | Specific Layout Syntax Diagram . . . . .         | B-41 |
| B-121 | Wrap Attributes Syntax Diagram . . . . .         | B-42 |
| B-122 | Layout Attributes Syntax Diagram . . . . .       | B-42 |

|       |  |      |
|-------|--|------|
| B-123 | Break Criteria Syntax Diagram . . . . .              | B-42 |
| B-124 | General Measure Syntax Diagram . . . . .             | B-43 |
| B-125 | General Size Syntax Diagram . . . . .                | B-43 |
| B-126 | Tab Stop List Syntax Diagram . . . . .               | B-43 |
| B-127 | Tab Stop Syntax Diagram . . . . .                    | B-43 |
| B-128 | Generalized Time Diagram . . . . .                   | B-44 |
| C-1   | DTIF Document Syntax Diagram . . . . .               | C-1  |
| C-2   | Document Descriptor Syntax Diagram . . . . .         | C-1  |
| C-3   | Document Header Syntax Diagram . . . . .             | C-2  |
| C-4   | External Reference Syntax Diagram . . . . .          | C-2  |
| C-5   | Storage System Tag Syntax Diagram . . . . .          | C-3  |
| C-6   | External References Index Syntax Diagram . . . . .   | C-3  |
| C-7   | Language Preference Table Syntax Diagram . . . . .   | C-3  |
| C-8   | Named Edit String Syntax Diagram . . . . .           | C-4  |
| C-9   | Table Definition Syntax Diagram . . . . .            | C-4  |
| C-10  | Table Metadata Syntax Diagram . . . . .              | C-5  |
| C-11  | Table Window Syntax Diagram . . . . .                | C-5  |
| C-12  | Table Rows Syntax Diagram . . . . .                  | C-6  |
| C-13  | Cell Data Syntax Diagram . . . . .                   | C-6  |
| C-14  | Cell Value Syntax Diagram . . . . .                  | C-7  |
| C-15  | Varying Text Syntax Diagram . . . . .                | C-7  |
| C-16  | Array Definition Syntax Diagram . . . . .            | C-8  |
| C-17  | Complex Float Syntax Diagram . . . . .               | C-8  |
| C-18  | Column Attributes Syntax Diagram . . . . .           | C-9  |
| C-19  | Data Type Syntax Diagram . . . . .                   | C-10 |
| C-20  | Format Info List Syntax Diagram . . . . .            | C-10 |
| C-21  | Language Preference Index Syntax Diagram . . . . .   | C-11 |
| C-22  | Format Type Syntax Diagram . . . . .                 | C-11 |
| C-23  | Edit String Index Syntax Diagram . . . . .           | C-12 |
| C-24  | Numeric Format Type Syntax Diagram . . . . .         | C-12 |
| C-25  | Numeric Format Precision Syntax Diagram . . . . .    | C-13 |
| C-26  | Predefined Text Types Syntax Diagram . . . . .       | C-13 |
| C-27  | Predefined Date Types Syntax Diagram . . . . .       | C-13 |
| C-28  | Format Flags Syntax Diagram . . . . .                | C-14 |
| C-29  | Date Time Syntax Diagram . . . . .                   | C-14 |
| C-30  | Application Private Syntax Diagram . . . . .         | C-14 |
| C-31  | Named Value List Syntax Diagram . . . . .            | C-15 |
| C-32  | Value Data Syntax Diagram . . . . .                  | C-15 |
| C-33  | ASCII String Syntax Diagram . . . . .                | C-15 |
| C-34  | Column Number Syntax Diagram . . . . .               | C-16 |
| C-35  | Row Number Syntax Diagram . . . . .                  | C-16 |
| C-36  | Cell Coordinates Syntax Diagram . . . . .            | C-16 |
| C-37  | Range Definition Syntax Diagram . . . . .            | C-17 |
| C-38  | Range Syntax Diagram . . . . .                       | C-17 |
| C-39  | Cell Range Syntax Diagram . . . . .                  | C-18 |
| C-40  | Row Range Syntax Diagram . . . . .                   | C-18 |
| C-41  | Column Range Syntax Diagram . . . . .                | C-18 |
| C-42  | Named Range Syntax Diagram . . . . .                 | C-18 |
| D-1   | Private Function Expression Syntax Diagram . . . . . | D-1  |

|      |  |     |
|------|--|-----|
| D-2  | Storage System Tag Syntax Diagram . . . . .              | D-2 |
| D-3  | Named Parameter Syntax Diagram . . . . .                 | D-2 |
| D-4  | Expression Syntax Diagram . . . . .                      | D-2 |
| D-5  | Expression List Syntax Diagram . . . . .                 | D-3 |
| D-6  | Text Syntax Diagram . . . . .                            | D-8 |
| D-7  | Varying Text Syntax Diagram . . . . .                    | D-8 |
| D-8  | Selector List Syntax Diagram . . . . .                   | D-8 |
| D-9  | Decimal String Syntax Diagram . . . . .                  | D-9 |
| D-10 | Edit String Syntax Diagram . . . . .                     | D-9 |
| D-11 | Parenthesized Expressions Syntax Diagram . . . . .       | D-9 |
| D-12 | Field Reference Syntax Diagram . . . . .                 | D-9 |
| E-1  | Edit String Syntax Diagram . . . . .                     | E-1 |
| E-2  | Edit String Buffer Syntax Diagram . . . . .              | E-1 |
| E-3  | Single Syntax Diagram . . . . .                          | E-2 |
| E-4  | Repeat Syntax Diagram . . . . .                          | E-3 |
| E-5  | Application Private Edit String Syntax Diagram . . . . . | E-3 |

---

**Tables**

|      |   |       |
|------|---|-------|
| 1-1  | CDA Item Data Types . . . . .   | 1-1   |
| 1-2  | DDIF Standard Aggregates . . . . .  | 1-5   |
| 1-3  | DDIF Tags . . . . .   | 1-8   |
| 4-1  | Generic Aggregate Items . . . . .   | 4-1   |
| 4-2  | Character Set Identifiers . . . . .   | 4-32  |
| 4-3  | Normal Horizontal Alignment . . . . .   | 4-107 |
| 4-4  | Normal Vertical Alignment . . . . .   | 4-108 |
| 4-5  | Line Style . . . . .  | 4-115 |
| 4-6  | Line Style . . . . .  | 4-128 |
| 5-1  | DTIF Generic Aggregate Items . . . . .  | 5-1   |
| 6-1  | CFE Generic Aggregate Items . . . . .   | 6-1   |
| 6-2  | Valid Arithmetic Expression Values for CFE\$_EXL_EXPR_C . . . . .             | 6-10  |
| 6-3  | Valid Binary Expression Values for CFE\$_EXL_EXPR_C . . . . .                 | 6-11  |
| 6-4  | Valid Boolean and Relational Expression Values for CFE\$_EXL_EXPR_C . . . . . | 6-12  |
| 6-5  | Valid Cell-Related Expression Values for CFE\$_EXL_EXPR_C . . . . .           | 6-14  |
| 6-6  | Valid Choose and Lookup Expression Values for CFE\$_EXL_EXPR_C . . . . .      | 6-17  |
| 6-7  | Valid Conversion Expression Values for CFE\$_EXL_EXPR_C . . . . .             | 6-19  |
| 6-8  | Valid Date and Time Expression Values for CFE\$_EXL_EXPR_C . . . . .          | 6-21  |
| 6-9  | Valid Financial Expression Values for CFE\$_EXL_EXPR_C . . . . .              | 6-27  |
| 6-10 | Valid Identification Expression Values for CFE\$_EXL_EXPR_C . . . . .         | 6-39  |
| 6-11 | Valid Literal Values for CFE\$_EXL_EXPR_C . . . . .                           | 6-41  |
| 6-12 | Valid Miscellaneous Expression Values for CFE\$_EXL_EXPR_C . . . . .          | 6-42  |
| 6-13 | Valid Series Expression Values for CFE\$_EXL_EXPR_C . . . . .                 | 6-43  |
| 6-14 | Valid Statistical Expression Values for CFE\$_EXL_EXPR_C . . . . .            | 6-44  |
| 6-15 | Valid String Expression Values for CFE\$_EXL_EXPR_C . . . . .                 | 6-46  |
| 6-16 | Valid Transcendental Expression Values for CFE\$_EXL_EXPR_C . . . . .         | 6-49  |
| 6-17 | Valid Trigonometric Expression Values for CFE\$_EXL_EXPR_C . . . . .          | 6-50  |
| 6-18 | Valid Variable Values for CFE\$_EXL_EXPR_C . . . . .                          | 6-51  |
| 7-1  | ESF Generic Aggregate Items . . . . .   | 7-1   |
| 7-2  | Valid Values for ESF\$_EDS_EDIT_STRING_C . . . . .                            | 7-5   |

|      |  |      |
|------|--|------|
| 7-3  | Valid Values for ESF\$_RPT_SEQ_C ..... | 7-24 |
| 11-1 | Top-Level Aggregate Types .....        | 11-7 |
| A-1  | DDIF Fill Patterns .....               | A-1  |
| B-1  | DDIS Built-In Primitives .....         | B-1  |
| B-2  | DDIS Built-In Constructors .....       | B-2  |
| B-3  | DDIS Built-In Operators .....          | B-3  |
| B-4  | DDIS Defined Types .....               | B-4  |
| F-1  | Tag Support Item Codes .....           | F-7  |



## Preface

---

This manual provides reference material for using the CDA (Compound Document Architecture) Toolkit to create compound document applications, converters, and viewer widgets. Information in this manual includes reference material for using the DDIF (DIGITAL Document Interchange Format) and DTIF (DIGITAL Table Interchange Format) aggregates that are processed by the CDA Toolkit routines.

The CDA Toolkit is a collection of data structures and routines that support the creation of CDA applications. The CDA Converter architecture is used to convert files of a specified input format to a specified output format. The CDA Viewer is used to display CDA-encoded files on a workstation display or character cell terminal.

CDA is supported in both the VMS and the ULTRIX environments. The information contained in this manual is appropriate for both systems. Any differences between the two implementations are called out in the text of this manual.

All of the following products support CDA-encoded files. If you intend to manipulate only DDIF files and do not have an interest in the particulars of the file format, you can use any one of these products to manipulate a CDA-encoded file.

|            |                 |                                |
|------------|-----------------|--------------------------------|
| DECpaint   | PrintScreen     | CardFiler                      |
| DEC GKS    | DEC GKS-3D      | PHIGS                          |
| CDA Viewer | DECwindows MAIL | DECImage Applications Services |
| Converters | MAIL            | DECwrite                       |
| DECchart   | DECdecision     | DEC Test Manager               |

---

## Intended Audience

This manual is intended for system and application programmers who already have been introduced to CDA and who are ready to use the CDA Toolkit to write compound document applications, converters, or viewers. Some knowledge of the tasks and terminology associated with document typesetting is helpful.

---

## Document Structure

This manual consists of 13 chapters, several appendixes, and a glossary, as follows:

- Chapter 1, Introduction provides an introduction to the reference material describing the aggregates and routines contained in the CDA Toolkit.

- Chapter 2, Bundled Converter Formats and Commands describes the VMS and ULTRIX converter formats and commands used to convert and to view CDA documents.
- Chapter 3, Transferring CDA Documents describes how to mail and to copy CDA documents on VMS and ULTRIX systems.
- Chapter 4, DDIF Structures describes each of the DDIF aggregate structures.
- Chapter 5, DTIF Structures describes each of the DTIF aggregate structures.
- Chapter 6, CFE Structures describes each of the CFE aggregate structures.
- Chapter 7, ESF Structures describes each of the ESF aggregate structures.
- Chapter 8, CDA Toolkit Routines describes each of the routines contained in the CDA Toolkit. The routines are documented in alphabetical order. Each routine description specifies the calling format, the encoding of the parameters, a detailed description of the function of the routine, and what condition values the routine can return.
- Chapter 9, User-Defined Routines describes the user-defined routines used to write CDA-conforming applications and front and back ends.
- Chapter 10, CDA Toolkit Example Program contains an example program that uses the CDA Toolkit to create a DDIF file, and an illustration of the file created by the example program.
- Chapter 11, CDA Converter Routines describes each of the converter routines that must be created in order to write a CDA-conforming front or back end.
- Chapter 12, Text Front End Source File contains the source code for the Text front end to be used as an example for those wanting to develop their own front or back ends.
- Chapter 13, CDA Viewer Routines describes each of the viewer routines used to create a character-cell or DECwindows viewer application.
- Appendix A, DDIF Fill Patterns illustrates the CDA-defined fill patterns.
- Appendix B, DDIF Syntax Diagrams contains a brief overview of DDIS (DIGITAL Data Interchange Syntax) followed by the syntax diagrams for the various constructs supported by the DDIF architecture.
- Appendix C, DTIF Syntax Diagrams contains the syntax diagrams for the various constructs supported by DTIF.
- Appendix D, CFE Syntax Diagrams contains the syntax diagrams for the various constructs supported by CFE.
- Appendix E, ESF Syntax Diagrams contains the syntax diagrams for the various constructs supported by ESF.
- Appendix F, VMS Support for CDA in DECwindows discusses the support provided by VMS for the CDA Toolkit and the tagging of DDIF-encoded files.
- Appendix G, CDA\$ Facility Messages lists and describes the CDA\$\_ facility messages generated by the CDA Toolkit.
- Glossary, Glossary of Terms defines the terminology associated with the CDA Toolkit and CDA Converter Architecture.

---

## Associated Documents

CDA is supported by a variety of DIGITAL products. Descriptions of the support provided by each product are contained in that product's documentation. For example, GKS support for CDA is described in the GKS documentation set, and so on.

The complete CDA documentation set includes two tutorials and a reference manual:

- *Introduction to the CDA Services*
- *Guide to Creating Compound Documents with the CDA Toolkit*
- *CDA Reference Manual*

The CDA documentation set is a separately orderable subkit available for purchase with the VMS and ULTRIX operating system documentation. Each manual in the CDA documentation set is also available for separate purchase.

The CDA Converter Library end-user documentation set describes additional document, graphics, image, and table data file formats that are supported by the CDA Converter architecture, but that are not bundled with the VMS or ULTRIX operating system. The following two manuals describe the additional interchange formats:

- *Guide to the CDA Converter Library*
- *Getting Started with the CDA Converter Library*

---

## Conventions

The following conventions are used in this manual:

|  |   |
|--|---|
| Ctrl/x   | A sequence such as Ctrl/x indicates that you must hold down the key labeled Ctrl while you press another key or a pointing device button.   |
| PF1 x  | A sequence such as PF1 x indicates that you must first press and release the key labeled PF1, then press and release another key or a pointing device button.   |
| <span style="border: 1px solid black; padding: 2px;">Return</span> | A key name is shown enclosed to indicate that you press a key on the keyboard.  |
| ...  | In examples, a horizontal ellipsis indicates one of the following possibilities: <ul style="list-style-type: none"><li>• Additional optional arguments in a statement have been omitted.</li><li>• The preceding item or items can be repeated one or more times.</li><li>• Additional parameters, values, or other information can be entered.</li></ul> |
| .<br>. .<br>. . .  | A vertical ellipsis indicates the omission of items from a code example or command format; the items are omitted because they are not important to the topic being discussed.   |

|                      |   |
|----------------------|---|
| ( )                  | In format descriptions, parentheses indicate that, if you choose more than one option, you must enclose the choices in parentheses.   |
| [ ]                  | In format descriptions, brackets indicate that whatever is enclosed is optional; you can select none, one, or all of the choices.   |
| { }                  | In format descriptions, braces surround a required choice of options; you must choose one of the options listed.  |
| red ink              | Red ink indicates information that you must enter from the keyboard or a screen object that you must choose or click on. For online manuals, user input is specified in <b>bold</b> .   |
| <i>italic text</i>   | <i>Italic text</i> represents the introduction of a new term or the name of an argument, an attribute, or a reason.   |
| <i>italic text</i>   | <i>Italic text</i> represents user-written routines (for example, <i>get-aggregate</i> ).   |
| <b>boldface text</b> | <b>Boldface text</b> represents information that can vary in system messages (for example, Internal error <b>number</b> ).  |
| UPPERCASE TEXT       | Uppercase letters indicate that you must enter a command (for example, enter OPEN/READ).  |
| UPPERCASE TEXT       | Uppercase letters indicate the name of a CDA Toolkit routine, the name of a file, the name of a file protection code, or the abbreviation for a system privilege.   |
| lowercase text       | Lowercase letters indicate the names of the CDA Toolkit VAX format routines and values that are portable to ULTRIX systems. Value names that appear in lowercase must be coded as such in order to be portable to ULTRIX systems. |
| -                    | Hyphens in coding examples indicate that additional arguments to the request are provided on the line that follows.   |
| numbers              | Unless otherwise noted, all numbers in the text are assumed to be decimal. Nondecimal radices—binary, octal, or hexadecimal—are explicitly indicated in the coding examples.  |

This chapter provides an introduction to reference material describing the aggregates and routines contained in the CDA Toolkit. Specifically, this chapter provides reference material for the following information:

- CDA item data types
- CDA default values
- DDIF aggregates: root, primary, and secondary
- DDIF tags
- DDIF initial values
- DDIF processing options
- DTIF aggregates
- CFE aggregates
- ESF aggregates

---

## 1.1 CDA Item Data Types

Table 1-1 lists and describes the CDA item data types.

**Table 1-1: CDA Item Data Types**

| Data Type            | Definition   |
|----------------------|--|
| AngleRef enumeration | <p>An enumeration that specifies the data type of an item of DDIF type AngleRef, which is encoded as a floating point or string. A DDIF AngleRef type can either specify a constant angle value, measured in degrees, or it can specify an angle value derived from the value of the referenced variable. Valid values are as follows:</p> <ul style="list-style-type: none"><li>• DDIF\$K_ANGLE_CONSTANT</li><li>• DDIF\$K_ANGLE_VARIABLE</li></ul> <p>If the value DDIF\$K_ANGLE_CONSTANT is chosen, the item following the AngleRef type is encoded as a single-precision floating point. If the value DDIF\$K_ANGLE_VARIABLE is chosen, the item following the AngleRef type is encoded as a string.</p> |

(continued on next page)

**Table 1-1 (Cont.): CDA Item Data Types**

| Data Type              | Definition  |
|------------------------|---|
| Array of <b>type</b>   | An array of items of data type <b>type</b> . To store or retrieve this array, you must specify one item value at a time, along with an <b>aggregate index</b> parameter. The aggregate index value starts at 0; you must increment this aggregate index to index into the individual array items.   |
| Bit string             | A string of bits. The length of the item buffer is expressed in bits rather than bytes.   |
| Boolean                | A byte representing a Boolean value. The length of the item buffer is always 1. If the low bit of the value is set, the value is true. If the low bit is clear, the value is false.   |
| Byte                   | An 8-bit encoded structure. The length of the item buffer is always 1.  |
| Character string       | A string of bytes in a particular character set (for example, ISO Latin1). The <b>add-info</b> parameter represents the character set designator. The symbolic constants for the character set designators are defined in module <code>cda\$def.h</code> on VMS systems and in module <code>cda_def.h</code> on ULTRIX systems.   |
| DDIF\$_xyz             | A longword aggregate handle to an aggregate of the specified type. The length of the item buffer is always 4.   |
| DTIF\$_xyz             | A longword aggregate handle to an aggregate of the specified type. The length of the item buffer is always 4.   |
| DDIS encoding          | A tag, length, value (TLV) encoding based on the ISO OSI ASN.1 standard.  |
| Document               | A longword aggregate handle that is the root aggregate handle of a subdocument.   |
| Enumeration            | A longword integer. The allowed values of the integer are defined by symbolic constants. The length of the item buffer is always 4.   |
| Expression enumeration | <p>An enumeration that specifies the data type of an item of DDIF type Expression, which is encoded as an integer or string. Either DDIF Expression type can specify a constant expression value, or it can specify an expression value derived from the value of the referenced variable. Valid values are as follows:</p> <ul style="list-style-type: none"> <li>• DDIF\$K_VALUE_CONSTANT</li> <li>• DDIF\$K_VALUE_VARIABLE</li> </ul> <p>If the value DDIF\$K_VALUE_CONSTANT is chosen, the item following the measurement enumeration type is encoded as an integer. If the value DDIF\$K_VALUE_VARIABLE is chosen, the item following the measurement enumeration type is encoded as a string.</p> |

(continued on next page)

**Table 1–1 (Cont.): CDA Item Data Types**

| Data Type               | Definition   |
|-------------------------|--|
| General floating point  | <p>A floating-point value. The type of floating-point format must be specified by the <b>add-info</b> parameter; <b>add-info</b> can take any one of the following values:</p> <ul style="list-style-type: none"> <li>• CDA\$K_F_FLOAT</li> <li>• CDA\$K_D_FLOAT</li> <li>• CDA\$K_G_FLOAT</li> <li>• CDA\$K_H_FLOAT</li> <li>• CDA\$K_IEEE_754_S_FLOAT</li> <li>• CDA\$K_IEEE_754_D_FLOAT</li> <li>• CDA\$K_NATIVE_S_FLOAT (VAX: F-float; non-VAX: IEEE 754 single)</li> <li>• CDA\$K_NATIVE_D_FLOAT (VAX: G-float; non-VAX: IEEE 754 double)</li> </ul> <p>Upon storing a floating-point value, the CDA Toolkit transforms the value to a generic DDIS quadruple-precision floating-point value with a precision of 33 decimal digits. During a call to the LOCATE ITEM routine, an application specifies whether the DDIS-encoded floating-point data should be converted to VAX or IEEE format by the CDA Toolkit.</p> |
| Integer                 | <p>A longword integer. The length of the item buffer is always 4.</p>  |
| Item change list        | <p>A vector of longwords in which each longword contains the item code of an item in a segment attribute aggregate (DDIF\$_SGA). The length of the vector is specified in bytes.</p>   |
| Longword                | <p>A longword bit-encoded structure. The bits are interpreted according to a defined structure. The length of the item buffer is always 4 bytes (octets).</p>  |
| Measurement enumeration | <p>An enumeration that specifies the data type of an item of DDIF type Measurement, which is encoded as an integer or string. Either DDIF Measurement type can specify a specific number of measurement units, or it can specify the number of measurement units given by the value of the referenced variable. Valid values are as follows:</p> <ul style="list-style-type: none"> <li>• DDIF\$K_VALUE_CONSTANT</li> <li>• DDIF\$K_VALUE_VARIABLE</li> </ul> <p>If the value DDIF\$K_VALUE_CONSTANT is chosen, the item following the measurement enumeration type is encoded as an integer. If the value DDIF\$K_VALUE_VARIABLE is chosen, the item following the measurement enumeration type is encoded as a string.</p>   |
| Object identifier       | <p>Two or more longwords that specify the value of the DDIS type OBJECT IDENTIFIER. (DDIS is the DIGITAL Data Interchange Syntax.) Each longword specifies a single component of the object handle. The length of the item buffer is expressed in bytes.</p>   |

(continued on next page)

**Table 1-1 (Cont.): CDA Item Data Types**

| <b>Data Type</b>                | <b>Definition</b>  |
|---------------------------------|--|
| Scaled integer                  | A variable integer accompanied by a scaling factor. The <b>add-info</b> parameter represents an integer scaling factor.  |
| Sequence                        | A linked list of aggregates of the specified type on the same level in the document hierarchy. The value of the aggregate item that is encoded as a sequence is actually the handle of the first aggregate in the sequence.  |
| Sequence of <b>agg-type</b>     | Indicates that the value of an item is zero or more aggregates of type <b>agg-type</b> in sequence.  |
| Single-precision floating-point | A VAX F floating-point value on VAX systems; an IEEE Standard 754 single-precision floating-point value on non-VAX systems. The length of the buffer is always 4. Upon writing a floating-point value to a DDIS stream, the CDA Toolkit transforms the value to a generic DDIS floating-point value. When reading a single-precision floating-point value from a DDIS stream, the DDIS-encoded floating-point data is converted to the native (VAX or IEEE) format by the CDA Toolkit. |
| String                          | A string of bytes. The length of the string is specified in bytes.   |
| String with <b>add-info</b>     | A string of bytes, with additional information, that represents the character set. The value of <b>add-info</b> is the DDIF type Tag, where standard tag values have been defined. As a service to the application, the CDA Toolkit provides encoding and decoding services for the standard tags.   |
| TextUnit                        | A string of bytes that represent the value of the DTIF type TextUnit.  |
| Variable                        | The data type of the item is determined by a reference to the value of the preceding enumeration item. A variable type is always preceded by an enumeration item that specifies the data type of the variable item.  |
| Variable integer                | A binary encoded integer. The length of the item buffer is specified in bytes.   |
| Word                            | A 16-bit encoded structure. The length of the item buffer is always 2.   |

## 1.2 CDA Default Values

The CDA standard specifications for DDIF and DTIF define default values for some items. These default values are defined where it is determined that there are common values that will be used. The advantage to having default values is the ability to reduce the size of documents.

Document creators that wish to use a default value can simply leave the item empty. If they would like to use a value other than the default value, they simply store the value that they wish. (DEFAULT is a special case of OPTIONAL.)

When reading DDIF and DTIF files using the CDA Toolkit, the CDA Toolkit automatically fills in the default values where they are empty in the encoding. No processing option is needed. This happens automatically when reading from a file or stream and when requesting aggregates in a back end.



## 1.3 DDIF Standard Aggregates

Table 1–2 lists and defines the document root, primary, and secondary DDIF aggregate types. Primary aggregates, also known as top-level aggregates, are returned by the GET AGGREGATE or CONVERT AGGREGATE routine. All the primary aggregates, except DDIF\$\_DHD, DDIF\$\_DSC, and DDIF\$\_EOS, are also content aggregates, which are contained by a segment or content definition. Secondary aggregates are children of the primary aggregates.

**Table 1–2: DDIF Standard Aggregates**

| <b>Aggregate</b>            | <b>Definition</b>         |
|-----------------------------|---------------------------|
| <b>Root Aggregates</b>      |                           |
| DDIF\$_DDF                  | DDIF root aggregate       |
| <b>Primary Aggregates</b>   |                           |
| DDIF\$_ARC                  | Arc content               |
| DDIF\$_BEZ                  | Bézier curve content      |
| DDIF\$_CRF                  | Content reference         |
| DDIF\$_DHD                  | DDIF header aggregate     |
| DDIF\$_DSC                  | DDIF descriptor aggregate |
| DDIF\$_EOS <sup>1</sup>     | End of segment            |
| DDIF\$_EXT                  | External content          |
| DDIF\$_FAS                  | Fill area set content     |
| DDIF\$_GLY                  | Layout galley             |
| DDIF\$_GTXT                 | General text content      |
| DDIF\$_HRD                  | Hard directive            |
| DDIF\$_HRV                  | Hard value directive      |
| DDIF\$_IMG                  | Image content             |
| DDIF\$_LIN                  | Polyline content          |
| DDIF\$_PVT                  | Private content           |
| DDIF\$_SEG                  | Segment                   |
| DDIF\$_SFT                  | Soft directive            |
| DDIF\$_SFV                  | Soft value directive      |
| DDIF\$_TXT                  | Latin1 text content       |
| <b>Secondary Aggregates</b> |                           |
| DDIF\$_CTS                  | Counter style             |
| DDIF\$_CTD                  | Content definition        |
| DDIF\$_ERF                  | External reference        |
| DDIF\$_FTD                  | Font definition           |
| DDIF\$_GLA                  | Galley attributes         |

<sup>1</sup>The DDIF\$\_EOS aggregate does not return an aggregate handle, but only the aggregate type of DDIF\$\_EOS. It is returned for the purpose of maintaining the hierarchical structure of the document.

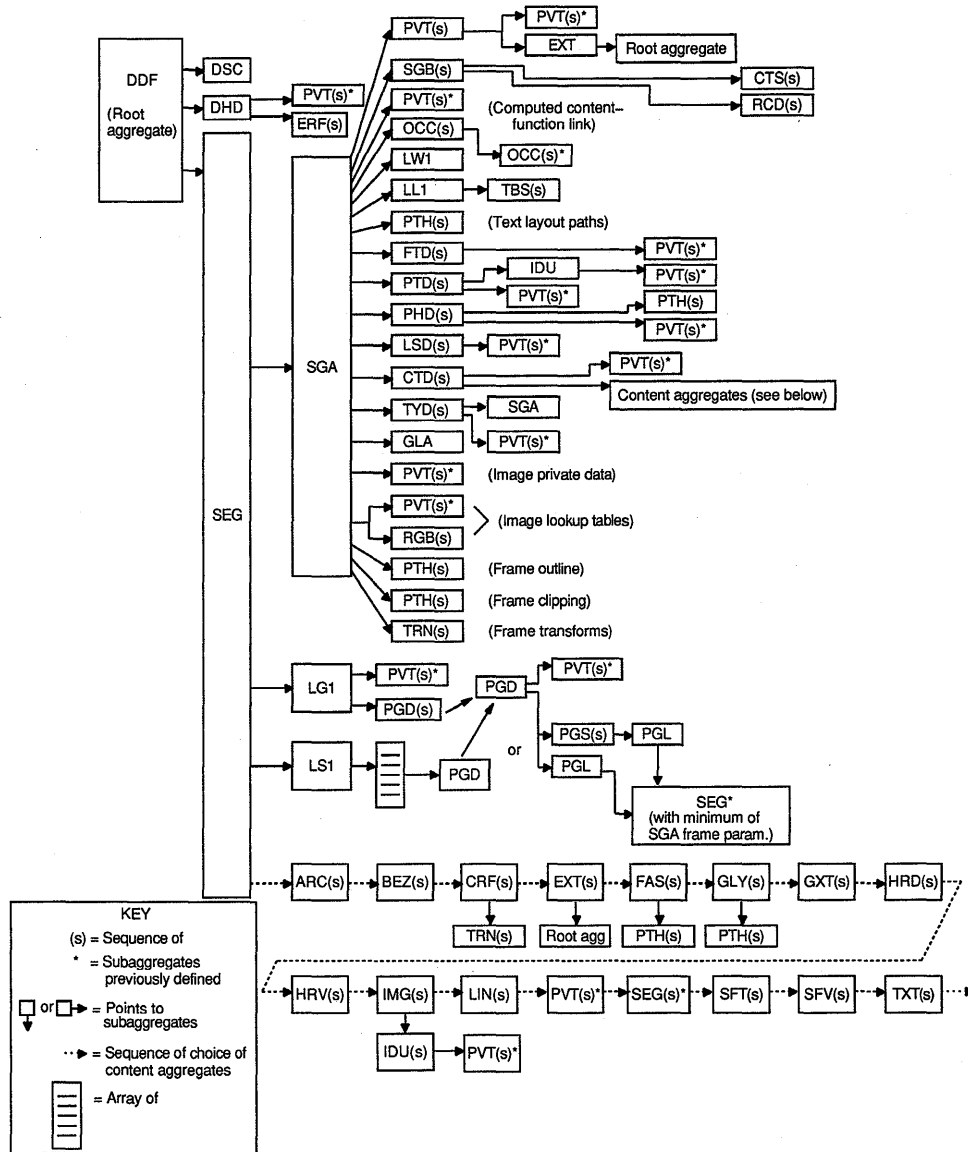
(continued on next page)

**Table 1-2 (Cont.): DDIF Standard Aggregates**

| <b>Aggregate</b>            | <b>Definition</b>      |
|-----------------------------|------------------------|
| <b>Secondary Aggregates</b> |                        |
| DDIF\$_IDU                  | Image data unit        |
| DDIF\$_LG1                  | Generic layout 1       |
| DDIF\$_LL1                  | Layout attributes 1    |
| DDIF\$_LS1                  | Specific layout 1      |
| DDIF\$_LSD                  | Line style definition  |
| DDIF\$_LW1                  | Wrap attributes 1      |
| DDIF\$_OCC                  | Occurrence definition  |
| DDIF\$_PGD                  | Page description       |
| DDIF\$_PGL                  | Page layout            |
| DDIF\$_PGS                  | Page select            |
| DDIF\$_PHD                  | Path definition        |
| DDIF\$_PTD                  | Pattern definition     |
| DDIF\$_PTH                  | Composite path         |
| DDIF\$_RCD                  | Record definition      |
| DDIF\$_RGB                  | RGB lookup table entry |
| DDIF\$_SGA                  | Segment attributes     |
| DDIF\$_SGB                  | Binding                |
| DDIF\$_TBS                  | Tab stop               |
| DDIF\$_TRN                  | Transformation         |
| DDIF\$_TYD                  | Type definition        |

*Segment attributes* define the presentation and processing characteristics of a segment of document content. Figure 1-1 shows the hierarchy and relationship of the DDIF aggregates, including the segment attributes.

Figure 1-1: DDIF Document Aggregate Hierarchy



ZK-1863A-GE

## 1.4 DDIF Tags

Table 1-3 lists and describes all the DDIF tag types used in processing compound document data.

**Table 1–3: DDIF Tags**

| <b>Tag Type</b>                    | <b>Definition</b>   |
|------------------------------------|---|
| Content categories                 | <p>DDIF content is divided into categories. The content category of a segment is denoted by a tag on that segment or a parent segment.</p> <p>Content category tags denote one of the following, which are specified by the DDIF\$_SGA_CONTENT_CATEGORY item:</p> <ul style="list-style-type: none"> <li>• \$2D (graphics)</li> <li>• \$I (image)</li> <li>• \$PDL (page description language)</li> <li>• \$T (text)</li> <li>• \$TBL (table)</li> <li>• \$PRIVATE</li> </ul>   |
| Content processing characteristics | <p>Segment tags are used to indicate the processing characteristics of content, including relationships to user interfaces and indications of special constraints on content.</p> <p>Content processing characteristics tags denote one of the following, which are specified by the DDIF\$_SGA_SEGMENT_TAGS item:</p> <ul style="list-style-type: none"> <li>• \$AN (annotation)</li> <li>• \$CRF (cross-reference)</li> <li>• \$E (emphasis)</li> <li>• \$F (figure)</li> <li>• \$FN (footnote)</li> <li>• \$GO (graphic object)</li> <li>• \$GRP (group member)</li> <li>• \$I (index)</li> <li>• \$L (list)</li> <li>• \$LBL (label)</li> <li>• \$LE (list element)</li> <li>• \$LIT (literal)</li> <li>• \$P (paragraph)</li> <li>• \$S (section)</li> <li>• \$TTL (title)</li> <li>• \$PRIVATE</li> </ul> |
| Content streams in layout          | <p>DDIF content elements may be tagged by stream. A given galley on a page accepts content only from certain streams. For example, footnote galleys accept content from the footnote stream only. While the footnote content is logically embedded within the content of the paragraph that references it, it appears in the galley at the bottom of the page, or even at the end of the chapter.</p>   |

(continued on next page)

**Table 1–3 (Cont.): DDIF Tags**

| Tag Type | Definition   |
|----------|--|
|          | <p>Content streams in layout tags denote one of the following types of streams, which are specified by the DDIF\$_SGA_CONTENT_STREAMS item and the DDIF\$_GLY_STREAMS item:</p> <ul style="list-style-type: none"><li>• \$DB (document body, initial content stream)</li><li>• \$EN (end note)</li><li>• \$FN (footnote)</li><li>• \$IX (index content)</li><li>• \$MN (margin note)</li><li>• \$TOC (table of contents)</li></ul> |

## 1.5 DDIF Initial Values

Example 1–1 is an analysis output of the initial segment attributes of a DDIF document. If the DDIF\$\_INHERIT\_ATTRIBUTES processing option is used by a receiver of a DDIF document, these attributes are applied to the root segment of the document as if they were inherited from a segment prior to the root segment. (Attributes specified on the root segment will override these.)

If the processing option is not used, the receiver must perform this initial attribute inheritance. For a description of the DDIF\$\_INHERIT\_ATTRIBUTES processing option, see Section 1.6.1.

## Example 1-1: Analysis Output of DDIF Initial Segment Attributes

---

```
SEG_SPECIFIC_ATTRIBUTES
{
  SGA_CONTENT_STREAMS
  (
    "%H244442" ! "$DB"
  )
  SGA_CONTENT_CATEGORY T_CATEGORY "$T"
  SGA_LANGUAGE 1
  SGA_LEGEND_UNIT_N 1
  SGA_LEGEND_UNIT_D 1
  SGA_LEGEND_UNIT_NAME
  (
    ISO_LATIN1 "inches"
  )
  SGA_UNITS_PER_MEASURE 1200
  SGA_UNIT_NAME
  (
    ISO_LATIN1 "BMU"
  )
  SGA_LAYOUT_C GALLEY_LAYOUT
  SGA_LAYGLY_WRAP
  {
    LW1_WRAP_FORMAT FMT_FLUSH_PATH_BEGIN
    LW1_QUAD_FORMAT FMT_FLUSH_PATH_BEGIN
    LW1_HYPHENATION_FLAGS "%B10000000000000000000000000000000"
    LW1_MAXIMUM_HYPH_LINES 3
    LW1_MAXIMUM_ORPHAN_SIZE 3
    LW1_MAXIMUM_WIDOW_SIZE 3
  }
  SGA_LAYGLY_LAYOUT
  {
    LLI_INITIAL_DIRECTIVE DIR_NULL
    LLI_BREAK_BEFORE BREAK_IF_NEEDED
    LLI_BREAK_WITHIN BREAK_IF_NEEDED
    LLI_BREAK_AFTER BREAK_IF_NEEDED
    LLI_INITIAL_INDENT_C VALUE_CONSTANT
    LLI_INITIAL_INDENT 0
    LLI_LEFT_INDENT_C VALUE_CONSTANT
    LLI_LEFT_INDENT 0
    LLI_RIGHT_INDENT_C VALUE_CONSTANT
    LLI_RIGHT_INDENT 0
    LLI_SPACE_BEFORE_C VALUE_CONSTANT
    LLI_SPACE_BEFORE 0
    LLI_SPACE_AFTER_C VALUE_CONSTANT
    LLI_SPACE_AFTER 0
    LLI_LEADING_CONSTANT_C VALUE_CONSTANT
    LLI_LEADING_CONSTANT 0
    LLI_TAB_STOPS
    {
      TBS_HORIZONTAL_POSITION_C VALUE_CONSTANT
      TBS_HORIZONTAL_POSITION 1200
      TBS_TYPE LEFT_TAB
    }
  }
  SGA_PATTERN_DEFNS
  {
    PTD_NUMBER 0
    PTD_DEFN_C SOLID_COLOR
    PTD_SOL_COLOR_C TRANSPARENCY
  }
  {

```

---

(continued on next page)

### Example 1-1 (Cont.): Analysis Output of DDIF Initial Segment Attributes

---

```
PTD_NUMBER 1
PTD_DEFN_C SOLID_COLOR
PTD_SOL_COLOR_C RGB_COLOR
PTD_SOL_COLOR_R "%F1.000000e+00"
PTD_SOL_COLOR_G "%F1.000000e+00"
PTD_SOL_COLOR_B "%F1.000000e+00"
}
{
PTD_NUMBER 2
PTD_DEFN_C SOLID_COLOR
PTD_SOL_COLOR_C RGB_COLOR
PTD_SOL_COLOR_R "%F0.000000e+00"
PTD_SOL_COLOR_G "%F0.000000e+00"
PTD_SOL_COLOR_B "%F0.000000e+00"
}

!          standard patterns 3 through 62

{
PTD_NUMBER 3
PTD_DEFN_C STANDARD_PATTERN
PTD_PAT_NUMBER 3
PTD_PAT_COLORS
(
  1
  2
)
}
{
PTD_NUMBER 4
PTD_DEFN_C STANDARD_PATTERN
PTD_PAT_NUMBER 4
PTD_PAT_COLORS
(
  1
  2
)
}
!          .
!          .
!          .
!          V

{
PTD_NUMBER 62
PTD_DEFN_C STANDARD_PATTERN
PTD_PAT_NUMBER 62
PTD_PAT_COLORS
(
  1
  2
)
}
}
```

---

(continued on next page)

## Example 1-1 (Cont.): Analysis Output of DDIF Initial Segment Attributes

---

```
SGA_LINE_STYLE_DEFNS
{
  LSD_NUMBER 1
}
{
  LSD_NUMBER 2
  LSD_PATTERN
  (
    2
    1
  )
}
{
  LSD_NUMBER 3
  LSD_PATTERN
  (
    1
    2
  )
}
{
  LSD_NUMBER 4
  LSD_PATTERN
  (
    2
    1
    1
    1
  )
}
SGA_TXT_MASK_PATTERN 2
SGA_TXT_FONT 1
SGA_TXT_RENDITION
(
  RND_DEFAULT
)
SGA_TXT_HEIGHT_C VALUE_CONSTANT
SGA_TXT_HEIGHT 200
SGA_TXT_SET_SIZE_N 1
SGA_TXT_SET_SIZE_D 1
SGA_TXT_DIRECTION TXT_DIR_FORWARD
SGA_TXT_DEC_ALIGNMENT
(
  ISO_LATIN1 "."
  ISO_LATIN1 ","
  ISO_LATIN1 ")"
)
SGA_TXT_LEADER_SPACE_C VALUE_CONSTANT
SGA_TXT_LEADER_SPACE 0
SGA_TXT_LEADER_BULLET ISO_LATIN1 "."
SGA_TXT_LEADER_ALIGN NON_ALIGNED_LEADER
SGA_TXT_LEADER_STYLE BULLET_LEADER
SGA_TXT_PAIR_KERNING FALSE
SGA_LIN_WIDTH_C VALUE_CONSTANT
SGA_LIN_WIDTH 12
SGA_LIN_STYLE SOLID_LINE_STYLE
SGA_LIN_PATTERN_SIZE_C VALUE_CONSTANT
SGA_LIN_PATTERN_SIZE 24
SGA_LIN_MASK_PATTERN 2
SGA_LIN_END_START ROUND_LINE_END
SGA_LIN_END_FINISH ROUND_LINE_END
```

---

(continued on next page)



### Example 1-1 (Cont.): Analysis Output of DDIF Initial Segment Attributes

---

```
SGA_LIN_END_SIZE_C VALUE_CONSTANT
SGA_LIN_END_SIZE 12
SGA_LIN_JOIN ROUNDED_LINE_JOIN
SGA_LIN_MITER_LIMIT_N 10
SGA_LIN_MITER_LIMIT_D 1
SGA_LIN_INTERIOR_PATTERN 1
SGA_MKR_STYLE DOT_MARKER
SGA_MKR_MASK_PATTERN 2
SGA_MKR_SIZE_C VALUE_CONSTANT
SGA_MKR_SIZE 12
SGA_GLY_ATTRIBUTES
{
  GLA_TOP_MARGIN_C VALUE_CONSTANT
  GLA_TOP_MARGIN 0
  GLA_LEFT_MARGIN_C VALUE_CONSTANT
  GLA_LEFT_MARGIN 0
  GLA_RIGHT_MARGIN_C VALUE_CONSTANT
  GLA_RIGHT_MARGIN 0
  GLA_BOTTOM_MARGIN_C VALUE_CONSTANT
  GLA_BOTTOM_MARGIN 0
}
SGA_IMG_PIXEL_PATH 0
SGA_IMG_LINE_PROGRESSION 270
SGA_IMG_PP_PIXEL_DIST 1
SGA_IMG_LP_PIXEL_DIST 1
SGA_IMG_BRT_POLARITY ZERO_MAX_INTENSITY
SGA_IMG_GRID_TYPE RECTANGULAR_GRID
SGA_IMG_SPECTRAL_MAPPING MONOCHROME_MAP
SGA_IMG_COMP_SPACE_ORG FULL_COMPACTION
SGA_IMG_PLANES_PER_PIXEL 1
SGA_IMG_PLANE_SIGNIF LSB_MSB
SGA_IMG_NUMBER_OF_COMP 1
SGA_IMG_BITS_PER_COMP
(
  1
)
}
```

---

## 1.6 DDIF Processing Options

This section describes the DDIF input processing options, which are specified when creating a root aggregate or when opening a DDIF file for input using the following CDA Toolkit routines:

- CREATE ROOT AGGREGATE
- OPEN FILE

The processing options are applied to the aggregates returned by these CDA Toolkit routines:

- CONVERT AGGREGATE
- GET AGGREGATE
- CONVERT DOCUMENT
- GET DOCUMENT

The DDIF processing options are for input processing only and are provided as a convenience to calling applications. They modify the document as it is read, and these modifications are not reversible. If you read a document with input processing options and write it out again, the resulting output document will not be the same as the input document.

---

## 1.6.1 Inherit Attributes Processing Option

When you specify the inherit attributes processing option (DDIF\$\_INHERIT\_ATTRIBUTES), attribute inheritance is executed for all segments returned by the CONVERT AGGREATE routine or the GET AGGREGATE routine, or for the entire document structure returned by the CONVERT DOCUMENT or GET DOCUMENT routine.

Attribute inheritance for document segments is executed in the following order:

1. If the segment has a type reference that corresponds to a type definition, the attributes of the type are applied to the attributes of the segment that do not yet have values.
2. If the segment is the root segment, and a style guide is referenced in the document's header, the definitions and layout from the style guide are applied to the root segment.
3. For the root segment only, standard defined initial values are applied to the attributes of the segment that do not yet have values. This is similar to defining an implicit segment containing these attribute values as a parent to the root segment.

DDIF writing applications that want to use these initial values should leave these items blank in the root segment.

DDIF reading applications can request that the CDA Toolkit apply these initial values when returning the root segment by specifying the DDIF\$\_INHERIT\_ATTRIBUTES processing option. Otherwise, if you intend to apply inheritance yourself, you must apply these initial values to the root segment attributes that do not already have values. Section 1.5 shows the initial segment attributes of a DDIF document in analysis output format.

4. If the segment is not the root segment, attribute values of its parent segment are applied to the attributes of the segment that do not yet have values.

---

### 1.6.1.1 Type References and Definitions

If the segment has a type reference (the DDIF\$\_SEG\_SEGMENT\_TYPE identifier) that corresponds to a type definition, the attributes of the type are applied to the segment. The type definition can be found (in the DDIF\$\_SGA\_TYPE\_DEFNS identifier) in any of the segment's parents at any outer level. The parent segments are searched back up the tree until the definition is found. Note that the root segment may have inherited the type definition from the style guide.

Attributes in the segment are not replaced by type attributes. Only segment attributes that do not have values acquire the values of the type attributes. A type definition may have a type parent. The attributes in the type parent are applied first and take precedence over the attributes of the type itself.

---

### 1.6.1.2 Style Guides

Definitions from a style guide are applied to a document's root segment. Each definition in the specific attributes and page description in the generic layout of the style guide's root segment is copied to the corresponding item in the specific attributes or generic layout of the document's root segment, unless the definition or page description with the same identifier already exists in the document's root segment. If the units-per-measure of the style guide is different from that of the root document, the measurements in the style guide definitions or page descriptions are scaled before merging them into the root document. After reading and processing the specific attributes and generic layout of the style guide's root segment, no more of the style guide is read—the style guide's document content is ignored.

If a style guide references another style guide, the definitions from the second style guide are applied to the first style guide, before the first style guide is applied to the document that references it. There is no limit to the nesting level of style guides.

The definitions applied from the style guide's root segment attributes include the following:

- Content definitions (DDIF\$\_CTD)
- Font definitions (DDIF\$\_FTD)
- Line style definitions (DDIF\$\_LSD)
- Path definitions (DDIF\$\_PHD)
- Pattern definitions (DDIF\$\_PTD)
- Type definitions (DDIF\$\_TYD)
- Variable binding definitions (DDIF\$\_SGB)

Page descriptions (DDIF\$\_PGD) are applied from the generic layout of the style guide's root segment.

---

### 1.6.2 Retain Definitions Processing Option

The retain definitions processing option (DDIF\$\_RETAIN\_DEFINITIONS) instructs the CDA Toolkit to retain the parent segments that are necessary for the operation of the FIND DEFINITION routine. This option is required only if neither the DDIF\$\_INHERIT\_ATTRIBUTES nor the DDIF\$\_EVALUATE\_CONTENT options are specified, because either of these options also cause the retention of the required segments.

---

### 1.6.3 Evaluate Content Processing Option

When you specify the evaluate content processing option (DDIF\$\_EVALUATE\_CONTENT), content references are resolved and external content for segments is imported.

Content reference (DDIF\$\_CRF) aggregates are replaced with the value of the definition (DDIF\$\_CTD) they reference. The content definition may contain the value or may refer to an external reference where the value is to be found.

The type of computed content can be specified in segment attributes by the DDIF\$\_SGA\_COMPUTE\_C item. If a segment has the DDIF\$\_SGA\_COMPUTE\_C item present in the segment's attributes (DDIF\$\_SGA), the content may be imported from an external reference. If the value of the DDIF\$\_SGA\_COMPUTE\_C item is DDIF\$K\_REMOTE\_COMPUTE, the external content is imported and replaces the segment's original content. If the value of the DDIF\$\_SGA\_COMPUTE\_C item is DDIF\$K\_COPY\_COMPUTE, the external content is imported only if the segment has no content.

---

### 1.6.3.1 Content References and Definitions

When processing a content reference (DDIF\$\_CRF), parent segments are searched (immediate parent first, back up the tree) for the corresponding content definition. The content definition may contain a value or may refer to an external reference, in which case the value is imported. The content reference is replaced with the value from the content definition. If the content definition or value cannot be found, a fatal error is returned.

If a content definition does not directly specify a content value, it can refer to indexed external references defined in the document header meaning that the value for the definition is to be located in the external reference. The CDA Toolkit only handles DDIF external references. No attempt is made to import content from non-DDIF external references.

The content definition may specify an external reference index and a reference target label. The target label refers to a segment label in the external document. Although omitting the index would imply that the target segment is to be found in the current document, support for this is not implemented in the CDA Toolkit; if a content definition does not specify a value, the CDA Toolkit requires the external reference index.

The reference target label is optional. If it is missing, the root segment of the external reference is chosen. If the target label is specified, a segment of that label must exist in the external reference, and only that segment is chosen. The chosen segment is searched for content definitions (DDIF\$\_CTD) that it may have in the content definitions item of its segment attributes (DDIF\$\_SGA). If a content definition with the same label as the original content definition is found in the external reference's segment, its value is used to resolve the content reference in the parent document. It is possible that the content definition in the external reference may refer for its value to yet another content definition in a nested external reference.

If an external reference has a different units-per-measure value than the root document, the values of measures in the imported content are scaled to match the root document.

---

### 1.6.3.2 Computed Segment Content and External References

Segment attributes can refer to indexed external references defined in the document header, meaning that the content for the segment is to be located in the external reference. The CDA Toolkit only handles DDIF external references. No attempt is made to import content from non-DDIF external references.

The DDIF\$\_SGA\_COMPUTE\_C item of a segment's attributes specifies the type of computed content. Only computed content of the DDIF\$K\_COPY\_COMPUTE or DDIF\$K\_REMOTE\_COMPUTE types can be processed by the CDA toolkit. For the other types (DDIF\$K\_VARIABLE, DDIF\$K\_XREF, and DDIF\$K\_FUNCTION), the content (if any) of the segment is left as it is; evaluation of these types of computed content may be implemented in a future release.

If the computed content is `DDIF$K_COPY_COMPUTE`, the external content is imported and becomes the segment's content only if the segment originally had no content. If the computed content has the value `DDIF$K_REMOTE_COMPUTE`, the external content is always imported and replaces any content the segment may have.

The segment attributes may specify an external reference index and a reference target label. The target label refers to a segment label in the external document. Although omitting the index would imply that the target segment is to be found in the current document, support for this is not implemented in the CDA Toolkit; if a content definition does not specify a value, the CDA Toolkit requires the external reference index.

The reference target label is optional. If it is missing when importing the computed content for a segment, the root segment (and all its content) of the external reference is imported. If the target label is specified, a segment of that label must exist in the external reference, and only that segment is imported. Note that if an imported segment refers to definitions in any of its parent segments, those definitions will not be able to be resolved, because only the target segment is imported. If an imported segment refers to definitions in its own style guide, these will not be able to be resolved either, because the style guide is not applied to the external reference.

If an external reference has a different units-per-measure value than the root document, the values of measures in the imported content are scaled to match the root document.

---

#### 1.6.4 Discard Segments Processing Options

The discard segments processing options (`DDIF$_DISCARD_I_SEGMENTS`, `DDIF$_DISCARD_2D_SEGMENTS`, `DDIF$_DISCARD_T_SEGMENTS`, `DDIF$_DISCARD_TBL_SEGMENTS`, and `DDIF$_DISCARD_PDL_SEGMENTS`) cause segments of the image (`$I`), graphics (`$2D`), text (`$T`), table (`$TBL`), and page descriptions language (`$PDL`) categories, respectively, to be discarded. Note that if a segment is discarded, any nested segments it may contain are also discarded.

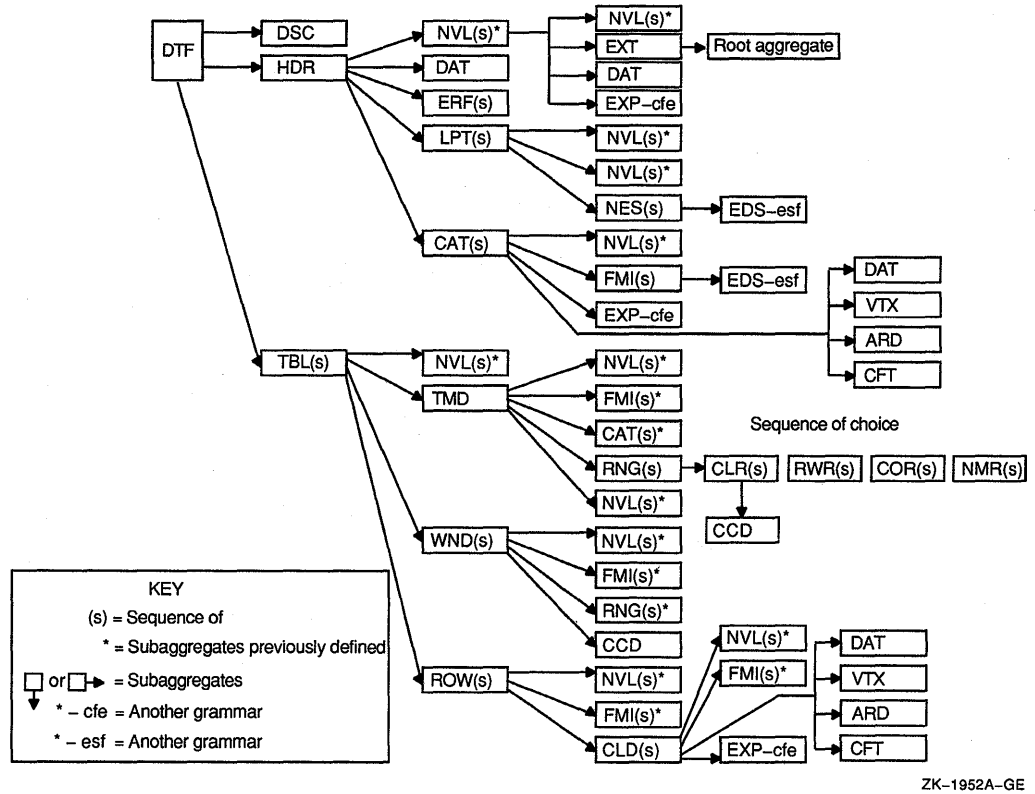
Since the decision to discard a segment is made before applying inheritance or initial values, if the discard text segments processing option is set and the root segment does not specify a content category, the root segment will not be discarded, even though the initial value for the content category (when not specified) is `$T`.

---

### 1.7 DTIF Document Aggregate Hierarchy

Figure 1–2 shows the hierarchy and relationship of the DTIF aggregates. In the DTIF domain, primary aggregates cannot be contained within other aggregates. The DTIF primary aggregates are `DTIF$_CLD` (cell data), `DTIF$_DSC` (document descriptor), `DTIF$_HDR` (document header), `DTIF$_ROW` (table rows), and `DTIF$_TBL` (table definition). The aggregate hierarchy for the DTIF domain does not permit recursive nesting of aggregates.

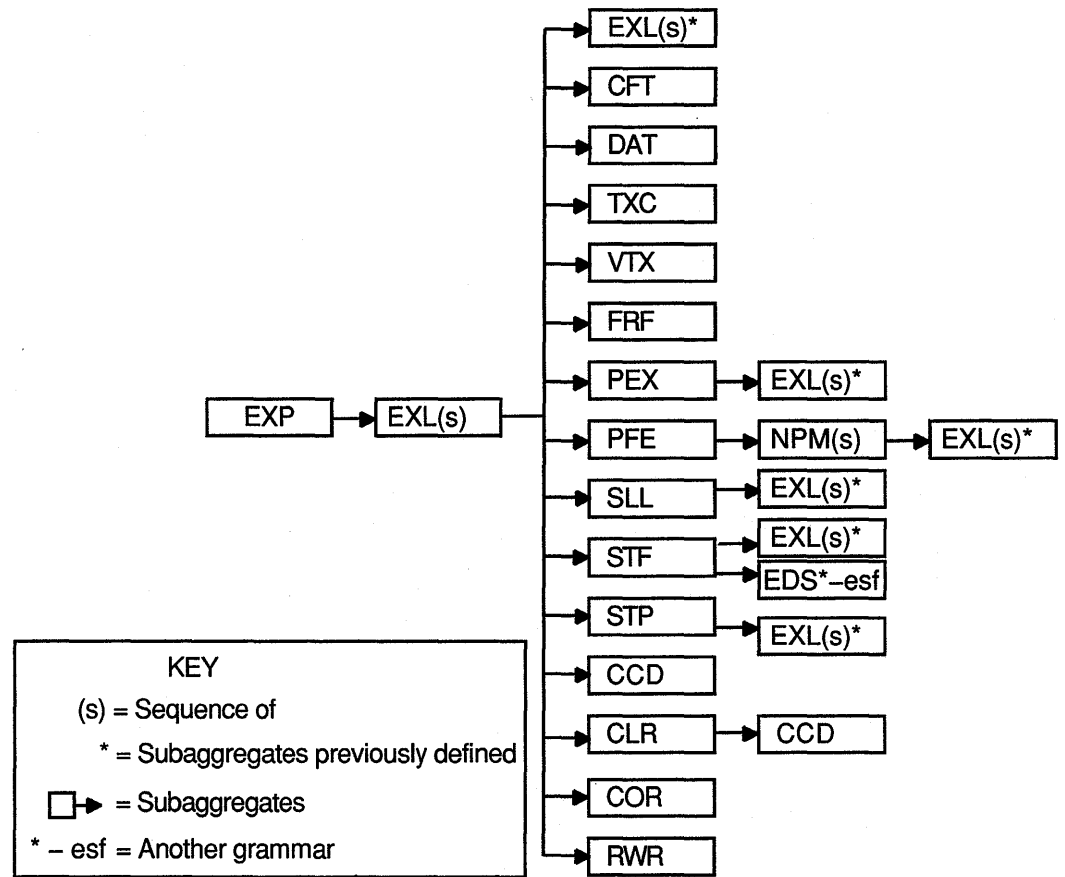
**Figure 1-2: DTIF Document Aggregate Hierarchy**



## 1.8 CFE Aggregate Hierarchy

Figure 1-3 shows the hierarchy and relationship of the CFE aggregates.

Figure 1-3: CFE Aggregate Hierarchy

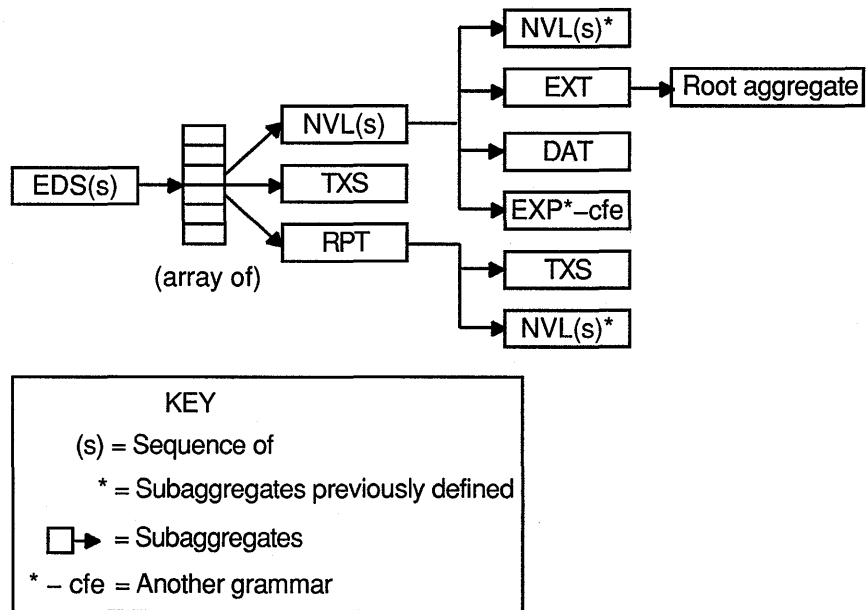


ZK-1953A-GE

## 1.9 ESF Aggregate Hierarchy

Figure 1-4 shows the hierarchy and relationship of the ESF aggregates.

**Figure 1-4: ESF Aggregate Hierarchy**



ZK-1954A-GE



# Bundled Converter Formats and Commands

---

This chapter describes the front and back ends, or input and output converter formats, provided with the CDA Toolkit on the VMS and ULTRIX operating systems. This chapter also describes the VMS and ULTRIX commands used to convert and to view CDA documents.

---

## 2.1 Base System Front Ends (Input Formats)

The CDA Converter architecture works by supplying a common converter kernel and front and back ends to support the various input and output formats. The following sections describe each supported front end, the data mapping between that input format and the in-memory format, any data loss that might occur during the conversion, and any other information specific to that front end.

---

### 2.1.1 DDIF Front End

The DDIF front end reads a file encoded in DDIF format and converts the information in the file to the CDA in-memory structure.

---

#### 2.1.1.1 Data Mapping

Because the input file format is DDIF, the information in the file maps directly to the CDA in-memory structure.

---

#### 2.1.1.2 Conversion Restrictions

The DDIF front end does not lose any data when converting a DDIF input file to the CDA in-memory structure. Again, this is because the input document type and the in-memory structure type are both DDIF.

---

#### 2.1.1.3 External File References

When the DDIF front end encounters an external file reference that is specified in the document header of your DDIF input file, it passes the reference through to the CDA Converter Kernel.

---

#### 2.1.1.4 Document Syntax Errors

If a document syntax error is encountered in the DDIF front end, that represents a fatal input processing error. The only way this can occur is if the input document is invalid. If the DDIF front end does encounter a document syntax error, the conversion process is stopped and no further input processing is performed.

---

## 2.1.2 DTIF Front End

The DTIF front end reads an input file stored in the DTIF format and converts the information in the file to the DTIF in-memory structure. For a description of the processing options you can specify when converting any table format to any document format, see Section 2.2.6.

---

### 2.1.2.1 Data Mapping

Because the input file format is DTIF, the information in the file maps directly to the DTIF in-memory structure.

---

### 2.1.2.2 Conversion Restrictions

The DTIF front end does not lose any data when converting a DTIF input file to the DTIF in-memory structure. Again, this is because the input document type and the in-memory structure type are both DTIF.

---

### 2.1.2.3 External File References

When the DTIF front end encounters an external file reference in your DTIF input file, it passes the reference through to the CDA Converter Kernel.

---

### 2.1.2.4 DTIF Syntax Errors

If a syntax error is encountered in the DTIF front end, that represents a fatal input processing error. The only way this can occur is if the input file is invalid. If the DTIF front end does encounter a syntax error, the conversion process terminates and no further input processing is performed.

---

## 2.1.3 Text Front End

The Text front end reads a standard text (ISO Latin1) file and converts the information in the file to the CDA in-memory structure. If the text file was entered as a DEC Multinational Character Set file on a character cell terminal or terminal emulator, the following conversions occur:

---

| Original Character       | Converted Character         |
|--------------------------|-----------------------------|
| Concurrency sign         | Diaeresis                   |
| Capital OE ligature      | Multiplication sign         |
| Capital Y with diaeresis | Capital Y with acute accent |
| Small oe ligature        | Division sign               |
| Small y with diaeresis   | Y with acute accent         |

---

---

### 2.1.3.1 Data Mapping

When you invoke the converter for a Text input file, all of the text in the input file is mapped to DDIF text content. Line breaks and form feeds are mapped to DDIF directives. One or more contiguous blank lines are interpreted as end-of-paragraph markers.

---

### 2.1.3.2 Conversion Restrictions

The Text front end does not lose any data when converting a simple text input file to the CDA in-memory structure. This is because no structure information is contained in a simple text file. All nonprinting characters (for example, ANSI escape sequences) are converted to space characters.

---

### 2.1.3.3 External File References

Text files do not contain external file references. Therefore, the Text front end does not evaluate external file references.

---

### 2.1.3.4 Document Syntax Errors

Because text files do not have any syntax defined, syntax errors cannot be encountered by the Text front end.

---

## 2.2 Base System Back Ends (Output Formats)

The following sections describe each back end supported by the CDA Converter architecture, the data mapping between the in-memory format and the particular output format, any data loss that might occur during the conversion, and any other information specific to that back end.

---

### 2.2.1 DDIF Back End

The DDIF back end converts the DDIF in-memory structure to the DDIF output format, and writes the information to the specified DDIF output file.

---

#### 2.2.1.1 Data Mapping

When you invoke the converter with the DDIF back end, the data mapping between the information in the CDA in-memory structure and the converted output file is one-to-one. This is because the in-memory structure type and the output document type are both DDIF.

---

#### 2.2.1.2 Conversion Restrictions

The DDIF back end does not lose any data when converting a CDA in-memory structure to a DDIF output file. Again, this is because the in-memory structure type and the output document type are both DDIF.

---

### 2.2.2 DTIF Back End

The DTIF back end converts the DTIF in-memory structure to the DTIF output format and writes the information to the specified DTIF output file.

---

#### 2.2.2.1 Data Mapping

Because the output file format is DTIF, the information in the DTIF in-memory structures maps directly to the DTIF output file.

---

### 2.2.2.2 Conversion Restrictions

The DTIF back end does not lose any data when converting a DTIF in-memory structure to a DTIF output file. Again, this is because the DTIF in-memory structure and the output file format are both DTIF. However, if the original document was DTIF, the loss will occur in the domain conversion.

---

### 2.2.2.3 External File References

The DTIF back end converts external file references stored in the DTIF in-memory structure to the output file format.

---

## 2.2.3 Text Back End

The Text back end converts only the text of the DDIF in-memory structure and writes the information to the specified text output file.

---

### 2.2.3.1 Data Mapping

When you invoke the converter for a text output file, all Latin1 text is written to the output text file. The formatting information is also preserved to the extent possible. Page coordinates are mapped to the nearest character cell (line,column) position.

---

### 2.2.3.2 Conversion Restrictions

When the Text back end is converting the in-memory structure to a text output file, all graphics and images are lost. Text attributes are also lost, although formatting is preserved to the extent possible.

It is possible that some text may be lost due to overwriting caused by preserving the layout, yet using a different font size. It is also possible that lines can be truncated if the specified page width is smaller than the page width specified in the document's format information. Neither of these cases can occur when the `OVERRIDE_FORMAT` processing option is specified because, in that case, the document's format information is ignored.

---

### 2.2.3.3 Processing Options

The text back end supports the following options:

|  |  |
|--|--|
| <code>ASCII_FALLBACK (ON,OFF)</code>   | This option causes the back end to output text in 7-bit ASCII. The fallback representation of the characters is described in the ANSI ASCII standard. If this option is not specified, the default is OFF; if this option is specified without a value, the default is assumed to be ON. |
| <code>CONTENT_MESSAGES (ON,OFF)</code> | This option causes the back end to put a message in the output file each time a nontext element is encountered in the in-memory CDA structures. If this option is not specified, the default is OFF; if this option is specified without a value, the default is assumed to be ON.       |

|                                       |  |
|---------------------------------------|--|
| <b>HEIGHT value</b>                   | This option lets you specify the maximum number of lines per page in your text output file. If you specify 0, the number of lines per page will correspond to the height specified in your document. If you additionally specify <code>OVERRIDE_FORMAT</code> , or if the document has no inherent page size, the document is formatted to the height value specified by this option. The default height is 66 lines.  |
| <code>OVERRIDE_FORMAT {ON,OFF}</code> | This option causes the text back end to ignore the document formatting information included in your document so that the text is formatted in a single large galley per page that corresponds to the size of the page as specified by the <code>HEIGHT</code> and <code>WIDTH</code> processing options. If this option is not specified, the default is <code>OFF</code> ; if this option is specified without a value, the default is assumed to be <code>ON</code> .  |
| <code>SOFT_DIRECTIVES {ON,OFF}</code> | This option causes the document to obey the soft directives contained in the document when creating your text output file. If this option is not specified, the default is <code>OFF</code> ; if this option is specified without a value, the default is assumed to be <code>ON</code> .  |
| <b>WIDTH value</b>                    | This option lets you specify the maximum number of columns of characters per page in your text output file. If you specify 0, the number of columns per page will correspond to the width specified in your document. If you additionally specify <code>OVERRIDE_FORMAT</code> , or if the document has no inherent page size, the document is formatted to the value specified by this processing option. If any lines of text exceed this width value, the additional columns are truncated. The default width is 80 characters. |

---

## 2.2.4 PostScript Back End

The PostScript back end converts the DDIF in-memory structure to PostScript and writes the information to the specified PostScript output file.

---

### 2.2.4.1 Data Mapping

When you invoke the converter for a PostScript output file, all document content is written to the output file.

---

### 2.2.4.2 Conversion Restrictions

When converting the in-memory structure to a PostScript output file, all document content is converted.

#### NOTE

On `ULTRIX` systems, the PostScript back end cannot process images. All images in a document that is converted to PostScript on an `ULTRIX` system are ignored.

---

### 2.2.4.3 Processing Options

The PostScript back end supports the following processing options:

- PAPER\_SIZE **paper-size**
- PAPER\_HEIGHT **paper-height**
- PAPER\_WIDTH **paper-width**
- PAPER\_TOP\_MARGIN **paper-top-margin**
- PAPER\_BOTTOM\_MARGIN **paper-bottom-margin**
- PAPER\_LEFT\_MARGIN **paper-left-margin**
- PAPER\_RIGHT\_MARGIN **paper-right-margin**
- PAPER\_ORIENTATION **orientation**
- EIGHT\_BIT\_OUTPUT **eight-bit-output-state**
- OUTPUT\_BUFFER\_SIZE **output-buffer-size**
- SOFT\_DIRECTIVES **soft-directives-state**
- WORD\_WRAP **word-wrap-state**
- PAGE\_WRAP **page-wrap-state**
- LAYOUT **layout-state**

The keyword is separated from its assigned value by one or more spaces or tabs. Note that, for all of the measurement options, the default unit of measure is inches (specified as “in”). Other supported units of measure are points (pts), centimeters (cm), and millimeters (mm).

The processing options are discussed individually in the following sections.

---

### 2.2.4.4 Paper Size Processing Option

The PAPER\_SIZE **paper-size** option lets you specify the size of the paper to be used when formatting the resulting PostScript output file. Valid values for **paper-size** are as follows:

---

| Keyword | Size  |
|---------|---|
| A0      | 841 x 1189 millimeters (33.13 x 46.85 inches) |
| A1      | 594 x 841 millimeters (23.40 x 33.13 inches)  |
| A2      | 420 x 594 millimeters (16.55 x 23.40 inches)  |
| A3      | 297 x 420 millimeters (11.70 x 16.55 inches)  |
| A4      | 210 x 297 millimeters (8.27 x 11.70 inches)   |
| A       | 8.5 x 11 inches                               |
| B       | 11 x 17 inches                                |
| C       | 17 x 22 inches                                |
| D       | 22 x 34 inches                                |
| E       | 34 x 44 inches                                |
| LEDGER  | 11 x 17 inches                                |
| LEGAL   | 8.5 x 14 inches                               |
| LETTER  | 8.5 x 11 inches                               |

| Keyword | Size             |
|---------|------------------|
| LP      | 13.7 x 11 inches |
| VT      | 8 x 5 inches     |

The A paper size (8.5 x 11 inches) is the default.

---

#### 2.2.4.5 Paper Height Processing Option

The PAPER\_HEIGHT **paper-height** processing option, in combination with the PAPER\_WIDTH processing option, lets you specify a paper size other than one of the predefined values provided. The default paper height is 11 inches.

---

#### 2.2.4.6 Paper Width Processing Option

The PAPER\_WIDTH **paper-width** processing option, in combination with the PAPER\_HEIGHT processing option, lets you specify a paper size other than one of the predefined sizes provided. The default paper width is 8.5 inches.

---

#### 2.2.4.7 Top Margin Processing Option

The PAPER\_TOP\_MARGIN **top-margin** processing option lets you select the width of the margin provided at the top of the page. The default value is .25 inches.

---

#### 2.2.4.8 Bottom Margin Processing Option

The PAPER\_BOTTOM\_MARGIN **bottom-margin** processing option lets you select the width of the margin provided at the bottom of the page. The default value is .25 inches.

---

#### 2.2.4.9 Left Margin Processing Option

The PAPER\_LEFT\_MARGIN **left-margin** processing option lets you select the width of the margin provided on the left-hand side of the page. The default value is .25 inches.

---

#### 2.2.4.10 Right Margin Processing Option

The PAPER\_RIGHT\_MARGIN **right-margin** processing option lets you select the width of the margin provided on the right-hand side of the page. The default value is .25 inches.

---

#### 2.2.4.11 Paper Orientation Processing Option

The PAPER\_ORIENTATION **orientation** processing option lets you select the paper orientation to be used in the output PostScript file. The valid values for the **orientation** argument are as follows:

| Keyword   | Meaning   |
|-----------|---|
| PORTRAIT  | The page is oriented so that the larger dimension is parallel to the vertical axis.   |
| LANDSCAPE | The page is oriented so that the larger dimension is parallel to the horizontal axis. |

The default is PORTRAIT.

---

#### 2.2.4.12 Eight Bit Output Processing Option

The `EIGHT_BIT_OUTPUT` **eight-bit-output-state** processing option lets you select whether or not the PostScript back end should use 8-bit output. You can specify a value of either ON or OFF for the **eight-bit-output-state** argument. The default is ON.

---

#### 2.2.4.13 Output Buffer Size Processing Option

The `OUTPUT_BUFFER_SIZE` **output-buffer-size** processing option lets you select the size of the output buffer. The value you specify must be within the following range:

$$64 \leq \text{output-buffer-size} \leq 256$$

The default is 132.

---

#### 2.2.4.14 Soft Directives Processing Option

The `SOFT_DIRECTIVES` **soft-directives-state** processing option lets you select whether or not the PostScript back end processes soft directives in the DDIF file in order to format output. (Soft directives specify such formatting commands as new line, new page, and tab.) If the PostScript back end processes soft directives, the output file will look more like you intended.

You can specify a value of either ON or OFF for the **soft-directive-state** argument. The default is ON.

---

#### 2.2.4.15 Word Wrap Processing Option

The `WORD_WRAP` **word-wrap-state** processing option lets you specify whether or not the PostScript back end performs word wrapping of any text that would exceed the right margin. You can specify a value of either ON or OFF for the **word-wrap-state** argument. The default is ON. If you specify OFF, the PostScript back end allows text to exceed the right margin.

---

#### 2.2.4.16 Page Wrap Processing Option

The `PAGE_WRAP` **page-wrap-state** processing option lets you specify whether or not the PostScript back end performs page wrapping of any text that would exceed the bottom margin. You can specify a value of either ON or OFF for the **page-wrap-state** argument. The default is ON.

---

#### 2.2.4.17 Layout Processing Option

The `LAYOUT` **layout-state** processing option lets you specify whether or not the PostScript back end processes the layout specified in the DDIF document. You can specify a value of either ON or OFF for the **layout-state** argument. The default is ON.

---

### 2.2.5 Analysis Back End

This back end produces an analysis of the CDA in-memory structure in the form of text output showing the named objects and values stored in the document. This is useful for debugging DDIF application programs.



The Analysis back end supports an `/INHERITANCE` processing option that specifies that the analysis is shown with attribute inheritance enabled. Inherited attributes are marked by “[default]” in the output. This option also causes external references to be imported into the main document.

---

## 2.2.6 Processing Options for Domain Conversion

When you are converting any table format to any document format, you can specify the following processing options using a format name of `DTIF_TO_DDIF`.

### **COLUMN\_TITLE**

Enables display of the column titles as contained in the column attributes centered at the top of the column.

### **CURRENT\_DATE**

Enables display of the current date and time in the bottom left corner of the page. The value is formatted according to the document’s specification for a default date and time.

### **DOCUMENT\_DATE**

Enables display of the document date and time as contained in the document header in the top left corner of the page. The value is formatted according to the document’s specification for a default date and time.

### **DOCUMENT\_TITLE**

Enables display of the document title(s) as contained in the document header centered at the top of the page, one string per line.

### **PAGE\_NUMBER**

Enables display of the current page number in the top right corner of the page.

### **PAPER\_SIZE = value**

Sets the paper size. The values are the same as those for the PostScript back end listed in Section 2.2.4.4.

### **PAPER\_HEIGHT = value**

Sets the paper height. The values are the same as those for the PostScript back end listed in Section 2.2.4.5.

### **PAPER\_WIDTH = value**

Sets the paper width. The values are the same as those for the PostScript back end listed in Section 2.2.4.6.

### **PAPER\_TOP\_MARGIN = value**

Sets the paper top margin. The values are the same as those for the PostScript back end listed in Section 2.2.4.7.

### **PAPER\_BOTTOM\_MARGIN = value**

Sets the paper bottom margin. The values are the same as those for the PostScript back end listed in Section 2.2.4.8.

### **PAPER\_LEFT\_MARGIN = value**

Sets the paper left margin. The values are the same as those for the PostScript back end listed in Section 2.2.4.9.

### **PAPER\_RIGHT\_MARGIN = value**

Sets the paper right margin. The values are the same as those for the PostScript back end listed in Section 2.2.4.10.

## **VMS Commands**

### **Base System Back Ends (Output Formats)**

**PAPER\_ORIENTATION = value**

Sets the paper orientation. The values are the same as those for the PostScript back end listed in Section 2.2.4.11.

---

## CONVERT/DOCUMENT

The VMS CONVERT/DOCUMENT command invokes the conversion of a revisable format file to another revisable or final form file from the DCL command line.

---

### Format

#### CONVERT/DOCUMENT

---

### QUALIFIERS

***[/OPTIONS=options-filename]***

Lets you specify the name of your options file that contains processing options. The default file extension for a VMS options file is .CDA\$OPTIONS.

---

### PARAMETERS

***input-file[/FORMAT=fmt-name]***

***input-file.DDIF/FORMAT=DDIF (default)***

Specifies the name of the input file to be converted. The default file type is DDIF. The /FORMAT qualifier enables you to specify the encoding format of the input file. The default input format is DDIF. Input formats bundled with the VMS operating system and their default file extensions are as follows:

| <b>Input Formats</b> | <b>File Extension</b> |
|----------------------|-----------------------|
| DDIF                 | .DDIF                 |
| DTIF                 | .DTIF                 |
| TEXT                 | .TXT                  |

---

Additional input formats are provided in Digital's CDA Converter Library, a layered product. Independent software vendors who write DDIF- and DTIF-conforming front and back ends also provide input formats that are layered on the VMS operating system. Contact your system manager for a complete list of input formats available on your system.

***output-file[/FORMAT=fmt-name]***

***output-file.DDIF/FORMAT=DDIF (default)***

Specifies the name of the output file to be converted. The default file type is DDIF. The /FORMAT qualifier enables you to specify the encoding format of the output file. The default output format is DDIF. Output formats bundled with the VMS operating system and their default file extensions are as follows:

## VMS Commands

### CONVERT/DOCUMENT

---

| Output Formats | File Extension |
|----------------|----------------|
| DDIF           | .DDIF          |
| DTIF           | .DTIF          |
| TEXT           | .TXT           |
| PS             | .PS            |
| ANALYSIS       | .CDA\$ANALYSIS |

---

Additional output formats are provided in Digital's CDA Converter Library, a layered product. Independent software vendors who write DDIF- and DTIF-conforming front and back ends also provide output formats that are layered on the VMS operating system. Contact your system manager for a complete list of output formats available on your system.

---

### Example

In this example, the command converts an input file named INPUT.DTIF, which has the DTIF format, to an output file named OUTPUT.DDIF, which has the DDIF format. The specified options file is named OPTIONS.CDA\$OPTIONS.

```
$ CONVERT/DOCUMENT -  
_ $ /OPTIONS=OPTIONS.CDA$OPTIONS -  
_ $ INPUT.DTIF/FORMAT=DTIF -  
_ $ OUTPUT.DDIF/FORMAT=DDIF
```

---

## VIEW

The VMS VIEW command invokes the CDA Viewer, which lets you view a compound document file on a character cell terminal or DECwindows display. Some display attributes are not processed when displaying the document because of the limitations of the viewing device. For a description of CDA Viewer support of Adobe font metrics, see Section 13.1.

---

### Format

**VIEW** [*input-file*]

---

### QUALIFIERS

**/FORMAT[=*fmt-name*]**

**/FORMAT=DDIF (default)**

Specifies the format of your input file. The input formats that you can use with the CDA Viewer depend on the CDA converters installed on your system. The default input format is DDIF. Input formats bundled with the VMS operating system and their default file extensions are as follows:

| Input Format | File Extension |
|--------------|----------------|
| DDIF         | .DDIF          |
| DTIF         | .DTIF          |
| PS           | .PS            |
| TEXT         | .TXT           |

PostScript file viewing is supported only in the DECwindows CDA Viewer and only when running to displays with servers containing the Display PostScript Extension. The CDA Viewer does not provide support of PostScript files on character cell terminals.

Additional input formats are provided in Digital's CDA Converter Library, a layered product. Independent software vendors who write DDIF- and DTIF-conforming front and back ends also provide input formats that are layered on the VMS operating system. Contact your system manager for a complete list of input formats available on your system.

**/HEIGHT=*nn***

Specifies the height of the page in number of characters. If you specify the /OVERRIDE\_FORMAT qualifier, or if the document being viewed has no inherent format, this page height is used. On the DECwindows display, the default height is 66 lines, which is equivalent to the default page height of 11 inches. On character cell displays, the page height defaults to your terminal's screen height. However, if you use the /OUTPUT qualifier, the page height depends on the page height of your document.

## VMS Commands

### VIEW

***/INTERFACE=DECWINDOWS***  
***/INTERFACE=CHARACTER\_CELL (default)***  
Specifies the type of display you are using.

***/OPTIONS=options-filename***  
Specifies the name of a file that contains processing options. The default file extension for a VMS options file is .CDA\$OPTIONS. This qualifier is used only with input formats for which you can specify processing options.

***/OUTPUT[=output-filename]***  
***/NOOUTPUT (default)***  
Specifies a file that you want to receive the output instead of having it displayed on the screen. You cannot use this qualifier when you have specified the ***/INTERFACE=DECWINDOWS*** qualifier.

If you specify ***/OUTPUT*** but you do not specify a file name, the CDA Viewer creates a file with the same name as your input file but with a file type of LIS. If you specify the ***/OUTPUT*** qualifier, you cannot also specify the ***/PAGE*** qualifier.

***/OVERRIDE\_FORMAT***  
***/NOOVERRIDE\_FORMAT (default)***  
Controls whether the CDA Viewer overrides the format of your document or uses the formatting information stored in your document.

***/PAGE***  
***/NOPAGE (default)***  
On a character cell terminal, this qualifier determines whether the output display of the CDA Viewer pauses after displaying each page of your file. If you specify the ***/PAGE*** qualifier, you can page backward and forward, or jump to the top or bottom of the document. Note that if you specify the ***/PAGE*** qualifier, you cannot specify either the ***/OUTPUT*** qualifier or the ***/INTERFACE=DECWINDOWS*** qualifier.

***/WIDTH=nn***  
Specifies the number of characters per line. If you specify the ***/OVERRIDE\_FORMAT*** qualifier, or if the document being viewed has no inherent format, this page width is used. On the DECwindows display, the default width is 85 characters, which is equivalent to the default page width of 8.5 inches. On character cell displays, the page width defaults to your terminal's screen width. However, if you use the ***/OUTPUT*** qualifier, the default is 132 columns.

---

## PARAMETER

***input-file***  
***input-file.DDIF (default)***  
Specifies the name of the file to be viewed. If you do not specify an input file name, you are prompted for one. You cannot use wildcard characters in the file specification. The default input file-encoding format is DDIF, and the default file type is DDIF. Valid input file formats are any of those for which there is a front end installed on the system.

### Example

This command invokes the CDA Viewer to view a file named INPUT.DTIF, which has the DTIF format. The display interface is DECwindows, and the CDA Viewer will override the document's default format. The display width will be 80 characters, and the display height will be 66 lines.

```
$ VIEW FOOBAR.DTIF -  
_ $ /FORMAT=DTIF -  
_ $ /OPTIONS=OPTIONS.CDA$OPTIONS -  
_ $ /NOOUTPUT -  
_ $ /NOPAGE -  
_ $ /INTERFACE=DECWINDOWS -  
_ $ /OVERRIDE_FORMAT -  
_ $ /WIDTH=80 -  
_ $ /HEIGHT=66
```

# ULTRIX Commands

## cdoc

---

## cdoc

The ULTRIX cdoc command converts the revisable format file, **inputfile**, to another revisable format or to a final form file. If **inputfile** is not specified, cdoc reads from standard input. Unless a destination file is specified with the **-o** option, the cdoc command writes files to standard output.

---

### Syntax

**cdoc** [*-s format*] [*-d format*] [*-O options\_file*] [*-o outputfile*] **inputfile**

---

### OPTIONS

#### ***[-s format]***

##### ***-s ddif (default)***

Specifies the format of **inputfile** and invokes an appropriate input converter as part of CDA. The ddif, dtif, dots (for analysis output only), and text converters are provided in the base system kit. Additional converters can be added by the CDA Converter Library and other layered products. Contact your system manager for a complete list of the input formats supported on your system. The default format is ddif.

#### ***[-d format]***

##### ***-d ddif (default)***

Specifies the format of **outputfile** and invokes an appropriate output converter as part of CDA. The ddif, dtif, text, analysis, and ps converters are provided in the base system kit. Additional converters can be added by the CDA Converter Library and other layered products. Contact your system manager for a complete list of the output formats supported on your system. The default format is ddif.

#### ***[-O options]***

Names the file passed to the input and output converters to control specific processing options for each converter. Refer to your documentation set for a description of converter options.

The options file has a default file type of `.cda_options`. Each line of the options file specifies a format name that can optionally be followed by `_input` or `_output` to restrict the option to either an input or output converter. The second word is a valid option preceded by one or more spaces or tabs, or a slash (/), and can contain upper- and lowercase letters, numbers, dollar signs, and underlines. The case of letters is not significant. If an option requires a value, then spaces, tabs, or an equal sign can separate the option from the value.

Each line can optionally be preceded by spaces and tabs and can be terminated by any character other than those that can be used to specify the format names and options. The syntax and interpretation of the text that follows the format name is specified by the supplier of the front and back end converters for the specified format.



To specify several options for the same input or output format, specify one option on a line. If an invalid option for an input or output format or an invalid value for an option is specified, the option may be ignored or an error message may be returned. Each input or output format that supports processing options specifies any restrictions or special formats required when specifying options.

By default, any messages that occur during processing of the options file are written to the system *standard error location*. For those input and output formats that support a LOG option, messages can be directed to a log file.

***[-o ofile]***  
***standard output (default)***

Specifies the name of the output file. If not specified, cdoc writes to standard output.

---

### Example

In this example, the command converts an input file named `input.dtif`, which has the `dtif` format, to an output file named `output.ddif`, which has the `ddif` format. The specified options file is named `options.cda_options`.

```
% cdoc \  
-s dtif \  
-d ddif \  
-O options.cda_options \  
-o output.ddif \  
input.dtif
```

# ULTRIX Commands

## dxvdoc

---

## dxvdoc

The ULTRIX dxvdoc command invokes the CDA Viewer, which enables you to view the input file on a workstation running DECwindows. The dxvdoc window contains a menu bar with menu items that allow you to view additional documents, change processing options, close open documents, or exit. If you invoke dxvdoc with no input file argument, you can specify the first document using the file selection box. For a description of CDA Viewer support of Adobe font metrics, see Section 13.1.

---

### Syntax

```
dxvdoc [-f format] [-O options_file] [-r] [-w paper_width] [-h  
paper_height] inputfile
```

---

### OPTIONS

#### ***-f format***

##### ***-f ddif (default)***

Specifies the format of **inputfile** and invokes an appropriate input converter as part of CDA. The ddif, dtif, and text converters are provided in the base system kit. Additional converters can be added by the CDA Converter Library and other layered products. Contact your system manager for a complete list of input formats supported on your system. The default format is ddif.

##### ***-O options***

Names the file passed to the input converter to control specific processing options in that converter. Refer to your documentation set for a description of converter options.

The options file has a default file type of .cda\_options. Each line of the options file specifies a format name that can optionally be followed by *\_input* or *\_output* to restrict the option to either input or output. The second word is a valid option preceded by one or more spaces, tabs, or a slash (/) and can contain upper- and lowercase letters, numbers, dollar signs, and underlines. The case of letters is not significant. If an option requires a value, then spaces, tabs, or an equal sign can separate the option from the value.

Each line can optionally be preceded by spaces and tabs and can be terminated by any character other than those that can be used to specify the format names and options. The syntax and interpretation of the text that follows the format name is specified by the supplier of the front and back ends for the specified format.

To specify several options for the same input or output format, specify one option on a line. If an invalid option for an input or output format or an invalid value for an option is specified, the option may be ignored or an error message may be returned. Each input or output format that supports processing options specifies any restrictions or special formats required when specifying options.

By default, any messages that occur during processing of the options file are written to the system *standard error location*. For those input formats that support a LOG option, messages can be directed to a log file.

**-r**

Specifies that the CDA Viewer is to override the format of the document. If the -r qualifier is not specified, the CDA Viewer retains the formatting information stored in the document.

**-w paper-width**

**-w 80 (default)**

Specifies the paper width in units of characters. Each character unit translates to 720 centipoints (7200 centipoints per inch or 10 characters per inch horizontally). The -w qualifier always specifies the fallback formatted document page width to be used when the -r (override format) qualifier is specified or when the document has no inherent format. If the -w qualifier is not specified and if the document has no inherent format, the default width is 85 characters, which is equivalent to the default page width of 8.5 inches.

**-h paper-height**

Specifies the paper height in units of characters. Each character unit translates to 1200 centipoints (7200 centipoints per inch or 6 characters per inch vertically). The -h qualifier always specifies the fallback formatted document page height to be used when the -r (override format) qualifier is specified or when the document has no inherent format. If the -h qualifier is not specified and if the document has no inherent format, the default height is 66 lines, which is equivalent to the default page height of 11 inches.

Command line parameters pertaining to XtInitialize() are also supported by dxvdoc (for example, -d \fnode\fp::0).

---

## Example

This command invokes the CDA Viewer to view a file named input.dtif, which has the dtif format. The display interface is DECwindows, and the CDA Viewer will override the document's default format. The display width will be 80 characters, and the display height will be 50 lines.

```
% dxvdoc -f dtif \  
-O options.cda_options \  
-r \  
-w 80 \  
-h 50 \  
input.dtif
```

# ULTRIX Commands

## vdoc

---

## vdoc

The ULTRIX vdoc command invokes the CDA Viewer that enables you to view the **inputfile** on a character-cell terminal. If **inputfile** is not specified, vdoc reads from standard input. For a description of CDA Viewer support of Adobe font metrics, see Section 13.1.

---

### Syntax

```
vdoc [-f format] [-O options_file] [-r] [-w paper_width] [-h  
paper_height] [-p] inputfile
```

---

### OPTIONS

#### ***[-f format]***

#### ***-f ddif (default)***

Specifies the format of **inputfile** and invokes an appropriate input converter as part of CDA. The ddif, dtif and text input converters are provided in the base system kit. Additional converters can be added by the CDA Converter Library and other layered products. Contact your system manager for a complete list of the input formats supported on your system. The default format is ddif.

#### ***-O options***

Names the file passed to the input converter to control specific processing options in that converter. Refer to your documentation set for a description of converter options.

The options file has a default file type of .cda\_options. Each line of the options file specifies a format name that can optionally be followed by **\_input** or **\_output** to restrict the option to either an input or output converter. The second word is a valid option preceded by one or more spaces, tabs, or a slash (/) and can contain upper- and lowercase letters, numbers, dollar signs, and underlines. The case of letters is not significant. If an option requires a value, then spaces, tabs, or an equal sign can separate the option from the value.

Each line can optionally be preceded by spaces and tabs and can be terminated by any character other than those that can be used to specify the format names and options. The syntax and interpretation of the text that follows the format name is specified by the supplier of the front and back end converters for the specified format.

To specify several options for the same input or output format, specify one option on a line. If an invalid option for an input or output format or an invalid value for an option is specified, the option may be ignored or an error message may be returned. Each input or output format that supports processing options specifies any restrictions or special formats required when specifying options.

By default, any messages that occur during processing of the options file are written to the system *standard error location*. For those input formats that support a LOG option, messages can be directed to a log file.

### **-r**

Specifies that the CDA Viewer is to override the format of the document. If the -r qualifier is not specified, the CDA Viewer retains the formatting information stored in the document.

### **-w paper-width**

#### **-w 80 (default)**

Specifies the paper width in units of characters. The -w qualifier always specifies the fallback formatted document page width to be used when the -r (override format) qualifier is specified or when the document has no inherent format. When used with the -p (page mode) qualifier, the display page width is determined from the terminal and is unrelated to the formatted page width. In nonpage mode, the specified -w value is used for both fallback document page width and the display page width. If the -w qualifier is not specified, the default width is 80 characters.

### **-h paper-height**

Specifies the paper height in units of characters. The -h qualifier always specifies the fallback formatted document page height to be used when the -r (override format) qualifier is specified or when the document has no inherent format. When used with the -p (page mode) qualifier, the display page height is determined from the terminal and is unrelated to the formatted page height. In nonpage mode, the specified -h value is used for both fallback document page height and the display page height. If the -h qualifier is not specified, the default height is dependent on the document.

### **-p**

Specifies that the CDA Viewer is to pause after displaying each page. The user can also page backward and go directly to the top or bottom of the document. If the -p qualifier is not specified, the CDA Viewer displays each page without pausing.

---

## Example

This command invokes the CDA Viewer to view a file named `input.dtif`, which has the `dtif` format. The display interface is a character cell terminal, and the CDA Viewer will override the document's default format. The display width will be 80 characters, and the display height will be 50 lines.

```
% vdoc -f dtif \  
-O options.cda_options \  
-r \  
-w 80 \  
-h 50 \  
-p \  
input.dtif
```



# Transferring CDA Documents

---

This chapter describes how to transfer CDA documents between VMS and ULTRIX systems using the mailing and copying procedures on both systems.

You can mail CDA documents in either revisable form or final PostScript format. The recipient of a PostScript mail message should extract the mail message to a file and delete any mail header information. The file can then be printed on a PostScript printer.

In general, use DECwindows mail on VMS and ULTRIX systems when mailing CDA documents. DECwindows mail ensures that all linked-to files, such as style files and images, are properly mailed along with the base document.

The file names of CDA documents that are mailed across operating systems will change to conform to the file naming conventions of the target operating system. For example, if you mail a document from a VMS system to an ULTRIX system and then extract the message, any dollar signs (\$) in the file name are changed to underscores (\_).

---

## 3.1 Sending CDA Documents on a VMS System

You can use DECwindows mail (`decw$mail.exe`) to send a CDA document and its links on a VMS system in two ways:

- Choose the File . . . menu item from the Create-Send menu to preview your file before sending it. A File Filter window will appear in which you can type the name of the file or select it by scrolling through the directory displayed in the dialog box. Press the OK button, and a Mail: Create window will use the CDA Viewer to display the file. You then specify the To:, CC:, and Subject: information and press the Send button.

You can also choose the File (no editor) . . . menu item from the Create-Send menu. A dialog box will prompt you for the name of the file you want to send. You can type the name of the file or select it by scrolling through the directory displayed in the dialog box. Press the OK button, and a Create window displays the text "Sending file **file name**". You then specify the To:, CC:, and Subject: information and press the Send button.

- Select the Create-Send box in the Mail window. In the Mail: Create window, specify the To:, CC:, and Subject: information. You can then choose one of the menu options from the File menu to specify the name of the file you want to send. Press the Send button.

You can also use the standard VAXmail SEND/NOEDIT command to send a CDA document on a VMS system. However, you can only mail simple documents, that is, documents that have no links to other files, including style files. If you attempt to use VAXmail to mail CDA documents with links, only the base document will be mailed.

---

## 3.2 Receiving CDA Documents on a VMS System

When you read a CDA document using DECwindows mail, the CDA Viewer widget displays the document. When you extract the mail message by selecting the Extract . . . menu item from the File menu, DECwindows mail displays a standard “Extracting” dialog box, which asks you for the name of the document. You can also specify whether you want to convert the file to a different output format. Press the OK button, and your document is extracted.

The Send operation changes all links so that all linked-to files, along with your base document, are placed in the same directory from which you invoked DECwindows mail.

---

## 3.3 Sending CDA Documents on an ULTRIX System

You can use DECwindows mail (dxmail) to send a CDA document and its links on an ULTRIX system in two ways:

- Choose the File . . . menu item from the Create-Send menu to preview your file before sending it. A File Filter window will appear in which you can type the name of the file or select it by scrolling through the directory displayed in the dialog box. Press the OK button, and a Mail: Create window will use the CDA Viewer to display the file. You then specify the To:, CC:, and Subject: information and press the Send button.

You can also choose the File (no editor) . . . menu item from the Create-Send menu, which creates a dialog box that prompts you for the name of the file you wish to send. You can type the name of file or select it by scrolling through the directory displayed in the dialog box. Press the OK button, and a Create window displays the text “Sending file **file name**”. You then specify the To:, CC:, and Subject: information and press the Send button.

- Select the Create-Send box in the Mail window. In the Mail: Create window, specify the To:, CC:, and Subject: information. You can then choose one of the menu options from the File menu to specify the name of the file you want to send. Press the Send button.

You can also use the standard ULTRIX mail utility to send a CDA document on an ULTRIX system. You can use the capsar utility if only the base system components are installed on your system, or the prompter command of mh if the optional mh subset is installed.

The following is an example of how to use the capsar utility:

```
capsar -c memo.doc | mail username
```

The following is an example of how to use the prompter command to send a DDIF document:

```
prompter -DDIF memo.doc
```

The prompter command creates a file called “draft”, which can be mailed using the “send” mh command.



---

## 3.4 Receiving CDA Documents on an ULTRIX System

When you read a CDA document using dxmail, the CDA Viewer widget displays the document. When you extract the mail message by selecting the Extract . . . menu item from the File menu, dxmail displays an "Extracting" dialog box, which asks you for the name of the directory in which to place all files. Press the OK button, and your document is extracted.

The Send operation changes all links so that all linked-to files, along with your base document, are placed in the same directory from which you invoked dxmail.

---

## 3.5 Copying CDA Documents on a VMS System

This section describes how to copy CDA documents while you are logged on to a VMS system.

- To copy a document from a VMS system to an ULTRIX system, type the following command:

```
$ exchange/network/transfer_mode=block -
_ $ document-name.doc ultrixnode::"/usr/users/username/document-name.doc"
```

- To copy a document from an ULTRIX system to a VMS system, type the following commands:

```
$ exchange/network/transfer_mode=block -
_ $ ultrixnode::"/usr/users/username/document-name.doc" document-name.doc
$ set file/semantics=[ddif,dtif] document-name.doc
```

- To copy a document from a VMS system to a VMS system, use the COPY command:

```
$ copy node::vms$disk:[username.dirname]doc-name doc-name
```

### NOTE

You must separately copy any files that are linked to the base CDA document, such as images or user-created style files. Also, ULTRIX file names are case sensitive. You must preserve the case of file names when copying them to ULTRIX systems by enclosing the ULTRIX file specification in double quotes (").

---

## 3.6 Copying CDA Documents on an ULTRIX System

This section describes how to copy CDA documents while you are logged on to an ULTRIX system.

- To copy a document from an ULTRIX system to a VMS system, type the following command:

```
% dcp -i document-name.doc \
vmsnode/username::' vms$disk:[username.dirname]document-name.doc'
```

The previous command prompts for a password for the VMS system. On the VMS system, the RMS semantic tag must be set after copying the document. To do this, type the following DCL command:

```
$ SET FILE/SEMANTICS=[DDIF,DTIF] document-name.doc
```

- To copy a document from a VMS system to an ULTRIX system, type the following command:

```
% dcp -i vmsnode::'vms$disk:[username.dirname]document-name.doc' document-name.doc
```

- To copy a document within an ULTRIX system, use the cp command:

```
% cp /user/users/username/doc-name.doc doc-name.doc
```

- To copy a document from an ULTRIX system to another ULTRIX system, use the dcp command:

```
% dcp -i ultrixnode::/usr/users/username/doc-name.doc doc-name.doc
```

This chapter provides an overview of the general structure of a DDIF document, and then provides detailed references for each DDIF-supported aggregate structure.

---

## 4.1 DDIF Document Structure Overview

Every DDIF document has the same general structure. The document must have a root aggregate, a document descriptor aggregate, a document header aggregate, and content. It is the content that differentiates one document from another; however, the overall document structure is the same.

Each DDIF aggregate type and its corresponding items is discussed in this chapter.

---

## 4.2 Generic Aggregate Items

In addition to the items defined by each individual aggregate, the CDA Toolkit also supports two “generic” aggregate items that can be specified for every DDIF aggregate described in this chapter. These items are described in Table 4–1.

**Table 4–1: Generic Aggregate Items**

| <b>Item Name</b>      | <b>Encoding</b> | <b>Meaning</b>  |
|-----------------------|-----------------|---|
| DDIF\$_USER_CONTEXT   | Longword        | Specifies additional longword for user                |
| DDIF\$_AGGREGATE_TYPE | Word            | Specifies the type of the aggregate; a read-only item |

## DDIF\$\_ARC

---

### DDIF\$\_ARC—Arc Content

The arc content aggregate specifies the control points for the creation of an arc. The DDIF\$\_ARC aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax      | Location    |
|-------------|-------------|
| Arc         | Figure B-26 |
| ArcPath     | Figure B-85 |
| AngleRef    | Figure B-67 |
| Size        | Figure B-72 |
| XCoordinate | Figure B-73 |
| YCoordinate | Figure B-74 |

---

### AGGREGATE FORMAT

| Item Name                   | Item Encoding           |
|-----------------------------|-------------------------|
| DDIF\$_ARC_FLAGS            | Longword                |
| DDIF\$_ARC_CENTER_X_C       | Measurement enumeration |
| DDIF\$_ARC_CENTER_X         | Variable                |
| DDIF\$_ARC_CENTER_Y_C       | Measurement enumeration |
| DDIF\$_ARC_CENTER_Y         | Variable                |
| DDIF\$_ARC_RADIUS_X_C       | Measurement enumeration |
| DDIF\$_ARC_RADIUS_X         | Variable                |
| DDIF\$_ARC_RADIUS_DELTA_Y_C | Measurement enumeration |
| DDIF\$_ARC_RADIUS_DELTA_Y   | Variable                |
| DDIF\$_ARC_START_C          | AngleRef enumeration    |
| DDIF\$_ARC_START            | Variable                |
| DDIF\$_ARC_EXTENT_C         | AngleRef enumeration    |
| DDIF\$_ARC_EXTENT           | Variable                |
| DDIF\$_ARC_ROTATION_C       | AngleRef enumeration    |
| DDIF\$_ARC_ROTATION         | Variable                |

---

**AGGREGATE ITEMS**
**DDIF\$\_ARC\_FLAGS****Encoding: longword**

A flags item that is used to control the rendition of the arc. Valid values for this item are as follows:

|                       |  |
|-----------------------|--|
| ddif\$m_arc_draw_arc  | If set, a line is drawn along the arc, rendered as specified by the active line attributes. The line-style pattern should begin at the starting point. If no flags are specified, this item is set by default.   |
| ddif\$m_arc_fill_arc  | If set, the arc is filled in the area defined by the arc primitive.  |
| ddif\$m_arc_pie_arc   | If set, the boundary for filling/outlining the arc is formed by the arc and the line segments joining the arc endpoints to the center. If the ddif\$m_arc_draw_arc flag is set, the entire pie arc will be outlined.   |
| ddif\$m_arc_close_arc | If set, and if the ddif\$m_draw_arc flag is set, the outline of the arc is closed. If ddif\$m_pie_arc is set, the outline is closed by lines joining the endpoints of the arc with the center. If ddif\$m_pie_arc is not set, the outline is closed by a line joining the two arc endpoints. |

**DDIF\$\_ARC\_CENTER\_X\_C****Encoding: measurement enumeration**

An arc center **x** indicator that indicates whether the **x**-coordinate of the center of the arc is specified as a variable or constant value. The coordinates of the center are relative to the frame which contains the arc.

**DDIF\$\_ARC\_CENTER\_X****Encoding: variable**

An arc center **x** item that specifies the **x**-coordinate of the center of the arc.

**DDIF\$\_ARC\_CENTER\_Y\_C****Encoding: measurement enumeration**

An arc center **y** indicator that indicates whether the **y**-coordinate of the center of the arc is specified as a variable or constant value. The coordinates of the center are relative to the frame which contains the arc.

**DDIF\$\_ARC\_CENTER\_Y****Encoding: variable**

An arc center **y** item that specifies the **y** coordinate of the center of the arc.

**DDIF\$\_ARC\_RADIUS\_X\_C****Encoding: measurement enumeration**

An arc radius **x** indicator that indicates whether the **x** radius of the arc is specified as a variable or constant value.

**DDIF\$\_ARC\_RADIUS\_X****Encoding: variable**

An arc radius **x** item that specifies the distance from the center of the arc to the perimeter of the arc as measured along the **x**-axis.

## DDIF\$\_ARC

### **DDIF\$\_ARC\_RADIUS\_DELTA\_Y\_C**

**Encoding:** *measurement enumeration*

An arc radius delta y indicator that indicates whether the delta y radius of the arc is specified as a variable or constant value.

### **DDIF\$\_ARC\_RADIUS\_DELTA\_Y**

**Encoding:** *variable*

An arc radius delta y item that specifies the length difference between the y radius and the x radius (for example, if the arc is the arc of an ellipse). The default value for this item is 0.

### **DDIF\$\_ARC\_START\_C**

**Encoding:** *AngleRef enumeration*

An arc start indicator that indicates whether the starting angle of the arc is specified as a variable or constant value.

### **DDIF\$\_ARC\_START**

**Encoding:** *variable*

An arc start item that specifies the angle at which the arc is begun. The default value for this item is 0. Arcs are measured in degrees counterclockwise starting from the positive x axis.

### **DDIF\$\_ARC\_EXTENT\_C**

**Encoding:** *AngleRef enumeration*

An arc extent indicator that indicates whether the extent of the arc is specified as a variable or constant value.

### **DDIF\$\_ARC\_EXTENT**

**Encoding:** *variable*

An arc extent item that is added to the arc start angle to determine the end of the arc. The default value for this item is 360 degrees.

### **DDIF\$\_ARC\_ROTATION\_C**

**Encoding:** *AngleRef enumeration*

An arc rotation indicator that indicates whether the angle of rotation of the arc is specified as a variable or constant value.

### **DDIF\$\_ARC\_ROTATION**

**Encoding:** *variable*

An arc rotation item that specifies the angle of rotation of the entire arc relative to the coordinate system. (This item is usually specified for elliptical arcs.) The default value for this item is 0 degrees.

---

## DDIF\$\_BEZ—Bézier Curve Content

A cubic Bézier curve is defined by four points. The first set of control points is the first four points in the sequence. Each subsequent set of three points uses the last point of the previous sequence as the first control point in the new sequence. The DDIF\$\_BEZ aggregate is referenced by the the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax          | Location    |
|-----------------|-------------|
| CubicBezier     | Figure B-25 |
| CubicBezierPath | Figure B-86 |

---

## AGGREGATE FORMAT

| Item Name         | Item Encoding                         |
|-------------------|---------------------------------------|
| DDIF\$_BEZ_FLAGS  | Longword                              |
| DDIF\$_BEZ_PATH_C | Array of type measurement enumeration |
| DDIF\$_BEZ_PATH   | Array of type variable                |

---

## AGGREGATE ITEMS

### **DDIF\$\_BEZ\_FLAGS**

**Encoding: longword**

A flags item that is used to control the rendition of the curve. The flags values are as follows:

|                         |  |
|-------------------------|--|
| ddif\$m_bez_draw_curve  | If set, the curve is drawn. If no flags are specified, this item is set by default.  |
| ddif\$m_bez_fill_curve  | If set, the area within the curve is filled according to the current fill attributes.  |
| ddif\$m_bez_close_curve | Determines whether an open or closed curve is drawn. (An open curve whose first and last points are connected by a straight line differs from a closed curve in that a closed curve reuses the first control point as the last control point. The total number of control points specified for a closed curve is a multiple of 3, and a closed cubic curve must consist of at least 6 points.) |

The default is ddif\$m\_bez\_draw\_curve.

## DDIF\$\_BEZ

### ***DDIF\$\_BEZ\_PATH\_C***

***Encoding: array of type measurement enumeration***

A curve path indicator that specifies whether the curve is specified as a variable or constant value.

### ***DDIF\$\_BEZ\_PATH***

***Encoding: array of type variable***

A curve path item that contains the **x,y** pairs that define the control points of the curve.

The points of the curve are stored in an array in a repeating **x,y**-pair format. For example, if you are storing values in this item, the first value you specify must be the **x** position of the first control point; the second value must be the **y** position of the first control point, and so on. Because these points are stored in an array, you must increment the aggregate index associated with the array each time you read or write a control point. The initial aggregate index value is 0.

The coordinates of control points are relative to the frame that contains the curve.



---

## DDIF\$\_CRF—Content Reference

The content reference content aggregate enables you to reference a generic content definition. The DDIF\$\_CRF aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax           | Location     |
|------------------|--------------|
| ContentReference | Figure B-41  |
| Transformation   | Figure B-104 |
| ContentDefnLabel | Figure B-61  |

---

### AGGREGATE FORMAT

| Item Name            | Item Encoding                     |
|----------------------|-----------------------------------|
| DDIF\$_CRF_TRANSFORM | Sequence of DDIF\$_TRN aggregates |
| DDIF\$_CRF_REFERENCE | String                            |

---

### AGGREGATE ITEMS

#### **DDIF\$\_CRF\_TRANSFORM**

**Encoding:** *sequence of DDIF\$\_TRN aggregates*

An optional content reference transformation item that specifies a transformation to be applied to all measurements in the referenced content definition. (For more information, see the description of the DDIF\$\_TRN aggregate.) If a transformation is not supplied, the measurements in the defined content are used unmodified.

#### **DDIF\$\_CRF\_REFERENCE**

**Encoding:** *string*

An optional content reference item that contains the label of the content definition being referenced. This item references the DDIF\$\_CTD\_LABEL item.

## DDIF\$\_CTD

---

### DDIF\$\_CTD—Content Definition

The content definition aggregate lets you specify a labeled generic content definition that can be referenced by nested segments. This enables you to include boilerplates or symbols from a library in your document.

A content definition references another content definition of the same name in the external document; the purpose of the label is simply to restrict the lookup of the segment. The content definition that is referenced must be in the segment associated with the specified label, or on the root segment of the external document if no label is specified. The DDIF\$\_CTD aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_CONTENT\_DEFNS.

Refer to these corresponding syntax diagrams:

---

| Syntax         | Location    |
|----------------|-------------|
| ContentDefn    | Figure B-60 |
| NamedValueList | Figure B-78 |

---

---

### AGGREGATE FORMAT

---

| Item Name                     | Item Encoding                     |
|-------------------------------|-----------------------------------|
| DDIF\$_CTD_LABEL              | String                            |
| DDIF\$_CTD_EXTERNAL_TARGET    | String                            |
| DDIF\$_CTD_EXTERNAL_ERF_INDEX | Integer                           |
| DDIF\$_CTD_VALUE              | Sequence of content               |
| DDIF\$_CTD_PRIVATE_DATA       | Sequence of DDIF\$_PVT aggregates |

---

---

### AGGREGATE ITEMS

#### **DDIF\$\_CTD\_LABEL**

**Encoding:** *string*

A content label item that specifies the label by which the content is referenced. This item is referenced by the DDIF\$\_CRF\_REFERENCE item.

#### **DDIF\$\_CTD\_EXTERNAL\_TARGET**

**Encoding:** *string*

An optional content external target item that specifies the label of the segment being referenced. If it is not specified, the entire document is being referenced. This item references the DDIF\$\_SEG\_ID item.

**DDIF\$\_CTD\_EXTERNAL\_ERF\_INDEX****Encoding: integer**

An optional external reference index item that specifies an index into a list of external references stored in the document header. If it is not specified, the reference is to the current document. This item references the DDIF\$\_DHD\_EXTERNAL\_REFERENCES item.

**DDIF\$\_CTD\_VALUE****Encoding: sequence of any of the following aggregates:**

|            |            |            |
|------------|------------|------------|
| DDIF\$_ARC | DDIF\$_BEZ | DDIF\$_CRF |
| DDIF\$_EXT | DDIF\$_FAS | DDIF\$_GLY |
| DDIF\$_GTX | DDIF\$_HRD | DDIF\$_HRV |
| DDIF\$_IMG | DDIF\$_LIN | DDIF\$_PVT |
| DDIF\$_SEG | DDIF\$_SFT | DDIF\$_SFV |
| DDIF\$_TXT |            |            |

An optional content value item that specifies the content elements being defined.

**DDIF\$\_CTD\_PRIVATE\_DATA****Encoding: sequence of DDIF\$\_PVT aggregates**

An optional content private data item that specifies the private data associated with the definition. For more information, see the description of the DDIF\$\_PVT aggregate.

## DDIF\$\_CTS

---

### DDIF\$\_CTS—Counter Style

The counter style aggregate describes a display style to be used for counters. The DDIF\$\_CTS aggregate is referenced by the parent aggregate item DDIF\$\_SGB\_CTR\_STYLE.

Refer to these corresponding syntax diagrams:

| Syntax       | Location     |
|--------------|--------------|
| CounterStyle | Figure B-109 |

---

### AGGREGATE FORMAT

| Item Name          | Item Encoding |
|--------------------|---------------|
| DDIF\$_CTS_STYLE_C | Enumeration   |
| DDIF\$_CTS_STYLE   | Variable      |

---

### AGGREGATE ITEMS

#### **DDIF\$\_CTS\_STYLE\_C**

**Encoding: enumeration; valid values are as follows:**

DDIF\$K\_NUMBER\_STYLE      The type of conversion used to present the variable as an alphanumeric string. In this case, the DDIF\$\_CTS\_STYLE item is encoded as an enumeration that accepts any one of the following values:

|                          |                               |
|--------------------------|-------------------------------|
| DDIF\$K_ARABIC_COUNTER   | Arabic numbers                |
| DDIF\$K_L_ROMAN_COUNTER  | Lowercase roman numerals      |
| DDIF\$K_U_ROMAN_COUNTER  | Uppercase roman numerals      |
| DDIF\$K_L_LATIN_COUNTER  | Lowercase Latin letters       |
| DDIF\$K_U_LATIN_COUNTER  | Uppercase Latin letters       |
| DDIF\$K_W_ARABIC_COUNTER | Wide arabic numbers           |
| DDIF\$K_WL_ROMAN_COUNTER | Wide lowercase roman numerals |
| DDIF\$K_WU_ROMAN_COUNTER | Wide uppercase roman numerals |
| DDIF\$K_WL_LATIN_COUNTER | Wide lowercase Latin letters  |
| DDIF\$K_WU_LATIN_COUNTER | Wide uppercase Latin letters  |

## DDIF\$\_CTS

|                          |                     |
|--------------------------|---------------------|
| DDIF\$K_WK_50_COUNTER    | Wide Katakana 50    |
| DDIF\$K_WK_IROHA_COUNTER | Wide Katakana Iroha |
| DDIF\$K_HEBREW_COUNTER   | Hebrew              |

**DDIF\$K\_BULLET\_STYLE** An array of type character string, for which the counter value constitutes an index that selects the bullet. If the counter value exceeds the number of elements in the array, then the array is reused. In this case, the DDIF\$\_CTS\_STYLE item is encoded as an array of type character string.

**DDIF\$K\_STYLE\_SEPARATOR** A constant text string added to the converted string as a value separator for the military style. In this case, the DDIF\$\_CTS\_STYLE item is encoded as a character string.

A counter style indicator that indicates the counter style to be used.

### **DDIF\$\_CTS\_STYLE**

**Encoding: variable**

A counter style item that contains the counter.

## DDIF\$\_DDF

---

### DDIF\$\_DDF—Document Root Aggregate

The DDIF document root aggregate identifies this particular instance of a DDIF document.

Refer to these corresponding syntax diagrams:

---

| Syntax       | Location   |
|--------------|------------|
| DDIFDocument | Figure B-6 |

---

---

### AGGREGATE FORMAT

---

| Item Name             | Item Encoding                  |
|-----------------------|--------------------------------|
| DDIF\$_DDF_DESCRIPTOR | Handle of DDIF\$_DSC aggregate |
| DDIF\$_DDF_HEADER     | Handle of DDIF\$_DHD aggregate |
| DDIF\$_DDF_CONTENT    | Handle of DDIF\$_SEG aggregate |

---

---

### AGGREGATE ITEMS

#### **DDIF\$\_DDF\_DESCRIPTOR**

**Encoding:** *handle of a DDIF\$\_DSC aggregate*

A document descriptor item that describes the document encoding. For more information, see the description of the DDIF\$\_DSC aggregate.

#### **DDIF\$\_DDF\_HEADER**

**Encoding:** *handle of a DDIF\$\_DHD aggregate*

A document header item that contains parameters and processing instructions that apply to the document as a whole. For more information, see the description of the DDIF\$\_DHD aggregate.

#### **DDIF\$\_DDF\_CONTENT**

**Encoding:** *handle of a DDIF\$\_SEG aggregate*

A document content item that specifies the content of the document. The DDIF\$\_SEG aggregate specifies the root or parent segment of the document. For more information, see the description of the DDIF\$\_SEG aggregate.

---

## DDIF\$\_DHD—Document Header

The document header aggregate contains data that pertains to the document as a whole; it describes the document to processors that receive it. The DDIF\$\_DHD aggregate is referenced by the parent aggregate item DDIF\$\_DDF\_HEADER.

Refer to these corresponding syntax diagrams:

| Syntax            | Location     |
|-------------------|--------------|
| DocumentHeader    | Figure B-8   |
| NamedValueList    | Figure B-78  |
| GeneralizedTime   | Figure B-128 |
| ExternalReference | Figure B-48  |

---

### AGGREGATE FORMAT

| Item Name                      | Item Encoding                             |
|--------------------------------|---|
| DDIF\$_DHD_PRIVATE_DATA        | Sequence of DDIF\$_PVT aggregates         |
| DDIF\$_DHD_TITLE               | Array of type character string            |
| DDIF\$_DHD_AUTHOR              | Array of type character string            |
| DDIF\$_DHD_VERSION             | Array of type character string            |
| DDIF\$_DHD_DATE                | String                                    |
| DDIF\$_DHD_CONFORMANCE_TAGS    | Array of type string with <b>add-info</b> |
| DDIF\$_DHD_EXTERNAL_REFERENCES | Sequence of DDIF\$_ERF aggregates         |
| DDIF\$_DHD_LANGUAGES_C         | Array of type enumeration                 |
| DDIF\$_DHD_LANGUAGES           | Array of type variable                    |
| DDIF\$_DHD_STYLE_GUIDE         | Integer                                   |

---

### AGGREGATE ITEMS

#### **DDIF\$\_DHD\_PRIVATE\_DATA**

**Encoding:** *sequence of DDIF\$\_PVT aggregates*

An optional private header data item that contains global information about the document not currently standardized by DDIF. For more information, see the description of the DDIF\$\_PVT aggregate. All interpretations of the private data are subject only to private agreements between the parties concerned.

#### **DDIF\$\_DHD\_TITLE**

**Encoding:** *array of type character string*

An optional title item that contains the user-visible name of the document.

## DDIF\$\_DHD

### **DDIF\$\_DHD\_AUTHOR**

**Encoding:** *array of type character string*

An optional author item that contains the name of the person or persons responsible for the information content of the document.

### **DDIF\$\_DHD\_VERSION**

**Encoding:** *array of type character string*

An optional version item that contains a character string used to distinguish this version of the document from all other versions.

### **DDIF\$\_DHD\_DATE**

**Encoding:** *string*

An optional date item that contains the date associated with this version of the document.

### **DDIF\$\_DHD\_CONFORMANCE\_TAGS**

**Encoding:** *array of type string with add-info*

An optional conformance tags item that contains a set of tags indicating the processing restrictions that apply to the document, and what subset of DDIF syntax has been used to describe the document. **Add-info** can take the following values:

|                             |   |
|-----------------------------|---|
| DDIF\$K_PRIVATE_CONFORMANCE | Indicates nonstandard processing restrictions                                     |
| DDIF\$K_SRQ_CONFORMANCE     | Indicates that the structure descriptions in this document were strictly observed |

### **DDIF\$\_DHD\_EXTERNAL\_REFERENCES**

**Encoding:** *sequence of DDIF\$\_ERF aggregates*

An optional external references item that contains a list of file names (or other system-specific file specifiers) that are referenced from within the document. (For more information, see the description of the DDIF\$\_ERF aggregate.) In the body of the document, external references are specified as indexes into this list. This item is referenced by the DDIF\$\_CTD\_EXTERNAL\_ERF\_INDEX item, the DDIF\$\_DHD\_STYLE\_GUIDE item, the DDIF\$\_PVT\_REFERENCE\_ERF\_INDEX item, and the DDIF\$\_SGA\_CPTCPY\_ERF\_INDEX item.

### **DDIF\$\_DHD\_LANGUAGES\_C**

**Encoding:** *ARRAY OF TYPE ENUMERATION*

An optional languages indicator that specifies the natural languages and programming languages that are delineated for processing by language tools. Valid values are as follows:

|                          |   |
|--------------------------|---|
| DDIF\$K_ISO_639_LANGUAGE | A string that selects a language and dialect that are specified using the ISO 639 Standard. In this case, the DDIF\$_DHD_LANGUAGES item is encoded as a string.   |
| DDIF\$K_OTHER_LANGUAGE   | A character string that indicates the language and dialect using a "user-readable" name; this is used for those languages and dialects not covered by the ISO 639 Standard. In this case, the DDIF\$_DHD_LANGUAGES item is encoded as a character string. |



**DDIF\$\_DHD\_LANGUAGES****Encoding: array of type variable**

An optional language item that contains a list of the actual languages from the selected language type that are delineated for processing.

If you specify DDIF\$\_DHD\_LANGUAGES\_C as DDIF\$\_K\_ISO\_639\_LANGUAGE, you must specify DDIF\$\_DHD\_LANGUAGES as one of the natural languages defined by the ISO 639 Standard, specifying the language symbol and country code. The following table illustrates some common examples:

| <b>Language/Country</b> | <b>String</b> |
|-------------------------|---------------|
| English/US              | E/USA/        |
| English/Great Britain   | E/GB/         |
| French/France           | F/F/          |
| German/Germany          | D/D/          |

**DDIF\$\_DHD\_STYLE\_GUIDE****Encoding: integer**

An optional style guide item that provides a reference to an external style guide that contains all or some of the presentation and layout attributes for the elements in the document. This item acts as an index into the DDIF\$\_DHD\_EXTERNAL\_REFERENCES item. The style guide must be encoded in DDIF format. This item references the DDIF\$\_DHD\_EXTERNAL\_REFERENCES item.

## DDIF\$\_DSC

---

### DDIF\$\_DSC—Document Descriptor

The document descriptor aggregate specifies the version level of DDIF used by this document, and identifies the software that created the document. The DDIF\$\_DSC aggregate is referenced by the parent aggregate item DDIF\$\_DDF\_DESCRIPTOR.

Refer to these corresponding syntax diagrams:

| Syntax             | Location   |
|--------------------|------------|
| DocumentDescriptor | Figure B-7 |

---

### AGGREGATE FORMAT

| Item Name                     | Item Encoding                  |
|-------------------------------|--------------------------------|
| DDIF\$_DSC_MAJOR_VERSION      | Integer                        |
| DDIF\$_DSC_MINOR_VERSION      | Integer                        |
| DDIF\$_DSC_PRODUCT_IDENTIFIER | String                         |
| DDIF\$_DSC_PRODUCT_NAME       | Array of type character string |

---

### AGGREGATE ITEMS

#### **DDIF\$\_DSC\_MAJOR\_VERSION**

**Encoding: integer**

A major version indicator that acts as the primary indicator of compatibility between the current version of DDIF and the version of DDIF used to encode the document.

The literal DDIF\$K\_MAJOR\_VERSION is defined to represent the highest major version supported by the CDA Toolkit. Applications should use this literal for the major version indicator. On output, the CDA Toolkit ignores the current value of this item and instead supplies the current version.

#### **DDIF\$\_DSC\_MINOR\_VERSION**

**Encoding: integer**

A minor version indicator that specifies the revision number of the current DDIF encoding.

The literal DDIF\$K\_MINOR\_VERSION is defined to represent the highest minor version supported by the CDA Toolkit. Applications should use this literal for the minor version indicator. On output, the CDA Toolkit ignores the current value of this item and instead supplies the current version.

***DDIF\$\_DSC\_PRODUCT\_IDENTIFIER******Encoding: string***

A product identifier item that contains a registered facility mnemonic representing the software that encoded the document.

The product identifier can be an acronym or abbreviation for the product name. This identifier is constant across versions of the product. If a product places private segment tags in the document, the product identifier string is used to prefix those segment tags.

***DDIF\$\_DSC\_PRODUCT\_NAME******Encoding: array of type character string***

A product name item that indicates the name of the product that encoded the document. If desired, the product name can be specified in multiple languages.

The product name string contains the version number of the product. The name of the product should be spelled in full, and should include a baselevel or version number.

## DDIF\$\_ERF

---

### DDIF\$\_ERF—External Reference

An external reference aggregate describes a source of data that is outside the document. It does so by specifying the data syntax and location of the external reference element. The DDIF\$\_ERF aggregate is referenced by the parent aggregate item DDIF\$\_DHD\_EXTERNAL\_REFERENCES.

Refer to these corresponding syntax diagrams:

| Syntax            | Location    |
|-------------------|-------------|
| ExternalReference | Figure B-48 |

---

### AGGREGATE FORMAT

| Item Name             | Item Encoding                  |
|-----------------------|--------------------------------|
| DDIF\$_ERF_DATA_TYPE  | Object identifier              |
| DDIF\$_ERF_DESCRIPTOR | Array of type character string |
| DDIF\$_ERF_LABEL      | Character string               |
| DDIF\$_ERF_LABEL_TYPE | String with <b>add-info</b>    |
| DDIF\$_ERF_CONTROL    | Enumeration                    |

---

### AGGREGATE ITEMS

#### **DDIF\$\_ERF\_DATA\_TYPE**

**Encoding:** *object identifier*

A reference data type item that identifies the data type of the external data object.

#### **DDIF\$\_ERF\_DESCRIPTOR**

**Encoding:** *array of type character string*

A reference descriptor item that provides a human-readable description of the data type.

#### **DDIF\$\_ERF\_LABEL**

**Encoding:** *character string*

A reference label item that provides the label by which the user or the system identifies the data object (that is, the file specification of the external file).

#### **DDIF\$\_ERF\_LABEL\_TYPE**

**Encoding:** *string with add-info*

A storage item that contains a tag that identifies the type of storage system in which the external reference is located. The following table lists the values for **add-info** and the corresponding string values.

|                            |  |
|----------------------------|--|
| DDIF\$K_PRIVATE_LABEL_TYPE | The label is a private label. In this case, the string can be any user-specified string.   |
| DDIF\$K_RMS_LABEL_TYPE     | The label is an RMS file specification. In this case, the string must be "\$RMS".  |
| DDIF\$K_UTX_LABEL_TYPE     | The label is an ULTRIX file specification. In this case, the string must be "\$UTX".   |
| DDIF\$K_MDS_LABEL_TYPE     | The label is an MS-DOS or OS/2 file specification. In this case, the string must be "\$MDS".   |
| DDIF\$K_STYLE_LABEL_TYPE   | The label is a style guide name. The category of the style guide is derived from the style guide tag. If the style guide is not specific to any category, then the style guide is part of the general category and the tag string must be "\$STYLE". A style guide with this tag can be used by any document type. If the style guide is specific to a category, then the style guide tag string is a qualified name of the format "\$STYLE.category". For example, "\$STYLE.chart" would be appropriate for style guides that are members of the chart category, and "\$STYLE.doc" would be appropriate for style guides that are members of the document category. |

For portability reasons, the style guide name must conform to the following rules:

- The name is always less than or equal to 8 characters.
- The name consists of the characters from the sets "a" through "z", "0" through "9", and "\_".
- The name always contains only the "simple" name of the style guide and does not contain a file type or a directory specification.

Style guide specifications are constructed from the style guide tag and the style guide name. The file name is the style guide name. The file type for general style guide files is .ddif. The file type for a category qualified style guide is formed by appending "\_style" to the tag suffix.

The directory for style guides is CDA\$LIBRARY on VMS systems and /usr/lib/cda on ULTRIX systems.

For example, if the DDIF\$\_ERF\_LABEL string is "wr\_examp" and the DDIF\$\_ERF\_LABEL\_TYPE tag string is "\$STYLE.doc", the resulting file specification is as follows:

VMS: CDA\$LIBRARY:wr\_examp.doc\_style  
ULTRIX: /usr/lib/cda/wr\_examp.doc\_style

### **DDIF\$\_ERF\_CONTROL**

**Encoding: enumeration; valid values are as follows:**

|                        |   |
|------------------------|---|
| DDIF\$K_COPY_REFERENCE | The referenced data object is transmitted along with the document, and is stored on the receiving system. |
|------------------------|---|

## DDIF\$\_ERF

DDIF\$K\_NO\_COPY\_REFERENCE

The referenced data is not transmitted with the document.

A control item that specifies how the referenced data object is treated when the document is transferred from one system to another. The default value for this item is DDIF\$K\_COPY\_REFERENCE.

---

## DDIF\$ \_EXT—External (PDL) Content

The external content aggregate specifies content that is external to the document. The DDIF\$ \_EXT aggregate is referenced by the parent aggregate items DDIF\$ \_CTD\_VALUE, DDIF\$ \_PVT\_DATA, and DDIF\$ \_SEG\_CONTENT.

Refer to the description of the EXTERNAL DDIF primitive in Table B-1.

---

### AGGREGATE FORMAT

| Item Name                         | Item Encoding     |
|-----------------------------------|-------------------|
| DDIF\$ _EXT_DIRECT_REFERENCE      | Object identifier |
| DDIF\$ _EXT_INDIRECT_REFERENCE    | Integer           |
| DDIF\$ _EXT_DATA_VALUE_DESCRIPTOR | String            |
| DDIF\$ _EXT_ENCODING_C            | Enumeration       |
| DDIF\$ _EXT_ENCODING              | Variable          |
| DDIF\$ _EXT_ENCODING_L            | Integer           |

---

### AGGREGATE ITEMS

#### **DDIF\$ \_EXT\_DIRECT\_REFERENCE**

**Encoding:** *object identifier*

An optional direct reference item that is used to identify the data type (syntax and semantics) of the external element.

#### **DDIF\$ \_EXT\_INDIRECT\_REFERENCE**

**Encoding:** *integer*

An optional indirect reference item. This item is reserved for future standardization.

#### **DDIF\$ \_EXT\_DATA\_VALUE\_DESCRIPTOR**

**Encoding:** *string*

An optional data value descriptor that is a text string describing the external data value to programs and/or users.

#### **DDIF\$ \_EXT\_ENCODING\_C**

**Encoding:** *enumeration; valid values are as follows:*

|                           |   |
|---------------------------|---|
| DDIF\$K_DOCUMENT_ENCODING | Nested document. In this case, the DDIF\$ _EXT_ENCODING item is encoded as a document root aggregate. |
| DDIF\$K_DDIS_ENCODING     | Nested document. In this case, the DDIF\$ _EXT_ENCODING item uses a DDIS encoding.                    |

## DDIF\$\_EXT

**DDIF\$\_K\_OCTET\_ENCODING** Octet-aligned encoding. In this case, the DDIF\$\_EXT\_ENCODING item is encoded as a string.

**DDIF\$\_K\_ARBITRARY\_ENCODING** Arbitrary. In this case, the DDIF\$\_EXT\_ENCODING item is encoded as a bit string.

An encoding indicator that indicates the method of encoding of the data value.

### ***DDIF\$\_EXT\_ENCODING***

***Encoding: variable***

An encoding item that specifies the external data value in the specified encoding.

### ***DDIF\$\_EXT\_ENCODING\_L***

***Encoding: integer***

An encoding length item that specifies the length (on input) of the encoding.



---

## DDIF\$\_FAS—Fill Area Set Content

The fill area set content aggregate specifies an arbitrary path that is filled as a unit, or an arbitrary outline. The DDIF\$\_FAS aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax        | Location    |
|---------------|-------------|
| FillAreaSet   | Figure B-27 |
| CompositePath | Figure B-84 |

---

### AGGREGATE FORMAT

| Item Name        | Item Encoding                     |
|------------------|-----------------------------------|
| DDIF\$_FAS_FLAGS | Longword                          |
| DDIF\$_FAS_PATH  | Sequence of DDIF\$_PTH aggregates |

---

### AGGREGATE ITEMS

#### **DDIF\$\_FAS\_FLAGS**

##### **Encoding: longword**

A fill area set flags item that is used to control the rendition of the fill area. Valid values for this item are as follows:

`ddif$m_fas_co_draw_border`

If set, a line is drawn along the path, using the current line attributes. If the start and end points of the path components are not coincident, a straight line connects the points. If no flags are specified, this item is set by default.

`ddif$m_fas_co_fill_area`

If set, the composite area is filled. The fill is performed using the even-odd rule, just as for polylines. (The even-odd rule states that if a ray is drawn from a point to infinity, the origin of the ray is considered inside the area (and hence is filled) if it crosses the area border an odd number of times.) If the start and end points of the path components are not coincident, a straight line connects the points.

The default value is `ddif$m_fas_co_draw_border`.

## DDIF\$\_FAS

### ***DDIF\$\_FAS\_PATH***

***Encoding: sequence of DDIF\$\_PTH aggregates***

A fill area set path item that specifies the composite path that constitutes the fill area set. For more information, see the description of the DDIF\$\_PTH aggregate.

---

## DDIF\$\_FTD—Font Definition

The font definition aggregate defines a font for use within a segment. The DDIF\$\_FTD aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_FONT\_DEFNS.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| FontDefn       | Figure B-49 |
| NamedValueList | Figure B-78 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DDIF\$_FTD_NUMBER       | Integer                           |
| DDIF\$_FTD_IDENTIFIER   | String                            |
| DDIF\$_FTD_PRIVATE_DATA | Sequence of DDIF\$_PVT aggregates |

---

### AGGREGATE ITEMS

**DDIF\$\_FTD\_NUMBER**

**Encoding: integer**

A font number item that is used to reference the font within the defining segment. This item is referenced by the DDIF\$\_SGA\_TXT\_FONT item.

**DDIF\$\_FTD\_IDENTIFIER**

**Encoding: string**

A font identifier item that specifies a font name.

**DDIF\$\_FTD\_PRIVATE\_DATA**

**Encoding: sequence of DDIF\$\_PVT aggregates**

An optional font private data item that specifies the private data associated with the definition. For more information, see the description of the DDIF\$\_PVT aggregate.

## DDIF\$\_GLA

---

### DDIF\$\_GLA—Galley Attributes

The galley attributes aggregate lets you specify the characteristics of a galley that can be acquired from a generic galley definition or specified locally. The DDIF\$\_GLA aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_GLY\_ATTRIBUTES.

Refer to these corresponding syntax diagrams:

| Syntax           | Location     |
|------------------|--------------|
| GalleyAttributes | Figure B-119 |

---

### AGGREGATE FORMAT

| Item Name                  | Item Encoding           |
|----------------------------|-------------------------|
| DDIF\$_GLA_TOP_MARGIN_C    | Measurement enumeration |
| DDIF\$_GLA_TOP_MARGIN      | Variable                |
| DDIF\$_GLA_LEFT_MARGIN_C   | Measurement enumeration |
| DDIF\$_GLA_LEFT_MARGIN     | Variable                |
| DDIF\$_GLA_RIGHT_MARGIN_C  | Measurement enumeration |
| DDIF\$_GLA_RIGHT_MARGIN    | Variable                |
| DDIF\$_GLA_BOTTOM_MARGIN_C | Measurement enumeration |
| DDIF\$_GLA_BOTTOM_MARGIN   | Variable                |

---

### AGGREGATE ITEMS

#### **DDIF\$\_GLA\_TOP\_MARGIN\_C**

**Encoding: measurement enumeration**

An optional galley top margin indicator that indicates whether the top margin is specified as a variable or constant value.

#### **DDIF\$\_GLA\_TOP\_MARGIN**

**Encoding: variable**

An optional galley top margin item that specifies the distance from the top of the galley to the top of the topmost text line or frame displayed in the galley. The initial value of this item is 0.

#### **DDIF\$\_GLA\_LEFT\_MARGIN\_C**

**Encoding: measurement enumeration**

An optional galley left margin indicator that indicates whether the left margin is specified as a variable or constant value.

**DDIF\$\_GLA\_LEFT\_MARGIN****Encoding: variable**

An optional galley left margin item that specifies the distance between the left side of the galley and the left side of the text lines and frames displayed in the galley. The initial value of this item is 0.

**DDIF\$\_GLA\_RIGHT\_MARGIN\_C****Encoding: measurement enumeration**

An optional galley right margin indicator that indicates whether the right margin is specified as a variable or constant value.

**DDIF\$\_GLA\_RIGHT\_MARGIN****Encoding: variable**

An optional galley right margin item that specifies the distance between the right side of the galley and the right side of the text lines and frames displayed in the galley. The initial value of this item is 0.

**DDIF\$GLA\_BOTTOM\_MARGIN\_C****Encoding: measurement enumeration**

An optional galley bottom margin indicator that indicates whether the bottom margin is specified as a variable or constant value.

**DDIF\$\_GLA\_BOTTOM\_MARGIN****Encoding: variable**

An optional galley bottom margin item that specifies the distance from the bottom of the galley to the bottom of the lowest text line or frame displayed in the galley. The initial value of this item is 0.

## DDIF\$\_GLY

---

### DDIF\$\_GLY—Layout Galley

The layout galley content aggregate lets you describe the shape and attributes of a single galley. Layout galley content can be specified in a content definition (DDIF\$\_CTD\_VALUE) or in segment content (DDIF\$\_SEG\_CONTENT). Layout galleys typically are specified as floating frames in page descriptions in layout.

A galley can be used to control the flow of text along a series of parallel paths. These paths are determined by a formatter based on the outline of the galley, the height of the characters on the lines, and other layout parameters such as leading.

Like graphic objects such as lines and curves, galleys are relative to a frame: either the page frame defined by a page layout description, or a floating frame. Also like graphic objects, galleys are imaged in the order in which they are described. Graphic elements can be described and imaged before, after, and between galleys. A galley is not imaged when it is selected for filling with text, but rather in the normal sequence in which objects in the frame are imaged. A page frame and its contents are imaged when the first galley on the page is selected.

The DDIF\$\_GLY aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

---

| Syntax        | Location     |
|---------------|--------------|
| LayoutGalley  | Figure B-118 |
| BoundingBox   | Figure B-42  |
| CompositePath | Figure B-84  |

---

---

### AGGREGATE FORMAT

---

| Item Name                      | Item Encoding           |
|--------------------------------|-------------------------|
| DDIF\$_GLY_ID                  | String                  |
| DDIF\$_GLY_BOUNDING_BOX_LL_X_C | Measurement enumeration |
| DDIF\$_GLY_BOUNDING_BOX_LL_X   | Variable                |
| DDIF\$_GLY_BOUNDING_BOX_LL_Y_C | Measurement enumeration |
| DDIF\$_GLY_BOUNDING_BOX_LL_Y   | Variable                |
| DDIF\$_GLY_BOUNDING_BOX_UR_X_C | Measurement enumeration |
| DDIF\$_GLY_BOUNDING_BOX_UR_X   | Variable                |
| DDIF\$_GLY_BOUNDING_BOX_UR_Y_C | Measurement enumeration |
| DDIF\$_GLY_BOUNDING_BOX_UR_Y   | Variable                |

---

| Item Name             | Item Encoding                     |
|-----------------------|-----------------------------------|
| DDIF\$_GLY_OUTLINE    | Sequence of DDIF\$_PTH aggregates |
| DDIF\$_GLY_FLAGS      | Longword                          |
| DDIF\$_GLY_STREAMS    | Array of type string              |
| DDIF\$_GLY_SUCCESOR_C | Enumeration                       |
| DDIF\$_GLY_SUCCESOR   | Variable                          |

## AGGREGATE ITEMS

### **DDIF\$\_GLY\_ID**

**Encoding: string**

A galley label item that specifies a label by which the galley can be referenced. This item is referenced by the DDIF\$\_GLY\_SUCCESOR item and by the DDIF\$\_LL1\_GALLEY\_SELECT item.

### **DDIF\$\_GLY\_BOUNDING\_BOX\_LL\_X\_C**

**Encoding: measurement enumeration**

A lower left corner **x** position indicator that indicates whether the lower left corner **x**-coordinate is specified as a variable or constant value.

### **DDIF\$\_GLY\_BOUNDING\_BOX\_LL\_X**

**Encoding: variable**

A lower left corner **x** position item that specifies the **x**-coordinate of the lower left corner of the galley.

### **DDIF\$\_GLY\_BOUNDING\_BOX\_LL\_Y\_C**

**Encoding: measurement enumeration**

A lower left corner **y** position indicator that indicates whether the lower left corner **y**-coordinate is specified as a variable or constant value.

### **DDIF\$\_GLY\_BOUNDING\_BOX\_LL\_Y**

**Encoding: variable**

A lower left corner **y** position item that specifies the **y**-coordinate of the lower left corner of the galley.

### **DDIF\$\_GLY\_BOUNDING\_BOX\_UR\_X\_C**

**Encoding: measurement enumeration**

An upper right corner **x** position indicator that indicates whether the upper right corner **x**-coordinate is specified as a variable or constant value.

### **DDIF\$\_GLY\_BOUNDING\_BOX\_UR\_X**

**Encoding: variable**

An upper right corner **x** position item that specifies the **x**-coordinate of the upper right corner of the galley.

### **DDIF\$\_GLY\_BOUNDING\_BOX\_UR\_Y\_C**

**Encoding: measurement enumeration**

An upper right corner **y** position indicator that indicates whether the upper right corner **y**-coordinate is specified as a variable or constant value.

## DDIF\$\_GLY

### **DDIF\$\_GLY\_BOUNDING\_BOX\_UR\_Y**

**Encoding: variable**

An upper right corner y position item that specifies the y-coordinate of the upper right corner of the galley.

### **DDIF\$\_GLY\_OUTLINE**

**Encoding: sequence of DDIF\$\_PTH aggregates**

An optional galley outline item that specifies the outline path. (For more information, see the description of the DDIF\$\_PTH aggregate.) The outline is constrained to fit within the bounding box, and defaults to the rectangle defined as the bounding box. Content is formatted inside the path, where the inside is determined by the even-odd winding rule. (The even-odd rule states that, if a ray is drawn from a point to infinity, the origin of the ray is considered inside the area (and hence will be filled) if it crosses the area border an odd number of times.)

### **DDIF\$\_GLY\_FLAGS**

**Encoding: longword**

An optional layout galley flags item that controls the display of the galley or its content. Valid values are as follows:

|                             |   |
|-----------------------------|---|
| ddif\$m_gly_vertical_align  | The elements in the galley are adjusted so that the vertical space in the galley is completely used.  |
| ddif\$m_gly_border          | A border is drawn around the outline of the galley.   |
| ddif\$m_gly_autoconnect     | If text overflows the galley during layout, it automatically flows into the successor galley. If the successor is a generic galley (is on a generic page), then an instance of that page will be created. |
| ddif\$m_gly_background_fill | The current fill pattern or color is used to fill the galley before the text that flows into the galley is imaged.  |

### **DDIF\$\_GLY\_STREAMS**

**Encoding: array of type string**

An optional galley streams item that specifies the content streams that can appear in the galley. The initial content stream is "\$DB", which denotes the document body. Other tags denoting content streams in layout that are registered in the DDIF Standard are the following:

|       |                          |
|-------|--------------------------|
| \$TOC | Table of contents stream |
| \$IX  | Index content stream     |
| \$FN  | Footnote stream          |
| \$MN  | Margin note stream       |
| \$EN  | End note stream          |

### **DDIF\$\_GLY\_SUCCESSOR\_C**

**Encoding: enumeration; valid values are as follows:**



## DDIF\$\_GLY

|                             |   |
|-----------------------------|---|
| DDIF\$K_GENERIC_GALLEY      | Indicates a galley on a page in the generic layout. In this case, the DDIF\$_GLY_SUCCESSOR item is encoded as a string.                             |
| DDIF\$K_SPECIFIC_GALLEY     | Indicates a galley on a page in specific layout. In this case, the DDIF\$_GLY_SUCCESSOR item is encoded as a string.                                |
| DDIF\$K_NO_SUCCESSOR_GALLEY | Indicates that there is no successor galley and overflow text is not displayed. In this case, you should not specify the DDIF\$_GLY_SUCCESSOR item. |

A galley successor indicator that indicates the type of galley to be used when text overflows.

### **DDIF\$\_GLY\_SUCCESSOR**

#### **Encoding: variable**

A galley successor item that specifies the galley used when text overflows. This item references the DDIF\$\_GLY\_ID item.

## DDIF\$\_GTX

---

### DDIF\$\_GTX—General Text Content

The general text content aggregate contains any text content of your document that uses a general character set. The DDIF\$\_GTX aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax        | Location    |
|---------------|-------------|
| TextPrimitive | Figure B-12 |

---

### AGGREGATE FORMAT

| Item Name          | Item Encoding    |
|--------------------|------------------|
| DDIF\$_GTX_CONTENT | Character string |

---

### AGGREGATE ITEMS

#### ***DDIF\$\_GTX\_CONTENT***

#### ***Encoding: character string***

A text content item that contains the text content.

The valid values for the character set identifier in the **add-info** parameter are listed in Table 4-2.

**Table 4-2: Character Set Identifiers**

| Identifier                | Character Set                       |
|---------------------------|-------------------------------------|
| CDA\$K_ISO_LATIN1         | ISO Latin 1                         |
| CDA\$K_ISO_LATIN2         | ISO Latin 2                         |
| CDA\$K_ISO_LATIN_ARABIC   | ISO Latin Arabic                    |
| CDA\$K_ISO_LATIN_GREEK    | ISO Latin Greek                     |
| CDA\$K_ISO_LATIN_HEBREW   | ISO Latin Hebrew                    |
| CDA\$K_JIS_KATAKANA       | JIS Roman, JIS Katakana             |
| CDA\$K_DEC_TECH           | DEC Special Graphics, DEC Technical |
| CDA\$K_DEC_MATH_ITALIC    | DEC Mathematics Italic              |
| CDA\$K_DEC_MATH_SYMBOL    | DEC Mathematics Symbol              |
| CDA\$K_DEC_MATH_EXTENSION | DEC Mathematics Extension           |

(continued on next page)

Table 4–2 (Cont.): Character Set Identifiers

| Identifier            | Character Set  |
|-----------------------|----------------|
| CDA\$K_DEC_PUBLISHING | DEC Publishing |
| CDA\$K_DEC_KANJI      | DEC Kanji      |
| CDA\$K_DEC_HANZI      | DEC Hanzi      |

## DDIF\$\_HRD

---

### DDIF\$\_HRD—Hard Directive

The hard directive content aggregate specifies a directive that is entered by the user. All directives are restricted to the \$T (text) content category. The DDIF\$\_HRD aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax              | Location    |
|---------------------|-------------|
| FormattingPrimitive | Figure B-18 |
| Directive           | Figure B-20 |

---

### AGGREGATE FORMAT

| Item Name            | Item Encoding |
|----------------------|---------------|
| DDIF\$_HRD_DIRECTIVE | Enumeration   |

---

### AGGREGATE ITEMS

#### **DDIF\$\_HRD\_DIRECTIVE**

*Encoding: enumeration; valid values are as follows:*

|                             |  |
|-----------------------------|--|
| DDIF\$K_DIR_NEW_PAGE        | Begins a new page.   |
| DDIF\$K_DIR_NEW_LINE        | Begins a new line of text.   |
| DDIF\$K_DIR_NEW_GALLEY      | Begins a new layout galley (such as a column). Software that does not support galley layout interprets the new galley directive as a new page.   |
| DDIF\$K_DIR_TAB             | Moves the horizontal text position to the next tab stop.   |
| DDIF\$K_DIR_SPACE           | Is treated as a space in the current font. The space directive is usually soft, and is used to indicate that software has inserted a space between wrapped lines.                          |
| DDIF\$K_DIR_HYPHEN_NEW_LINE | Specifies that the line break is preceded by a hyphen. This directive is typically soft, and is used to indicate that software has inserted a hyphen at the place where it broke the line. |

|                                     |  |
|-------------------------------------|--|
| <b>DDIF\$K_DIR_WORD_BREAK_POINT</b> | Identifies an embedded point at which a word may be broken, if need be, for justification.   |
| <b>DDIF\$K_DIR_LEADERS</b>          | Inserts leader characters according to the current leader attributes. A leader directive is treated like a space during justification, except that leader characters are inserted instead of space. The rendering of leaders is controlled by the current leader attributes and other text attributes. |
| <b>DDIF\$K_DIR_BACKSPACE</b>        | Specifies that the first character following this directive should be centered over the last character imaged.   |
| <b>DDIF\$K_NULL</b>                 | Suppresses the inheritance of the initial-directive element of layout attributes. This directive has no effect on imaging or processing.   |
| <b>DDIF\$K_DIR_NO_HYPHEN_WORD</b>   | Suppresses hyphenation until the next space character or space directive is encountered.   |

A hard directive item that specifies the type of hard directive (for example, a user-specified page break) to insert in the document.

## DDIF\$\_HRV

---

### DDIF\$\_HRV—Hard Value Directive

The hard value directive content aggregate specifies a hard directive that has a parametric value. The DDIF\$\_HRV aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax              | Location    |
|---------------------|-------------|
| FormattingPrimitive | Figure B-18 |
| ValueDirective      | Figure B-19 |
| EscapementDirective | Figure B-21 |
| VariableReset       | Figure B-22 |
| Escapement          | Figure B-47 |
| Ratio               | Figure B-70 |

---

### AGGREGATE FORMAT

| Item Name                 | Item Encoding           |
|---------------------------|-------------------------|
| DDIF\$_HRV_C              | Enumeration             |
| DDIF\$_HRV_ESC_RATIO_N    | Integer                 |
| DDIF\$_HRV_ESC_RATIO_D    | Integer                 |
| DDIF\$_HRV_ESC_CONSTANT_C | Measurement enumeration |
| DDIF\$_HRV_ESC_CONSTANT   | Variable                |
| DDIF\$_HRV_RESET_VARIABLE | String                  |
| DDIF\$_HRV_RESET_VALUE_C  | Expression enumeration  |
| DDIF\$_HRV_RESET_VALUE    | Variable                |

---

### AGGREGATE ITEMS

#### **DDIF\$\_HRV\_C**

**Encoding: enumeration; valid values are as follows:**

DDIF\$\_K\_DIR\_ESCAPEMENT

Indicates an escapement directive that specifies the relative or constant distance by which to increment the current text position. If you specify this value, you must supply values for the items DDIF\$\_HRV\_ESC\_RATIO\_N through DDIF\$\_HRV\_ESC\_CONSTANT.

**DDIF\$\_K\_DIR\_VARIABLE\_RESET** Indicates a variable reset directive that specifies a directive to reset the value of the specified variable. If you specify this value, you must supply values for the items DDIF\$\_HRV\_RESET\_VARIABLE through DDIF\$\_HRV\_RESET\_VALUE.

A hard value directive indicator that specifies whether the hard value directive is an escapement directive or a variable reset directive.

### **DDIF\$\_HRV\_ESC\_RATIO\_N**

**Encoding: integer**

An escapement ratio numerator item that specifies the magnitude of a ratio, which multiplies the em-space width for the current font. The width of an em space is often the same as the width of the capital letter M, but this depends on the font. The default value is 1, if this item is missing and if the numerator is present.

If both the numerator and the denominator are not specified, the ratio is not the default value, but is instead considered optionally absent.

### **DDIF\$\_HRV\_ESC\_RATIO\_D**

**Encoding: integer**

An escapement ratio denominator item that specifies the units of precision used in the ratio. The default value is 100, if this item is missing and if the numerator is present.

If both the numerator and the denominator are not specified, the ratio is not the default value, but is instead considered optionally absent.

### **DDIF\$\_HRV\_ESC\_CONSTANT\_C**

**Encoding: measurement enumeration**

An escapement constant indicator that indicates whether the escapement constant is specified as a variable or constant value.

### **DDIF\$\_HRV\_ESC\_CONSTANT**

**Encoding: variable**

An escapement constant item that specifies the constant measurement to be used as an escapement.

### **DDIF\$\_HRV\_RESET\_VARIABLE**

**Encoding: string**

A reset variable item that specifies the label of the variable to be reset by the hard value directive.

### **DDIF\$\_HRV\_RESET\_VALUE\_C**

**Encoding: expression enumeration**

A reset value indicator that indicates whether the hard value directive reset value is specified as a variable or constant value.

### **DDIF\$\_HRV\_RESET\_VALUE**

**Encoding: variable**

A reset value item that specifies the new value of the variable.

# DDIF\$\_IDU

---

## DDIF\$\_IDU—Image Data Unit

The image data unit aggregate describes an image in terms of its image coding attributes and the actual image data. The DDIF\$\_IDU aggregate is referenced by the parent aggregate items DDIF\$\_IMG\_CONTENT and DDIF\$\_PTD\_RAS\_PATTERN.

Refer to these corresponding syntax diagrams:

| Syntax           | Location    |
|------------------|-------------|
| ImageCodingAttrs | Figure B-35 |
| ImageDataUnit    | Figure B-34 |
| NamedValueList   | Figure B-78 |

---

## AGGREGATE FORMAT

| Item Name                      | Item Encoding                     |
|--------------------------------|-----------------------------------|
| DDIF\$_IDU_PRIVATE_CODING_ATTR | Sequence of DDIF\$_PVT aggregates |
| DDIF\$_IDU_PIXELS_PER_LINE     | Integer                           |
| DDIF\$_IDU_NUMBER_OF_LINES     | Integer                           |
| DDIF\$_IDU_COMPRESSION_TYPE    | Enumeration                       |
| DDIF\$_IDU_COMPRESSION_PARAMS  | Sequence of DDIF\$_PVT aggregates |
| DDIF\$_IDU_DATA_OFFSET         | Integer                           |
| DDIF\$_IDU_PIXEL_STRIDE        | Integer                           |
| DDIF\$_IDU_SCANLINE_STRIDE     | Integer                           |
| DDIF\$_IDU_PIXEL_ORDER         | Enumeration                       |
| DDIF\$_IDU_BITS_PER_PIXEL      | Integer                           |
| DDIF\$_IDU_PLANE_DATA          | String                            |

---

## AGGREGATE ITEMS

### **DDIF\$\_IDU\_PRIVATE\_CODING\_ATTR**

**Encoding:** *sequence of DDIF\$\_PVT aggregates*

An optional private data item that allows for the inclusion of application-private image coding attributes. For more information, see the description of the DDIF\$\_PVT aggregate.

### **DDIF\$\_IDU\_PIXELS\_PER\_LINE**

**Encoding:** *integer*

A pixels-per-line item that specifies the total number of pixels per scanline. Note that the pixels-per-line item does not necessarily represent the total number of bits per scanline.



**DDIF\$\_IDU\_NUMBER\_OF\_LINES****Encoding: integer**

A number-of-lines item that specifies the total number of scanlines in an image.

**DDIF\$\_IDU\_COMPRESSION\_TYPE****Encoding: enumeration; valid values are as follows:**

|                              |  |
|------------------------------|--|
| DDIF\$_K_PRIVATE_COMPRESSION | Private compression scheme   |
| DDIF\$_K_PCM_COMPRESSION     | Raw bitmap   |
| DDIF\$_K_G31D_COMPRESSION    | Consultative Committee on International<br>Telephony and Telegraphy (CCITT)<br>Group 3 1-dimensional |
| DDIF\$_K_G32D_COMPRESSION    | CCITT Group 3 2-dimensional  |
| DDIF\$_K_G42D_COMPRESSION    | CCITT Group 4 2-dimensional  |

A compression type item that indicates the compression scheme used to encode a particular plane of image data. DDIF\$\_K\_PCM\_COMPRESSION is the default.

**DDIF\$\_IDU\_COMPRESSION\_PARAMS****Encoding: sequence of DDIF\$\_PVT aggregates**

An optional compression parameters item that contains the parameters required for the specified compression. For more information, see the description of the DDIF\$\_PVT aggregate.

**DDIF\$\_IDU\_DATA\_OFFSET****Encoding: integer**

A data offset item that specifies the offset (in bits) from the start of the octet string to the first bit of image data. The default for the data offset item is 0.

**DDIF\$\_IDU\_PIXEL\_STRIDE****Encoding: integer**

An optional pixel stride item that specifies the difference in bit addresses between successive pixels.

Pixel stride is typically equal to the number of bits per pixel stored in a particular data plane. If pixel alignment requires fill bits between pixels, the difference between this value and the number of bits per pixel per component equals the fill value.

**DDIF\$\_IDU\_SCANLINE\_STRIDE****Encoding: integer**

An optional scanline stride item that specifies the difference in bit addresses between the starting bits of successive scanlines.

Scanline stride is typically equal to the number of bits (not pixels) per scanline. If scanline alignment requires fill bits between scanlines, the difference between scanline stride and the number of bits per pixel multiplied by the number of pixels per scanline equals the fill value. When image data is compressed, scanline stride has little meaning and is not present.

**DDIF\$\_IDU\_PIXEL\_ORDER****Encoding: enumeration; valid values are as follows:**

|                               |                                |
|-------------------------------|--------------------------------|
| DDIF\$_K_STANDARD_PIXEL_ORDER | Indicates standard pixel order |
|-------------------------------|--------------------------------|

## DDIF\$\_IDU

**DDIF\$K\_REVERSE\_PIXEL\_ORDER**                      Indicates reverse pixel order  
A pixel order item that specifies the order in which pixel data is stored within each byte. The default value is DDIF\$K\_STANDARD\_PIXEL\_ORDER.

### **DDIF\$\_IDU\_BITS\_PER\_PIXEL**

**Encoding: integer**

An optional plane-bits-per-pixel item that indicates the total number of bits per pixel. This value also represents the sum of the number of bits per component for all components. For bitonal images, the plane-bits-per-pixel item always has a value of 1, and is therefore omitted.

### **DDIF\$\_IDU\_PLANE\_DATA**

**Encoding: string**

A plane data item that specifies the actual data.

---

## DDIF\$\_IMG—Image Content

The image content aggregate represents image data. The DDIF\$\_IMG aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE, DDIF\$\_PGL\_CONTENT, and DDIF\$\_SEG\_CONTENT.

For image frames, the bounding box is the physical size of the image contained in the frame. The physical size is used to calculate the image resolution, along with the height and width of the image in pixels, and the pixel shape (aspect ratio).

Refer to these corresponding syntax diagrams:

| Syntax           | Location    |
|------------------|-------------|
| ImagePrimitive   | Figure B-34 |
| ImageCodingAttrs | Figure B-35 |

---

### AGGREGATE FORMAT

| Item Name          | Item Encoding                     |
|--------------------|-----------------------------------|
| DDIF\$_IMG_CONTENT | Sequence of DDIF\$_IDU aggregates |

---

### AGGREGATE ITEMS

#### ***DDIF\$\_IMG\_CONTENT***

#### ***Encoding: sequence of DDIF\$\_IDU aggregates***

An image content item that specifies the content of the image. For more information, see the description of the DDIF\$\_IDU aggregate.

For image frames, the bounding box is the physical size of the image contained in the frame. The physical size is used to calculate the image resolution, along with the height and width of the image in pixels and the pixel shape (aspect ratio).

It is important to note that the bounding box items of the frame attributes must be respecified in the segment attributes aggregate (type DDIF\$\_SGA) associated with image content; frame attributes for image content are not inherited from a type definition.

## DDIF\$\_LG1

---

### DDIF\$\_LG1—Generic Layout

The generic layout aggregate specifies a set of page descriptions along with rules about when to use a particular page description. It also enables you to describe a set of content descriptions that can be referenced from generic and/or specific pages to form content that appears on one or more pages. The DDIF\$\_LG1 aggregate is referenced by the parent aggregate item DDIF\$\_SEG\_GENERIC\_LAYOUT.

Refer to these corresponding syntax diagrams:

---

| Syntax          | Location     |
|-----------------|--------------|
| GenericLayout   | Figure B-113 |
| NamedValueList  | Figure B-78  |
| PageDescription | Figure B-114 |

---

---

### AGGREGATE FORMAT

---

| Item Name                    | Item Encoding                     |
|------------------------------|-----------------------------------|
| DDIF\$_LG1_PRIVATE_DATA      | Sequence of DDIF\$_PVT aggregates |
| DDIF\$_LG1_PAGE_DESCRIPTIONS | Sequence of DDIF\$_PGD aggregates |

---

---

### AGGREGATE ITEMS

#### **DDIF\$\_LG1\_PRIVATE\_DATA**

**Encoding: sequence of DDIF\$\_PVT aggregates**

An optional private data item that specifies nonstandard information associated with the generic layout descriptions. For more information, see the description of the DDIF\$\_PVT aggregate. The private data is typically used to associate names or relationships with the page and/or content descriptions.

#### **DDIF\$\_LG1\_PAGE\_DESCRIPTIONS**

**Encoding: sequence of DDIF\$\_PGD aggregates**

A page descriptions item that provides descriptions of actual page templates and rules for their use. For more information, see the description of the DDIF\$\_PGD aggregate.

---

## DDIF\$\_LIN—Polyline Content

The polyline content aggregate represents polylines, polymarkers, and filled areas. The DDIF\$\_LIN aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE, and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax       | Location    |
|--------------|-------------|
| Polyline     | Figure B-24 |
| PolyLinePath | Figure B-88 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                         |
|-------------------------|---------------------------------------|
| DDIF\$_LIN_FLAGS        | Longword                              |
| DDIF\$_LIN_DRAW_PATTERN | Bit string                            |
| DDIF\$_LIN_PATH_C       | Array of type measurement enumeration |
| DDIF\$_LIN_PATH         | Array of type variable                |

---

### AGGREGATE ITEMS

#### **DDIF\$\_LIN\_FLAGS**

##### **Encoding: longword**

A flags item that is used to control the rendering of the polyline. Valid values for this item are as follows:

|                             |   |
|-----------------------------|---|
| ddif\$m_lin_draw_polyline   | If set, a line is drawn between the specified points; if clear, no line is drawn. If no flags are specified, this item is set by default. |
| ddif\$m_lin_fill_polyline   | If set, the area defined by the points is filled; if clear, the area is not filled.   |
| ddif\$m_lin_draw_markers    | If set, a marker is placed at each point; if clear, no markers are drawn.   |
| ddif\$m_lin_regular_polygon | If set, the object is a regular polygon.  |
| ddif\$m_lin_close_polyline  | If set, the last point of the object is connected to the first.   |

## DDIF\$\_LIN

ddif\$m\_lin\_rounderpolyline

If set, the line joints of the polyline are rounded.

ddif\$m\_lin\_rectangular\_polygon

If set, the polyline represents a rectangle. The polyline must consist of four points. If all four lines must be drawn, the ddif\$m\_lin\_close\_polyline value must also be specified.

### **DDIF\$\_LIN\_DRAW\_PATTERN**

**Encoding: bit string**

A draw pattern item that determines which line segments are drawn.

Starting from the first bit and the line between the first two points of the object, if the corresponding bit is set, the line is drawn. Otherwise, the line is not drawn, but does limit the fill area.

The number of bits in the draw pattern does not have to match the number of line segments in the polyline. If the draw pattern contains fewer flags than the object contains line segments, the pattern is repeated. For example, a bit pattern of 1 causes every line to be drawn, and a pattern of 0 suppresses all lines. A pattern of 01 causes every other line to be drawn, beginning with the second. The default is "1"B.

A draw pattern can be provided even if the ddif\$m\_lin\_draw\_polyline flag is clear, with the implication that it forms the pattern if the flag is later set.

### **DDIF\$\_LIN\_PATH\_C**

**Encoding: array of type measurement enumeration**

A line path indicator that specifies whether the layout of the polyline is specified as a variable or constant value.

### **DDIF\$\_LIN\_PATH**

**Encoding: array of type variable**

A line path item that lists the control points of the polyline.

The points of the polyline are stored in an array in a repeating **x,y**-pair format. For example, if you are storing values in this item, the first value you specify must be the **x** position of the first control point; the second value must be the **y** position of the first control point, and so on. Because these points are stored in an array, you must increment the aggregate index associated with the array each time you read or write a control point. The initial aggregate index value is 0.

The coordinates of the line are relative to the frame that contains the line.

---

## DDIF\$\_LL1—Layout Attributes

The layout attributes aggregate specifies certain layout attributes. The DDIF\$\_LL1 aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_LAYGLY\_LAYOUT.

Refer to these corresponding syntax diagrams:

| Syntax           | Location     |
|------------------|--------------|
| LayoutAttributes | Figure B–122 |
| Directive        | Figure B–20  |
| BreakCriteria    | Figure B–123 |
| Escapement       | Figure B–47  |
| TabStopList      | Figure B–126 |

---

### AGGREGATE FORMAT

| Item Name                     | Item Encoding                     |
|-------------------------------|-----------------------------------|
| DDIF\$_LL1_INITIAL_DIRECTIVE  | Enumeration                       |
| DDIF\$_LL1_GALLEY_SELECT      | String                            |
| DDIF\$_LL1_BREAK_BEFORE       | Enumeration                       |
| DDIF\$_LL1_BREAK_WITHIN       | Enumeration                       |
| DDIF\$_LL1_BREAK_AFTER        | Enumeration                       |
| DDIF\$_LL1_INITIAL_INDENT_C   | Measurement enumeration           |
| DDIF\$_LL1_INITIAL_INDENT     | Variable                          |
| DDIF\$_LL1_LEFT_INDENT_C      | Measurement enumeration           |
| DDIF\$_LL1_LEFT_INDENT        | Variable                          |
| DDIF\$_LL1_RIGHT_INDENT_C     | Measurement enumeration           |
| DDIF\$_LL1_RIGHT_INDENT       | Variable                          |
| DDIF\$_LL1_SPACE_BEFORE_C     | Measurement enumeration           |
| DDIF\$_LL1_SPACE_BEFORE       | Variable                          |
| DDIF\$_LL1_SPACE_AFTER_C      | Measurement enumeration           |
| DDIF\$_LL1_SPACE_AFTER        | Variable                          |
| DDIF\$_LL1_LEADING_RATIO_N    | Integer                           |
| DDIF\$_LL1_LEADING_RATIO_D    | Integer                           |
| DDIF\$_LL1_LEADING_CONSTANT_C | Measurement enumeration           |
| DDIF\$_LL1_LEADING_CONSTANT   | Variable                          |
| DDIF\$_LL1_TAB_STOPS          | Sequence of DDIF\$_TBS aggregates |

---

## DDIF\$\_LL1

---

### AGGREGATE ITEMS

#### **DDIF\$\_LL1\_INITIAL\_DIRECTIVE**

**Encoding: enumeration; valid values are as follows:**

|                               |  |
|-------------------------------|--|
| DDIF\$_K_DIR_NEW_PAGE         | Begins a new page.   |
| DDIF\$_K_DIR_NEW_LINE         | Begins a new line of text.   |
| DDIF\$_K_DIR_NEW_GALLEY       | Begins a new layout galley (such as a column). Software that does not support galley layout interprets the new galley directive as a new page.                                   |
| DDIF\$_K_DIR_TAB              | Moves the horizontal text position to the next tab stop.   |
| DDIF\$_K_DIR_SPACE            | Specifies a space in the current font. The space directive is usually soft, and is used to indicate that software inserted a space between wrapped lines.                        |
| DDIF\$_K_DIR_HYPHEN_NEW_LINE  | Specifies that the line break is preceded by a hyphen. This directive is typically soft, and is used to indicate that software inserted a hyphen at the place it broke the line. |
| DDIF\$_K_DIR_WORD_BREAK_POINT | Identifies an embedded point at which a word may be broken, if need be, for justification.   |
| DDIF\$_K_DIR_LEADERS          | Inserts leader characters according to the current leader attributes.  |
| DDIF\$_K_DIR_BACKSPACE        | Specifies that the first character following this directive should be centered over the last character imaged.   |
| DDIF\$_K_NULL                 | Suppresses the inheritance of the initial-directive element of layout attributes. This directive has no effect on imaging or processing.   |
| DDIF\$_K_DIR_NO_HYPHEN_WORD   | Suppresses hyphenation until the next space character or space directive is encountered.   |

An optional initial directive item that forces a new line, galley, or page by means of a directive.

#### **DDIF\$\_LL1\_GALLEY\_SELECT**

**Encoding: string**

An optional galley selection item that forces the selection of a new layout galley by name. This item references the DDIF\$\_GLY\_ID item.

#### **DDIF\$\_LL1\_BREAK\_BEFORE**

**Encoding: enumeration; valid values are as follows:**

|                          |  |
|--------------------------|--|
| DDIF\$_K_BREAK_ALWAYS    | Always break to a new galley or page.                              |
| DDIF\$_K_BREAK_NEVER     | Never break to a new galley or page.                               |
| DDIF\$_K_BREAK_IF_NEEDED | The formatter can break to a new galley or page at its discretion. |

An optional pre-segment break condition item that specifies the condition on which a break occurs before the segment.



**DDIF\$\_LL1\_BREAK\_WITHIN****Encoding: enumeration; valid values are as follows:**

|                          |  |
|--------------------------|--|
| DDIF\$_K_BREAK_ALWAYS    | Always break to a new galley or page.                              |
| DDIF\$_K_BREAK_NEVER     | Never break to a new galley or page.                               |
| DDIF\$_K_BREAK_IF_NEEDED | The formatter can break to a new galley or page at its discretion. |

An optional in-segment break condition item that specifies the condition on which a break occurs within a segment.

**DDIF\$\_LL1\_BREAK\_AFTER****Encoding: enumeration; valid values are as follows:**

|                          |  |
|--------------------------|--|
| DDIF\$_K_BREAK_ALWAYS    | Always break to a new galley or page.                              |
| DDIF\$_K_BREAK_NEVER     | Never break to a new galley or page.                               |
| DDIF\$_K_BREAK_IF_NEEDED | The formatter can break to a new galley or page at its discretion. |

An optional post-segment break condition item that specifies the condition on which a break occurs after the segment.

**DDIF\$\_LL1\_INITIAL\_INDENT\_C****Encoding: measurement enumeration**

An optional initial indent indicator that specifies whether the initial indent value is specified as a variable or constant value.

**DDIF\$\_LL1\_INITIAL\_INDENT****Encoding: variable**

An optional initial indent item that specifies the distance added to the current left indent to determine the minimum distance between the start of the path and the left alignment point of the first character in the text layout path. The current left indent is the new left indent created by an associated DDIF\$\_LL1\_LEFT\_INDENT item (if any). That is, the initial indent item can be a positive or negative value relative to the new left indent. The initial value is 0.

**DDIF\$\_LL1\_LEFT\_INDENT\_C****Encoding: measurement enumeration**

An optional left indent indicator that indicates whether the left indent value is specified as a variable or constant value.

**DDIF\$\_LL1\_LEFT\_INDENT****Encoding: variable**

An optional left indent item that specifies the distance added to the current left indent to create a new left indent, which determines the minimum distance between the start of the text layout path and the left alignment position of the first character on every wrapped line. If no initial indent is specified, the left indent is used for the initial indent. The initial value of the left indent is 0. Note that the left indent inherited by a segment is the sum of the left indents specified by its parent segments.

**DDIF\$\_LL1\_RIGHT\_INDENT\_C****Encoding: measurement enumeration**

An optional right indent indicator that indicates whether the right indent value is specified as a variable or constant value.

## DDIF\$\_LL1

### **DDIF\$\_LL1\_RIGHT\_INDENT**

**Encoding: variable**

An optional right indent item that specifies the distance added to the current right indent to determine the new right indent, which is the minimum distance between the end of the text path and the last character imaged along the path. The initial value of the right indent is 0. Note that the right indent inherited by a segment is the sum of the right indents specified by its parent segments.

### **DDIF\$\_LL1\_SPACE\_BEFORE\_C**

**Encoding: measurement enumeration**

A space-before indicator that indicates whether the space-before value is specified as a variable or constant value.

### **DDIF\$\_LL1\_SPACE\_BEFORE**

**Encoding: variable**

A space-before item that specifies the amount of space before the segment. This item has a default value of 0.

### **DDIF\$\_LL1\_SPACE\_AFTER\_C**

**Encoding: measurement enumeration**

A space-after indicator that indicates whether the space-after value is specified as a variable or constant value.

### **DDIF\$\_LL1\_SPACE\_AFTER**

**Encoding: variable**

A space-after item that specifies the amount of space after the segment. This item has a default value of 0.

### **DDIF\$\_LL1\_LEADING\_RATIO\_N**

**Encoding: integer**

An optional leading ratio numerator item that specifies the magnitude of the escapement ratio to be used to increment or decrement the interline spacing in layout. This ratio specifies the proportion of the normal line spacing used as “additional” line spacing. For example, a leading ratio of 1:1 doubles the total line spacing, and 2:1 triples it.

### **DDIF\$\_LL1\_LEADING\_RATIO\_D**

**Encoding: integer**

An optional leading ratio denominator item that specifies the units of precision used in the escapement ratio that is used to increment or decrement the interline spacing in layout.

### **DDIF\$\_LL1\_LEADING\_CONSTANT\_C**

**Encoding: measurement enumeration**

An optional leading constant indicator that indicates whether the interline spacing value is specified as a variable or constant value.

### **DDIF\$\_LL1\_LEADING\_CONSTANT**

**Encoding: variable**

An optional leading constant item that specifies the interline spacing value in the current measurement units.

***DDIF\$\_LL1\_TAB\_STOPS******Encoding: sequence of DDIF\$\_TBS aggregates***

An optional tab stops item that specifies a sequence of fields along the current text path that cause text between tab directives to become aligned within the fields. For more information, see the description of the DDIF\$\_TBS aggregate.

## DDIF\$\_LS1

---

### DDIF\$\_LS1—Specific Layout

The specific layout aggregate contains one page description for each page of the document, although pages that have identical layout can share a description for the sake of representational efficiency.

A document that contains specific layout can also have a generic layout specification, which is used to add new pages to the document. Specific page layouts can be derived from a generic layout, they can be manually generated, or they can be user-modified versions of layouts derived from generic layouts.

The specific layout of a document is represented as a list of page descriptions, or references to page descriptions that have been previously declared. The first specific page description is by default the first page, but you can override this by making selections within the content stream.

The DDIF\$\_LS1 aggregate is referenced by the parent aggregate item DDIF\$\_SEG\_SPECIFIC\_LAYOUT.

Refer to these corresponding syntax diagrams:

---

| Syntax          | Location     |
|-----------------|--------------|
| SpecificLayout  | Figure B-120 |
| PageDescription | Figure B-114 |

---

---

### AGGREGATE FORMAT

---

| Item Name           | Item Encoding             |
|---------------------|---------------------------|
| DDIF\$_LS1_LAYOUT_C | Array of type enumeration |
| DDIF\$_LS1_LAYOUT   | Array of type variable    |

---

---

### AGGREGATE ITEMS

#### **DDIF\$\_LS1\_LAYOUT\_C**

**Encoding:** array of type enumeration; valid values are as follows:

**DDIF\$K\_SPECIFIC\_PAGE** Indicates that the layout specified is the description of a specific page. In this case, the DDIF\$\_LS1\_LAYOUT item is encoded as the handle of a DDIF\$\_PGD aggregate.

**DDIF\$K\_REFERENCED\_PAGE** Indicates that the layout specified is actually the label of a page layout description previously defined. In this case, the DDIF\$\_LS1\_LAYOUT item is encoded as a string.

A layout indicator that indicates whether the layout is for a specific page or is a reference to a previously defined page.

**DDIF\$\_LS1\_LAYOUT****Encoding: array of type variable**

A layout item that defines the specific layout. Note that the array items in each array must correspond. For example, if the first value in the layout indicator array specifies a referenced page, the first value in the layout array must contain a string specifying the label of the page layout description being referenced, and so on. This item references the DDIF\$\_PGD\_LABEL item.

## DDIF\$\_LSD

---

### DDIF\$\_LSD—Line-Style Definition

The line-style definition aggregate models the description of a line-style pattern for reference within the assigned scope. The DDIF\$\_LSD aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_LINE\_STYLE\_DEFNS.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| LineDefn       | Figure B-87 |
| NamedValueList | Figure B-78 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DDIF\$_LSD_NUMBER       | Integer                           |
| DDIF\$_LSD_PATTERN      | Array of type integer             |
| DDIF\$_LSD_PRIVATE_DATA | Sequence of DDIF\$_PVT aggregates |

---

### AGGREGATE ITEMS

#### **DDIF\$\_LSD\_NUMBER**

##### **Encoding: integer**

A line-style number that specifies a number by which the defined line style is referenced from within the scope of the definition. This item is referenced by the DDIF\$\_SGA\_LIN\_STYLE item.

#### **DDIF\$\_LSD\_PATTERN**

##### **Encoding: array of type integer**

A line-style pattern item that specifies the line-style pattern being defined.

Each integer in the array is used to determine the relative length of the on portions and off portions of the line. The first integer corresponds to the start of the line and an on portion. The next integer corresponds to an off portion of the line, and so on. The actual length of each line portion is determined by multiplying the integer by the line pattern size attribute (DDIF\$\_SGA\_LIN\_PATTERN\_SIZE). If the length of the line exceeds the number of specified integers, the array is reused in a cyclical fashion.

For example, a dotted line could be specified with the following integer array:

1

This means one unit on, then (by reusing the array) one unit off, and so on.

A dashed line could be the following:

2  
1

This means two units on, then one unit off, and so on.

A solid line is specified by omitting this aggregate item.

If your pattern must be defined using more than 32 bits, you must use additional longwords in an array to specify the pattern. For each longword specified, you must increment the aggregate index by 1. The initial value of the aggregate index is 0.

***DDIF\$\_LSD\_PRIVATE\_DATA***

***Encoding: sequence of DDIF\$\_PVT aggregates***

An optional line-style private data item that specifies the private data associated with the definition. For more information, see the description of the DDIF\$\_PVT aggregate.

## DDIF\$\_LW1

---

### DDIF\$\_LW1—Wrap Attributes

The wrap attributes aggregate specifies attributes that control the wrapping of text in a document. The DDIF\$\_LW1 aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_LAYGLY\_WRAP.

Refer to these corresponding syntax diagrams:

| Syntax         | Location     |
|----------------|--------------|
| WrapAttributes | Figure B-121 |
| Format         | Figure B-50  |

---

### AGGREGATE FORMAT

| Item Name                      | Item Encoding |
|--------------------------------|---------------|
| DDIF\$_LW1_WRAP_FORMAT         | Enumeration   |
| DDIF\$_LW1_QUAD_FORMAT         | Enumeration   |
| DDIF\$_LW1_HYPHENATION_FLAGS   | Longword      |
| DDIF\$_LW1_MAXIMUM_HYPH_LINES  | Integer       |
| DDIF\$_LW1_MAXIMUM_ORPHAN_SIZE | Integer       |
| DDIF\$_LW1_MAXIMUM_WIDOW_SIZE  | Integer       |

---

### AGGREGATE ITEMS

#### **DDIF\$\_LW1\_WRAP\_FORMAT**

**Encoding: enumeration; valid values are as follows:**

|                              |   |
|------------------------------|---|
| DDIF\$K_FMT_FLUSH_PATH_BEGIN | The first character is imaged at the start of the text path, and successive characters are imaged at successive positions determined by the escapement of the characters imaged.  |
| DDIF\$K_FMT_CENTER_OF_PATH   | The length of text strings, as given by the sum of the character escapements, is subtracted from the length of the path, and the remaining space is evenly distributed between the first character and the start of the path, and the last character and the end of the path. |
| DDIF\$K_FMT_FLUSH_PATH_END   | The text string is imaged such that the right alignment point of the last character is aligned with the end of the text string when normal escapement is applied.   |



DDIF\$K\_FMT\_FLUSH\_PATH\_BOTH

The text string is imaged such that the left alignment point of the first character is aligned with the start of the text path, and the right alignment point of the last character is aligned with the end of the path.

An optional wrap format item that specifies the format of text lines wrapped by the formatter.

**DDIF\$\_LW1\_QUAD\_FORMAT**

**Encoding: enumeration; valid values are as follows:**

DDIF\$K\_FMT\_FLUSH\_PATH\_BEGIN

The first character is imaged at the start of the text path, and successive characters are imaged at successive positions determined by the escapement of the characters imaged.

DDIF\$K\_FMT\_CENTER\_OF\_PATH

The length of text strings, as given by the sum of the character escapements, is subtracted from the length of the path, and the remaining space is evenly distributed between the first character and the start of the path, and the last character and the end of the path.

DDIF\$K\_FMT\_FLUSH\_PATH\_END

The text string is imaged such that the right alignment point of the last character is aligned with the end of the text string when normal escapement is applied.

DDIF\$K\_FMT\_FLUSH\_PATH\_BOTH

The text string is imaged such that the left alignment point of the first character is aligned with the start of the text path, and the right alignment point of the last character is aligned with the end of the path.

An optional quad format item that specifies the format of text lines that end in a hard (user-entered) new line.

**DDIF\$\_LW1\_HYPHENATION\_FLAGS**

**Encoding: longword**

An optional hyphenation flags item that specifies the Boolean parameters that affect hyphenation. The possible flag values are as follows:

ddif\$m\_hyph\_allowed

If set, hyphenation is allowed in this segment.

ddif\$m\_hyph\_paragraph\_end

If set, the last line in the paragraph can end in a hyphen.

ddif\$m\_hyph\_galley\_end

If set, hyphenation is allowed at the end of a galley.

ddif\$m\_hyph\_page\_end

If set, words can be hyphenated across pages.

ddif\$m\_hyph\_capitalized\_word

If set, capitalized words can be hyphenated.

## DDIF\$\_LW1

### ***DDIF\$\_LW1\_MAXIMUM\_HYPH\_LINES***

***Encoding: integer***

An optional maximum hyphenation lines item that specifies the maximum number of consecutive lines that can end with a hyphen.

### ***DDIF\$\_LW1\_MAXIMUM\_ORPHAN\_SIZE***

***Encoding: integer***

An optional maximum orphan size that specifies the maximum orphan size. This value specifies the maximum number of lines of text within the segment that can be left at the bottom of the galley if the rest of the lines are on the succeeding galley.

The default maximum value is 3. However, you can override the default with a setting of 1, which allows the paragraph break to occur between any two lines of the paragraph. You can therefore allow a single line of the paragraph to occur in a particular galley. Specifying 0 does not add any additional level of control.

### ***DDIF\$\_LW1\_MAXIMUM\_WIDOW\_SIZE***

***Encoding: integer***

An optional maximum widow size that specifies the maximum widow size. This value specifies the maximum number of lines of text within the segment that can be placed in the succeeding galley when the first line or lines are in the current galley.

The default maximum value is 3. However, you can override the default with a setting of 1, which allows the paragraph break to occur between any two lines of the paragraph. You can therefore allow a single line of the paragraph to occur in a particular galley. Specifying 0 does not add any additional level of control.

---

## DDIF\$\_OCC—Occurrence Definition

The occurrence definition aggregate describes the number of times the element of a structure definition can occur, and whether or not it can be omitted. The DDIF\$\_OCC aggregate is referenced by the parent aggregate items DDIF\$\_OCC\_STRUCTURE\_ELEMENT and DDIF\$\_SGA\_STRUCTURE\_DESC.

Refer to these corresponding syntax diagrams:

| Syntax           | Location    |
|------------------|-------------|
| StructureDefn    | Figure B-94 |
| OccurrenceDefn   | Figure B-95 |
| StructureElement | Figure B-96 |

---

### AGGREGATE FORMAT

| Item Name                      | Item Encoding |
|--------------------------------|---------------|
| DDIF\$_OCC_OCCURRENCE_C        | Enumeration   |
| DDIF\$_OCC_STRUCTURE_ELEMENT_C | Enumeration   |
| DDIF\$_OCC_STRUCTURE_ELEMENT   | Variable      |

---

### AGGREGATE ITEMS

#### **DDIF\$\_OCC\_OCCURRENCE\_C**

**Encoding: enumeration; valid values are as follows:**

|                             |   |
|-----------------------------|---|
| DDIF\$K_REQUIRED_OCCURRENCE | The construction must occur once and only once. |
| DDIF\$K_OPTIONAL_OCCURRENCE | The construction can occur once or not at all.  |
| DDIF\$K_REPEAT_OCCURRENCE   | The construction can occur one or more times.   |
| DDIF\$K_OPT_RPT_OCCURRENCE  | The construction can occur zero or more times.  |

An occurrence indicator that specifies the type of occurrence to be permitted. There is no default or initial value for this aggregate item.

## DDIF\$\_OCC

### **DDIF\$\_OCC\_STRUCTURE\_ELEMENT\_C**

**Encoding: enumeration; valid values are as follows:**

|                            |  |
|----------------------------|--|
| DDIF\$K_SEQUENCE_STRUCTURE | Indicates a sequence of element occurrences that are constrained to occur in the order specified. In this case, the DDIF\$_OCC_STRUCTURE_ELEMENT item is encoded as a sequence of DDIF\$_OCC aggregates. |
| DDIF\$K_SET_STRUCTURE      | Indicates a set of element occurrences that are <i>not</i> constrained with respect to order. In this case, the DDIF\$_OCC_STRUCTURE_ELEMENT item is encoded as a sequence of DDIF\$_OCC aggregates.     |
| DDIF\$K_CHOICE_STRUCTURE   | Indicates a group of element occurrences from which only one can be selected. In this case, the DDIF\$_OCC_STRUCTURE_ELEMENT item is encoded as a sequence of DDIF\$_OCC aggregates.                     |
| DDIF\$K_REFERENCED_TYPE    | Indicates the label assigned to the type reference whose occurrence in the document structure is being constrained. In this case, the DDIF\$_OCC_STRUCTURE_ELEMENT item is encoded as a string.          |

A structure element indicator that indicates whether a given element in the structure definition is the label of the referenced type or is a structure definition that is itself a defined substructure.

### **DDIF\$\_OCC\_STRUCTURE\_ELEMENT**

**Encoding: variable**

A structure item that specifies the structure itself. This item references the DDIF\$\_TYD\_LABEL item.

---

## DDIF\$\_PGD—Page Description

The page description aggregate describes a page either as a single page layout or as a set of page layouts with conditions under which the different page layouts are used. A page layout is used when one of the galleys on the page is given text content. Galleys are connected to form a chain of successors used to format a flow of text. As each galley is invoked, the page on which it is described is invoked. The DDIF\$\_PGD aggregate is referenced by the parent aggregate items DDIF\$\_LG1\_PAGE\_DESCRIPTIONS and DDIF\$\_LS1\_LAYOUT.

Refer to these corresponding syntax diagrams:

| Syntax          | Location     |
|-----------------|--------------|
| PageDescription | Figure B-114 |
| NamedValueList  | Figure B-78  |
| PageLayout      | Figure B-116 |
| PageSet         | Figure B-115 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DDIF\$_PGD_LABEL        | String                            |
| DDIF\$_PGD_PRIVATE_DATA | Sequence of DDIF\$_PVT aggregates |
| DDIF\$_PGD_DESC_C       | Enumeration                       |
| DDIF\$_PGD_DESC         | Variable                          |

---

### AGGREGATE ITEMS

#### **DDIF\$\_PGD\_LABEL**

**Encoding:** *string*

A page description label item that specifies the label by which the page description is referenced. This item is referenced by the DDIF\$\_LS1\_LAYOUT item and by the DDIF\$\_PGL\_PROTOTYPE item.

#### **DDIF\$\_PGD\_PRIVATE\_DATA**

**Encoding:** *sequence of DDIF\$\_PVT aggregates*

An optional private data item that allows for the inclusion of application-private data. For more information, see the description of the DDIF\$\_PVT aggregate.

## DDIF\$\_PGD

### **DDIF\$\_PGD\_DESC\_C**

**Encoding: enumeration; valid values are as follows:**

DDIF\$K\_PAGE\_SET\_DESC      A description of a set of page layouts, one of which is chosen based on the criteria presented in the page set. In this case, the DDIF\$\_PGD\_DESC item is encoded as a sequence of DDIF\$\_PGS aggregates.

DDIF\$K\_PAGE\_LAYOUT      A page layout description defined for reference from content or from page set descriptions. In this case, the DDIF\$\_PGD\_DESC item is encoded as the handle of a DDIF\$\_PGL aggregate.

A page description indicator that indicates whether the page description is actually a set of page layouts or is a page layout defined for reference.

### **DDIF\$\_PGD\_DESC**

**Encoding: variable**

A page description item that specifies the actual page description to be used.

---

## DDIF\$\_PGL—Page Layout

The page layout aggregate describes a page, including its size, the galleys on the page (defined by the DDIF\$\_GLY aggregate), and any content specific to that particular page. The same page layout is shared by generic and specific layout. The DDIF\$\_PGL aggregate is referenced by the parent aggregate items DDIF\$\_PGD\_DESC and DDIF\$\_PGS\_SELECT\_PAGE\_LAYOUT.

Refer to these corresponding syntax diagrams:

| Syntax     | Location     |
|------------|--------------|
| PageLayout | Figure B-116 |
| GenMeasure | Figure B-124 |
| GenSize    | Figure B-125 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                    |
|-------------------------|----------------------------------|
| DDIF\$_PGL_LAYOUT_ID    | String                           |
| DDIF\$_PGL_SIZE_X_NOM_C | Measurement enumeration          |
| DDIF\$_PGL_SIZE_X_NOM   | Variable                         |
| DDIF\$_PGL_SIZE_X_STR_C | Measurement enumeration          |
| DDIF\$_PGL_SIZE_X_STR   | Variable                         |
| DDIF\$_PGL_SIZE_X_SHR_C | Measurement enumeration          |
| DDIF\$_PGL_SIZE_X_SHR   | Variable                         |
| DDIF\$_PGL_SIZE_Y_NOM_C | Measurement enumeration          |
| DDIF\$_PGL_SIZE_Y_NOM   | Variable                         |
| DDIF\$_PGL_SIZE_Y_STR_C | Measurement enumeration          |
| DDIF\$_PGL_SIZE_Y_STR   | Variable                         |
| DDIF\$_PGL_SIZE_Y_SHR_C | Measurement enumeration          |
| DDIF\$_PGL_SIZE_Y_SHR   | Variable                         |
| DDIF\$_PGL_ORIENTATION  | Enumeration                      |
| DDIF\$_PGL_PROTOTYPE    | String                           |
| DDIF\$_PGL_CONTENT      | Handle of a DDIF\$_SEG aggregate |

# DDIF\$\_PGL

---

## AGGREGATE ITEMS

### ***DDIF\$\_PGL\_LAYOUT\_ID***

**Encoding:** *string*

A page layout identifier item that specifies a label used to reference the page layout. This item is referenced by the DDIF\$\_PGS\_SELECT\_PAGE\_LAYOUT item.

### ***DDIF\$\_PGL\_SIZE\_X\_NOM\_C***

**Encoding:** *measurement enumeration*

A page size nominal measure indicator that indicates whether the nominal **x** measurement is specified as a variable or constant value.

### ***DDIF\$\_PGL\_SIZE\_X\_NOM***

**Encoding:** *variable*

A page size nominal measure item that specifies the nominal **x** measurement. The default value for this item is 0.

### ***DDIF\$\_PGL\_SIZE\_X\_STR\_C***

**Encoding:** *measurement enumeration*

A page size **x** stretch indicator that indicates whether the **x** stretch amount is specified as a variable or constant value.

### ***DDIF\$\_PGL\_SIZE\_X\_STR***

**Encoding:** *variable*

A page size **x** stretch item that specifies the amount by which the **x** measurement can be extended. The default value for this item is 0.

### ***DDIF\$\_PGL\_SIZE\_X\_SHR\_C***

**Encoding:** *measurement enumeration*

A page size **x** shrink indicator that indicates whether the **x** shrink amount is specified as a variable or constant value.

### ***DDIF\$\_PGL\_SIZE\_X\_SHR***

**Encoding:** *variable*

A page size **x** shrink item that specifies the amount by which the **x** measurement can be contracted. The default value for this item is 0.

### ***DDIF\$\_PGL\_SIZE\_Y\_NOM\_C***

**Encoding:** *measurement enumeration*

A page size nominal measure indicator that indicates whether the nominal **y** measurement is specified as a variable or constant value.

### ***DDIF\$\_PGL\_SIZE\_Y\_NOM***

**Encoding:** *variable*

A page size nominal measure item that specifies the nominal **y** measurement. The default value for this item is 0.

### ***DDIF\$\_PGL\_SIZE\_Y\_STR\_C***

**Encoding:** *measurement enumeration*

A page size **y** stretch indicator that indicates whether the **y** stretch amount is specified as a variable or constant value.



**DDIF\$\_PGL\_SIZE\_Y\_STR****Encoding: variable**

A page size y stretch item that specifies the amount by which the y measurement can be extended. The default value for this item is 0.

**DDIF\$\_PGL\_SIZE\_Y\_SHR\_C****Encoding: measurement enumeration**

A page size y shrink indicator that indicates whether the y shrink amount is specified as a variable or constant value.

**DDIF\$\_PGL\_SIZE\_Y\_SHR****Encoding: variable**

A page size y shrink item that specifies the amount by which the y measurement can be contracted. The default value for this item is 0.

**DDIF\$\_PGL\_ORIENTATION****Encoding: enumeration; valid values are as follows:**

|                           |  |
|---------------------------|--|
| DDIF\$_K_PORTRAIT_ORIENT  | Portrait orientation puts the y axis along the height of the page. |
| DDIF\$_K_LANDSCAPE_ORIENT | Landscape orientation puts the y axis along the width of the page. |

A page orientation item that defines the orientation of the page relative to the height and width.

**DDIF\$\_PGL\_PROTOTYPE****Encoding: string**

An optional page prototype item that specifies the label of the generic page description from which the layout being defined was derived. Any objects other than galleys in the page frame of the prototype definition are imaged in the new page layout. This item references the DDIF\$\_PGD\_LABEL item.

**DDIF\$\_PGL\_CONTENT****Encoding: handle of a DDIF\$\_SEG aggregate**

An optional page content item that must represent a frame whose origin is located at the lower lefthand corner of the page.

The DDIF\$\_PGL\_CONTENT item contains the handle of a DDIF\$\_SEG aggregate containing the page content. Page content can reference definitions in the document content, but the document content cannot reference definitions in the page content. All page content coordinates are relative to the page coordinate system, but frames can be nested in the page content.

## DDIF\$\_PGS

---

### DDIF\$\_PGS—Page Select

The page selection aggregate consists of one or more pages, one of which is selected based on the current formatting state. Each page selection aggregate consists of a pointer to a page in the list of page layouts, and the criteria that cause that particular page in the set to be selected. The DDIF\$\_PGS aggregate is referenced by the parent aggregate item DDIF\$\_PGD\_DESC.

Refer to these corresponding syntax diagrams:

| Syntax     | Location     |
|------------|--------------|
| PageSet    | Figure B-115 |
| PageLayout | Figure B-116 |

---

### AGGREGATE FORMAT

| Item Name                       | Item Encoding |
|---------------------------------|---------------|
| DDIF\$_PGS_PAGE_SIDE_CRITERIA   | Enumeration   |
| DDIF\$_PGS_SELECT_PAGE_LAYOUT_C | Enumeration   |
| DDIF\$_PGS_SELECT_PAGE_LAYOUT   | Variable      |

---

### AGGREGATE ITEMS

#### **DDIF\$\_PGS\_SIDE\_CRITERIA**

**Encoding: enumeration; valid values are as follows:**

|                     |   |
|---------------------|---|
| DDIF\$K_LEFT_PAGE   | Used for left-hand pages when two pages are side by side. A page set that contains a left page must also contain a right page, and cannot contain a page specified as either page.  |
| DDIF\$K_RIGHT_PAGE  | Used for right-hand pages when two pages are side by side. A page set that contains a right page must also contain a left page, and cannot contain a page specified as either page. |
| DDIF\$K_EITHER_PAGE | The same page description is used for either left or right pages.   |

A page-side criteria item that specifies the criteria for the side of the page that must be satisfied to use this page layout description. The default is DDIF\$K\_EITHER\_PAGE.

**DDIF\$\_PGS\_SELECT\_PAGE\_LAYOUT\_C**

**Encoding: enumeration; valid values are as follows:**

|                         |  |
|-------------------------|--|
| DDIF\$K_SELECT_BY_LABEL | Selects a page layout by specifying the label. In this case, the DDIF\$_PGS_SELECT_PAGE_LAYOUT item is encoded as a string.                                  |
| DDIF\$K_SELECT_BY_DEFN  | Selects a page layout by specifying its definition. In this case, the DDIF\$_PGS_SELECT_PAGE_LAYOUT item is encoded as the handle of a DDIF\$_PGL aggregate. |

A select page layout indicator that indicates whether the selected page layout is specified by label or by definition.

**DDIF\$\_PGS\_SELECT\_PAGE\_LAYOUT**

**Encoding: variable**

A select page layout item that specifies the selected page layout. This item references the DDIF\$\_PGL\_LAYOUT ID item.

## DDIF\$\_PHD

---

### DDIF\$\_PHD—Path Definition

The path definition aggregate models the description of a composite path for reference within the assigned scope. The DDIF\$\_PHD aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_PATH\_DEFNS.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| PathDefn       | Figure B-83 |
| CompositePath  | Figure B-84 |
| NamedValueList | Figure B-78 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DDIF\$_PHD_NUMBER       | Integer                           |
| DDIF\$_PHD_DESCRIPTION  | Sequence of DDIF\$_PTH aggregates |
| DDIF\$_PHD_PRIVATE_DATA | Sequence of DDIF\$_PVT aggregates |

---

### AGGREGATE ITEMS

#### **DDIF\$\_PHD\_NUMBER**

**Encoding:** *integer*

A path number item that specifies a number by which the defined path is referenced from within the scope of the definition. This item is referenced by the DDIF\$\_PTH\_REFERENCE item.

#### **DDIF\$\_PHD\_DESCRIPTION**

**Encoding:** *sequence of DDIF\$\_PTH aggregates*

A path description item that specifies the composite path being defined. For more information, see the description of the DDIF\$\_PTH aggregate.

#### **DDIF\$\_PHD\_PRIVATE\_DATA**

**Encoding:** *sequence of DDIF\$\_PVT aggregates*

An optional path private data item that specifies the private data associated with the definition. For more information, see the description of the DDIF\$\_PVT aggregate.

---

## DDIF\$\_PTD—Pattern Definition

The pattern definition aggregate models the description of any type of pattern. The DDIF\$\_PTD aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_PATTERN\_DEFNS.

Refer to these corresponding syntax diagrams:

| Syntax          | Location    |
|-----------------|-------------|
| Color           | Figure B-43 |
| RGB             | Figure B-44 |
| PatternDefn     | Figure B-89 |
| StandardPattern | Figure B-90 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DDIF\$_PTD_NUMBER       | Integer                           |
| DDIF\$_PTD_DEFN_C       | Enumeration                       |
| DDIF\$_PTD_SOL_COLOR_C  | Enumeration                       |
| DDIF\$_PTD_SOL_COLOR_R  | Single-precision floating-point   |
| DDIF\$_PTD_SOL_COLOR_G  | Single-precision floating-point   |
| DDIF\$_PTD_SOL_COLOR_B  | Single-precision floating-point   |
| DDIF\$_PTD_PAT_NUMBER   | Integer                           |
| DDIF\$_PTD_PAT_COLORS   | Array of type integer             |
| DDIF\$_PTD_RAS_PATTERN  | Handle of DDIF\$_IDU aggregate    |
| DDIF\$_PTD_PRIVATE_DATA | Sequence of DDIF\$_PVT aggregates |

---

### AGGREGATE ITEMS

#### **DDIF\$\_PTD\_NUMBER**

**Encoding:** *integer*

A pattern number item that specifies a number by which the pattern is referenced. Any integer may be used as a pattern number. However, the numbers 0 through 63 have predefined values, which are described in Appendix A. These predefined patterns may be overwritten with new definitions, if desired.

## DDIF\$\_PTD

### **DDIF\$\_PTD\_DEFN\_C**

**Encoding: enumeration; valid values are as follows:**

|                          |   |
|--------------------------|---|
| DDIF\$K_SOLID_COLOR      | Indicates a predefined solid fill pattern, assigned a single color. If this value is specified, you must supply values for the item DDIF\$_PTD_SOL_COLOR_C.   |
| DDIF\$K_STANDARD_PATTERN | Indicates a reference to a standard pattern and a color map for it. The color map is defined in terms of previously defined solid patterns. If this value is specified, you must supply values for the items DDIF\$_PTD_PAT_NUMBER and DDIF\$_PTD_PAT_COLORS. |
| DDIF\$K_RASTER_PATTERN   | Indicates an image data unit that represents the pattern. If this value is specified, you must supply a value for the item DDIF\$_PTD_RAS_PATTERN.  |

A pattern definition indicator that selects the definition of the pattern as either a solid color or a standard pattern.

### **DDIF\$\_PTD\_SOL\_COLOR\_C**

**Encoding: enumeration; valid values are as follows:**

|                      |   |
|----------------------|---|
| DDIF\$K_RGB_COLOR    | Indicates that red/green/blue colors are available. If you specify this color type, you must supply values for the items DDIF\$_PTD_SOL_COLOR_R through DDIF\$_PTD_SOL_COLOR_B. |
| DDIF\$K_TRANSPARENCY | Indicates that colors are not available. If you specify this color type, you should not supply any values for the additional background color items.                            |

An optional solid color indicator that must be completed if DDIF\$\_PTD\_DEFN\_C was specified as DDIF\$K\_SOLID\_COLOR.

### **DDIF\$\_PTD\_SOL\_COLOR\_R**

**Encoding: single-precision floating-point**

A red intensity item that indicates the level of red intensity. This value can be in the range of 0.0 to 1.0.

### **DDIF\$\_PTD\_SOL\_COLOR\_G**

**Encoding: single-precision floating-point**

A green intensity item that indicates the level of green intensity. This value can be in the range of 0.0 to 1.0.

### **DDIF\$\_PTD\_SOL\_COLOR\_B**

**Encoding: single-precision floating-point**

A blue intensity item that indicates the level of blue intensity. This value can be in the range of 0.0 to 1.0.

### **DDIF\$\_PTD\_PAT\_NUMBER**

**Encoding: integer**

A standard pattern number item that must be completed if DDIF\$\_PTD\_DEFN\_C was specified as DDIF\$K\_STANDARD\_PATTERN. This item specifies the number of a standard pattern selected from the available patterns. The standard patterns consist of pixel masks. Pixels are imaged in the indicated pattern color, according to the pixel values.

Standard pattern numbers run from 3 to 63. In the list of predefined patterns in Appendix A, patterns numbered from 3 to 63 reference the corresponding standard patterns using black and white as the pattern colors.

**DDIF\$\_PTD\_PAT\_COLORS**

*Encoding: array of type integer*

A pattern colors item that must be completed if DDIF\$\_PTD\_DEFN\_C was specified as DDIF\$K\_STANDARD\_PATTERN. This item specifies a sequence of colors that form the color map for the pattern mask.

The sequence of colors models an array in which the color of each entry maps to the number formed by the corresponding bit pattern in the pattern definition. A single bit-plane pattern mask has two colors, while a two-plane pattern has four. The significance of bits in the bit plane is specified along with the standard pattern definitions.

**DDIF\$\_PTD\_RAS\_PATTERN**

*Encoding: handle of a DDIF\$\_IDU aggregate*

A raster pattern item that must be completed if DDIF\$\_PTD\_DEFN\_C was specified as DDIF\$K\_RASTER\_PATTERN. This item specifies the image data unit that represents the pattern. For more information, see the description of the DDIF\$\_IDU aggregate.

**DDIF\$\_PTD\_PRIVATE\_DATA**

*Encoding: sequence of DDIF\$\_PVT aggregates*

An optional pattern private data item that specifies the private data associated with the definition. For more information, see the description of the DDIF\$\_PVT aggregate.

## DDIF\$\_PTH

---

### DDIF\$\_PTH—Composite Path

A composite path type defines an arbitrary path as a sequence of other path types (polylines, arcs, cubic Béziers, and other composite paths). A composite path is represented as a sequence of DDIF\$\_PTH aggregates. The DDIF\$\_PTH aggregate is referenced by the parent aggregate items DDIF\$\_FAS\_PATH, DDIF\$\_GLY\_OUTLINE, DDIF\$\_PHD\_DESCRIPTION, DDIF\$\_SGA\_FRM\_OUTLINE, and DDIF\$\_SGA\_FRM\_CLIPPING,

Refer to these corresponding syntax diagrams:

| Syntax          | Location    |
|-----------------|-------------|
| AngleRef        | Figure B-67 |
| Measurement     | Figure B-68 |
| XCoordinate     | Figure B-73 |
| YCoordinate     | Figure B-74 |
| PathNumber      | Figure B-81 |
| CompositePath   | Figure B-84 |
| ArcPath         | Figure B-85 |
| CubicBezierPath | Figure B-86 |
| PolyLinePath    | Figure B-88 |

---

### AGGREGATE FORMAT

| Item Name                       | Item Encoding                         |
|---------------------------------|---------------------------------------|
| DDIF\$_PTH_C                    | Enumeration                           |
| DDIF\$_PTH_LIN_PATH_C           | Array of type measurement enumeration |
| DDIF\$_PTH_LIN_PATH             | Array of type variable                |
| DDIF\$_PTH_BEZ_PATH_C           | Array of type measurement enumeration |
| DDIF\$_PTH_BEZ_PATH             | Array of type variable                |
| DDIF\$_PTH_ARC_CENTER_X_C       | Measurement enumeration               |
| DDIF\$_PTH_ARC_CENTER_X         | Variable                              |
| DDIF\$_PTH_ARC_CENTER_Y_C       | Measurement enumeration               |
| DDIF\$_PTH_ARC_CENTER_Y         | Variable                              |
| DDIF\$_PTH_ARC_RADIUS_X_C       | Measurement enumeration               |
| DDIF\$_PTH_ARC_RADIUS_X         | Variable                              |
| DDIF\$_PTH_ARC_RADIUS_DELTA_Y_C | Measurement enumeration               |
| DDIF\$_PTH_ARC_RADIUS_DELTA_Y   | Variable                              |
| DDIF\$_PTH_ARC_START_C          | AngleRef enumeration                  |



| Item Name                 | Item Encoding        |
|---------------------------|----------------------|
| DDIF\$_PTH_ARC_START      | Variable             |
| DDIF\$_PTH_ARC_EXTENT_C   | AngleRef enumeration |
| DDIF\$_PTH_ARC_EXTENT     | Variable             |
| DDIF\$_PTH_ARC_ROTATION_C | AngleRef enumeration |
| DDIF\$_PTH_ARC_ROTATION   | Variable             |
| DDIF\$_PTH_REFERENCE      | Integer              |

## AGGREGATE ITEMS

### **DDIF\$\_PTH\_C**

**Encoding: enumeration; valid values are as follows:**

|                        |  |
|------------------------|--|
| DDIF\$K_PATH_LINE      | Indicates a polyline component of the path. If you specify this value, you must supply values for the items DDIF\$_PTH_LIN_PATH_C through DDIF\$_PTH_LIN_PATH.     |
| DDIF\$K_PATH_BEZIER    | Indicates a cubic Bézier component of the path. If you specify this value, you must supply values for the items DDIF\$_PTH_BEZ_PATH_C through DDIF\$_PTH_BEZ_PATH. |
| DDIF\$K_PATH_ARC       | Indicates an arc component of the path. If you specify this value, you must supply values for the items DDIF\$_PTH_ARC_CENTER_X_C through DDIF\$_PTH_ARC_ROTATION. |
| DDIF\$K_PATH_REFERENCE | Indicates a reference to a defined component of the path. If you specify this value, you must supply a value for the item DDIF\$_PTH_REFERENCE.                    |

A path indicator that indicates the type of path component being defined.

### **DDIF\$\_PTH\_LIN\_PATH\_C**

**Encoding: array of type measurement enumeration**

A line path indicator that specifies whether the layout of the polyline is specified as a variable or constant value.

### **DDIF\$\_PTH\_LIN\_PATH**

**Encoding: array of type variable**

A line path item that lists the control points of the polyline.

The points of the polyline are stored in an array in a repeating **x,y**-pair format. For example, if you are storing values in this item, the first value you specify must be the **x** position of the first control point; the second value must be the **y** position of the first control point, and so on. Because these points are stored in an array, you must increment the aggregate index associated with the array each time you read or write a control point. The initial aggregate index value is 0.

Note that each coordinate is relative to the frame in which it is being rendered.

## DDIF\$\_PTH

### **DDIF\$\_PTH\_BEZ\_PATH\_C**

**Encoding: array of type measurement enumeration**

A curve path indicator that specifies whether the layout of the curve is specified as a variable or constant value.

### **DDIF\$\_PTH\_BEZ\_PATH**

**Encoding: array of type variable**

A curve path item that contains the **x,y** pairs that define the control points of the curve.

The points of the curve are stored in an array in a repeating **x,y**-pair format. For example, if you are storing values in this item, the first value you specify must be the **x** position of the first control point; the second value must be the **y** position of the first control point, and so on. Because these points are stored in an array, you must increment the aggregate index associated with the array each time you read or write a control point. The initial aggregate index value is 0.

### **DDIF\$\_PTH\_ARC\_CENTER\_X\_C**

**Encoding: measurement enumeration**

An arc center **x** indicator that indicates whether the **x**-coordinate of the center of the circle of which this arc is a part is specified as a variable or constant value.

### **DDIF\$\_PTH\_ARC\_CENTER\_X**

**Encoding: variable**

An arc center **x** item that specifies the **x**-coordinate of the center of the circle of which this arc is a part.

### **DDIF\$\_PTH\_ARC\_CENTER\_Y\_C**

**Encoding: measurement enumeration**

An arc center **y** indicator that indicates whether the **y**-coordinate of the center of the circle of which this arc is a part is specified as a variable or constant value.

### **DDIF\$\_PTH\_ARC\_CENTER\_Y**

**Encoding: variable**

An arc center **y** item that specifies the **y**-coordinate of the center of the circle of which this arc is a part.

### **DDIF\$\_PTH\_ARC\_RADIUS\_X\_C**

**Encoding: measurement enumeration**

An arc radius **x** indicator that indicates whether the **x** radius of the arc is specified as a variable or constant value.

### **DDIF\$\_PTH\_ARC\_RADIUS\_X**

**Encoding: variable**

An arc radius **x** item that specifies the distance from the center of the arc to the perimeter of the arc as measured along the **x**-axis.

### **DDIF\$\_PTH\_ARC\_RADIUS\_DELTA\_Y\_C**

**Encoding: measurement enumeration**

An arc radius delta **y** indicator that indicates whether the delta **y** radius of the arc is specified as a variable or constant value.

**DDIF\$ \_PTH\_ARC\_RADIUS\_DELTA\_Y****Encoding: variable**

An arc radius delta y item that specifies the length difference between the y radius and the x radius (for example, if the arc is the arc of an ellipse). The default value for this item is 0.

**DDIF\$ \_PTH\_ARC\_START\_C****Encoding: AngleRef enumeration**

An arc start indicator that indicates whether the starting angle of the arc is specified as a variable or constant value.

**DDIF\$ \_PTH\_ARC\_START****Encoding: variable**

An arc start item that specifies the angle at which the arc is begun. The default value for this item is 0.

Angles are measured in degrees counterclockwise starting from the positive x axis.

**DDIF\$ \_PTH\_ARC\_EXTENT\_C****Encoding: AngleRef enumeration**

An arc extent indicator that indicates whether the extent of the arc is specified as a variable or constant value.

**DDIF\$ \_PTH\_ARC\_EXTENT****Encoding: variable**

An arc extent item that is added to the arc start angle to determine the end of the arc. The default value for this item is 360 degrees.

**DDIF\$ \_PTH\_ARC\_ROTATION\_C****Encoding: AngleRef enumeration**

An arc rotation indicator that indicates whether the angle of rotation of the arc is specified as a variable or as a constant value.

**DDIF\$ \_PTH\_ARC\_ROTATION****Encoding: variable**

An arc rotation item that specifies the angle of rotation of the entire arc relative to the coordinate system. (This item is usually specified for elliptical arcs.) The default value for this item is 0 degrees.

**DDIF\$ \_PTH\_REFERENCE****Encoding: integer**

A path reference item that provides a reference to a defined component of the path, which is itself a composite path. This item references the DDIF\$\_PHD\_NUMBER item.

## DDIF\$\_PVT

---

### DDIF\$\_PVT—Private Content

Private data is defined as compound document semantics that are restricted either to a particular document processing implementation, or to a set of related implementations that support identical private encodings. The private content aggregate lets you specify private data in your document. The DDIF\$\_PVT content aggregate is referenced by these parent aggregate items:

- DDIF\$\_CTD\_VALUE
- DDIF\$\_CTD\_PRIVATE\_DATA
- DDIF\$\_DHD\_PRIVATE\_DATA
- DDIF\$\_FTD\_PRIVATE\_DATA
- DDIF\$\_IDU\_PRIVATE\_CODING\_ATTR
- DDIF\$\_IDU\_COMPRESSION\_PARAMS
- DDIF\$\_LG1\_PRIVATE\_DATA
- DDIF\$\_LSD\_PRIVATE\_DATA
- DDIF\$\_PGD\_PRIVATE\_DATA
- DDIF\$\_PGL\_CONTENT
- DDIF\$\_PHD\_PRIVATE\_DATA
- DDIF\$\_PTD\_PRIVATE\_DATA
- DDIF\$\_SEG\_CONTENT
- DDIF\$\_SGA\_PRIVATE\_DATA
- DDIF\$\_SGA\_CPTFNC\_PARAMETERS
- DDIF\$\_SGA\_IMG\_PRIVATE\_DATA

Refer to these corresponding syntax diagrams:

| Syntax    | Location    |
|-----------|-------------|
| ValueData | Figure B-77 |
| Reference | Figure B-91 |

---

### AGGREGATE FORMAT

| Item Name         | Item Encoding |
|-------------------|---------------|
| DDIF\$_PVT_NAME   | String        |
| DDIF\$_PVT_DATA_C | Enumeration   |
| DDIF\$_PVT_DATA   | Variable      |

| Item Name                      | Item Encoding |
|--------------------------------|---------------|
| DDIF\$_PVT_REFERENCE_ERF_INDEX | Integer       |

## AGGREGATE ITEMS

### **DDIF\$\_PVT\_NAME**

**Encoding: string**

A value name item that uniquely identifies the value.

### **DDIF\$\_PVT\_DATA\_C**

**Encoding: enumeration; valid values are as follows:**

|                          |  |
|--------------------------|--|
| DDIF\$_K_VALUE_BOOLEAN   | Indicates a Boolean value. In this case, the DDIF\$_PVT_DATA item is encoded as a type Boolean.  |
| DDIF\$_K_VALUE_INTEGER   | Indicates an integer value. In this case, the DDIF\$_PVT_DATA item is encoded as an integer.   |
| DDIF\$_K_VALUE_TEXT      | Indicates a text string value. In this case, the DDIF\$_PVT_DATA item is encoded as an array of type character string.   |
| DDIF\$_K_VALUE_GENERAL   | Indicates a stream of bytes in any format. In this case, the DDIF\$_PVT_DATA item is encoded as a string.  |
| DDIF\$_K_VALUE_REFERENCE | Indicates a data value that is a reference to a segment in the document or a segment in another document. In this case, the DDIF\$_PVT_DATA item is encoded as a string. For this case, DDIF\$_PVT_REFERENCE_ERF_INDEX must also be specified.   |
| DDIF\$_K_VALUE_LIST      | Indicates a list of data values such as the above. In this case, the DDIF\$_PVT_DATA item is encoded as a sequence of DDIF\$_PVT aggregates. The value list encoding is defined to be solely a sequence of data. In the nested DDIF\$_PVT aggregates, therefore, the DDIF\$_PVT_NAME item is not written to the output stream. |
| DDIF\$_K_VALUE_EXTERNAL  | Indicates a data value that is represented in a syntax. In this case, the DDIF\$_PVT_DATA item is encoded as the handle of an aggregate of type DDIF\$_EXT.  |

A value data indicator that indicates the type of data that has been named.

### **DDIF\$\_PVT\_DATA**

**Encoding: variable**

A value data item that specifies the data value of the specified type.

### **DDIF\$\_PVT\_REFERENCE\_ERF\_INDEX**

**Encoding: integer**

An external reference index item that specifies an index into a list of external references. This item references the DDIF\$\_DHD\_EXTERNAL\_REFERENCES item.

## DDIF\$\_RCD

---

### DDIF\$\_RCD—Record Definition

The record definition aggregate defines a record structure that consists of one or more primitive data types, expressed as references to variables. Records are used in the calculation of computed content items, such as tables of figures and indexes. The DDIF\$\_RCD aggregate is referenced by the parent aggregate item DDIF\$\_SGB\_RCD\_LIST.

Refer to these corresponding syntax diagrams:

| Syntax     | Location     |
|------------|--------------|
| RecordDefn | Figure B-112 |

---

### AGGREGATE FORMAT

| Item Name           | Item Encoding        |
|---------------------|----------------------|
| DDIF\$_RCD_TYPE     | String               |
| DDIF\$_RCD_TAG      | String               |
| DDIF\$_RCD_CONTENTS | Array of type string |

---

### AGGREGATE ITEMS

**DDIF\$\_RCD\_TYPE**

**Encoding: string**

A record type item that specifies the record type that will be applied to the variable when it is displayed.

**DDIF\$\_RCD\_TAG**

**Encoding: string**

A record tag item that specifies an identifier that indicates which segments within the scope of this record definition cause the creation of a data record of this type.

**DDIF\$\_RCD\_CONTENTS**

**Encoding: array of type string**

A record contents item that specifies the variables of the record. Each variable name and its value at the segment in question become part of the record.

---

## DDIF\$\_RGB—Image Lookup Table Entry

The image (RGB) lookup table entry aggregate provides a method for creating a sequence of lookup table entries, where each entry describes a lookup table index that corresponds to the pixel that it maps. The DDIF\$\_RGB aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_LOOKUP\_TABLES.

Refer to these corresponding syntax diagrams:

| Syntax | Location    |
|--------|-------------|
| RGB    | Figure B-44 |

---

### AGGREGATE FORMAT

| Item Name              | Item Encoding                   |
|------------------------|---------------------------------|
| DDIF\$_RGB_LUT_INDEX   | Integer                         |
| DDIF\$_RGB_RED_VALUE   | Single-precision floating-point |
| DDIF\$_RGB_GREEN_VALUE | Single-precision floating-point |
| DDIF\$_RGB_BLUE_VALUE  | Single-precision floating-point |

---

### AGGREGATE ITEMS

#### ***DDIF\$\_RGB\_LUT\_INDEX***

***Encoding: integer***

A lookup table index item that specifies the integer value of the lookup-table-mapped pixel. This value can range between 0 and  $2^{16} - 1$ .

#### ***DDIF\$\_RGB\_RED\_VALUE***

***Encoding: single-precision floating-point***

A lookup table red value item that specifies the red intensity value for the lookup-table-mapped pixel. This item has a value between 0.0 and 1.0.

#### ***DDIF\$\_RGB\_GREEN\_VALUE***

***Encoding: single-precision floating-point***

A lookup table green value item that specifies the green intensity value for the lookup-table-mapped pixel. This item has a value between 0.0 and 1.0.

#### ***DDIF\$\_RGB\_BLUE\_VALUE***

***Encoding: single-precision floating-point***

A lookup table blue value item that specifies the blue intensity value for the lookup-table-mapped pixel. This item has a value between 0.0 and 1.0.

## DDIF\$\_SEG

---

# DDIF\$\_SEG—Document Segment

The content of a document is contained in a single segment called the **root segment**. The root segment, in turn, contains zero or more content segments or elements, including (but not restricted to) text, graphics, images, galley layout, and nested segments. Each individual content segment aggregate type is discussed in a separate section of this chapter.

The standard content aggregates specify the basic contents of a document, including text, graphics, and images. Each of these content types can be presented on a video display or hardcopy device. The presentation style for the document content is governed by the presentation attributes specified for the segment in which the various aggregates are contained. By grouping the various aggregates in segments, you can create larger units (for example, paragraphs of text).

The DDIF\$\_SEG aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE, DDIF\$\_DDF\_CONTENT, DDIF\$\_PGL\_CONTENT, and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax            | Location    |
|-------------------|-------------|
| BeginSegment      | Figure B-11 |
| SegmentAttributes | Figure B-92 |

---

## AGGREGATE FORMAT

| Item Name                      | Item Encoding                  |
|--------------------------------|--------------------------------|
| DDIF\$_SEG_ID                  | String                         |
| DDIF\$_SEG_USER_LABEL          | Array of type character string |
| DDIF\$_SEG_SEGMENT_TYPE        | String                         |
| DDIF\$_SEG_SPECIFIC_ATTRIBUTES | Handle of DDIF\$_SGA aggregate |
| DDIF\$_SEG_GENERIC_LAYOUT      | Handle of DDIF\$_LG1 aggregate |
| DDIF\$_SEG_SPECIFIC_LAYOUT     | Handle of DDIF\$_LS1 aggregate |
| DDIF\$_SEG_CONTENT             | Sequence of content            |

---

## AGGREGATE ITEMS

### **DDIF\$\_SEG\_ID**

**Encoding:** string

An optional segment identifier that identifies the segment for reference from other segments.



References to labeled segments are not limited to those segments nested under the labeled segment; labeled segments can be referenced from any segment in the document and from other documents. Note that segments should be labeled only if they are referenced. This item is referenced by the DDIF\$\_CTD\_EXTERNAL\_TARGET item and by the DDIF\$\_TYD\_PARENT item.

#### **DDIF\$\_SEG\_USER\_LABEL**

**Encoding:** *array of type character string*

An optional segment user label item that specifies the user-assigned name of the particular segment of content. This string is only for use by the user; it cannot be used to reference the segment from other segments. A typical use of a user label would be to allow users to name graphic objects and manipulate them by name.

#### **DDIF\$\_SEG\_SEGMENT\_TYPE**

**Encoding:** *string*

An optional segment type item that references a segment type definition in the segment attributes of a parent segment or in the style guide.

This string is equivalent to the string specified by the DDIF\$\_TYD\_LABEL item in the type definition (DDIF\$\_TYD) aggregate. For more information, see the description of the DDIF\$\_TYD aggregate. Note that when a segment references a segment type, it acquires the attributes bound to the segment type. However, attribute values already defined by the DDIF\$\_SEG\_SPECIFIC\_ATTRIBUTES item are never replaced by attributes of the segment type.

#### **DDIF\$\_SEG\_SPECIFIC\_ATTRIBUTES**

**Encoding:** *handle of a DDIF\$\_SGA aggregate*

An optional segment attribute item that binds presentation and processing attributes to the segment, and defines generic types and content for reference from nested segments. For more information, see the description of the DDIF\$\_SGA aggregate.

#### **DDIF\$\_SEG\_GENERIC\_LAYOUT**

**Encoding:** *handle of a DDIF\$\_LG1 aggregate*

An optional segment generic layout item that specifies an element of generic layout for the segment. (For more information, see the description of the DDIF\$\_LG1 aggregate.) Note that this item can only be specified on the root segment of a document. Generic layout descriptions placed on segments other than the root segment are ignored.

#### **DDIF\$\_SEG\_SPECIFIC\_LAYOUT**

**Encoding:** *handle of a DDIF\$\_LS1 aggregate*

An optional segment specific layout item that specifies an element of specific layout for the segment. (For more information, see the description of the DDIF\$\_LS1 aggregate.) Note that this item can only be specified on the root segment of a document. Specific layout descriptions placed on segments other than the root segment are ignored.

#### **DDIF\$\_SEG\_CONTENT**

**Encoding:** *sequence of any of the following aggregates:*

|            |            |            |
|------------|------------|------------|
| DDIF\$_ARC | DDIF\$_BEZ | DDIF\$_CRF |
| DDIF\$_EXT | DDIF\$_FAS | DDIF\$_GLY |

## DDIF\$\_SEG

|            |            |            |
|------------|------------|------------|
| DDIF\$_GTX | DDIF\$_HRD | DDIF\$_HRV |
| DDIF\$_IMG | DDIF\$_LIN | DDIF\$_PVT |
| DDIF\$_SEG | DDIF\$_SFT | DDIF\$_SFV |
| DDIF\$_TXT |            |            |

An optional segment content item that specifies the content of the segment. The DDIF\$\_SEG\_CONTENT item contains the handle of the first aggregate in the sequence of content aggregates.

---

## DDIF\$\_SFT—Soft Directive

The soft directive content aggregate specifies a directive that is inserted into the document by some application. All directives are restricted to the \$T (text) content category. The DDIF\$\_SFT aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax              | Location    |
|---------------------|-------------|
| FormattingPrimitive | Figure B-18 |
| Directive           | Figure B-20 |

---

## AGGREGATE FORMAT

| Item Name            | Item Encoding |
|----------------------|---------------|
| DDIF\$_SFT_DIRECTIVE | Enumeration   |

---

## AGGREGATE ITEMS

### **DDIF\$\_SFT\_DIRECTIVE**

*Encoding: enumeration; valid values are as follows:*

|                             |  |
|-----------------------------|--|
| DDIF\$K_DIR_NEW_PAGE        | Begins a new page.   |
| DDIF\$K_DIR_NEW_LINE        | Begins a new line of text.   |
| DDIF\$K_DIR_NEW_GALLEY      | Begins a new layout galley (such as a column). Software that does not support galley layout interprets the new galley directive as a new page.   |
| DDIF\$K_DIR_TAB             | Moves the horizontal text position to the next tab stop.   |
| DDIF\$K_DIR_SPACE           | Is treated as a space in the current font. The space directive is usually soft, and is used to indicate that software has inserted a space between wrapped lines.                          |
| DDIF\$K_DIR_HYPHEN_NEW_LINE | Specifies that the line break is preceded by a hyphen. This directive is typically soft, and is used to indicate that software has inserted a hyphen at the place where it broke the line. |

## DDIF\$\_SFT

|                              |  |
|------------------------------|--|
| DDIF\$K_DIR_WORD_BREAK_POINT | Identifies an embedded point at which a word may be broken, if need be, for justification.   |
| DDIF\$K_DIR_LEADERS          | Inserts leader characters according to the current leader attributes. A leader directive is treated like a space during justification, except that leader characters are inserted instead of space. The rendering of leaders is controlled by the current leader attributes and other text attributes. |
| DDIF\$K_DIR_BACKSPACE        | Specifies that the first character following this directive should be centered over the last character imaged.   |
| DDIF\$K_NULL                 | Suppresses the inheritance of the initial-directive element of layout attributes. This directive has no effect on imaging or processing.   |
| DDIF\$K_DIR_NO_HYPHEN_WORD   | Suppresses hyphenation until the next space character or space directive is encountered.   |

A soft directive item that specifies the type of soft directive (for example, a software-inserted page break) to insert in the document.

---

## DDIF\$\_SFV—Soft Value Directive

The soft value directive content aggregate is a soft directive that has a parametric value. The DDIF\$\_SFV aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to these corresponding syntax diagrams:

| Syntax              | Location    |
|---------------------|-------------|
| FormattingPrimitive | Figure B-18 |
| ValueDirective      | Figure B-19 |
| EscapementDirective | Figure B-21 |
| VariableReset       | Figure B-22 |
| Escapement          | Figure B-47 |
| Ratio               | Figure B-70 |

---

## AGGREGATE FORMAT

| Item Name                 | Item Encoding           |
|---------------------------|-------------------------|
| DDIF\$_SFV_C              | Enumeration             |
| DDIF\$_SFV_ESC_RATIO_N    | Integer                 |
| DDIF\$_SFV_ESC_RATIO_D    | Integer                 |
| DDIF\$_SFV_ESC_CONSTANT_C | Measurement enumeration |
| DDIF\$_SFV_ESC_CONSTANT   | Variable                |
| DDIF\$_SFV_RESET_VARIABLE | String                  |
| DDIF\$_SFV_RESET_VALUE_C  | Expression enumeration  |
| DDIF\$_SFV_RESET_VALUE    | Variable                |

---

## AGGREGATE ITEMS

### **DDIF\$\_SFV\_C**

**Encoding: enumeration; valid values are as follows:**

|                        |   |
|------------------------|---|
| DDIF\$K_DIR_ESCAPEMENT | Indicates an escapement directive that specifies the relative or constant distance by which to increment the current text position. If you specify this value, you must supply values for the items DDIF\$_SFV_ESC_RATIO_N through DDIF\$_SFV_ESC_CONSTANT. |
|------------------------|---|

## DDIF\$\_SFV

**DDIF\$\_K\_DIR\_VARIABLE\_RESET** Indicates a variable reset directive that specifies a directive to reset the value of the specified variable. If you specify this value, you must supply values for the items DDIF\$\_SFV\_RESET\_VARIABLE through DDIF\$\_SFV\_RESET\_VALUE.

A soft value directive indicator that specifies whether the soft value directive is an escapement directive or a variable reset directive.

### **DDIF\$\_SFV\_ESC\_RATIO\_N**

**Encoding: integer**

An escapement ratio numerator item that specifies the magnitude of a ratio, which multiplies the em-space width for the current font. The width of an em space is often the same as the width of the capital letter M, but this depends on the font. The default value is 1, if this item is missing and if the numerator is present.

If both the numerator and the denominator are not specified, the ratio is not the default value, but is instead considered optionally absent.

### **DDIF\$\_SFV\_ESC\_RATIO\_D**

**Encoding: integer**

An escapement ratio denominator item that specifies the units of precision used in the ratio. The default value is 100, if the item is missing and if the numerator is present.

If both the numerator and the denominator are not specified, the ratio is not the default value, but is instead considered optionally absent.

### **DDIF\$\_SFV\_ESC\_CONSTANT\_C**

**Encoding: measurement enumeration**

An escapement constant indicator that indicates whether the escapement constant is specified as a variable or constant value.

### **DDIF\$\_SFV\_ESC\_CONSTANT**

**Encoding: variable**

An escapement constant item that specifies the constant measurement to be used as an escapement.

### **DDIF\$\_SFV\_RESET\_VARIABLE**

**Encoding: string**

A reset variable item that specifies the label of the variable to be reset by the soft value directive.

### **DDIF\$\_SFV\_RESET\_VALUE\_C**

**Encoding: expression enumeration**

A reset value indicator that indicates whether the soft value directive reset value is specified as a variable or constant value.

### **DDIF\$\_SFV\_RESET\_VALUE**

**Encoding: variable**

A reset value item that specifies the new value of the variable.

---

## DDIF\$\_SGA—Segment Attributes

The segment attributes aggregate defines the presentation and processing characteristics of a segment of document content. The DDIF\$\_SGA aggregate is referenced by the parent aggregate item DDIF\$\_SEG\_SPECIFIC\_ATTRIBUTES.

The items in the DDIF\$\_SGA aggregate are described in the following logical groups:

- General segment attribute items
- Computed content attribute items
- Structure items
- A language attribute item
- Legend items
- Measurement items
- An alternate presentation item
- Layout items
- A font definitions item
- A pattern definitions item
- A path definitions item
- A line-style definitions item
- A content definitions item
- A type definitions item
- Text attribute items
- Line attribute items
- Marker attribute items
- A galley attribute item
- Image attribute items
- Image space items
- Frame items
- An item-change-list item

Each of these items, or groups of items, is discussed in the following sections.

Refer to these corresponding syntax diagrams and to the following subsections for nested syntax references:

| Syntax            | Location    |
|-------------------|-------------|
| SegmentAttributes | Figure B-92 |

# DDIF\$\_SGA

---

## AGGREGATE FORMAT

| Item Name                    | Item Encoding                             |
|------------------------------|---|
| <b>General Segment</b>       |   |
| DDIF\$_SGA_PRIVATE_DATA      | Sequence of DDIF\$_PVT aggregates         |
| DDIF\$_SGA_CONTENT_STREAMS   | Array of type string                      |
| DDIF\$_SGA_CONTENT_CATEGORY  | String with <b>add-info</b>               |
| DDIF\$_SGA_SEGMENT_TAGS      | Array of type string with <b>add-info</b> |
| DDIF\$_SGA_BINDING_DEFNS     | Sequence of DDIF\$_SGB aggregates         |
| <b>Computed content</b>      |   |
| DDIF\$_SGA_COMPUTE_C         | Enumeration                               |
| DDIF\$_SGA_CPTCPY_TARGET     | String                                    |
| DDIF\$_SGA_CPTCPY_ERF_INDEX  | Integer                                   |
| DDIF\$_SGA_CPTVAR_VARIABLE   | String                                    |
| DDIF\$_SGA_CPTXRF_TARGET     | String                                    |
| DDIF\$_SGA_CPTXRF_ERF_INDEX  | Integer                                   |
| DDIF\$_SGA_CPTXRF_VARIABLE   | String                                    |
| DDIF\$_SGA_CPTFNC_NAME       | String                                    |
| DDIF\$_SGA_CPTFNC_PARAMETERS | Sequence of DDIF\$_PVT aggregates         |
| <b>Structure</b>             |   |
| DDIF\$_SGA_STRUCTURE_DESC_C  | Enumeration                               |
| DDIF\$_SGA_STRUCTURE_DESC    | Sequence of DDIF\$_OCC aggregates         |
| <b>Language</b>              |   |
| DDIF\$_SGA_LANGUAGE          | Integer                                   |
| <b>Legend</b>                |   |
| DDIF\$_SGA_LEGEND_UNIT_N     | Integer                                   |
| DDIF\$_SGA_LEGEND_UNIT_D     | Integer                                   |
| DDIF\$_SGA_LEGEND_UNIT_NAME  | Array of type character string            |
| <b>Measurement</b>           |   |
| DDIF\$_SGA_UNITS_PER_MEASURE | Integer                                   |
| DDIF\$_SGA_UNIT_NAME         | Array of type character string            |
| <b>Alternative</b>           |   |
| DDIF\$_SGA_ALT_PRESENTATION  | Array of type character string            |
| <b>Layout</b>                |   |
| DDIF\$_SGA_LAYOUT_C          | Enumeration                               |
| DDIF\$_SGA_LAYGLY_WRAP       | Handle of DDIF\$_LW1 aggregate            |
| DDIF\$_SGA_LAYGLY_LAYOUT     | Handle of DDIF\$_LL1 aggregate            |
| DDIF\$_SGA_LAYPTH_PATH       | Sequence of DDIF\$_PTH aggregates         |



| Item Name                       | Item Encoding                     |
|---------------------------------|-----------------------------------|
| DDIF\$_SGA_LAYPTH_FORMAT        | Enumeration                       |
| DDIF\$_SGA_LAYPTH_ORIENTATION_C | Enumeration                       |
| DDIF\$_SGA_LAYPTH_ORIENTATION   | Variable                          |
| DDIF\$_SGA_LAYPTH_H_ALIGN       | Enumeration                       |
| DDIF\$_SGA_LAYPTH_V_ALIGN       | Enumeration                       |
| DDIF\$_SGA_LAYREL_H_RATIO_N     | Integer                           |
| DDIF\$_SGA_LAYREL_H_RATIO_D     | Integer                           |
| DDIF\$_SGA_LAYREL_H_CONSTANT_C  | Measurement enumeration           |
| DDIF\$_SGA_LAYREL_H_CONSTANT    | Variable                          |
| DDIF\$_SGA_LAYREL_V_RATIO_N     | Integer                           |
| DDIF\$_SGA_LAYREL_V_RATIO_D     | Integer                           |
| DDIF\$_SGA_LAYREL_V_CONSTANT_C  | Measurement enumeration           |
| DDIF\$_SGA_LAYREL_V_CONSTANT    | Variable                          |
| DDIF\$_SGA_LAYPOS_TEXT_POSITION | Enumeration                       |
| <b>Definitions</b>              |                                   |
| DDIF\$_SGA_BINDING_DEFNS        | Sequence of DDIF\$_SGB aggregates |
| DDIF\$_SGA_FONT_DEFNS           | Sequence of DDIF\$_FTD aggregates |
| DDIF\$_SGA_PATTERN_DEFNS        | Sequence of DDIF\$_PTD aggregates |
| DDIF\$_SGA_PATH_DEFNS           | Sequence of DDIF\$_PHD aggregates |
| DDIF\$_SGA_LINE_STYLE_DEFNS     | Sequence of DDIF\$_LSD aggregates |
| DDIF\$_SGA_CONTENT_DEFNS        | Sequence of DDIF\$_CTD aggregates |
| DDIF\$_SGA_TYPE_DEFNS           | Sequence of DDIF\$_TYD aggregates |
| <b>Text</b>                     |                                   |
| DDIF\$_SGA_TXT_MASK_PATTERN     | Integer                           |
| DDIF\$_SGA_TXT_FONT             | Integer                           |
| DDIF\$_SGA_TXT_RENDITION        | Array of type enumeration         |
| DDIF\$_SGA_TXT_HEIGHT_C         | Measurement enumeration           |
| DDIF\$_SGA_TXT_HEIGHT           | Variable                          |
| DDIF\$_SGA_TXT_SET_SIZE_N       | Integer                           |
| DDIF\$_SGA_TXT_SET_SIZE_D       | Integer                           |
| DDIF\$_SGA_TXT_DIRECTION        | Enumeration                       |
| DDIF\$_SGA_TXT_DEC_ALIGNMENT    | Array of type character string    |
| DDIF\$_SGA_TXT_LEADER_SPACE_C   | Measurement enumeration           |
| DDIF\$_SGA_TXT_LEADER_SPACE     | Variable                          |
| DDIF\$_SGA_TXT_LEADER_BULLET    | Character string                  |
| DDIF\$_SGA_TXT_LEADER_ALIGN     | Enumeration                       |
| DDIF\$_SGA_TXT_LEADER_STYLE     | Enumeration                       |
| DDIF\$_SGA_TXT_PAIR_KERNING     | Boolean                           |

## DDIF\$\_SGA

| Item Name                        | Item Encoding                     |
|----------------------------------|-----------------------------------|
| <b>Line</b>                      |                                   |
| DDIF\$_SGA_LIN_WIDTH_C           | Measurement enumeration           |
| DDIF\$_SGA_LIN_WIDTH             | Variable                          |
| DDIF\$_SGA_LIN_STYLE             | Integer                           |
| DDIF\$_SGA_LIN_PATTERN_SIZE_C    | Measurement enumeration           |
| DDIF\$_SGA_LIN_PATTERN_SIZE      | Variable                          |
| DDIF\$_SGA_LIN_MASK_PATTERN      | Integer                           |
| DDIF\$_SGA_LIN_END_START         | Enumeration                       |
| DDIF\$_SGA_LIN_END_FINISH        | Enumeration                       |
| DDIF\$_SGA_LIN_END_SIZE_C        | Measurement enumeration           |
| DDIF\$_SGA_LIN_END_SIZE          | Variable                          |
| DDIF\$_SGA_LIN_JOIN              | Enumeration                       |
| DDIF\$_SGA_LIN_MITER_LIMIT_N     | Integer                           |
| DDIF\$_SGA_LIN_MITER_LIMIT_D     | Integer                           |
| DDIF\$_SGA_LIN_INTERIOR_PATTERN  | Integer                           |
| <b>Marker</b>                    |                                   |
| DDIF\$_SGA_MKR_STYLE             | Enumeration                       |
| DDIF\$_SGA_MKR_MASK_PATTERN      | Integer                           |
| DDIF\$_SGA_MKR_SIZE_C            | Measurement enumeration           |
| DDIF\$_SGA_MKR_SIZE              | Variable                          |
| <b>Galley</b>                    |                                   |
| DDIF\$_SGA_GLY_ATTRIBUTES        | Handle of DDIF\$_GLA aggregate    |
| <b>Image</b>                     |                                   |
| DDIF\$_SGA_IMG_PRIVATE_DATA      | Sequence of DDIF\$_PVT aggregates |
| DDIF\$_SGA_IMG_PIXEL_PATH        | Integer                           |
| DDIF\$_SGA_IMG_LINE_PROGRESSION  | Integer                           |
| DDIF\$_SGA_IMG_PP_PIXEL_DIST     | Integer                           |
| DDIF\$_SGA_IMG_LP_PIXEL_DIST     | Integer                           |
| DDIF\$_SGA_IMG_BRT_POLARITY      | Enumeration                       |
| DDIF\$_SGA_IMG_GRID_TYPE         | Enumeration                       |
| DDIF\$_SGA_IMG_SPECTRAL_MAPPING  | Enumeration                       |
| DDIF\$_SGA_IMG_LOOKUP_TABLES_C   | Enumeration                       |
| DDIF\$_SGA_IMG_LOOKUP_TABLES     | Variable                          |
| DDIF\$_SGA_IMG_COMP_WAVELENGTH_C | Enumeration                       |
| DDIF\$_SGA_IMG_COMP_WAVELENGTH   | Variable                          |
| DDIF\$_SGA_IMG_COMP_SPACE_ORG    | Enumeration                       |
| DDIF\$_SGA_IMG_PLANES_PER_PIXEL  | Integer                           |
| DDIF\$_SGA_IMG_PLANE_SIGNIF      | Enumeration                       |
| DDIF\$_SGA_IMG_NUMBER_OF_COMP    | Integer                           |
| DDIF\$_SGA_IMG_BITS_PER_COMP     | Array of type integer             |

| Item Name                       | Item Encoding                     |
|---------------------------------|-----------------------------------|
| <b>Frame</b>                    |                                   |
| DDIF\$_SGA_FRM_FLAGS            | Longword                          |
| DDIF\$_SGA_FRM_BOX_LL_X_C       | Measurement enumeration           |
| DDIF\$_SGA_FRM_BOX_LL_X         | Variable                          |
| DDIF\$_SGA_FRM_BOX_LL_Y_C       | Measurement enumeration           |
| DDIF\$_SGA_FRM_BOX_LL_Y         | Variable                          |
| DDIF\$_SGA_FRM_BOX_UR_X_C       | Measurement enumeration           |
| DDIF\$_SGA_FRM_BOX_UR_X         | Variable                          |
| DDIF\$_SGA_FRM_BOX_UR_Y_C       | Measurement enumeration           |
| DDIF\$_SGA_FRM_BOX_UR_Y         | Variable                          |
| DDIF\$_SGA_FRM_OUTLINE          | Sequence of DDIF\$_PTH aggregates |
| DDIF\$_SGA_FRM_CLIPPING         | Sequence of DDIF\$_PTH aggregates |
| DDIF\$_SGA_FRM_POSITION_C       | Enumeration                       |
| DDIF\$_SGA_FRMFXD_POSITION_X_C  | Measurement enumeration           |
| DDIF\$_SGA_FRMFXD_POSITION_X    | Variable                          |
| DDIF\$_SGA_FRMFXD_POSITION_Y_C  | Measurement enumeration           |
| DDIF\$_SGA_FRMFXD_POSITION_Y    | Variable                          |
| DDIF\$_SGA_FRMINL_BASE_OFFSET_C | Measurement enumeration           |
| DDIF\$_SGA_FRMINL_BASE_OFFSET   | Variable                          |
| DDIF\$_SGA_FRMGly_VERTICAL      | Enumeration                       |
| DDIF\$_SGA_FRMGly_HORIZONTAL    | Enumeration                       |
| DDIF\$_SGA_FRMMAR_BASE_OFFSET_C | Measurement enumeration           |
| DDIF\$_SGA_FRMMAR_BASE_OFFSET   | Variable                          |
| DDIF\$_SGA_FRMMAR_NEAR_OFFSET_C | Measurement enumeration           |
| DDIF\$_SGA_FRMMAR_NEAR_OFFSET   | Variable                          |
| DDIF\$_SGA_FRMMAR_HORIZONTAL    | Enumeration                       |
| DDIF\$_SGA_FRM_TRANSFORM        | Sequence of DDIF\$_TRN aggregates |
| <b>Item change</b>              |                                   |
| DDIF\$_SGA_ITEM_CHANGE_LIST     | Item change list                  |

## DDIF\$\_SGA General Segment Attributes

---

### General Segment Attributes

General segment attributes specify any attributes that can be generally applied to a segment.

Refer to these corresponding syntax diagrams:

| Syntax            | Location    |
|-------------------|-------------|
| SegmentAttributes | Figure B-92 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_PRIVATE\_DATA**

**Encoding:** *sequence of DDIF\$\_PVT aggregates*

An optional private attributes item that specifies any product-specific attributes for the segment. (For more information, see the description of the DDIF\$\_PVT aggregate.) No initial private attributes are defined.

#### **DDIF\$\_SGA\_CONTENT\_STREAMS**

**Encoding:** *array of type string*

An optional content streams item that specifies the content streams to which the segment content belongs. The initial content stream is "\$DB", which denotes the document body. Other tags denoting content streams in layout that are registered in the DDIF Standard are the following:

|       |                          |
|-------|--------------------------|
| \$TOC | Table of contents stream |
| \$IX  | Index content stream     |
| \$FN  | Footnote stream          |
| \$MN  | Margin note stream       |
| \$EN  | End note stream          |

#### **DDIF\$\_SGA\_CONTENT\_CATEGORY**

**Encoding:** *string with add-info*

An optional content category item that indicates the category of content, such as text (\$T), graphics (\$2D), or image (\$I), to which the content of the segment belongs. **Add-info** can take the following values:

|                          |  |
|--------------------------|--|
| DDIF\$K_PRIVATE_CATEGORY | The content is nonstandard or was standardized after the release of the CDA Toolkit. |
| DDIF\$K_I_CATEGORY       | The content is of the image (\$I) category.  |
| DDIF\$K_2D_CATEGORY      | The content is of the graphics (\$2D) category.                                      |
| DDIF\$K_T_CATEGORY       | The content is of the text (\$T) category.   |
| DDIF\$K_TBL_CATEGORY     | The content is of the table (\$TBL) category.  |
| DDIF\$K_PDL_CATEGORY     | The content is of the page description language (\$PDL) category.                    |

The initial value is DDIF\$K\_T\_CATEGORY, meaning that the content category is text (\$T).

## DDIF\$ \_SGA General Segment Attributes

### **DDIF\$ \_SGA\_SEGMENT\_TAGS**

**Encoding:** *array of type string with add-info*

An optional segment tags item that specifies tags that denote the processing characteristics of the content. **Add-info** can take the following values:

|                     |   |
|---------------------|---|
| DDIF\$K_PRIVATE_TAG | The segment tag is a nonstandard tag.             |
| DDIF\$K_CRF_TAG     | The segment tag is a cross-reference (\$CRF) tag. |
| DDIF\$K_F_TAG       | The segment tag is a figure (\$F) tag.            |
| DDIF\$K_P_TAG       | The segment tag is a paragraph (\$P) tag.         |
| DDIF\$K_S_TAG       | The segment tag is a section (\$S) tag.           |
| DDIF\$K_I_TAG       | The segment tag is an index (\$I) tag.            |
| DDIF\$K_E_TAG       | The segment tag is an emphasis (\$E) tag.         |
| DDIF\$K_L_TAG       | The segment tag is a list (\$L) tag.              |
| DDIF\$K_LE_TAG      | The segment tag is a list element (\$LE) tag.     |
| DDIF\$K_LIT_TAG     | The segment tag is a literal (\$LIT) tag.         |
| DDIF\$K_FN_TAG      | The segment tag is a footnote (\$FN) tag.         |
| DDIF\$K_AN_TAG      | The segment tag is an annotation (\$AN) tag.      |
| DDIF\$K_LBL_TAG     | The segment tag is a label (\$LBL) tag.           |
| DDIF\$K_TTL_TAG     | The segment tag is a title (\$TTL) tag.           |
| DDIF\$K_GRP_TAG     | The segment tag is a group member (\$GRP) tag.    |
| DDIF\$K_GO_TAG      | The segment tag is a graphic object (\$GO) tag.   |

Initially, there are no segment tags specified.

### **DDIF\$ \_SGA\_BINDING\_DEFNS**

**Encoding:** *sequence of DDIF\$ \_SGB aggregates*

An optional segment binding item that lists the variables bound to the segment. (For more information, see the description of the DDIF\$ \_SGB aggregate. Initially there are no segment bindings.)

# DDIF\$\_SGA

## Computed Content Attributes

---

### Computed Content Attributes

The segment attributes aggregate contains items used to control computed content attributes. The computed content attributes are specified by first selecting the type of computed content, and then specifying the appropriate information for that type.

Refer to these corresponding syntax diagrams:

| Syntax      | Location    |
|-------------|-------------|
| ComputeDefn | Figure B-45 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_COMPUTE\_C**

**Encoding: enumeration; valid values are as follows:**

|                          |   |
|--------------------------|---|
| DDIF\$K_COPY_COMPUTE     | Indicates that the content originates from another segment in this document, or an external document, and that the content is updated only at the user's request. If you specify this computed content type, you must supply values for the items DDIF\$_SGA_CPTCPY_TARGET and DDIF\$_SGA_CPTCPY_ERF_INDEX. |
| DDIF\$K_REMOTE_COMPUTE   | Indicates that the content originates from another segment in this document, or an external document, and that the content is updated every time it is displayed. If you specify this computed content type, you must supply values for the items DDIF\$_SGA_CPTCPY_TARGET and DDIF\$_SGA_CPTCPY_ERF_INDEX. |
| DDIF\$K_VARIABLE_COMPUTE | Indicates the content source as the current value that is bound to a variable by this segment or in some parent segment. If you specify this computed content type, you must supply a value for the item DDIF\$_SGA_CPTVAR_VARIABLE.  |
| DDIF\$K_XREF_COMPUTE     | Indicates the content source as the current value that is bound to a variable at the indicated target segment. If you specify this computed content type, you must supply values for the items DDIF\$_SGA_CPTXRF_TARGET through DDIF\$_SGA_CPTXRF_VARIABLE.   |
| DDIF\$K_FUNCTION_COMPUTE | Indicates the content source as the result of some external process applied to parameters. If you specify this computed content type, you must supply values for the items DDIF\$_SGA_CPTFNC_NAME and DDIF\$_SGA_CPTFNC_PARAMETERS.   |

A computed content indicator that selects the type of computed content. Each of these computed content types is discussed in the following sections, along with its corresponding items.

---

## Copied and Remote Computed Content

The copied computed content is selected by specifying `DDIF$ _SGA_COMPUTE_C` as `DDIF$K_COPY_COMPUTE` or `DDIF$K_REMOTE_COMPUTE`.

In the case of remote computed content, the same aggregate items apply. The difference is that, for copied computed content, the content of the segment is updated only at the user's request. In the case of remote content, the content of the segment is updated when the document is received.

Refer to these corresponding syntax diagrams:

| Syntax           | Location    |
|------------------|-------------|
| ExternalRefIndex | Figure B-58 |
| Reference        | Figure B-91 |

---

## AGGREGATE ITEMS

### ***DDIF\$ \_SGA\_CPTCPY\_TARGET***

***Encoding: string***

A reference target item that indicates the label of the segment being referenced. If this item is not specified, the entire document is being referenced.

### ***DDIF\$ \_SGA\_CPTCPY\_ERF\_INDEX***

***Encoding: integer***

A reference index item that specifies an index into a list of external references stored in the document header. This item is encoded as an integer. If this item is not specified, the reference is to the current document. This item references the `DDIF$ _DHD_EXTERNAL_REFERENCES` item.

## **DDIF\$ \_SGA**

### **Variable Computed Content**

---

### **Variable Computed Content**

The variable computed content is selected by specifying DDIF\$ \_SGA\_COMPUTE\_C as DDIF\$K\_VARIABLE\_COMPUTE.

Refer to these corresponding syntax diagrams:

| <b>Syntax</b> | <b>Location</b> |
|---------------|-----------------|
| VariableLabel | Figure B-61     |

---

### **AGGREGATE ITEMS**

***DDIF\$ \_SGA\_CPTVAR\_VARIABLE***

***Encoding: string***

A variable item that specifies the name of the variable.



---

## Cross-Reference Computed Content

The cross-reference computed content is selected by specifying DDIF\$ \_SGA\_ COMPUTE\_C as DDIF\$K\_XREF\_COMPUTE.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| CrossReference | Figure B-46 |

---

## AGGREGATE ITEMS

### **DDIF\$ \_SGA\_CPTXRF\_TARGET**

**Encoding: string**

A cross-reference target segment label that specifies the label by which the target segment is referenced. If you do not specify a target segment label, the document root segment is referenced.

### **DDIF\$ \_SGA\_CPTXRF\_ERF\_INDEX**

**Encoding: integer**

A cross-reference index item that specifies an index into a list of external references stored in the document header. If you do not specify a value for this item, the current document is referenced.

### **DDIF\$ \_SGA\_CPTXRF\_VARIABLE**

**Encoding: string**

A cross-reference variable label that specifies the name of the variable containing the value being referenced.

## DDIF\$\_SGA Function Computed Content

---

### Function Computed Content

The function computed content is selected by specifying DDIF\$\_SGA\_COMPUTE\_C as DDIF\$K\_FUNCTION\_COMPUTE.

Refer to these corresponding syntax diagrams:

| Syntax       | Location    |
|--------------|-------------|
| FunctionLink | Figure B-57 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_CPTFNC\_NAME**

**Encoding:** *string*

A function name item that specifies the name of the function, which is used in conjunction with user-preference information to uniquely identify a program that is to be invoked with the indicated parameters.

#### **DDIF\$\_SGA\_CPTFNC\_PARAMETERS**

**Encoding:** *sequence of DDIF\$\_PVT aggregates*

A function parameters item that indicates the sequence of parameters required by the function. For more information, see the description of the DDIF\$\_PVT aggregate.

## Structure Description

The structure attributes specify the legal logical structure of references to segment type definitions within the segment. They describe a set of constraints placed on the ordering, the grouping, and the number of segments with type references. The structure description is initially absent—all combinations of reference are valid.

Refer to these corresponding syntax diagrams:

| Syntax        | Location    |
|---------------|-------------|
| StructureDefn | Figure B-94 |

## AGGREGATE ITEMS

### **DDIF\$ \_SGA \_STRUCTURE \_DESC \_C**

**Encoding: enumeration; valid values are as follows:**

|                            |   |
|----------------------------|---|
| DDIF\$K_SEQUENCE_STRUCTURE | Indicates a sequence of element occurrences that are constrained to occur in the order specified. In this case, the DDIF\$ _SGA _STRUCTURE _DESC item is encoded as a sequence of DDIF\$ _OCC aggregates. |
| DDIF\$K_SET_STRUCTURE      | Indicates a set of element occurrences that are <i>not</i> constrained with respect to order. In this case, the DDIF\$ _SGA _STRUCTURE _DESC item is encoded as a sequence of DDIF\$ _OCC aggregates.     |
| DDIF\$K_CHOICE_STRUCTURE   | Indicates a group of element occurrences from which only one can be selected. In this case, the DDIF\$ _SGA _STRUCTURE _DESC item is encoded as a sequence of DDIF\$ _OCC aggregates.                     |

A structure description indicator that specifies the type of legal logical structure.

### **DDIF\$ \_SGA \_STRUCTURE \_DESC**

**Encoding: sequence of DDIF\$ \_OCC aggregates**

A structure description item that specifies the structure itself. This item is encoded as a sequence of DDIF\$ \_OCC aggregates, regardless of which structure is selected using the DDIF\$ \_SGA \_STRUCTURE \_DESC \_C item. (For more information, see the description of the DDIF\$ \_OCC aggregate.)

## DDIF\$\_SGA Language Attributes

---

### Language Attributes

The language item lets you specify the natural or synthetic (programming) language of text in the segment.

Refer to these corresponding syntax diagrams:

| Syntax        | Location    |
|---------------|-------------|
| LanguageIndex | Figure B-59 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_LANGUAGE**

**Encoding:** *integer*

An optional language item that defines the natural or synthetic (programming) language of text in the segment. This item is an index into the table of languages included in the DDIF\$\_DHD\_LANGUAGES item.

The language does not imply text direction or formatting conventions, as these are expressed by presentation and layout attributes. Instead, the language is used to select language tools such as spelling checkers. The initial language index value is 1.

---

## Legend Attributes

The legend attributes describe the world coordinate system for the content of a segment. Legend units do not affect the rendition of document content. Instead, they indicate the scale of an illustration.

Refer to these corresponding syntax diagrams:

---

| <b>Syntax</b> | <b>Location</b> |
|---------------|-----------------|
| LegendUnits   | Figure B-65     |

---

## AGGREGATE ITEMS

### ***DDIF\$\_SGA\_LEGEND\_UNIT\_N***

***Encoding: integer***

An optional legend unit numerator item that specifies the magnitude of the ratio of the user coordinate system to the document coordinate system. The default value of the numerator is 1.

### ***DDIF\$\_SGA\_LEGEND\_UNIT\_D***

***Encoding: integer***

An optional legend unit denominator item that specifies the units of precision used in the ratio. The default value of the denominator is 100.

### ***DDIF\$\_SGA\_LEGEND\_UNIT\_NAME***

***Encoding: array of type character string***

A legend unit name item that specifies the name of the user coordinate system. This item has an initial value of "inches."

## DDIF\$\_SGA Measurement Attributes

---

### Measurement Attributes

The optional measurement attributes describe the coordinate system used within the segment. Measurement units always specify the number of units per inch, regardless of the nesting of segments with measurement unit declarations. The measurement attribute items specify the precision of measurements, rather than the scale of measurements. Note that measurement units specified in specific attributes are in effect for the measurements specified in subsequent attributes.

Refer to these corresponding syntax diagrams:

| Syntax           | Location    |
|------------------|-------------|
| MeasurementUnits | Figure B-75 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_UNITS\_PER\_MEASURE**

**Encoding: integer**

A units per measurement item that specifies the number of units per inch. This item has an initial value of 1200.

#### **DDIF\$\_SGA\_UNIT\_NAME**

**Encoding: array of type character string**

A unit name item that specifies the name of the measurement system. This item has an initial value of BMU. The BMU is a Basic Measurement Unit that is a standard unit of measure used in DDIF and equal to 1/1200th of an inch.

---

## Alternate Presentation Attribute

The optional alternate presentation item lets you specify a string that can be presented to the user when the content of the segment cannot be displayed.

Refer to these corresponding syntax diagrams:

| Syntax            | Location    |
|-------------------|-------------|
| SegmentAttributes | Figure B-92 |

---

## AGGREGATE ITEMS

### ***DDIF\$\_SGA\_ALT\_PRESENTATION***

***Encoding: array of type character string***

The optional alternate presentation item contains a string that can be presented to the user when the content of the segment cannot be displayed.

This is an optional string for use with the application's error message under that particular condition. This string is initially absent.

# DDIF\$\_SGA

## Layout Attributes

---

### Layout Attributes

Layout attributes define how a text processor images characters along paths. DDIF defines four mechanisms for describing the layout path of text:

1. **Galley-based layout** describes the flow of text among galleys (columns and pages). The parameters used to describe galley-based layout include layout blocks, margins, page sizes, external hyphenation libraries, widow and orphan penalties, and user-specified layout directives such as *new-page*.

In galley-based layout, the location of each successive path is determined algorithmically, but the algorithm may require several passes in order to optimize white space or arrange an illustration close to its referencing text.

Layout of text content in the Text (\$T) content category is always galley based. Positional graphics text is usually path based.

2. **Path-based layout** describes the flow of text along a path. This path can be a straight line, a series of line segments, or a curve. Along the path, characters have an orientation with respect to the path itself or with respect to the frame in which they are imaged. For example, characters can be tangent to the path, or upright with respect to the frame. Path-based layout is restricted to the Graphics (\$2D) content category.

While segments that specify layout paths are not normally nested within other segments that specify a layout path, such a situation has a defined behavior: text within a segment is placed on the current path. At the end of a nested segment, the previous path is restored.

3. **Position-relative layout** describes the position of text relative to the current text position.
4. **Text-position layout** describes the position of the text as a defined character-relative position such as “superscript.”

The layout of the content is described by first selecting the type of layout and then specifying the appropriate information for that type.

Refer to these corresponding syntax diagrams:

| Syntax     | Location    |
|------------|-------------|
| TextLayout | Figure B-16 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_LAYOUT\_C**

**Encoding: enumeration; valid values are as follows:**

|                       |  |
|-----------------------|--|
| DDIF\$K_GALLEY_LAYOUT | Indicates text laid out in galleys. If you specify this layout type, you must supply values for the items DDIF\$_SGA_LAYGLY_WRAP and DDIF\$_SGA_LAYGLY_LAYOUT. |
|-----------------------|--|



## **DDIF\$\_SGA Layout Attributes**

### **DDIF\$K\_PATH\_LAYOUT**

Defines a path along which all strings in the segment are imaged. If you specify this layout type, you must supply values for the items DDIF\$\_SGA\_LAYPTH\_PATH through DDIF\$\_SGA\_LAYPTH\_V\_ALIGN.

### **DDIF\$K\_RELATIVE\_LAYOUT**

Indicates that the text is positioned relative to the current text position. If you specify this layout type, you must supply values for the items DDIF\$\_SGA\_LAYREL\_H\_RATIO\_N through DDIF\$\_SGA\_LAYREL\_V\_CONSTANT.

### **DDIF\$K\_POSITION\_LAYOUT**

Specifies the position of the segment relative to the current baseline. If you specify this layout type, you must supply a value for the item DDIF\$\_SGA\_LAYPOS\_TEXT\_POSITION.

An optional layout indicator that indicates the layout type.

## DDIF\$\_SGA Galley-Based Layout

---

### Galley-Based Layout

The galley-based layout is selected by specifying DDIF\$\_SGA\_LAYOUT\_C as DDIF\$\_K\_GALLEY\_LAYOUT.

Refer to these corresponding syntax diagrams:

| Syntax     | Location    |
|------------|-------------|
| TextLayout | Figure B-16 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_LAYGLY\_WRAP**

**Encoding:** *handle of a DDIF\$\_LW1 aggregate*

An optional wrap attributes item that indicates the wrap attributes of the galley layout. For more information, see the description of the DDIF\$\_LW1 aggregate.

#### **DDIF\$\_SGA\_LAYGLY\_LAYOUT**

**Encoding:** *handle of a DDIF\$\_LL1 aggregate*

An optional galley layout item that specifies the general layout attributes. For more information, see the description of the DDIF\$\_LL1 aggregate.

---

## Path-Based Layout

The path-based layout is selected by specifying DDIF\$\_SGA\_LAYOUT\_C as DDIF\$K\_PATH\_LAYOUT.

Refer to these corresponding syntax diagrams:

| Syntax       | Location    |
|--------------|-------------|
| TextLayout   | Figure B-16 |
| StringLayout | Figure B-17 |
| Format       | Figure B-50 |
| Angle        | Figure B-66 |
| RightAngle   | Figure B-71 |

---

## AGGREGATE ITEMS

### **DDIF\$ \_SGA\_LAYPTH\_PATH**

**Encoding:** *sequence of DDIF\$ \_PTH aggregates*

A layout path item that identifies the path along which strings are imaged. For more information, see the description of the DDIF\$ \_PTH aggregate.

### **DDIF\$ \_SGA\_LAYPTH\_FORMAT**

**Encoding:** *enumeration; valid values are as follows:*

|                              |  |
|------------------------------|--|
| DDIF\$K_FMT_FLUSH_PATH_BEGIN | The first character is imaged at the start of the text path, and successive characters are imaged at successive positions determined by the escapement of the characters imaged. If the string layout path is shorter than the text string in this case, the path is extended tangent to the slope at the end of the path from the end of the path to the frame clipping outline.  |
| DDIF\$K_FMT_CENTER_OF_PATH   | The length of text strings, as given by the sum of the character escapements, is subtracted from the length of the path; the remaining space is evenly distributed between the first character and the start of the path, and the last character and the end of the path. If the string layout path is shorter than the text string in this case, the text is forced onto the path by reducing the escapement of the characters in the string. |

## DDIF\$\_SGA Path-Based Layout

**DDIF\$K\_FMT\_FLUSH\_PATH\_END** The text string is imaged such that the right alignment point of the last character is aligned with the end of the text string when normal escapement is applied. If the string layout path is shorter than the text string in this case, the path is extended tangent to the beginning of the path, from the beginning of the path to the frame clipping outline.

**DDIF\$K\_FMT\_FLUSH\_PATH\_BOTH** The text string is imaged such that the left alignment point of the first character is aligned with the start of the text path, and the right alignment point of the last character is aligned with the end of the path. If the string layout path is shorter than the text string in this case, the text will be forced onto the path by reducing the escapement of the characters in the string.

A layout format item that specifies the format of text strings along the string path. The start and end points of the path define the end points for justification. The default is **DDIF\$K\_FMT\_FLUSH\_PATH\_BEGIN**.

### **DDIF\$\_SGA\_LAYPTH\_ORIENTATION\_C**

**Encoding: enumeration; valid values are as follows:**

**DDIF\$K\_PATH\_FIXED** The characters are oriented at a fixed angle relative to the current frame. In this case, the **DDIF\$\_SGA\_LAYPTH\_ORIENTATION** item is encoded as a single-precision floating-point value.

**DDIF\$K\_PATH\_RELATIVE** The characters are oriented at an angle that is relative to the slope of the path at the point at which the character is imaged. In this case, the **DDIF\$\_SGA\_LAYPTH\_ORIENTATION** item is encoded as an enumeration.

A layout path orientation indicator that selects the format used to specify the orientation of characters along the path. The default is **DDIF\$\_PATH\_RELATIVE**.

### **DDIF\$\_SGA\_LAYPTH\_ORIENTATION**

**Encoding: variable; valid values are as follows:**

**DDIF\$K\_RIGHT\_ANGLE\_RIGHT** An angle at 0 degrees with respect to the current coordinate system

**DDIF\$K\_RIGHT\_ANGLE\_LEFT** An angle at 180 degrees with respect to the current coordinate system

**DDIF\$K\_RIGHT\_ANGLE\_UP** An angle at 90 degrees with respect to the current coordinate system

**DDIF\$K\_RIGHT\_ANGLE\_DOWN** An angle at 270 degrees with respect to the current coordinate system

A layout path orientation item that specifies the actual character orientation along the path. The default is **DDIF\$K\_RIGHT\_ANGLE\_UP**. If **DDIF\$\_SGA\_LAYPTH\_ORIENTATION\_C** was specified as **DDIF\$K\_PATH\_RELATIVE**, the following values are valid:

### **DDIF\$ \_SGA\_LAYPTH\_H\_ALIGN**

**Encoding: enumeration; valid values are as follows:**

|                                |   |
|--------------------------------|---|
| DDIF\$K_PATH_NORMAL_HORIZONTAL | Characters are horizontally aligned relative to the active position using the value defined for normal horizontal alignment in Table 4–3. |
| DDIF\$K_PATH_LEFTLINE          | Characters are horizontally aligned such that the active position is a point on the left line of the character.                           |
| DDIF\$K_PATH_CENTERLINE        | Characters are horizontally aligned such that the active position is a point on the center line of the character.                         |
| DDIF\$K_PATH_RIGHTLINE         | Characters are horizontally aligned such that the active position is a point on the right line of the character.                          |

A horizontal alignment item that specifies the horizontal alignment point for characters along a path. The default is DDIF\$K\_PATH\_NORMAL\_HORIZONTAL.

When the layout path horizontal alignment (DDIF\$ \_SGA\_LAYPTH\_H\_ALIGN) is defined as normal (DDIF\$K\_PATH\_NORMAL\_HORIZONTAL), the horizontal alignment depends on the layout path orientation. The following table shows the normal horizontal alignments for the various orientations.

**Table 4–3: Normal Horizontal Alignment**

| Orientation          | Horizontal Alignment |
|----------------------|----------------------|
| UP                   | LEFTLINE             |
| RIGHT                | CENTERLINE           |
| DOWN                 | RIGHTLINE            |
| LEFT                 | CENTERLINE           |
| <i>angle</i> (fixed) | CENTERLINE           |

### **DDIF\$ \_SGA\_LAYPTH\_V\_ALIGN**

**Encoding: enumeration; valid values are as follows:**

|                              |  |
|------------------------------|--|
| DDIF\$K_PATH_NORMAL_VERTICAL | The character is aligned using the value defined for normal vertical alignment in Table 4–4. |
| DDIF\$K_PATH_BASELINE        | Characters are vertically aligned such that the active position is a point on the baseline.  |
| DDIF\$K_PATH_CAPLINE         | Characters are vertically aligned such that the active position is a point on the cap line.  |

# DDIF\$\_SGA

## Path-Based Layout

- DDIF\$K\_PATH\_BOTTOMLINE Characters are vertically aligned such that the active position is a point on the bottom line.
- DDIF\$K\_PATH\_HALFLINE Characters are vertically aligned such that the active position is a point on the half line.
- DDIF\$K\_PATH\_TOPLINE Characters are vertically aligned such that the active position is a point on the top line.

A vertical alignment item that specifies the vertical alignment point for characters along a path. The default is DDIF\$K\_PATH\_NORMAL\_VERTICAL.

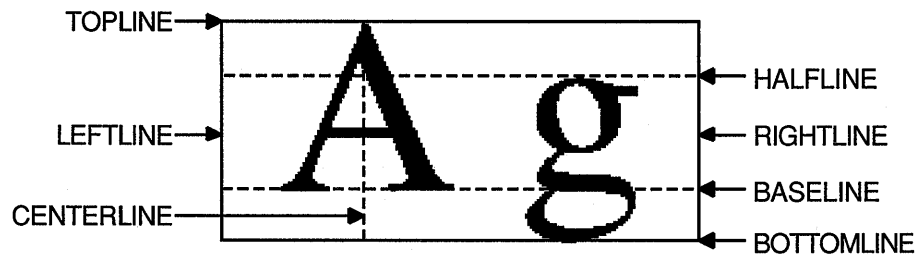
When the layout path vertical alignment (DDIF\$\_SGA\_LAYPTH\_V\_ALIGN) is defined as normal (DDIF\$K\_PATH\_NORMAL\_VERTICAL), the vertical alignment depends on the layout path orientation. The following table shows the normal vertical alignments for the various orientations.

**Table 4-4: Normal Vertical Alignment**

| Orientation          | Vertical Alignment |
|----------------------|--------------------|
| UP                   | BASELINE           |
| RIGHT                | BOTTOMLINE         |
| DOWN                 | BASELINE           |
| LEFT                 | TOPLINE            |
| <i>angle</i> (fixed) | HALFLINE           |

Figure 4-1 shows the orientation of upper- and lowercase characters relative to the values for horizontal and vertical alignment.

**Figure 4-1: Character Orientation**



ZK-1269A-GE

---

## Position-Relative Layout

Position-relative layout specifies that the characters in the segment are positioned relative to the current text position. This layout type is selected by specifying DDIF\$ \_SGA\_LAYOUT\_C as DDIF\$K\_RELATIVE\_LAYOUT.

Refer to these corresponding syntax diagrams:

| Syntax     | Location    |
|------------|-------------|
| TextLayout | Figure B-16 |
| Escapement | Figure B-47 |

---

## AGGREGATE ITEMS

### **DDIF\$ \_SGA\_LAYREL\_H\_RATIO\_N**

**Encoding: integer**

An escapement ratio numerator item that specifies the magnitude of a ratio, which multiplies the em-space width for the current font. The width of an em space is often the same as the width of the capital letter M, but this depends on the font. The default value is 1, if this item is missing and if the numerator is present.

If both the numerator and the denominator are not specified, the ratio is not the default value, but is instead considered optionally absent.

### **DDIF\$ \_SGA\_LAYREL\_H\_RATIO\_D**

**Encoding: integer**

A horizontal ratio denominator item that specifies the units of precision used in the escapement ratio to be used in determining the horizontal position of the character relative to the current text. The default value is 100, if this item is missing and if the numerator is specified.

If both the numerator and the denominator are not specified, the ratio is not the default value, but is instead considered optionally absent.

### **DDIF\$ \_SGA\_LAYREL\_H\_CONSTANT\_C**

**Encoding: measurement enumeration**

A relative horizontal position constant indicator that indicates whether the horizontal position is specified as a variable or constant value.

### **DDIF\$ \_SGA\_LAYREL\_H\_CONSTANT**

**Encoding: variable**

A relative horizontal position constant item that specifies a constant measurement to be used as an escapement.

### **DDIF\$ \_SGA\_LAYREL\_V\_RATIO\_N**

**Encoding: integer**

An escapement ratio numerator item that specifies the magnitude of a ratio, which multiplies the line spacing for the current font. The default value is 1, if this item is missing and if the numerator is present.

## **DDIF\$\_SGA**

### **Position-Relative Layout**

If both the numerator and the denominator are not specified, the ratio is not the default value, but is instead considered optionally absent.

#### ***DDIF\$\_SGA\_LAYREL\_V\_RATIO\_D***

***Encoding: integer***

A vertical ratio denominator item that specifies the units of precision used in the escapement ratio to be used in determining the vertical position of the character relative to the current text. The default value is 100, if this item is missing and if the numerator is specified.

If both the numerator and the denominator are not specified, the ratio is not the default value, but is instead considered optionally absent.

#### ***DDIF\$\_SGA\_LAYREL\_V\_CONSTANT\_C***

***Encoding: measurement enumeration***

A relative vertical position constant indicator that indicates whether the vertical position is specified as a variable or constant value.

#### ***DDIF\$\_SGA\_LAYREL\_V\_CONSTANT***

***Encoding: variable***

A relative vertical position constant item that specifies a constant measurement to be used as an escapement.



---

## Text Position Layout

The text position layout is selected by specifying DDIF\$ \_SGA\_LAYOUT\_C as DDIF\$K\_POSITION\_LAYOUT.

Refer to these corresponding syntax diagrams:

| Syntax     | Location    |
|------------|-------------|
| TextLayout | Figure B-16 |

---

## AGGREGATE ITEMS

### **DDIF\$ \_SGA\_LAYPOS\_TEXT\_POSITION**

*Encoding: enumeration; valid values are as follows:*

|                                |   |
|--------------------------------|---|
| DDIF\$K_TEXT_POS_BASE          | The text in the segment forms the base for special positions in the segment. The text rests on the current baseline.      |
| DDIF\$K_TEXT_POS_L_SUBSCRIPT   | The right alignment position of the last character of the subscript string is placed at the left subscript position.      |
| DDIF\$K_TEXT_POS_L_SUPERSCRIPT | The right alignment position of the last character of the superscript string is placed at the left superscript position.  |
| DDIF\$K_TEXT_POS_R_SUBSCRIPT   | The left alignment position of the first character of the subscript string is placed at the right subscript position.     |
| DDIF\$K_TEXT_POS_R_SUPERSCRIPT | The left alignment position of the first character of the superscript string is placed at the right superscript position. |
| DDIF\$K_TEXT_POS_TOP_CENTER    | The segment is centered above the total string of the base segment.   |
| DDIF\$K_TEXT_POS_BOTTOM_CENTER | The segment is centered below the total string of the base segment.   |
| DDIF\$K_TEXT_POS_RUBI          | The segment is centered above the total string of the base segment.   |

A text position indicator that indicates the relational position of the segment relative to the current baseline.

There are certain restrictions that must be observed when specifying text position:

- No changes in segment layout can take place within positional layout segments unless those segments are in a frame.
- Frames in positional layout segments must have an inline frame position.
- The base segment must be the first child of the parent segment.

## DDIF\$\_SGA Font Definitions

---

### Font Definitions

The font definitions item specifies a list of fonts for use within the segment.

Refer to these corresponding syntax diagrams:

| Syntax     | Location    |
|------------|-------------|
| FontDefn   | Figure B-49 |
| FontNumber | Figure B-79 |

---

### AGGREGATE ITEMS

#### ***DDIF\$\_SGA\_FONT\_DEFNS***

***Encoding: sequence of DDIF\$\_FTD aggregates***

A font definitions item that specifies a list of fonts defined for use within the segment. (For more information, see the description of the DDIF\$\_FTD aggregate.) Each font definition assigns a number to a font (DDIF\$\_FTD\_NUMBER) by which it is referenced (from DDIF\$\_SGA\_TXT\_FONT) within the segment. Initially, there are no font definitions.

---

## **Pattern Definitions**

The pattern definitions item specifies a list of patterns and solid colors defined for use within the segment.

Refer to these corresponding syntax diagrams:

| <b>Syntax</b> | <b>Location</b> |
|---------------|-----------------|
| PatternDefn   | Figure B-89     |

---

## **AGGREGATE ITEMS**

### ***DDIF\$\_SGA\_PATTERN\_DEFNS***

***Encoding: sequence of DDIF\$\_PTD aggregates***

A pattern definition item that specifies a list of patterns and solid colors defined for use within the segment. For more information, see the description of the DDIF\$\_PTD aggregate.

The initial value for this item is a sequence of pattern definitions (DDIF\$\_PTD aggregates) numbered from 0 through 63 (using the DDIF\$\_PTD\_NUMBER item) and which correspond to the predefined patterns described in Appendix A.

## DDIF\$\_SGA Path Definitions

---

### Path Definitions

The path definitions item specifies a list of predefined paths that can be referenced from within the segment.

Refer to these corresponding syntax diagrams:

| Syntax   | Location    |
|----------|-------------|
| PathDefn | Figure B-83 |

---

### AGGREGATE ITEMS

#### ***DDIF\$\_SGA\_PATH\_DEFNS***

***Encoding: sequence of DDIF\$\_PHD aggregates***

A path definitions item that specifies a list of predefined paths that can be referenced within the segment. For more information, see the description of the DDIF\$\_PHD aggregate. Initially, no paths are defined.

---

## Line-Style Definitions

The line-style definitions item specifies a list of predefined line styles that can be referenced within the document.

Refer to these corresponding syntax diagrams:

| Syntax   | Location    |
|----------|-------------|
| LineDefn | Figure B-87 |

---

## AGGREGATE ITEMS

### ***DDIF\$ \_SGA \_LINE \_STYLE \_DEFNS***

#### ***Encoding: sequence of DDIF\$ \_LSD aggregates***

A line-style definitions item that specifies a list of predefined line styles that can be referenced within the document. This item references the DDIF\$ \_LSD \_NUMBER item. For more information, see the description of the DDIF\$ \_LSD aggregate.

The initial value for this item is a sequence of line style definitions (DDIF\$ \_LSD aggregates) numbered from 1 through 4 (using the DDIF\$ \_LSD \_NUMBER item), as defined in Table 4-5.

**Table 4-5: Line Style**

| Line Style Number | Line Style                      | Repeating Pattern |
|-------------------|---------------------------------|-------------------|
| 1                 | DDIF\$K _SOLID _LINE _STYLE     | 1111              |
| 2                 | DDIF\$K _DASH _LINE _STYLE      | 110               |
| 3                 | DDIF\$K _DOT _LINE _STYLE       | 100               |
| 4                 | DDIF\$K _DASH _DOT _LINE _STYLE | 11010             |

The initial line style is DDIF\$K \_SOLID \_LINE \_STYLE.

## **DDIF\$\_SGA**

### **Content Definitions**

---

## **Content Definitions**

The optional content definitions item specifies a list of content definitions that can be referenced within the segment.

Refer to these corresponding syntax diagrams:

| <b>Syntax</b> | <b>Location</b> |
|---------------|-----------------|
| ContentDefn   | Figure B-60     |

---

## **AGGREGATE ITEMS**

### ***DDIF\$\_SGA\_CONTENT\_DEFNS***

***Encoding: sequence of DDIF\$\_CTD aggregates***

An optional content definitions item that specifies a list of content definitions that can be referenced within the segment. For more information, see the description of the DDIF\$\_CTD aggregate. Initially, there are no content definitions.

---

## **Type Definitions**

The type definitions item specifies a list of segment type definitions that can be referenced within the segment.

Refer to these corresponding syntax diagrams:

| <b>Syntax</b> | <b>Location</b> |
|---------------|-----------------|
| SegTypeDefn   | Figure B-93     |

---

## **AGGREGATE ITEMS**

### ***DDIF\$\_SGA\_TYPE\_DEFNS***

***Encoding: sequence of DDIF\$\_TYD aggregates***

A type definitions item that specifies a list of segment type definitions that can be referenced within the segment. For more information, see the description of the DDIF\$\_TYD aggregate. Initially, there are no type definitions.

## DDIF\$\_SGA Text Attributes

---

### Text Attributes

The text attribute items define the default presentation attributes of text within the segment. The text attribute items fall into the following groups:

- Text mask pattern
- Text font
- Text rendition
- Text size
- Text direction
- Text character decimal alignment
- Text leader attributes
- Text kerning
- Text kerning delta attributes
- Text letter spacing

The items in each of these groups are discussed in the following sections.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| TextAttributes | Figure B-13 |



---

## **Text Mask Pattern**

The text mask pattern item specifies the pattern and color of glyphs, using an index into the current list of patterns.

Refer to these corresponding syntax diagrams:

| <b>Syntax</b>  | <b>Location</b> |
|----------------|-----------------|
| TextAttributes | Figure B-13     |
| PatternNumber  | Figure B-82     |

---

## **AGGREGATE ITEMS**

### ***DDIF\$\_SGA\_TXT\_MASK\_PATTERN***

***Encoding: integer***

A text mask pattern item that specifies the pattern and color of glyphs, using an index into the current list of patterns. In addition to user-defined pattern numbers, several predefined patterns are supplied. These patterns are listed in Appendix A.

The text mask pattern is initialized to DDIF\$K\_PATT\_FOREGROUND, which is defined as solid black.

## DDIF\$\_SGA Text Font

---

### Text Font

The text font item specifies the font in which the text is rendered.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| TextAttributes | Figure B-13 |
| FontNumber     | Figure B-79 |

---

### AGGREGATE ITEMS

#### ***DDIF\$\_SGA\_TXT\_FONT***

***Encoding: integer***

A text font item that specifies the font in which the text is rendered. The text font is a reference to a font number defined by the DDIF\$\_FTD\_NUMBER item in the DDIF\$\_FTD aggregate defined in the current or previous DDIF\$\_SGA aggregate.

The character set specified in the font identifier of the referenced font definition must match the character set of the text content that appears in the segment. The text font is initialized to font number 1. This item references the DDIF\$\_FTD\_NUMBER item.

---

## Text Rendition

The text rendition item specifies one or more text renditions. (A text rendition modifies the appearance of characters or strings.)

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| TextAttributes | Figure B-13 |
| RenditionCode  | Figure B-14 |

---

## AGGREGATE ITEMS

### **DDIF\$\_SGA\_TXT\_RENDITION**

**Encoding: array of type enumeration; valid values are as follows:**

|                        |  |
|------------------------|--|
| DDIF\$K_RND_DEFAULT    | The text is imaged as defined by the current “nonrendition” text presentation attributes, without any additional change in rendition.  |
| DDIF\$K_RND_HIGHLIGHT  | The text is rendered in a higher than normal intensity, or a heavier typeface. This rendition is usually used when the document is intended for a video display device.  |
| DDIF\$K_RND_FAINT      | The text is rendered in a lower than normal intensity. This rendition is usually used when the document is intended for a video display device.  |
| DDIF\$K_RND_ITALIC     | The text is rendered in the italic or slant style of the current font.   |
| DDIF\$K_RND_NORMAL     | The text is rendered in normal intensity.  |
| DDIF\$K_RND_SLOW_BLINK | The intensity of the characters alternates between two states at a relatively slow rate. This is used only for documents intended primarily for video display. The fallback rendition on static displays is text in a different color. |
| DDIF\$K_RND_FAST_BLINK | The intensity of the characters alternates between two states at a relatively high rate. This is used only for documents intended primarily for video display. The fallback rendition on static displays is text in a different color. |
| DDIF\$K_RND_NO_BLINK   | The intensity of the characters is steady.   |
| DDIF\$K_RND_NEGATIVE   | The normal relationship between the text foreground and background color is reversed.  |
| DDIF\$K_RND_POSITIVE   | The text color is not reversed.  |

## DDIF\$\_SGA Text Rendition

|  |  |
|--|--|
| DDIF\$K_RND_CONCEAL  | The text string occupies the same space as usual but the characters are not imaged. Note that underlines, overlines, and cross-outs are not concealed by this attribute.                         |
| DDIF\$K_RND_NO_CONCEAL   | The text is imaged rather than concealed.  |
| DDIF\$K_RND_UNDERLINE  | A line parallel with the text path is drawn under the text. Note that spaces are underlined except when the space is omitted from the presentation form by word wrap and justification software. |
| DDIF\$K_RND_2_UNDERLINE  | The text is underlined twice, with an implementation-defined distance between the lines.   |
| DDIF\$K_RND_NO_UNDERLINE   | Text is not underlined.  |
| DDIF\$K_RND_CROSS_OUT  | A line that is thin compared to the weight of the text is drawn through the string. The location of the line is determined by the implementation.  |
| DDIF\$K_RND_BOX  | The text is enclosed in a box. The size of the box is the smallest that will enclose the text without touching any character.  |
| DDIF\$K_RND_ENCIRCLE   | The text is enclosed in an ellipse or rounded rectangle. The total area of the ellipse is the minimum that will enclose the text without touching any character.                                 |
| DDIF\$K_RND_OVERLINE   | A line is drawn parallel to the text path and above it relative to the text.   |
| DDIF\$K_RND_IDEO_UNDERLINE   | A line parallel to the text path is drawn under the text, or along the right side of text that is presented vertically.  |
| DDIF\$K_RND_IDEO_2_UNDERLINE   | Two lines parallel to the text path are drawn under the text, or along the right side of text that is presented vertically.  |
| DDIF\$K_RND_IDEO_OVERLINE  | A line parallel to the text path is drawn over the text, or along the left side of text that is presented vertically.  |
| DDIF\$K_RND_IDEO_2_OVERLINE  | Two lines parallel to the text path are drawn under the text, or along the left side of text that is presented vertically.   |
| DDIF\$K_RND_IDEO_STRESS  | Characters have ideographic stress markers.  |
| The text rendition item that specifies one or more text renditions. The initial value of this item is DDIF\$K_RND_DEFAULT. |  |

---

## Text Size

The text size attributes specify the height and width of the text in the segment. Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| TextAttributes | Figure B-13 |
| Size           | Figure B-72 |

---

## AGGREGATE ITEMS

### **DDIF\$ \_SGA\_TXT\_HEIGHT\_C**

**Encoding: measurement enumeration**

A text height indicator that indicates whether the text height is specified as a variable or constant value.

### **DDIF\$ \_SGA\_TXT\_HEIGHT**

**Encoding: variable**

A text height item that specifies the height of the text in the segment. The current font of the segment is scaled if the type size specified in its font metrics definition does not equal the text size. The initial value of this item is 200 BMUs (12 points), which is scaled to match the document's units per measure (DDIF\$ \_SBA\_UNITS\_PER\_MEASURE) in the document's root segment.

### **DDIF\$ \_SGA\_TXT\_SET\_SIZE\_N**

**Encoding: integer**

A text size numerator item that specifies the magnitude of the ratio of the actual character width to the design width for the current font at the current text height. This item has a default value of 1, if this item is missing and if the denominator is specified.

If both the numerator and the denominator are not specified, the ratio is not the default value, but is instead considered optionally absent.

### **DDIF\$ \_SGA\_TXT\_SET\_SIZE\_D**

**Encoding: integer**

A text size denominator item that specifies the units of precision used in the character width ratio. This item has a default value of 100, if this item is missing and if the numerator is specified.

If both the numerator and the denominator are not specified, the ratio is not the default value, but is instead considered optionally absent.

## DDIF\$\_SGA Text Direction

---

### Text Direction

The text direction item defines the placement of characters along the current text path with respect to the logical ordering of the characters.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| TextAttributes | Figure B-13 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_TXT\_DIRECTION**

**Encoding: enumeration; valid values are as follows:**

|                          |   |
|--------------------------|---|
| DDIF\$K_TXT_DIR_FORWARD  | The text proceeds in the direction of the path.       |
| DDIF\$K_TXT_DIR_BACKWARD | The text proceeds opposite the direction of the path. |

A text direction item that defines the placement of characters along the current text path with respect to the logical ordering of the characters. The initial value of this item is DDIF\$K\_TXT\_DIR\_FORWARD.

---

## Text Character Decimal Alignment

The text character decimal alignment item specifies the characters in a decimal-aligned tab field on which the alignment occurs.

Refer to these corresponding syntax diagrams:

---

| Syntax         | Location    |
|----------------|-------------|
| TextAttributes | Figure B-13 |

---

### AGGREGATE ITEMS

***DDIF\$ \_SGA\_TXT\_DEC\_ALIGNMENT***  
***Encoding: array of type character string***

A text character decimal alignment item that specifies the characters in a decimal-aligned tab field on which the alignment occurs. The order in which the characters are listed indicates their alignment priority. The initial value of this item contains the following characters:

- Period .
- Comma ,
- Close parenthesis )

# DDIF\$\_SGA

## Text Leader

---

### Text Leader

The optional text leader attributes items describe the presentation attributes of leaders. Leaders are rows of dashes or dots that are used to guide the eye across the page.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| TextAttributes | Figure B-13 |
| LeaderStyle    | Figure B-15 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_TXT\_LEADER\_SPACE\_C**

**Encoding: measurement enumeration**

An optional leader space indicator that indicates whether the leader space is specified as a variable or constant value.

#### **DDIF\$\_SGA\_TXT\_LEADER\_SPACE**

**Encoding: variable**

An optional leader space item that specifies the amount of additional space that is inserted between leader characters. The initial value of this item is 0.

#### **DDIF\$\_SGA\_TXT\_LEADER\_BULLET**

**Encoding: character string**

An optional leader bullet item that specifies the text string, usually a single character, that is used to fill leader space. Characters are selected from the current font. The initial value is a period (.).

#### **DDIF\$\_SGA\_TXT\_LEADER\_ALIGN**

**Encoding: enumeration; valid values are as follows:**

- |                            |  |
|----------------------------|--|
| DDIF\$K_ALIGNED_LEADER     | Leader characters should be aligned.                     |
| DDIF\$K_STAGGERED_LEADER   | The center points of leader characters should alternate. |
| DDIF\$K_NON_ALIGNED_LEADER | No alignment has been selected.                          |

An optional leader alignment item that specifies the alignment of leaders. The initial value is DDIF\$K\_NON\_ALIGNED\_LEADER.

#### **DDIF\$\_SGA\_TXT\_LEADER\_STYLE**

**Encoding: enumeration; valid values are as follows:**

- |                       |  |
|-----------------------|--|
| DDIF\$K_X_RULE_LEADER | Draws a horizontal rule.               |
| DDIF\$K_BULLET_LEADER | Uses the current leader-bullet string. |

An optional leader style item that specifies the type of leader to use. The initial value is DDIF\$K\_BULLET\_LEADER.



---

## Text Kerning

In typesetting, **kerning** is defined as the operation of subtracting the space between two characters so that they appear closer together. This concept is used in proportionally spaced fonts to make the distance between characters appear equal. The text pair kerning item controls whether text in the segment is kerned based on kerning pair tables for the current font.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| TextAttributes | Figure B-13 |

---

## AGGREGATE ITEMS

### ***DDIF\$ \_SGA\_TXT\_PAIR\_KERNING***

***Encoding: Boolean***

A text pair kerning item that specifies a Boolean value that controls whether text in the segment is kerned based on kerning pair tables for the current font. If no kerning pair information is available for the font, all kerning deltas for that font are assumed to be zero. The initial value for this item is false.

## DDIF\$\_SGA Line Attributes

---

### Line Attributes

The line attributes specify such information as line style, line width, line pattern, mask pattern, line ends, and line joins.

Refer to these corresponding syntax diagrams:

---

| Syntax          | Location    |
|-----------------|-------------|
| LineAttributes  | Figure B-28 |
| LineStyleNumber | Figure B-29 |
| LineEndNumber   | Figure B-30 |
| LineJoin        | Figure B-31 |
| Ratio           | Figure B-70 |
| PatternNumber   | Figure B-82 |

---

---

### AGGREGATE ITEMS

#### ***DDIF\$\_SGA\_LIN\_WIDTH\_C***

***Encoding: measurement enumeration***

An optional line width indicator that indicates whether the line width is specified as a variable or constant value.

#### ***DDIF\$\_SGA\_LIN\_WIDTH***

***Encoding: variable***

An optional line width item that specifies the width of the line in Basic Measurement Units (BMUs). A line width value of zero indicates the thinnest visible line width on the display device. The initial value for the DDIF\$\_SGA\_LIN\_WIDTH item is 12.

#### ***DDIF\$\_SGA\_LIN\_STYLE***

***Encoding: integer***

An optional line style item that references a line style definition (DDIF\$\_LSD). Line style numbers 1 through 4 (using the DDIF\$\_LSD\_NUMBER item) are initially defined as listed in Table 4-6. Other line styles may be defined as DDIF\$\_LSD aggregates. Valid values are listed in Table 4-6.

**Table 4-6: Line Style**

---

| Line Style Number | Line Style                  | Repeating Pattern |
|-------------------|-----------------------------|-------------------|
| 1                 | DDIF\$K_SOLID_LINE_STYLE    | 1111              |
| 2                 | DDIF\$K_DASH_LINE_STYLE     | 110               |
| 3                 | DDIF\$K_DOT_LINE_STYLE      | 100               |
| 4                 | DDIF\$K_DASH_DOT_LINE_STYLE | 11010             |

---

The initial line style is DDIF\$K\_SOLID\_LINE\_STYLE.

**DDIF\$ \_SGA \_LIN \_PATTERN \_SIZE \_C**

**Encoding: measurement enumeration**

An optional line pattern size indicator that indicates whether the pattern size is specified as a variable or constant value.

**DDIF\$ \_SGA \_LIN \_PATTERN \_SIZE**

**Encoding: variable**

An optional line pattern size item that specifies the size of the line pattern. The initial value of this item is 24. This item acts as a multiplier for the line pattern specified by DDIF\$ \_LSD \_PATTERN.

**DDIF\$ \_SGA \_LIN \_MASK \_PATTERN**

**Encoding: integer**

An optional line mask pattern that specifies the mask pattern of the line as an index into the current pattern definitions. This item is encoded as an integer. In addition to the user-defined pattern numbers, several predefined patterns are provided. These patterns are illustrated in Appendix A.

The initial line mask pattern is DDIF\$K \_PATT \_BACKGROUND, which is defined as solid black.

**DDIF\$ \_SGA \_LIN \_END \_START**

**Encoding: enumeration; valid values are as follows:**

|                            |   |
|----------------------------|---|
| DDIF\$K _BUTT _LINE _END   | The line begins exactly at the starting point, with a flat end.   |
| DDIF\$K _ROUND _LINE _END  | The line begins with a circle the width of the line centered at the starting point.   |
| DDIF\$K _SQUARE _LINE _END | The line begins with a square the width of the line centered at the starting point.   |
| DDIF\$K _ARROW _LINE _END  | The line begins with a triangular area, with the same mask pattern as the line itself, whose base is three times the width of the line and centered on the starting point of the line. The apex of the triangle is on a line tangent to the direction of the line at its starting point. The distance from the apex to the beginning of the line is equal to the width of the line. |

An optional line-end start item that determines the shape of the line ending at the first point on the path that describes the line. The initial value of this item is DDIF\$K \_ROUND \_LINE \_END.

**NOTE**

The DDIF\$ \_SGA \_LIN \_END \_START and DDIF\$ \_LIN \_END \_FINISH items are only different for lines where one end has an arrow and the other does not.

**DDIF\$ \_SGA \_LIN \_END \_FINISH**

**Encoding: enumeration; valid values are as follows:**

|                           |  |
|---------------------------|--|
| DDIF\$K _BUTT _LINE _END  | The line ends exactly at the end point, with a flat end.                     |
| DDIF\$K _ROUND _LINE _END | The line ends with a circle the width of the line centered at the end point. |

## DDIF\$\_SGA Line Attributes

|                         |  |
|-------------------------|--|
| DDIF\$K_SQUARE_LINE_END | The line ends with a square the width of the line centered at the end point.   |
| DDIF\$K_ARROW_LINE_END  | The line ends with a triangular area, with the same mask pattern as the line itself, whose base is three times the width of the line and centered on the end point of the line. The apex of the triangle is on a line tangent to the direction of the line at its end point. The distance from the apex to the line end is equal to the width of the line. |

An optional line-end finish item that determines the shape of the line ending. The initial value of this item is DDIF\$K\_ROUND\_LINE\_END.

### **DDIF\$\_SGA\_LIN\_END\_SIZE\_C**

**Encoding: measurement enumeration**

An optional line-end size indicator that indicates whether the ending size of the line is specified as a variable or constant value.

### **DDIF\$\_SGA\_LIN\_END\_SIZE**

**Encoding: variable**

An optional line-end size item that specifies the ending size of the line. The initial value of this item is 12.

### **DDIF\$\_SGA\_LIN\_JOIN**

**Encoding: enumeration; valid values are as follows:**

|                           |                                  |
|---------------------------|----------------------------------|
| DDIF\$K_MITERED_LINE_JOIN | The join of the line is mitered. |
| DDIF\$K_ROUNDED_LINE_JOIN | The join of the line is rounded. |
| DDIF\$K_BEVELED_LINE_JOIN | The join of the line is beveled. |

An optional line join item that specifies an integer with defined values that determine the shape of line joins. The initial value of this item is DDIF\$K\_ROUNDED\_LINE\_JOIN.

### **DDIF\$\_SGA\_LIN\_MITER\_LIMIT\_N**

**Encoding: integer**

An optional miter limit numerator item that specifies the magnitude of the allowed ratio between the length of the mitered line joint and the width of the line. When the miter limit is exceeded, the joint is beveled instead. The initial value for this item is 10.

### **DDIF\$\_SGA\_LIN\_MITER\_LIMIT\_D**

**Encoding: integer**

An optional miter limit denominator item that specifies the units of precision of the allowed ratio between the length of the mitered line joint and the width of the line. The initial value for this item is 1.

### **DDIF\$\_SGA\_LIN\_INTERIOR\_PATTERN**

**Encoding: integer**

The line interior pattern item specifies the fill pattern or solid color to be used for objects designated as filled or as having a background, including polylines, arcs, curves, fill area sets, frame borders, and galley borders. In addition to the user-defined pattern numbers, several predefined patterns are provided. These patterns are described in Appendix A.

## **DDIF\$\_SGA Line Attributes**

The initial value for this item is DDIF\$K\_PATT\_BACKGROUND, which is defined as solid white. The application of the fill pattern is controlled by a flag on the object to be filled.

## DDIF\$\_SGA Marker Attributes

---

### Marker Attributes

The marker attributes specify the default presentation attributes for markers within the segment.

Refer to these corresponding syntax diagrams:

| Syntax           | Location    |
|------------------|-------------|
| MarkerAttributes | Figure B-32 |
| MarkerNumber     | Figure B-33 |
| PatternNumber    | Figure B-82 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_MKR\_STYLE**

**Encoding:** enumeration; valid values are as follows:

|                         |                       |
|-------------------------|-----------------------|
| DDIF\$K_DOT_MARKER      | Dot marker            |
| DDIF\$K_PLUS_MARKER     | Plus sign marker      |
| DDIF\$K_ASTERISK_MARKER | Asterisk marker       |
| DDIF\$K_CIRCLE_MARKER   | Circle marker         |
| DDIF\$K_CROSS_MARKER    | Diagonal cross marker |

An optional marker style item that specifies the symbol used as the marker. The marker type is initially defined to be DDIF\$K\_DOT\_MARKER.

#### **DDIF\$\_SGA\_MKR\_MASK\_PATTERN**

**Encoding:** integer

An optional marker mask pattern item that defines an index into the pattern list for markers. In addition to the user-defined pattern numbers, several predefined patterns are provided. These patterns are described in Appendix A. The initial marker mask pattern is DDIF\$K\_PATT\_FOREGROUND, which is defined as solid black.

#### **DDIF\$\_SGA\_MKR\_SIZE\_C**

**Encoding:** measurement enumeration

An optional marker size indicator that indicates whether the marker size is specified as a variable or constant value.

#### **DDIF\$\_SGA\_MKR\_SIZE**

**Encoding:** variable

An optional marker size item that defines the size of markers in BMUs (which can be scaled). The initial marker size is 12.

---

## **Galley Attributes**

Galley attributes apply to galleys defined within a segment. The galley attributes of a segment containing text within the document body do not affect the layout of text. Thus, galley attributes are normally used only in the context of defining galleys in a page frame or in a floating frame that has galleys.

Refer to these corresponding syntax diagrams:

| <b>Syntax</b>    | <b>Location</b> |
|------------------|-----------------|
| GalleyAttributes | Figure B-119    |

---

## **AGGREGATE ITEMS**

### ***DDIF\$\_SGA\_GLY\_ATTRIBUTES***

***Encoding: handle of a DDIF\$\_GLA aggregate***

A galley attributes item that controls the presentation attributes of galleys in a segment. For more information, see the description of the DDIF\$\_GLA aggregate.

## DDIF\$\_SGA Image Attributes

---

### Image Attributes

The image attributes control the default presentation attributes of images within the segment.

Refer to these corresponding syntax diagrams:

| Syntax          | Location    |
|-----------------|-------------|
| ImageAttributes | Figure B-36 |
| ImgLutData      | Figure B-37 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_IMG\_PRIVATE\_DATA**

**Encoding:** *sequence of DDIF\$\_PVT aggregates*

An optional private data item that allows for the inclusion of application-private data needed for the presentation of image data. For more information, see the description of the DDIF\$\_PVT aggregate.

#### **DDIF\$\_SGA\_IMG\_PIXEL\_PATH**

**Encoding:** *integer*

An optional pixel path item that specifies the direction of the pixel capture path along an individual scanline. This integer value corresponds to an angular measure in minutes of an arc with respect to the standard orientation of an image. To ensure compatibility with ISO and CCITT standards, values equivalent to 0, 90, 180, and 270 degrees should be used. The initial value is 0 degrees.

#### **DDIF\$\_SGA\_IMG\_LINE\_PROGRESSION**

**Encoding:** *integer*

An optional line progression item that specifies the direction of scanline capture across the image plane. This integer value corresponds to an angular measure in degrees of an arc with respect to the standard orientation of an image. Legal values for this attribute when using ISL for image processing are 90 (meaning that the lines of the image work their way up the page) or 270 (meaning that the lines of the image work their way down the page in the usual direction). The initial value is 270 degrees, which is equivalent to 16200 minutes.

#### **DDIF\$\_SGA\_IMG\_PP\_PIXEL\_DIST**

**Encoding:** *integer*

An optional pixel path aspect ratio item that specifies the ratio of the distance between pixel centers along the pixel path and along the line progression path. The default ratio is 1:1 or 1.

#### **DDIF\$\_SGA\_IMG\_LP\_PIXEL\_DIST**

**Encoding:** *integer*

An optional line progression path aspect ratio item that specifies the aspect ratio along the line progression path. The initial ratio is 1:1 or 1.



**DDIF\$ \_SGA\_IMG\_BRT\_POLARITY**

**Encoding: enumeration; valid values are as follows:**

DDIF\$K\_ZERO\_MAX\_INTENSITY           Zero represents the maximum intensity.

DDIF\$K\_ZERO\_MIN\_INTENSITY           Zero represents the minimum intensity.

An optional brightness polarity item that is used to interpret the manner in which pixel values represent minimum and maximum intensity; that is, whether a value of 0 represents the minimum or maximum intensity value. The default is DDIF\$K\_ZERO\_MAX\_INTENSITY.

**DDIF\$ \_SGA\_IMG\_GRID\_TYPE**

**Encoding: enumeration; valid values are as follows:**

DDIF\$K\_RECTANGULAR\_GRID           Rectangular grid

DDIF\$K\_HEX\_EVEN\_INDENT           Hexagonal grid with even indentation

DDIF\$K\_HEX\_ODD\_INDENT           Hexagonal grid with odd indentation

An optional grid type item that identifies the physical format of the pixel grid. The initial value is DDIF\$K\_RECTANGULAR\_GRID.

**DDIF\$ \_SGA\_IMG\_SPECTRAL\_MAPPING**

**Encoding: enumeration; valid values are as follows:**

DDIF\$K\_PRIVATE\_MAP           Correlation is privately mapped.

DDIF\$K\_MONOCHROME\_MAP       Correlation is monochrome mapped.

DDIF\$K\_GENERAL\_MAP       Correlation is general multispectral.

DDIF\$K\_LUT\_MAP           Correlation is lookup table mapped.

DDIF\$K\_RGB\_MAP           Correlation is RGB (red/green/blue) mapped.

DDIF\$K\_CMY\_MAP           Correlation is CMY (cyan/magenta/yellow) mapped.

DDIF\$K\_YUV\_MAP           Correlation is YUV mapped.

DDIF\$K\_HSV\_MAP           Correlation is HSV (hue saturation value) mapped.

DDIF\$K\_HLS\_MAP           Correlation is HLS (hue lightness saturation) mapped.

DDIF\$K\_YIQ\_MAP           Correlation is YIQ mapped.

An optional spectral component mapping item that designates the correlation between the physical image data and the spectral components of an image. The initial value of this item is DDIF\$K\_MONOCHROME\_MAP.

**DDIF\$ \_SGA\_IMG\_LOOKUP\_TABLES\_C**

**Encoding: enumeration; valid values are as follows:**

DDIF\$K\_PRIVATE\_LUT           The lookup table contains a sequence of one or more named values, where each named value contains lookup table information that is private to the creator of the document. In this case, DDIF\$ \_SGA\_IMG\_LOOKUP\_TABLES is encoded as a sequence of DDIF\$ \_PVT aggregates.

## DDIF\$\_SGA Image Attributes

**DDIF\$K\_RGB\_LUT** The lookup table contains a sequence of lookup table entries, where each entry describes a lookup table index corresponding to the pixel that it maps, and describes the red, green, and blue intensities that are generated for that pixel. The index corresponds to the integer value of the lookup-table-mapped pixel, and can range in value between 0 and  $2^{16} - 1$ . In this case, DDIF\$\_SGA\_IMG\_LOOKUP\_TABLES is encoded as a sequence of DDIF\$\_RGB aggregates.

An optional lookup table indicator that specifies the type of lookup table to be specified.

### **DDIF\$\_SGA\_IMG\_LOOKUP\_TABLES**

**Encoding: variable**

An optional lookup table item that contains an octet string containing application private lookup tables.

### **DDIF\$\_SGA\_IMG\_COMP\_WAVELENGTH\_C**

**Encoding: enumeration; valid values are as follows:**

**DDIF\$K\_APPLICATION\_WAVELENGTH** Specifies application-specific data for each component. In this case, the DDIF\$\_SGA\_IMG\_COMP\_WAVELENGTH item must be encoded as an array of type string.

**DDIF\$K\_WAVELENGTH\_MEASURE** Specifies a wavelength measure in angstroms that can represent either a single wavelength or the most significant frequency within a range of frequencies. In this case, the DDIF\$\_SGA\_IMG\_COMP\_WAVELENGTH item must be encoded as an array of type integer.

**DDIF\$K\_WAVELENGTH\_BAND\_ID** Specifies the spectral band identification codes that are permitted by the application. In this case, the DDIF\$\_SGA\_IMG\_COMP\_WAVELENGTH item must be encoded as an array of type string.

An optional component wavelength indicator that specifies the wavelength being supplied by the DDIF\$\_SGA\_IMG\_COMP\_WAVELENGTH item.

### **DDIF\$\_SGA\_IMG\_COMP\_WAVELENGTH**

**Encoding: variable**

An optional component wavelength information item that specifies the information selected by DDIF\$\_SGA\_IMG\_COMP\_WAVELENGTH\_C.

## Image Component Space

The image component space attributes describe characteristics of the component space.

Refer to these corresponding syntax diagrams:

| Syntax          | Location    |
|-----------------|-------------|
| ImgCmptSpcAttrs | Figure B-38 |

### AGGREGATE ITEMS

#### **DDIF\$ \_SGA \_IMG \_COMP \_SPACE \_ORG**

**Encoding: enumeration; valid values are as follows:**

- |                             |   |
|-----------------------------|---|
| DDIF\$K _FULL _COMPACTION   | Indicates that all the component bits for a pixel are collected into a single data plane and are adjacent to one another within the physical bit field designated as a single logical pixel. For example, in a 3-3-2 RGB image, a single pixel comprises three bits of red, followed by three bits of green, followed by two bits of blue. The next logical pixel is of identical composition. Aside from possible padding at the end of the component bits for each pixel, this organization implies maximal adjacency between uncompressed pixel component data. This organization always implies that only one data plane exists for each content element.             |
| DDIF\$K _PARTIAL _EXPANSION | Indicates that the component bits for a pixel are spread across multiple data planes in the following manner: the pixel data for each component occupies a separate data plane. This organization only applies to multispectral images. For example, the data for an RGB image can be partitioned such that the first plane contains the red bits for all pixels, the second plane the green bits, and the third plane the blue bits, for a total of three planes.  |
| DDIF\$K _FULL _EXPANSION    | Indicates that the component bits for a pixel are spread across multiple data planes in the following manner: each bit per component exists in a separate data plane, so that the logical index into the pixel data of a single plane physically references a bit field that is a single bit in length, and the logical index into the data plane set references the pixel component bits by order of significance. For example, the data for a 3-3-2 RGB image would occupy eight data planes: three for red, three for green, and two for blue. In this organization, the pixel bits of a gray-scale image could be expanded by significance into separate data planes. |

A component space organization item that designates how the component space data is physically organized. The initial value of this item is DDIF\$K \_FULL \_

## **DDIF\$\_SGA**

### **Image Component Space**

EXPANSION.

#### ***DDIF\$\_SGA\_IMG\_PLANES\_PER\_PIXEL***

***Encoding: integer***

An optional data-planes-per-pixel item that specifies the number of data planes per pixel (and consequently per image) used to span the component space. This integer value corresponds to the number of image data units used to represent a particular image. The initial value is 1.

#### ***DDIF\$\_SGA\_IMG\_PLANE\_SIGNIF***

***Encoding: enumeration; valid values are as follows:***

DDIF\$K\_LSB\_MSB                   Least significant bit to most significant bit

DDIF\$K\_MSB\_LSB                   Most significant bit to least significant bit

An optional data plane significance item that only has meaning for image data organized in Expanded Component Sequential Form. The default is DDIF\$K\_LSB\_MSB.

#### ***DDIF\$\_SGA\_IMG\_NUMBER\_OF\_COMP***

***Encoding: integer***

An optional number-of-components item that specifies the number of spectral components in a multispectral image.

#### ***DDIF\$\_SGA\_IMG\_BITS\_PER\_COMP***

***Encoding: array of type integer***

An optional bits-per-component item that specifies the number of bits used for each image component in a data plane. The sum of all bits per component for all data planes equals the number of bits per pixel. The initial value is a single integer of 1.

---

## **Frame Parameters**

The frame parameters cause the content of the segment to be bounded within a frame whose origin is located at the lower left-hand corner of the frame. The frame parameters fall into the following categories:

- Frame flags
- Frame bounding box
- Frame outline
- Frame clipping
- Frame position
- Frame content transformation
- Frame border attributes
- Frame background color
- Frame galleys

The items used to specify each of these categories are discussed in the following sections. Note that there are no initial frame parameters.

Refer to these corresponding syntax diagrams:

---

| <b>Syntax</b>   | <b>Location</b> |
|-----------------|-----------------|
| FrameParameters | Figure B-51     |

---

## DDIF\$\_SGA Frame Flags

---

### Frame Flags

The optional frame flags item specifies the flags that control the presentation of the frame and/or text around the frame.

Refer to these corresponding syntax diagrams:

| Syntax          | Location    |
|-----------------|-------------|
| FrameParameters | Figure B-51 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_FRM\_FLAGS**

**Encoding: longword**

The optional frame flags item specifies the flags that control the presentation of the frame and/or text around the frame. This item is encoded as a longword. Valid frame flag values are as follows:

ddif\$m\_flow\_around

Document text flows around the path given by the frame outline.

ddif\$m\_frame\_border

A line is drawn around the frame outline using the current line attributes.

ddif\$m\_frame\_background\_fill

The frame is filled with the pattern or color given by the current line interior fill item (DDIF\$\_SGA\_LIN\_INTERIOR\_PATTERN) before the content of the frame is imaged.

---

## Frame Bounding Box

The frame bounding box items specify a rectangular area that outlines the frame and defines the origin of the frame. For image frames, the bounding box is the physical size of the image contained in the frame.

Refer to these corresponding syntax diagrams:

| Syntax          | Location    |
|-----------------|-------------|
| FrameParameters | Figure B-51 |
| BoundingBox     | Figure B-42 |
| Position        | Figure B-69 |

---

## AGGREGATE ITEMS

### **DDIF\$\_SGA\_FRM\_BOX\_LL\_X\_C**

**Encoding: measurement enumeration**

A lower left corner **x** position indicator that indicates whether the lower left corner **x**-coordinate is specified as a variable or constant value.

### **DDIF\$\_SGA\_FRM\_BOX\_LL\_X**

**Encoding: variable**

A lower left corner **x** position item that specifies the **x**-coordinate of the lower left corner of the frame bounding box.

### **DDIF\$\_SGA\_FRM\_BOX\_LL\_Y\_C**

**Encoding: measurement enumeration**

A lower left corner **y** position indicator that indicates whether the lower left corner **y**-coordinate is specified as a variable or constant value.

### **DDIF\$\_SGA\_FRM\_BOX\_LL\_Y**

**Encoding: variable**

A lower left corner **y** position item that specifies the **y**-coordinate of the lower left corner of the frame bounding box.

### **DDIF\$\_SGA\_FRM\_BOX\_UR\_X\_C**

**Encoding: measurement enumeration**

An upper right corner **x** position indicator that indicates whether the upper right corner **x**-coordinate is specified as a variable or constant value.

### **DDIF\$\_SGA\_FRM\_BOX\_UR\_X**

**Encoding: variable**

An upper right corner **x** position item that specifies the **x**-coordinate of the upper right corner of the frame bounding box.

### **DDIF\$\_SGA\_FRM\_BOX\_UR\_Y\_C**

**Encoding: measurement enumeration**

An upper right corner **y** position indicator that indicates whether the upper right corner **y**-coordinate is specified as a variable or constant value.

## **DDIF\$\_SGA**

### **Frame Bounding Box**

***DDIF\$\_SGA\_FRM\_BOX\_UR\_Y***

***Encoding: variable***

An upper right corner y position item that specifies the y-coordinate of the upper right corner of the frame bounding box.



---

## Frame Outline

The optional frame outline item specifies the path to which text flowing around the frame is aligned. The frame border, if any, also follows this outline path.

Refer to these corresponding syntax diagrams:

| <b>Syntax</b>   | <b>Location</b> |
|-----------------|-----------------|
| FrameParameters | Figure B-51     |
| CompositePath   | Figure B-84     |

---

## AGGREGATE ITEMS

### ***DDIF\$\_SGA\_FRM\_OUTLINE***

***Encoding: sequence of DDIF\$\_PTH aggregates***

An optional frame outline item that specifies the path to which text flowing around the frame is aligned. The frame border, if any, also follows this outline path. For more information, see the description of the DDIF\$\_PTH aggregate.

If the frame outline item is not specified, the default path is the path given by the bounding box. The path defined by the frame outline is constrained to fit within the specified bounding box.

## DDIF\$ \_SGA Frame Clipping

---

### Frame Clipping

The optional frame clipping item specifies the clipping path of the frame, specified as a path whose coordinates are relative to the origin (0,0) of the frame.

Refer to these corresponding syntax diagrams:

| Syntax          | Location    |
|-----------------|-------------|
| FrameParameters | Figure B-51 |
| CompositePath   | Figure B-84 |

---

### AGGREGATE ITEMS

#### ***DDIF\$ \_SGA\_FRM\_CLIPPING***

***Encoding: sequence of DDIF\$ \_PTH aggregates***

An optional frame clipping item that specifies the clipping path of the frame, specified as a path whose coordinates are relative to the origin (0,0) of the frame. For more information, see the description of the DDIF\$ \_PTH aggregate.

The path that is specified as the clipping region is constrained to fit within the specified bounding box, and it can be different from the outline. No content is imaged outside the clipping region. The inside of the clipping region is determined by the odd winding rule. (The odd winding rule states that, if a ray is drawn from a point to infinity, the origin of the ray is considered inside the area (and hence will be filled) if it crosses the area border an odd number of times.)

---

## Frame Position

The frame position items specify the fixed or preferred position of the frame relative to the enclosing frame. The frame position information is described by first selecting the type of position, and then specifying the appropriate information for that position type. (The origin of a frame is located at the lower lefthand corner.)

Refer to these corresponding syntax diagrams:

| Syntax                   | Location    |
|--------------------------|-------------|
| FrameParameters          | Figure B-51 |
| Position                 | Figure B-69 |
| XCoordinate              | Figure B-73 |
| YCoordinate              | Figure B-74 |
| Measurement              | Figure B-68 |
| InlineFrameParams        | Figure B-52 |
| Size                     | Figure B-72 |
| GalleyFrameParams        | Figure B-53 |
| Format                   | Figure B-50 |
| GalleyVerticalPosition   | Figure B-54 |
| MarginFrameParams        | Figure B-55 |
| MarginHorizontalPosition | Figure B-56 |

---

## AGGREGATE ITEMS

### **DDIF\$\_SGA\_FRM\_POSITION\_C**

**Encoding: enumeration; valid values are as follows:**

|                      |  |
|----------------------|--|
| DDIF\$K_FRAME_FIXED  | The origin of the frame is placed at the specified position relative to the current frame of reference (a page or a frame). If you specify this position type, you must supply values for the items DDIF\$_SGA_FRMFXD_POSITION_X_C through DDIF\$_SGA_FRMFXD_POSITION_Y.   |
| DDIF\$K_FRAME_INLINE | The origin of the frame is positioned along the current text path. The frame behaves like a character the width of the frame. If you specify this position type, you must supply values for the items DDIF\$_SGA_FRMINL_BASE_OFFSET_C and DDIF\$_SGA_FRMINL_BASE_OFFSET.   |
| DDIF\$K_FRAME_GALLEY | The origin of the frame is placed at a preferred position within the current galley. This type of frame positioning should be specified only for content using galley-based layout. If you specify this position type, you must supply values for the items DDIF\$_SGA_FRMGLY_VERTICAL and DDIF\$_SGA_FRMGLY_HORIZONTAL. |

## **DDIF\$\_SGA**

### **Frame Position**

**DDIF\$K\_FRAME\_MARGIN**      The origin of the frame is placed at a preferred position relative to the current position, but outside the current galley. This type of frame positioning should be specified only for content using galley-based layout. If you specify this position type, you must supply values for the items **DDIF\$\_SGA\_FRMMAR\_BASE\_OFFSET\_C** through **DDIF\$\_SGA\_FRMMAR\_HORIZONTAL**.

A position item that selects the type of frame position to be used. The following sections discuss each of these frame positions.

---

## Fixed Frame

The fixed position frame parameters are selected by specifying DDIF\$\_SGA\_FRM\_POSITION\_C as DDIF\$K\_FRAME\_FIXED.

Refer to these corresponding syntax diagrams:

| Syntax          | Location    |
|-----------------|-------------|
| FrameParameters | Figure B-51 |
| Position        | Figure B-69 |
| XCoordinate     | Figure B-73 |
| YCoordinate     | Figure B-74 |

---

## AGGREGATE ITEMS

### **DDIF\$\_SGA\_FRMFXD\_POSITION\_X\_C**

**Encoding: measurement enumeration**

An **x** position indicator that indicates whether the **x** position is specified as a variable or constant value.

### **DDIF\$\_SGA\_FRMFXD\_POSITION\_X**

**Encoding: variable**

An **x** position item that specifies the **x** position of the origin of the frame.

### **DDIF\$\_SGA\_FRMFXD\_POSITION\_Y\_C**

**Encoding: measurement enumeration**

A **y** position indicator that indicates whether the **y** position is specified as a variable or constant value.

### **DDIF\$\_SGA\_FRMFXD\_POSITION\_Y**

**Encoding: variable**

A **y** position item that specifies the **y** position of the origin of the frame.

## DDIF\$\_SGA Inline Frame

---

### Inline Frame

The inline position frame parameters are selected by specifying DDIF\$\_SGA\_FRM\_POSITION\_C as DDIF\$K\_FRAME\_INLINE.

Refer to these corresponding syntax diagrams:

| Syntax            | Location    |
|-------------------|-------------|
| FrameParameters   | Figure B-51 |
| InlineFrameParams | Figure B-52 |
| Size              | Figure B-72 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_FRMINL\_BASE\_OFFSET\_C**

**Encoding: measurement enumeration**

A base offset indicator that indicates whether the base offset value is specified as a variable or constant value.

#### **DDIF\$\_SGA\_FRMINL\_BASE\_OFFSET**

**Encoding: variable**

A base offset item that specifies the vertical offset of the origin (0,0) of the frame relative to the baseline on which the frame is positioned.

---

## Galley Frame

The galley frame parameters are selected by specifying DDIF\$ \_SGA\_FRM\_POSITION\_C as DDIF\$K\_FRAME\_GALLEY.

Refer to these corresponding syntax diagrams:

| Syntax                 | Location    |
|------------------------|-------------|
| FrameParameters        | Figure B-51 |
| GalleyFrameParams      | Figure B-53 |
| GalleyVerticalPosition | Figure B-54 |
| Format                 | Figure B-50 |

---

## AGGREGATE ITEMS

### **DDIF\$ \_SGA\_FRMGLY\_VERTICAL**

**Encoding: enumeration; valid values are as follows:**

|                              |   |
|------------------------------|---|
| DDIF\$K_FRMGLY_BELOW_CURRENT | The frame is positioned so that the top of the frame is on what would be the next baseline.                           |
| DDIF\$K_FRMGLY_BOTTOM        | The frame is positioned so that the lower edge of the frame is on the lower edge of the galley in which it is imaged. |
| DDIF\$K_FRMGLY_TOP           | The frame is positioned so that the upper edge of the frame is on the upper edge of the galley in which it is imaged. |

A vertical galley frame parameter that defines a standard or private label that specifies the preferred vertical positioning of the lower edge of the frame.

### **DDIF\$ \_SGA\_FRMGLY\_HORIZONTAL**

**Encoding: enumeration; valid values are as follows:**

|                              |  |
|------------------------------|--|
| DDIF\$K_FMT_FLUSH_PATH_BEGIN | The frame's left edge is on the left edge of the galley.   |
| DDIF\$K_FMT_CENTER_OF_PATH   | The frame is centered horizontally in the galley.          |
| DDIF\$K_FMT_FLUSH_PATH_END   | The frame's right edge is on the right edge of the galley. |

A horizontal galley frame parameter that specifies the horizontal position of the frame relative to its reference frame.

## DDIF\$\_SGA Margin Frame

---

### Margin Frame

The margin frame parameters are selected by specifying DDIF\$\_SGA\_FRM\_POSITION\_C as DDIF\$K\_FRAME\_MARGIN.

Refer to these corresponding syntax diagrams:

| Syntax                   | Location    |
|--------------------------|-------------|
| FrameParameters          | Figure B-51 |
| MarginFrameParams        | Figure B-55 |
| MarginHorizontalPosition | Figure B-56 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGA\_FRMMAR\_BASE\_OFFSET\_C**

**Encoding: measurement enumeration**

A margin base offset indicator that indicates whether the base offset is specified as a variable or constant value.

#### **DDIF\$\_SGA\_FRMMAR\_BASE\_OFFSET**

**Encoding: variable**

A margin base offset item that specifies the vertical offset from the current baseline for the lower edge of the frame.

#### **DDIF\$\_SGA\_FRMMAR\_NEAR\_OFFSET\_C**

**Encoding: measurement enumeration**

A margin near offset indicator that indicates whether the horizontal offset is specified as a variable or constant value.

#### **DDIF\$\_SGA\_FRMMAR\_NEAR\_OFFSET**

**Encoding: variable**

A margin near offset item that specifies the horizontal offset from the side of the frame nearest the reference frame to the corresponding side of the reference frame.

#### **DDIF\$\_SGA\_FRMMAR\_HORIZONTAL**

**Encoding: enumeration; valid values are as follows:**

DDIF\$K\_FRMMAR\_CLOSEST\_EDGE

The position of the frame depends on the page side. If the page is a left page, the frame is positioned to the left of the left-most galley; if the page is a right page, the frame is positioned to the right of the right-most galley.



## **DDIF\$ \_SGA Margin Frame**

**DDIF\$K\_FRMMAR\_FURTHEST\_EDGE**

The frame is positioned opposite the page side. If the page is a left page, the frame is positioned to the right of the right-most galley; if the page is a right page, the frame is positioned to the left of the left-most galley.

**DDIF\$K\_FRMMAR\_LEFT**

The frame is positioned so that it is to the left of the left-most galley.

**DDIF\$K\_FRMMAR\_RIGHT**

The frame is positioned so that it is to the right of the right-most galley.

A margin horizontal item that defines a standard or private label that specifies the preferred horizontal position of the lower left corner of the frame.

## DDIF\$\_SGA Frame Content Transformation

---

### Frame Content Transformation

The optional frame content transformation item specifies a transformation to be applied to the coordinates of content element within the frame, but not to the clipping region, outline, or other parameters associated with the frame.

Refer to these corresponding syntax diagrams:

| Syntax          | Location     |
|-----------------|--------------|
| FrameParameters | Figure B-51  |
| Transformation  | Figure B-104 |
| Angle           | Figure B-66  |

---

### AGGREGATE ITEMS

#### ***DDIF\$\_SGA\_FRM\_TRANSFORM***

***Encoding: sequence of DDIF\$\_TRN aggregates***

An optional frame content transformation item that specifies a transformation to be applied to the coordinates of content element within the frame, but not to the clipping region, outline, or other parameters associated with the frame. For more information, see the description of the DDIF\$\_TRN aggregate.

Frame content transformations are normally used when it is desirable to keep the coordinates of the content untransformed while providing the ability to view the content under different transformations. This avoids using repeated transformations on the content that would have the effect of altering the precision of the coordinates due to arithmetic roundoff during matrix multiplication.

---

## **Item Change List**

The segment attributes aggregate supplies an item-change-list item that specifies which attributes, as defined in this segment attributes aggregate, are explicitly defined at this segment level. That is, the item change list is a counted vector of item codes that correspond to those items that are specifically defined in the segment attributes aggregate. The following longwords contain the item codes corresponding to attribute items that are specifically defined in the segment attributes aggregate.

The item change list is encoded as a vector of longwords, the first of which is the length, in bytes, of the remaining portion of the vector. Items that are inherited at this level from either default DDIF values (supplied by the CDA Toolkit), or from attributes defined at higher segment levels, are not referenced in this change list. Also, item codes of empty attributes are not included as part of this list.

A call to the LOCATE ITEM routine returns the length (in bytes) of the item codes in the vector and a pointer to the beginning of the item codes.

Specifically, those item codes that return a status of CDA\$\_NORMAL in response to a call to the LOCATE ITEM routine make up this item change list. By using the item change list, an application can locate only those items in the segment attributes aggregate that are explicitly specified and interesting to the application.

---

## **AGGREGATE ITEMS**

### ***DDIF\$\_SGA\_ITEM\_CHANGE\_LIST***

***Encoding: array of type longword***

An item change list item, in which each longword contains the item code of the corresponding attribute items that are specified on this segment. This item is only valid if DDIF\$\_INHERIT\_ATTRIBUTES is specified as a processing option.

## DDIF\$\_SGB

---

### DDIF\$\_SGB—Segment Binding

The segment binding aggregate defines a variable by its name, and defines the method used to calculate its value. The DDIF\$\_SGB aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_BINDING\_DEFNS.

Refer to these corresponding syntax diagrams:

| Syntax           | Location     |
|------------------|--------------|
| Binding          | Figure B-105 |
| CounterDefn      | Figure B-106 |
| StringExpression | Figure B-110 |
| RecordList       | Figure B-111 |

---

### AGGREGATE FORMAT

| Item Name                    | Item Encoding                     |
|------------------------------|-----------------------------------|
| DDIF\$_SGB_VARIABLE_NAME     | String                            |
| DDIF\$_SGB_VARIABLE_VALUE_C  | Enumeration                       |
| DDIF\$_SGB_CTR_TRIGGER_C     | Enumeration                       |
| DDIF\$_SGB_CTR_TRIGGER       | Variable                          |
| DDIF\$_SGB_CTR_INIT_C        | Expression enumeration            |
| DDIF\$_SGB_CTR_INIT          | Variable                          |
| DDIF\$_SGB_CTR_STYLE         | Sequence of DDIF\$_CTS aggregates |
| DDIF\$_SGB_CTR_TYPE          | Enumeration                       |
| DDIF\$_SGB_COM_STRING_EXPR_C | Array of type enumeration         |
| DDIF\$_SGB_COM_STRING_EXPR   | Array of type variable            |
| DDIF\$_SGB_RCD_LIST          | Sequence of DDIF\$_RCD aggregates |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGB\_VARIABLE\_NAME**

**Encoding:** *string*

A variable name item that specifies the name of the variable being defined.

#### **DDIF\$\_SGB\_VARIABLE\_VALUE\_C**

**Encoding:** *enumeration; valid values are as follows:*

**DDIF\$K\_COUNTER\_VARIABLE**

A variable that counts occurrences of nested segments with a specified tag, or occurrences of designated types of layout objects within nested segments. Note that the value of a counter variable varies within the segment, and cannot be cross-referenced from outside the segment. However, its value can at some point be captured in the definition of computed variables, which can be cross-referenced if the segment has a segment identifier. If you specify this value, you must supply values for the items DDIF\$\_SGB\_CTR\_TRIGGER\_C through DDIF\$\_SGB\_CTR\_TYPE.

**DDIF\$K\_COMPUTED\_VARIABLE**

A variable that has a constant value throughout the segment; its value is the value of the expression at the point of definition. If you specify this value, you must supply values for the items DDIF\$\_SGB\_COM\_STRING\_EXPR\_C and DDIF\$\_SGB\_COM\_STRING\_EXPR.

**DDIF\$K\_LIST\_VARIABLE**

A variable that contains an array of records. If you specify this value, you must supply a value for the item DDIF\$\_SGB\_RCD\_LIST.

A variable value indicator that indicates the type of variable value: counter, computed, or list. Each of these types of variable values is discussed in the following sections, along with its corresponding aggregate items.

## DDIF\$\_SGB Counter Variable Values

---

### Counter Variable Values

Counter variable values are selected by specifying DDIF\$\_SGB\_VARIABLE\_VALUE\_C as DDIF\$K\_COUNTER\_VARIABLE.

Refer to these corresponding syntax diagrams:

| Syntax           | Location     |
|------------------|--------------|
| Binding          | Figure B-105 |
| CounterDefn      | Figure B-106 |
| SegmentTag       | Figure B-101 |
| LayoutObjectType | Figure B-107 |
| Expression       | Figure B-108 |
| CounterStyle     | Figure B-109 |

---

### AGGREGATE ITEMS

#### **DDIF\$\_SGB\_CTR\_TRIGGER\_C**

*Encoding: enumeration; valid values are as follows:*

|                                |  |
|--------------------------------|--|
| DDIF\$K_TAGGED_SEGMENT_TRIGGER | Counts tagged segments. In this case, the DDIF\$_SGB_CTR_TRIGGER item is encoded as a string.  |
| DDIF\$K_LAYOUT_OBJECT_TRIGGER  | Counts layout objects. In this case, the DDIF\$_SGB_CTR_TRIGGER item is encoded as an enumeration that can accept any one of the following values: |
| DDIF\$K_DOCUMENT_LAYOUT_OBJECT | Specifies that document layout objects are to be counted.  |
| DDIF\$K_PAGE_SET_LAYOUT_OBJECT | Specifies that page set layout objects are to be counted.  |
| DDIF\$K_PAGE_LAYOUT_OBJECT     | Specifies that page layout objects are to be counted.  |
| DDIF\$K_FRAME_LAYOUT_OBJECT    | Specifies that frame layout objects are to be counted.   |
| DDIF\$K_BLOCK_LAYOUT_OBJECT    | Specifies that block layout objects are to be counted.   |
| DDIF\$K_LINE_LAYOUT_OBJECT     | Specifies that line layout objects are to be counted.  |

An optional counter trigger indicator that indicates the type of object to be counted.

#### **DDIF\$\_SGB\_CTR\_TRIGGER**

*Encoding: variable*

A counter trigger item that specifies the object to be counted.

## DDIF\$\_SGB Counter Variable Values

### **DDIF\$\_SGB\_CTR\_INIT\_C**

**Encoding: expression enumeration**

A counter initialization indicator that indicates the method used to express the initial value for the counter.

### **DDIF\$\_SGB\_CTR\_INIT**

**Encoding: variable**

A counter initialization item that specifies the initial value for the counter. The default value for this item is 1.

### **DDIF\$\_SGB\_CTR\_STYLE**

**Encoding: sequence of DDIF\$\_CTS aggregates**

An optional counter style item that determines how the counter value should be converted to text for display. For more information, see the description of the DDIF\$\_CTS aggregate.

### **DDIF\$\_SGB\_CTR\_TYPE**

**Encoding: enumeration; valid values are as follows:**

|                               |  |
|-------------------------------|--|
| DDIF\$K_MILITARY_COUNTER      | All variables of this name in the current and parent segments are displayed, separated by text.  |
| DDIF\$K_OFFICE_COUNTER        | Only the value of the variable in the current segment is displayed.  |
| DDIF\$K_PAGE_RELATIVE_COUNTER | This style is never hierarchical, and is reset for every page. Footnote numbering on a per-page basis is an example of page-relative counting. |

A counter type item that determines how nested occurrences of counted objects should be displayed, and on what conditions the counter should be reset to its initial value.

## DDIF\$\_SGB

### Computed Variable Values

---

## Computed Variable Values

Computed variable values are selected by specifying DDIF\$\_SGB\_VARIABLE\_VALUE\_C as DDIF\$K\_COMPUTED\_VARIABLE.

Refer to these corresponding syntax diagrams:

| Syntax           | Location     |
|------------------|--------------|
| Binding          | Figure B-105 |
| StringExpression | Figure B-110 |

---

## AGGREGATE ITEMS

### **DDIF\$\_SGB\_COM\_STRING\_EXPR\_C**

**Encoding: array of type enumeration; valid values are as follows:**

DDIF\$K\_TEXT\_ELEMENT      An element of the expression is a text constant. In this case, DDIF\$\_SGB\_COM\_STRING\_EXPR is encoded as a character string.

DDIF\$K\_VARIABLE\_ELEMENT      An element of the expression is a string representation. In this case, DDIF\$\_SGB\_COM\_STRING\_EXPR is encoded as a string.

A computed string expression indicator that indicates whether an element of the expression is a text constant or a string representation.

### **DDIF\$\_SGB\_COM\_STRING\_EXPR**

**Encoding: array of type variable**

A computed string expression item that specifies the string expression.



---

## List Variable Values

List variable values are selected by specifying DDIF\$\_SGB\_VARIABLE\_VALUE\_C as DDIF\$K\_LIST\_VARIABLE.

Refer to these corresponding syntax diagrams:

| Syntax     | Location     |
|------------|--------------|
| Binding    | Figure B-105 |
| RecordList | Figure B-111 |
| RecordDefn | Figure B-112 |

---

## AGGREGATE ITEMS

### ***DDIF\$\_SGB\_RCD\_LIST***

***Encoding: sequence of DDIF\$\_RCD aggregates***

A record list item that defines a record structure that consists of one or more primitive data types, expressed as references to variables. For more information, see the description of the DDIF\$\_RCD aggregate.

## DDIF\$\_TBS

---

### DDIF\$\_TBS—Tab Stop

The tab stop aggregate defines a set of fields along a text path. The tab stop measurements are always relative to the current path. A tab directive selects the next tab stop beyond the current text position in the current text direction. If no further tab stops are defined, the tab settings are repeated by adding the position of the last tab to each of the defined tab stops. All tab stops are relative to the beginning of the current path as defined by a galley or a string layout. The DDIF\$\_TBS aggregate is referenced by the parent aggregate item DDIF\$\_LL1\_TAB\_STOPS.

Refer to these corresponding syntax diagrams:

---

| Syntax  | Location     |
|---------|--------------|
| TabStop | Figure B-127 |

---

---

### AGGREGATE FORMAT

---

| Item Name                        | Item Encoding           |
|----------------------------------|-------------------------|
| DDIF\$_TBS_HORIZONTAL_POSITION_C | Measurement enumeration |
| DDIF\$_TBS_HORIZONTAL_POSITION   | Variable                |
| DDIF\$_TBS_TYPE                  | Enumeration             |
| DDIF\$_TBS_LEADER                | Character string        |

---

---

### AGGREGATE ITEMS

**DDIF\$\_TBS\_HORIZONTAL\_POSITION\_C**

**Encoding: measurement enumeration**

A tab stop horizontal position indicator that indicates whether the horizontal position of the tab stop is specified as a variable or constant value.

**DDIF\$\_TBS\_HORIZONTAL\_POSITION**

**Encoding: variable**

A tab stop horizontal position item that specifies the position of the tab stop relative to the origin of the current text path.

**DDIF\$\_TBS\_TYPE**

**Encoding: enumeration; valid values are as follows:**

DDIF\$\_K\_LEFT\_TAB

The characters in the tab field are positioned with the left alignment point of the first character at the tab position.

|                     |   |
|---------------------|---|
| DDIF\$K_CENTER_TAB  | The character following the tab directive is positioned such that the center alignment point is on the horizontal position of the tab stop.   |
| DDIF\$K_RIGHT_TAB   | The string of characters is positioned such that the right alignment point of the last character is on the position of the right tab.   |
| DDIF\$K_DECIMAL_TAB | The first decimal point character subsequent to the tab directive is positioned such that the center alignment point of that character is at the horizontal position of the tab stop. |

A tab stop type item that specifies the type of tab stop alignment. The default tab type is DDIF\$K\_LEFT\_TAB.

**DDIF\$\_TBS\_LEADER*****Encoding: character string***

An optional tab stop leader item that specifies an optional leader character to appear repeatedly between the tab directive in the document text and the character following the tab directive.

If no leader character is specified, none appears after that tab. The leader character is presented in the typeface and size attributes of the segment in which the tab directive occurs. Only one character can be specified.

## DDIF\$\_TRN

---

### DDIF\$\_TRN—Transformation

The transformation aggregate provides mapping from one coordinate system to another. It provides the following capabilities:

- Asymmetric scaling
- Symmetric rotation or skewing of the axes
- Translation

The DDIF\$\_TRN aggregate is referenced by the parent aggregate items DDIF\$\_CRF\_TRANSFORM and DDIF\$\_SGA\_FRM\_TRANSFORM. When a sequence of DDIF\$\_TRN aggregates is specified, the resulting transformation is to be formed by concatenating the transformations in the order in which they appear in the sequence.

Refer to these corresponding syntax diagrams:

| Syntax         | Location     |
|----------------|--------------|
| Transformation | Figure B-104 |
| Angle          | Figure B-66  |

---

### AGGREGATE FORMAT

| Item Name              | Item Encoding |
|------------------------|---------------|
| DDIF\$_TRN_PARAMETER_C | Enumeration   |
| DDIF\$_TRN_PARAMETER   | Variable      |

---

### AGGREGATE ITEMS

#### **DDIF\$\_TRN\_PARAMETER\_C**

**Encoding: enumeration; valid values are as follows:**

|                     |  |
|---------------------|--|
| DDIF\$K_X_SCALE     | Indicates the scale factor for <b>x</b> -coordinates. In this case, the DDIF\$_TRN_PARAMETER item is encoded as a single-precision floating-point value.   |
| DDIF\$K_Y_SCALE     | Indicates the scale factor for <b>y</b> -coordinates. In this case, the DDIF\$_TRN_PARAMETER item is encoded as a single-precision floating-point value.   |
| DDIF\$K_X_TRANSLATE | Indicates translation values for <b>x</b> -coordinates. In this case, the DDIF\$_TRN_PARAMETER item is encoded as a single-precision floating-point value. |

|                       |  |   |   |   |   |   |   |   |   |   |
|-----------------------|--|---|---|---|---|---|---|---|---|---|
| DDIF\$K_Y_TRANSLATE   | Indicates translation values for y-coordinates. In this case, the DDIF\$_TRN_PARAMETER item is encoded as a single-precision floating-point value.   |   |   |   |   |   |   |   |   |   |
| DDIF\$K_ROTATE        | Indicates angle of rotation of x- and y-coordinates. In this case, the DDIF\$_TRN_PARAMETER item is encoded as a single-precision floating-point value.  |   |   |   |   |   |   |   |   |   |
| DDIF\$K_SKEW          | Indicates a difference in rotation angle of x- and y-coordinates. In this case, the DDIF\$_TRN_PARAMETER item is encoded as a single-precision floating-point value.   |   |   |   |   |   |   |   |   |   |
| DDIF\$K_MATRIX_2_BY_3 | Indicates two columns of a 3x3 transformation matrix, specified in column order. Given 6 numbers in the order A-B-C-D-E-F, the matrix is as follows: <table style="margin-left: auto; margin-right: auto;"> <tr><td>A</td><td>D</td></tr> <tr><td>B</td><td>E</td></tr> <tr><td>C</td><td>F</td></tr> </table> <p>In this case, the DDIF\$_TRN_PARAMETER item is encoded as an array (with 6 elements) of single-precision floating-point values.</p>                      | A | D | B | E | C | F |   |   |   |
| A                     | D  |   |   |   |   |   |   |   |   |   |
| B                     | E  |   |   |   |   |   |   |   |   |   |
| C                     | F  |   |   |   |   |   |   |   |   |   |
| DDIF\$K_MATRIX_3_BY_3 | Indicates a 3x3 transformation matrix, specified in column order. Given 9 numbers in the order A-B-C-D-E-F-G-H-I, the matrix is as follows: <table style="margin-left: auto; margin-right: auto;"> <tr><td>A</td><td>D</td><td>G</td></tr> <tr><td>B</td><td>E</td><td>H</td></tr> <tr><td>C</td><td>F</td><td>I</td></tr> </table> <p>In this case, the DDIF\$_TRN_PARAMETER item is encoded as an array (with 9 elements) of single-precision floating-point values.</p> | A | D | G | B | E | H | C | F | I |
| A                     | D  | G |   |   |   |   |   |   |   |   |
| B                     | E  | H |   |   |   |   |   |   |   |   |
| C                     | F  | I |   |   |   |   |   |   |   |   |

A transformation parameter indicator that indicates which parameter is being specified by DDIF\$\_TRN\_PARAMETER.

**DDIF\$\_TRN\_PARAMETER****Encoding: variable**

A transformation parameter item that contains the actual value of the translation parameter.

## DDIF\$\_TXT

---

### DDIF\$\_TXT—Latin1 Text Content

The Latin1 text content aggregate contains any text content of your document that uses the Latin1 character set. The DDIF\$\_TXT aggregate is referenced by the parent aggregate items DDIF\$\_CTD\_VALUE and DDIF\$\_SEG\_CONTENT.

Refer to the description of the Latin1-String DDIS defined type in Table B-4.

---

#### AGGREGATE FORMAT

| Item Name          | Item Encoding |
|--------------------|---------------|
| DDIF\$_TXT_CONTENT | String        |

---

#### AGGREGATE ITEMS

**DDIF\$\_TXT\_CONTENT**

**Encoding:** *string*

A Latin1 text content item that indicates that the character set to be used is Latin1.

---

## DDIF\$\_TYD—Type Definition

The segment type definition aggregate defines a labeled set of generic segment attributes for reference from nested segments. The DDIF\$\_TYD aggregate is referenced by the parent aggregate item DDIF\$\_SGA\_TYPE\_DEFNS.

Refer to these corresponding syntax diagrams:

| Syntax            | Location    |
|-------------------|-------------|
| SegTypeDefn       | Figure B-93 |
| TypeDefnLabel     | Figure B-61 |
| SegmentAttributes | Figure B-92 |
| NamedValueList    | Figure B-78 |

---

## AGGREGATE FORMAT

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DDIF\$_TYD_LABEL        | String                            |
| DDIF\$_TYD_PARENT       | String                            |
| DDIF\$_TYD_ATTRIBUTES   | Handle of DDIF\$_SGA aggregate    |
| DDIF\$_TYD_PRIVATE_DATA | Sequence of DDIF\$_PVT aggregates |

---

## AGGREGATE ITEMS

### **DDIF\$\_TYD\_LABEL**

#### **Encoding: string**

A type label item that specifies the label by which the type is referenced. If segment types with the same name are defined in the document, the most recent definition is used. This item is referenced by the DDIF\$\_OCC\_STRUCTURE\_ELEMENT item and by the DDIF\$\_SEG\_SEGMENT\_TYPE item.

### **DDIF\$\_TYD\_PARENT**

#### **Encoding: string**

An optional type parent item that specifies the label of a segment type whose attributes are applied prior to applying the attributes of this type. This item references the DDIF\$\_SEG\_ID item.

### **DDIF\$\_TYD\_ATTRIBUTES**

#### **Encoding: handle of a DDIF\$\_SGA aggregate**

An optional type attributes item that specifies the segment attributes that are applied to segments that reference the type being defined. For more information, see the description of the DDIF\$\_SGA aggregate.

## DDIF\$\_TYD

### ***DDIF\$\_TYD\_PRIVATE\_DATA***

***Encoding: sequence of DDIF\$\_PVT aggregates***

An optional type private data item that specifies the private data associated with the definition. For more information, see the description of the DDIF\$\_PVT aggregate.



---

This chapter provides an overview of the general structure of a DTIF document and then provides detailed references for each DTIF-supported aggregate structure. Chapter 6 provides descriptions of the CFE-supported aggregate structures. Chapter 7 provides descriptions of the ESF-supported aggregate structures.

---

## 5.1 DTIF Document Structure Overview

The structure, content, and display format of a DTIF document differentiate one document from another. However, each document must have a root aggregate, a document descriptor aggregate, a document header aggregate, and at least one table definition aggregate.

Each DTIF aggregate type and its corresponding items are discussed in this chapter.

---

## 5.2 Generic Aggregate Items

In addition to the items defined by each individual aggregate, the CDA Toolkit also supports two “generic” aggregate items that can be specified for every DTIF aggregate described in this chapter. Table 5–1 lists and describes these items.

**Table 5–1: DTIF Generic Aggregate Items**

| Item Name             | Encoding | Meaning   |
|-----------------------|----------|---|
| DTIF\$_USER_CONTEXT   | Longword | Specifies additional longword for user                |
| DTIF\$_AGGREGATE_TYPE | Word     | Specifies the type of the aggregate; a read-only item |

## DTIF\$\_ARD

---

# DTIF\$\_ARD—Array Definition Aggregate

The array definition aggregate contains data that pertains to those applications that need to encode many data points as a single item. Each data point must have the same data type and size (in bytes). The application is responsible for the creation, interpretation, and extraction of individual data points in the array. Sparse encoding of array elements is not supported. Individual data points within the array are not explicitly tagged. This reduces the size of the encoding, but also reduces the interchangeability of the data. The DTIF\$\_ARD aggregate is referenced by the parent aggregate items DTIF\$\_CAT\_DEFAULT\_VALUE, DTIF\$\_CAT\_MISSING\_VALUE, and DTIF\$\_CLD\_VALUE.

Refer to these corresponding syntax diagrams:

---

| Syntax    | Location    |
|-----------|-------------|
| ArrayDefn | Figure C-16 |

---

---

## AGGREGATE FORMAT

---

| Item Name                   | Item Encoding |
|-----------------------------|---------------|
| DTIF\$_ARD_DESCRIPTION      | String        |
| DTIF\$_ARD_ELEM_TYPE_SIZE_C | Enumeration   |
| DTIF\$_ARD_ELEM_TYPE_SIZE   | Variable      |
| DTIF\$_ARD_X_DIMENSION      | Integer       |
| DTIF\$_ARD_Y_DIMENSION      | Integer       |
| DTIF\$_ARD_Z_DIMENSION      | Integer       |
| DTIF\$_ARD_VALUES           | String        |

---

---

## AGGREGATE ITEMS

### ***DTIF\$\_ARD\_DESCRIPTION***

**Encoding:** *string*

An optional descriptor item that describes the array.

### ***DTIF\$\_ARD\_ELEM\_TYPE\_SIZE\_C***

**Encoding:** *enumeration*

An array element type size indicator that specifies the type and size in bytes of the value chosen for each data value encoded in the array. Valid values for this item are as follows:

|                   |   |
|-------------------|---|
| DTIF\$_K_STD_TYPE | Defines standard data types. In this case, the DTIF\$_ARD_ELEM_TYPE_SIZE item is encoded as an enumeration.                       |
| DTIF\$_K_VAR_TYPE | Defines the size (in bytes) of nonstandard data types. In this case, the DTIF\$_ARD_ELEM_TYPE_SIZE item is encoded as an integer. |

## **DTIF\$\_ARD\_ELEM\_TYPE\_SIZE**

### **Encoding: variable**

A value item that contains the actual element type size for the type selected in the previous item. If you specified DTIF\$\_ARD\_ELEM\_TYPE\_SIZE\_C as DTIF\$\_K\_STD\_TYPE, you must select a value from the following list.

|                      |                      |
|----------------------|----------------------|
| DTIF\$_K_ELEM_WORD   | Word                 |
| DTIF\$_K_ELEM_LONG   | Longword             |
| DTIF\$_K_ELEM_FFLOAT | VAX F-floating-point |
| DTIF\$_K_ELEM_DFLOAT | VAX D-floating-point |
| DTIF\$_K_ELEM_GFLOAT | VAX G-floating-point |
| DTIF\$_K_ELEM_HFLOAT | VAX H-floating-point |

## **DTIF\$\_ARD\_X\_DIMENSION**

### **Encoding: integer**

A value item that is the **x** dimension of the array.

## **DTIF\$\_ARD\_Y\_DIMENSION**

### **Encoding: integer**

An optional value item that is the **y** dimension of the array.

## **DTIF\$\_ARD\_Z\_DIMENSION**

### **Encoding: integer**

An optional value item that is the **z** dimension of the array.

## **DTIF\$\_ARD\_VALUES**

### **Encoding: string**

An array values item that is the array of values.

## DTIF\$\_CAT

---

### DTIF\$\_CAT—Column Attributes Aggregate

The column attributes aggregate contains data describing the number and types of columns within a table at either the generic- or table-column level. The DTIF\$\_CAT aggregate is referenced by the parent aggregate items DTIF\$\_HDR\_GENERIC\_COLUMNS and DTIF\$\_TMD\_COLUMNS.

Refer to these corresponding syntax diagrams:

| Syntax      | Location    |
|-------------|-------------|
| ColAttrList | Figure C-18 |
| Datatype    | Figure C-19 |

---

### AGGREGATE FORMAT

| Item Name                  | Item Encoding                     |
|----------------------------|-----------------------------------|
| DTIF\$_CAT_NAME            | String                            |
| DTIF\$_CAT_ID              | Integer                           |
| DTIF\$_CAT_APPL_PRIVATE    | Sequence of DTIF\$_NVL aggregates |
| DTIF\$_CAT_GENERIC_REF     | Integer                           |
| DTIF\$_CAT_DESCRIPTION     | Array of type character string    |
| DTIF\$_CAT_FORMATS         | Sequence of DTIF\$_FMI aggregates |
| DTIF\$_CAT_COMPUTED_BY     | Handle of CFE\$_EXP aggregate     |
| DTIF\$_CAT_DEFAULT_VALUE_C | Enumeration                       |
| DTIF\$_CAT_DEFAULT_VALUE   | Variable                          |
| DTIF\$_CAT_MISSING_VALUE_C | Enumeration                       |
| DTIF\$_CAT_MISSING_VALUE   | Variable                          |
| DTIF\$_CAT_QUERY_NAME      | Character string                  |
| DTIF\$_CAT_COLUMN_HDR      | Character string                  |
| DTIF\$_CAT_DATA_TYPE       | Enumeration                       |
| DTIF\$_CAT_DATA_LENGTH     | Integer                           |
| DTIF\$_CAT_SCALE_FACTOR    | Integer                           |
| DTIF\$_CAT_FLAGS           | Longword                          |

---

### AGGREGATE ITEMS

#### **DTIF\$\_CAT\_NAME**

**Encoding:** *string*

An optional identifier item that uniquely identifies this column attribute. A generic- and table-column attribute can have the same name, and different tables

can use the same names. However, all generic column names must be unique, and all column names within each table must be unique. This identifier is for an application program; it is not referenced by any DTIF item.

**DTIF\$\_CAT\_ID****Encoding: integer**

An optional value item that uniquely identifies this column attribute and is used as both a generic attribute and a table column attribute. If DTIF\$\_CAT\_ID is omitted, the value is that of the previous DTIF\$\_CAT\_ID item plus 1. If there is no previous DTIF\$\_CAT\_ID item, the value is 1.

When used as a generic attribute, the value of this item is arbitrary; when used as a table column attribute, this item identifies the table column number, and its value must not exceed that of DTIF\$\_TBL\_MAX\_COLS (if specified). For more information, see the description of the DTIF\$\_TBL aggregate. This item is referenced by the DTIF\$\_CAT\_GENERIC\_REF item and by the DTIF\$\_CLD\_COL\_NUM item.

**DTIF\$\_CAT\_APPL\_PRIVATE****Encoding: sequence of DTIF\$\_NVL aggregates**

An optional private column attribute data item that contains application-private information about the column that is not currently standardized by DTIF. All interpretations of the private data are subject only to private agreements between the parties concerned. For more information, see the description of the DTIF\$\_NVL aggregate.

**DTIF\$\_CAT\_GENERIC\_REF****Encoding: integer**

An optional generic attribute reference item that specifies that this table column inherits attributes from the generic attribute whose DTIF\$\_CAT\_ID matches the value of this item. Note that DTIF\$\_CAT\_GENERIC\_REF is not used when encoding a sequence of DTIF\$\_CAT aggregates for DTIF\$\_HDR\_GENERIC\_COLUMNS. This item references the DTIF\$\_CAT\_ID item.

**DTIF\$\_CAT\_DESCRIPTION****Encoding: array of type character string**

An optional column descriptor item that contains a list of text strings describing this column, its revision history, and restrictions.

**DTIF\$\_CAT\_FORMATS****Encoding: sequence of DTIF\$\_FMI aggregates**

An optional format item that specifies one or more sets of default format attributes for data values. These attributes can be overridden at a lower level. For more information, see the description of the DTIF\$\_FMI aggregate.

When specified in a table column, this item supplies the default formats for all cells in the column.

**DTIF\$\_CAT\_COMPUTED\_BY****Encoding: handle of a CFE\$\_EXP aggregate**

An optional expression item that calculates values for this column, which may be derived from other cell values or columns. For more information, see the description of the CFE\$\_EXP aggregate.

## DTIF\$\_CAT

### **DTIF\$\_CAT\_DEFAULT\_VALUE\_C**

#### **Encoding: enumeration**

An optional default values indicator that specifies the type of default value to be used for null values. The default value is used when no explicit value is specified for cells in the table column (that is, when the cells are empty). Valid values for this item are as follows:

|                            |   |
|----------------------------|---|
| DTIF\$_K_CV_INTEGER        | Indicates an integer value. In this case, the DTIF\$_CAT_DEFAULT_VALUE item is encoded as a variable integer.   |
| DTIF\$_K_CV_LATIN1_TEXT    | Indicates a string value from the Latin1 character set. In this case, the DTIF\$_CAT_DEFAULT_VALUE item is encoded as a string.   |
| DTIF\$_K_CV_SIMPLE_TEXT    | Indicates a character string value from any (single) character set. In this case, the DTIF\$_CAT_DEFAULT_VALUE item is encoded as a character string.   |
| DTIF\$_K_CV_DATE           | Indicates a date string value. In this case, the DTIF\$_CAT_DEFAULT_VALUE item is encoded as the handle of an aggregate of type DTIF\$_DAT. For more information, see the description of the DTIF\$_DAT aggregate.            |
| DTIF\$_K_CV_SCALED_INTEGER | Indicates a scaled integer value. In this case, the DTIF\$_CAT_DEFAULT_VALUE item is encoded as a scaled integer.   |
| DTIF\$_K_CV_VTEXT          | Indicates a varying text string value. In this case, the DTIF\$_CAT_DEFAULT_VALUE item is encoded as the handle of an aggregate of type DTIF\$_VTX. For more information, see the description of the DTIF\$_VTX aggregate.    |
| DTIF\$_K_CV_ARRAY          | Indicates an array value. In this case, the DTIF\$_CAT_DEFAULT_VALUE item is encoded as the handle of an aggregate of type DTIF\$_ARD. For more information, see the description of the DTIF\$_ARD aggregate.                 |
| DTIF\$_K_CV_COMPLEX        | Indicates a complex floating-point value. In this case, the DTIF\$_CAT_DEFAULT_VALUE item is encoded as the handle of an aggregate of type DTIF\$_CFT. For more information, see the description of the DTIF\$_CFT aggregate. |
| DTIF\$_K_CV_FLOAT          | Indicates a floating-point value. In this case, the DTIF\$_CAT_DEFAULT_VALUE item is encoded as a general floating-point value.   |
| DTIF\$_K_CV_BOOLEAN        | Indicates a Boolean value. In this case, the DTIF\$_CAT_DEFAULT_VALUE item is encoded as a Boolean value.   |

### **DTIF\$\_CAT\_DEFAULT\_VALUE**

#### **Encoding: variable**

A default value item that specifies the actual default value for the default value type selected in the previous item.

**DTIF\$\_CAT\_MISSING\_VALUE\_C****Encoding: enumeration**

An optional numeric or character string indicator that specifies the type of missing or null value chosen from those that are delineated for processing. Valid values for this item are as follows:

|                           |   |
|---------------------------|---|
| DTIF\$K_CV_INTEGER        | Indicates an integer value. In this case, the DTIF\$_CAT_MISSING_VALUE item is encoded as a variable integer.   |
| DTIF\$K_CV_LATIN1_TEXT    | Indicates a string value from the Latin1 character set. In this case, the DTIF\$_CAT_MISSING_VALUE item is encoded as a string.   |
| DTIF\$K_CV_SIMPLE_TEXT    | Indicates a character string value from any (single) character set. In this case, the DTIF\$_CAT_MISSING_VALUE item is encoded as a character string.   |
| DTIF\$K_CV_DATE           | Indicates a date string value. In this case, the DTIF\$_CAT_MISSING_VALUE item is encoded as the handle of an aggregate of type DTIF\$_DAT. For more information, see the description of the DTIF\$_DAT aggregate.            |
| DTIF\$K_CV_SCALED_INTEGER | Indicates a scaled integer value. In this case, the DTIF\$_CAT_MISSING_VALUE item is encoded as a scaled integer.   |
| DTIF\$K_CV_VTEXT          | Indicates a varying text string value. In this case, the DTIF\$_CAT_MISSING_VALUE item is encoded as the handle of an aggregate of type DTIF\$_VTX. For more information, see the description of the DTIF\$_VTX aggregate.    |
| DTIF\$K_CV_ARRAY          | Indicates an array value. In this case, the DTIF\$_CAT_MISSING_VALUE item is encoded as the handle of an aggregate of type DTIF\$_ARD. For more information, see the description of the DTIF\$_ARD aggregate.                 |
| DTIF\$K_CV_COMPLEX        | Indicates a complex floating-point value. In this case, the DTIF\$_CAT_MISSING_VALUE item is encoded as the handle of an aggregate of type DTIF\$_CFT. For more information, see the description of the DTIF\$_CFT aggregate. |
| DTIF\$K_CV_FLOAT          | Indicates a floating-point value. In this case, the DTIF\$_CAT_MISSING_VALUE item is encoded as a general floating-point value.   |
| DTIF\$K_CV_BOOLEAN        | Indicates a Boolean value. In this case, the DTIF\$_CAT_MISSING_VALUE item is encoded as a Boolean value.   |

**DTIF\$\_CAT\_MISSING\_VALUE****Encoding: variable**

A missing value item that specifies the actual missing value for the missing or null value type selected in the previous item. This item must be the same data type as DTIF\$\_CAT\_DATA\_TYPE below.

## DTIF\$\_CAT

### ***DTIF\$\_CAT\_QUERY\_NAME***

#### ***Encoding: character string***

An optional column query name item that specifies a shorthand identifier for the column attribute name that can be used in expressions. This item is user-specified and must be unique within each table.

### ***DTIF\$\_CAT\_COLUMN\_HDR***

#### ***Encoding: character string***

An optional column header name item that specifies a more descriptive label to title the column in graphs and reports.

### ***DTIF\$\_CAT\_DATA\_TYPE***

#### ***Encoding: enumeration***

An optional data type item that specifies the data type for this column attribute. DTIF\$\_DAT\_DATA\_TYPE describes a column for creation purposes. Valid values for this item are as follows:

|                     |  |
|---------------------|--|
| DTIF\$_K_DT_UNKNOWN | Unknown  |
| DTIF\$_K_DT_WORD    | Signed word integer  |
| DTIF\$_K_DT_LONG    | Signed longword integer  |
| DTIF\$_K_DT_QUAD    | Signed quadword integer  |
| DTIF\$_K_DT_FFLOAT  | VAX F-floating-point, which is encoded as 4 bytes<br>(bits : 1 sign, 8 exponent, 23 fraction)    |
| DTIF\$_K_DT_DFLOAT  | VAX D-floating-point, which is encoded as 8 bytes<br>(bits : 1 sign, 8 exponent, 55 fraction)    |
| DTIF\$_K_DT_GFLOAT  | VAX G-floating-point, which is encoded as 8 bytes<br>(bits : 1 sign, 11 exponent, 52 fraction)   |
| DTIF\$_K_DT_HFLOAT  | VAX H-floating-point, which is encoded as 16 bytes<br>(bits : 1 sign, 15 exponent, 112 fraction) |
| DTIF\$_K_DT_ABSDATE | Absolute date/time   |
| DTIF\$_K_DT_TEXT    | Text string  |
| DTIF\$_K_DT_VTEXT   | Varying text string  |
| DTIF\$_K_DT_SEGSTR  | Segmented string   |

### ***DTIF\$\_CAT\_DATA\_LENGTH***

#### ***Encoding: integer***

An optional data length item that specifies the number of characters for text data types, or the maximum characters for the DTIF\$\_VTX aggregate items. For other data types, this item is optional. For more information, see the description of the DTIF\$\_VTX aggregate.

### ***DTIF\$\_CAT\_SCALE\_FACTOR***

#### ***Encoding: integer***

An optional scale factor item that specifies the scale factor for integer data type values (DTIF\$\_K\_DT\_WORD, DTIF\$\_K\_DT\_LONG, or DTIF\$\_K\_DT\_QUAD) specified for the DTIF\$\_CAT\_DATA\_TYPE item. DTIF\$\_CAT\_SCALE\_FACTOR describes the column for creation purposes. Cells within a column should apply DTIF\$\_CAT\_SCALE\_FACTOR also. The scale factor is multiplied by the value (value \* factor). The factor's value is a power of 10, so a positive factor value moves the decimal point to the right; a negative value moves it to the left. Although this item is encoded as an integer, only the lowest byte is used.



## **DTIF\$\_CAT\_FLAGS**

### **Encoding: longword**

An optional flags item that specifies whether the column is recomputed whenever a change is made or contains display or descriptive information that should not be updated. This item has the following flag values:

|                                     |   |
|-------------------------------------|---|
| <code>dtif\$m_cat_autorecalc</code> | If set, the column is automatically recomputed whenever a change is made.         |
| <code>dtif\$m_cat_readonly</code>   | If set, the column is for display purposes only and may not be updated.           |
| <code>dtif\$m_cat_annotation</code> | If set, the column comprises label values only and should not be treated as data. |

## DTIF\$\_CCD

---

### DTIF\$\_CCD—DTIF Cell Coordinates Aggregate

The cell coordinates aggregate contains data that is used to define the active, or current, cell within the window; cell ranges; or an expression stored within a cell. The DTIF\$\_CCD aggregate is referenced by the parent aggregate items DTIF\$\_CLR\_RANGE\_BEGIN, DTIF\$\_CLR\_RANGE\_END, and DTIF\$\_WND\_ACTIVE\_LOC.

Refer to these corresponding syntax diagrams:

| Syntax    | Location    |
|-----------|-------------|
| CellCoord | Figure C-36 |
| ColNum    | Figure C-34 |
| RowNum    | Figure C-35 |

---

### AGGREGATE FORMAT

| Item Name         | Item Encoding |
|-------------------|---------------|
| DTIF\$_CCD_ROW    | Integer       |
| DTIF\$_CCD_COLUMN | Integer       |
| DTIF\$_CCD_FLAGS  | Enumeration   |

---

### AGGREGATE ITEMS

#### ***DTIF\$\_CCD\_ROW***

**Encoding:** *integer*

A row identifier item that indicates the row number.

#### ***DTIF\$\_CCD\_COLUMN***

**Encoding:** *integer*

A column identifier item that indicates the column number. In spreadsheet programs, columns are often specified using letter names rather than numbers. In this case, the column number must be assigned in ascending order, beginning with 1.

#### ***DTIF\$\_CCD\_FLAGS***

**Encoding:** *enumeration*

A flags item that indicates whether the row and column references are relative or absolute. Valid values are as follows:

|                       |                              |
|-----------------------|------------------------------|
| DTIF\$K_RELROW_RELCOL | Relative Row/Relative Column |
| DTIF\$K_RELROW_ABSCOL | Relative Row/Absolute Column |
| DTIF\$K_ABSROW_RELCOL | Absolute Row/Relative Column |
| DTIF\$K_ABSROW_ABSCOL | Absolute Row/Absolute Column |

The default is DTIF\$K\_RELROW\_RELCOL.

A relative reference indicates that the reference can be updated to reflect the position of the cell's new location relative to its old location. If the decoding application does not support this concept, it can ignore this item.

An absolute reference to a row (column) indicates that the reference can remain unchanged wherever the cell is being copied to within a table. Absolute references are usually specified in spreadsheet programs by prefixing a character, such as a dollar sign (\$), before the row or column identifier in a cell coordinate name.

## DTIF\$\_CFT

---

### DTIF\$\_CFT—DTIF Complex Float Aggregate

The complex float aggregate models a complex floating-point value. The DTIF\$\_CFT aggregate is referenced by the parent aggregate items DTIF\$\_CAT\_DEFAULT\_VALUE, DTIF\$\_CAT\_MISSING\_VALUE, and DTIF\$\_CLD\_VALUE.

Refer to these corresponding syntax diagrams:

---

| Syntax       | Location    |
|--------------|-------------|
| ComplexFloat | Figure C-17 |

---

### AGGREGATE FORMAT

---

| Item Name                 | Item Encoding          |
|---------------------------|------------------------|
| DTIF\$_CFT_REAL_PART      | General floating-point |
| DTIF\$_CFT_IMAGINARY_PART | General floating-point |

---

### AGGREGATE ITEMS

**DTIF\$\_CFT\_REAL\_PART**

**Encoding: general floating-point**

A value item that specifies the real portion of the complex number.

**DTIF\$\_CFT\_IMAGINARY\_PART**

**Encoding: general floating-point**

A value item that specifies the imaginary portion of the complex number.

---

## DTIF\$\_CLD—DTIF Cell Data Aggregate

The cell data aggregate contains data that pertains to the contents of a particular column in a particular row of a table. Each cell must be encoded only once and cells must be stored in increasing (numeric column) order within a row, that is, first column, second column, . . . , last column.

Empty cells can be omitted from the encoding, allowing sparsely filled rows to be efficiently encoded. It is necessary to encode only those cells that contain data, using the DTIF\$\_CLD\_COL\_NUM item to identify each cell in terms of its column number within the rows.

If a cell is omitted from a table encoding, it is considered *null*. A cell that is null is different from one whose value is 0 (for numeric data, that is). DTIF provides two ways to express null values. One is to omit the cell from the encoding. The second is to encode the cell with the DTIF\$\_K\_CLD\_STATE\_CS\_ISNULL value, which allows the encoding application to leave a placeholder for the cell and which may contain other formatting information.

*Missing values* are similar to null values. A missing value is the value a cell has if it is null, and it is usually used for display purposes. Typically, this value is outside the normal domain of values for cells in this column, so that it will not be confused with supplied data. DTIF provides two ways of expressing missing values: using the DTIF\$\_CAT or DTIF\$\_CLD aggregate. For more information, see the description of the DTIF\$\_CAT aggregate.

The DTIF\$\_CAT\_MISSING\_VALUE item supplies the missing value for all null cells within that column.

The DTIF\$\_CLD\_VALUE item supplies the missing value as the cell value and overrides the missing value supplied by the DTIF\$\_CAT\_MISSING\_VALUE item. When reading the DTIF document, decoding applications first check the DTIF\$\_CLD\_STATE\_CS\_ISNULL value to determine how to interpret the cell value.

If the state is null, use the default value specified for the DTIF\$\_CAT\_DEFAULT\_VALUE item. If the cell value is the same as the DTIF\$\_CAT\_MISSING\_VALUE item, the cell value is considered missing.

If neither a cell- nor column attributes-level missing value is found for a null cell and the decoding application needs to display some value, it may use whatever default is appropriate. The suggested defaults are zero for numeric columns, blanks for text columns, and the base time for date columns.

The DTIF\$\_CLD aggregate is referenced by the parent aggregate item DTIF\$\_ROW\_CELLS.

Refer to these corresponding syntax diagrams:

| Syntax    | Location    |
|-----------|-------------|
| CellData  | Figure C-13 |
| CellValue | Figure C-14 |

## DTIF\$\_CLD

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DTIF\$_CLD_COL_NUM      | Integer                           |
| DTIF\$_CLD_STATE        | Enumeration                       |
| DTIF\$_CLD_DESCRIPTION  | Array of type character string    |
| DTIF\$_CLD_APPL_PRIVATE | Sequence of DTIF\$_NVL aggregates |
| DTIF\$_CLD_FORMATS      | Sequence of DTIF\$_FMI aggregates |
| DTIF\$_CLD_VALUE_C      | Enumeration                       |
| DTIF\$_CLD_VALUE        | Variable                          |
| DTIF\$_CLD_FORMULA_CFE  | Handle of CFE\$_EXP aggregate     |

---

### AGGREGATE ITEMS

#### ***DTIF\$\_CLD\_COL\_NUM***

##### ***Encoding: integer***

An optional column number item that specifies the column number of this cell item in the DTIF\$\_CAT aggregate. Column numbers begin with 1 and increase sequentially by 1. If not present, this item is derived from the previous cell (previous column number plus 1). If not specified, the first value of this item in a row defaults to 1. This item must correspond to the DTIF\$\_CAT\_ID item in the DTIF\$\_CAT aggregate. For more information, see the description of the DTIF\$\_CAT aggregate.

#### ***DTIF\$\_CLD\_STATE***

##### ***Encoding: enumeration***

An optional cell state item that determines whether the cell has a valid value or is empty. Valid values are as follows:

DTIF\$K\_CS\_ISVALUE

If set, the cell contains a formula, a value, or both.

DTIF\$K\_CS\_ISNULL

If set, the cell contains no value or formula. For calculation purposes, the cell is considered empty, and a default value can be applied to the cell. The cell can contain other information, such as format, that does not influence calculation.

DTIF\$K\_CS\_ISERROR

If set, the cell evaluates to an ERROR value. This state is selected unless one of the additional error states (see below) that identifies the specific type of error is selected.

**DTIF\$\_K\_CS\_ISNOVALUE**

If set, the cell has a formula, but no specific value. This state can occur in certain spreadsheets, if the user has disabled auto-recalculation, created a new cell, and then saved the grid without enabling auto-recalculation. The spreadsheet saves the cell formula but has no cell value to save. This state can also be selected if the cell contains a formula that does not evaluate to a single value, such as a financial function or database query function that returns a pointer to a stream of values. Decoding applications that do not wish to recompute cell values, or that do not support multivalued cells, can treat cells of this nature as null (as if the DTIF\$\_K\_CS\_ISNULL state were selected).

The default value is DTIF\$\_K\_CS\_ISVALUE.

If a cell evaluates to one of the following predefined error conditions, the appropriate error state is selected. Otherwise, the DTIF\$\_K\_CS\_ISERROR value (see above) is selected:

**DTIF\$\_K\_CS\_ISUNDERFLOW**

If selected, the error is due to a numeric underflow condition.

**DTIF\$\_K\_CS\_ISOVERFLOW**

If selected, the error is due to a numeric overflow condition.

**DTIF\$\_K\_CS\_ISUNDEFREF**

If selected, the cell references an undefined cell and cannot be evaluated. Note that this may not be considered an error by some applications.

**DTIF\$\_K\_CS\_ISDIVZERO**

If selected, the error is due to a divide by zero condition.

**DTIF\$\_K\_CS\_ISRECURSIVE**

If selected, the cell formula participates in a recursive expression. Note that this may not be considered an error by some applications.

**DTIF\$\_CLD\_DESCRIPTION****Encoding: array of type character string**

An optional cell descriptor item that describes the data contained in the cell. This item can be displayed by applications as a comment or note stored with a cell.

**DTIF\$\_CLD\_APPL\_PRIVATE****Encoding: sequence of DTIF\$\_NVL aggregates**

An optional private cell data item that contains application-private data about the cell not currently standardized by DTIF. All interpretations of the private data are subject only to private agreements between the parties concerned. For more information, see the description of the DTIF\$\_NVL aggregate.

**DTIF\$\_CLD\_FORMATS****Encoding: sequence of DTIF\$\_FMI aggregates**

An optional format item that specifies the default format attributes for this cell value. Note that the DTIF\$\_FMI\_WIDTH item should not be specified at cell level. If the DTIF\$\_FMI\_WIDTH item is present at cell level, it should be ignored. For more information, see the description of the DTIF\$\_FMI aggregate.

## DTIF\$\_CLD

### **DTIF\$\_CLD\_VALUE\_C**

#### **Encoding: enumeration**

An optional value data indicator that specifies the type of value chosen for the cell value. Valid values for this item are as follows:

|                            |   |
|----------------------------|---|
| DTIF\$_K_CV_INTEGER        | Indicates an integer value. In this case, the DTIF\$_CLD_VALUE item is encoded as a variable integer.   |
| DTIF\$_K_CV_LATIN1_TEXT    | Indicates a text string value from the Latin1 character set. In this case, the DTIF\$_CLD_VALUE item is encoded as a string.  |
| DTIF\$_K_CV_SIMPLE_TEXT    | Indicates a character string value from any (single) character set. In this case, the DTIF\$_CLD_VALUE item is encoded as a character string.   |
| DTIF\$_K_CV_DATE           | Indicates a date string value. In this case, the DTIF\$_CLD_VALUE item is encoded as the handle of an aggregate of type DTIF\$_DAT. For more information, see the description of the DTIF\$_DAT aggregate.            |
| DTIF\$_K_CV_SCALED_INTEGER | Indicates a scaled integer value. In this case, the DTIF\$_CLD_VALUE item is encoded as a scaled integer.   |
| DTIF\$_K_CV_VTEXT          | Indicates a varying text string value. In this case, the DTIF\$_CLD_VALUE item is encoded as the handle of an aggregate of type DTIF\$_VTX. For more information, see the description of the DTIF\$_VTX aggregate.    |
| DTIF\$_K_CV_ARRAY          | Indicates an array value. In this case, the DTIF\$_CLD_VALUE item is encoded as the handle of an aggregate of type DTIF\$_ARD. For more information, see the description of the DTIF\$_ARD aggregate.                 |
| DTIF\$_K_CV_COMPLEX        | Indicates a complex floating-point value. In this case, the DTIF\$_CLD_VALUE item is encoded as the handle of an aggregate of type DTIF\$_CFT. For more information, see the description of the DTIF\$_CFT aggregate. |
| DTIF\$_K_CV_FLOAT          | Indicates a floating-point value. In this case, the DTIF\$_CLD_VALUE item is encoded as a general floating-point value.   |
| DTIF\$_K_CV_BOOLEAN        | Indicates a Boolean value. In this case, the DTIF\$_CLD_VALUE item is encoded as a type Boolean.  |

### **DTIF\$\_CLD\_VALUE**

#### **Encoding: variable**

A value item that contains the actual data value for the value type selected in the previous item.



***DTIF\$\_CLD\_FORMULA\_CFE******Encoding: handle of a CFE\$\_EXP aggregate***

An optional expression item that calculates the value of the cell and is derived from other cell values. For more information, see the description of the CFE\$\_EXP aggregate.

## DTIF\$\_CLR

---

### DTIF\$\_CLR—DTIF Cell Range Aggregate

The cell range aggregate defines explicit starting and ending cells using two cell references. A single-cell range is indicated by the absence of an ending cell. The DTIF\$\_CLR aggregate is referenced by the parent aggregate item DTIF\$\_RNG\_REGION.

Refer to these corresponding syntax diagrams:

| Syntax    | Location    |
|-----------|-------------|
| CellRange | Figure C-39 |

---

### AGGREGATE FORMAT

| Item Name              | Item Encoding                  |
|------------------------|--------------------------------|
| DTIF\$_CLR_RANGE_BEGIN | Handle of DTIF\$_CCD aggregate |
| DTIF\$_CLR_RANGE_END   | Handle of DTIF\$_CCD aggregate |

---

### AGGREGATE ITEMS

#### **DTIF\$\_CLR\_RANGE\_BEGIN**

**Encoding:** *handle of a DTIF\$\_CCD aggregate*

A range begin item that is the starting cell in the range that corresponds to the upper left cell. For more information, see the description of the DTIF\$\_CCD aggregate.

#### **DTIF\$\_CLR\_RANGE\_END**

**Encoding:** *handle of a DTIF\$\_CCD aggregate*

An optional range end item that is the ending cell in the range that corresponds to the lower right cell. For more information, see the description of the DTIF\$\_CCD aggregate.

---

## DTIF\$\_COR—DTIF Column Range Aggregate

The column range aggregate defines a range using starting and ending column numbers. A column range differs from a cell range in that a column range refers to an indeterminate number of cells. A single-column range is indicated by the absence of an ending column. The DTIF\$\_COR aggregate is referenced by the parent aggregate item DTIF\$\_RNG\_REGION.

Refer to these corresponding syntax diagrams:

| Syntax   | Location    |
|----------|-------------|
| ColRange | Figure C-41 |

---

### AGGREGATE FORMAT

| Item Name            | Item Encoding |
|----------------------|---------------|
| DTIF\$_COR_COL_BEGIN | Integer       |
| DTIF\$_COR_COL_END   | Integer       |

---

### AGGREGATE ITEMS

***DTIF\$\_COR\_COL\_BEGIN***

***Encoding: integer***

A column begin item that specifies the starting column number that corresponds to the first column in the range.

***DTIF\$\_COR\_COL\_END***

***Encoding: integer***

An optional column end item that specifies the ending column number that corresponds to the last column in the range.

## DTIF\$\_DAT

---

### DTIF\$\_DAT—DTIF Date and Time Aggregate

The DTIF date and time aggregate specifies a date/time value that is defined as a sequence of two octet strings. The first octet string defines the date and time as a binary value; the second octet string defines an optional time differential. The DTIF\$\_DAT aggregate is referenced by the parent aggregate items DTIF\$\_CAT\_DEFAULT\_VALUE, DTIF\$\_CAT\_MISSING\_VALUE, DTIF\$\_CLD\_VALUE, DTIF\$\_HDR\_DATE, and DTIF\$\_NVL\_VALUE.

Refer to these corresponding syntax diagrams:

---

| Syntax   | Location    |
|----------|-------------|
| DateTime | Figure C-29 |

---

### AGGREGATE FORMAT

---

| Item Name              | Item Encoding |
|------------------------|---------------|
| DTIF\$_DAT_DATETIME    | String        |
| DTIF\$_DAT_TIME_DIFF_C | Enumeration   |
| DTIF\$_DAT_TIME_DIFF   | Variable      |

---

### AGGREGATE ITEMS

#### ***DTIF\$\_DAT\_DATETIME***

**Encoding:** *string*

A date and time item that is a sequence of octets representing a date/time value. Each octet is interpreted as an unsigned integer value, as shown in the following table.

---

| Octet Number | Date/Time Element  |
|--------------|--|
| 0            | Century digits in the range of values from 0 to 99; for example, 19 in the year 1967 |
| 1            | Year digits in the range of values from 0 to 99; for example, 67 in the year 1967    |
| 2            | Month in the range of values from 1 to 12  |
| 3            | Day in the range of values from 1 to 31  |
| 4            | Hour in the range of values from 0 to 23   |
| 5            | Minute in the range of values from 0 to 59   |
| 6            | Second in the range of values from 0 to 59   |
| 7            | Hundredths of seconds in the range of values from 0 to 99                            |

---

**DTIF\$\_DAT\_TIME\_DIFF\_C****Encoding: enumeration**

An optional time difference indicator that specifies the type of time differential value chosen from those for the time differential item. Valid values for this item are as follows:

|                   |  |
|-------------------|--|
| DTIF\$K.UTC_TIME  | A value that represents Coordinate Universal Time (UTC), or Greenwich Mean Time. This is equivalent to a time differential of 0 hours, 0 seconds. In this case, the DTIF\$_DAT_TIME_DIFF item is left blank. |
| DTIF\$K.PLUS_DIFF | An octet string representing a positive time differential, as shown in the table below. In this case, the DTIF\$_DAT_TIME_DIFF item is encoded as an octet.  |
| DTIF\$K.NEG_DIFF  | An octet string representing a negative time differential, as shown in the table below. In this case, the DTIF\$_DAT_TIME_DIFF item is encoded as an octet.  |

The following table shows the encoding of the 2 octets that represent the encoding of the DAT\_TIME\_DIFF item for the DTIF\$K.PLUS\_DIFF and DTIF\$K.NEG\_DIFF cases. Each octet is interpreted as an unsigned integer value, as shown in the following table.

| Octet Number | Date/Time Element   |
|--------------|---|
| 0            | Hours in the range of values from 0 to 13 for DTIF\$K.PLUS_DIFF and in the range of values from 0 to 12 for DTIF\$K.NEG_DIFF. |
| 1            | Minutes in the range of values from 0 to 59.  |

**DTIF\$\_DAT\_TIME\_DIFF****Encoding: variable**

A time difference item that contains the actual time differential for the time differential type selected in the previous item.

If the DTIF\$\_DAT\_TIME\_DIFF\_C item is present, the DTIF\$\_DAT\_DATETIME item represents the Coordinate Universal Time, and the value chosen for DTIF\$\_DAT\_TIME\_DIFF represents the local time differential.

If the DTIF\$\_DAT\_TIME\_DIFF\_C item is not present, the value specified for the DTIF\$\_DAT\_DATETIME item represents local time.

## DTIF\$\_DSC

---

### DTIF\$\_DSC—Document Descriptor Aggregate

The DTIF document descriptor aggregate specifies the version level of the DTIF encoding and identifies the software that created the document. The DTIF\$\_DSC aggregate is referenced by the parent aggregate item DTIF\$\_DTF\_DESCRIPTOR.

Refer to these corresponding syntax diagrams:

| Syntax             | Location   |
|--------------------|------------|
| DocumentDescriptor | Figure C-2 |

---

### AGGREGATE FORMAT

| Item Name                       | Item Encoding                  |
|---------------------------------|--------------------------------|
| DTIF\$_DSC_MAJOR_VERSION        | Integer                        |
| DTIF\$_DSC_MINOR_VERSION        | Integer                        |
| DTIF\$_DSC_PRODUCT_IDENTIFIER   | String                         |
| DTIF\$_DSC_PRODUCT_NAME         | Array of type character string |
| DTIF\$_DSC_ENCODE_MAJOR_VERSION | Integer                        |
| DTIF\$_DSC_ENCODE_MINOR_VERSION | Integer                        |

---

### AGGREGATE ITEMS

#### ***DTIF\$\_DSC\_MAJOR\_VERSION***

***Encoding: integer***

A major version indicator that specifies the major version number of the encoding application.

#### ***DTIF\$\_DSC\_MINOR\_VERSION***

***Encoding: integer***

A minor version indicator that specifies the minor version number of the encoding application.

#### ***DTIF\$\_DSC\_PRODUCT\_IDENTIFIER***

***Encoding: string***

A product identifier item that contains a registered facility mnemonic representing the software that encoded the DTIF document.

The product identifier can be an acronym or abbreviation for the product name. This identifier is constant across versions of the product. The product identifier string is used to prefix all tags placed in the table by the product.

***DTIF\$ \_DSC\_PRODUCT\_NAME******Encoding: array of type character string***

A product name item that indicates the name of the product that encoded the document.

The product name string contains the version number of the product. The name of the product should be spelled in full.

***DTIF\$ \_DSC\_ENCODE\_MAJOR\_VERSION******Encoding: integer***

An encoding major version indicator (type DTIF\$ \_DSC\_ENCODE\_MAJOR\_VERSION) that acts as the primary indicator of compatibility between DTIF processors and the encoding of the present DTIF document.

The literal DTIF\$K \_MAJOR\_VERSION is defined to represent the highest major version supported by the CDA Toolkit. Applications should use this literal for the encoding major version indicator. On output, the CDA Toolkit supplies the current version if the application has stored a different value. The default value is DTIF\$K \_MAJOR\_VERSION.

***DTIF\$ \_DSC\_ENCODE\_MINOR\_VERSION******Encoding: integer***

An encoding minor version indicator that specifies the minor version number of the DTIF encoding.

The literal DTIF\$K \_MINOR\_VERSION is defined to represent the highest minor version supported by the CDA Toolkit. Applications should use this literal for the encoding minor version indicator. On output, the CDA Toolkit supplies the current version if the application has stored a different value. The default value is DTIF\$K \_MINOR\_VERSION.

## DTIF\$\_DTF

---

### DTIF\$\_DTF—DTIF Document Root Aggregate

The DTIF document root aggregate identifies this particular instance of a DTIF document.

Refer to these corresponding syntax diagrams:

---

| Syntax       | Location   |
|--------------|------------|
| DTIFDocument | Figure C-1 |

---

---

### AGGREGATE FORMAT

---

| Item Name             | Item Encoding                     |
|-----------------------|-----------------------------------|
| DTIF\$_DTF_DESCRIPTOR | Handle of DTIF\$_DSC aggregate    |
| DTIF\$_DTF_HEADER     | Handle of DTIF\$_HDR aggregate    |
| DTIF\$_DTF_TABLES     | Sequence of DTIF\$_TBL aggregates |

---

---

### AGGREGATE ITEMS

#### ***DTIF\$\_DTF\_DESCRIPTOR***

***Encoding: handle of a DTIF\$\_DSC aggregate***

A document descriptor item that describes the document encoding. For more information, see the description of the DTIF\$\_DSC aggregate.

#### ***DTIF\$\_DTF\_HEADER***

***Encoding: handle of a DTIF\$\_HDR aggregate***

A document header item that contains parameters and processing instructions that apply to the document as a whole. For more information, see the description of the DTIF\$\_HDR aggregate.

#### ***DTIF\$\_DTF\_TABLES***

***Encoding: sequence of DTIF\$\_TBL aggregates***

A document tables item that specifies the tables of the document. For more information, see the description of the DTIF\$\_TBL aggregate.



---

## DTIF\$\_ERF—External Reference Aggregate

The external reference aggregate specifies a source of data that is outside the DTIF document. It does so by specifying the data syntax and location of the external reference element. In the body of tables, external references are specified as indexes into the external reference aggregate. Items in this list are referenced within the table by a cell formula containing the CFE\$K\_CELL\_EXTRACT value for CFE\$\_EXL\_EXPR\_C. For more information, see the description of the CFE\$\_EXL aggregate. The first external reference has an index of 1, and subsequent external references are numbered increasing by 1 (2, 3 . . . ). The DTIF\$\_ERF aggregate is referenced by the parent aggregate item DTIF\$\_HDR\_EXTERNAL\_REFERENCES.

Refer to these corresponding syntax diagrams:

| Syntax            | Location   |
|-------------------|------------|
| ExternalReference | Figure C-4 |
| StorageSystemTag  | Figure C-5 |
| ExternalRefIndex  | Figure C-6 |

---

### AGGREGATE FORMAT

| Item Name             | Item Encoding                  |
|-----------------------|--------------------------------|
| DTIF\$_ERF_DATA_TYPE  | Object identifier              |
| DTIF\$_ERF_DESCRIPTOR | Array of type character string |
| DTIF\$_ERF_LABEL      | Character string               |
| DTIF\$_ERF_LABEL_TYPE | String with <b>add-info</b>    |
| DTIF\$_ERF_CONTROL    | Enumeration                    |

---

### AGGREGATE ITEMS

#### **DTIF\$\_ERF\_DATA\_TYPE**

**Encoding:** *object identifier*

A reference data type item that identifies the data type of the external data object.

#### **DTIF\$\_ERF\_DESCRIPTOR**

**Encoding:** *array of type character string*

A reference descriptor item that provides a human-readable description of the data type.

## DTIF\$\_ERF

### **DTIF\$\_ERF\_LABEL**

#### **Encoding: character string**

A reference label item that provides the label by which the user and/or the system identifies the data object.

When interchanging tables between different systems, the decoding application can replace the reference label with one suited for that system. Decoding applications are not required to preserve the label, but rather can ask the receiving user to specify a new label for the data object, or generate one automatically.

### **DTIF\$\_ERF\_LABEL\_TYPE**

#### **Encoding: string with add-info**

A storage item that contains a tag identifying the type of storage system in which the external reference is located. The following table lists the values for **add-info** and the corresponding string values.

| <b>Add-Info</b>            | <b>String</b>  |
|----------------------------|--|
| DTIF\$K_PRIVATE_LABEL_TYPE | The label is a private label. In this case, the string can be any user-specified string.     |
| DTIF\$K_RMS_LABEL_TYPE     | The label is an RMS file specification. In this case, the string must be "\$RMS".            |
| DTIF\$K_UTX_LABEL_TYPE     | The label is an ULTRIX file specification. In this case, the string must be "\$UTX".         |
| DTIF\$K_MDS_LABEL_TYPE     | The label is an MS-DOS or OS/2 file specification. In this case, the string must be "\$MDS". |

### **DTIF\$\_ERF\_CONTROL**

#### **Encoding: enumeration**

A control item that specifies how the referenced data object is treated when the document is transferred from one system to another. Valid values for this item are as follows:

|                           |  |
|---------------------------|--|
| DTIF\$K_COPY_REFERENCE    | The referenced data object is transmitted along with the document and is stored on the receiving system. |
| DTIF\$K_NO_COPY_REFERENCE | The referenced data is not transmitted with the document.  |

The default is DTIF\$K\_COPY\_REFERENCE.

---

## DTIF\$\_EXT—DTIF Application Private Aggregate

The external content aggregate specifies content that is external to the document. The DTIF\$\_EXT aggregate is referenced by the parent aggregate item DTIF\$\_NVL\_VALUE.

Refer to the corresponding EXTERNAL DDIS built-in primitive described in Table B-1.

---

### AGGREGATE FORMAT

| Item Name                        | Item Encoding     |
|----------------------------------|-------------------|
| DTIF\$_EXT_DIRECT_REFERENCE      | Object identifier |
| DTIF\$_EXT_INDIRECT_REFERENCE    | Integer           |
| DTIF\$_EXT_DATA_VALUE_DESCRIPTOR | String            |
| DTIF\$_EXT_ENCODING_C            | Enumeration       |
| DTIF\$_EXT_ENCODING              | Variable          |
| DTIF\$_EXT_ENCODING_L            | Integer           |

---

### AGGREGATE ITEMS

#### ***DTIF\$\_EXT\_DIRECT\_REFERENCE***

***Encoding: object identifier***

An optional direct reference item that is used to identify the data type (syntax and semantics) of the external element. Each use of the external reference can be accompanied by a statement constraining the range of permitted data types.

#### ***DTIF\$\_EXT\_INDIRECT\_REFERENCE***

***Encoding: integer***

An optional indirect reference item. This item is reserved for future standardization.

#### ***DTIF\$\_EXT\_DATA\_VALUE\_DESCRIPTOR***

***Encoding: string***

An optional data value descriptor item that is a text string describing the data value to programs, to people, or to both.

## DTIF\$\_EXT

### **DTIF\$\_EXT\_ENCODING\_C**

#### **Encoding: enumeration**

An encoding indicator that specifies the type of value chosen from those for the method of encoding the data value. Valid values for this item are as follows:

|                            |  |
|----------------------------|--|
| DTIF\$K_DOCUMENT_ENCODING  | Nested document. In this case, the DTIF\$_EXT_ENCODING item is encoded as a document root aggregate.                         |
| DTIF\$K_DDIS_ENCODING      | Nested document. In this case, the DTIF\$_EXT_ENCODING item is encoded as a DIGITAL Data Interchange Syntax (DDIS) encoding. |
| DTIF\$K_OCTET_ENCODING     | Octet-aligned encoding. In this case, the DTIF\$_EXT_ENCODING item is encoded as a string.                                   |
| DTIF\$K_ARBITRARY_ENCODING | Arbitrary. In this case, the DTIF\$_EXT_ENCODING item is encoded as a bit string.  |

### **DTIF\$\_EXT\_ENCODING**

#### **Encoding: variable**

An encoding item that specifies the actual data value for the encoding value type selected in the previous item.

### **DTIF\$\_EXT\_ENCODING\_L**

#### **Encoding: integer**

A read-only encoding length item that specifies the length (on output) of the encoding. The application cannot modify this item.

## DTIF\$\_FMI—Format Information Aggregate

The format information aggregate defines the presentation of a data value, that is, how the value appears when printed or displayed. Format information has no effect upon the calculation or precision of a value.

Format information can be specified at and inherited from several levels within a DTIF document. These levels are listed here in order of increasing precedence: table, window, column, row, and cell. This multilevel scheme allows applications to define default format attributes at, for example, table or column level, and then to override only those attributes that differ at a lower level. Generic format attributes override column format attributes only when referenced by a particular table column. The DTIF\$\_FMI aggregate is referenced by the parent aggregate items DTIF\$\_CAT\_FORMATS, DTIF\$\_CLD\_FORMATS, DTIF\$\_ROW\_FORMATS, DTIF\$\_TMD\_DEFAULT\_FMTS, and DTIF\$\_WND\_FORMATS.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| FormatInfoList | Figure C-20 |
| FormatType     | Figure C-22 |
| NumericFmt     | Figure C-24 |
| FmtPrec        | Figure C-25 |
| TextFmt        | Figure C-26 |
| DateFmt        | Figure C-27 |
| FmtFlags       | Figure C-28 |
| LangPrefIndex  | Figure C-21 |
| EditStrIndex   | Figure C-23 |

## AGGREGATE FORMAT

| Item Name                        | Item Encoding                 |
|----------------------------------|-------------------------------|
| DTIF\$_FMI_WINDOW_ID             | Integer                       |
| DTIF\$_FMI_C                     | Enumeration                   |
| <i>DTIF\$_FMI_NUM_DATATYPE</i>   | Longword                      |
| <i>DTIF\$_FMI_NUM_C</i>          | Enumeration                   |
| <i>DTIF\$_FMI_NUMSTD_TYPE</i>    | Enumeration                   |
| <i>DTIF\$_FMI_NUMSTD_DIGITS</i>  | Integer                       |
| <i>DTIF\$_FMI_NUMSTD_FRAC</i>    | Integer                       |
| <i>DTIF\$_FMI_NUMEDS_EDITSTR</i> | Handle of ESF\$_EDS aggregate |

## DTIF\$\_FMI

| Item Name                    | Item Encoding                  |
|------------------------------|--------------------------------|
| DTIF\$_FMI_NUMEID_EDITSTR_ID | Integer                        |
| DTIF\$_FMI_NUM_RNDTRUNC      | Enumeration                    |
| <hr/>                        |                                |
| <i>DTIF\$_FMI_TXT_C</i>      | Enumeration                    |
| DTIF\$_FMI_TXTSTD_TYPE       | Enumeration                    |
| DTIF\$_FMI_TXTEDS_EDITSTR    | Handle of ESF\$_EDS aggregate  |
| DTIF\$_FMI_TXTEID_EDITSTR_ID | Integer                        |
| <hr/>                        |                                |
| <i>DTIF\$_FMI_DAT_C</i>      | Enumeration                    |
| DTIF\$_FMI_DATSTD_TYPE       | Enumeration                    |
| DTIF\$_FMI_DATSTD_ORDER      | Enumeration                    |
| DTIF\$_FMI_DATEDS_EDITSTR    | Handle of ESF\$_EDS aggregate  |
| DTIF\$_FMI_DATEID_EDITSTR_ID | Integer                        |
| <hr/>                        |                                |
| DTIF\$_FMI_FLAGS             | Longword                       |
| DTIF\$_FMI_WIDTH             | Integer                        |
| DTIF\$_FMI_LANG_ID           | Integer                        |
| DTIF\$_FMI_DIRECTION         | Enumeration                    |
| DTIF\$_FMI_UNIT_DESC         | Array of type character string |
| DTIF\$_FMI_ALIGNMENT         | Enumeration                    |
| DTIF\$_FMI_BORDER            | Longword                       |

## AGGREGATE ITEMS

### *DTIF\$\_FMI\_WINDOW\_ID*

**Encoding:** *integer*

An optional window identifier item that specifies the window number for a given format. This item allows one format to be defined for a cell when it is displayed in one window, and a different format to be defined for the same cell when it is displayed in a different window. This item references the DTIF\$\_WND\_ID item.

Window 1 is considered the main window. If the DTIF\$\_FMI\_WINDOW\_ID item is not specified, the format can be interpreted as applying to the main window. Applications can choose to apply the main window format as the default for all other windows of a table, unless specific formatting is included for the other windows.

Because not all DTIF applications use windows, Digital recommends that those applications that do recognize windows store the format information according to the guidelines described at the end of this section.

**DTIF\$\_FMI\_C****Encoding: enumeration**

An optional general format type indicator that indicates the type of format to be used.

For each numeric, text, or date data type, the format can be specified as a predefined DTIF format type, such as \$MONEY or \$PERCENT, as an ESF, or as a reference to an edit string defined in a language preference table.

The predefined formats are implicit edit strings that define commonly used formats for a given data type; however, those applications that do not support edit strings can consider a value formatted with an edit string as equivalent to not having an explicit format. Such an application can inform the user that it does not support the DTIF-defined format and can choose an alternate presentation. Valid values are as follows:

|                        |  |
|------------------------|--|
| DTIF\$K_NUMERIC_FORMAT | Indicates the format to be used when displaying numeric values. If you specify this format type, you must supply values for the items DTIF\$_FMI_NUM_DATATYPE through DTIF\$_FMI_NUM_RNDTRUNC. |
| DTIF\$K_TEXT_FORMAT    | Indicates the format to be used when displaying text values. If you specify this format type, you must supply values for the items DTIF\$_FMI_TXT_C through DTIF\$_FMI_TXTEID_EDITSTR_ID.      |
| DTIF\$K_DATE_FORMAT    | Indicates the format to be used when displaying date/time values. If you specify this format type, you must supply values for the items DTIF\$_FMI_DAT_C through DTIF\$_FMI_DATEID_EDITSTR_ID. |

**DTIF\$\_FMI\_NUM\_DATATYPE****Encoding: longword**

A numeric data type item that defines the data type of the numeric value. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$K\_NUMERIC\_FORMAT. Valid flag values are as follows:

|                         |   |
|-------------------------|---|
| dtif\$m_fmi_num_all     | If set, this is the data format used when the application does not distinguish between the formats for integer and floating-point values. |
| dtif\$m_fmi_num_integer | If set, this is the data format used to display integer values.   |
| dtif\$m_fmi_num_float   | If set, this is the data format used to display floating-point values.  |

The default is dtif\$m\_fmi\_num\_all.

**DTIF\$\_FMI\_NUM\_C****Encoding: enumeration**

A numeric format choice indicator that specifies the type of value chosen from those for the numeric format item. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$K\_NUMERIC\_FORMAT. Valid values are as follows:

## DTIF\$\_FMI

|                        |  |
|------------------------|--|
| DTIF\$K_NUM_STANDARD   | Indicates a standard format numeric item. If you specify this numeric format, you must supply values for the items DTIF\$_FMI_NUMSTD_TYPE through DTIF\$_FMI_NUMSTD_FRAC.  |
| DTIF\$K_NUM_EDITSTR    | Indicates a user-defined numeric edit string item. If you specify this numeric format, you must supply a value for the item DTIF\$_FMI_NUMEDS_EDITSTR.   |
| DTIF\$K_NUM_EDITSTR_ID | Indicates an index reference into the list of DTIF\$_NES aggregates defined for the DTIF\$_LPT_EDITSTRS item. If you specify this numeric format, you must supply a value for the item DTIF\$_FMI_NUMEID_EDITSTR_ID. For more information, see the description of the DTIF\$_NES aggregate. For more information on the DTIF\$_LPT_EDITSTRS item, see the description of the DTIF\$_LPT aggregate. |

### **DTIF\$\_FMI\_NUMSTD\_TYPE**

#### **Encoding: enumeration**

An optional numeric standard type item that specifies standard format types for integer and floating-point values. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$K\_NUMERIC\_FORMAT and DTIF\$\_FMI\_NUM\_C as DTIF\$K\_NUM\_STANDARD.

The decoding application must select the appropriate representations for each format type. If this item is not specified, the attribute value is inherited from the next higher level of DTIF\$\_FMI. Valid values are as follows:

|                        |   |
|------------------------|---|
| DTIF\$K_NUMSTD_GENERAL | Indicates a general-purpose format for numeric values, which specifies the most concise display for the value. If a number is very large or very small, it can be expressed in scientific notation.   |
| DTIF\$K_NUMSTD_INTEGER | Indicates a whole number and usually displays the value with no trailing radix symbol. If the DTIF-defined edit string name \$INTEGER is defined by the DTIF\$_LPT_EDITSTRS item, it can be used to display values with this format. If you specify this standard numeric type, you can supply a value for the DTIF\$_FMI_NUM_RNDTRUNC item. For more information on the DTIF\$_LPT_EDITSTRS item, see the description of the DTIF\$_LPT aggregate.   |
| DTIF\$K_NUMSTD_FIXEDPT | Indicates a fixed-point value, which is usually displayed with a trailing radix character. If the DTIF-defined edit string name \$FLOAT is defined by the DTIF\$_LPT_EDITSTRS item, it can be used to display values with this format. If you specify this standard numeric type, you can supply values for the DTIF\$_FMI_NUMSTD_DIGITS, DTIF\$_FMI_NUMSTD_FRAC, and DTIF\$_FMI_NUM_RNDTRUNC items. For more information on the DTIF\$_LPT_EDITSTRS item, see the description of the DTIF\$_LPT aggregate. |



|                           |  |
|---------------------------|--|
| DTIF\$K_NUMSTD_SCIENTIFIC | Indicates scientific notation, which specifies that the value can be displayed using exponential notation to base 10. For example, 123456 can be displayed as $1.23456E + 5$ . If you specify this standard numeric type, you can supply a value for the DTIF\$_FMI_WIDTH item.  |
| DTIF\$K_NUMSTD_MONEY      | Indicates a fixed-point integer value with a currency symbol. Three-digit groupings to the left of the radix point can be separated by the digit separator character. The currency symbol, radix point, and digit separator characters in effect at the current level can be used. For example, 23.45 can be displayed as \$23.45, and 12345.67 can be displayed as \$12,345.67. If the DTIF-defined edit string name \$MONEY is defined by the DTIF\$_LPT_EDITSTRS item, it can be used to display values with this format (it varies among countries). If a decoding application does not understand the edit string, it is responsible for determining the correct format. If you specify this standard numeric type, you can supply values for the DTIF\$_FMI_NUMSTD_DIGITS, DTIF\$_FMI_NUMSTD_FRAC, and DTIF\$_FMI_NUM_RNDTRUNC items. For more information on the DTIF\$_LPT_EDITSTRS item, see the description of the DTIF\$_LPT aggregate. |
| DTIF\$K_NUMSTD_COMMA      | Indicates a fixed-point integer value identical to that specified by DTIF\$K_NUMSTD_MONEY, but without a currency symbol. For example, 12345.67 can be displayed as 12,345.67.   |
| DTIF\$K_NUMSTD_PERCENT    | Indicates a fixed-point integer value with a trailing percent (%) sign. The value is displayed with an implied scale factor of +2. For example, the value .23 can be displayed as 23. If the DTIF-defined edit string name \$PERCENT is defined by the DTIF\$_LPT_EDITSTRS item, it can be used to display values with this format.  |
| DTIF\$K_NUMSTD_PHONE      | Indicates a phone number. For example, 3055551212 can be displayed as (305) 555 - 1212. If the DTIF-defined edit string name \$PHONE is defined by the DTIF\$_LPT_EDITSTRS item, it can be used to specify the format for a phone number (it varies among countries). If the decoding application does not understand the edit string, it is responsible for determining the correct format. For more information on the DTIF\$_LPT_EDITSTRS item, see the description of the DTIF\$_LPT aggregate.  |

## DTIF\$\_FMI

|                      |   |
|----------------------|---|
| DTIF\$_K_NUMSTD_BAR  | Indicates that the value is a pseudo-bar graph. The integer value is used to determine the number of bars to be displayed. The decoding application is responsible for selecting the appropriate display method.<br><br>In some spreadsheets, plus signs (+) are used to display positive values, minus signs (-) are used to display negative values, and a period (.) indicates 0. In other spreadsheets, only positive values are displayed. Some applications may require that this format type be used in conjunction with a special function, such as PLOT(). |
| DTIF\$_K_NUMSTD_TEXT | Indicates that the value is displayed as text. This item can be interpreted by decoding applications to mean that the cell formula rather than the cell value is displayed.   |

### **DTIF\$\_FMI\_NUMSTD\_DIGITS**

#### **Encoding: integer**

An optional standard numeric digits item that indicates the total number of digits of precision to be used when choosing to format the integer and fractional portions of the value. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$\_K\_NUMERIC\_FORMAT, DTIF\$\_FMI\_NUM\_C as DTIF\$\_K\_NUM\_STANDARD, and DTIF\$\_NUMSTD\_TYPE as either DTIF\$\_K\_NUMSTD\_FIXEDPT or DTIF\$\_K\_NUMSTD\_MONEY.

### **DTIF\$\_FMI\_NUMSTD\_FRAC**

#### **Encoding: integer**

An optional numeric standard fraction item that is an integer indicating the number of fractional digits of precision to be used when choosing to format the value. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$\_K\_NUMERIC\_FORMAT, DTIF\$\_FMI\_NUM\_C as DTIF\$\_K\_NUM\_STANDARD, and DTIF\$\_NUMSTD\_TYPE as DTIF\$\_K\_NUMSTD\_INTEGER, DTIF\$\_K\_NUMSTD\_FIXEDPT, or DTIF\$\_K\_NUMSTD\_MONEY.

### **DTIF\$\_FMI\_NUMEDS\_EDITSTR**

#### **Encoding: handle of a ESF\$\_EDS aggregate**

A numeric edit string item that specifies a user-defined ESF format. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$\_K\_NUMERIC\_FORMAT and DTIF\$\_FMI\_NUM\_C as DTIF\$\_K\_NUM\_EDITSTR.

### **DTIF\$\_FMI\_NUMEID\_EDITSTR\_ID**

#### **Encoding: integer**

A numeric edit string index item that is an index reference into the list of DTIF\$\_LPT\_EDITSTRS items in effect for the DTIF\$\_FMI type. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$\_K\_NUMERIC\_FORMAT and DTIF\$\_FMI\_NUM\_C as DTIF\$\_K\_NUM\_EDITSTR\_ID. For more information on the DTIF\$\_LPT\_EDITSTRS item, see the description of the DTIF\$\_LPT aggregate.

The aggregate in effect for a particular DTIF\$\_FMI\_C type is defined by the DTIF\$\_FMI\_LANG\_ID item value in the DTIF\$\_FMI sequence in which the DTIF\$\_FMI\_C type is stored. If no DTIF\$\_FMI\_LANG\_ID value is encoded, the default is 1, which is the first \$DTIF\_LPT aggregate.

**DTIF\$ \_FMI\_NUM\_RNDTRUNC****Encoding: enumeration**

An optional numeric rounding item that specifies whether the displayed numeric value is rounded or truncated. This item is valid if you specified DTIF\$ \_FMI\_C as DTIF\$K\_NUMERIC\_FORMAT, DTIF\$ \_FMI\_NUM\_C as DTIF\$K\_NUM\_STANDARD, and DTIF\$ \_FMI\_NUMSTD\_TYPE as DTIF\$K\_NUMSTD\_INTEGER, DTIF\$K\_NUMSTD\_FIXEDPT, or DTIF\$K\_NUMSTD\_MONEY. Valid values are as follows:

|                      |  |
|----------------------|--|
| DTIF\$K_NUM_ROUND    | Indicates that the displayed numeric value is rounded to the number of digits specified by the DTIF\$ _FMI_NUMSTD_FRAC item. For example, 1.25 rounded to 1 digit is displayed as 1.3.     |
| DTIF\$K_NUM_TRUNCATE | Indicates that the displayed numeric value is truncated to the number of digits specified by the DTIF\$ _FMI_NUMSTD_FRAC item. For example, 1.25 truncated to 1 digit is displayed as 1.2. |

**DTIF\$ \_FMI\_TXT\_C****Encoding: enumeration**

A text data type item that specifies the type of value chosen from those for the text format item. This item is valid if you specified DTIF\$ \_FMI\_C as DTIF\$K\_TEXT\_FORMAT. Valid values are as follows:

|                        |   |
|------------------------|---|
| DTIF\$K_TXT_STANDARD   | Indicates a text standard format item. If you specify this text format, you must supply a value for the item DTIF\$ _FMI_TXTSTD_TYPE.   |
| DTIF\$K_TXT_EDITSTR    | Indicates a user-defined text edit string item. If you specify this text format, you must supply a value for the item DTIF\$ _FMI_TXTEDS_EDITSTR.   |
| DTIF\$K_TXT_EDITSTR_ID | Indicates an index reference into the list of DTIF\$ _NES aggregates defined for the DTIF\$ _LPT_EDITSTRS item. If you specify this text format, you must supply a value for the item DTIF\$ _FMI_TXTEID_EDITSTR_ID. For more information, see the description of the DTIF\$ _NES aggregate. For more information on the DTIF\$ _LPT_EDITSTRS item, see the description of the DTIF\$ _LPT aggregate. |

**DTIF\$ \_FMI\_TXTSTD\_TYPE****Encoding: enumeration**

An optional text standard type item that specifies standard format types for the selected text format. This item is encoded is valid if you specified DTIF\$ \_FMI\_C as DTIF\$K\_TEXT\_FORMAT and DTIF\$ \_FMI\_TXT\_C as DTIF\$K\_TXT\_STANDARD. Valid values are as follows:

## DTIF\$\_FMI

DTIF\$K\_TXTSTD\_PHONE

Indicates that the text value is displayed as a phone number. If the DTIF-defined edit string name \$PHONE is defined by the DTIF\$\_LPT\_EDITSTRS item, it can be used to specify the format for a phone number (it varies among countries). If the decoding application does not understand the edit string, it is responsible for determining the correct format.

DTIF\$K\_TXTSTD\_TEXT

Indicates that the text value is displayed as a text string.

DTIF\$K\_TXTSTD\_REPEAT

Indicates that the text value is displayed as a repeating string. The contents of the text string are repeated across the display width. For example, the string ABC displayed in a column width of 10 characters would appear as ABCABCABCA.

### ***DTIF\$\_FMI\_TXTEDS\_EDITSTR***

#### ***Encoding: handle of a ESF\$\_EDS aggregate***

A text edit string item that specifies a user-defined ESF format. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$K\_TEXT\_FORMAT and DTIF\$\_FMI\_TXT\_C as DTIF\$K\_TXT\_EDITSTR.

### ***DTIF\$\_FMI\_TXTEID\_EDITSTR\_ID***

#### ***Encoding: integer***

A text edit string index item that is an index reference into the list of DTIF\$\_LPT\_EDITSTRS items in effect for the DTIF\$\_FMI type. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$K\_TEXT\_FORMAT and DTIF\$\_FMI\_TXT\_C as DTIF\$K\_TXT\_EDITSTR\_ID. For more information on the DTIF\$\_LPT\_EDITSTRS item, see the description of the DTIF\$\_LPT aggregate.

The aggregate in effect for a particular DTIF\$\_FMI\_C type is defined by the DTIF\$\_FMI\_LANG\_ID item value in the DTIF\$\_FMI sequence in which the DTIF\$\_FMI\_C type is stored. If no DTIF\$\_FMI\_LANG\_ID value is encoded, the default is 1, which is the first \$DTIF\_LPT aggregate.

### ***DTIF\$\_FMI\_DAT\_C***

#### ***Encoding: enumeration***

A date format choice indicator that specifies the type of value chosen from those for the date format item. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$K\_DATE\_FORMAT. Valid values are as follows:

|                        |   |
|------------------------|---|
| DTIF\$K_DAT_STANDARD   | Indicates a date standard format item. If you specify this date format, you must supply values for the items DTIF\$_FMI_DATSTD_TYPE through DTIF\$_FMI_DATSTD_ORDER.  |
| DTIF\$K_DAT_EDITSTR    | Indicates a user-defined date edit string item. If you specify this date format, you must supply a value for the item DTIF\$_FMI_DATEDS_EDITSTR.  |
| DTIF\$K_DAT_EDITSTR_ID | Indicates an index reference into the list of DTIF\$_NES aggregates defined for the DTIF\$_LPT_EDITSTRS item. If you specify this date format, you must supply a value for the item DTIF\$_FMI_DATEID_EDITSTR_ID. For more information on the DTIF\$_NES aggregate, see the description of the DTIF\$_NES aggregate. For more information on the DTIF\$_LPT_EDITSTRS item, see the description of the DTIF\$_LPT aggregate. |

**DTIF\$\_FMI\_DATSTD\_TYPE****Encoding: enumeration**

An optional date standard type item that specifies standard format types for date values. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$K\_DATE\_FORMAT and DTIF\$\_FMI\_DAT\_C as DTIF\$K\_DAT\_STANDARD. Valid values are as follows:

|                            |   |
|----------------------------|---|
| DTIF\$K_DATSTD_DATEONLY    | Indicates that only the date portion (day, month, year) of the value is displayed. If the DTIF-defined edit string name \$DATE is defined by the DTIF\$_LPT_EDITSTRS item, it can be used to display values with this format. For more information on the DTIF\$_LPT_EDITSTRS item, see the description of the DTIF\$_LPT aggregate.                |
| DTIF\$K_DATSTD_TIMEONLY    | Indicates that only the time portion (hours, minutes, seconds, . . . ) of the value is displayed. If the DTIF-defined edit string name \$TIME is defined by the DTIF\$_LPT_EDITSTRS item, it can be used to display values with this format. For more information on the DTIF\$_LPT_EDITSTRS item, see the description of the DTIF\$_LPT aggregate. |
| DTIF\$K_DATSTD_DATEANDTIME | Indicates that both the date and time portions of the value are displayed. If the DTIF-defined edit string name \$DATETIME is defined by the DTIF\$_LPT_EDITSTRS item, it can be used to display values with this format. For more information on the DTIF\$_LPT_EDITSTRS item, see the description of the DTIF\$_LPT aggregate.                    |

**DTIF\$\_FMI\_DATSTD\_ORDER****Encoding: enumeration**

An optional date standard order item that specifies the order to be used when displaying a date portion of the value. This item is valid if you specified DTIF\$\_

## DTIF\$\_FMI

FMI\_C as DTIF\$K\_DATE\_FORMAT and DTIF\$\_FMI\_DAT\_C as DTIF\$K\_DAT\_STANDARD. Valid values are as follows:

|                    |   |
|--------------------|---|
| DTIF\$K_DATSTD_MDY | Indicates that the month-day-year order is used when displaying the date. |
| DTIF\$K_DATSTD_DMY | Indicates that the day-month-year order is used when displaying the date. |

### **DTIF\$\_FMI\_DATEDS\_EDITSTR**

#### **Encoding: handle of a ESF\$\_EDS aggregate**

A date edit string item that specifies a user-defined ESF format. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$K\_DATE\_FORMAT and DTIF\$\_FMI\_DAT\_C as DTIF\$K\_DAT\_EDITSTR.

### **DTIF\$\_FMI\_DATEID\_EDITSTR\_ID**

#### **Encoding: integer**

A date edit string index item that is an index reference into the list of DTIF\$\_LPT\_EDITSTRS items in effect for the DTIF\$\_FMI type. This item is valid if you specified DTIF\$\_FMI\_C as DTIF\$K\_DATE\_FORMAT and DTIF\$\_FMI\_DAT\_C as DTIF\$K\_DATSTD\_EDITSTR\_ID. For more information on the DTIF\$\_LPT\_EDITSTRS item, see the description of the DTIF\$\_LPT aggregate.

The aggregate in effect for a particular DTIF\$\_FMI\_C type is defined by the DTIF\$\_FMI\_LANG\_ID item value in the DTIF\$\_FMI sequence in which the DTIF\$\_FMI\_C type is stored. If no DTIF\$\_FMI\_LANG\_ID value is encoded, the default is 1, which is the first \$DTIF\_LPT aggregate.

### **DTIF\$\_FMI\_FLAGS**

#### **Encoding: longword**

An optional flags item that contains additional information pertinent to the display of a value. If neither the positive nor negative flags bit is set, the attribute is unspecified and is inherited from the next higher level. If both positive and negative flag bits are set, the positive bit has precedence and the attribute is enabled. This item can accept any one of the following values:

|  |   |
|--|---|
| <code>dtif\$m_fmi_readonly</code>        | If set, the cell contents are read-only and are not modified.   |
| <code>dtif\$m_fmi_noreadonly</code>      | If set, the cell contents are not displayed.  |
| <code>dtif\$m_fmi_bold</code>            | If set, the cell contents are displayed as bold.  |
| <code>dtif\$m_fmi_nobold</code>          | If set, the cell contents are not displayed as bold.  |
| <code>dtif\$m_fmi_italic</code>          | If set, the cell contents are displayed as italic.  |
| <code>dtif\$m_fmi_noitalic</code>        | If set, the cell contents are not displayed as italic.  |
| <code>dtif\$m_fmi_underline</code>       | If set, the cell contents are displayed as underlined.  |
| <code>dtif\$m_fmi_nounderline</code>     | If set, the cell contents are not displayed as underlined.  |
| <code>dtif\$m_fmi_valuehidden</code>     | If set, the cell value is hidden. This can be used as a safety feature. Applications can interpret the hidden flag in different ways; they may choose not to display the value at all, or to display a string indicating that the value is not to be displayed.       |
| <code>dtif\$m_fmi_novaluehidden</code>   | If set, the cell value is not hidden.   |
| <code>dtif\$m_fmi_formulahidden</code>   | If set, the cell formula is hidden. This can be used as a safety feature. Applications can interpret the hidden flag in different ways; they may choose not to display the formula at all, or to display a string indicating that the formula is not to be displayed. |
| <code>dtif\$m_fmi_noformulahidden</code> | If set, the cell formula is not hidden.   |
| <code>dtif\$m_fmi_running</code>         | If set, the cell contents are displayed as a running label (across cell boundaries). It is the responsibility of the decoding application to determine the appropriate method for displaying this format type.  |
| <code>dtif\$m_fmi_norunning</code>       | If set, the cell contents are not displayed as a running label.   |

**DTIF\$\_FMI\_WIDTH****Encoding: integer**

An optional display width item that contains the number of characters to be used as the display column width. The DTIF\$\_FMI\_WIDTH item should not be specified at cell level by the DTIF\$\_CLD\_FORMATS item. If this item is present at cell level, it should be ignored. Column width is applied after formatting. For more information on the DTIF\$\_CLD\_FORMATS item, see the description of the DTIF\$\_CLD aggregate.

**DTIF\$\_FMI\_LANG\_ID****Encoding: integer**

An optional language preference index item that contains an integer index into the list of language preference tables stored in the table document header. Applications that perform different formatting based on language can use this item to determine the language for a particular value.

## DTIF\$\_FMI

The DTIF\$\_FMI\_LANG\_ID integer value corresponds to the value of the DTIF\$\_LPT\_LANGUAGE\_INDEX item for a particular preference table. The language preference tables (DDIF\$\_LPT) are located in the DTIF\$\_HDR\_LANGUAGE\_PREF\_TABLES item in the DTIF\$\_HDR header aggregate. For more information on the DTIF\$\_LPT\_LANGUAGE\_INDEX item, see the description of the DTIF\$\_LPT aggregate.

### **DTIF\$\_FMI\_DIRECTION**

#### **Encoding: enumeration**

An optional format direction item that specifies the direction in which the data is formatted. This item has the following value:

|                       |  |
|-----------------------|--|
| DTIF\$_K_DIR_OPPOSITE | Displays the formatted value in reverse order with respect to the text path. For example, \$1,123.45 is displayed as 54.321,1\$. |
|-----------------------|--|

### **DTIF\$\_FMI\_UNIT\_DESC**

#### **Encoding: array of type character string**

An optional format descriptor item that contains a sequence of text strings that can be used to indicate the units of the displayed value. The decoding application determines the appropriate display form; one possibility is to include the units as a string following the displayed value. An application can also define an edit string for the same purpose.

### **DTIF\$\_FMI\_ALIGNMENT**

#### **Encoding: enumeration**

An optional alignment item that specifies the format alignment to be used when choosing to format the cell. Valid values are as follows:

|                       |  |
|-----------------------|--|
| DTIF\$_K_ALIGN_LEFT   | If set, the cell contents are left-justified within the display column.  |
| DTIF\$_K_ALIGN_CENTER | If set, the cell contents are centered within the display column.        |
| DTIF\$_K_ALIGN_RIGHT  | If set, the cell contents are right-justified within the display column. |

### **DTIF\$\_FMI\_BORDER**

#### **Encoding: longword**

An optional border item that specifies whether the cell is displayed with an explicit border in addition to any grid lines that are displayed for the table. A border edge can be explicitly disabled (removed) by selecting the negated border bit. If neither the positive nor negative border bit is set, the attribute is unspecified and is inherited from the next higher level. If both positive and negative border bits are set, the positive bit has precedence and the attribute is enabled. Valid values are as follows:



|  |   |
|--|---|
| <code>dtif\$m_fmi_border_left</code>     | Indicates that the cell is displayed with a left border. This value has precedence over the <code>dtif\$m_fmi_border_noleft</code> value.     |
| <code>dtif\$m_fmi_border_noleft</code>   | Indicates that the cell is displayed with no left border.   |
| <code>dtif\$m_fmi_border_top</code>      | Indicates that the cell is displayed with a top border. This value has precedence over the <code>dtif\$m_fmi_border_notop</code> value.       |
| <code>dtif\$m_fmi_border_notop</code>    | Indicates that the cell is displayed with no top border.  |
| <code>dtif\$m_fmi_border_right</code>    | Indicates that the cell is displayed with a right border. This value has precedence over the <code>dtif\$m_fmi_border_noright</code> value.   |
| <code>dtif\$m_fmi_border_noright</code>  | Indicates that the cell is displayed with no right border.  |
| <code>dtif\$m_fmi_border_bottom</code>   | Indicates that the cell is displayed with a bottom border. This value has precedence over the <code>dtif\$m_fmi_border_nobottom</code> value. |
| <code>dtif\$m_fmi_border_nobottom</code> | Indicates that the cell is displayed with no bottom border.   |

The guidelines for storing format information for windows are as follows:

1. Store the format attributes for each data type within the main window at the table level, using the `DTIF$_TMD_DEFAULT_FMTS` item. For more information on the `DTIF$_TMD_DEFAULT_FMTS` item, see the description of the `DTIF$_TMD` aggregate. Either omit the `DTIF$_FMI_WINDOW_ID` item or specify the value as 1. The intent is to store the main window default format information as the default formatting in effect for the entire table. This information can also be stored in the `DTIF$_WND_FORMATS` item, with the `DTIF$_FMI_WINDOW_ID` item equal to 1. For more information on the `DTIF$_WND_FORMATS` item, see the description of the `DTIF$_WND` aggregate.
2. Store the default format attributes for each data type within the remaining windows (excluding the main) at the window level, using the `DTIF$_WND_FORMATS` type. Store the appropriate `DTIF$_FMI_WINDOW_ID` for each window within the `DTIF$_FMI` aggregate. For more information on the `DTIF$_WND_FORMATS` item, see the description of the `DTIF$_WND` aggregate.
3. Store the format attributes for each column (within a window) at the table columns level, using the `DTIF$_CAT_FORMATS` item within the `DTIF$_TMD_COLUMNS` item. If the column has different format attributes in different windows, the `DTIF$_FMI_WINDOW_ID` item must be specified. If the column has no window-specific format attributes (for example, they are identical for all windows), the `DTIF$_FMI_WINDOW_ID` item can be omitted or specified as 1. For more information on the `DTIF$_CAT_FORMATS` item, see the description of the `DTIF$_CAT` aggregate. For more information on the `DTIF$_TMD_COLUMNS` item, see the description of the `DTIF$_TMD` aggregate.

## DTIF\$\_FMI

4. Store the format attributes for each row (within a window) at the row level, using the DTIF\$\_ROW\_FORMATS item within the DTIF\$\_ROW aggregate. If the row has window-specific format attributes, the DTIF\$\_FMI\_WINDOW\_ID item must be specified. If the row has no window-specific format attributes (for example, all rows appear identical in all windows), the DTIF\$\_FMI\_WINDOW\_ID item can be omitted or specified as 1. For more information, see the description of the DTIF\$\_ROW aggregate.
5. Store the format attributes for each cell (within a window) at the cell level, using the DTIF\$\_CLD\_FORMATS item within the DTIF\$\_CLD aggregate. If the cell has window-specific format attributes, the DTIF\$\_FMI\_WINDOW\_ID item must be specified. If the cell has no window-specific format attributes (for example, the cell appears identical in all windows), the DTIF\$\_FMI\_WINDOW\_ID item can be omitted or specified as 1. For more information, see the description of the DTIF\$\_CLD aggregate.

---

## DTIF\$ \_HDR—Document Header Aggregate

The DTIF document header aggregate contains the title and creation date of the document and any resources used or needed to process the tables within the document. Resources include references to data external to the document, languages and preference definitions used within the table or tables, and generic column definitions referred to by table columns. The DTIF\$ \_HDR aggregate is referenced by the parent aggregate item DTIF\$ \_DTF\_HEADER.

Refer to these corresponding syntax diagrams:

| Syntax         | Location   |
|----------------|------------|
| DocumentHeader | Figure C-3 |

---

### AGGREGATE FORMAT

| Item Name                        | Item Encoding                      |
|----------------------------------|------------------------------------|
| DTIF\$ _HDR_PRIVATE_DATA         | Sequence of DTIF\$ _NVL aggregates |
| DTIF\$ _HDR_TITLE                | Array of type character string     |
| DTIF\$ _HDR_DATE                 | Handle of DTIF\$ _DAT aggregate    |
| DTIF\$ _HDR_EXTERNAL_REFERENCES  | Sequence of DTIF\$ _ERF aggregates |
| DTIF\$ _HDR_LANGUAGES_C          | Array of type enumeration          |
| DTIF\$ _HDR_LANGUAGES            | Array of type variable             |
| DTIF\$ _HDR_LANGUAGE_PREF_TABLES | Sequence of DTIF\$ _LPT aggregates |
| DTIF\$ _HDR_GENERIC_COLUMNS      | Sequence of DTIF\$ _CAT aggregates |

---

### AGGREGATE ITEMS

#### ***DTIF\$ \_HDR\_PRIVATE\_DATA***

**Encoding:** *sequence of DTIF\$ \_NVL aggregates*

An optional private header data item that contains global information about the DTIF document not currently standardized by DTIF. All interpretations of the private data are subject only to private agreements between the parties concerned. For more information, see the description of the DTIF\$ \_NVL aggregate.

#### ***DTIF\$ \_HDR\_TITLE***

**Encoding:** *array of type character string*

An optional title item that contains the user-visible name of the document.

## DTIF\$\_HDR

### **DTIF\$\_HDR\_DATE**

**Encoding:** *handle of a DTIF\$\_DAT aggregate*

An optional date item that contains the date associated with this version of the document. For more information, see the description of the DTIF\$\_DAT aggregate.

### **DTIF\$\_HDR\_EXTERNAL\_REFERENCES**

**Encoding:** *sequence of DTIF\$\_ERF aggregates*

An optional external references item that specifies a list of file names (or other system-specific file specifiers) that are referenced from within the tables. In the body of the DTIF document, external references are specified as indexes into this list. For more information, see the description of the DTIF\$\_ERF aggregate.

### **DTIF\$\_HDR\_LANGUAGES\_C**

**Encoding:** *array of type enumeration*

An optional languages indicator that specifies the type of natural language or programming language chosen from those for processing by language tools. Valid values for this item are as follows:

DTIF\$\_K\_ISO\_639\_LANGUAGE

A string that selects a language and dialect that is specified using the ISO 639 Standard. In this case, the DTIF\$\_HDR\_LANGUAGES item is encoded as a string.

DTIF\$\_K\_OTHER\_LANGUAGE

A character string that indicates the language and dialect using a “user-readable” name; this is used for those languages and dialects not covered by the ISO 639 Standard. In this case, the DTIF\$\_HDR\_LANGUAGES item is encoded as a character string.

### **DTIF\$\_HDR\_LANGUAGES**

**Encoding:** *array of type variable*

A language item that contains the actual language for the language type selected in the previous item.

### **DTIF\$\_HDR\_LANGUAGE\_PREF\_TABLES**

**Encoding:** *sequence of DTIF\$\_LPT aggregates*

An optional language preference table item that specifies to the decoding application additional formatting, display, and semantic information that is specific to a particular language or application. For more information, see the description of the DTIF\$\_LPT aggregate.

The first preference table has an index of 1 and supplies the default values for the document. Subsequent language preference tables are numbered increasing by 1 (2, 3, . . .).

The DTIF\$\_LPT\_LANGUAGE\_INDEX item for each preference table is referenced at any point within a table by the DTIF\$\_FMI\_LANG\_ID item in the DTIF\$\_FMI aggregate. The first preference table has an index of 1, and subsequent language preference tables are numbered increasing by 1 (2, 3, . . .). For more information, see the description of the DTIF\$\_FMI aggregate.

### **DTIF\$\_HDR\_GENERIC\_COLUMNS**

**Encoding:** *sequence of DTIF\$\_CAT aggregates*

An optional generic attributes item that supplies column attributes for any table column in the DTIF document that references this generic attributes item. For

## DTIF\$\_HDR

more information, see the description of the DTIF\$\_CAT aggregate. Note that the DTIF\$\_CAT aggregate item DTIF\$\_CAT\_GENERIC\_REF is not used when encoding DTIF\$\_HDR\_GENERIC\_COLUMNS.

## DTIF\$\_LPT

---

### DTIF\$\_LPT—Language Preference Tables Aggregate

The language preference table aggregate contains display, format, and semantic information that is specific to a particular language or country, application, or user's preferences. More than one preference table can be defined for a document; each is uniquely identified by its position in the list. The first preference table has an index of 1 and supplies the default values for the document. Subsequent language preference tables are numbered increasing by 1 (2, 3, . . . ).

The DTIF\$\_LPT\_LANGUAGE\_INDEX item for each preference table is referenced at any point within a table by the DTIF\$\_FMI\_LANG\_ID item in the DTIF\$\_FMI aggregate. For more information, see the description of the DTIF\$\_FMI aggregate. If language preference tables are not defined in the document, the decoding application must select the appropriate representations. A decoding application can decide to ignore or to override the originating format, if applicable. The DTIF\$\_LPT aggregate is referenced by the parent aggregate item DTIF\$\_HDR\_LANGUAGE\_PREF\_TABLES.

Refer to these corresponding syntax diagrams:

| Syntax        | Location   |
|---------------|------------|
| LangPrefTable | Figure C-7 |

---

### AGGREGATE FORMAT

| Item Name                 | Item Encoding                     |
|---------------------------|-----------------------------------|
| DTIF\$_LPT_LANGUAGE_INDEX | Integer                           |
| DTIF\$_LPT_APPL_PRIVATE   | Sequence of DTIF\$_NVL aggregates |
| DTIF\$_LPT_ITEMS          | Sequence of DTIF\$_NVL aggregates |
| DTIF\$_LPT_EDITSTRS       | Sequence of DTIF\$_NES aggregates |
| DTIF\$_LPT_COLLATE_SEQ    | String with <b>add-info</b>       |
| DTIF\$_LPT_COLLATE_TABLE  | String                            |

---

### AGGREGATE ITEMS

#### **DTIF\$\_LPT\_LANGUAGE\_INDEX**

**Encoding:** *integer*

An optional language index item that defines an index into the list of languages in the DTIF\$\_HDR aggregate. This item references the DTIF\$\_HDR\_LANGUAGE\_PREF\_TABLES item. For more information, see the description of the DTIF\$\_HDR aggregate.

**DTIF\$\_LPT\_APPL\_PRIVATE****Encoding: sequence of DTIF\$\_NVL aggregates**

An optional private preference data item that contains application-private information about the preference table that is not currently standardized by DTIF. All interpretations of the private data are subject only to private agreements between the parties concerned. For more information, see the description of the DTIF\$\_NVL aggregate.

**DTIF\$\_LPT\_ITEMS****Encoding: sequence of DTIF\$\_NVL aggregates**

An optional item name item that specifies presentation attributes consisting of an item name and one or more values for the preference table. For more information, see the description of the DTIF\$\_NVL aggregate.

An application or set of cooperating applications can use this item to designate either a DTIF-defined item name defined in the following table or other item names. In this case, the application is responsible for the interpretation of the item names. The item names listed here are not case sensitive.

|          |  |
|----------|--|
| \$CUR    | Defines a text string for the currency symbol. The item value is a text string. For example: \$, Fr, and DM.   |
| \$CURPOS | Defines a text string for the position of the currency symbol. The item value is an integer, defined as follows: 0=leading, 1=trailing, and 2=radix point.   |
| \$CURRAD | Defines a text string for the currency radix symbol. The item value is a text string. The radix string is used to separate the whole number from the fraction in a monetary item. The currency radix can differ from the numeric radix. Examples of item values for \$CURRAD are the decimal point (.), as in \$1.23, or the comma (,), as in 1,23 DM. Another example is F, as in the value 12 F 20. In this example, the currency radix point is used to display the currency symbol (and the \$CUR item may not be defined). Note that \$CURRAD and \$CURSEP should not be the same string. |
| \$CURSEP | Defines a text string for the currency digit separator symbol. The item value is a text string. The digit separator is the character or characters that separate groupings of three digits (in the whole number part) of a monetary item. This string can differ from the numeric digit separator. Examples of item values for \$CURSEP are the comma (,), as in 1,234.00 DM, or a space (( )), as in 1 234,00. Note that \$CURRAD and \$CURSEP should not be the same string.   |
| \$NUMRAD | Defines a text string for the numeric radix symbol. The item value is a text string. The radix string is used to separate the whole number part from the fraction part in a numeric (floating-point, integer) item. The numeric radix can differ from the currency radix. Examples of item values for \$NUMRAD are the period (.) or the comma (,). Note that \$NUMRAD and \$NUMSEP should not be defined as the same string.  |

## DTIF\$\_LPT

|          |  |
|----------|--|
| \$NUMSEP | Defines a text string for the numeric digit separator symbol. The item value is a text string. The digit separator is the character or characters that separate groupings of three digits (in the whole number part) of a numeric item. This string can differ from the numeric digit separator. Examples of item values for \$NUMSEP are the comma (,) or the space (( )). Note that \$NUMRAD and \$NUMSEP should not be defined as the same string.  |
| \$DAYS   | The days of the week. The strings in the list are ordered sequentially; the first day of the week corresponds to Sunday and the last day of the week corresponds to Saturday. The text strings listed here can be used by the decoding application when displaying the names of weekdays. For example, the English language list starts with Sunday and ends with Saturday; the French list starts with dimanche and ends with samedi. The encoding application can choose to use abbreviated names for the weekdays, such as MON. |
| \$MONTHS | A sequence of text strings corresponding to the months of the year. The strings in the list are ordered sequentially; the first month of the year is January (in English), and the last month is December (in English). The text strings listed here may be used by the decoding application when displaying the names of months. For example, the French language list starts with Janvier and ends with Décembre. The encoding application can choose to use abbreviated names for the month names, such as JAN.                 |
| \$AM     | A text string to be displayed for time values between 0:00:00 A.M. and 11:59:00 A.M.   |
| \$PM     | A text string to be displayed for time values between 12:00:00 P.M. and 11:59:00 P.M.  |
| \$TRUE   | The string to be used when displaying the logical TRUE value.  |
| \$FALSE  | The string to be used when displaying the logical FALSE value.   |

### **DTIF\$\_LPT\_EDITSTRS**

#### **Encoding: sequence of DTIF\$\_NES aggregates**

An optional edit string name item that specifies how to format types of data (such as money and phone). The first DTIF\$\_NES aggregate has an index of 1.

The DTIF\$\_FMI\_TXTEID\_EDITSTR\_ID item, the DTIF\$\_FMI\_DATEID\_EDITSTR\_ID item, and the DTIF\$\_FMI\_NUMEID\_EDITSTR\_ID item are index references into the sequence of DTIF\$\_NES aggregates. For more information on the DTIF\$\_NES aggregate, see the named edit string aggregate.

An application or set of cooperating applications can use this item to designate either a DTIF-defined edit string name defined in the following table or other edit string names. In this case, the application is responsible for the interpretation of the edit string names. The edit string names listed here are not case sensitive.



|            |   |
|------------|---|
| \$DATE     | Specifies the format for date values.           |
| \$DATETIME | Specifies the format for date-and-time values.  |
| \$TEXT     | Specifies the format for text values.           |
| \$FLOAT    | Specifies the format for floating-point values. |
| \$INTEGER  | Specifies the format for integer values.        |
| \$MONEY    | Specifies the format for money values.          |
| \$PERCENT  | Specifies the format for percent values.        |
| \$PHONE    | Specifies the format for phone values.          |
| \$TIME     | Specifies the format for time values.           |

**DTIF\$\_LPT\_COLLATE\_SEQ****Encoding: string with add-info**

An optional collating sequence item that specifies a predefined named collating sequence for the current language. The following table lists the values for **add-info** and the corresponding string values.

| <b>Add-Info</b>             | <b>String</b>   |
|-----------------------------|---|
| DTIF\$K_PRIVATE_COLLATE_SEQ | The collating sequence is private.  |
| DTIF\$K_ASCII_COLLATE_SEQ   | The collating sequence is ASCII ("ASCII"), and the order = 00 through FF. |
| DTIF\$K_MCS_COLLATE_SEQ     | The collating sequence is DEC Multinational Character Set ("MCS").        |
| DTIF\$K_DANISH_COLLATE_SEQ  | The collating sequence is Danish ("DANISH").                              |
| DTIF\$K_FINSWD_COLLATE_SEQ  | The collating sequence is Finnish/Swedish ("FINSWD").                     |
| DTIF\$K_NORWEG_COLLATE_SEQ  | The collating sequence is Norwegian ("NORWEG").                           |
| DTIF\$K_SPANSH_COLLATE_SEQ  | The collating sequence is Spanish ("SPANSH").                             |

An application or set of cooperating applications can use this item to designate either one of the Digital standard collating sequences defined in the previous table or another named collating sequence. If another named collating sequence is defined, the application is responsible for the interpretation of the collating sequences. If the application wants to encode the collating sequence itself using DTIF, it can use the DTIF\$\_LPT\_COLLATE\_TABLE item.

**NOTE**

A leading dollar sign (\$) indicates an identifier defined by DTIF and is reserved for Digital products. The underscore character (\_) is reserved for Digital customers. Applications must prefix their private identifiers with a string that is representative of their company or product name and an underscore (for example, XYZCORP\_XXX). Digital assumes no responsibility for potential conflict in customer-derived names.

## **DTIF\$\_LPT**

### ***DTIF\$\_LPT\_COLLATE\_TABLE***

***Encoding: string***

An optional customized collating sequence item that permits private collating schemes to be self-contained within the document.

---

## DTIF\$\_NES—Named Edit String Aggregate

The named edit string aggregate describes how certain types of data are to be formatted within a language preference table. The DTIF\$\_NES aggregate is referenced by the parent aggregate item DTIF\$\_LPT\_EDITSTRS.

Refer to these corresponding syntax diagrams:

| Syntax          | Location   |
|-----------------|------------|
| NamedEditString | Figure C-8 |

---

### AGGREGATE FORMAT

| Item Name       | Item Encoding                 |
|-----------------|-------------------------------|
| DTIF\$_NES_NAME | String                        |
| DTIF\$_NES_DEFN | Handle of ESF\$_EDS aggregate |

---

### AGGREGATE ITEMS

**DTIF\$\_NES\_NAME**

**Encoding:** *string*

An edit string name item that can be used when a cell is formatted with a corresponding format type.

**DTIF\$\_NES\_DEFN**

**Encoding:** *handle of ESF\$\_EDS aggregate*

An edit string definition item. For more information, see the description of the ESF\$\_EDS aggregate.

## DTIF\$\_NMR

---

### DTIF\$\_NMR—Named Range Aggregate

The named range aggregate identifies a range by name. The name corresponds to a DTIF\$\_RNG\_NAME item defined in the table. For more information on the DTIF\$\_RNG\_NAME item, see the description of the DTIF\$\_RNG aggregate. The DTIF\$\_NMR aggregate is referenced by the parent aggregate item DTIF\$\_RNG\_REGION.

Refer to these corresponding syntax diagrams:

---

| Syntax     | Location    |
|------------|-------------|
| NamedRange | Figure C-42 |

---

---

### AGGREGATE FORMAT

---

| Item Name             | Item Encoding                  |
|-----------------------|--------------------------------|
| DTIF\$_NMR_NAMEDRANGE | Array of type character string |

---

---

### AGGREGATE ITEMS

***DTIF\$\_NMR\_NAMEDRANGE***

***Encoding: array of type character string***

A named range data item that specifies the name of the named range. This item references the DTIF\$\_RNG\_NAME item.

---

## DTIF\$\_NVL—DTIF Named Value Aggregate

The named value aggregate specifies a name, type, and value. The value may be a list of values. The named value aggregate defines document data that is restricted either to a particular data processing implementation, or to a set of related implementations that support identical private encodings. The DTIF\$\_NVL aggregate is referenced by the parent aggregate items DTIF\$\_CAT\_APPL\_PRIVATE, DTIF\$\_CLD\_APPL\_PRIVATE, DTIF\$\_HDR\_PRIVATE\_DATA, DTIF\$\_LPT\_APPL\_PRIVATE, DTIF\$\_LPT\_ITEMS, DTIF\$\_NVL\_VALUE, DTIF\$\_ROW\_APPL\_PRIVATE, DTIF\$\_TBL\_APPL\_PRIVATE, DTIF\$\_TMD\_APPL\_PRIVATE, and DTIF\$\_WND\_APPL\_PRIVATE. Each of these parent aggregate items references a sequence of DTIF\$\_NVL aggregates.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| NamedValueList | Figure C-31 |
| ValueData      | Figure C-32 |

---

### AGGREGATE FORMAT

| Item Name          | Item Encoding |
|--------------------|---------------|
| DTIF\$_NVL_NAME    | String        |
| DTIF\$_NVL_VALUE_C | Enumeration   |
| DTIF\$_NVL_VALUE   | Variable      |

---

### AGGREGATE ITEMS

**DTIF\$\_NVL\_NAME**

**Encoding: string**

A value name item that is limited to the characters of the ASCII character set and that identifies the named value.

**DTIF\$\_NVL\_VALUE\_C**

**Encoding: enumeration**

A value data indicator that specifies the type of value chosen from those that are delineated for the named data value. Valid values for this item are as follows:

|                       |   |
|-----------------------|---|
| DTIF\$K_VALUE_BOOLEAN | Indicates a Boolean value. In this case, the DTIF\$_NVL_VALUE item is encoded as a Boolean value. |
|-----------------------|---|

## DTIF\$\_NVL

|                        |   |
|------------------------|---|
| DTIF\$K_VALUE_INTEGER  | Indicates an integer value. In this case, the DTIF\$_NVL_VALUE item is encoded as an integer.   |
| DTIF\$K_VALUE_TEXT     | Indicates a text string value. In this case, the DTIF\$_NVL_VALUE item is encoded as an array of type character string.   |
| DTIF\$K_VALUE_GENERAL  | Indicates a stream of bytes in any format. In this case, the DTIF\$_NVL_VALUE item is encoded as a string.  |
| DTIF\$K_VALUE_LIST     | Indicates a list of data values such as the above. In this case, the DTIF\$_NVL_VALUE item is encoded as a sequence of DTIF\$_NVL aggregates. In the nested DTIF\$_NVL aggregates, the DTIF\$_NVL_NAME item is ignored.             |
| DTIF\$K_VALUE_EXTERNAL | Indicates a data value that is represented in a syntax. In this case, the DTIF\$_NVL_VALUE item is encoded as the handle of an aggregate of type DTIF\$_EXT. For more information, see the description of the DTIF\$_EXT aggregate. |
| DTIF\$K_VALUE_FLOAT    | Indicates a floating-point value. In this case, the DTIF\$_NVL_VALUE item is encoded as a general floating-point value.   |
| DTIF\$K_VALUE_DATE     | Indicates a date/time value. In this case, the DTIF\$_NVL_VALUE item is encoded as the handle of a DTIF\$_DAT aggregate. For more information, see the description of the DTIF\$_DAT aggregate.                                     |
| DTIF\$K_VALUE_EXPR     | Indicates an expression whose result is the value. In this case, the DTIF\$_NVL_VALUE item is encoded as the handle of a CFE\$_EXP aggregate. For more information, see the description of the CFE\$_EXP aggregate.                 |

### **DTIF\$\_NVL\_VALUE**

#### **Encoding: variable**

A value data item that contains the actual data value for the value type selected by the previous item.

---

## DTIF\$\_RNG—Range Definition Aggregate

The range definition aggregate defines a logically grouped set of cells specified at table or window level using a list of one or more DTIF\$\_RNG aggregates. The cells in the range can be contiguous or disjoint. A contiguous range will be interpreted as including all cells from the starting cell to the ending cell. A disjoint range is a sequence of one or more contiguous ranges; the ranges within the sequence need not be adjacent to each other. For example, a range named Q1FY86 that is defined as the list of ranges named Jan, Feb, and Mar consists of cells defined by these three ranges.

Some applications do not support operations on disjoint ranges. In this case, these applications can either accept the first range of the sequence as the entire range, or discard the range altogether. The application informs the user if it cannot fully process the range.

Some applications also allow range names to be used in place of cell coordinates within formulas.

The DTIF\$\_RNG aggregate is referenced by the parent aggregate items DTIF\$\_TMD\_RANGES and DTIF\$\_WND\_RANGES.

Refer to these corresponding syntax diagrams:

| Syntax        | Location    |
|---------------|-------------|
| RangeDefnList | Figure C-37 |
| Range         | Figure C-38 |

---

## AGGREGATE FORMAT

| Item Name              | Item Encoding   |
|------------------------|---|
| DTIF\$_RNG_NAME        | Array of type character string  |
| DTIF\$_RNG_TYPE        | Enumeration   |
| DTIF\$_RNG_REGION      | Sequence of DTIF\$_CLR, DTIF\$_RWR, DTIF\$_COR, DTIF\$_NMR aggregates |
| DTIF\$_RNG_SORT_KEYNUM | Integer   |

---

## AGGREGATE ITEMS

### ***DTIF\$\_RNG\_NAME***

***Encoding: array of type character string***

An optional range name item that identifies this range. This item must be present if the DTIF\$\_RNG\_TYPE item is set to DTIF\$K\_RT\_NAMED\_RANGE. This item is referenced by the DTIF\$\_NMR\_NAMEDRANGE item.

## DTIF\$\_RNG

### **DTIF\$\_RNG\_TYPE**

#### **Encoding: enumeration**

A range type item that identifies the type of range. Valid values are as follows:

|                         |   |
|-------------------------|---|
| DTIF\$K_RT_NAMED_RANGE  | A named range that can be referenced in formulas.   |
| DTIF\$K_RT_VIEW_RANGE   | A view range used in some spreadsheets to define a viewing area. The first cell defines the upper left cell displayed in a window, and the second cell defines the cursor location, which is the active cell.   |
| DTIF\$K_RT_COL_TITLE    | A range of cells in the column title area. This range type is typically used in a window to identify the parts of the window.   |
| DTIF\$K_RT_ROW_TITLE    | A range of cells in the row title area. This range type is typically used in a window to identify the parts of the window.  |
| DTIF\$K_RT_DISPLAY_DATA | A range of cells currently visible in the scroll region of the window. This range must be a subset of DTIF\$K_RT_DATA_RANGE. This range type is typically used in a window to identify the parts of the window. |
| DTIF\$K_RT_DATA_RANGE   | The entire set of cells that compose the data portion of the window. This range type is typically used in a window to identify the parts of the window.   |
| DTIF\$K_RT_SORT_RANGE   | A range of cells used for sorting within the table.   |

The default is DTIF\$K\_RT\_NAMED\_RANGE.

### **DTIF\$\_RNG\_REGION**

#### **Encoding: sequence of DTIF\$\_CLR, DTIF\$\_RWR, DTIF\$\_COR, or DTIF\$\_NMR aggregates**

An optional range region item that is a list of one or more ranges defined by any of the aggregates DTIF\$\_CLR, DTIF\$\_RWR, DTIF\$\_COR, DTIF\$\_NMR. If more than one range is present, the range region is the union of all cells in the ranges. For more information, see the descriptions of the DTIF\$\_CLR aggregate, DTIF\$\_RWR aggregate, DTIF\$\_COR aggregate, and the DTIF\$\_NMR aggregate.

### **DTIF\$\_RNG\_SORT\_KEYNUM**

#### **Encoding: integer**

An optional sort key item that identifies the cardinal sort key (1 = primary, 2 = secondary, . . . ) for sort ranges.



---

## DTIF\$\_ROW—Table Row Aggregate

The table row aggregate contains data that pertains to each row of the table. Each row must be encoded only once, and rows must be stored in increasing (numeric) order, that is, first row, second row, . . . , last row. The DTIF\$\_ROW aggregate is referenced by the parent aggregate item DTIF\$\_TBL\_ROWS.

Refer to these corresponding syntax diagrams:

| Syntax  | Location    |
|---------|-------------|
| RowDefn | Figure C-12 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DTIF\$_ROW_NUM          | Integer                           |
| DTIF\$_ROW_APPL_PRIVATE | Sequence of DTIF\$_NVL aggregates |
| DTIF\$_ROW_FORMATS      | Sequence of DTIF\$_FMI aggregates |
| DTIF\$_ROW_FLAGS        | Longword                          |
| DTIF\$_ROW_CELLS        | Sequence of DTIF\$_CLD aggregates |

---

### AGGREGATE ITEMS

#### ***DTIF\$\_ROW\_NUM***

##### ***Encoding: integer***

An optional identifier item that identifies the row. This item must be encoded in increasing numerical order, starting with 1 and increasing sequentially by 1. If this item is omitted, the value is derived by adding 1 to the value of the previous DTIF\$\_ROW\_NUM item (if there is no previous row, the value is assumed to be 1). When sequential rows are encoded, it is necessary to specify this item only for the first row of the sequence; the subsequent values for this item can be omitted.

#### ***DTIF\$\_ROW\_APPL\_PRIVATE***

##### ***Encoding: sequence of DTIF\$\_NVL aggregates***

An optional private row data item that contains application-private data about the row not currently standardized by DTIF. All interpretations of the private data are subject only to private agreements between the parties concerned. For more information, see the description of the DTIF\$\_NVL aggregate.

#### ***DTIF\$\_ROW\_FORMATS***

##### ***Encoding: sequence of DTIF\$\_FMI aggregates***

An optional format item that specifies the default format attributes for each cell stored within this row. This item's attributes apply to the current row; they do not apply to any other row within the table. In addition, this item's attributes

## DTIF\$\_ROW

are applied only if the `dtif$m_tmd_fmtbyrow` flag value is enabled by the `DTIF$_TMD_FLAGS` item. For more information on the `DTIF$_TMD_FLAGS` item, see the description of the `DTIF$_TMD` aggregate. For more information, see the description of the `DTIF$_FMI` aggregate.

If the `dtif$m_tmd_fmtbycol` flag value is enabled, this item's attributes have no effect upon the cells of the row; in this case, the attributes specified for the column corresponding to the cell are used instead. For more information on the `dtif$m_tmd_fmtbycol` flag value, see the description of the `DTIF$_TMD` aggregate.

### ***DTIF\$\_ROW\_FLAGS***

***Encoding: longword***

An optional flags item that defines attributes for the row. Valid values for this item are as follows:

`dtif$m_row_annotation`

If set, this row does not contain actual data and is used for annotation purposes only. For example, it can contain row separators consisting of dashes.

### ***DTIF\$\_ROW\_CELLS***

***Encoding: sequence of DTIF\$\_CLD aggregates***

An optional cell item that specifies a sequence of one or more cells in this row. If the row contains no cell data, this item can be omitted. For more information, see the description of the `DTIF$_CLD` aggregate.

---

## DTIF\$\_RWR—DTIF Row Range Aggregate

The row range aggregate defines a range using starting and ending row numbers. A row range differs from a cell range in that a row range refers to an indeterminate number of cells. A single-row range is indicated by the absence of an ending row. The DTIF\$\_RWR aggregate is referenced by the parent aggregate item DTIF\$\_RNG\_REGION.

Refer to these corresponding syntax diagrams:

| Syntax   | Location    |
|----------|-------------|
| RowRange | Figure C-40 |

---

### AGGREGATE FORMAT

| Item Name            | Item Encoding |
|----------------------|---------------|
| DTIF\$_RWR_ROW_BEGIN | Integer       |
| DTIF\$_RWR_ROW_END   | Integer       |

---

### AGGREGATE ITEMS

***DTIF\$\_RWR\_ROW\_BEGIN***

***Encoding: integer***

A row-begin data item that specifies the starting row number that corresponds to the first row in the range.

***DTIF\$\_RWR\_ROW\_END***

***Encoding: integer***

An optional row-end data item that specifies the ending row number that corresponds to the last row in the range.

## DTIF\$\_TBL

---

### DTIF\$\_TBL—Table Definition Aggregate

The table definition aggregate contains data that pertains to a single table stored within the DTIF document. Each table is considered to be independent of every other table in the document.

However, information stored in the document header, such as language preference tables, external references, and generic attributes, can be referenced from any table within the document. The DTIF\$\_TBL aggregate is referenced by the parent aggregate item DTIF\$\_DTF\_TABLES.

Refer to these corresponding syntax diagrams:

| Syntax    | Location   |
|-----------|------------|
| TableDefn | Figure C-9 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DTIF\$_TBL_MAX_COLS     | Integer                           |
| DTIF\$_TBL_MAX_ROWS     | Integer                           |
| DTIF\$_TBL_APPL_PRIVATE | Sequence of DTIF\$_NVL aggregates |
| DTIF\$_TBL_METADATA     | Handle of DTIF\$_TMD aggregate    |
| DTIF\$_TBL_WINDOWS      | Sequence of DTIF\$_WND aggregates |
| DTIF\$_TBL_ROWS         | Sequence of DTIF\$_ROW aggregates |

---

### AGGREGATE ITEMS

#### **DTIF\$\_TBL\_MAX\_COLS**

**Encoding:** *integer*

An optional number of columns item that specifies the highest-numbered column encoded in the table (that is, the DTIF\$\_CAT\_ID of the rightmost column).

Decoding applications can use this item with the DTIF\$\_TBL\_MAX\_ROWS item to determine the table dimensions.

Decoding applications can also use this information to decide whether they are able either to process a table of this size or to preallocate data structures prior to reading the table data. This item must contain a nonnegative integer value. (A value of 0 indicates that there are no columns in the table.)

If the table contains empty columns, the number of columns actually encoded in the table can be less than the value of this item. However, the number of columns encoded in the table must not exceed the value of this item.

If this item is omitted, the number of columns in the table is considered *unknown*. (It is possible that the encoding application does not know the table dimensions at the time the table is encoded.) Decoding applications must choose the appropriate action when reading a table of unknown size.

#### **DTIF\$\_TBL\_MAX\_ROWS**

**Encoding:** *integer*

An optional number-of-rows item that specifies the highest-numbered row encoded in the table (that is, the DTIF\$\_ROW\_NUM of the last row). Decoding applications can may use this item with the DTIF\$\_TBL\_MAX\_COLS item to determine the table dimensions.

Decoding applications can also use this information to decide whether they are able either to process a table of this size or to preallocate data structures prior to reading the table data. This item must contain a nonnegative integer value. (A value of 0 indicates that there are no rows in the table.)

If the table contains empty rows, the number of rows actually encoded in the table can be less than the value of this item. However, the number of rows encoded in the table must not exceed the value of this item.

If this item is omitted, the number of rows in the table is considered *unknown*. (It is possible that the encoding application does not know the table dimensions at the time the table is encoded.) Decoding applications must choose the appropriate action when reading a table of unknown size.

#### **DTIF\$\_TBL\_APPL\_PRIVATE**

**Encoding:** *sequence of DTIF\$\_NVL aggregates*

An optional private table data item that contains application-private information about the table not currently standardized by DTIF. All interpretations of the private data are subject only to private agreements between the parties concerned. For more information, see the description of the DTIF\$\_NVL aggregate.

#### **DTIF\$\_TBL\_METADATA**

**Encoding:** *handle of a DTIF\$\_TMD aggregate*

An optional descriptive table information item that contains information pertaining to the table as a whole. For more information, see the description of the DTIF\$\_TMD aggregate.

#### **DTIF\$\_TBL\_WINDOWS**

**Encoding:** *sequence of DTIF\$\_WND aggregates*

An optional window item that contains a list of windows defined for the table. For more information, see the description of the DTIF\$\_WND aggregate.

#### **DTIF\$\_TBL\_ROWS**

**Encoding:** *sequence of DTIF\$\_ROW aggregates*

An optional rows item that contains a list of rows defined for the table. For more information, see the description of the DTIF\$\_ROW aggregate.

## DTIF\$\_TMD

---

### DTIF\$\_TMD—Table Metadata Aggregate

The table metadata aggregate describes the organization and structure of a table. The DTIF\$\_TMD aggregate is referenced by the parent aggregate item DTIF\$\_TBL\_METADATA.

Refer to these corresponding syntax diagrams:

| Syntax  | Location    |
|---------|-------------|
| TableMd | Figure C-10 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DTIF\$_TMD_NAME         | Array of type character string    |
| DTIF\$_TMD_ID           | Integer                           |
| DTIF\$_TMD_APPL_PRIVATE | Sequence of DTIF\$_NVL aggregates |
| DTIF\$_TMD_DESCRIPTION  | Array of type character string    |
| DTIF\$_TMD_FLAGS        | Longword                          |
| DTIF\$_TMD_DEFAULT_FMTS | Sequence of DTIF\$_FMI aggregates |
| DTIF\$_TMD_COLUMNS      | Sequence of DTIF\$_CAT aggregates |
| DTIF\$_TMD_RANGES       | Sequence of DTIF\$_RNG aggregates |
| DTIF\$_TMD_SYMBOLS      | Sequence of DTIF\$_NVL aggregates |

---

### AGGREGATE ITEMS

#### **DTIF\$\_TMD\_NAME**

**Encoding:** *array of type character string*

An optional table name item that identifies the table. This item must be unique among all tables within the document.

#### **DTIF\$\_TMD\_ID**

**Encoding:** *integer*

An optional table identifier item that uniquely identifies the table, is used as a shorthand reference to this table, and is valid only within the DTIF document.

#### **DTIF\$\_TMD\_APPL\_PRIVATE**

**Encoding:** *sequence of DTIF\$\_NVL aggregates*

An optional private table data item that contains application-private information about the table not currently standardized by DTIF. All interpretations of the private data are subject only to private agreements between the parties concerned. For more information, see the description of the DTIF\$\_NVL aggregate.

**DTIF\$\_TMD\_DESCRIPTION****Encoding: array of type character string**

An optional table descriptor item that describes this table, its revision history, and restrictions.

**DTIF\$\_TMD\_FLAGS****Encoding: longword**

A flags item that controls table attributes. The following table lists the possible flag values. These flag values are grouped by function; you can select only one value from each functional group.

| <b>Automatic Operations</b> |  |
|-----------------------------|--|
| dtif\$m_tmd_autorecalc      | If set, automatic recalculation mode is enabled. Each time a value in the table is changed, all cells that depend on the changed value are automatically recalculated. |
| dtif\$m_tmd_autoresort      | If set, automatic re-sort mode is enabled (used in conjunction with sort ranges).  |
| <b>Calculation Controls</b> |  |
| dtif\$m_tmd_calbycol        | If set, calculation order is by column. Cell values are calculated in column-major order.  |
| dtif\$m_tmd_calbyrow        | If set, calculation order is by row. Cell values are calculated in row-major order.  |
| dtif\$m_tmd_calnatural      | If set, natural ordering is used for calculations. Cell values are calculated based upon cell dependencies.  |
| <b>Format Attributes</b>    |  |
| dtif\$m_tmd_fmtbycol        | If set, column format attributes take precedence over row format attributes.   |
| dtif\$m_tmd_fmtbyrow        | If set, row format attributes take precedence over column format attributes.   |

The default is dtif\$m\_tmd\_fmtbycol.

**DTIF\$\_TMD\_DEFAULT\_FMTS****Encoding: sequence of DTIF\$\_FMI aggregates**

An optional default format item that specifies the default format attributes for values in this table. The format attributes are based on data type (text, integer, or date). The attributes specified for the table become the default attributes for all column attributes, rows, and cells within the table, unless they are redefined at one of those levels. For more information, see the description of the DTIF\$\_FMI aggregate.

**DTIF\$\_TMD\_COLUMNS****Encoding: sequence of DTIF\$\_CAT aggregates**

An optional column attributes item that is a list of columns encoded within the table. For more information, see the description of the DTIF\$\_CAT aggregate.

## DTIF\$\_TMD

Each column in the table must be defined by a table column attributes type stored in this item; that is, for every cell encoded in the DTIF\$\_ROW aggregate, a DTIF\$\_CAT aggregate whose DTIF\$\_CAT\_ID is equal to the DTIF\$\_CLD\_COL\_NUM of the cell must be encoded in this item. For more information, see the description of the DTIF\$\_ROW aggregate.

For more information on the DTIF\$\_CLD\_COL\_NUM item, see the description of the DTIF\$\_CLD aggregate.

DTIF does not support implied column definitions, in which a column definition is derived from the existence of a cell stored within the column.

Every column defined in this item need not have cell data associated with it, but can instead be empty. In this case, this item is used to “hold a place” for the column within the table.

Although this item is optional, it may be omitted only if the DTIF\$\_ROW aggregate is also omitted. In this case, the table contains neither rows nor columns. It is valid to specify this item, yet still omit the DTIF\$\_ROWS aggregate. In this case, the table is defined by its columns, but contains neither rows nor cells.

### ***DTIF\$\_TMD\_RANGES***

#### ***Encoding: sequence of DTIF\$\_RNG aggregates***

An optional ranges item that specifies a logically grouped set of cell values. For more information, see the description of the DTIF\$\_RNG aggregate.

### ***DTIF\$\_TMD\_SYMBOLS***

#### ***Encoding: sequence of DTIF\$\_NVL aggregates***

An optional symbols item that specifies a list of symbols that may be referenced by expressions within a table by means of the CFE\$K\_IDENTIFIER function. For more information on the CFE\$K\_IDENTIFIER function, see the description of the CFE\$\_EXL aggregate. For more information, see the description of the DTIF\$\_NVL aggregate.



---

## DTIF\$\_VTX—DTIF Varying Text Aggregate

The varying text aggregate contains data that is used to store strings with trailing blanks without actually encoding the blank characters. The DTIF\$\_VTX aggregate is referenced by the parent aggregate items DTIF\$\_CAT\_DEFAULT\_VALUE, DTIF\$\_CAT\_MISSING\_VALUE, and DTIF\$\_CLD\_VALUE.

Refer to these corresponding syntax diagrams:

| Syntax      | Location    |
|-------------|-------------|
| VaryingText | Figure C-15 |

---

### AGGREGATE FORMAT

| Item Name            | Item Encoding    |
|----------------------|------------------|
| DTIF\$_VTX_VTEXT_LEN | Integer          |
| DTIF\$_VTX_VTEXT_STR | Character string |

---

### AGGREGATE ITEMS

#### ***DTIF\$\_VTX\_VTEXT\_LEN***

***Encoding: integer***

A text length item that specifies the defined length (number of characters) in the string.

#### ***DTIF\$\_VTX\_VTEXT\_STR***

***Encoding: character string***

A character string data item that defines the varying text.

The length of this item must not exceed the value of the DTIF\$\_VTX\_VTEXT\_LEN item. If DTIF\$\_VTX\_VTEXT\_LEN exceeds the length of DTIF\$\_VTX\_VTEXT\_STR, the string must be padded with enough trailing blanks for the string length to reach the DTIF\$\_VTX\_VTEXT\_LEN value.

## DTIF\$\_WND

---

### DTIF\$\_WND—Table Windows Aggregate

The table window aggregate describes a particular view of a table and is used primarily by spreadsheet applications. The table window aggregate defines display-specific information pertinent to a table. The DTIF\$\_WND aggregate is referenced by the parent aggregate item DTIF\$\_TBL\_WINDOWS.

Refer to these corresponding syntax diagrams:

---

| Syntax     | Location    |
|------------|-------------|
| WindowDefn | Figure C-11 |

---

### AGGREGATE FORMAT

---

| Item Name               | Item Encoding                     |
|-------------------------|-----------------------------------|
| DTIF\$_WND_NAME         | Array of type character string    |
| DTIF\$_WND_ID           | Integer                           |
| DTIF\$_WND_APPL_PRIVATE | Sequence of DTIF\$_NVL aggregates |
| DTIF\$_WND_CARDINAL_NUM | Integer                           |
| DTIF\$_WND_DESCRIPTION  | Array of type character string    |
| DTIF\$_WND_FLAGS        | Longword                          |
| DTIF\$_WND_FORMATS      | Sequence of DTIF\$_FMI aggregates |
| DTIF\$_WND_RANGES       | Sequence of DTIF\$_RNG aggregates |
| DTIF\$_WND_ACTIVE_LOC   | Handle of DTIF\$_CCD aggregate    |

---

### AGGREGATE ITEMS

#### **DTIF\$\_WND\_NAME**

**Encoding:** *array of type character string*

An optional window name item that identifies the window. This item must be unique among all windows specified in the document.

#### **DTIF\$\_WND\_ID**

**Encoding:** *integer*

An optional window identifier item that identifies the window and that is used to reference this window within the DTIF\$\_FMI aggregate. Window numbering begins at 1 and increases sequentially. Window 1 defines the initial screen display or “main” window. This item is referenced by the DTIF\$\_FMI\_WINDOW\_ID item. For more information, see the description of the DTIF\$\_FMI aggregate.

**DTIF\$\_WND\_APPL\_PRIVATE****Encoding: sequence of DTIF\$\_NVL aggregates**

An optional private window data item that contains application-private information about the window not currently standardized by DTIF. All interpretations of the private data are subject only to private agreements between the parties concerned. For more information, see the description of the DTIF\$\_NVL aggregate.

**DTIF\$\_WND\_CARDINAL\_NUM****Encoding: integer**

An optional cardinal number item that specifies the relative importance of this window, that is, whether it is the primary (1), secondary (2), . . . , window for this table. This item may or may not have the same value as that for the DTIF\$\_WND\_ID item. It can be used to determine the displayed order of the windows.

**DTIF\$\_WND\_DESCRIPTION****Encoding: array of type character string**

An optional window descriptor item that describes this window.

**DTIF\$\_WND\_FLAGS****Encoding: longword**

An optional flags item that defines attributes for the window. Valid values for this item are as follows:

|                            |  |
|----------------------------|--|
| dtif\$m_wnd_active         | If set, this window is currently active (has input focus).                                     |
| dtif\$m_wnd_hidden         | If set, this window is currently hidden (not displayed).                                       |
| dtif\$m_wnd_formula_hidden | If set, this window's cell formulas are currently hidden (not displayed). This is the default. |
| dtif\$m_wnd_value_hidden   | If set, this window's cell values are currently hidden (not displayed).                        |
| dtif\$m_wnd_colhdr_hidden  | If set, this window's column headers are currently hidden (not displayed).                     |
| dtif\$m_wnd_rowhdr_hidden  | If set, this window's row headers are currently hidden (not displayed).                        |
| dtif\$m_wnd_lines_hidden   | If set, this window's grid lines are currently hidden (not displayed).                         |

**DTIF\$\_WND\_FORMATS****Encoding: sequence of DTIF\$\_FMI aggregates**

An optional format item that specifies the default format attributes for values displayed in this window. The format attributes are based on data type (text, integer, or date). For more information, see the description of the DTIF\$\_FMI aggregate.

**DTIF\$\_WND\_RANGES****Encoding: sequence of DTIF\$\_RNG aggregates**

An optional ranges item that specifies a list of ranges used within the window. For more information, see the description of the DTIF\$\_RNG aggregate.

## DTIF\$\_WND

### ***DTIF\$\_WND\_ACTIVE\_LOC***

***Encoding: handle of a DTIF\$\_CCD aggregate***

An optional active location item that specifies the active, or current, cell within the window. For more information, see the description of the DTIF\$\_CCD aggregate.

This chapter provides a description of each CFE-supported aggregate structure.

---

## 6.1 CFE Generic Aggregate Items

In addition to the items defined by each individual aggregate, the CDA Toolkit also supports two “generic” aggregate items that can be specified for every CFE aggregate described in this chapter. Table 6–1 lists and describes these items.

**Table 6–1: CFE Generic Aggregate Items**

| <b>Item Name</b>     | <b>Encoding</b> | <b>Meaning</b>  |
|----------------------|-----------------|---|
| CFE\$_USER_CONTEXT   | Longword        | Specifies additional longword for user                |
| CFE\$_AGGREGATE_TYPE | Word            | Specifies the type of the aggregate; a read-only item |

## CFE\$\_CCD

---

# CFE\$\_CCD—CFE Cell Coordinates Aggregate

The cell coordinates aggregate contains data that specifies a particular cell in a spreadsheet table and is referenced by the CFE\$\_K\_CELL\_COORD value for the CFE\$\_EXL aggregate. For more information, see the description of the CFE\$\_EXL aggregate.

The CFE\$\_CCD aggregate is referenced by the parent aggregate items CFE\$\_CLR\_RANGE\_BEGIN, CFE\$\_CLR\_END, and CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| ExpressionList | Figure D-5  |
| CellCoord      | Figure C-36 |

---

## AGGREGATE FORMAT

| Item Name        | Item Encoding |
|------------------|---------------|
| CFE\$_CCD_ROW    | Integer       |
| CFE\$_CCD_COLUMN | Integer       |
| CFE\$_CCD_FLAGS  | Enumeration   |

---

## AGGREGATE ITEMS

### **CFE\$\_CCD\_ROW**

**Encoding: integer**

A row identifier item that indicates the row number.

### **CFE\$\_CCD\_COLUMN**

**Encoding: integer**

A column identifier item that indicates the column number.

### **CFE\$\_CCD\_FLAGS**

**Encoding: enumeration**

A flags item that indicates whether the row and column references are relative or absolute. A relative reference indicates that the reference can be updated to reflect the position of the cell's new location relative to its old location. If the decoding application does not support this updating, it can ignore this item.

An absolute reference to a row or column indicates that the reference can remain unchanged wherever the cell is being copied to within a table. Absolute references are usually specified in spreadsheet programs by prefixing a character, such as a dollar sign (\$), before the row or column identifier in a cell coordinate name.

Valid values for this item are as follows:

|                     |                              |
|---------------------|------------------------------|
| CFE\$_RELROW_RELCOL | Relative Row/Relative Column |
| CFE\$_RELROW_ABSCOL | Relative Row/Absolute Column |
| CFE\$_ABSROW_RELCOL | Absolute Row/Relative Column |
| CFE\$_ABSROW_ABSCOL | Absolute Row/Absolute Column |

The default is CFE\$\_RELROW\_RELCOL.

## CFE\$\_CFT

---

### CFE\$\_CFT—CFE Complex Float Aggregate

The complex float aggregate models a complex floating-point value. The CFE\$\_CFT aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| ExpressionList | Figure D-5  |
| ComplexFloat   | Figure C-17 |

---

### AGGREGATE FORMAT

| Item Name                | Item Encoding          |
|--------------------------|------------------------|
| CFE\$_CFT_REAL_PART      | General floating-point |
| CFE\$_CFT_IMAGINARY_PART | General floating-point |

---

### AGGREGATE ITEMS

**CFE\$\_CFT\_REAL\_PART**

**Encoding: general floating-point**

A real part item that specifies the real portion of the complex number.

**CFE\$\_CFT\_IMAGINARY\_PART**

**Encoding: general floating-point**

An imaginary part item that specifies the imaginary portion of the complex number.



---

## CFE\$\_CLR—CFE Cell Range Aggregate

The cell range aggregate defines explicit starting and ending cells using two cell references. The column letters can be encoded as integer values. The CFE\$\_CLR aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| ExpressionList | Figure D-5  |
| ColRange       | Figure C-41 |

---

### AGGREGATE FORMAT

| Item Name             | Item Encoding                 |
|-----------------------|-------------------------------|
| CFE\$_CLR_RANGE_BEGIN | Handle of CFE\$_CCD aggregate |
| CFE\$_CLR_RANGE_END   | Handle of CFE\$_CCD aggregate |

---

### AGGREGATE ITEMS

#### **CFE\$\_CLR\_RANGE\_BEGIN**

**Encoding:** *handle of a CFE\$\_CCD aggregate*

A range-begin item that is the starting cell in the range (the upper left cell). For more information, see the description of the CFE\$\_CCD aggregate.

#### **CFE\$\_CLR\_RANGE\_END**

**Encoding:** *handle of a CFE\$\_CCD aggregate*

An optional range-end item that is the ending cell in the range (the lower right cell). If this item is omitted, it indicates a single-cell range. For more information, see the description of the CFE\$\_CCD aggregate.

## CFE\$\_COR

---

### CFE\$\_COR—CFE Column Range Aggregate

The column range aggregate defines a range using starting and ending column numbers. A column range differs from a cell range in that a column range refers to an indeterminate number of cells. The CFE\$\_COR aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| ExpressionList | Figure D-5  |
| ColRange       | Figure C-41 |

---

### AGGREGATE FORMAT

| Item Name           | Item Encoding |
|---------------------|---------------|
| CFE\$_COR_COL_BEGIN | Integer       |
| CFE\$_COR_COL_END   | Integer       |

---

### AGGREGATE ITEMS

#### **CFE\$\_COR\_COL\_BEGIN**

**Encoding:** *integer*

A column-begin item that specifies the starting column number (the first column in the range).

#### **CFE\$\_COR\_COL\_END**

**Encoding:** *integer*

An optional column-end item that specifies the ending column number (the last column in the range). If this item is omitted, it indicates a single-column range.

---

## CFE\$\_DAT—CFE Date and Time Aggregate

The CFE date and time aggregate specifies a date/time value that is defined as a sequence of two octet strings. The first octet string defines the date and time as a binary value; the second octet string defines an optional time differential. The CFE\$\_DAT aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| ExpressionList | Figure D-5  |
| DateTime       | Figure C-29 |

---

### AGGREGATE FORMAT

| Item Name             | Item Encoding |
|-----------------------|---------------|
| CFE\$_DAT_DATETIME    | String        |
| CFE\$_DAT_TIME_DIFF_C | Enumeration   |
| CFE\$_DAT_TIME_DIFF   | Variable      |

---

### AGGREGATE ITEMS

#### **CFE\$\_DAT\_DATETIME**

**Encoding:** *string*

A date and time item that is a sequence of octets representing a date/time value. Each octet is interpreted as an unsigned integer value, as shown in the following table.

| Octet Number | Date/Time Element  |
|--------------|--|
| 0            | Century digits in the range of values from 0 to 99; for example, 19 in the year 1967 |
| 1            | Year digits in the range of values from 0 to 99; for example, 67 in the year 1967    |
| 2            | Month in the range of values from 1 to 12  |
| 3            | Day in the range of values from 1 to 31  |
| 4            | Hour in the range of values from 0 to 23   |
| 5            | Minute in the range of values from 0 to 59   |
| 6            | Second in the range of values from 0 to 59   |
| 7            | Hundredths of seconds in the range of values from 0 to 99                            |

## CFE\$ \_DAT

### **CFE\$ \_DAT\_TIME\_DIFF\_C**

#### **Encoding: enumeration**

An optional time difference indicator that specifies the type of time differential value chosen from those that are delineated for the time differential item. Valid values for this item are as follows:

|                  |  |
|------------------|--|
| CFE\$K.UTC.TIME  | A value that represents Coordinate Universal Time (UTC), or Greenwich Mean Time. This is equivalent to a time differential of 0 hours, 0 seconds. In this case, the CFE\$ _DAT_TIME_DIFF item is left blank. |
| CFE\$K.PLUS.DIFF | A string that is a positive time differential is represented by a sequence of two octets, as shown in the following table. In this case, the CFE\$ _DAT_TIME_DIFF item is encoded as a string.               |
| CFE\$K.NEG.DIFF  | A string that is a negative time differential is represented by a sequence of two octets, as shown in the following table. In this case, the CFE\$ _DAT_TIME_DIFF item is encoded as a string.               |

The following table shows how the two octets that represent the encoding of the DAT\_TIME\_DIFF item for the CFE\$K.PLUS.DIFF and CFE\$K.NEG.DIFF cases are encoded.

| Octet Number | Date/Time Element  |
|--------------|--|
| 0            | Hours in the range of values from 0 to 13 for CFE\$K.PLUS.DIFF and in the range of values from 0 to 12 for CFE\$K.NEG.DIFF |
| 1            | Minutes in the range of values from 0 to 59  |

### **CFE\$ \_DAT\_TIME\_DIFF**

#### **Encoding: variable**

A time difference item that contains the actual time differential for the time differential type selected in the previous item.

If the CFE\$ \_DAT\_TIME\_DIFF\_C item is present, the CFE\$ \_DAT\_DATETIME item represents Coordinate Universal Time, and the value chosen for CFE\$ \_DAT\_TIME\_DIFF represents the local time differential.

If the CFE\$ \_DAT\_TIME\_DIFF\_C item is not present, the value specified for the CFE\$ \_DAT\_DATETIME item represents local time.

---

## CFE\$\_EXL—Expression List Aggregate

The expression list aggregate contains a list of all the expressions defined by CFE. The CFE\$\_EXL aggregate is referenced by the parent aggregate items CFE\$\_EXL\_EXPR\_C, CFE\$\_EXP\_LIST, CFE\$\_NPM\_VALUE, CFE\$\_PEX\_VALUE\_EXPR, CFE\$\_SLL\_CRITERIA, CFE\$\_SLL\_SELECTION, CFE\$\_STF\_SOURCE, and CFE\$\_STP\_STRING\_EXPR.

Refer to these corresponding syntax diagrams:

| Syntax            | Location    |
|-------------------|-------------|
| ExpressionList    | Figure D-5  |
| PrivateFuncExpr   | Figure D-1  |
| ParenthesizedExpr | Figure D-11 |
| FieldRef          | Figure D-12 |
| Text              | Figure D-6  |
| VaryingText       | Figure D-7  |
| SelectorList      | Figure D-8  |
| DecimalString     | Figure D-9  |
| EditString        | Figure D-10 |
| ComplexFloat      | Figure C-17 |
| DateTime          | Figure C-29 |
| ColNum            | Figure C-34 |
| RowNum            | Figure C-35 |
| CellCoord         | Figure C-36 |
| CellRange         | Figure C-39 |
| RowRange          | Figure C-40 |
| ColRange          | Figure C-41 |
| NamedRange        | Figure C-42 |

---

### AGGREGATE FORMAT

| Item Name        | Item Encoding |
|------------------|---------------|
| CFE\$_EXL_EXPR_C | Enumeration   |
| CFE\$_EXL_EXPR   | Variable      |

# CFE\$\_EXL

## AGGREGATE ITEMS

### CFE\$\_EXL\_EXPR\_C

#### Encoding: enumeration

An expression indicator that specifies the types of values chosen from those that are delineated for this expression item.

Valid arithmetic expression values for this item are shown in Table 6–2.

Table 6–2: Valid Arithmetic Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$K_ADD           | Sequence of CFE\$_EXL aggregates | <i>Addition</i> adds the second value ( <b>addend2</b> ) to the first value ( <b>addend1</b> ). Argument list order: <b>addend1, addend2</b>                                       |
| CFE\$K_DIVIDE        | Sequence of CFE\$_EXL aggregates | <i>Division</i> divides the first value expression ( <b>dividend</b> ) by the second value expression ( <b>divisor</b> ). Argument list order: <b>dividend, divisor</b>            |
| CFE\$K_MULTIPLY      | Sequence of CFE\$_EXL aggregates | <i>Multiplication</i> multiplies the first expression ( <b>multiplicand</b> ) by the second expression ( <b>multiplier</b> ). Argument list order: <b>multiplicand, multiplier</b> |
| CFE\$K_NEGATE        | Sequence of CFE\$_EXL aggregates | <i>Negation</i> negates a value ( <b>source</b> ), returning – source. Argument list order: <b>source</b>  |
| CFE\$K_PERCENT       | Sequence of CFE\$_EXL aggregates | <i>Percent</i> indicates that the value is to be calculated as a percentage (divided by 100). Argument list order: <b>value1</b>   |
| CFE\$K_POWER         | Sequence of CFE\$_EXL aggregates | <i>Raise to a power</i> raises the first value ( <b>base</b> ) to the power specified by the second expression ( <b>power</b> ). Argument list order: <b>base, power</b>           |

(continued on next page)

Table 6–2 (Cont.): Valid Arithmetic Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$_K_SUBTRACT     | Sequence of CFE\$_EXL aggregates | <i>Subtraction</i> subtracts the second expression ( <b>subtrahend</b> ) from the first expression ( <b>minuend</b> ). Argument list order: <b>minuend, subtrahend</b>   |
| CFE\$_K_UNARY_PLUS   | Sequence of CFE\$_EXL aggregates | <i>Unary plus</i> indicates that the value is explicitly positive. UNARY_PLUS is the opposite of NEGATE. While UNARY_PLUS is not needed for calculation, it can be important for data interchange and for certain applications. Argument list order: <b>value1</b> |

Valid binary expression values for this item are shown in Table 6–3.

Table 6–3: Valid Binary Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$_K_AS_L         | Sequence of CFE\$_EXL aggregates | <i>Arithmetic shift left</i> performs arithmetic shift left by shifting the bits in the first expression to the left by the number of bits specified by the second expression. Argument list order: <b>source, shift-count</b>    |
| CFE\$_K_AS_R         | Sequence of CFE\$_EXL aggregates | <i>Arithmetic shift right</i> performs arithmetic shift right by shifting the bits in the first expression to the right by the number of bits specified by the second expression. Argument list order: <b>source, shift-count</b> |
| CFE\$_K_ONES_CMP     | Sequence of CFE\$_EXL aggregates | <i>One's complement</i> returns the one's complement of a value ( <b>source</b> ). Argument list order: <b>source</b>   |

# CFE\$ \_EXL

Valid Boolean and relational expression values for this item are shown in Table 6-4.

**Table 6-4: Valid Boolean and Relational Expression Values for CFE\$ \_EXL\_EXPR\_C**

| <b>Expression Indicator</b> | <b>Expression Value</b>           | <b>Description and Storage Method</b>   |
|-----------------------------|-----------------------------------|---|
| CFE\$K_ABS_VALUE            | Sequence of CFE\$ _EXL aggregates | <i>Absolute value</i> calculates the absolute value of an expression ( <b>source</b> ).<br>Argument list order:<br><b>source</b>  |
| CFE\$K_AND                  | Sequence of CFE\$ _EXL aggregates | <i>Logical AND</i> performs the logical AND of two Boolean expressions.<br>Argument list order:<br><b>value1 [BOOLEAN], value2 [BOOLEAN]</b>  |
| CFE\$K_BETWEEN              | Sequence of CFE\$ _EXL aggregates | <i>Between</i> returns TRUE if the first expression is between the values of the second and third expressions and returns FALSE otherwise. Argument list order: <b>value1, value2, value3</b> |
| CFE\$K_EQL                  | Sequence of CFE\$ _EXL aggregates | <i>Equal to</i> returns TRUE if <b>value1</b> is equal to <b>value2</b> and FALSE otherwise. Argument list order: <b>value1, value2</b>   |
| CFE\$K_GEQ                  | Sequence of CFE\$ _EXL aggregates | <i>Greater than or equal to</i> returns TRUE if <b>value1</b> is greater than or equal to <b>value2</b> and returns FALSE otherwise. Argument list order: <b>value1, value2</b>               |
| CFE\$K_GTR                  | Sequence of CFE\$ _EXL aggregates | <i>Greater than</i> returns TRUE if <b>value1</b> is greater than <b>value2</b> and returns FALSE otherwise. Argument list order: <b>value1, value2</b>                                       |

(continued on next page)



Table 6-4 (Cont.): Valid Boolean and Relational Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_IF_THEN_ELSE  | Sequence of CFE\$_EXL aggregates | <i>If-then-else</i> evaluates <b>if-expr</b> (first argument). If the test is TRUE, it returns the value of <b>then-expr</b> . If the test is FALSE, it returns the value of <b>else-expr</b> . If the <b>else-expr</b> is not specified, the decoding application selects the appropriate return value. Argument list order: <b>if-expr, then-expr, else-expr (optional)</b> |
| CFE\$K_LEQ           | Sequence of CFE\$_EXL aggregates | <i>Less than or equal to</i> returns TRUE if <b>value1</b> is less than or equal to <b>value2</b> and returns FALSE otherwise. Argument list order: <b>value1, value2</b>   |
| CFE\$K_LSS           | Sequence of CFE\$_EXL aggregates | <i>Less than</i> returns TRUE if <b>value1</b> is less than <b>value2</b> and returns FALSE otherwise. Argument list order: <b>value1, value2</b>   |
| CFE\$K_MODULO        | Sequence of CFE\$_EXL aggregates | <i>Modulus</i> returns the remainder obtained when the first expression ( <b>dividend</b> ) is divided by the second expression ( <b>divisor</b> ). Argument list order: <b>dividend, divisor</b>   |
| CFE\$K_NEQ           | Sequence of CFE\$_EXL aggregates | <i>Not equal to</i> returns TRUE if <b>value1</b> is not equal to <b>value2</b> and returns FALSE otherwise. Argument list order: <b>value1, value2</b>   |
| CFE\$K_NOT           | Sequence of CFE\$_EXL aggregates | <i>Logical NOT</i> performs the logical negation of a Boolean expression. Argument list order: <b>boolean-expr</b>  |

(continued on next page)

## CFE\$\_EXL

Table 6-4 (Cont.): Valid Boolean and Relational Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_OR            | Sequence of CFE\$_EXL aggregates | <i>Logical OR</i> performs the logical OR of two Boolean expressions. Argument list order: <b>value1, value2</b>  |
| CFE\$K_SQRT          | Sequence of CFE\$_EXL aggregates | <i>Square root</i> returns the square root of an expression ( <b>source</b> ). Argument list order: <b>source</b> |

Valid cell-related expression values for this item are shown in Table 6-5.

Table 6-5: Valid Cell-Related Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_CELL_COL      | Sequence of CFE\$_EXL aggregates | <i>Column portion of cell name</i> returns the column number of a cell coordinate. Argument list order: <b>cell coordinate</b>  |
| CFE\$K_CELL_EXTRACT  | Sequence of CFE\$_EXL aggregates | <i>Cell extract</i> extracts the value of a cell coordinate (first argument) from the name or index into a list of external references (see the description of the DTIF\$_ERF aggregate) specified in the second argument by using the password, if specified, to open the table. Argument list order: <b>cell coordinate, table name (string) or an index into a list of external references (only if used within a DTIF table), password (string; optional)</b> |

(continued on next page)

Table 6-5 (Cont.): Valid Cell-Related Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$K_CELL_INDIRECT | Sequence of CFE\$_EXL aggregates | <i>Cell indirection</i> returns the contents of the cell referenced by the expression argument. The argument is expected to be a cell reference. For example, CELL_INDIRECT(A1) means to return the contents of the cell referenced by A1. In this example, if A1 contains a reference to cell B1, the CELL_INDIRECT function returns the contents of B1. Argument list order: <b>value1</b> |
| CFE\$K_CELL_NAME     | Sequence of CFE\$_EXL aggregates | <i>Constructed cell name</i> returns the name of a cell constructed from a row number and a column number. Argument list order: <b>row number, column number</b>   |
| CFE\$K_CELL_ROW      | Sequence of CFE\$_EXL aggregates | <i>Row portion of cell name</i> returns the row number of a cell coordinate. Argument list order: <b>cell coordinate</b>   |
| CFE\$K_COUNT_COLS    | Sequence of CFE\$_EXL aggregates | <i>Count columns</i> counts the total number of columns in all expressions in the expression list. Each expression should be defined as a cell or range expression. Argument list order: <b>1, 2, . . . , n cell or range expressions</b>  |

(continued on next page)

## CFE\$\_EXL

Table 6-5 (Cont.): Valid Cell-Related Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_COUNT_ROWS    | Sequence of CFE\$_EXL aggregates | <i>Count rows</i> counts the total number of rows in all expressions in the expression list. Each expression should be defined as a cell or range expression. Argument list order: <b>cell or range expression; 1, 2, . . . , n cell or range expressions</b> |
| CFE\$K_CUR_CELL      | None                             | <i>Current cell</i> returns the currently active cell coordinate.   |
| CFE\$K_CUR_COL       | None                             | <i>Current column</i> returns the current column number.  |
| CFE\$K_CUR_ROW       | None                             | <i>Current row</i> returns the current row number.  |
| CFE\$K_ERROR         | None                             | <i>Error</i> is the ERROR value.  |
| CFE\$K_NOT_AVAIL     | None                             | <i>Not available</i> is a constant value that is application dependent. Some spreadsheet applications use the constant as a function, NA(). This function has no arguments.   |
| CFE\$K_NOT_CALC      | None                             | <i>Not calculable</i> is a constant value that is application dependent. Some spreadsheet applications use this constant as a function, NC(). This function has no arguments and is also used to mean "not calculated."                                       |
| CFE\$K_NULL          | None                             | <i>Null</i> is the NULL value.  |

Valid choose and lookup expression values for this item are shown in Table 6-6.

**Table 6-6: Valid Choose and Lookup Expression Values for CFE\$\_EXL\_EXPR\_C**

| <b>Expression Indicator</b> | <b>Expression Value</b>          | <b>Description and Storage Method</b>   |
|-----------------------------|----------------------------------|---|
| CFE\$K_CHOOSE               | Sequence of CFE\$_EXL aggregates | <i>Choose</i> uses the value of the first expression as an index to return the corresponding value from the list of expressions following the index value. Argument list order: <b>index; 2, 3, . . . , n</b>   |
| CFE\$K_HLOOKUP              | Sequence of CFE\$_EXL aggregates | <i>Hlookup</i> searches the first row of <b>compare-range</b> for the largest value that is less than or equal to <b>value</b> . After finding that value, if the <b>index</b> argument is specified, it moves down the rows of <b>compare-range</b> by index number and returns the value stored in the cell. <b>Compare-range</b> should be a range expression. Argument list order: <b>value1, compare-range, index (optional)</b>                                       |
| CFE\$K_INDEX                | Sequence of CFE\$_EXL aggregates | <i>Index</i> returns a reference to a cell within a range ( <b>range</b> ), selected using the row index and column index values from the first cell of the range. If the range is a disjoint set of ranges, and the range area is specified, use the <b>range area</b> argument to select the section within the range. Then apply the row index and column index to the selected range. Argument list order: <b>range, row index, column index, range area (optional)</b> |

(continued on next page)

## CFE\$\_EXL

Table 6-6 (Cont.): Valid Choose and Lookup Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_IN_TABLE      | Sequence of CFE\$_EXL aggregates | <i>Field in table</i> returns TRUE if the <b>search-exp</b> is found in the <b>search-list</b> and returns FALSE otherwise. Argument list order: <b>search expression; search list</b>  |
| CFE\$K_MATCHES       | Sequence of CFE\$_EXL aggregates | <i>Matches</i> searches for a value in the range. If found, it returns the index into the range where the value was found. The first cell in the range is index 1. If the <b>type</b> argument is specified, its value governs the search. Type = 1 means return the highest index that matches the value. Type = -1 means return the lowest index that matches the value. Type = 0 means return the first index that matches the value. If no match is found in the range, the decoding application selects an appropriate return value. Argument list order: <b>value, range, type (optional)</b> |
| CFE\$K_TABLE         | Sequence of CFE\$_EXL aggregates | <i>Table</i> searches the first column or row of <b>compare-range</b> for the largest value that is less than or equal to <b>value</b> and returns the value of the corresponding cell in the <b>result-range</b> . Both <b>compare-range</b> and <b>result-range</b> should be single row or column ranges. Argument list order: <b>value1, compare-range, result-range</b>  |

(continued on next page)

Table 6–6 (Cont.): Valid Choose and Lookup Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$_K_VLOOKUP      | Sequence of CFE\$_EXL aggregates | <i>Vlookup</i> searches the first column of <b>compare-range</b> for the largest value that is less than or equal to <b>value1</b> . After finding that value, if the <b>index</b> argument is specified, it moves across the rows of <b>compare-range</b> using the optional index number and returns the value stored in the cell. <b>Compare-range</b> should be a range expression. Argument list order: <b>value1</b> , <b>compare-range</b> , <b>index</b> (optional) |

Valid conversion expression values for this item are shown in Table 6–7.

Table 6–7: Valid Conversion Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator   | Expression Value                 | Description and Storage Method   |
|------------------------|----------------------------------|--|
| CFE\$_K_CVT_TO_VALUE   | Sequence of CFE\$_EXL aggregates | <i>Convert to value</i> converts the argument to a numeric value. Argument list order: <b>value1</b>   |
| CFE\$_K_DECIMAL_STRING | String                           | <i>Decimal string</i> evaluates an ASCII string that is interpreted as a numeric value. For example: DECIMAL_STRING ("1.23") returns the value 1.23. |
| CFE\$_K_INT            | Sequence of CFE\$_EXL aggregates | <i>Integer</i> converts a floating-point value to an integer value. Argument list order: <b>value1</b>   |

(continued on next page)

# CFE\$\_EXL

Table 6-7 (Cont.): Valid Conversion Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$K_ROUND         | Sequence of CFE\$_EXL aggregates | <i>Round</i> rounds the source argument to the number of decimal places specified by the precision argument. If precision is not specified, the default value is 0 (round to nearest integer). If precision is a positive value, it indicates digits to the right of the decimal point. If precision is a negative value, it indicates digits to the left of the decimal point. Argument list order: <b>source, precision number of decimal places</b> |
| CFE\$K_TRUNCATE      | Sequence of CFE\$_EXL aggregates | <i>Truncate</i> truncates the source argument to the number of decimal places specified by the precision argument. If precision is not specified, the default value is 0 (truncate to nearest integer). If precision is a positive value, it indicates digits to the right of the decimal point. If precision is a negative value, it indicates digits to the left of the decimal point. Argument list order: <b>source, precision</b>                 |



Valid date and time expression values for this item are shown in Table 6–8.

**Table 6–8: Valid Date and Time Expression Values for CFE\$ \_EXL\_EXPR\_C**

| <b>Expression Indicator</b> | <b>Expression Value</b>           | <b>Description and Storage Method</b>  |
|-----------------------------|-----------------------------------|--|
| CFE\$K_CVT_TO_DATE          | Sequence of CFE\$ _EXL aggregates | <i>Convert string to date</i> converts a string expression to a date expression. The function has a single argument that represents a string value and returns a date value. Argument list order: <b>string1</b>   |
| CFE\$K_CVT_TO_TIME          | Sequence of CFE\$ _EXL aggregates | <i>Convert string to time</i> converts a string expression to a time expression. The function has a single argument that represents a string value and returns a time value. Argument list order: <b>string1</b>   |
| CFE\$K_DIFF_DAY             | Sequence of CFE\$ _EXL aggregates | <i>Difference day</i> returns the number of days between <b>date1</b> and <b>date2</b> . A positive integer indicates that <b>date2</b> is after <b>date1</b> . A negative integer indicates that <b>date2</b> is before <b>date1</b> . Argument list order: <b>date1, date2</b>       |
| CFE\$K_DIFF_HOUR            | Sequence of CFE\$ _EXL aggregates | <i>Difference hour</i> returns the number of hours between <b>date1</b> and <b>date2</b> . A positive integer indicates that <b>date2</b> is after <b>date1</b> . A negative integer indicates that <b>date2</b> is before <b>date1</b> . Argument list order: <b>date1, date2</b>     |
| CFE\$K_DIFF_MIN             | Sequence of CFE\$ _EXL aggregates | <i>Difference minute</i> returns the number of minutes between <b>date1</b> and <b>date2</b> . A positive integer indicates that <b>date2</b> is after <b>date1</b> . A negative integer indicates that <b>date2</b> is before <b>date1</b> . Argument list order: <b>date1, date2</b> |

(continued on next page)

# CFE\$\_EXL

Table 6–8 (Cont.): Valid Date and Time Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$K_DIFF_MONTH    | Sequence of CFE\$_EXL aggregates | <i>Difference month</i> returns the number of months between <b>date1</b> and <b>date2</b> . A positive integer indicates that <b>date2</b> is after <b>date1</b> . A negative integer indicates that <b>date2</b> is before <b>date1</b> . Argument list order: <b>date1, date2</b>   |
| CFE\$K_DIFF_SEC      | Sequence of CFE\$_EXL aggregates | <i>Difference second</i> returns the number of seconds between <b>date1</b> and <b>date2</b> . A positive integer indicates that <b>date2</b> is after <b>date1</b> . A negative integer indicates that <b>date2</b> is before <b>date1</b> . Argument list order: <b>date1, date2</b> |
| CFE\$K_DIFF_WEEK     | Sequence of CFE\$_EXL aggregates | <i>Difference week</i> returns the number of weeks between <b>date1</b> and <b>date2</b> . A positive integer indicates that <b>date2</b> is after <b>date1</b> . A negative integer indicates that <b>date2</b> is before <b>date1</b> . Argument list order: <b>date1, date2</b>     |
| CFE\$K_DIFF_YEAR     | Sequence of CFE\$_EXL aggregates | <i>Difference year</i> returns the number of years between <b>date1</b> and <b>date2</b> . A positive integer indicates that <b>date2</b> is after <b>date1</b> . A negative integer indicates that <b>date2</b> is before <b>date1</b> . Argument list order: <b>date1, date2</b>     |
| CFE\$K_EXT_DAY       | Sequence of CFE\$_EXL aggregates | <i>Extract day</i> returns the day portion of a date (1 to 31). Argument list order: <b>date/time source value</b>   |
| CFE\$K_EXT_HOUR      | Sequence of CFE\$_EXL aggregates | <i>Extract hour</i> returns the hour portion of a date (0 to 23). Argument list order: <b>date/time source value</b>   |

(continued on next page)

Table 6–8 (Cont.): Valid Date and Time Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$_K_EXT_MINUTE   | Sequence of CFE\$_EXL aggregates | <i>Extract minute</i> returns the minute portion of a date (0 to 59). Argument list order: <b>date/time source value</b>  |
| CFE\$_K_EXT_MONTH    | Sequence of CFE\$_EXL aggregates | <i>Extract month</i> returns the month portion of a date (1 to 12). Argument list order: <b>date/time source value</b>  |
| CFE\$_K_EXT_SECOND   | Sequence of CFE\$_EXL aggregates | <i>Extract second</i> returns the seconds portion of a date (0 to 59). Argument list order: <b>date/time source value</b>   |
| CFE\$_K_EXT_YEAR     | Sequence of CFE\$_EXL aggregates | <i>Extract year</i> returns the year portion of a date. Argument list order: <b>date /time source value</b>   |
| CFE\$_K_NAME_DAY     | Sequence of CFE\$_EXL aggregates | <i>Date day of the week</i> returns the day of the week corresponding to the date (Sunday, . . . , Saturday). If used in the context of a DTIF table, the day name must correspond to the DTIF\$_LPT_ITEMS text string assigned to represent the days of the week. For more information on the DTIF\$_LPT_ITEMS item, see the description of the DTIF\$_LPT aggregate. Argument list order: <b>date</b> |

(continued on next page)

# CFE\$\_EXL

Table 6–8 (Cont.): Valid Date and Time Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$K_NAME_DAYNUM   | Sequence of CFE\$_EXL aggregates | <i>Day of the week</i> returns the day of the week corresponding to its number (1=Sunday, 2=Monday, . . . ,7=Saturday). If used in the context of a DTIF table, the day name must correspond to the DTIF\$_LPT_ITEMS text string assigned to represent the days of the week. For more information on the DTIF\$_LPT_ITEMS item, see the description of the DTIF\$_LPT aggregate. Argument list order: <b>weekday index</b> |
| CFE\$K_NAME_MONTH    | Sequence of CFE\$_EXL aggregates | <i>Date month name</i> returns the month portion of the date as a string (January, . . . , December). If used in the context of a DTIF table, the month name must correspond to the DTIF\$_LPT_ITEMS text string assigned to represent the months of the year. For more information on the DTIF\$_LPT_ITEMS item, see the description of the DTIF\$_LPT aggregate. Argument list order: <b>date</b>                        |

(continued on next page)

Table 6–8 (Cont.): Valid Date and Time Expression Values for CFE\$ \_EXL\_EXPR\_C

| Expression Indicator | Expression Value                  | Description and Storage Method  |
|----------------------|-----------------------------------|---|
| CFE\$K_NAME_MONTHNUM | Sequence of CFE\$ _EXL aggregates | <i>Month name</i> returns the month name corresponding to its number (1=January, . . . , 12=December). If used in the context of a DTIF table, the month name must correspond to the DTIF\$_LPT_ITEMS text string assigned to represent the months of the year. For more information on the DTIF\$_LPT_ITEMS item, see the description of the DTIF\$_LPT aggregate. Argument list order: <b>month index</b> |
| CFE\$K_NOW           | None                              | <i>Now</i> returns the current date and time.   |
| CFE\$K_PLUS_DAYS     | Sequence of CFE\$ _EXL aggregates | <i>Plus days</i> returns a date that is <b>value</b> days from the date. Argument list order: <b>starting date/time, offset value (positive or negative)</b>  |
| CFE\$K_PLUS_HOURS    | Sequence of CFE\$ _EXL aggregates | <i>Plus hours</i> returns a date that is <b>value</b> hours from the date. Argument list order: <b>starting date/time, offset value (positive or negative)</b>  |
| CFE\$K_PLUS_MINS     | Sequence of CFE\$ _EXL aggregates | <i>Plus minutes</i> returns a date that is <b>value</b> minutes from the date. Argument list order: <b>starting date/time, offset value (positive or negative)</b>  |
| CFE\$K_PLUS_MONTHS   | Sequence of CFE\$ _EXL aggregates | <i>Plus months</i> returns a date that is <b>value</b> months from the date. Argument list order: <b>starting date/time, offset value (positive or negative)</b>  |

(continued on next page)

## CFE\$\_EXL

Table 6–8 (Cont.): Valid Date and Time Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$K_PLUS_SECS     | Sequence of CFE\$_EXL aggregates | <i>Plus seconds</i> returns a date that is <b>value</b> seconds from the date. Argument list order: <b>starting date/time, offset value (positive or negative)</b> |
| CFE\$K_PLUS_WEEKS    | Sequence of CFE\$_EXL aggregates | <i>Plus weeks</i> returns a date that is <b>value</b> weeks from the date. Argument list order: <b>starting date/time, offset value (positive or negative)</b>     |
| CFE\$K_PLUS_YEARS    | Sequence of CFE\$_EXL aggregates | <i>Plus years</i> returns a date that is <b>value</b> years from the date. Argument list order: <b>starting date/time, offset value (positive or negative)</b>     |
| CFE\$K_TODAY         | None                             | <i>Today</i> returns the current date (no time).   |
| CFE\$K_TOMORROW      | None                             | <i>Tomorrow</i> returns tomorrow's date, which is a day (24 hours) later than the current date. The time is not included in the result value.                      |
| CFE\$K_YESTERDAY     | None                             | <i>Yesterday</i> returns yesterday's date, which is one day (24 hours) earlier than the current date. The time is not included in the result value.                |

Valid financial expression values for this item are shown in Table 6-9.

**Table 6-9: Valid Financial Expression Values for CFE\$ \_EXL \_EXPR \_C**

| <b>Expression Indicator</b> | <b>Expression Value</b>           | <b>Description and Storage Method</b>   |
|-----------------------------|-----------------------------------|---|
| CFE\$K_APPREC               | Sequence of CFE\$ _EXL aggregates | <i>Appreciation</i> calculates a stream of values corresponding to the appreciation of a principal based on a given interest per period and an optional offset into the period. The offset must be a numeric value between 0 (the beginning of the period) and 1 (the end of the period). For example, APPREC is used to calculate the yield on a \$1200 investment over 4 years, given an annual interest rate of 11 percent, with payments made at the beginning of each year (offset = 0). Argument list order: <b>principal, interest rate per period, offset into period, start period, end period, optional section of fields or cells to receive resultant values</b> (can be a list of discontinuous range names or coordinates stored in a parenthesized expression); if this argument is not specified, the function returns a sequence of resultant values from the start period to the end period. The processing of these values is application dependent. |

(continued on next page)

# CFE\$\_EXL

Table 6-9 (Cont.): Valid Financial Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_DEP_CROSS     | Sequence of CFE\$_EXL aggregates | <p><i>Depreciation—declining balance with crossover to straight line</i> calculates the depreciation of an initial value using the declining balance method with crossover to straight line. The declining balance method is used until the amount is less than the straight line amount would be; then the straight line method is used for the remaining periods. Argument list order: <b>initial value, number of periods, percentage decline, salvage value, start period, end period, optional section of fields or cells to receive resultant values (can be a list of discontinuous range names or coordinates stored in a parenthesized expression); if this argument is not specified, the function returns a sequence of resultant values from the start period to the end period. The processing of these values is application dependent.</b></p> |

(continued on next page)



Table 6-9 (Cont.): Valid Financial Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$_K_DEP_DB       | Sequence of CFE\$_EXL aggregates | <i>Depreciation—declining balance</i> calculates the depreciation of an initial value using the declining balance method. Argument list order: <b>initial value, number of periods, percentage decline, start period, end period, optional section of fields or cells to receive resultant values</b> (can be a list of <b>discontiguous range names or coordinates stored in a parenthesized expression</b> ); if this argument is not specified, the function returns a sequence of resultant values from the start period to the end period. The processing of these values is application dependent. |
| CFE\$_K_DEP_DDB      | Sequence of CFE\$_EXL aggregates | <i>Depreciation—double declining balance</i> calculates the depreciation of an initial value using the double declining balance method. Argument list order: <b>initial value, number of periods, start period, end period, optional section of fields or cells to receive resultant values</b> (can be a list of <b>discontiguous range names or coordinates stored in a parenthesized expression</b> ); if this argument is not specified, the function returns a sequence of resultant values from the start period to the end period. The processing of these values is application dependent.       |

(continued on next page)

# CFE\$\_EXL

Table 6-9 (Cont.): Valid Financial Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_DEP_SLINE     | Sequence of CFE\$_EXL aggregates | <i>Depreciation—straight line</i> calculates the depreciation of an initial value using the straight line method. Argument list order: <b>initial value, number of periods, salvage value, start period, end period, optional section of fields or cells to receive resultant values (can be a list of discontinuous range names or coordinates stored in a parenthesized expression);</b> if this argument is not specified, the function returns a sequence of resultant values from the start period to the end period. The processing of these values is application dependent.       |
| CFE\$K_DEP_SOYD      | Sequence of CFE\$_EXL aggregates | <i>Depreciation—sum of year's digits</i> calculates the depreciation of an initial value using the sum-of-years method. Argument list order: <b>initial value, number of periods, salvage value, start period, end period, optional section of fields or cells to receive resultant values (can be a list of discontinuous range names or coordinates stored in a parenthesized expression);</b> if this argument is not specified, the function returns a sequence of resultant values from the start period to the end period. The processing of these values is application dependent. |

(continued on next page)

Table 6-9 (Cont.): Valid Financial Expression Values for CFE\$ \_EXL \_EXPR \_C

| Expression Indicator | Expression Value                  | Description and Storage Method   |
|----------------------|-----------------------------------|--|
| CFE\$K_DISCOUNT      | Sequence of CFE\$ _EXL aggregates | <i>Discount</i> calculates the discounted values given the principal, an interest per period, and an optional offset into the period. The offset must be a numeric value between 0 (the beginning of the period) and 1 (the end of the period). Argument list order: <b>principal, interest rate per period, offset into period, start period, end period, range of fields or cells to receive resultant values (can be a list of discontinuous range names or coordinates stored in a parenthesized expression)</b> |
| CFE\$K_FV            | Sequence of CFE\$ _EXL aggregates | <i>Future value</i> calculates the future value of a stream of cash flows, given a constant interest rate per period and an offset into the period when the payment occurs (0 = beginning of the period, 1 = end of the period). Argument list order: <b>section of fields or cells containing cash flows for each period, interest rate per period, offset into period</b>  |

(continued on next page)

# CFE\$ \_EXL

Table 6-9 (Cont.): Valid Financial Expression Values for CFE\$ \_EXL \_EXPR \_C

| Expression Indicator | Expression Value                  | Description and Storage Method   |
|----------------------|-----------------------------------|--|
| CFE\$K_FVA           | Sequence of CFE\$ _EXL aggregates | <i>Future value of an annuity</i> calculates the future value of an annuity, given the payment per period, interest rate per period, and number of periods. For example, FVA is used to calculate the future value of a 30-year annuity using monthly payments of \$100 and a 10.9 percent annual interest rate. Argument list order: <b>payment per period, interest rate per period, number of periods</b> |
| CFE\$K_FVPV          | Sequence of CFE\$ _EXL aggregates | <i>Future value of a single sum</i> calculates the future value of a single sum, given the present value, interest rate per period, and the number of periods. For example, FVPV is used to calculate the future value of \$1000 earning 10.8 percent annual interest and compounded monthly for 30 years. Argument list order: <b>present value, interest rate per period, number of periods</b>            |

(continued on next page)

Table 6-9 (Cont.): Valid Financial Expression Values for CFE\$ \_EXL \_EXPR \_C

| Expression Indicator | Expression Value                  | Description and Storage Method   |
|----------------------|-----------------------------------|--|
| CFE\$K_INTEREST      | Sequence of CFE\$ _EXL aggregates | <p><i>Interest payments</i> calculates a stream of interest payments, given the loan amount, interest rate, and number of periods. For example, INTEREST is used to calculate the amount of interest paid for each period (year) on \$1000 at 10 percent annual interest for 5 years. Argument list order: <b>loan amount (original principal), interest rate, number of periods, start period, end period, optional section of fields or cells to receive resultant values (can be a list of discontinuous range names or coordinates stored in a parenthesized expression); if this argument is not specified, the function returns a sequence of resultant values from the start period to the end period. The processing of these values is application dependent.</b></p> |

(continued on next page)

# CFE\$\_EXL

Table 6-9 (Cont.): Valid Financial Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_IRR           | Sequence of CFE\$_EXL aggregates | <p><i>Internal rate of return</i> calculates the internal rate of return, given a stream of cash flows per period and an optional guess at the discount rate. The internal rate of return is the interest rate that gives the series of cash flows a net present value of 0. The cash flow values are interpreted literally—a negative value represents a negative cash flow (that is, initial investment), and a positive value represents a positive cash flow (that is, return on the investment during a period). The discount rate should be specified as a percentage value between 0 and 1. Argument list order: <b>range of fields/ cells containing cash flow values, first guess at the discount rate</b></p> |
| CFE\$K_MIRR          | Sequence of CFE\$_EXL aggregates | <p><i>Modified internal rate of return</i> calculates the modified internal rate of return, given a stream of cash flows, a safe rate of return, and the risk rate. The cash flow values are interpreted literally—a negative value represents a negative cash flow (that is, initial investment), and a positive value represents a positive cash flow (that is, return on the investment during a period). Argument list order: <b>range of fields/ cells containing cash flow values, safe rate of return, risk rate</b></p>   |

(continued on next page)

Table 6-9 (Cont.): Valid Financial Expression Values for CFE\$ \_EXL\_EXPR\_C

| Expression Indicator | Expression Value                  | Description and Storage Method  |
|----------------------|-----------------------------------|---|
| CFE\$K_NPV           | Sequence of CFE\$ _EXL aggregates | <i>Net present value</i> calculates the net present value of a stream of estimated cash flows that are discounted at a constant interest rate per period and uses an offset into the period when the payment occurs (0 = beginning of the period, 1 = end of the period). The net present value is the difference between the future value and the present value. Future values are not reinvested. Argument list order: <b>range of fields/cells containing cash flow values, interest rate per period, offset into period</b> |
| CFE\$K_PAYBACK       | Sequence of CFE\$ _EXL aggregates | <i>Payback</i> calculates the payback period given an investment, a sequence of discount cash flows, and a discount rate per period. Argument list order: <b>initial investment, range of fields/cells containing cash flow values, discount rate per period</b>  |
| CFE\$K_PERFV         | Sequence of CFE\$ _EXL aggregates | <i>Periods to achieve future value</i> calculates the number of periods to achieve a future value, given the payment per period and the interest rate. For example, PERFV is used to calculate the number of monthly payments of \$300.00, at 5% interest, required to earn \$10,000.00. Argument list order: <b>future value, payment per period, interest rate</b>  |

(continued on next page)

## CFE\$\_EXL

Table 6–9 (Cont.): Valid Financial Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_PERPMT        | Sequence of CFE\$_EXL aggregates | <i>Number of periods to achieve future value</i> calculates the number of periods required to achieve a given future value of an annuity, given the present value and the interest rate per period. For example, an account earns 9 percent annual interest compounded monthly, and the balance is \$1742.78. PERPMT is used to calculate how long, in months, it takes to accumulate \$3500. Argument list order: <b>future value of an annuity, present value, interest rate per period</b> |
| CFE\$K_PERPV         | Sequence of CFE\$_EXL aggregates | <i>Number of periods given present value</i> calculates the number of periods over which payments will be made given a present value, the amount of each per-period payment, and the interest rate per period. For example, PERPV is used to calculate how long, in months, it takes to pay off a \$12,000 loan at a 17 percent annual interest rate, if the payments are \$600 per month. Argument list order: <b>present value, payment per period, interest rate per period</b>            |

(continued on next page)



Table 6–9 (Cont.): Valid Financial Expression Values for CFE\$ \_EXL \_EXPR \_C

| Expression Indicator | Expression Value                  | Description and Storage Method  |
|----------------------|-----------------------------------|---|
| CFE\$K_PMTFV         | Sequence of CFE\$ _EXL aggregates | <i>Payment per period to achieve future value</i> calculates the payment per period required to achieve a future value, given the interest rate per period and number of periods. For example, PMTFV is used to calculate the value of each monthly payment required to generate \$100,000 in 25 years on an account that earns 11.5 percent annual interest. Argument list order: <b>future value, interest rate per period, number of periods</b> |
| CFE\$K_PMTPV         | Sequence of CFE\$ _EXL aggregates | <i>Payment per period given present value</i> calculates the payment per period given a present value, the interest rate per period, and the number of periods. For example, PMTPV is used to calculate the monthly payments on a \$100,000 loan at a 11.5 percent annual interest rate over 30 years. Argument list order: <b>present value (principal), interest rate per period, number of periods</b>   |

(continued on next page)

# CFE\$\_EXL

Table 6–9 (Cont.): Valid Financial Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$_K_PRINCIPAL    | Sequence of CFE\$_EXL aggregates | <i>Principal</i> calculates a stream of principal values paid given the amount of the loan, the interest rate per period, and the number of periods. Argument list order: <b>loan amount, interest rate per period, number of periods, start period, end period, optional section of fields or cells to receive resultant values or nth period (can be a list of discontinuous range names or coordinates stored in a parenthesized expression); if this argument is not specified, the function returns a sequence of resultant values from the start period to the end period. The processing of these values is application dependent.</b> |
| CFE\$_K_PVA          | Sequence of CFE\$_EXL aggregates | <i>Present value of an annuity</i> calculates the present value of an annuity, based on a constant payment per period, the interest rate per period, and the number of periods in the life of the annuity. Argument list order: <b>payment per period, interest rate per period, number of periods</b>  |

(continued on next page)

Table 6–9 (Cont.): Valid Financial Expression Values for CFE\$ \_EXL \_EXPR \_C

| Expression Indicator | Expression Value                  | Description and Storage Method   |
|----------------------|-----------------------------------|--|
| CFE\$K_PV FV         | Sequence of CFE\$ _EXL aggregates | <i>Present value to achieve future value</i> calculates the present value required to achieve a given future value, given the number of periods and the interest rate per period. For example, PV FV is used to calculate the initial deposit that makes the account that earns 17 percent annual interest, compounded quarterly, worth \$200,000 in 10 years. Argument list order: <b>future value, number of periods, interest rate per period</b> |
| CFE\$K_RATE          | Sequence of CFE\$ _EXL aggregates | <i>Interest rate</i> calculates the interest rate required to achieve a given future value, given a known present value and the number of periods. Argument list order: <b>future value, present value, number of periods</b>  |

Valid identification expression values for this item are shown in Table 6–10.

Table 6–10: Valid Identification Expression Values for CFE\$ \_EXL \_EXPR \_C

| Expression Indicator | Expression Value                  | Description and Storage Method   |
|----------------------|-----------------------------------|--|
| CFE\$K_ISBLANK       | Sequence of CFE\$ _EXL aggregates | <i>Is blank</i> returns TRUE if the value is blank and FALSE otherwise. Argument list order: <b>value1</b> |
| CFE\$K_ISDATE        | Sequence of CFE\$ _EXL aggregates | <i>Is date</i> returns TRUE if the value is a date and FALSE otherwise. Argument list order: <b>value1</b> |

(continued on next page)

# CFE\$ \_EXL

**Table 6–10 (Cont.): Valid Identification Expression Values for CFE\$ \_EXL\_EXPR\_C**

| <b>Expression Indicator</b> | <b>Expression Value</b>           | <b>Description and Storage Method</b>   |
|-----------------------------|-----------------------------------|---|
| CFE\$K_ISERROR              | Sequence of CFE\$ _EXL aggregates | <i>Is error</i> returns TRUE if the value is an error and FALSE otherwise. Argument list order: <b>value1</b>   |
| CFE\$K_ISNOT_AVAIL          | Sequence of CFE\$ _EXL aggregates | <i>Is not available</i> returns TRUE if the value is equal to the constant NOT_AVAIL and FALSE otherwise. The specific value associated with this constant is application dependent. Some spreadsheet applications use this constant as the result of calculations or as a function, NA(). Argument list order: <b>value1</b> |
| CFE\$K_ISNOT_CALC           | Sequence of CFE\$K_EXL aggregates | <i>Is not calculable</i> returns TRUE if the value is equal to the constant NOT_CALC and FALSE otherwise. The specific value associated with this constant is application dependent. Some spreadsheet applications use this constant as the result of calculations or as a function, NC(). Argument list order: <b>value1</b> |
| CFE\$K_ISNULL               | Sequence of CFE\$ _EXL aggregates | <i>Is null</i> returns TRUE if the value is missing and FALSE otherwise. Argument list order: <b>value1</b>   |
| CFE\$K_ISNUMBER             | Sequence of CFE\$ _EXL aggregates | <i>Is number</i> returns TRUE if the value is a number and FALSE otherwise. Argument list order: <b>value1</b>  |
| CFE\$K_ISREF                | Sequence of CFE\$ _EXL aggregates | <i>Is referenced</i> returns TRUE if the value is referenced by another value's formula and FALSE otherwise. Argument list order: <b>value1</b>   |

(continued on next page)

Table 6–10 (Cont.): Valid Identification Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$_K_ISSTRING     | Sequence of CFE\$_EXL aggregates | <i>Is string</i> returns TRUE if the value is a string and FALSE otherwise. Argument list order: <b>value1</b> |

Valid literal values for this item are shown in Table 6–11.

Table 6–11: Valid Literal Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator      | Expression Value              | Description and Storage Method  |
|---------------------------|-------------------------------|---|
| CFE\$_K_LIT_COMPLEX_FLOAT | Handle of CFE\$_CFT aggregate | <i>Complex floating-point</i> defines a sequence of two floating-point constants. The first floating-point constant represents the real portion of the complex number, and the second floating-point constant represents the imaginary portion of the complex number. |
| CFE\$_K_LIT_DATE          | Handle of CFE\$_DAT aggregate | <i>Date</i> defines a date constant.  |
| CFE\$_K_LIT_FALSE         | None                          | <i>False</i> is logical FALSE.  |
| CFE\$_K_LIT_FLOAT         | General floating-point        | <i>Floating-point</i> defines a general floating-point constant.  |
| CFE\$_K_LIT_INTEGER       | Integer                       | <i>Literal integer</i> defines a signed or unsigned integer.  |
| CFE\$_K_LIT_PI            | None                          | <i>PI</i> returns the value of PI (3.14159265359 ...).  |

(continued on next page)

## CFE\$\_EXL

Table 6–11 (Cont.): Valid Literal Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator      | Expression Value              | Description and Storage Method  |
|---------------------------|-------------------------------|---|
| CFE\$K_LIT_SCALED_INTEGER | Scaled integer                | <i>Scaled integer</i> represents a floating-point number using two integer values. The scale value is a signed integer that represents the position of an implicit decimal point. A positive scale value indicates that the decimal point is moved to the left, and a negative scale value indicates that the decimal point is moved to the right. The value of a scaled integer is the integer value times 10 raised to the power of the scale factor. |
| CFE\$K_LIT_TEXT           | Handle of CFE\$_TXC aggregate | <i>Text string</i> defines a fixed-length text string.  |
| CFE\$K_LIT_TRUE           | None                          | <i>True</i> is logical TRUE.  |
| CFE\$K_LIT_VTEXT          | Handle of CFE\$_VTX aggregate | <i>Varying length text</i> defines a text string that usually contains implied trailing blank characters.   |

Valid miscellaneous expression values for this item are shown in Table 6–12.

Table 6–12: Valid Miscellaneous Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator   | Expression Value              | Description and Storage Method  |
|------------------------|-------------------------------|---|
| CFE\$K_FIELD_REFERENCE | Handle of CFE\$_FRF aggregate | <i>Field reference</i> refers to a database field.  |
| CFE\$K_PARENTHESESIZED | Handle of CFE\$_PEX aggregate | <i>Parenthesized expression</i> defines a nested expression and its representation delimited by, for example, an open and close parenthesis or the words BEGIN and END. |

(continued on next page)

Table 6–12 (Cont.): Valid Miscellaneous Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator    | Expression Value                 | Description and Storage Method  |
|-------------------------|----------------------------------|---|
| CFE\$K_PRIVATE_FUNCTION | Handle of CFE\$_PFE aggregate    | <i>Private</i> specifies a call to a function not defined elsewhere in CFE. If an application supports a function that does not correspond to a CFE-defined function, it must use this value.                       |
| CFE\$K_RANDOM_U         | None                             | <i>Random number</i> calculates a random number uniformly distributed between 0 and 1.  |
| CFE\$K_SIGN             | Sequence of CFE\$_EXL aggregates | <i>Sign</i> indicates the sign of a value expression. If the value is positive, it returns 1. If the value is equal to 0, it returns 0. If the value is negative, it returns -1. Argument list order: <b>source</b> |

Valid series expression values for this item are shown in Table 6–13.

Table 6–13: Valid Series Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_INTEGRATE     | Sequence of CFE\$_EXL aggregates | <i>Integrate</i> is reserved for future use and is not to be used by conforming applications.     |
| CFE\$K_LOGEST        | Sequence of CFE\$_EXL aggregates | <i>Logest</i> is reserved for future use and is not to be used by conforming applications.        |
| CFE\$K_LSQR          | Sequence of CFE\$_EXL aggregates | <i>Least squares</i> is reserved for future use and is not to be used by conforming applications. |
| CFE\$K_SIGMA         | Sequence of CFE\$_EXL aggregates | <i>Sigma</i> is reserved for future use and is not to be used by conforming applications.         |

(continued on next page)

## CFE\$\_EXL

Table 6–13 (Cont.): Valid Series Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method  |
|----------------------|----------------------------------|---|
| CFE\$K_TREND         | Sequence of CFE\$_EXL aggregates | <i>Trend</i> is reserved for future use and is not to be used by conforming applications. |

Valid statistical expression values for this item are shown in Table 6–14.

Table 6–14: Valid Statistical Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value              | Description and Storage Method  |
|----------------------|-------------------------------|---|
| CFE\$K_AVG           | Handle of CFE\$_SLL aggregate | <i>Average</i> averages the values of all expressions in the selection argument that match the <b>criteria</b> argument. If <b>criteria</b> is not specified, average defaults to TRUE, meaning that all expressions in the CFE\$_EXL aggregate are included in the average. Argument list order: <b>criteria, selection</b>                      |
| CFE\$K_COUNT         | Handle of CFE\$_SLL aggregate | <i>Count</i> counts the number of expressions in the selection argument that match the <b>criteria</b> argument. If <b>criteria</b> is not specified, count defaults to TRUE, meaning that all expressions in the CFE\$_EXL aggregate are counted. Argument list order: <b>criteria, selection</b>  |
| CFE\$K_MAX           | Handle of CFE\$_SLL aggregate | <i>Maximum</i> calculates the maximum value of all expressions in the selection argument that match the <b>criteria</b> argument. If <b>criteria</b> is not specified, maximum defaults to TRUE, meaning that all expressions in the CFE\$_EXL aggregate are used to calculate the maximum value. Argument list order: <b>criteria, selection</b> |

(continued on next page)



Table 6–14 (Cont.): Valid Statistical Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value              | Description and Storage Method   |
|----------------------|-------------------------------|--|
| CFE\$K_MIN           | Handle of CFE\$_SLL aggregate | <i>Minimum</i> calculates the minimum value of all expressions in the selection argument that match the <b>criteria</b> argument. If <b>criteria</b> is not specified, minimum defaults to TRUE, meaning that all expressions in the CFE\$_EXL aggregate are used to calculate the minimum value. Argument list order: <b>criteria, selection</b>  |
| CFE\$K_STDEV         | Handle of CFE\$_SLL aggregate | <i>Standard deviation</i> calculates the standard deviation of all expressions in the <b>selection</b> argument that match the <b>criteria</b> argument. If <b>criteria</b> is not specified, standard deviation defaults to TRUE, meaning that all expressions in the CFE\$_EXL aggregate are used to calculate the standard deviation. Argument list order: <b>criteria, selection</b> |
| CFE\$K_SUM           | Handle of CFE\$_SLL aggregate | <i>Summation</i> sums the values of all expressions in the <b>selection</b> argument that match the <b>criteria</b> argument. If <b>criteria</b> is not specified, sum defaults to TRUE, meaning that all expressions in the CFE\$_EXL aggregate are included in the sum. Argument list order: <b>criteria, selection</b>  |

(continued on next page)

# CFE\$\_EXL

**Table 6–14 (Cont.): Valid Statistical Expression Values for CFE\$\_EXL\_EXPR\_C**

| <b>Expression Indicator</b> | <b>Expression Value</b>       | <b>Description and Storage Method</b>  |
|-----------------------------|-------------------------------|--|
| CFE\$K_VAR                  | Handle of CFE\$_SLL aggregate | <i>Variance</i> calculates the variance of all expressions in the <b>selection</b> argument that match the <b>criteria</b> argument. If <b>criteria</b> is not specified, variance defaults to TRUE, meaning that all expressions in the CFE\$_EXL aggregate are used to calculate the variance. Argument list order: <b>criteria, selection</b> |

Valid string expression values for this item are shown in Table 6–15.

**Table 6–15: Valid String Expression Values for CFE\$\_EXL\_EXPR\_C**

| <b>Expression Indicator</b> | <b>Expression Value</b>          | <b>Description and Storage Method</b>  |
|-----------------------------|----------------------------------|--|
| CFE\$K_STR_CHAR             | Sequence of CFE\$_EXL aggregates | <i>String character</i> returns the character corresponding to the designated character code. The character set is application dependent. Argument list order: <b>character-code</b>                                   |
| CFE\$K_STR_CODE             | Sequence of CFE\$_EXL aggregates | <i>String character code</i> returns the character code for the first character of <b>string1</b> . The character set is application dependent. Argument list order: <b>string1</b>                                    |
| CFE\$K_STR_CONCAT           | Sequence of CFE\$_EXL aggregates | <i>String concatenate</i> concatenates two strings by appending the characters of the second string ( <b>string2</b> ) to the end of the first string ( <b>string1</b> ). Argument list order: <b>string1, string2</b> |

(continued on next page)

Table 6–15 (Cont.): Valid String Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$K_CONTAINS      | Sequence of CFE\$_EXL aggregates | <i>Contains substring</i> returns TRUE if the first expression contains the second expression as a substring and returns FALSE otherwise. Argument list order: <b>string, substring</b>  |
| CFE\$K_STARTS        | Sequence of CFE\$_EXL aggregates | <i>String starts with</i> returns TRUE if <b>string1</b> starts with <b>string2</b> and returns FALSE otherwise. Argument list order: <b>string1, string2</b>  |
| CFE\$K_STR_EXTRACT   | Sequence of CFE\$_EXL aggregates | <i>String extract</i> extracts characters from <b>string1</b> starting with the character at the starting position up to and including the character at the ending position. The first character is located at position 1. Argument list order: <b>string1, starting position, ending position</b> |
| CFE\$K_STR_FIND      | Sequence of CFE\$_EXL aggregates | <i>String find substring</i> searches a string for a substring pattern and returns the character position where <b>substring</b> was found (or returns 0 if not found). The first character is located at position 1. Argument list order: <b>string1, substring</b>                               |
| CFE\$K_STR_FIXED     | Sequence of CFE\$_EXL aggregates | <i>String fixed</i> rounds the value of <b>expression1</b> to the number of decimal places in <b>expression2</b> and returns the resulting value as a string. Argument list order: <b>expression1, expression2</b>   |

(continued on next page)

## CFE\$\_EXL

Table 6–15 (Cont.): Valid String Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$_STR_FORMAT     | Handle of CFE\$_STF              | <i>String format</i> formats a source expression using an edit string pattern. Argument list order: <b>source, edit-string</b>   |
| CFE\$_K_STR_LEFT     | Sequence of CFE\$_EXL aggregates | <i>Extract substring left</i> extracts a number of characters ( <b>count</b> ) from the leftmost end of a string ( <b>source</b> ). Argument list order: <b>source, count</b>  |
| CFE\$_K_STR_LENGTH   | Sequence of CFE\$_EXL aggregates | <i>String length</i> returns the number of characters in <b>string1</b> . Argument list order: <b>string1</b>  |
| CFE\$_K_STR_LOWER    | Sequence of CFE\$_EXL aggregates | <i>String lowercase</i> changes all characters in <b>string1</b> to lowercase. Argument list order: <b>string1</b>   |
| CFE\$_K_STR_PRETTY   | Handle of CFE\$_STP aggregate    | <i>String pretty</i> is a combination of string formats and applies all flags enabled in the CFE\$_STP_PRETTY_FLAGS item to a string specified by <b>string-expr</b> . Argument list order: <sup>1</sup> <b>string-expr, flags</b> |
| CFE\$_K_STR_PROPER   | Sequence of CFE\$_EXL aggregates | <i>String proper</i> converts the first nonblank character of <b>string1</b> to uppercase. Argument list order: <b>string1</b>   |
| CFE\$_K_STR_REPEAT   | Sequence of CFE\$_EXL aggregates | <i>String repeat</i> repeats a character sequence ( <b>string1</b> ) for a specified number of times ( <b>repeat-count</b> ). Argument list order: <b>string1, repeat-count</b>  |

<sup>1</sup>Only **argument1** is in the expression list; the second argument is the next element in the sequence after the expression list.

(continued on next page)

Table 6–15 (Cont.): Valid String Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$K_STR_REPLACE   | Sequence of CFE\$_EXL aggregates | <i>String replace</i> replaces the characters in the source string, beginning at the starting character position and continuing for the character count, with the string in the replacement string. The first character is located at position 1. Argument list order: <b>source, starting character position, character count, replacement string</b> |
| CFE\$K_STR_REVERSE   | Sequence of CFE\$_EXL aggregates | <i>String reverse</i> returns a string created by reversing <b>string1</b> . Argument list order: <b>string1</b>   |
| CFE\$K_STR_RIGHT     | Sequence of CFE\$_EXL aggregates | <i>Extract substring right</i> extracts a number of characters ( <b>count</b> ) from the rightmost end of a string ( <b>source</b> ). Argument list order: <b>source, count</b>  |
| CFE\$K_STR_TRIM      | Sequence of CFE\$_EXL aggregates | <i>String trim</i> removes leading and trailing white space (blanks and tabs) from a string. Argument list order: <b>string1</b>   |
| CFE\$K_STR_UPPER     | Sequence of CFE\$_EXL aggregates | <i>String uppercase</i> changes all characters in a string to uppercase. Argument list order: <b>string1</b>   |

Valid transcendental expression values for this item are shown in Table 6–16.

Table 6–16: Valid Transcendental Expression Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value                 | Description and Storage Method   |
|----------------------|----------------------------------|--|
| CFE\$K ALOG          | Sequence of CFE\$_EXL aggregates | <i>Antilog</i> calculates 10 to the power of a value, which is the inverse of the LOG10 function. Argument list order: <b>value1</b> |

(continued on next page)

## CFE\$\_EXL

**Table 6–16 (Cont.): Valid Transcendental Expression Values for CFE\$\_EXL\_EXPR\_C**

| <b>Expression Indicator</b> | <b>Expression Value</b>          | <b>Description and Storage Method</b>  |
|-----------------------------|----------------------------------|--|
| CFE\$K_EXPONENT             | Sequence of CFE\$_EXL aggregates | <i>Exponent</i> raises the value <b>e</b> to the power indicated by <b>exponent</b> . Argument list order: <b>exponent</b> |
| CFE\$K_FACTORIAL            | Sequence of CFE\$_EXL aggregates | <i>Factorial</i> calculates the factorial of a expression. Argument list order: <b>value1</b>                              |
| CFE\$K_LOG10                | Sequence of CFE\$_EXL aggregates | <i>Log, base 10</i> calculates the base-10 logarithm of a value. Argument list order: <b>value1</b>                        |
| CFE\$K_LOGN                 | Sequence of CFE\$_EXL aggregates | <i>Log, base e</i> calculates the natural logarithm of a value. Argument list order: <b>value1</b>                         |

Valid trigonometric expression values for this item are shown in Table 6–17.

**Table 6–17: Valid Trigonometric Expression Values for CFE\$\_EXL\_EXPR\_C**

| <b>Expression Indicator</b> | <b>Expression Value</b>          | <b>Description and Storage Method</b>  |
|-----------------------------|----------------------------------|--|
| CFE\$K_ACOS                 | Sequence of CFE\$_EXL aggregates | <i>Arc cosine</i> calculates the angle in degrees whose cosine is the indicated value. Argument list order: <b>cosine-value</b>    |
| CFE\$K_ASIN                 | Sequence of CFE\$_EXL aggregates | <i>Arc sine</i> calculates the angle in degrees whose sine is the indicated value. Argument list order: <b>sine-value</b>          |
| CFE\$K_ATAN                 | Sequence of CFE\$_EXL aggregates | <i>Arc tangent</i> calculates the angle in degrees whose tangent is the indicated value. Argument list order: <b>tangent-value</b> |

(continued on next page)

Table 6–17 (Cont.): Valid Trigonometric Expression Values for CFE\$ \_EXL \_EXPR \_C

| Expression Indicator | Expression Value                  | Description and Storage Method   |
|----------------------|-----------------------------------|--|
| CFE\$K_ATAN2         | Sequence of CFE\$ _EXL aggregates | <i>Arc tangent 2</i> calculates the 4-quadrant arc tangent in degrees by calculating the arc tangent of the first argument divided by the second argument. For example:<br>(ATAN (value1/value2))<br>Argument list order:<br><b>value1, value2</b> |
| CFE\$K_COS           | Sequence of CFE\$ _EXL aggregates | <i>Cosine</i> calculates the cosine of an angle specified in degrees. Argument list order: <b>angle in degrees</b>   |
| CFE\$K_SIN           | Sequence of CFE\$ _EXL aggregates | <i>Sine</i> calculates the sine of an angle specified in degrees. Argument list order: <b>angle in degrees</b>   |
| CFE\$K_TAN           | Sequence of CFE\$ _EXL aggregates | <i>Tangent</i> calculates the tangent of an angle specified in degrees. Argument list order: <b>angle in degrees</b>   |

Valid variable values for this item are shown in Table 6–18.

Table 6–18: Valid Variable Values for CFE\$ \_EXL \_EXPR \_C

| Expression Indicator | Expression Value               | Description and Storage Method  |
|----------------------|--------------------------------|---|
| CFE\$K_CELL_COORD    | Handle of CFE\$ _CCD aggregate | <i>Cell coordinate</i> specifies a particular cell within a spreadsheet table.                            |
| CFE\$K_CELL_RANGE    | Handle of CFE\$ _CLR aggregate | <i>Cell range</i> specifies a range of cells delineated by two cell references.                           |
| CFE\$K_COL_NAME      | String                         | <i>Column name</i> is an identifier that uniquely identifies a column.                                    |
| CFE\$K_COL_NUM       | Integer                        | <i>Column number</i> is used in encoding cell data and in referencing cell coordinates and column ranges. |

(continued on next page)

## CFE\$\_EXL

Table 6–18 (Cont.): Valid Variable Values for CFE\$\_EXL\_EXPR\_C

| Expression Indicator | Expression Value               | Description and Storage Method  |
|----------------------|--------------------------------|---|
| CFE\$K_COL_RANGE     | Handle of CFE\$_COR aggregate  | <i>Column range</i> defines a column range using starting and ending column numbers.  |
| CFE\$K_CURRENT_VALUE | None                           | <i>Current value</i> is valid only within the CFE\$_SLL aggregate, which is used in CFE to define statistical functions. CURRENT_VALUE is specified by the CFE\$_SLL_CRITERIA item to substitute the current value for each expression in the CFE\$_SLL_SELECTION item.   |
| CFE\$K_IDENTIFIER    | Handle of CFE\$_TXC aggregate  | <i>Identifier</i> is a symbol reference used to reference named symbols that are defined in the DTIF\$_TMD aggregate. If this function is used outside DTIF, the symbol's value must be previously defined by the decoding application. For example, if CFE is used to represent a programming language, the symbol could be used to reference a variable name. |
| CFE\$K_NAMED_RANGE   | Array of type character string | <i>Named range</i> identifies a range by name.  |
| CFE\$K_ROW_NUM       | Integer                        | <i>Row number</i> is used in encoding table rows and in referencing cell coordinates and row ranges.  |
| CFE\$K_ROW_RANGE     | Handle of CFE\$_RWR aggregate  | <i>Row range</i> defines a row range using starting and ending row numbers.   |

### CFE\$\_EXL\_EXPR

#### Encoding: variable

An expression item that specifies the actual data value for the expression value type selected in the previous item.



---

## CFE\$ \_EXP—Expression Aggregate

The expression aggregate defines a CFE expression that can be used to compute the values for a column or for a particular cell. The CFE\$ \_EXP aggregate is referenced by the parent aggregate items DTIF\$ \_CAT\_COMPUTED\_BY, DTIF\$ \_CLD\_FORMULA\_CFE, DTIF\$ \_NVL\_VALUE, and ESF\$ \_NVL\_VALUE.

Refer to these corresponding syntax diagrams:

| Syntax     | Location   |
|------------|------------|
| Expression | Figure D-4 |

---

### AGGREGATE FORMAT

| Item Name                | Item Encoding                     |
|--------------------------|-----------------------------------|
| CFE\$ _EXP_MAJOR_VERSION | Integer                           |
| CFE\$ _EXP_MINOR_VERSION | Integer                           |
| CFE\$ _EXP_LIST          | Sequence of CFE\$ _EXL aggregates |

---

### AGGREGATE ITEMS

#### **CFE\$ \_EXP\_MAJOR\_VERSION**

**Encoding: integer**

An optional major version item that specifies the major version number of the CFE encoding. This item can be omitted if it is used within DTIF, in which case it is assumed that the value of this item is the same as the value for the DTIF\$ \_DSC\_ENCODE\_MAJOR\_VERSION item in DTIF.

This is the primary indicator of compatibility between CFE processors and the encoding of the current document. This item is updated if changes are made to the CFE encoding that are not backward compatible.

#### **CFE\$ \_EXP\_MINOR\_VERSION**

**Encoding: integer**

An optional minor version item that specifies the minor version number of the CFE encoding. This item can be omitted if it is used within DTIF, in which case it is assumed that the value of this item is the same as the value for the DTIF\$ \_DSC\_ENCODE\_MINOR\_VERSION item in DTIF.

Updated versions of this item indicate that changes to the standard CFE encoding have been made that are backward compatible. This item is reset to 0 for each new value of the CFE\$ \_EXP\_MAJOR\_VERSION item.

## **CFE\$\_EXP**

### ***CFE\$\_EXP\_LIST***

***Encoding: sequence of CFE\$\_EXL aggregates***

An expression list item that contains a list of expressions. For more information, see the description of the CFE\$\_EXL aggregate.

---

## CFE\$\_FRF—Field Reference Aggregate

The field reference aggregate contains data that refers to a database field defined by the CFE\$\_EXL\_FIELD\_REF item. The CFE\$\_FRF aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| ExpressionList | Figure D-5  |
| FieldRef       | Figure D-12 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding        |
|-------------------------|----------------------|
| CFE\$_FRF_FIELD_CONTEXT | String               |
| CFE\$_FRF_PATH          | Array of type string |

---

### AGGREGATE ITEMS

#### **CFE\$\_FRF\_FIELD\_CONTEXT**

##### **Encoding: string**

An optional field context item that specifies an identifier used to represent the field during a particular operation (such as record selection).

#### **CFE\$\_FRF\_PATH**

##### **Encoding: array of type string**

A field path item that specifies a sequence of Latin1 text strings that identify the field path. Each string in the sequence identifies a portion of the field path, such as table name, field name, and subfield name.

## CFE\$\_NPM

---

### CFE\$\_NPM—Named Parameter Aggregate

The named parameter aggregate contains data that defines the parameters in the CFE\$\_PFE\_PARAMS item. The CFE\$\_NPM aggregate is referenced by the parent aggregate item CFE\$\_PFE\_PARAMS.

Refer to these corresponding syntax diagrams:

| Syntax         | Location   |
|----------------|------------|
| NamedParameter | Figure D-3 |

---

### AGGREGATE FORMAT

| Item Name       | Item Encoding                    |
|-----------------|----------------------------------|
| CFE\$_NPM_NAME  | String                           |
| CFE\$_NPM_VALUE | Sequence of CFE\$_EXL aggregates |

---

### AGGREGATE ITEMS

**CFE\$\_NPM\_NAME**

**Encoding:** *string*

An optional name item that specifies the name identifying the parameter value. If this item is missing, the parameter is treated as positional.

**CFE\$\_NPM\_VALUE**

**Encoding:** *sequence of CFE\$\_EXL aggregates*

A value item that specifies one or more expressions that evaluate to the value of the parameter. For more information, see the description of the CFE\$\_EXL aggregate.

---

## CFE\$\_PEX—Parenthesized Expression Aggregate

The parenthesized expression aggregate contains data that is nested. The CFE\$\_PEX aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax            | Location    |
|-------------------|-------------|
| ExpressionList    | Figure D-5  |
| ParenthesizedExpr | Figure D-11 |

---

### AGGREGATE FORMAT

| Item Name            | Item Encoding                    |
|----------------------|----------------------------------|
| CFE\$_PEX_BEGIN_EXPR | Array of type character string   |
| CFE\$_PEX_VALUE_EXPR | Sequence of CFE\$_EXL aggregates |
| CFE\$_PEX_END_EXPR   | Array of type character string   |

---

### AGGREGATE ITEMS

#### ***CFE\$\_PEX\_BEGIN\_EXPR***

***Encoding: array of type character string***

An optional text string item that specifies the displayed representation of the beginning of the nested expression. Examples are an open parenthesis or the word BEGIN.

#### ***CFE\$\_PEX\_VALUE\_EXPR***

***Encoding: sequence of CFE\$\_EXL aggregates***

An expression item that specifies the nested expression. For more information, see the description of the CFE\$\_EXL aggregate. Any expression is valid in this context.

#### ***CFE\$\_PEX\_END\_EXPR***

***Encoding: array of type character string***

An optional text string item that specifies the displayed representation of the end of the nested expression. Examples are a close parenthesis or the word END.

## CFE\$\_PFE

---

### CFE\$\_PFE—CFE Application Private Aggregate

The private function aggregate defines functions that are restricted either to a particular data processing implementation, or to a set of related implementations that support identical private encodings. The CFE\$\_PFE aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax           | Location   |
|------------------|------------|
| ExpressionList   | Figure D-5 |
| PrivateFuncExpr  | Figure D-1 |
| StorageSystemTag | Figure D-2 |

---

### AGGREGATE FORMAT

| Item Name                | Item Encoding                    |
|--------------------------|----------------------------------|
| CFE\$_PFE_FACILITY       | String                           |
| CFE\$_PFE_NAME           | String                           |
| CFE\$_PFE_REF_LABEL      | String                           |
| CFE\$_PFE_REF_LABEL_TYPE | String with <b>add-info</b>      |
| CFE\$_PFE_RETURN_TYPE    | Longword                         |
| CFE\$_PFE_PARAMS         | Sequence of CFE\$_NPM aggregates |

---

### AGGREGATE ITEMS

#### **CFE\$\_PFE\_FACILITY**

**Encoding:** *string*

A facility name item that specifies the name of the facility that encoded the function. The name must correspond to the encoding application's registered facility name.

#### **CFE\$\_PFE\_NAME**

**Encoding:** *string*

A function name item that specifies the name of the function.

#### **CFE\$\_PFE\_REF\_LABEL**

**Encoding:** *string*

An optional reference label item that specifies the file name containing the function. This is used for externally defined functions. If this item is omitted, the application must know where to find the definition of the function.

**CFE\$\_PFE\_REF\_LABEL\_TYPE****Encoding: string with add-info**

An optional reference storage item that specifies a storage system tag representing a VAX/RMS, ULTRIX, or MS-DOS or OS/2 file name. The following table lists the values for **add-info** and the corresponding string values.

| <b>Add-Info</b>            | <b>String</b>  |
|----------------------------|--|
| CFE\$_K_PRIVATE_LABEL_TYPE | The label is a private label. In this case, the string can be any user-specified string.     |
| CFE\$_K_RMS_LABEL_TYPE     | The label is an RMS file specification. In this case, the string must be "\$RMS".            |
| CFE\$_K_UTX_LABEL_TYPE     | The label is an ULTRIX file specification. In this case, the string must be "\$UTX".         |
| CFE\$_K_MDS_LABEL_TYPE     | The label is an MS-DOS or OS/2 file specification. In this case, the string must be "\$MDS". |

**CFE\$\_PFE\_RETURN\_TYPE****Encoding: longword**

An optional return type item that specifies the type of function in terms of its return value. Valid values for this item are as follows:

|                            |                         |
|----------------------------|-------------------------|
| cfe\$_m_pfe_fncret_numeric | Returns a numeric value |
| cfe\$_m_pfe_fncret_boolean | Returns a Boolean value |
| cfe\$_m_pfe_fncret_date    | Returns a date value    |
| cfe\$_m_pfe_fncret_text    | Returns a string value  |

**CFE\$\_PFE\_PARAMS****Encoding: sequence of CFE\$\_NPM aggregates**

An optional parameters item that specifies a list of one or more parameters to be passed to the function. For more information, see the description of the CFE\$\_NPM aggregate. If the function has no parameters, this item can be omitted. Each parameter is defined by the CFE\$\_NPM aggregate.

## CFE\$\_RWR

---

### CFE\$\_RWR—CFE Row Range Aggregate

The row range aggregate defines a range using starting and ending row numbers. A row range differs from a cell range in that a row range refers to an indeterminate number of cells. The CFE\$\_RWR aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| ExpressionList | Figure D-5  |
| RowRange       | Figure C-40 |

---

### AGGREGATE FORMAT

| Item Name           | Item Encoding |
|---------------------|---------------|
| CFE\$_RWR_ROW_BEGIN | Integer       |
| CFE\$_RWR_ROW_END   | Integer       |

---

### AGGREGATE ITEMS

#### **CFE\$\_RWR\_ROW\_BEGIN**

**Encoding: integer**

A row-begin data item that specifies the starting row number (the first row in the range).

#### **CFE\$\_RWR\_ROW\_END**

**Encoding: integer**

An optional row-end data item that specifies the ending row number (the last row in the range). If this item is omitted, it indicates a single-row range.



---

## CFE\$\_SLL—Selector List Aggregate

The selector list aggregate is used to specify statistical functions in CFE based on a selection of expressions that match certain defined criteria. The CFE\$\_SLL aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location   |
|----------------|------------|
| ExpressionList | Figure D-5 |
| SelectorList   | Figure D-8 |

---

### AGGREGATE FORMAT

| Item Name           | Item Encoding                    |
|---------------------|----------------------------------|
| CFE\$_SLL_CRITERIA  | Sequence of CFE\$_EXL aggregates |
| CFE\$_SLL_SELECTION | Sequence of CFE\$_EXL aggregates |

---

### AGGREGATE ITEMS

#### **CFE\$\_SLL\_CRITERIA**

##### **Encoding: sequence of CFE\$\_EXL aggregates**

An optional criteria item that is used to select expressions in the CFE\$\_SLL\_SELECTION item that match a specific criteria. For more information, see the description of the CFE\$\_EXL aggregate.

For example, criteria can be used to select all cells in a column that are nonblank, or to count the cells in a range that are greater than 5. If the Boolean expression evaluates to TRUE when applied to an expression in the CFE\$\_SLL\_SELECTION item, the expression is included in the statistical operation.

If the CFE\$\_SLL\_CRITERIA item is not specified, it defaults to TRUE, meaning that all expressions in the CFE\$\_SLL\_SELECTION item match the criteria. This is the simplest case and indicates that the operator can be applied to all arguments specified in the selection list.

#### **CFE\$\_SLL\_SELECTION**

##### **Encoding: sequence of CFE\$\_EXL aggregates**

A selection item that specifies a list of one or more expressions to be used as the arguments for the statistical operator. For more information, see the description of the CFE\$\_EXL aggregate.

## CFE\$\_STF

---

### CFE\$\_STF—String Format Aggregate

The string format aggregate formats a source expression using either a predefined DTIF edit string or a user-defined edit string. The CFE\$\_STF aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| ExpressionList | Figure D-5  |
| EditString     | Figure D-10 |

---

### AGGREGATE FORMAT

| Item Name             | Item Encoding                    |
|-----------------------|----------------------------------|
| CFE\$_STF_SOURCE      | Sequence of CFE\$_EXL aggregates |
| CFE\$_STF_EDIT_STRING | Handle of ESF\$_EDS aggregate    |

---

### AGGREGATE ITEMS

**CFE\$\_STF\_SOURCE**

**Encoding:** *sequence of CFE\$\_EXL aggregates*

A value item that specifies a source expression.

**CFE\$\_STF\_EDIT\_STRING**

**Encoding:** *handle of an ESF\$\_EDS aggregate*

An edit string item that specifies an ESF-defined edit string. For more information, see the description of the ESF\$\_EDS aggregate.

---

## CFE\$\_STP—String Pretty Aggregate

The string pretty aggregate specifies a combination of string formats and applies all flags enabled in the CFE\$\_STP\_PRETTY\_FLAGS item to a string specified by the CFE\$\_STP\_STRING\_EXPR item. The CFE\$\_STP aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location   |
|----------------|------------|
| ExpressionList | Figure D-5 |

---

### AGGREGATE FORMAT

| Item Name              | Item Encoding                    |
|------------------------|----------------------------------|
| CFE\$_STP_STRING_EXPR  | Sequence of CFE\$_EXL aggregates |
| CFE\$_STP_PRETTY_FLAGS | Longword                         |

---

### AGGREGATE ITEMS

#### **CFE\$\_STP\_STRING\_EXPR**

**Encoding:** *sequence of CFE\$\_EXL aggregates*

A string item that is formatted using the flag values specified by the CFE\$\_STP\_PRETTY\_FLAGS item. For more information, see the description of the CFE\$\_EXL aggregate.

#### **CFE\$\_STP\_PRETTY\_FLAGS**

**Encoding:** *longword*

A flags item that specifies the values to be applied to the CFE\$\_STP\_STRING\_EXPR item. This item has the following values:

|                         |   |
|-------------------------|---|
| cfe\$m_pretty_collapse  | Removes all spaces and tabs from CFE\$_STP_STRING_EXPR.         |
| cfe\$m_pretty_compress  | Replaces multiple space and tab characters with a single space. |
| cfe\$m_pretty_lowercase | Converts all characters in CFE\$_STP_STRING_EXPR to lowercase.  |
| cfe\$m_pretty_trim      | Removes all leading and trailing spaces and tabs.               |
| cfe\$m_pretty_uncomment | Removes comment delimiter (!) from CFE\$_STP_STRING_EXPR.       |
| cfe\$m_pretty_upcase    | Converts all characters in CFE\$_STP_STRING_EXPR to uppercase.  |

## CFE\$\_TXC

---

### CFE\$\_TXC—Text Choice Aggregate

The text choice aggregate contains a text string that is the value for the CFE\$K\_LIT\_TEXT expression list value. The CFE\$\_TXC aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location   |
|----------------|------------|
| ExpressionList | Figure D-5 |
| Text           | Figure D-6 |

---

### AGGREGATE FORMAT

| Item Name        | Item Encoding |
|------------------|---------------|
| CFE\$_TXC_TEXT_C | Enumeration   |
| CFE\$_TXC_TEXT   | Variable      |

---

### AGGREGATE ITEMS

#### **CFE\$\_TXC\_TEXT\_C**

##### **Encoding: enumeration**

A text type indicator that specifies the type of value chosen from those that are delineated for this text string. This item has the following values:

|                     |  |
|---------------------|--|
| CFE\$K_LATIN1_TEXT  | Indicates Latin1 text. In this case, the CFE\$_TXC_TEXT item is encoded as a string.                           |
| CFE\$K_SIMPLE_TEXT  | Indicates simple text. In this case, the CFE\$_TXC_TEXT item is encoded as a character string.                 |
| CFE\$K_COMPLEX_TEXT | Indicates complex text. In this case, the CFE\$_TXC_TEXT item is encoded as an array of type character string. |

#### **CFE\$\_TXC\_TEXT**

##### **Encoding: variable**

A text type item that specifies the actual data value for the text string value type selected in the previous item.

---

## CFE\$\_VTX—CFE Varying Text Aggregate

The varying text aggregate contains data that is used to store strings with trailing blanks without actually encoding the blank characters. The CFE\$\_VTX aggregate is referenced by the parent aggregate item CFE\$\_EXL\_EXPR\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location   |
|----------------|------------|
| ExpressionList | Figure D-5 |
| VaryingText    | Figure D-7 |

---

### AGGREGATE FORMAT

| Item Name           | Item Encoding    |
|---------------------|------------------|
| CFE\$_VTX_VTEXT_LEN | Integer          |
| CFE\$_VTX_VTEXT_STR | Character string |

---

### AGGREGATE ITEMS

#### **CFE\$\_VTX\_VTEXT\_LEN**

**Encoding: integer**

A varying text-length item that specifies the defined length (number of characters) in the string.

#### **CFE\$\_VTX\_VTEXT\_STR**

**Encoding: character string**

A character string data item that defines the varying text.

The length of this item must not exceed the value of the CFE\$\_VTX\_VTEXT\_LEN item. If CFE\$\_VTX\_VTEXT\_LEN exceeds the value of CFE\$\_VTX\_VTEXT\_STR, the string must be padded with enough trailing blanks to reach the CFE\$\_VTX\_VTEXT\_LEN value.



This chapter provides a description of each ESF-supported aggregate structure.

---

## 7.1 ESF Generic Aggregate Items

In addition to the items defined by each individual aggregate, the CDA Toolkit also supports two “generic” aggregate items that can be specified for every ESF aggregate described in this chapter. Table 7–1 lists and describes these items.

**Table 7–1: ESF Generic Aggregate Items**

| <b>Item Name</b>     | <b>Encoding</b> | <b>Meaning</b>  |
|----------------------|-----------------|---|
| ESF\$_USER_CONTEXT   | Longword        | Specifies additional longword for user                |
| ESF\$_AGGREGATE_TYPE | Word            | Specifies the type of the aggregate; a read-only item |

---

## ESF\$\_DAT

---

### ESF\$\_DAT—ESF Date and Time Aggregate

The ESF date and time aggregate specifies a date/time value that is defined as a sequence of two octet strings. The first octet defines the date and time as a binary value; the second octet string defines an optional time differential. The ESF\$\_DAT aggregate is referenced by the parent aggregate item ESF\$\_NVL\_VALUE.

Refer to these corresponding syntax diagrams:

---

| Syntax   | Location    |
|----------|-------------|
| DateTime | Figure C-29 |

---

### AGGREGATE FORMAT

---

| Item Name             | Item Encoding |
|-----------------------|---------------|
| ESF\$_DAT_DATETIME    | String        |
| ESF\$_DAT_TIME_DIFF_C | Enumeration   |
| ESF\$_DAT_TIME_DIFF   | Variable      |

---

### AGGREGATE ITEMS

#### *ESF\$\_DAT\_DATETIME*

**Encoding:** *string*

A date and time item that is a sequence of octets representing a date/time value. Each octet is interpreted as an unsigned integer value, as shown in the following table.

The following table shows how the two octets that represent the encoding of the DAT\_TIME\_DIFF item for the ESF\$\_K\_PLUS\_DIFF and ESF\$\_K\_NEG\_DIFF cases are encoded.

---

| Octet Number | Date/Time Element  |
|--------------|--|
| 0            | Century digits in the range of values from 0 to 99; for example, 19 in the year 1967 |
| 1            | Year digits in the range of values from 0 to 99; for example, 67 in the year 1967    |
| 2            | Month in the range of values from 1 to 12  |
| 3            | Day in the range of values from 1 to 31  |
| 4            | Hour in the range of values from 0 to 23   |
| 5            | Minute in the range of values from 0 to 59   |
| 6            | Second in the range of values from 0 to 59   |
| 7            | Hundredths of seconds in the range of values from 0 to 99                            |

---



**ESF\$ \_DAT\_TIME\_DIFF\_C**

**Encoding: enumeration**

An optional time difference indicator that specifies the type of time differential value chosen from those that are delineated for the time differential item. Valid values for this item are as follows:

|                  |   |
|------------------|---|
| ESF\$K_UTC_TIME  | A value that represents Coordinated Universal Time (UTC), or Greenwich Mean Time. This is equivalent to a time differential of 0 hours, 0 seconds. In this case, the ESF\$ _DAT_TIME_DIFF item is left blank. |
| ESF\$K_PLUS_DIFF | A string that is a positive time differential that is represented by a sequence of two octets, as shown in the following table. In this case, the ESF\$ _DAT_TIME_DIFF item is encoded as a string.           |
| ESF\$K_NEG_DIFF  | A string that is a negative time differential is represented by a sequence of two octets, as shown in the following table. In this case, the ESF\$ _DAT_TIME_DIFF item is encoded as a string.                |

The following table shows how the two octets that represent the encoding of the DAT\_TIME\_DIFF item for the ESF\$K\_PLUS\_DIFF and ESF\$K\_NEG\_DIFF cases are encoded.

| Octet Number | Date/Time Element  |
|--------------|--|
| 0            | Hours in the range of values from 0 to 13 for ESF\$K_PLUS_DIFF and in the range of values from 0 to 12 for ESF\$K_NEG_DIFF |
| 1            | Minutes in the range of values from 0 to 59  |

**ESF\$ \_DAT\_TIME\_DIFF**

**Encoding: variable**

A time difference item that contains the actual time differential for the time differential type selected in the previous item.

If the ESF\$ \_DAT\_TIME\_DIFF item is present, the ESF\$ \_DAT\_DATETIME item represents Coordinate Universal Time, and the value chosen for ESF\$ \_DAT\_TIME\_DIFF represents the local time differential.

If the ESF\$ \_DAT\_TIME\_DIFF item is not present, the value specified for the ESF\$ \_DAT\_DATETIME item represents local time.

## ESF\$\_EDS

---

### ESF\$ \_EDS—Edit String Aggregate

The edit string aggregate defines a single edit string pattern. The pattern is a sequence of one or more edit string tags. A tag may be specified singly or as a repeating sequence. The ESF\$\_EDS aggregate is referenced by the parent aggregate items DTIF\$\_FMI\_NUMEDS\_EDITSTR, DTIF\$\_FMI\_TXTEDS\_EDITSTR, DTIF\$\_FMI\_DATEDS\_EDITSTR, DTIF\$\_NES\_DEFN, and CFE\$\_STF\_EDIT\_STRING.

Refer to these corresponding syntax diagrams:

| Syntax      | Location   |
|-------------|------------|
| EditString  | Figure E-1 |
| EditStrBuff | Figure E-2 |
| Single      | Figure E-3 |
| Repeat      | Figure E-4 |

---

### AGGREGATE FORMAT

| Item Name               | Item Encoding             |
|-------------------------|---------------------------|
| ESF\$_EDS_MAJOR_VERSION | Integer                   |
| ESF\$_EDS_MINOR_VERSION | Integer                   |
| ESF\$_EDS_EDIT_STRING_C | Array of type enumeration |
| ESF\$_EDS_EDIT_STRING   | Array of type variable    |

---

### AGGREGATE ITEMS

#### ***ESF\$\_EDS\_MAJOR\_VERSION***

***Encoding: integer***

An optional major version item that specifies the major version number of the ESF encoding. This item can be omitted if it is used within DTIF, in which case it is assumed that the value of this item is the same as the value for the DTIF\$\_DSC\_ENCODE\_MAJOR\_VERSION item.

This is the primary indicator of compatibility between ESF processors and the encoding of the current document. This item must be updated if changes are made to the ESF encoding that are not backward compatible.

#### ***ESF\$\_EDS\_MINOR\_VERSION***

***Encoding: integer***

An optional minor version item that specifies the revision version number of the ESF encoding. This item can be omitted if it is used within DTIF, in which case it is assumed that the value of this item is the same as the value for the DTIF\$\_DSC\_ENCODE\_MINOR\_VERSION item.

Updated values for this item indicate that changes to the standard ESF encoding have been made that are backward compatible. This item is reset to 0 for each new value of ESF\$\_EDS\_MAJOR\_VERSION.

**ESF\$\_EDS\_EDIT\_STRING\_C**

**Encoding: array of enumeration**

An edit string indicator that specifies the type of value chosen from those that are delineated for the edit string item. Valid values are shown in Table 7-2.

**NOTE**

In Table 7-2, None means that the edit string must still be stored as an integer in the ESF\$\_EDS\_EDIT\_STRING item. The integer value is not output to the DDIS stream, but serves as a placeholder in the array.

**Table 7-2: Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C**

| <b>Edit String Indicator</b> | <b>Edit String Value</b> | <b>Description</b>  |
|------------------------------|--------------------------|---|
| ESF\$K_EDS_ALPHABETIC        | None                     | Causes the next character in the data value content to be moved to the edited string. This character is expected to be alphabetic. If the next character is not alphabetic, the action to be taken is the responsibility of the application.  |
| ESF\$K_EDS_AM_PM             | None                     | Causes one character from the AM or PM indicator string to be moved to the edited string. The use of two consecutive occurrences of this tag is recommended. This tag is often used with the HOUR_12 tag and is placed at the end of the edit string.<br><br>This tag should be used only if the data value contains a date/time value. The application must select the AM or PM value according to the data value.<br><br>If this tag is used within the context of a DTIF table, the DTIF\$_LPT_ITEMS text strings assigned to represent time values must be used as the AM and PM indicator strings. For more information on the DTIF\$_LPT_ITEMS item, see the description of the DTIF\$_LPT aggregate. |

(continued on next page)

## ESF\$\_EDS

Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator   | Edit String Value                | Description  |
|-------------------------|----------------------------------|--|
| ESF\$K_EDS_ANY_CASE     | None                             | Is used to indicate that any alphabetic characters displayed in the edit string, from the current character position on, are to be displayed as shown in the original edit string. The ANY_CASE tag is typically used in conjunction with the LOWERCASE and UPPERCASE tags. The ANY_CASE tag does not reserve any space in the edited string; it is used to modify the behavior of the remaining edit string tags.   |
| ESF\$K_EDS_ANY_CHAR     | None                             | Causes the next octet (8 bits) in the data value's content to be moved to the edited string. This tag may be used with any data type.  |
| ESF\$K_EDS_APPL_PRIVATE | Sequence of ESF\$_NVL aggregates | Defines an application-specific string.  |
| ESF\$K_EDS_BINARY_DIGIT | None                             | Causes one binary digit to be moved from the data value to the edited string.  |
| ESF\$K_EDS_CURRENCY     | None                             | Has two behaviors, depending on the number of times it appears in an edit string. If the tag is specified only once, it causes the currency symbol to be inserted in the next character position of the edited string. If two or more consecutive CURRENCY tags occur at the beginning of the edit string, any leading zeros or digit separators that the tag matches are replaced with blanks. One currency symbol is displayed to the immediate left of the leftmost digit of the edited value, as determined by the remaining characters in the edit string. The latter case is used to display the currency symbol as a floating character; its position in the displayed string varies according to the number of digits in the data value. |

(continued on next page)

Table 7-2 (Cont.): Valid Values for ESF\$ \_EDS\_EDIT\_STRING\_C

| Edit String Indicator   | Edit String Value              | Description  |
|-------------------------|--------------------------------|--|
| ESF\$K_EDS_CURRENCY_LIT | Handle of ESF\$ _TXS aggregate | <p>In the context of a DTIF table, the DTIF\$ _LPT_ITEMS text string assigned to represent the currency symbol must be used as the currency symbol character string. The DTIF\$ _LPT_ITEMS text strings assigned to represent the currency radix and currency digit separator symbols must be used for the radix-point and digit-sep characters, respectively. For more information on the DTIF\$ _LPT_ITEMS item, see the description of the DTIF\$ _LPT aggregate.</p> <p>Has a behavior identical to that of the CURRENCY tag, except that the characters to be inserted as the currency symbol are explicitly included in the CURRENCY_LIT tag.</p>  |
| ESF\$K_EDS_DAY_NUMBER   | None                           | <p>The CURRENCY_LIT tag has two behaviors, depending on the number of times it appears in an edit string. If the tag is specified once, it causes the currency symbol to be inserted in the next character position of the edited string. If two or more consecutive CURRENCY_LIT tags occur at the beginning of the edit string, any leading zeros or digit separators that the tag matches are replaced with blanks. One currency symbol is displayed to the immediate left of the leftmost digit of the edited value, as determined by the remaining characters in the edit string. The latter case is used to display the currency symbol as a floating character; its position in the displayed string varies according to the number of digits in the data value.</p> <p>Causes the next digit of the day of the month to be placed in the edited string. The use of two consecutive occurrences of this tag is recommended. This tag can be used only for data values representing date/time values.</p> <p>If the day number is a single digit value and if there is more than one DAY_NUMBER tag in the edit string, the DAY_NUMBER may be filled with leading zeros.</p> |

(continued on next page)

# ESF\$\_EDS

Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator    | Edit String Value             | Description  |
|--------------------------|-------------------------------|--|
| ESF\$K_EDS_DECIMAL_DIGIT | None                          | <p>Causes the next decimal digit from the data value's content to be moved to the edited string. This tag can appear before or after a RADIX_POINT tag. If it appears to the left of the RADIX_POINT tag, it must correspond to the integer portion of the data value. If it appears to the right of the RADIX_POINT tag, it must correspond to the fraction-second portion of the data value.</p> <p>The DECIMAL_DIGIT tag must not be used in the same edit string with the HEX_DIGIT or OCTAL_DIGIT tags, or with tags used for nonnumeric values (for example, date/time or text).</p> <p>If the number of DECIMAL_DIGIT tags exceeds the number of digits in the data value, the remaining characters should be displayed as leading zeros, unless the edit string includes a FLOAT_BLANK_SUPR tag or a ZERO_REPLACE tag.</p>                             |
| ESF\$K_EDS_DIGIT_SEP     | None                          | <p>Has two defined behaviors, depending on the data value type. If the data value is alphabetic, a digit separator character is inserted in the next character position in the edited string. If the data value is numeric, and if all digits currently in the edited string are suppressed zeros, this tag causes another suppressed zero to be moved to the edited string. If the edited string contains digits, a digit separator character is inserted in the next character position in the edited string.</p> <p>If the edit string is used in the context of a DTIF table, the DTIF\$_LPT_ITEMS text string assigned to represent the currency digit separator or numeric digit separator symbol must be used as the digit separator character. For more information on the DTIF\$_LPT_ITEMS item, see the description of the DTIF\$_LPT aggregate.</p> |
| ESF\$K_EDS_DIGIT_SEP_LIT | Handle of ESF\$_TXS aggregate | <p>Has a behavior identical to that of the DIGIT_SEP tag, except that the characters to be inserted as the digit separator are explicitly included in the DIGIT_SEP_LIT tag.</p>   |

(continued on next page)

Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator    | Edit String Value | Description   |
|--------------------------|-------------------|---|
| ESF\$K_EDS_ENCODED_MINUS | None              | <p>The DIGIT_SEP_LIT tag has two defined behaviors, depending on the data value type. If the data value is alphabetic, the digit separator character is inserted in the next character position in the edited string. If the data value is numeric, and if all digits currently in the edited string are suppressed zeros, this tag causes another suppressed zero to be moved to the edited string. If the edited string contains digits, the digit separator character is inserted in the next character position in the edited string.</p> <p>Is used to represent a negative value with an encoded (overpunched) minus sign. If the data value is negative, this tag causes the next digit in the data value to be overpunched with a minus sign and to be placed in the edited string. The overpunched character is the character resulting from superimposing (overpunching) the minus sign (-) character directly over the digit character. If the data value is nonnegative, this tag functions like the DECIMAL_DIGIT, HEX_DIGIT, or OCTAL_DIGIT tag, depending on the other tags in the edit string.</p> <p>This tag should be specified at either the beginning or end of the edit string. It must not be used in the same edit string as any other tag designating a sign (for example, MINUS). The data value should be numeric.</p> |
| ESF\$K_EDS_ENCODED_PLUS  | None              | <p>Is used to represent a positive value with an overpunched plus sign. If the data value is positive, this tag causes the next digit in the data value to be overpunched with a plus sign and to be placed in the edited string. The overpunched character is the character resulting from superimposing (overpunching) the plus sign (+) character directly over the digit character. If the data value is negative, this tag functions like the DECIMAL_DIGIT, HEX_DIGIT, or OCTAL_DIGIT tag, depending on the other tags in the edit string.</p>  |

(continued on next page)

# ESF\$\_EDS

Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator       | Edit String Value | Description  |
|-----------------------------|-------------------|--|
| ESF\$K_EDS_ENCODED_SIGN     | None              | <p>This tag can be specified at either the beginning or end of the edit string. It must not be used in the same edit string as any other tag designating a sign (for example, MINUS) when used for numeric values.</p> <p>Is used to represent a signed value with an overpunched sign (minus if negative and plus if positive). If the data value is negative, this tag causes the next digit in the data value to be overpunched with a minus sign and to be placed in the edited string. If the data value is positive, this tag causes the next digit in the data value to be overpunched with a plus sign and to be placed in the edited string. If the value is 0, this tag functions like the DECIMAL_DIGIT, HEX_DIGIT, or OCTAL_DIGIT tag, depending on the other tags in the edit string.</p>   |
| ESF\$K_EDS_EXPONENT         | None              | <p>This tag can be specified at either the beginning or end of the edit string. It must not be used in the same edit string as any other tag designating a sign (for example, MINUS) when used for numeric values.</p> <p>Is used to divide the edit string into two parts for floating-point or scientific notation. The edit string characters preceding this tag must correspond to the mantissa edit string, and the characters following this tag must correspond to the exponent edit string. At the point in the edit string where this tag occurs, the character E is inserted in the edited string. The EXPONENT tag can appear only once in an edit string pattern.</p> <p>The edit-string pattern for the mantissa can take precedence over the edit string pattern for the exponent of a data value. The edited value can be displayed using the indicated number of digits in the mantissa portion of the edit string, and the exponent data value can be adjusted accordingly.</p> |
| ESF\$K_EDS_FLOAT_BLANK_SUPR | None              | <p>Indicates that the preceding tag is to be repeated one time and provides for variable-length data values. If the edited value is a blank, the blank is suppressed and not placed in the edited string.</p>  |

(continued on next page)



Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator      | Edit String Value | Description   |
|----------------------------|-------------------|---|
| ESF\$K_EDS_FRACTION_SECOND | None              | Is used to display the fractional digits of a second in a time value. Two instances of this tag denote hundredths of a second. For each instance of this tag, the next digit of the value's fractional second is moved to the edited string.  |
| ESF\$K_EDS_HEX_DIGIT       | None              | Causes one hexadecimal digit from the data value to be moved to the edited string. This tag must not be used in the same edit string as the DECIMAL_DIGIT or OCTAL_DIGIT tags.  |
| ESF\$K_EDS_HOUR_12         | None              | Causes one digit of the hour, in 12-hour mode, to be moved to the edited string. Digital recommends that two consecutive instances of this tag be used. Digital also recommends that two consecutive instances of the AM_PM tag be included in the edit string.<br><br>The HOUR_12 tag must not be used in the same edit string as the HOUR_24 tag. The HOUR_12 tag must be used only for time data values. |
| ESF\$K_EDS_HOUR_24         | None              | Causes one digit of the hour, in 24-hour mode, to be moved to the edited string. Digital also recommends that two consecutive instances of this tag be used.<br><br>The HOUR_24 tag must not be used in the same edit string as the HOUR_12 or AM_PM tags. The HOUR_24 tag must be used only for time data values.  |
| ESF\$K_EDS_JULIAN_DIGIT    | None              | Causes one digit of the Julian date to be moved to the edited string. It is recommended that three consecutive instances of this tag be used. The Julian date is the day of the year corresponding to a date (for example, January 1 is equal to 1).  |
| ESF\$K_EDS_LOGICAL_CHAR    | None              | Is used to display logical (TRUE and FALSE) data values as the text string TRUE or FALSE. Each instance of this tag causes one character of the text string to be moved to the edited string. The decoding application determines if a data value evaluates to TRUE or FALSE.   |

(continued on next page)

## ESF\$\_EDS

Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator | Edit String Value | Description  |
|-----------------------|-------------------|--|
| ESF\$K_EDS_LONG_TEXT  | None              | Is used to indicate a long text string. Each instance of this tag reserves a character position in the edited string. The output edited string is printed in segments of length determined by the number of LONG_TEXT tags. For example, if five instances of the LONG_TEXT tag are used, and the data value is a text string of 15 characters, three lines of text are displayed, with each line containing five characters.  |
| ESF\$K_EDS_LOWERCASE  | None              | Is used to indicate that any alphabetic characters displayed in the edit string, from the current character position on, are to be displayed as lowercase characters. This tag does not reserve any space in the edited string; it is used to modify the behavior of the remaining edit string tags.   |
| ESF\$K_EDS_MINUS      | None              | <p>Has two behaviors, depending on the number of times it appears in an edit string. If the tag is specified only once, and the data value is negative, the minus sign (-) is inserted in the next character position of the edited string. If the tag is specified only once, and the data value is nonnegative, a blank is inserted in the next character position of the edited string.</p> <p>If two or more consecutive MINUS tags occur at the beginning of the edit string, any leading zeros or digit separators that the tag matches are replaced with blanks. If the data value is negative, a minus sign is displayed to the immediate left of the leftmost character position, as determined by the rest of the edit string. If the data value is positive, a blank is displayed to the left of the leftmost character position.</p> <p>If the data value is nonnumeric, each instance of this tag is treated as a single occurrence of the tag: a minus sign is moved to the edited string for each occurrence of this tag. For a nonnumeric value, a MINUS tag is equivalent to a STR_LITERAL tag with the value "-". The MINUS tag must not appear in the same edit string with another tag designating a sign (for example, PLUS, ENCODED_MINUS) when used for numeric values.</p> |

(continued on next page)

Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator    | Edit String Value             | Description   |
|--------------------------|-------------------------------|---|
| ESF\$K_EDS_MINUS_LITERAL | Handle of ESF\$_TXS aggregate | <p>Has two behaviors, depending on the number of times it appears in an edit string. If the tag is specified only once, and the data value is negative, the literal (text string) is inserted in the next character position of the edited string. If the tag is specified only once, and the data value is nonnegative, as many blanks as characters in the text literal are moved to the edited string (thereby preserving alignment for both positive and negative values).</p> <p>If two or more of these tags occur consecutively before the numeric value portion of the edit string, the minus literal text floats to the beginning of the numeric data value in the edited string. Any leading zeros or digit separators that the tag matches are replaced with blanks. If the data value is negative, the text literal is moved to the next character position of the edited string. If the data value is positive, as many blanks as there are characters in the text literal are moved to the edited string.</p> <p>If the data value is nonnumeric, each instance of this tag is treated as a single occurrence of the tag; the text literal is moved to the edited string for each occurrence of this tag. For nonnumeric values, a MINUS_LITERAL tag is equivalent to a STR_LITERAL tag with the text literal value.</p> <p>The MINUS_LITERAL tag must not be used in the same edit string as any other tag designating a sign (for example, MINUS, PLUS, or PLUS_LITERAL).</p> |
| ESF\$K_EDS_MINUTE        | None                          | <p>Indicates that one digit from the minutes portion of a time value is to be moved to the edited string. The use of two consecutive instances of this tag is recommended. This tag must be used only in an edit string for time values.</p>  |

(continued on next page)

## ESF\$\_EDS

Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator   | Edit String Value | Description   |
|-------------------------|-------------------|---|
| ESF\$K_EDS_MISSING_SEP  | None              | Is used to separate an edit string into two parts: the first part is an edit string to be used if a data value is present, and the second part is an edit string to be used if the data value is missing. The first part consists of the edit string tags preceding the MISSING_SEP tag; the second part consists of the tags following the MISSING_SEP tag.  |
| ESF\$K_EDS_MONTH_NAME   | None              | <p>Causes the next character of the month name to be moved to the edited string. This tag must be used only for date/time data values.</p> <p>The first occurrence of this tag corresponds to the first character in the month name. The number of MONTH_NAME tags in the edit string determines the number of significant characters in the month name that are displayed. If there are more instances of the MONTH_NAME tag than characters in the month name, a blank is moved to the edited string.</p> <p>If the edit string is used within the context of a DTIF table, the DTIF\$_LPT_ITEMS text strings corresponding to the months of the year must be used to display the appropriate text for each month name. For more information on the DTIF\$_LPT_ITEMS item, see the description of the DTIF\$_LPT aggregate.</p> |
| ESF\$K_EDS_MONTH_NUMBER | None              | Causes the next digit of the month number to be moved to the edited string. Two consecutive instances of this tag are recommended. This tag must be used only for date/time data values.  |
| ESF\$K_EDS_OCTAL_DIGIT  | None              | Causes the next octal digit from the data value to be moved to the edited string. This tag must not be used in the same edit string as the DECIMAL_DIGIT or HEX_DIGIT tags.   |

(continued on next page)

Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator   | Edit String Value             | Description  |
|-------------------------|-------------------------------|--|
| ESF\$K_EDS_PLUS         | None                          | <p>Has two behaviors, depending on the number of times it appears in an edit string. If the tag is specified only once, and the data value is positive, the plus sign (+) is inserted in the next character position of the edited string. If the tag is specified only once, and the data value is negative, a blank is inserted in the next character position of the edited string.</p> <p>If two or more consecutive PLUS tags occur at the beginning of the edit string, any leading zeros that the tag matches are replaced with blanks. If the data value is positive, a plus sign (+) is displayed to the immediate left of the leftmost character position determined by the rest of the edit string. If the data value is negative, a blank ( ) is displayed to the left of the leftmost character position.</p> <p>If the data value is nonnumeric, each instance of this tag is treated as a single occurrence of the tag: a plus sign (+) is moved to the edited string for each occurrence of this tag. For a nonnumeric value, a PLUS tag is equivalent to a STR_LITERAL tag with the plus value (+).</p> <p>The PLUS tag must not appear in the same edit string with another tag designating a sign (for example, MINUS, ENCODED_MINUS) when used for numeric values.</p> |
| ESF\$K_EDS_PLUS_LITERAL | Handle of ESF\$_TXS aggregate | <p>Has two behaviors, depending on the number of times it appears in an edit string. If the tag is specified only once, and the data value is positive, the literal (text string) is inserted in the next character position of the edited string. If the tag is specified only once, and the data value is nonpositive, as many blanks as characters in the text literal are moved to the edited string (thereby preserving alignment for both positive and negative values).</p>   |

(continued on next page)

## ESF\$\_EDS

Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator   | Edit String Value | Description   |
|-------------------------|-------------------|---|
| ESF\$_K_EDS_RADIX_POINT | None              | <p>If two or more of these tags occur consecutively before the numeric value portion of the edit string, the plus literal text floats to the beginning of the numeric data value in the edited string. Any leading zeros or digit separators that the tag matches are replaced with blanks. If the data value is positive, the text literal is moved to the next character position of the edited string. If the data value is negative, as many blanks as there are characters in the text literal are moved to the edited string.</p> <p>If the data value is nonnumeric, each instance of this tag is treated as a single occurrence of the tag; the text literal is moved to the edited string for each occurrence of this tag. For nonnumeric values, a PLUS_LITERAL tag is equivalent to a STR_LITERAL tag with the text literal value.</p> <p>The PLUS_LITERAL tag must not be used in the same edit string as any other tag designating a sign (for example, MINUS, PLUS, or MINUS_LITERAL).</p> <p>Inserts the radix-point character into the edited string and indicates the start of the fractional portion of the numeric item. The radix-point character is usually a period (.) or comma (,), depending on the language and country.</p> <p>In the context of a DTIF table, the DTIF\$_LPT_ITEMS text string assigned to represent the currency radix or numeric radix symbol must be used to select the radix-point characters. For more information on the DTIF\$_LPT_ITEMS item, see the description of the DTIF\$_LPT aggregate.</p> <p>If present, this tag can be specified only once within an edit string. However, if this tag appears more than once in an edit string, the radix-point character is again inserted into the edited string, but this has no effect upon the next fractional digit. In this case, the second RADIX_POINT tag functions like a STR_LITERAL tag with the radix-point character as its value.</p> |

(continued on next page)

Table 7-2 (Cont.): Valid Values for ESF\$ \_EDS\_EDIT\_STRING\_C

| Edit String Indicator      | Edit String Value              | Description   |
|----------------------------|--------------------------------|---|
| ESF\$K_EDS_RADIX_POINT_LIT | Handle of ESF\$ _TXS aggregate | <p>Has a behavior identical to that of the RADIX_POINT tag, except that the characters to be inserted as the radix point are explicitly included in the RADIX_POINT_LIT tag.</p> <p>The RADIX_POINT_LIT tag inserts the radix-point character into the edited string and indicates the start of the fractional portion of the numeric item. If present, this tag can be specified only once within an edit string. However, if this tag appears more than once in an edit string, the radix-point character is again inserted into the edited string, but this has no effect upon the next fractional digit. In this case, the second radix-point functions like a STR_LITERAL tag with the radix-point character as its value.</p> |
| ESF\$K_EDS_REPEAT          | Handle of ESF\$ _RPT aggregate | Defines a repeated sequence of a single edit string tag.  |
| ESF\$K_EDS_REVERSE         | None                           | <p>Is used to indicate that all edited characters, from the current character position to the end of the edit-string or the next reverse tag, are to be displayed in reverse order. This tag "reverses" the edited output and is applied <i>after</i> the value is formatted.</p>   |
| ESF\$K_EDS_SECOND          | None                           | Causes one digit from the seconds portion of a date/time value to be moved to the edited string. Two consecutive occurrences of this tag are recommended.   |
| ESF\$K_EDS_SIGN            | None                           | <p>Has two behaviors, depending on the number of times it appears in an edit string. If the tag is specified only once and the data value is negative, the minus sign (-) is inserted in the next character position of the edited string. If the tag is specified only once, and the data value is nonnegative, then a plus sign (+) is inserted in the next character position of the edited string.</p>  |

(continued on next page)

# ESF\$\_EDS

Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator  | Edit String Value             | Description  |
|------------------------|-------------------------------|--|
| ESF\$K_EDS_STR_LITERAL | Handle of ESF\$ TXS aggregate | <p>If two or more consecutive SIGN tags occur at the beginning of the edit string, any leading zeros that the tag matches are suppressed (not displayed), and the sign of the value is displayed to the immediate left of the leftmost character position determined by the rest of the edit string.</p> <p>If the data value is nonnumeric, each instance of this tag is treated as a single occurrence of the PLUS tag: a plus sign (+) is moved to the edited string for each occurrence of this tag. For a nonnumeric value, a SIGN tag is equivalent to a STR_LITERAL tag with the plus value (+).</p> <p>The SIGN tag must not appear in the same edit string with another tag designating a sign (for example, PLUS, ENCODED_MINUS) when used for numeric values.</p> |
| ESF\$K_EDS_UPPERCASE   | None                          | <p>Is used to insert a text string into the edited string at the current position. The value of this tag is the text string to be inserted, as defined by the text-string type.</p> <p>It is the decoding application's responsibility to determine whether a data value is missing. This usually corresponds to a value that has not been specified (for example, left blank).</p> <p>Is used to indicate that any alphabetic characters displayed in the edit string, from the current character position on, are to be displayed as uppercase characters. This tag does not reserve any space in the edited string for the displayed value; it is used to modify the behavior of the remaining edit string tags.</p>  |
| ESF\$K_EDS_WEEKDAYNAME | None                          | <p>Causes the next letter from the name of the day of the week to be moved to the edited string. This tag should only be used for date/time data values.</p>   |

(continued on next page)



Table 7-2 (Cont.): Valid Values for ESF\$\_EDS\_EDIT\_STRING\_C

| Edit String Indicator   | Edit String Value             | Description  |
|-------------------------|-------------------------------|--|
| ESF\$K_EDS_YEAR         | None                          | <p>If the edit string is used within the context of a DTIF table, the DTIF\$_LPT_ITEMS text strings assigned to represent the days of the week must be used to display the appropriate text for each week day name. For more information on the DTIF\$_LPT_ITEMS item, see the description of the DTIF\$_LPT aggregate.</p> <p>Causes the next digit of the year portion of a date value to be moved to the edited string. This tag should only be used for date/time data values. Digital recommends that at least two consecutive instances of this tag be used in an edit string. The number of year tags determines the number of <i>least</i> significant digits to be displayed.</p> |
| ESF\$K_EDS_ZERO_REPLACE | Handle of ESF\$_TXS aggregate | <p>Is used to define a (text) STR_LITERAL value to be displayed in place of a leading zero digit in the data value. If the data value digit is nonzero, the ZERO_REPLACE tag is equivalent to the DECIMAL_DIGIT, HEX_DIGIT, or OCTAL_DIGIT tag, depending on the other tags in the edit string. The ZERO_REPLACE tag uses a text-string primitive as its value. This tag counts as a digit position in the edited string.</p>  |

**ESF\$\_EDS\_EDIT\_STRING****Encoding: array of type variable**

A list item that contains a list of the actual data values for the edit string value types selected in the previous item.

## ESF\$\_EXT

---

### ESF\$\_EXT—ESF Application Private Aggregate

The application private aggregate defines edit strings that are restricted either to a particular data processing implementation, or to a set of related implementations that support identical private encodings. The ESF\$\_EXT aggregate is referenced by the parent aggregate item ESF\$\_NVL\_VALUE.

Refer to these corresponding syntax diagrams:

| Syntax      | Location   |
|-------------|------------|
| ApplPrivate | Figure E-5 |
| Single      | Figure E-3 |

---

### AGGREGATE FORMAT

| Item Name                       | Item Encoding     |
|---------------------------------|-------------------|
| ESF\$_EXT_DIRECT_REFERENCE      | Object identifier |
| ESF\$_EXT_INDIRECT_REFERENCE    | Integer           |
| ESF\$_EXT_DATA_VALUE_DESCRIPTOR | String            |
| ESF\$_EXT_ENCODING_C            | Enumeration       |
| ESF\$_EXT_ENCODING              | Variable          |
| ESF\$_EXT_ENCODING_L            | Integer           |

---

### AGGREGATE ITEMS

#### **ESF\$\_EXT\_DIRECT\_REFERENCE**

**Encoding: object identifier**

An optional direct reference item that is used to identify the data type (syntax and semantics) of the external edit string.

#### **ESF\$\_EXT\_INDIRECT\_REFERENCE**

**Encoding: integer**

An optional indirect reference item.

#### **ESF\$\_EXT\_DATA\_VALUE\_DESCRIPTOR**

**Encoding: string**

An optional data value descriptor item that is a text string describing the data value to programs, to people, or to both. Conforming encoding applications must use one of the standard strings assigned to the data type.

**ESF\$\_EXT\_ENCODING\_C****Encoding: enumeration**

An encoding indicator that specifies the type of value chosen from those delineated as methods of encoding the data value. Valid values for this item are as follows:

|                            |   |
|----------------------------|---|
| ESF\$_K_DOCUMENT_ENCODING  | Nested document. In this case, the ESF\$_EXT_ENCODING item is encoded as a document root aggregate.                             |
| ESF\$_K_DDIS_ENCODING      | Nested document. In this case, the ESF\$_EXT_ENCODING item is encoded as a DIGITAL Document Interchange Syntax (DDIS) encoding. |
| ESF\$_K_OCTET_ENCODING     | Octet-aligned encoding. In this case, the ESF\$_EXT_ENCODING item is encoded as a string.                                       |
| ESF\$_K_ARBITRARY_ENCODING | Arbitrary. In this case, the ESF\$_EXT_ENCODING item is encoded as a bit string.  |

**ESF\$\_EXT\_ENCODING****Encoding: variable**

An encoding item that specifies the actual data value for the value type selected in the previous item.

**ESF\$\_EXT\_ENCODING\_L****Encoding: integer**

An encoding length item that specifies the length (on output) of the encoding.

## ESF\$\_NVL

---

### ESF\$\_NVL—ESF Named Value Aggregate

The ESF named value aggregate specifies a list of named values, each of which is in turn defined as a name followed by a value. The value is one of a choice of values. The ESF\$\_NVL aggregate is referenced by the parent aggregate items ESF\$\_EDS\_EDIT\_STRING\_C, ESF\$\_NVL\_VALUE, and ESF\$\_RPT\_SEQ\_C.

Refer to these corresponding syntax diagrams:

| Syntax         | Location    |
|----------------|-------------|
| ApplPrivate    | Figure E-5  |
| NamedValueList | Figure C-31 |

---

### AGGREGATE FORMAT

| Item Name         | Item Encoding |
|-------------------|---------------|
| ESF\$_NVL_NAME    | String        |
| ESF\$_NVL_VALUE_C | Enumeration   |
| ESF\$_NVL_VALUE   | Variable      |

---

### AGGREGATE ITEMS

#### **ESF\$\_NVL\_NAME**

**Encoding: string**

A value name item that is limited to the characters of the ASCII character set and that identifies the named value.

#### **ESF\$\_NVL\_VALUE\_C**

**Encoding: enumeration**

A value data indicator that specifies the type of value chosen from those that are delineated for the named data value. Valid values for this item are as follows:

|                      |  |
|----------------------|--|
| ESF\$K_VALUE_BOOLEAN | Indicates a Boolean value. In this case, the ESF\$_NVL_VALUE item is encoded as a Boolean value.                       |
| ESF\$K_VALUE_INTEGER | Indicates an integer value. In this case, the ESF\$_NVL_VALUE item is encoded as an integer.                           |
| ESF\$K_VALUE_TEXT    | Indicates a text string value. In this case, the ESF\$_NVL_VALUE item is encoded as an array of type character string. |

|                       |  |
|-----------------------|--|
| ESF\$K_VALUE_GENERAL  | Indicates a stream of bytes in any format. In this case, the ESF\$_NVL_VALUE item is encoded as a string.  |
| ESF\$K_VALUE_LIST     | Indicates a list of data values such as the preceding. In this case, the ESF\$_NVL_VALUE item is encoded as a sequence of ESF\$_NVL aggregates. In the nested ESF\$_NVL aggregates, the ESF\$_NVL_NAME item is ignored.          |
| ESF\$K_VALUE_EXTERNAL | Indicates a data value that is represented in a syntax. In this case, the ESF\$_NVL_VALUE item is encoded as the handle of an aggregate of type ESF\$_EXT. For more information, see the description of the ESF\$_EXT aggregate. |
| ESF\$K_VALUE_FLOAT    | Indicates a floating-point value. In this case, the ESF\$_NVL_VALUE item is encoded as a general floating-point value.   |
| ESF\$K_VALUE_DATE     | Indicates a date/time value. In this case, the ESF\$_NVL_VALUE item is encoded as the handle of a ESF\$_DAT aggregate. For more information, see the description of the ESF\$_DAT aggregate.                                     |
| ESF\$K_VALUE_EXPR     | Indicates an expression whose result is the value. In this case, the ESF\$_NVL_VALUE item is encoded as the handle of a CFE\$_EXP aggregate. For more information, see the description of the CFE\$_EXP aggregate.               |

## **ESF\$\_NVL\_VALUE**

### **Encoding: variable**

A value data item that contains the actual data value for the value type selected by the previous item.

## ESF\$\_RPT

---

### ESF\$\_RPT—Repeat Aggregate

The repeat aggregate defines a repeated sequence of a single edit string tag. The ESF\$\_RPT aggregate is referenced by the parent aggregate item ESF\$\_EDS\_EDIT\_STRING\_C.

Refer to these corresponding syntax diagrams:

---

| Syntax | Location   |
|--------|------------|
| Repeat | Figure E-4 |

---

---

### AGGREGATE FORMAT

---

| Item Name       | Item Encoding |
|-----------------|---------------|
| ESF\$_RPT_COUNT | Integer       |
| ESF\$_RPT_SEQ_C | Enumeration   |
| ESF\$_RPT_SEQ   | Variable      |

---

---

### AGGREGATE ITEMS

#### ***ESF\$\_RPT\_COUNT***

**Encoding:** *integer*

A repeat count item that specifies the number of times the ESF\$\_RPT\_SEQ item is to be repeated.

#### ***ESF\$\_RPT\_SEQ\_C***

**Encoding:** *enumeration*

A single edit string indicator that specifies the type of value chosen from those that are delineated for the named data item. Valid values for this item are shown in Table 7-3.

**Table 7-3: Valid Values for ESF\$\_RPT\_SEQ\_C**

---

|                         |      |
|-------------------------|------|
| ESF\$K_EDS_ALPHABETIC   | None |
| ESF\$K_EDS_AM_PM        | None |
| ESF\$K_EDS_ANY_CHAR     | None |
| ESF\$K_EDS_ANY_CASE     | None |
| ESF\$K_EDS_BINARY_DIGIT | None |

---

(continued on next page)

Table 7-3 (Cont.): Valid Values for ESF\$\_RPT\_SEQ\_C

---

|                             |                               |
|-----------------------------|-------------------------------|
| ESF\$K_EDS_DIGIT_SEP        | None                          |
| ESF\$K_EDS_DAY_NUMBER       | None                          |
| ESF\$K_EDS_DECIMAL_DIGIT    | None                          |
| ESF\$K_EDS_RADIX_POINT      | None                          |
| ESF\$K_EDS_ENCODED_MINUS    | None                          |
| ESF\$K_EDS_ENCODED_PLUS     | None                          |
| ESF\$K_EDS_ENCODED_SIGN     | None                          |
| ESF\$K_EDS_EXPONENT         | None                          |
| ESF\$K_EDS_ZERO_REPLACE     | Handle of ESF\$_TXS aggregate |
| ESF\$K_EDS_CURRENCY         | None                          |
| ESF\$K_EDS_MINUS            | None                          |
| ESF\$K_EDS_PLUS             | None                          |
| ESF\$K_EDS_SIGN             | None                          |
| ESF\$K_EDS_FLOAT_BLANK_SUPR | None                          |
| ESF\$K_EDS_FRACTION_SECOND  | None                          |
| ESF\$K_EDS_HEX_DIGIT        | None                          |
| ESF\$K_EDS_HOUR_12          | None                          |
| ESF\$K_EDS_HOUR_24          | None                          |
| ESF\$K_EDS_JULIAN_DIGIT     | None                          |
| ESF\$K_EDS_LOGICAL_CHAR     | None                          |
| ESF\$K_EDS_LONG_TEXT        | None                          |
| ESF\$K_EDS_LOWERCASE        | None                          |
| ESF\$K_EDS_MINUS_LITERAL    | Handle of \$ESF_TXS aggregate |
| ESF\$K_EDS_MINUS_LIT_END    | None                          |
| ESF\$K_EDS_MINUTE           | None                          |
| ESF\$K_EDS_MONTH_NAME       | None                          |
| ESF\$K_EDS_MONTH_NUMBER     | None                          |
| ESF\$K_EDS_OCTAL_DIGIT      | None                          |
| ESF\$K_EDS_PLUS_LITERAL     | Handle of ESF\$_TXS aggregate |
| ESF\$K_EDS_REVERSE          | None                          |
| ESF\$K_EDS_SECOND           | None                          |
| ESF\$K_EDS_STR_LITERAL      | Handle of ESF\$_TXS aggregate |
| ESF\$K_EDS_MISSING_SEP      | None                          |
| ESF\$K_EDS_UPPERCASE        | None                          |
| ESF\$K_EDS_WEEKDAYNAME      | None                          |
| ESF\$K_EDS_YEAR             | None                          |

---

(continued on next page)

## ESF\$\_RPT

Table 7-3 (Cont.): Valid Values for ESF\$\_RPT\_SEQ\_C

---

|                            |                                  |
|----------------------------|----------------------------------|
| ESF\$K_EDS_APPL_PRIVATE    | Sequence of ESF\$_NVL aggregates |
| ESF\$K_EDS_DIGIT_SEP_LIT   | Handle of ESF\$_TXS aggregate    |
| ESF\$K_EDS_RADIX_POINT_LIT | Handle of ESF\$_TXS aggregate    |
| ESF\$K_EDS_CURRENCY_LIT    | Handle of ESF\$_TXS aggregate    |

---

### ***ESF\$\_RPT\_SEQ***

#### ***Encoding: variable***

A single edit string item that contains the actual data value for the repeat edit string value type selected in the previous item.



---

## ESF\$ \_TXS—Text String Aggregate

The text string aggregate contains data that pertains to those edit strings encoded as text strings. The ESF\$ \_TXT aggregate is referenced by the parent aggregate items ESF\$ \_EDS\_EDIT\_STRING\_C and ESF\$ \_RPT\_SEQ\_C.

Refer to these corresponding syntax diagrams:

| Syntax      | Location   |
|-------------|------------|
| Single      | Figure E-3 |
| Text-String | Table B-4  |

---

### AGGREGATE FORMAT

| Item Name              | Item Encoding                  |
|------------------------|--------------------------------|
| ESF\$ _TXS_TEXT_STRING | Array of type character string |

---

### AGGREGATE ITEMS

***ESF\$ \_TXS\_TEXT\_STRING***

***Encoding: array of type character string***

A text string item that contains data pertaining to an edit string.



## A

- Absolute value construct
  - See ExpressionList syntax diagram and CFE expressions, Boolean and relational
- Abs-value syntax diagram
  - See ExpressionList syntax diagram
- Acos syntax diagram
  - See ExpressionList syntax diagram
- Add-info
  - See LOCATE ITEM routine; STORE ITEM routine
- Addition construct
  - See ExpressionList syntax diagram and CFE expressions, arithmetic
- Add syntax diagram
  - See ExpressionList syntax diagram
- aggregate
  - DTIF\$\_LPT\_APPL\_PRIVATE item in, 5-46
- Aggregate
  - See also Root aggregate
  - copying, 8-33
  - creating, 8-36
  - deleting, 8-58
  - determining number of array elements in, 8-94
  - inserting, 8-108
  - locating an item in, 8-115
  - locating next in sequence, 8-120
  - reading from a front end, 8-24
  - reading from a stream, 8-88
  - removing from a document, 8-145
  - removing from a sequence, 8-158
  - writing the contents of, 8-150, 8-160
- Aggregates, DDIF, 1-5
- Aggregates, list of, 1-5
- AGGREGATE TYPE TO OBJECT ID routine, 8-3
- Allocate/Deallocate routines, 9-2
- Alog syntax diagram
  - See ExpressionList syntax diagram
- Alternate presentation, 4-101
- Analysis back end, 2-8
- And syntax diagram
  - See ExpressionList syntax diagram
- AngleRef enumeration, 1-1
- AngleRef syntax diagram, B-25
- Angle syntax diagram, B-25
- \$AN tag, defined, 4-91
- Antilog construct
  - See ExpressionList syntax diagram and CFE expressions, transcendental
- ANY built-in operator, defined, B-3
- Application images
  - linking on ULTRIX, 8-2
  - linking on VMS, 8-2
- Application private data
  - See DTIF application private aggregate
- AppIPrivate syntax diagram, B-5, C-14, E-3
- Appreciation construct
  - See ExpressionList syntax diagram and CFE expressions, financial
- Apprec syntax diagram
  - See ExpressionList syntax diagram
- Arc
  - controlling the rendition of, 4-3
  - specifying angle of rotation of, 4-4
  - specifying center x-coordinate of, 4-3
  - specifying center y-coordinate of, 4-3
  - specifying delta y radius of, 4-4
  - specifying the extent of, 4-4
  - specifying the starting angle of, 4-4
  - specifying x radius of, 4-3
- Arc content aggregate, 4-2 to 4-4
  - See also DDIF\$\_ARC aggregate
  - arc extent indicator item in, 4-4
  - arc start indicator item in, 4-4
  - center x indicator item in, 4-3
  - center y indicator item in, 4-3
  - delta y indicator item, 4-4
  - flags item in, 4-3
  - items in, 4-2t
  - rotation indicator item in, 4-4
  - x radius item in, 4-3
- Arc cosine construct
  - See ExpressionList syntax diagram and CFE expressions, trigonometric
- ArcPath syntax diagram, B-30
- Arc sine construct
  - See ExpressionList syntax diagram and CFE expressions, trigonometric
- Arc syntax diagram, B-12
- Arc tangent 2 construct
  - See ExpressionList syntax diagram and CFE expressions, trigonometric
- Arc tangent construct
  - See ExpressionList syntax diagram and CFE expressions, trigonometric
- Arithmetic shift left construct
  - See ExpressionList syntax diagram and CFE expressions, binary

Arithmetic shift right construct  
 See ExpressionList syntax diagram and CFE expressions, binary

Array definition aggregate, 5–2 to 5–3  
 array values item in, 5–3  
 descriptor item in, 5–2  
 element type size indicator item in, 5–2  
 items in, 5–2  
 value item in, 5–3  
 x dimension item in, 5–3  
 y dimension item in, 5–3  
 z dimension item in, 5–3

ArrayDefn syntax diagram, C–8

Arrays  
 See Array values

Array values  
 specifying array element type size indicator for, 5–2  
 specifying descriptor item for, 5–2  
 specifying nonstandard data types for, 5–3  
 specifying standard data types for, 5–2  
 specifying the x dimension for, 5–3  
 specifying the y dimension for, 5–3  
 specifying the z dimension for, 5–3  
 specifying value item for, 5–3

\$ASCII collating sequence, 5–49

ASCIIString syntax diagram, B–25, C–15

Asin syntax diagram  
 See ExpressionList syntax diagram

Asl syntax diagram  
 See ExpressionList syntax diagram

Asr syntax diagram  
 See ExpressionList syntax diagram

Assignment built-in operator, defined, B–3

Atan2 syntax diagram  
 See ExpressionList syntax diagram

Atan syntax diagram  
 See ExpressionList syntax diagram

Attribute  
 specifying for a galley, 4–133  
 specifying for a line, 4–128 to 4–131  
 specifying for a marker, 4–132  
 specifying for an image, 4–134 to 4–136  
 specifying for image component space, 4–137 to 4–138  
 specifying for text, 4–118 to 4–127

Average construct  
 See ExpressionList syntax diagram and CFE expressions, statistical

Avg syntax diagram  
 See ExpressionList syntax diagram

## B

Back end  
 analysis, 2–8  
 DDIF, 2–3  
 ddif\$write\_format entry point, 11–16  
 domain\$write\_format entry point, 11–15  
 dtif\$write\_format entry point, 11–16  
 entry point, 8–16  
 PostScript, 2–5 to 2–8  
 text, 2–4 to 2–5

Background  
 See Pattern definition; Pattern definition aggregate; Image component space

BeginSegment syntax diagram, B–7

Between construct  
 See ExpressionList syntax diagram and CFE expressions, Boolean and relational

Bezier curve  
 controlling rendition of, 4–5  
 specifying layout of, 4–6

Bezier curve aggregate, 4–5 to 4–6  
 See also DDIF\$\_BEZ aggregate  
 curve path indicator item in, 4–6  
 flags item in, 4–5  
 items in, 4–5t

Binding syntax diagram, B–36

BIT STRING built-in primitive, defined, B–1

BMU (Basic Measurement Unit)  
 definition of, 4–100

BOOLEAN built-in primitive, defined, B–1

BoundingBox syntax diagram, B–18

BreakCriteria syntax diagram, B–42

## C

Callable interface  
 CDA viewer routines, 13–1

CategoryTag syntax diagram, B–35

cda\$aggregate\_type\_to\_object\_id, 8–3

cda\$close\_file, 8–6

cda\$close\_stream, 8–9

cda\$close\_text\_file, 8–11

cda\$convert, 8–13

cda\$convert\_aggregate, 8–24

cda\$convert\_document, 8–28

cda\$convert\_position, 8–31

cda\$copy\_aggregate, 8–33

cda\$create\_aggregate, 8–36

cda\$create\_file, 8–39

cda\$create\_root\_aggregate, 8–45

cda\$create\_stream, 8–50

cda\$create\_text\_file, 8–54

cda\$delete\_aggregate, 8–58

cda\$delete\_root\_aggregate, 8–60

cda\$enter\_scope, 8–62

cda\$erase\_item, 8–76

cda\$find\_definition, 8–79

cda\$find\_transformation, 8–83

cda\$flush\_stream, 8–86

cda\$get\_aggregate, 8–88

cda\$get\_array\_size, 8–94

cda\$get\_document, 8–97

cda\$get\_external\_encoding, 8–100

cda\$get\_stream\_position, 8–103

cda\$get\_text\_position, 8–106

cda\$insert\_aggregate, 8–108

cda\$leave\_scope, 8–112

cda\$locate\_item, 8–115

cda\$next\_aggregate, 8–120

cda\$object\_id\_to\_aggregate\_type, 8–123

cda\$open\_converter, 8–126

cda\$open\_file, 8–130

cda\$open\_stream, 8–138

cda\$open\_text\_file, 8–141

cda\$prune\_aggregate, 8–145

cda\$prune\_position, 8–148

- cda\$put\_aggregate, 8–150
- cda\$put\_document, 8–153
- cda\$read\_text\_file, 8–156
- cda\$remove\_aggregate, 8–158
- cda\$store\_item, 8–160
- cda\$write\_text\_file, 8–168
- CDA\$\_facility messages, G–1 to G–6
- CdaAggregateTypeToObjectId, 8–3
- CdaCloseFile, 8–6
- CdaCloseStream, 8–9
- CdaCloseTextFile, 8–11
- CdaConvert, 8–13
- CdaConvertAggregate, 8–24
- CdaConvertDocument, 8–28
- CdaConvertPosition, 8–31
- CdaCopyAggregate, 8–33
- CdaCreateAggregate, 8–36
- CdaCreateFile, 8–39
- CdaCreateRootAggregate, 8–45
- CdaCreateStream, 8–50
- CdaCreateTextFile, 8–54
- CDA default values, 1–4
- CdaDeleteAggregate, 8–58
- CdaDeleteRootAggregate, 8–60
- CdaEnterScope, 8–62
- CdaEraseItem, 8–76
- CdaFindDefinition, 8–79
- CdaFindTransformation, 8–83
- CdaFlushStream, 8–86
- CdaGetAggregate, 8–88
- CdaGetArraySize, 8–94
- CdaGetDocument, 8–97
- CdaGetExternalEncoding, 8–100
- CdaGetStreamPosition, 8–103
- CdaGetTextPosition, 8–106
- CdaInsertAggregate, 8–108
- CDA item data types, 1–1t
- CdaLeaveScope, 8–112
- CdaLocateItem, 8–115
- CdaNextAggregate, 8–120
- CdaObjectIdToAggregateType, 8–123
- CdaOpenConverter, 8–126
- CdaOpenFile, 8–130
- CdaOpenStream, 8–138
- CdaOpenTextFile, 8–141
- CdaPruneAggregate, 8–145
- CdaPrunePosition, 8–148
- CdaPutAggregate, 8–150
- CdaPutDocument, 8–153
- CdaReadTextFile, 8–156
- CdaRemoveAggregate, 8–158
- CdaStoreItem, 8–160
- CDA viewer
  - callable interface, 13–1
- CdaWriteTextFile, 8–168
- cdoc command, 2–16 to 2–17
  - d format qualifier, 2–16
  - O options qualifier, 2–17
  - s format qualifier, 2–16
- Cell-col syntax diagram
  - See ExpressionList syntax diagram
- Cell coordinate construct
  - See ExpressionList syntax diagram and CFE expressions, variables
- Cell coordinates aggregate
  - See CFE cell coordinates aggregate
- Cell coordinates aggregate (Cont.)
  - See DTIF cell coordinates aggregate
- Cell-coord syntax diagram
  - See ExpressionList syntax diagram
- CellCoord syntax diagram, C–16
- Cell data
  - empty cells in, 5–13
  - missing values in, 5–13
  - null cells in, 5–13
  - sparse encoding of, 5–13
  - specifying a cell state in, 5–14
  - specifying a column number in, 5–14
  - specifying a descriptor item for, 5–15
  - specifying a value data indicator for, 5–15
  - specifying expressions for, 5–16
  - specifying formats for, 5–15
  - specifying private data for, 5–15
- Cell data aggregate, 5–13 to 5–17
  - cell descriptor item in, 5–15
  - cell state item in, 5–14
  - column number item in, 5–14
  - expression item in, 5–16
  - format item in, 5–15
  - items in, 5–14
  - private cell data item in, 5–15
  - value data indicator item in, 5–15
  - value item in, 5–16
- CellData syntax diagram, C–6
- Cell extract construct
  - See ExpressionList syntax diagram and CFE expressions, cell-related
- Cell-extract syntax diagram
  - See ExpressionList syntax diagram
- Cell indirection construct
  - See ExpressionList syntax diagram and CFE expressions, cell-related
- Cell-indirect syntax diagram
  - See ExpressionList syntax diagram
- Cell-name syntax diagram
  - See ExpressionList syntax diagram
- Cell range aggregate
  - See CFE cell range aggregate
  - See DTIF cell range aggregate
- Cell range construct
  - See ExpressionList syntax diagram and CFE expressions, variables
- Cell-range syntax diagram
  - See ExpressionList syntax diagram
- CellRange syntax diagram, C–18
- Cell-row syntax diagram
  - See ExpressionList syntax diagram
- Cells
  - See Cell data
- Cell states
  - to determine valid cell values, 5–15
- CellValue syntax diagram, C–7
- CFE\$\_AGGREGATE\_TYPE aggregate item, 6–1
- CFE\$\_CCD aggregate, 6–2 to 6–3
  - CFE\$\_CCD\_COLUMN item in, 6–2
  - CFE\$\_CCD\_FLAGS item in, 6–2
  - CFE\$\_CCD\_ROW item in, 6–2
  - items in, 6–2
- CFE\$\_CFT aggregate, 6–4
  - CFE\$\_CFT\_REAL\_PART item in, 6–4
  - items in, 6–4

- CFE\$\_CLR aggregate, 6-5
  - CFE\$\_CLR\_RANGE\_BEGIN item in, 6-5
  - CFE\$\_CLR\_RANGE\_END item in, 6-5
  - items in, 6-5
- CFE\$\_COR aggregate, 6-6
  - CFE\$\_COR\_COL\_BEGIN item in, 6-6
  - CFE\$\_COR\_COL\_END item in, 6-6
  - items in, 6-6
- CFE\$\_DAT aggregate, 6-7 to 6-8
  - CFE\$\_DAT\_DATETIME item in, 6-7
  - CFE\$\_DAT\_TIME\_DIFF item in, 6-8
  - CFE\$\_DAT\_TIME\_DIFF\_C item in, 6-7
  - items in, 6-7
- CFE\$\_EXL aggregate, 6-9 to 6-52
  - CFE\$\_EXL\_EXPR item in, 6-52
  - items in, 6-9
- CFE\$\_EXP aggregate, 6-53 to 6-54
  - CFE\$\_EXP\_LIST item in, 6-53
  - CFE\$\_EXP\_MAJOR\_VERSION item in, 6-53
  - CFE\$\_EXP\_MINOR\_VERSION item in, 6-53
  - items in, 6-53
- CFE\$\_FLT aggregate
  - CFE\$\_CFT\_IMAGINARY\_PART item in, 6-4
- CFE\$\_FRF aggregate, 6-55
  - CFE\$\_FRF\_FIELD\_CONTEXT item in, 6-55
  - CFE\$\_FRF\_PATH item in, 6-55
  - items in, 6-55
- CFE\$\_NPM aggregate, 6-56
  - CFE\$\_NPM\_NAME item in, 6-56
  - CFE\$\_NPM\_VALUE item in, 6-56
  - items in, 6-56
- CFE\$\_PEX
  - CFE\$\_PEX\_END\_EXPR item in, 6-57
- CFE\$\_PEX aggregate, 6-57
  - CFE\$\_PEX\_BEGIN\_EXPR item in, 6-57
  - CFE\$\_PEX\_VALUE\_EXPR item in, 6-57
  - CFE\$\_PFE\_FACILITY item in, 6-58
  - CFE\$\_PFE\_NAME item in, 6-58
  - items in, 6-57
- CFE\$\_PFE aggregate, 6-58 to 6-59
  - CFE\$\_PFE\_PARAMS item in, 6-59
  - CFE\$\_PFE\_REF\_LABEL item in, 6-58
  - CFE\$\_PFE\_REF\_LABEL\_TYPE item in, 6-58
  - CFE\$\_PFE\_RETURN\_TYPE item in, 6-59
  - items in, 6-58
- CFE\$\_RWR aggregate, 6-60
  - CFE\$\_RWR\_ROW\_BEGIN item in, 6-60
  - CFE\$\_RWR\_ROW\_END item in, 6-60
  - items in, 6-60
- CFE\$\_SLL aggregate, 6-61
  - CFE\$\_SLL\_CRITERIA item in, 6-61
  - CFE\$\_SLL\_SELECTION item in, 6-61
  - items in, 6-61
- CFE\$\_STF aggregate, 6-62
  - CFE\$\_STF\_EDIT\_STRING item in, 6-62
  - CFE\$\_STF\_SOURCE item in, 6-62
  - items in, 6-62
- CFE\$\_STP aggregate, 6-63
  - CFE\$\_STP\_PRETTY\_FLAGS item in, 6-63
  - CFE\$\_STP\_STRING\_EXPR item in, 6-63
  - items in, 6-63
- CFE\$\_TXC aggregate, 6-64
  - CFE\$\_TXC\_TEXT\_C item in, 6-64
  - items in, 6-64
- CFE\$\_USER\_CONTEXT aggregate item, 6-1
- CFE\$\_VTX aggregate, 6-65
  - CFE\$\_VTX\_VTEXT\_LEN item in, 6-65

- CFE\$\_VTX aggregate (Cont.)
  - CFE\$\_VTX\_VTEXT\_STR item in, 6-65
  - items in, 6-65
- CFE application private aggregate, 6-58 to 6-59
  - facility name item in, 6-58
  - function name item in, 6-58
  - items in, 6-58
  - parameters item in, 6-59
  - reference label item in, 6-58
  - reference storage item in, 6-58
  - return type item in, 6-59
- CFE cell coordinates
  - absolute references, 6-2
  - relative references, 6-2
- CFE cell coordinates aggregate, 6-2 to 6-3
  - column identifier item in, 6-2
  - flags item in, 6-2
  - items in, 6-2
  - row identifier item in, 6-2
- CFE cell range aggregate, 6-5
  - items in, 6-5
  - range begin item in, 6-5
  - range end item in, 6-5
- CFE column range aggregate, 6-6
  - column begin item in, 6-6
  - column end item in, 6-6
  - items in, 6-6
- CFE complex float aggregate, 6-4
  - imaginary portion value item in, 6-4
  - items in, 6-4
  - real portion value item in, 6-4
- CFE date and time aggregate, 6-7 to 6-8
  - date and time item in, 6-7
  - items in, 6-7
  - time difference item in, 6-7
- CFE document
  - specifying date and time values for, 6-7
- CFE expressions
  - arithmetic
    - addition, 6-10
    - division, 6-10
    - multiplication, 6-10
    - negation, 6-10
    - percent, 6-10
    - raise to a power, 6-10
    - subtraction, 6-11
    - unary plus, 6-11
  - binary
    - arithmetic shift left, 6-11
    - arithmetic shift right, 6-11
    - one's complement, 6-11
  - Boolean and relational
    - absolute value, 6-12
    - between, 6-12
    - conditional (if-then-else), 6-13
    - equal to, 6-12
    - greater than, 6-12
    - greater than or equal to, 6-12
    - less than, 6-13
    - less than or equal to, 6-13
    - logical AND, 6-12
    - logical NOT, 6-13
    - logical OR, 6-14
    - modulus, 6-13
    - not equal to, 6-13
    - square root, 6-14

## CFE expressions, (Cont.)

- cell-related
  - cell extract, 6-14
  - cell indirection, 6-15
  - column portion of cell name, 6-14
  - constructed cell reference, 6-15
  - count columns, 6-15
  - count rows, 6-16
  - current cell, 6-16
  - current column, 6-16
  - current row, 6-16
  - error, 6-16
  - not available, 6-16
  - not calculable, 6-16
  - null, 6-16
  - row portion of cell name, 6-15
- choose and lookup
  - choose, 6-17
  - field in table, 6-18
  - hlookup, 6-17
  - index, 6-17
  - matches, 6-18
  - table, 6-18
  - vlookup, 6-19
- contants
  - true, 6-42
- conversion
  - convert to value, 6-19
  - decimal string, 6-19
  - integer, 6-19
  - round to the nearest whole number, 6-20
  - truncate, 6-20
- convert string to date, 6-21
- convert string to time, 6-21
- date difference
  - day, 6-21
  - hour, 6-21
  - minute, 6-21
  - month, 6-22
  - second, 6-22
  - week, 6-22
  - year, 6-22
- date/time
  - date day of the week, 6-23
  - date month name, 6-24
  - day of the week, 6-24
  - month name, 6-25
- date/time addition
  - days, 6-25
  - hours, 6-25
  - minutes, 6-25
  - months, 6-25
  - seconds, 6-26
  - weeks, 6-26
  - years, 6-26
- date/time extraction
  - day, 6-22
  - hour, 6-22
  - minute, 6-23
  - month, 6-23
  - second, 6-23
  - year, 6-23
- date/time literals
  - now, 6-25
  - today, 6-26
  - tomorrow, 6-26
  - yesterday, 6-26

## CFE expressions (Cont.)

- field reference, 6-42
- financial
  - appreciation, 6-27
  - depreciation—declining balance, 6-29
  - depreciation—declining balance with crossover to straight line, 6-28
  - depreciation—straight line, 6-30
  - depreciation—sum of year's digits, 6-30
  - discount, 6-31
  - financial
    - depreciation—double declining balance, 6-29
  - future value, 6-31
  - future value of an annuity, 6-32
  - future value of a single sum, 6-32
  - interest payments, 6-33
  - interest rate, 6-39
  - internal rate of return, 6-34
  - modified internal rate of return, 6-34
  - net present value, 6-35
  - number of periods given present value, 6-36
  - number of periods to achieve future value, 6-36
  - payback, 6-35
  - payment per period given present value, 6-37
  - payment per period to achieve future value, 6-37
  - periods to achieve future value, 6-35
  - present value of an annuity, 6-38
  - present value to achieve future value, 6-39
  - principal, 6-38
- identification
  - isblank, 6-39
  - isdate, 6-39
  - iserror, 6-40
  - is not available, 6-40
  - is not calculable, 6-40
  - isnull, 6-40
  - isnumber, 6-40
  - isref, 6-40
  - isstring, 6-41
- literals
  - complex floating-point, 6-41
  - date, 6-41
  - floating-point, 6-41
  - integer, 6-41
  - scaled integer, 6-42
  - text string, 6-42
  - varying length text, 6-42
- miscellaneous
  - parenthesized, 6-42
- private, 6-43
- random number, 6-43
- series
  - integrate, 6-43
  - least squares, 6-43
  - logest, 6-43
  - sigma, 6-43
  - trend, 6-44
- sign, 6-43
- specifying for a cell, 5-16
- specifying for a column, 5-5
- statistical
  - average, 6-44
  - count, 6-44
  - maximum, 6-44

## CFE expressions

### statistical (Cont.)

- minimum, 6–45
- standard deviation, 6–45
- summation, 6–45
- variance, 6–46

### string

- contains substring, 6–47
- extract substring left, 6–48
- extract substring right, 6–49
- string character, 6–46
- string character code, 6–46
- string concatenate, 6–46
- string extract, 6–47
- string find substring, 6–47
- string fixed, 6–47
- string format, 6–48
- string length, 6–48
- string lowercase, 6–48
- string pretty, 6–48
- string proper, 6–48
- string repeat, 6–48
- string replace, 6–49
- string reverse, 6–49
- string starts with, 6–47
- string trim, 6–49
- string uppercase, 6–49

### transcendental

- antilog, 6–49
- exponent, 6–50
- factorial, 6–50
- log, base 10, 6–50
- log, base e (natural), 6–50

### trigonometric

- arc cosine, 6–50
- arc sine, 6–50
- arc tangent, 6–50
- arc tangent 2, 6–51
- cosine, 6–51
- sine, 6–51
- tangent, 6–51

### variables

- cell coordinate, 6–51
- cell range, 6–51
- column name, 6–51
- column number, 6–51
- column range, 6–52
- current value, 6–52
- false, 6–41
- identifier, 6–52
- named range, 6–52
- PI, 6–41
- row number, 6–52
- row range, 6–52

## CFE generic aggregate items, 6–1

- CFE\$\_AGGREGATE\_TYPE, 6–1
- CFE\$\_USER\_CONTEXT, 6–1

## CFE row range aggregate, 6–60

- items in, 6–60
- row begin data item in, 6–60
- row end data item in, 6–60

## CFE syntax diagrams, D–1 to D–9

## CFE varying text aggregate, 6–65

- character string data item in, 6–65
- items in, 6–65
- varying text length item in, 6–65

## Character set

- identifiers for, 4–32t

## Character string, defined, 1–2

## Character-String defined type, defined, B–4

## Character-String syntax diagram, B–5

## CHOICE built-in operator, defined, B–3

## Choose construct

- See ExpressionList syntax diagram and CFE expressions, choose and lookup

## Circle

- See Arc

## Close entry point, 11–3

## CLOSE FILE routine, 8–6

## CLOSE STREAM routine, 8–9

## CLOSE TEXT FILE routine, 8–11

## ColAttributes syntax diagram, C–9

## Collating sequences

- ASCII, 5–49
- Danish, 5–49
- Finnish, 5–49
- Multinational Character Set, 5–49
- Norwegian, 5–49
- private, 5–49
- Spanish, 5–49

## Col-name syntax diagram

- See ExpressionList syntax diagram

## Col-num syntax diagram

- See ExpressionList syntax diagram

## ColNum syntax diagram, C–16

## Color

- See also Image component space

- See Pattern definition

## Color syntax diagram, B–19

## Col-range syntax diagram

- See ExpressionList syntax diagram

## ColRange syntax diagram, C–18

## Column attributes

- specifying a column header name for, 5–8
  - specifying a column query name for, 5–7
  - specifying a descriptor for, 5–5
  - specifying a generic attribute reference in, 5–5
  - specifying an identifier for, 5–4
  - specifying a value for, 5–5
  - specifying default values for, 5–5
  - specifying expressions for, 5–5
  - specifying flags for, 5–8
  - specifying formats for, 5–5
  - specifying numeric or character string indicator for, 5–6
  - specifying private data for, 5–5
  - specifying the data length of, 5–8
  - specifying the data type of, 5–8
  - specifying the scale factor of, 5–8
  - specifying within table metadata, 5–63
- ## Column attributes aggregate, 5–4 to 5–9
- column descriptor item in, 5–5
  - column header name item in, 5–8
  - column query name item in, 5–7
  - data length item in, 5–8
  - data type item in, 5–8
  - default values indicator item in, 5–5
  - default values item in, 5–6
  - expression item in, 5–5
  - flags item in, 5–8
  - format item in, 5–5
  - generic attribute reference item in, 5–5



- Column attributes aggregate (Cont.)
  - identifier item in, 5-4
  - items in, 5-4
  - numeric or missing character string item in, 5-7
  - private column attribute data item in, 5-5
  - scale factor item in, 5-8
  - value item in, 5-5
- Column name construct
  - See ExpressionList syntax diagram and CFE expressions, variables
- Column number construct
  - See ExpressionList syntax diagram and CFE expressions, variables
- Column portion of cell name construct
  - See ExpressionList syntax diagram and CFE expressions, cell-related
- Column range aggregate
  - See CFE column range aggregate
  - See DTIF column range aggregate
- Column range construct
  - See ExpressionList syntax diagram and CFE expressions, variables
- Columns
  - See Column attributes
- Comment built-in operator, defined, B-3
- Complex float aggregate
  - See CFE complex float aggregate
  - See DTIF complex float aggregate
- Complex floating-point construct
  - See ExpressionList syntax diagram and CFE expressions, literals
- Complex floating-point values
  - specifying the imaginary portion for, 5-12, 6-4
  - specifying the real portion for, 5-12, 6-4
- ComplexFloat syntax diagram, C-8
- Composite path
  - arc component of, 4-71
  - Bézier component of, 4-71
  - indicating angle of rotation of, 4-73
  - indicating center x-coordinate of an arc in, 4-72
  - indicating center y-coordinate of an arc in, 4-72
  - indicating delta y radius of an arc in, 4-72
  - indicating extent of an arc in, 4-73
  - indicating type of path defined in, 4-71
  - indicating x radius of an arc in, 4-72
  - polyline component of, 4-71
  - referencing a component of, 4-71, 4-73
  - specifying layout of the curve in, 4-72
  - specifying polyline path in, 4-71
  - specifying starting angle of an arc in, 4-73
- Composite path aggregate, 4-70 to 4-73
  - See also DDIF\$\_PTH aggregate
  - arc center x indicator item in, 4-72
  - arc center y indicator item in, 4-72
  - arc extent indicator item in, 4-73
  - arc radius delta y indicator item in, 4-72
  - arc radius x indicator item in, 4-72
  - arc rotation indicator item in, 4-73
  - arc start indicator item in, 4-73
  - curve path indicator item in, 4-72
  - items in, 4-70t
  - line path indicator item in, 4-71
  - path indicator item in, 4-71
  - path reference item in, 4-73
- CompositePath syntax diagram, B-30
- Compound document
  - validating contents of, 8-130
- Computed content
  - copied, 4-93
  - cross-reference, 4-95
  - function, 4-96
  - indicating function parameters for, 4-96
  - specifying an index into a list of references for, 4-93
  - specifying attributes for, 4-92 to 4-96
  - specifying function name in, 4-96
  - specifying label of a segment being referenced by, 4-93
  - specifying label of the target segment, 4-95
  - specifying name of the variable in, 4-94
  - specifying reference index for, 4-95
  - specifying the name of the referenced variable in, 4-95
  - variable, 4-94
- Computed content external, 1-16
- ComputeDefn syntax diagram, B-19
- ConformanceTag syntax diagram, B-35
- Constructed cell reference construct
  - See ExpressionList syntax diagram and CFE expressions, cell-related
- Contains substring construct
  - See ExpressionList syntax diagram and CFE expressions, string
- Contains syntax diagram
  - See ExpressionList syntax diagram
- Content
  - computed
    - specifying attributes for, 4-92 to 4-96
  - indicating relational position of a segment of, 4-111
  - presentation styles for, 4-78
  - restricted
    - external, 4-21 to 4-22
    - specifying alternate presentation string for, 4-101
    - specifying character horizontal alignment point for, 4-107
    - specifying character vertical alignment point for, 4-108
    - specifying general character set for, 4-32
    - specifying general layout attributes for, 4-104
    - specifying Latin1 character set for, 4-164
    - specifying magnitude of coordinate system ratio for, 4-99
    - specifying name of the measurement system of, 4-100
    - specifying number of units per inch of, 4-100
    - specifying precision in coordinate system ratio for, 4-99
    - specifying string format of, 4-106
    - specifying the character orientation in, 4-106
    - specifying the layout of, 4-103
    - specifying the name of the coordinate system of, 4-99
    - specifying the string imaging path of, 4-105
    - specifying world coordinate system for, 4-99
    - specifying wrap attributes for, 4-104
  - text
    - general, 4-32
    - Latin1, 4-164
- Content aggregates, defined, 1-5
- Content definition
  - specifying elements in, 4-9

Content definition (Cont.)  
 specifying index into list of external references in, 4-9  
 specifying label of, 4-8  
 specifying label of the referenced segment in, 4-8  
 specifying private data for, 4-9

Content definition aggregate, 4-8 to 4-9  
 See also DDIF\$\_CTD aggregate  
 external reference index item in, 4-9  
 external target item in, 4-8  
 items in, 4-8t  
 label item in, 4-8  
 private data item in, 4-9  
 value item in, 4-9

Content definitions, 1-16

ContentDefnLabel syntax diagram, B-24

ContentDefn syntax diagram, B-24

ContentPrimitive syntax diagram, B-6

Content reference  
 specifying label for, 4-7  
 specifying transformation for, 4-7

Content reference aggregate, 4-7  
 See also DDIF\$\_CRF aggregate  
 items in, 4-7t  
 label item in, 4-7  
 transformation item in, 4-7

ContentReferencePrimitive syntax diagram, B-18

Content references, 1-15, 1-16

ContentReference syntax diagram, B-18

Content syntax diagram, B-6

Content tags  
 See DDIF tags

Contiguous ranges, 5-55  
 See also Ranges

Conversion  
 input formats, 2-1 to 2-3  
 output formats, 2-3 to 2-9

Conversion restrictions  
 DTIF back end, 2-4  
 DTIF front end, 2-2  
 in DDIF back end, 2-3  
 in DDIF front end, 2-1  
 in PostScript back end, 2-5  
 in Text back end, 2-4  
 in Text front end, 2-3

CONVERT AGGREGATE routine, 8-24

CONVERT/DOCUMENT command, 2-11 to 2-12  
 /OPTIONS qualifier, 2-11

CONVERT DOCUMENT routine, 8-28

Converter  
 activating, 8-126  
 calling from an application, 8-13

Converters  
 linking on ULTRIX, 11-2  
 linking on VMS, 11-2

CONVERT POSITION routine, 8-31

CONVERT routine, 8-13

Convert string to date construct  
 See ExpressionList syntax diagram and CFE expressions, convert string to date

Convert string to time construct  
 See ExpressionList syntax diagram and CFE expressions, convert string to time

Convert to value construct  
 See ExpressionList syntax diagram and CFE expressions, conversion

Copied computed content, 4-93

COPY AGGREGATE routine, 8-33

Copying CDA documents  
 on an ULTRIX system, 3-3  
 on a VMS system, 3-3

Cosine construct  
 See ExpressionList syntax diagram and CFE expressions, trigonometric

Cos syntax diagram  
 See ExpressionList syntax diagram

Count-cols syntax diagram  
 See ExpressionList syntax diagram

Count columns construct  
 See ExpressionList syntax diagram and CFE expressions, cell-related

Count construct  
 See ExpressionList syntax diagram and CFE expressions, statistical

Counter  
 specifying style for, 4-11

CounterDefn syntax diagram, B-37

Counter style aggregate, 4-10 to 4-11  
 See also DDIF\$\_CTS aggregate  
 items in, 4-10t  
 style indicator item in, 4-11

CounterStyle syntax diagram, B-38

Count rows construct  
 See ExpressionList syntax diagram and CFE expressions, cell-related

Count-rows syntax diagram  
 See ExpressionList syntax diagram

CREATE AGGREGATE routine, 8-36

CREATE FILE routine, 8-39

CREATE ROOT AGGREGATE routine, 8-45

CREATE STREAM routine, 8-50

CREATE TEXT FILE routine, 8-54

\$CRF tag, defined, 4-91

Cross-reference computed content, 4-95

CrossRef syntax diagram, B-20

CubicBezier  
 See also Bezier curve

CubicBezierPath syntax diagram, B-31

CubicBezier syntax diagram, B-12

Cur-cell syntax diagram  
 See ExpressionList syntax diagram

Cur-col syntax diagram  
 See ExpressionList syntax diagram

Current cell construct  
 See ExpressionList syntax diagram and CFE expressions, cell-related

Current column construct  
 See ExpressionList syntax diagram and CFE expressions, cell-related

Current row construct  
 See ExpressionList syntax diagram and CFE expressions, cell-related

Current value construct  
 See ExpressionList syntax diagram and CFE expressions, variables

Current-value syntax diagram  
 See ExpressionList syntax diagram

Cur-row syntax diagram  
 See ExpressionList syntax diagram

Curve  
 See Bezier curve

Cvt-to-date syntax diagram  
 See ExpressionList syntax diagram  
 Cvt-to-time syntax diagram  
 See ExpressionList syntax diagram  
 Cvt-to-value syntax diagram  
 See ExpressionList syntax diagram

## D

\$DANISH collating sequence, 5–49

### Data

private, 4–74

### Data mapping

DTIF back end, 2–3  
 DTIF front end, 2–2  
 in DDIF back end, 2–3  
 in DDIF front end, 2–1  
 in PostScript back end, 2–5  
 in Text back end, 2–4  
 in Text front end, 2–2

Datatype syntax diagram, C–10

### Date construct

See ExpressionList syntax diagram and CFE expressions, literals

### Date day of the week construct

See ExpressionList syntax diagram and CFE expressions, date/time

\$DATE editstring-name, 5–49

DateFmt syntax diagram, C–13

### Date month name construct

See ExpressionList syntax diagram and CFE expressions, date/time

### Date/time addition constructs

See ExpressionList syntax diagram and CFE expressions, date/time addition

### Date/time difference constructs

See ExpressionList syntax diagram and CFE expressions, date difference

\$DATETIME editstring-name, 5–49

### Date/time extraction constructs

See ExpressionList syntax diagram and CFE expressions, date/time extraction

DateTime syntax diagram, C–14

### Day of the week construct

See ExpressionList syntax diagram and CFE expressions, date/time

\$DB tag, defined, 4–30, 4–90

### DDIF\$\_FTD aggregate

DIF\$\_FTD\_NUMBER item in, 4–25

DDIF\$READ\_format, 8–15

ddif\$write\_format, 8–16

DDIF\$\_AGGREGATE\_TYPE aggregate item, defined, 4–1

### DDIF\$\_ARC aggregate, 4–2 to 4–4

DDIF\$\_ARC\_CENTER\_X item in, 4–3  
 DDIF\$\_ARC\_CENTER\_X\_C item in, 4–3  
 DDIF\$\_ARC\_CENTER\_Y item in, 4–3  
 DDIF\$\_ARC\_CENTER\_Y\_C item in, 4–3  
 DDIF\$\_ARC\_EXTENT item in, 4–4  
 DDIF\$\_ARC\_EXTENT\_C item in, 4–4  
 DDIF\$\_ARC\_FLAGS item in, 4–3  
 DDIF\$\_ARC\_RADIUS\_DELTA\_Y item in, 4–4  
 DDIF\$\_ARC\_RADIUS\_DELTA\_Y\_C item in, 4–4  
 DDIF\$\_ARC\_RADIUS\_X item in, 4–3  
 DDIF\$\_ARC\_RADIUS\_X\_C item in, 4–3  
 DDIF\$\_ARC\_ROTATION item in, 4–4

### DDIF\$\_ARC aggregate (Cont.)

DDIF\$\_ARC\_ROTATION\_C item in, 4–4  
 DDIF\$\_ARC\_START item in, 4–4  
 DDIF\$\_ARC\_START\_C item in, 4–4  
 items in, 4–2t

### DDIF\$\_BEZ aggregate, 4–5 to 4–6

DDIF\$\_BEZ\_FLAGS item in, 4–5  
 DDIF\$\_BEZ\_PATH item in, 4–6  
 DDIF\$\_BEZ\_PATH\_C item in, 4–6  
 items in, 4–5t

### DDIF\$\_CRF aggregate, 4–7

DDIF\$\_CRF\_REFERENCE item in, 4–7  
 DDIF\$\_CRF\_TRANSFORM item in, 4–7  
 items in, 4–7t

### DDIF\$\_CTD aggregate, 4–8 to 4–9

DDIF\$\_CTD\_EXTERNAL\_ERF\_INDEX item in, 4–9  
 DDIF\$\_CTD\_EXTERNAL\_TARGET item in, 4–8  
 DDIF\$\_CTD\_LABEL item in, 4–8  
 DDIF\$\_CTD\_PRIVATE\_DATA item in, 4–9  
 DDIF\$\_CTD\_VALUE item in, 4–9  
 items in, 4–8t

### DDIF\$\_CTS aggregate, 4–10 to 4–11

DDIF\$\_CTS\_STYLE item in, 4–11  
 DDIF\$\_CTS\_STYLE\_C item in, 4–11  
 items in, 4–10t

### DDIF\$\_DDF aggregate, 4–12

DDIF\$\_DDF\_CONTENT item in, 4–12  
 DDIF\$\_DDF\_DESCRIPTOR item in, 4–12  
 DDIF\$\_DDF\_HEADER item in, 4–12

### DDIF\$\_DDIF\$\_SGB aggregate

DDIF\$\_SGB\_CTR\_TRIGGER item in, 4–156

### DDIF\$\_DHD aggregate, 4–13

DDIF\$\_DHD\_AUTHOR item in, 4–14  
 DDIF\$\_DHD\_CONFORMANCE\_TAGS item in, 4–14  
 DDIF\$\_DHD\_DATE item in, 4–14  
 DDIF\$\_DHD\_EXTERNAL\_REFERENCES item in, 4–14

DDIF\$\_DHD\_LANGUAGES item in, 4–15  
 DDIF\$\_DHD\_LANGUAGES\_C item in, 4–14  
 DDIF\$\_DHD\_PRIVATE\_DATA item in, 4–13  
 DDIF\$\_DHD\_STYLE\_GUIDE item in, 4–15  
 DDIF\$\_DHD\_TITLE item in, 4–13  
 DDIF\$\_DHD\_VERSION item in, 4–14

### DDIF\$\_DSC aggregate, 4–16 to 4–17

DDIF\$\_DSC\_MAJOR\_VERSION item in, 4–16  
 DDIF\$\_DSC\_MINOR\_VERSION item in, 4–16  
 DDIF\$\_DSC\_PRODUCT\_IDENTIFIER item in, 4–17  
 DDIF\$\_DSC\_PRODUCT\_NAME item in, 4–17

### DDIF\$\_ERF aggregate, 4–18 to 4–20

DDIF\$\_ERF\_CONTROL item in, 4–20  
 DDIF\$\_ERF\_DATA\_TYPE item in, 4–18  
 DDIF\$\_ERF\_DESCRIPTOR item in, 4–18  
 DDIF\$\_ERF\_LABEL item in, 4–18  
 DDIF\$\_ERF\_LABEL\_TYPE item in, 4–18  
 items in, 4–18t

### DDIF\$\_EXT aggregate, 4–21 to 4–22

DDIF\$\_EXT\_DATA\_VALUE\_DESCRIPTOR item in, 4–21  
 DDIF\$\_EXT\_DIRECT\_REFERENCE item in, 4–21  
 DDIF\$\_EXT\_ENCODING item in, 4–22  
 DDIF\$\_EXT\_ENCODING\_C item in, 4–22  
 DDIF\$\_EXT\_ENCODING\_L item in, 4–22  
 DDIF\$\_EXT\_INDIRECT\_REFERENCE item in, 4–21

DDIF\$\_EXT aggregate (Cont.)  
 items in, 4-21t

DDIF\$ \_FAS aggregate, 4-23 to 4-24  
 DDIF\$ \_FAS\_FLAGS item in, 4-23  
 DDIF\$ \_FAS\_PATH item in, 4-24  
 items in, 4-23t

DDIF\$ \_FTD aggregate, 4-25  
 DDIF\$ \_FTD\_IDENTIFIER item in, 4-25  
 DDIF\$ \_FTD\_PRIVATE\_DATA item in, 4-25  
 items in, 4-25t

DDIF\$ \_GLA aggregate, 4-26 to 4-27  
 DDIF\$ \_GLA\_BOTTOM\_MARGIN item in, 4-27  
 DDIF\$ \_GLA\_BOTTOM\_MARGIN\_C item in, 4-27  
 DDIF\$ \_GLA\_LEFT\_MARGIN item in, 4-27  
 DDIF\$ \_GLA\_LEFT\_MARGIN\_C item in, 4-26  
 DDIF\$ \_GLA\_RIGHT\_MARGIN item in, 4-27  
 DDIF\$ \_GLA\_RIGHT\_MARGIN\_C item in, 4-27  
 DDIF\$ \_GLA\_TOP\_MARGIN item in, 4-26  
 DDIF\$ \_GLA\_TOP\_MARGIN\_C item in, 4-26  
 items in, 4-26t

DDIF\$ \_GLY aggregate, 4-28 to 4-31  
 DDIF\$ \_GLY\_BOUNDING\_BOX\_LL\_X item in,  
 4-29  
 DDIF\$ \_GLY\_BOUNDING\_BOX\_LL\_X\_C item in,  
 4-29  
 DDIF\$ \_GLY\_BOUNDING\_BOX\_LL\_Y item in,  
 4-29  
 DDIF\$ \_GLY\_BOUNDING\_BOX\_LL\_Y\_C item in,  
 4-29  
 DDIF\$ \_GLY\_BOUNDING\_BOX\_UR\_X item in,  
 4-29  
 DDIF\$ \_GLY\_BOUNDING\_BOX\_UR\_X\_C item in,  
 4-29  
 DDIF\$ \_GLY\_BOUNDING\_BOX\_UR\_Y item in,  
 4-30  
 DDIF\$ \_GLY\_BOUNDING\_BOX\_UR\_Y\_C item in,  
 4-29  
 DDIF\$ \_GLY\_FLAGS item in, 4-30  
 DDIF\$ \_GLY\_ID item in, 4-29  
 DDIF\$ \_GLY\_OUTLINE item in, 4-30  
 DDIF\$ \_GLY\_STREAMS item in, 4-30  
 DDIF\$ \_GLY\_SUCCESSOR item in, 4-31  
 DDIF\$ \_GLY\_SUCCESSOR\_C item in, 4-31  
 items in, 4-28t

DDIF\$ \_GTX aggregate, 4-32  
 DDIF\$ \_GTX\_CONTENT item in, 4-32

DDIF\$ \_HRD aggregate, 4-34  
 DDIF\$ \_HRD\_DIRECTIVE item in, 4-35

DDIF\$ \_HRV aggregate, 4-36 to 4-37  
 DDIF\$ \_HRV\_C item in, 4-37  
 DDIF\$ \_HRV\_ESC\_CONSTANT item in, 4-37  
 DDIF\$ \_HRV\_ESC\_CONSTANT\_C item in, 4-37  
 DDIF\$ \_HRV\_ESC\_RATIO\_D item in, 4-37  
 DDIF\$ \_HRV\_ESC\_RATIO\_N item in, 4-37  
 DDIF\$ \_HRV\_RESET\_VALUE item in, 4-37  
 DDIF\$ \_HRV\_RESET\_VALUE\_C item in, 4-37  
 DDIF\$ \_HRV\_RESET\_VARIABLE item in, 4-37

DDIF\$ \_IDU aggregate, 4-38 to 4-40  
 DDIF\$ \_IDU\_BITS\_PER\_PIXEL item in, 4-40  
 DDIF\$ \_IDU\_COMPRESSION\_PARAMS item in,  
 4-39  
 DDIF\$ \_IDU\_COMPRESSION\_TYPE item in, 4-39  
 DDIF\$ \_IDU\_DATA\_OFFSET item in, 4-39  
 DDIF\$ \_IDU\_NUMBER\_OF\_LINES item in, 4-39  
 DDIF\$ \_IDU\_PIXELS\_PER\_LINE item in, 4-38  
 DDIF\$ \_IDU\_PIXEL\_ORDER item in, 4-40  
 DDIF\$ \_IDU\_PIXEL\_STRIDE item in, 4-39

DDIF\$ \_IDU aggregate (Cont.)  
 DDIF\$ \_IDU\_PLANE\_DATA item in, 4-40  
 DDIF\$ \_IDU\_PRIVATE\_CODING\_ATTR item in,  
 4-38  
 DDIF\$ \_IDU\_SCANLINE\_STRIDE item in, 4-39  
 items in, 4-38t

DDIF\$ \_IMG aggregate, 4-41  
 DDIF\$ \_IMG\_CONTENT item in, 4-41  
 items in, 4-41t

DDIF\$ \_LG1 aggregate, 4-42  
 DDIF\$ \_LG1\_PAGE\_DESCRIPTIONS item in, 4-42  
 DDIF\$ \_LG1\_PRIVATE\_DATA item in, 4-42  
 items in, 4-42t

DDIF\$ \_LIN aggregate, 4-43 to 4-44  
 DDIF\$ \_LIN\_DRAW\_PATTERN item in, 4-44  
 DDIF\$ \_LIN\_FLAGS item in, 4-43  
 DDIF\$ \_LIN\_PATH item in, 4-44  
 DDIF\$ \_LIN\_PATH\_C item in, 4-44  
 items in, 4-43t

DDIF\$ \_LL1 aggregate, 4-45 to 4-49  
 DDIF\$ \_LL1\_BREAK\_AFTER item in, 4-47  
 DDIF\$ \_LL1\_BREAK\_BEFORE item in, 4-46  
 DDIF\$ \_LL1\_BREAK\_WITHIN item in, 4-47  
 DDIF\$ \_LL1\_GALLEY\_SELECT item in, 4-46  
 DDIF\$ \_LL1\_INITIAL\_DIRECTIVE item in, 4-46  
 DDIF\$ \_LL1\_INITIAL\_INDENT item in, 4-47  
 DDIF\$ \_LL1\_INITIAL\_INDENT\_C item in, 4-47  
 DDIF\$ \_LL1\_LEADING\_CONSTANT item in, 4-48  
 DDIF\$ \_LL1\_LEADING\_CONSTANT\_C item in,  
 4-48  
 DDIF\$ \_LL1\_LEADING\_RATIO\_D item in, 4-48  
 DDIF\$ \_LL1\_LEADING\_RATIO\_N item in, 4-48  
 DDIF\$ \_LL1\_LEFT\_INDENT item in, 4-47  
 DDIF\$ \_LL1\_LEFT\_INDENT\_C item in, 4-47  
 DDIF\$ \_LL1\_RIGHT\_INDENT item in, 4-48  
 DDIF\$ \_LL1\_RIGHT\_INDENT\_C item in, 4-47  
 DDIF\$ \_LL1\_SPACE\_AFTER item in, 4-48  
 DDIF\$ \_LL1\_SPACE\_AFTER\_C item in, 4-48  
 DDIF\$ \_LL1\_SPACE\_BEFORE item in, 4-48  
 DDIF\$ \_LL1\_SPACE\_BEFORE\_C item in, 4-48  
 DDIF\$ \_LL1\_TAB\_STOPS item in, 4-49  
 items in, 4-45t

DDIF\$ \_LS1 aggregate, 4-50 to 4-51  
 DDIF\$ \_LS1\_LAYOUT item in, 4-51  
 DDIF\$ \_LS1\_LAYOUT\_C item in, 4-50  
 items in, 4-50t

DDIF\$ \_LSD aggregate, 4-52 to 4-53  
 DDIF\$ \_LSD\_NUMBER item in, 4-52  
 DDIF\$ \_LSD\_PATTERN item in, 4-52, 4-129  
 DDIF\$ \_LSD\_PRIVATE\_DATA item in, 4-53  
 items in, 4-52t

DDIF\$ \_LW1 aggregate, 4-54 to 4-56  
 DDIF\$ \_LW1\_HYPHENATION\_FLAGS item in,  
 4-55  
 DDIF\$ \_LW1\_MAXIMUM\_HYPH\_LINES item in,  
 4-56  
 DDIF\$ \_LW1\_MAXIMUM\_ORPHAN\_SIZE item in,  
 4-56  
 DDIF\$ \_LW1\_MAXIMUM\_WIDOW\_SIZE item in,  
 4-56  
 DDIF\$ \_LW1\_QUAD\_FORMAT item in, 4-55  
 DDIF\$ \_LW1\_WRAP\_FORMAT item in, 4-55  
 items in, 4-54t

DDIF\$ \_OCC aggregate, 4-57 to 4-58  
 DDIF\$ \_OCC\_OCCURRENCE\_C item in, 4-57  
 DDIF\$ \_OCC\_STRUCTURE\_ELEMENT item in,  
 4-58

DDIF\$\_OCC aggregate (Cont.)  
 DDIF\$\_OCC\_STRUCTURE\_ELEMENT\_C item in, 4-58  
 items in, 4-57t

DDIF\$\_PGD aggregate, 4-59 to 4-60  
 DDIF\$\_PGD\_DESC item in, 4-60  
 DDIF\$\_PGD\_DESC\_C item in, 4-60  
 DDIF\$\_PGD\_LABEL item in, 4-59  
 DDIF\$\_PGD\_PRIVATE\_DATA item in, 4-59  
 items in, 4-59t

DDIF\$\_PGL aggregate, 4-61 to 4-63  
 DDIF\$\_PGL\_CONTENT item in, 4-63  
 DDIF\$\_PGL\_LAYOUT\_ID item in, 4-62  
 DDIF\$\_PGL\_ORIENTATION item in, 4-63  
 DDIF\$\_PGL\_PROTOTYPE item in, 4-63  
 DDIF\$\_PGL\_SIZE\_X\_NOM item in, 4-62  
 DDIF\$\_PGL\_SIZE\_X\_NOM\_C item in, 4-62  
 DDIF\$\_PGL\_SIZE\_X\_SHR item in, 4-62  
 DDIF\$\_PGL\_SIZE\_X\_SHR\_C item in, 4-62  
 DDIF\$\_PGL\_SIZE\_X\_STR item in, 4-62  
 DDIF\$\_PGL\_SIZE\_X\_STR\_C item in, 4-62  
 DDIF\$\_PGL\_SIZE\_Y\_NOM item in, 4-62  
 DDIF\$\_PGL\_SIZE\_Y\_NOM\_C item in, 4-62  
 DDIF\$\_PGL\_SIZE\_Y\_SHR item in, 4-63  
 DDIF\$\_PGL\_SIZE\_Y\_SHR\_C item in, 4-63  
 DDIF\$\_PGL\_SIZE\_Y\_STR item in, 4-63  
 DDIF\$\_PGL\_SIZE\_Y\_STR\_C item in, 4-62  
 items in, 4-61t

DDIF\$\_PGS aggregate, 4-64 to 4-65  
 DDIF\$\_PGS\_SELECT\_PAGE\_LAYOUT item in, 4-65  
 DDIF\$\_PGS\_SELECT\_PAGE\_LAYOUT\_C item in, 4-65  
 DDIF\$\_PGS\_SIDE\_CRITERIA item in, 4-64  
 items in, 4-64t

DDIF\$\_PHD aggregate, 4-66  
 DDIF\$\_PHD\_DESCRIPTION item in, 4-66  
 DDIF\$\_PHD\_NUMBER item in, 4-66  
 DDIF\$\_PHD\_PRIVATE\_DATA item in, 4-66  
 items in, 4-66t

DDIF\$\_PTD aggregate, 4-67 to 4-69  
 DDIF\$\_PTD\_DEFN\_C item in, 4-68  
 DDIF\$\_PTD\_NUMBER item in, 4-67  
 DDIF\$\_PTD\_PAT\_COLORS item in, 4-69  
 DDIF\$\_PTD\_PAT\_NUMBER item in, 4-68  
 DDIF\$\_PTD\_PRIVATE\_DATA item in, 4-69  
 DDIF\$\_PTD\_RAS\_PATTERN item in, 4-69  
 DDIF\$\_PTD\_SOL\_COLOR\_B item in, 4-68  
 DDIF\$\_PTD\_SOL\_COLOR\_C item in, 4-68  
 DDIF\$\_PTD\_SOL\_COLOR\_G item in, 4-68  
 DDIF\$\_PTD\_SOL\_COLOR\_R item in, 4-68  
 items in, 4-67t

DDIF\$\_PTH aggregate, 4-70 to 4-73  
 DDIF\$\_PTH\_ARC\_CENTER\_X item in, 4-72  
 DDIF\$\_PTH\_ARC\_CENTER\_X\_C item in, 4-72  
 DDIF\$\_PTH\_ARC\_CENTER\_Y item in, 4-72  
 DDIF\$\_PTH\_ARC\_CENTER\_Y\_C item in, 4-72  
 DDIF\$\_PTH\_ARC\_EXTENT item in, 4-73  
 DDIF\$\_PTH\_ARC\_EXTENT\_C item in, 4-73  
 DDIF\$\_PTH\_ARC\_RADIUS\_DELTA\_Y item in, 4-73  
 DDIF\$\_PTH\_ARC\_RADIUS\_DELTA\_Y\_C item in, 4-72  
 DDIF\$\_PTH\_ARC\_RADIUS\_X item in, 4-72  
 DDIF\$\_PTH\_ARC\_RADIUS\_X\_C item in, 4-72  
 DDIF\$\_PTH\_ARC\_ROTATION item in, 4-73  
 DDIF\$\_PTH\_ARC\_ROTATION\_C item in, 4-73

DDIF\$\_PTH aggregate (Cont.)  
 DDIF\$\_PTH\_ARC\_START item in, 4-73  
 DDIF\$\_PTH\_ARC\_START\_C item in, 4-73  
 DDIF\$\_PTH\_BEZ\_PATH item in, 4-72  
 DDIF\$\_PTH\_BEZ\_PATH\_C item in, 4-72  
 DDIF\$\_PTH\_C item in, 4-71  
 DDIF\$\_PTH\_LIN\_PATH item in, 4-71  
 DDIF\$\_PTH\_LIN\_PATH\_C item in, 4-71  
 DDIF\$\_PTH\_REFERENCE item in, 4-73  
 items in, 4-70t

DDIF\$\_PVT aggregate, 4-74 to 4-75  
 DDIF\$\_PVT\_DATA item in, 4-75  
 DDIF\$\_PVT\_DATA\_C item in, 4-75  
 DDIF\$\_PVT\_NAME item in, 4-75  
 DDIF\$\_PVT\_REFERENCE\_ERF\_INDEX item in, 4-75  
 items in, 4-74t

DDIF\$\_RCD aggregate, 4-76  
 DDIF\$\_RCD\_CONTENTS item in, 4-76  
 DDIF\$\_RCD\_TAG item in, 4-76  
 DDIF\$\_RCD\_TYPE item in, 4-76  
 items in, 4-76t

DDIF\$\_RGB aggregate, 4-77  
 DDIF\$\_RGB\_BLUE\_VALUE item in, 4-77  
 DDIF\$\_RGB\_GREEN\_VALUE item in, 4-77  
 DDIF\$\_RGB\_LUT\_INDEX item in, 4-77  
 DDIF\$\_RGB\_RED\_VALUE item in, 4-77  
 items in, 4-77t

DDIF\$\_SEG aggregate, 4-78  
 DDIF\$\_SEG\_CONTENT item in, 4-80  
 DDIF\$\_SEG\_GENERIC\_LAYOUT item in, 4-79  
 DDIF\$\_SEG\_ID item in, 4-78  
 DDIF\$\_SEG\_SEGMENT\_TYPE item in, 4-79  
 DDIF\$\_SEG\_SPECIFIC\_ATTRIBUTES item in, 4-79  
 DDIF\$\_SEG\_SPECIFIC\_LAYOUT item in, 4-79  
 DDIF\$\_SEG\_USER\_LABEL item in, 4-79

DDIF\$\_SFT aggregate, 4-81  
 DDIF\$\_SFT\_DIRECTIVE item in, 4-82

DDIF\$\_SFV aggregate, 4-83 to 4-84  
 DDIF\$\_SFV\_C item in, 4-84  
 DDIF\$\_SFV\_ESC\_CONSTANT item in, 4-84  
 DDIF\$\_SFV\_ESC\_CONSTANT\_C item in, 4-84  
 DDIF\$\_SFV\_ESC\_RATIO\_D item in, 4-84  
 DDIF\$\_SFV\_ESC\_RATIO\_N item in, 4-84  
 DDIF\$\_SFV\_RESET\_VALUE item in, 4-84  
 DDIF\$\_SFV\_RESET\_VALUE\_C item in, 4-84  
 DDIF\$\_SFV\_RESET\_VARIABLE item in, 4-84

DDIF\$\_SGA aggregate, 4-85 to 4-153  
 DDIF\$\_SGA\_ALT\_PRESENTATION item in, 4-101  
 DDIF\$\_SGA\_BINDING\_DEFNS item in, 4-91  
 DDIF\$\_SGA\_COMPUTE\_C item in, 4-92  
 DDIF\$\_SGA\_CONTENT\_CATEGORY item in, 4-90  
 DDIF\$\_SGA\_CONTENT\_DEFNS item in, 4-116  
 DDIF\$\_SGA\_CONTENT\_STREAMS item in, 4-90  
 DDIF\$\_SGA\_CPTCPY\_ERF\_INDEX item in, 4-93  
 DDIF\$\_SGA\_CPTCPY\_TARGET item in, 4-93  
 DDIF\$\_SGA\_CPTFNC\_NAME item in, 4-96  
 DDIF\$\_SGA\_CPTFNC\_PARAMETERS item in, 4-96  
 DDIF\$\_SGA\_CPTVAR\_VARIABLE item in, 4-94  
 DDIF\$\_SGA\_CPTXRF\_ERF\_INDEX item in, 4-95  
 DDIF\$\_SGA\_CPTXRF\_TARGET item in, 4-95  
 DDIF\$\_SGA\_CPTXRF\_VARIABLE item in, 4-95  
 DDIF\$\_SGA\_FONT\_DEFNS item in, 4-112

## DDIF\$\_SGA aggregate (Cont.)

DDIF\$ \_SGA\_FRMFXD\_POSITION\_X item in, 4-147  
 DDIF\$ \_SGA\_FRMFXD\_POSITION\_X\_C item in, 4-147  
 DDIF\$ \_SGA\_FRMFXD\_POSITION\_Y item in, 4-147  
 DDIF\$ \_SGA\_FRMFXD\_POSITION\_Y\_C item in, 4-147  
 DDIF\$ \_SGA\_FRMGly\_HORIZONTAL item in, 4-149  
 DDIF\$ \_SGA\_FRMGly\_VERTICAL item in, 4-149  
 DDIF\$ \_SGA\_FRMINL\_BASE\_OFFSET item in, 4-148  
 DDIF\$ \_SGA\_FRMINL\_BASE\_OFFSET\_C item in, 4-148  
 DDIF\$ \_SGA\_FRMMAR\_BASE\_OFFSET item in, 4-150  
 DDIF\$ \_SGA\_FRMMAR\_BASE\_OFFSET\_C item in, 4-150  
 DDIF\$ \_SGA\_FRMMAR\_HORIZONTAL item in, 4-151  
 DDIF\$ \_SGA\_FRMMAR\_NEAR\_OFFSET item in, 4-150  
 DDIF\$ \_SGA\_FRMMAR\_NEAR\_OFFSET\_C item in, 4-150  
 DDIF\$ \_SGA\_FRM\_BOX\_LL\_X item in, 4-141  
 DDIF\$ \_SGA\_FRM\_BOX\_LL\_X\_C item in, 4-141  
 DDIF\$ \_SGA\_FRM\_BOX\_LL\_Y item in, 4-141  
 DDIF\$ \_SGA\_FRM\_BOX\_LL\_Y\_C item in, 4-141  
 DDIF\$ \_SGA\_FRM\_BOX\_UR\_X item in, 4-141  
 DDIF\$ \_SGA\_FRM\_BOX\_UR\_X\_C item in, 4-141  
 DDIF\$ \_SGA\_FRM\_BOX\_UR\_Y item in, 4-142  
 DDIF\$ \_SGA\_FRM\_BOX\_UR\_Y\_C item in, 4-141  
 DDIF\$ \_SGA\_FRM\_CLIPPING item in, 4-144  
 DDIF\$ \_SGA\_FRM\_FLAGS item in, 4-140  
 DDIF\$ \_SGA\_FRM\_OUTLINE item in, 4-143  
 DDIF\$ \_SGA\_FRM\_POSITION\_C item in, 4-146  
 DDIF\$ \_SGA\_FRM\_TRANSFORM item in, 4-152  
 DDIF\$ \_SGA\_GLY\_ATTRIBUTES item in, 4-133  
 DDIF\$ \_SGA\_IMG\_BITS\_PER\_COMP item in, 4-138  
 DDIF\$ \_SGA\_IMG\_BRT\_POLARITY item in, 4-135  
 DDIF\$ \_SGA\_IMG\_COMP\_SPACE\_ORG item in, 4-137  
 DDIF\$ \_SGA\_IMG\_COMP\_WAVELENGTH item in, 4-136  
 DDIF\$ \_SGA\_IMG\_COMP\_WAVELENGTH\_C item in, 4-136  
 DDIF\$ \_SGA\_IMG\_GRID\_TYPE item in, 4-135  
 DDIF\$ \_SGA\_IMG\_LINE\_PROGRESSION item in, 4-134  
 DDIF\$ \_SGA\_IMG\_LOOKUP\_TABLES item in, 4-136  
 DDIF\$ \_SGA\_IMG\_LOOKUP\_TABLES\_C item in, 4-136  
 DDIF\$ \_SGA\_IMG\_LP\_PIXEL\_DIST item in, 4-134  
 DDIF\$ \_SGA\_IMG\_NUMBER\_OF\_COMP item in, 4-138  
 DDIF\$ \_SGA\_IMG\_PIXEL\_PATH item in, 4-134  
 DDIF\$ \_SGA\_IMG\_PLANES\_PER\_PIXEL item in, 4-138  
 DDIF\$ \_SGA\_IMG\_PLANE\_SIGNIF item in, 4-138  
 DDIF\$ \_SGA\_IMG\_PP\_PIXEL\_DIST item in, 4-134  
 DDIF\$ \_SGA\_IMG\_PRIVATE\_DATA item in, 4-134  
 DDIF\$ \_SGA\_IMG\_SPECTRAL\_MAPPING item in, 4-135

## DDIF\$ \_SGA aggregate (Cont.)

DDIF\$ \_SGA\_ITEM\_CHANGE\_LIST item in, 4-153  
 DDIF\$ \_SGA\_LANGUAGE item in, 4-98  
 DDIF\$ \_SGA\_LAYGLY\_LAYOUT item in, 4-104  
 DDIF\$ \_SGA\_LAYGLY\_WRAP item in, 4-104  
 DDIF\$ \_SGA\_LAYOUT\_C item in, 4-103  
 DDIF\$ \_SGA\_LAYPOS\_TEXT\_POSITION item in, 4-111  
 DDIF\$ \_SGA\_LAYPTH\_FORMAT item in, 4-106  
 DDIF\$ \_SGA\_LAYPTH\_H\_ALIGN item in, 4-107  
 DDIF\$ \_SGA\_LAYPTH\_ORIENTATION item in, 4-106  
 DDIF\$ \_SGA\_LAYPTH\_ORIENTATION\_C item in, 4-106  
 DDIF\$ \_SGA\_LAYPTH\_PATH item in, 4-105  
 DDIF\$ \_SGA\_LAYPTH\_V\_ALIGN item in, 4-108  
 DDIF\$ \_SGA\_LAYREL\_H\_CONSTANT item in, 4-109  
 DDIF\$ \_SGA\_LAYREL\_H\_CONSTANT\_C item in, 4-109  
 DDIF\$ \_SGA\_LAYREL\_H\_RATIO\_D item in, 4-109  
 DDIF\$ \_SGA\_LAYREL\_H\_RATIO\_N item in, 4-109  
 DDIF\$ \_SGA\_LAYREL\_V\_CONSTANT item in, 4-110  
 DDIF\$ \_SGA\_LAYREL\_V\_CONSTANT\_C item in, 4-110  
 DDIF\$ \_SGA\_LAYREL\_V\_RATIO\_D item in, 4-110  
 DDIF\$ \_SGA\_LAYREL\_V\_RATIO\_N item in, 4-109  
 DDIF\$ \_SGA\_LEGEND\_UNIT\_D item in, 4-99  
 DDIF\$ \_SGA\_LEGEND\_UNIT\_NAME item in, 4-99  
 DDIF\$ \_SGA\_LEGEND\_UNIT\_N item in, 4-99  
 DDIF\$ \_SGA\_LINE\_STYLE\_DEFNS item in, 4-115  
 DDIF\$ \_SGA\_LIN\_END\_FINISH item in, 4-130  
 DDIF\$ \_SGA\_LIN\_END\_SIZE item in, 4-130  
 DDIF\$ \_SGA\_LIN\_END\_SIZE\_C item in, 4-130  
 DDIF\$ \_SGA\_LIN\_END\_START item in, 4-129  
 DDIF\$ \_SGA\_LIN\_INTERIOR\_PATTERN item in, 4-130  
 DDIF\$ \_SGA\_LIN\_JOIN item in, 4-130  
 DDIF\$ \_SGA\_LIN\_MASK\_PATTERN item in, 4-129  
 DDIF\$ \_SGA\_LIN\_MITER\_LIMIT\_D item in, 4-130  
 DDIF\$ \_SGA\_LIN\_MITER\_LIMIT\_N item in, 4-130  
 DDIF\$ \_SGA\_LIN\_PATTERN\_SIZE item in, 4-129  
 DDIF\$ \_SGA\_LIN\_PATTERN\_SIZE\_C item in, 4-129  
 DDIF\$ \_SGA\_LIN\_STYLE item in, 4-128  
 DDIF\$ \_SGA\_LIN\_WIDTH item in, 4-128  
 DDIF\$ \_SGA\_LIN\_WIDTH\_C item in, 4-128  
 DDIF\$ \_SGA\_MKR\_MASK\_PATTERN item in, 4-132  
 DDIF\$ \_SGA\_MKR\_SIZE item in, 4-132  
 DDIF\$ \_SGA\_MKR\_SIZE\_C item in, 4-132  
 DDIF\$ \_SGA\_MKR\_STYLE item in, 4-132  
 DDIF\$ \_SGA\_PATH\_DEFNS item in, 4-114  
 DDIF\$ \_SGA\_PATTERN\_DEFNS item in, 4-113  
 DDIF\$ \_SGA\_PRIVATE\_DATA item in, 4-90  
 DDIF\$ \_SGA\_SEGMENT\_TAGS item in, 4-91  
 DDIF\$ \_SGA\_STRUCTURE\_DESC item in, 4-97  
 DDIF\$ \_SGA\_STRUCTURE\_DESC\_C item in, 4-97  
 DDIF\$ \_SGA\_TXT\_DEC\_ALIGNMENT item in, 4-125  
 DDIF\$ \_SGA\_TXT\_DIRECTION item in, 4-124  
 DDIF\$ \_SGA\_TXT\_FONT item in, 4-120  
 DDIF\$ \_SGA\_TXT\_HEIGHT item in, 4-123  
 DDIF\$ \_SGA\_TXT\_HEIGHT\_C item in, 4-123  
 DDIF\$ \_SGA\_TXT\_LEADER\_ALIGN item in, 4-126

- DDIF\$\_SGA aggregate (Cont.)
  - DDIF\$ \_SGA\_TXT\_LEADER\_BULLET item in, 4-126
  - DDIF\$ \_SGA\_TXT\_LEADER\_SPACE item in, 4-126
  - DDIF\$ \_SGA\_TXT\_LEADER\_SPACE\_C item in, 4-126
  - DDIF\$ \_SGA\_TXT\_LEADER\_STYLE item in, 4-126
  - DDIF\$ \_SGA\_TXT\_MASK\_PATTERN item in, 4-119
  - DDIF\$ \_SGA\_TXT\_PAIR\_KERNING item in, 4-127
  - DDIF\$ \_SGA\_TXT\_RENDITION item in, 4-122
  - DDIF\$ \_SGA\_TXT\_SET\_SIZE\_D item in, 4-123
  - DDIF\$ \_SGA\_TXT\_SET\_SIZE\_N item in, 4-123
  - DDIF\$ \_SGA\_TYPE\_DEFNS item in, 4-117
  - DDIF\$ \_SGA\_UNITS\_PER\_MEASURE item in, 4-100
  - DDIF\$ \_SGA\_UNIT\_NAME item in, 4-100
  - items in, 4-86t
- DDIF\$ \_SGB aggregate, 4-154 to 4-159
  - DDIF\$ \_SGB\_COM\_STRING\_EXPR item in, 4-158
  - DDIF\$ \_SGB\_COM\_STRING\_EXPR\_C item in, 4-158
  - DDIF\$ \_SGB\_CTR\_INIT item in, 4-157
  - DDIF\$ \_SGB\_CTR\_INIT\_C item in, 4-157
  - DDIF\$ \_SGB\_CTR\_STYLE item in, 4-157
  - DDIF\$ \_SGB\_CTR\_TRIGGER\_C item in, 4-156
  - DDIF\$ \_SGB\_CTR\_TYPE item in, 4-157
  - DDIF\$ \_SGB\_RCD\_LIST item in, 4-159
  - DDIF\$ \_SGB\_VARIABLE\_NAME item in, 4-154
  - DDIF\$ \_SGB\_VARIABLE\_VALUE\_C item in, 4-155
  - items in, 4-154t
- DDIF\$ \_TBS aggregate, 4-160 to 4-161
  - DDIF\$ \_TBS\_HORIZONTAL\_POSITION item in, 4-160
  - DDIF\$ \_TBS\_HORIZONTAL\_POSITION\_C item in, 4-160
  - DDIF\$ \_TBS\_LEADER item in, 4-161
  - DDIF\$ \_TBS\_TYPE item in, 4-161
  - items in, 4-160t
- DDIF\$ \_TRN aggregate, 4-162 to 4-163
  - DDIF\$ \_TRN\_PARAMETER item in, 4-163
  - DDIF\$ \_TRN\_PARAMETER\_C item in, 4-163
  - items in, 4-162t
- DDIF\$ \_TXT aggregate, 4-164
  - DDIF\$ \_TXT\_CONTENT item in, 4-164
- DDIF\$ \_TYD aggregate, 4-165 to 4-166
  - DDIF\$ \_TYD\_ATTRIBUTES item in, 4-165
  - DDIF\$ \_TYD\_LABEL item in, 4-165
  - DDIF\$ \_TYD\_PARENT item in, 4-165
  - DDIF\$ \_TYD\_PRIVATE\_DATA item in, 4-166
  - items in, 4-165t
- DDIF\$ \_USER\_CONTEXT aggregate item, defined, 4-1
- DDIF (DIGITAL Document Interchange Format)
  - analyzing files encoded in, 2-8
  - VMS RMS support of, F-1
- DDIF aggregate hierarchy, 1-6fig
- DDIF aggregates, list of, 1-5
- DDIF back end, 2-3
  - conversion restrictions, 2-3
  - data mapping in, 2-3
- DDIFDocument syntax diagram, B-5
- DDIF front end, 2-1
  - conversion restrictions, 2-1
  - data mapping in, 2-1
  - document syntax errors in, 2-1
- DDIF front end (Cont.)
  - external file references in, 2-1
- DDIF initial values, 1-9
- DDIF processing options, 1-13
  - attribute inheritance, 1-14
  - discard segments, 1-17
  - evaluate content, 1-15
    - computed content, 1-16
    - content definitions, 1-16
    - content references, 1-16
    - retain definitions, 1-15
- DDIF standard aggregates, 1-5t
- DDIF tags, 1-7
- DDIF-to-Text RMS extension, F-1
- DDIS built-in constructors
  - SEQUENCE, B-2
  - SEQUENCE OF, B-2
- DDIS built-in operators
  - ANY, B-3
  - assignment, B-3
  - CHOICE, B-3
  - comment, B-3
  - DEFAULT, B-3
  - named number, B-3
  - OPTIONAL, B-3
- DDIS built-in primitives
  - BIT STRING, B-1
  - BOOLEAN, B-1
  - EXTERNAL, B-2
  - FLOATING-POINT, B-1
  - INTEGER, B-1
  - NULL, B-1
  - OBJECT IDENTIFIER, B-2
  - OCTET STRING, B-1
- DDIS defined types
  - Character-String, B-4
  - Latin1-String, B-4
  - ObjectDescriptor, B-4
  - Text-String, B-4
- DDIS encoding
  - definition of, 1-2
- Decimal string construct
  - See ExpressionList syntax diagram and CFE expressions, conversion
- Decimal-string syntax diagram
  - See ExpressionList syntax diagram
- DEFAULT built-in operator, defined, B-3
- Default values
  - for CDA, 1-4
- DELETE AGGREGATE routine, 8-58
- DELETE ROOT AGGREGATE routine, 8-60
- Dep-cross syntax diagram
  - See ExpressionList syntax diagram
- Dep-db syntax diagram
  - See ExpressionList syntax diagram
- Dep-ddb syntax diagram
  - See ExpressionList syntax diagram
- Depreciation, declining balance construct
  - See ExpressionList syntax diagram and CFE expressions, financial
- Depreciation, declining balance with crossover to straight line construct
  - See ExpressionList syntax diagram and CFE expressions, financial

Depreciation, straight line construct  
 See ExpressionList syntax diagram and CFE expressions, financial

Depreciation, sum of year's digits construct  
 See ExpressionList syntax diagram and CFE expressions, financial

Depreciation double declining balance construct  
 See ExpressionList syntax diagram and CFE expressions, financial

Dep-sline syntax diagram  
 See ExpressionList syntax diagram

Dep-soyd syntax diagram  
 See ExpressionList syntax diagram

-d format qualifier  
 for the cdoc command, 2-16

Diff-day syntax diagram  
 See ExpressionList syntax diagram

Diff-hour syntax diagram  
 See ExpressionList syntax diagram

Diff-min syntax diagram  
 See ExpressionList syntax diagram

Diff-month syntax diagram  
 See ExpressionList syntax diagram

Diff-sec syntax diagram  
 See ExpressionList syntax diagram

Diff-week syntax diagram  
 See ExpressionList syntax diagram

Diff-year syntax diagram  
 See ExpressionList syntax diagram

DIGITAL Document Interchange Format  
 See DDIF

Directive  
 hard, 4-34  
 hard value, 4-36 to 4-37  
 soft, 4-81  
 soft value, 4-83 to 4-84  
 values for, 4-34t, 4-81t

Directive syntax diagram, B-10

Discard segments processing options, 1-17

Discount construct  
 See ExpressionList syntax diagram and CFE expressions, financial

Disjoint ranges, 5-55  
 See also Ranges

Divide syntax diagram  
 See ExpressionList syntax diagram

Division construct  
 See ExpressionList syntax diagram and CFE expressions, arithmetic

Document  
 creating for output, 8-39  
 describing the encoding of, 4-12  
 distinguishing versions of, 4-14  
 indicating the name of, 4-17  
 reading from a stream, 8-97  
 representing the encoding software of, 4-17  
 returning position in, 8-148  
 returning size of, 8-148  
 specifying external style guide for, 4-15  
 specifying file references in, 4-14  
 specifying parameters for, 4-12  
 specifying private information for, 4-13  
 specifying processing languages in, 4-14  
 specifying processing restrictions for, 4-14

## Document (Cont.)

specifying the author of, 4-14  
 specifying the content of, 4-12  
 specifying the title of, 4-13  
 specifying version date of, 4-14  
 structure of, 4-1  
 testing the compatibility of versions for, 4-16  
 writing, 8-153

Document content aggregate, 4-78  
 See also DDIF\$\_SEG aggregate  
 generic layout item in, 4-79  
 items in, 4-78t  
 segment attribute item in, 4-79  
 segment content item in, 4-80  
 segment identifier item in, 4-78  
 segment type item in, 4-79  
 segment user label item in, 4-79  
 specific layout item in, 4-79

Document descriptor aggregate, 4-16 to 4-17  
 See also DDIF\$\_DSC aggregate  
 items in, 4-16t  
 major version item in, 4-16  
 minor version item in, 4-16  
 product identifier item in, 4-17  
 product name item in, 4-17

DocumentDescriptor syntax diagram, B-5, C-1

Document header aggregate, 4-13  
 See also DDIF\$\_DHD aggregate  
 author item in, 4-14  
 conformance tags item in, 4-14  
 date item in, 4-14  
 external references item in, 4-14  
 items in, 4-13t  
 language item in, 4-15  
 languages indicator item in, 4-14  
 private header data item in, 4-13  
 style guide item in, 4-15  
 title item in, 4-13  
 version item in, 4-14

DocumentHeader syntax diagram, B-6, C-2

Document root segment syntax diagram, B-6

Document scope  
 completing, 8-112  
 entering, 8-62

Document syntax errors  
 in DDIF front end, 2-1  
 in Text front end, 2-3

*domain\$read\_format* entry point routine, 11-10

*domain\$write\_format* entry point, 11-15

\$2D tag, defined, 4-90

DTIF\$K\_MAJOR\_VERSION literal, 5-23

DTIF\$K\_MINOR\_VERSION literal, 5-23

*dtif\$write\_format*, 8-16

DTIF\$\_AGGREGATE\_TYPE aggregate item, 5-1

DTIF\$\_ARD aggregate, 5-2 to 5-3  
 DTIF\$\_ARD\_DESCRIPTION item in, 5-2  
 DTIF\$\_ARD\_ELEM\_TYPE\_SIZE item in, 5-3  
 DTIF\$\_ARD\_ELEM\_TYPE\_SIZE\_C item in, 5-2  
 DTIF\$\_ARD\_VALUES item in, 5-3  
 DTIF\$\_ARD\_X\_DIMENSION item in, 5-3  
 DTIF\$\_ARD\_Y\_DIMENSION item in, 5-3  
 DTIF\$\_ARD\_Z\_DIMENSION item in, 5-3  
 items in, 5-2

DTIF\$\_CAT aggregate, 5-4 to 5-9  
 DTIF\$\_CAT\_APPL\_PRIVATE item in, 5-5  
 DTIF\$\_CAT\_COLUMN\_HDR item in, 5-8



DTIF\$\_CAT aggregate (Cont.)

- DTIF\$\_CAT\_COMPUTED\_BY item in, 5-5
- DTIF\$\_CAT\_DATA\_LENGTH item in, 5-8
- DTIF\$\_CAT\_DATA\_TYPE item in, 5-8
- DTIF\$\_CAT\_DEFAULT\_VALUE item in, 5-6
- DTIF\$\_CAT\_DEFAULT\_VALUE\_C item in, 5-5
- DTIF\$\_CAT\_DESCRIPTION item in, 5-5
- DTIF\$\_CAT\_FLAGS item in, 5-8
- DTIF\$\_CAT\_FORMATS item in, 5-5
- DTIF\$\_CAT\_GENERIC\_REF item in, 5-5
- DTIF\$\_CAT\_ID item in, 5-5
- DTIF\$\_CAT\_MISSING\_VALUE item in, 5-7
- DTIF\$\_CAT\_NAME item in, 5-4
- DTIF\$\_CAT\_QUERY\_NAME item in, 5-7
- DTIF\$\_CAT\_SCALE\_FACTOR item in, 5-8
- items in, 5-4

DTIF\$\_CCD aggregate, 5-10 to 5-11

- DTIF\$\_CCD\_COLUMN item in, 5-10
- DTIF\$\_CCD\_FLAGS item in, 5-10
- DTIF\$\_CCD\_ROW item in, 5-10
- items in, 5-10

DTIF\$\_CFT aggregate, 5-12

- DTIF\$\_CFT\_IMAGINARY\_PART item in, 5-12
- DTIF\$\_CFT\_REAL\_PART item in, 5-12
- items in, 5-12

DTIF\$\_CLD aggregate, 5-13 to 5-17

- DTIF\$\_CLD\_APPL\_PRIVATE item in, 5-15
- DTIF\$\_CLD\_COL\_NUM item in, 5-14
- DTIF\$\_CLD\_DESCRIPTION item in, 5-15
- DTIF\$\_CLD\_FORMATS item in, 5-15
- DTIF\$\_CLD\_FORMULA\_CFE item in, 5-16
- DTIF\$\_CLD\_STATE item in, 5-14
- DTIF\$\_CLD\_VALUE item in, 5-16
- DTIF\$\_CLD\_VALUE\_C item in, 5-15
- items in, 5-14

DTIF\$\_CLR aggregate, 5-18

- DTIF\$\_CLR\_RANGE\_BEGIN item in, 5-18
- DTIF\$\_CLR\_RANGE\_END item in, 5-18
- items in, 5-18

DTIF\$\_COR aggregate, 5-19

- DTIF\$\_COR\_COL\_BEGIN item in, 5-19
- DTIF\$\_COR\_COL\_END item in, 5-19
- items in, 5-19

DTIF\$\_DAT aggregate, 5-20 to 5-21

- DTIF\$\_DAT\_DATETIME item in, 5-20
- DTIF\$\_DAT\_TIME\_DIFF item in, 5-21
- DTIF\$\_DAT\_TIME\_DIFF\_C item in, 5-20
- items in, 5-20

DTIF\$\_DSC aggregate, 5-22 to 5-23

- DTIF\$\_DSC\_ENCODE\_MAJOR\_VERSION item in, 5-23
- DTIF\$\_DSC\_ENCODE\_MINOR\_VERSION, 5-23
- DTIF\$\_DSC\_MAJOR\_VERSION item in, 5-22
- DTIF\$\_DSC\_MINOR\_VERSION item in, 5-22
- DTIF\$\_DSC\_PRODUCT\_IDENTIFIER item in, 5-22
- DTIF\$\_DSC\_PRODUCT\_NAME item in, 5-22
- items in, 5-22

DTIF\$\_DTF aggregate, 5-24

- DTIF\$\_DTF\_DESCRIPTOR item in, 5-24
- DTIF\$\_DTF\_HEADER item in, 5-24
- DTIF\$\_DTF\_TABLES item in, 5-24
- items in, 5-24

DTIF\$\_ERF aggregate, 5-25 to 5-26

- DTIF\$\_ERF\_CONTROL item in, 5-26
- DTIF\$\_ERF\_DATA\_TYPE item in, 5-25
- DTIF\$\_ERF\_DESCRIPTOR item in, 5-25

DTIF\$\_ERF aggregate (Cont.)

- DTIF\$\_ERF\_LABEL item in, 5-25
- DTIF\$\_ERF\_LABEL\_TYPE item in, 5-26
- items in, 5-25

DTIF\$\_EXT aggregate, 5-27 to 5-28

- DTIF\$\_EXT\_DATA\_VALUE\_DESCRIPTOR item in, 5-27
- DTIF\$\_EXT\_DIRECT\_REFERENCE item in, 5-27
- DTIF\$\_EXT\_ENCODING item in, 5-28
- DTIF\$\_EXT\_ENCODING\_C item in, 5-27
- DTIF\$\_EXT\_ENCODING\_L item in, 5-28
- DTIF\$\_EXT\_INDIRECT\_REFERENCE item in, 5-27
- items in, 5-27

DTIF\$\_FMI aggregate, 5-29 to 5-42

- DTIF\$\_FMI\_ALIGNMENT item in, 5-40
- DTIF\$\_FMI\_BORDER item in, 5-40
- DTIF\$\_FMI\_C item in, 5-30
- DTIF\$\_FMI\_DATEDS\_EDITSTR item in, 5-38
- DTIF\$\_FMI\_DATEID\_EDITSTR\_ID item in, 5-38
- DTIF\$\_FMI\_DATSTD\_ORDER item in, 5-37
- DTIF\$\_FMI\_DATSTD\_TYPE item in, 5-37
- DTIF\$\_FMI\_DAT\_C item in, 5-36
- DTIF\$\_FMI\_DIRECTION item in, 5-40
- DTIF\$\_FMI\_FLAGS item in, 5-38
- DTIF\$\_FMI\_LANG\_ID item in, 5-39
- DTIF\$\_FMI\_NUMEDS\_EDITSTR item in, 5-34
- DTIF\$\_FMI\_NUMEID\_EDITSTR\_ID item in, 5-34
- DTIF\$\_FMI\_NUMSTD\_DIGITS item in, 5-34
- DTIF\$\_FMI\_NUMSTD\_FRAC item in, 5-34
- DTIF\$\_FMI\_NUMSTD\_TYPE item in, 5-32
- DTIF\$\_FMI\_NUM\_C item in, 5-31
- DTIF\$\_FMI\_NUM\_DATATYPE item in, 5-31
- DTIF\$\_FMI\_NUM\_RNDTRUNC item in, 5-34
- DTIF\$\_FMI\_TXTEDS\_EDITSTR item in, 5-36
- DTIF\$\_FMI\_TXTEID\_EDITSTR\_ID item in, 5-36
- DTIF\$\_FMI\_TXTSTD\_TYPE item in, 5-35
- DTIF\$\_FMI\_UNIT\_DESC item in, 5-40
- DTIF\$\_FMI\_WIDTH item in, 5-39
- DTIF\$\_FMI\_WINDOW\_ID item in, 5-30
- DTIF\$\_TXT\_C item in, 5-35
- items in, 5-29

DTIF\$\_HDR aggregate, 5-43 to 5-45

- DTIF\$\_HDR\_DATE item in, 5-43
- DTIF\$\_HDR\_EXTERNAL\_REFERENCES item in, 5-44
- DTIF\$\_HDR\_GENERIC\_COLUMNS item in, 5-44
- DTIF\$\_HDR\_LANGUAGES item in, 5-44
- DTIF\$\_HDR\_LANGUAGES\_C item in, 5-44
- DTIF\$\_HDR\_LANGUAGE\_PREF\_TABLES item in, 5-44
- DTIF\$\_HDR\_PRIVATE\_DATA item in, 5-43
- DTIF\$\_HDR\_TITLE item in, 5-43
- items in, 5-43

DTIF\$\_LPT aggregate, 5-46 to 5-50

- DTIF\$\_LPT\_COLLATE\_SEQ item in, 5-49
- DTIF\$\_LPT\_COLLATE\_TABLE item in, 5-49
- DTIF\$\_LPT\_EDITSTRS item in, 5-48
- DTIF\$\_LPT\_ITEMS item in, 5-47
- DTIF\$\_LPT\_LANGUAGE\_INDEX item in, 5-46
- items in, 5-46

DTIF\$\_NES aggregate, 5-51

- DTIF\$\_NES\_DEFN item in, 5-51
- DTIF\$\_NES\_NAME item in, 5-51
- items in, 5-51

DTIF\$\_NMR aggregate, 5-52

- DTIF\$\_NMR\_NAMEDRANGE item in, 5-52

- DTIF\$\_NMR aggregate (Cont.)
  - items in, 5-52
- DTIF\$\_NVL
  - items in, 5-53
- DTIF\$\_NVL aggregate, 5-53 to 5-54
  - DTIF\$\_NVL\_NAME item in, 5-53
  - DTIF\$\_NVL\_VALUE item in, 5-54
  - DTIF\$\_NVL\_VALUE\_C item in, 5-53
- DTIF\$\_READ *format*, 8-15
- DTIF\$\_RNG aggregate, 5-55 to 5-56
  - DTIF\$\_RNG\_NAME item in, 5-55
  - DTIF\$\_RNG\_REGION item in, 5-56
  - DTIF\$\_RNG\_SORT\_KEYNUM item in, 5-56
  - DTIF\$\_RNG\_TYPE item in, 5-55
  - items in, 5-55
- DTIF\$\_ROW aggregate, 5-57 to 5-58
  - DTIF\$\_ROW\_APPL\_PRIVATE item in, 5-57
  - DTIF\$\_ROW\_CELLS item in, 5-58
  - DTIF\$\_ROW\_FLAGS item in, 5-58
  - DTIF\$\_ROW\_FORMATS item in, 5-57
  - DTIF\$\_ROW\_NUM item in, 5-57
  - items in, 5-57
- DTIF\$\_RWR aggregate, 5-59
  - DTIF\$\_RWR\_ROW\_BEGIN item in, 5-59
  - DTIF\$\_RWR\_ROW\_END item in, 5-59
  - items in, 5-59
- DTIF\$\_TBL aggregate, 5-60 to 5-61
  - DTIF\$\_TBL\_APPL\_PRIVATE item in, 5-61
  - DTIF\$\_TBL\_MAX\_COLS item in, 5-60
  - DTIF\$\_TBL\_MAX\_ROWS item in, 5-61
  - DTIF\$\_TBL\_METADATA item in, 5-61
  - DTIF\$\_TBL\_ROWS item in, 5-61
  - DTIF\$\_TBL\_WINDOWS item in, 5-61
  - items in, 5-60
- DTIF\$\_TMD aggregate, 5-62 to 5-64
  - DTIF\$\_TMD\_APPL\_PRIVATE item in, 5-62
  - DTIF\$\_TMD\_COLUMNS item in, 5-63
  - DTIF\$\_TMD\_DEFAULT\_FMTS item in, 5-63
  - DTIF\$\_TMD\_DESCRIPTION item in, 5-62
  - DTIF\$\_TMD\_FLAGS item in, 5-63
  - DTIF\$\_TMD\_ID item in, 5-62
  - DTIF\$\_TMD\_NAME item in, 5-62
  - DTIF\$\_TMD\_RANGES item in, 5-64
  - DTIF\$\_TMD\_SYMBOLS item in, 5-64
  - items in, 5-62
- DTIF\$\_USER\_CONTEXT aggregate item, defined, 5-1
- DTIF\$\_VTX aggregate, 5-65
  - DTIF\$\_VTX\_VTEXT\_LEN item in, 5-65
  - DTIF\$\_VTX\_VTEXT\_STR item in, 5-65
  - items in, 5-65
- DTIF\$\_WND aggregate, 5-66 to 5-68
  - DTIF\$\_WND\_ACTIVE\_LOC item in, 5-67
  - DTIF\$\_WND\_APPL\_PRIVATE item in, 5-66
  - DTIF\$\_WND\_CARDINAL\_NUM item in, 5-67
  - DTIF\$\_WND\_DESCRIPTION item in, 5-67
  - DTIF\$\_WND\_FLAGS item in, 5-67
  - DTIF\$\_WND\_FORMATS item in, 5-67
  - DTIF\$\_WND\_ID item in, 5-66
  - DTIF\$\_WND\_NAME item in, 5-66
  - DTIF\$\_WND\_RANGES item in, 5-67
  - items in, 5-66
- DTIF application private aggregate, 5-27 to 5-28
  - data value descriptor item in, 5-27
  - direct reference item in, 5-27
  - encoding indicator item in, 5-27
  - encoding item in, 5-28
- DTIF application private aggregate (Cont.)
  - encoding length item in, 5-28
  - indirect reference item in, 5-27
  - items in, 5-27
- DTIF back end
  - conversion restrictions, 2-4
  - data mapping, 2-3
  - external file references, 2-4
  - overview, 2-3
- DTIF cell coordinates
  - absolute references to, 5-11
  - relative references to, 5-11
- DTIF cell coordinates aggregate, 5-10 to 5-11
  - column identifier item in, 5-10
  - flags item in, 5-10
  - items in, 5-10
  - row identifier item in, 5-10
- DTIF cell range aggregate, 5-18
  - items in, 5-18
  - range begin item in, 5-18
  - range end item in, 5-18
- DTIF column range aggregate, 5-19
  - column begin item in, 5-19
  - column end item in, 5-19
  - items in, 5-19
- DTIF complex float aggregate, 5-12
  - imaginary portion value item in, 5-12
  - items in, 5-12
  - real portion value item in, 5-12
- DTIF date and time aggregate, 5-20 to 5-21
  - date and time item in, 5-20
  - items in, 5-20
  - time difference item in, 5-20
- DTIF document
  - distinguishing versions of, 5-23
  - encoding edit strings in, 7-4
  - encoding expressions in, 6-53
  - generic aggregate items
    - DTIF\$\_AGGREGATE\_TYPE, 5-1
    - DTIF\$\_USER\_CONTEXT, 5-1
  - representing the encoding software of, 5-23
  - specifying array values, 5-2
  - specifying cell coordinates for, 5-10
  - specifying cell data for, 5-13
  - specifying cell ranges for, 5-18
  - specifying column attributes for, 5-4
  - specifying column ranges for, 5-19
  - specifying complex floating-point values, 5-12
  - specifying date and time values for, 5-20, 7-2
  - specifying external references for, 5-44
  - specifying format information for, 5-29
  - specifying generic attributes for, 5-44
  - specifying language preference tables for, 5-44, 5-46
  - specifying named edit strings for, 5-51
  - specifying named ranges for, 5-52
  - specifying named values for, 5-53, 7-22
  - specifying natural and programming languages for, 5-44
  - specifying private data for, 5-27, 5-43
  - specifying product identifier in, 5-22
  - specifying product name in, 5-22
  - specifying ranges for, 5-55
  - specifying row ranges for, 5-59
  - specifying rows for, 5-57
  - specifying table definitions for, 5-60
  - specifying table metadata for, 5-62

DTIF document (Cont.)  
 specifying table windows for, 5-66  
 specifying the title of, 5-43  
 specifying the version date of, 5-43  
 specifying varying text string values, 5-65

DTIF document aggregate  
 language item in, 5-44  
 product identifier item in, 5-22

DTIF document descriptor aggregate, 5-22 to 5-23  
 encoding major version indicator item in, 5-23  
 encoding minor version indicator item in, 5-23  
 items in, 5-22  
 major version indicator item in, 5-22  
 minor version indicator item in, 5-22  
 product name item in, 5-22

DTIF document header aggregate, 5-43 to 5-45  
 date item in, 5-43  
 external references item in, 5-44  
 generic attributes item in, 5-44  
 items in, 5-43  
 language preference table item in, 5-44  
 languages indicator item in, 5-44  
 private header data item in, 5-43  
 title item in, 5-43

DTIF document root aggregate, 5-24  
 document descriptor item in, 5-24  
 document header item in, 5-24  
 document tables item in, 5-24  
 items in, 5-24

DTIFDocument syntax diagram, C-1

DTIF external reference  
 specifying data type of, 5-25  
 specifying description of the data type of, 5-25  
 specifying label for, 5-25  
 specifying storage system of, 5-26  
 specifying treatment of, 5-26

DTIF external reference aggregate, 5-25 to 5-26  
 control item in, 5-26  
 items in, 5-25  
 reference data type item in, 5-25  
 reference descriptor item in, 5-25  
 reference label item in, 5-25  
 storage item in, 5-26

DTIF front end  
 conversion restrictions, 2-2  
 data mapping, 2-2  
 external file references, 2-2  
 overview, 2-2

DTIF named value aggregate, 5-53 to 5-54  
 items in, 5-53  
 value data item in, 5-53, 5-54  
 value name item in, 5-53

DTIF named values  
 specifying the name for, 5-53  
 specifying the value data for, 5-53

DTIF row range aggregate, 5-59  
 items in, 5-59  
 row begin data item in, 5-59  
 row end data item in, 5-59

DTIF syntax diagrams, C-1 to C-18

DTIF tables  
 See DTIF document  
 See Table definition  
 See Table metadata

DTIF varying text aggregate, 5-65  
 character string data item in, 5-65  
 items in, 5-65

DTIF varying text aggregate (Cont.)  
 text length item in, 5-65

dxvdoc command, 2-18 to 2-19  
 -f format qualifier, 2-18  
 -h paper-height qualifier, 2-19  
 -O options qualifier, 2-19  
 -r qualifier, 2-19  
 -w paper-width qualifier, 2-19

## E

EditStrBuff syntax diagram, E-1

EditStrIndex syntax diagram, C-12

EditString aggregate, 7-4 to 7-19  
 edit string indicator item in, 7-5  
 items in, 7-4  
 major version item in, 7-4  
 minor version item in, 7-4

Edit strings  
 See ESF edit strings

EditString syntax diagram, E-1

Ellipse  
 See Arc

\$EN tag, defined, 4-30, 4-90

ENTER SCOPE routine, 8-62

Enumeration  
 AngleRef, 1-1  
 encoding of, 1-2  
 expression, 1-2  
 measurement, 1-3

Eql syntax diagram  
 See ExpressionList syntax diagram

Equal to construct  
 See ExpressionList syntax diagram and CFE expressions, Boolean and relational

ERASE ITEM routine, 8-76

ERF  
 See External reference; External reference aggregate

Error conditions  
 for cell values, 5-15

Error construct  
 See ExpressionList syntax diagram and CFE expressions, cell-related

EscapementDirective syntax diagram, B-11

Escapement syntax diagram, B-20

Escapement type, B-20

ESF\$\_AGGREGATE\_TYPE aggregate item, 7-1

ESF\$\_DAT aggregate, 7-2 to 7-3  
 ESF\$\_DAT\_DATETIME item in, 7-2  
 ESF\$\_DAT\_TIME\_DIFF item in, 7-3  
 ESF\$\_DAT\_TIME\_DIFF\_C item in, 7-3  
 items in, 7-2

ESF\$\_EDS aggregate, 7-4 to 7-19  
 ESF\$\_EDS\_EDIT\_STRING\_C item in, 7-5  
 ESF\$\_EDS\_MAJOR\_VERSION item in, 7-4  
 ESF\$\_EDS\_MINOR\_VERSION item in, 7-4  
 ESF\$\_EXT\_DIRECT\_REFERENCE item in, 7-20  
 items in, 7-4

ESF\$\_EXT aggregate, 7-20 to 7-21  
 ESF\$\_EXT\_DATA\_VALUE\_DESCRIPTOR item in, 7-20  
 ESF\$\_EXT\_ENCODING\_C item in, 7-20  
 ESF\$\_EXT\_INDIRECT\_REFERENCE item in, 7-20  
 items in, 7-20

- ESF\$\_NVL
  - items in, 7-22
- ESF\$\_NVL aggregate, 7-22 to 7-23
  - ESF\$\_NVL\_NAME item in, 7-22
  - ESF\$\_NVL\_VALUE item in, 7-23
  - ESF\$\_NVL\_VALUE\_C item in, 7-22
- ESF\$\_RPT aggregate, 7-24 to 7-26
  - ESF\$\_RPT\_COUNT item in, 7-24
  - ESF\$\_RPT\_SEQ item in, 7-26
  - ESF\$\_RPT\_SEQ\_C item in, 7-24
  - items in, 7-24
- ESF\$\_TXS aggregate, 7-27
  - ESF\$\_TXS\_TEXT\_STRING item in, 7-27
  - items in, 7-27
- ESF\$\_USER\_CONTEXT aggregate item, 7-1
- ESF application private aggregate, 7-20 to 7-21
  - data value descriptor item in, 7-20
  - direct reference item in, 7-20
  - encoding indicator item in, 7-20
  - indirect-reference item in, 7-20
  - items in, 7-20
- ESF date and time aggregate, 7-2 to 7-3
  - date and time item in, 7-2
  - items in, 7-2
  - time difference item in, 7-3
- ESF edit strings
  - alphabetic, 7-5
  - am-pm, 7-5
  - any case, 7-6
  - any character, 7-6
  - application private, 7-6
  - binary digit, 7-6
  - CURRENCY\_LIT, 7-7
  - day number, 7-7
  - decimal digit, 7-8
  - digit separator, 7-8
  - digit separator literal, 7-9
  - encoded minus, 7-9
  - encoded plus, 7-9
  - encoded sign, 7-10
  - exponent, 7-10
  - float currency, 7-6
  - floating blank suppression, 7-10
  - floating minus, 7-12
  - floating plus, 7-15
  - floating sign, 7-17
  - float zero replace, 7-19
  - fraction second, 7-11
  - hexadecimal digit, 7-11
  - hour 12, 7-11
  - hour 24, 7-11
  - julian digit, 7-11
  - logical character, 7-11
  - long text, 7-12
  - lowercase, 7-12
  - minute, 7-13
  - missing separator, 7-14
  - month name, 7-14
  - month number, 7-14
  - octal digit, 7-14
  - radix point, 7-16
  - radix-point literal, 7-17
  - repeat, 7-17
  - reverse, 7-17
  - second, 7-17
  - string literal, 7-18
  - UPPERCASE, 7-18

- ESF edit strings (Cont.)
  - week day name, 7-18
  - year, 7-19
- ESF generic aggregate items, 7-1
  - ESF\$\_AGGREGATE\_TYPE, 7-1
  - ESF\$\_USER\_CONTEXT, 7-1
- ESF named value aggregate, 7-22 to 7-23
  - items in, 7-22
  - value data item in, 7-22, 7-23
  - value name item in, 7-22
- ESF named values
  - specifying the name for, 7-22
  - specifying the value data for, 7-22
- ESF syntax diagrams, E-1 to E-3
- \$E tag, defined, 4-91
- Exponent construct
  - See ExpressionList syntax diagram and CFE expressions, transcendental
- ExprChoice syntax diagram, D-3
  - See also ExpressionList syntax diagram
- Expression aggregate, 6-53 to 6-54
  - items in, 6-53
  - list item in, 6-53
  - major version item in, 6-53
  - minor version item in, 6-53
- Expression enumeration, 1-2
- Expression list aggregate, 6-9 to 6-52
  - expression item in, 6-52
  - items in, 6-9
- ExpressionList syntax diagram, D-3
- Expressions
  - See CFE expressions
- Expression syntax diagram, B-37, D-2
- Ext-day syntax diagram
  - See ExpressionList syntax diagram
- EXTERNAL built-in primitive, defined, B-2
- External content, 1-15
- External file references
  - DTIF back end, 2-4
  - DTIF front end, 2-2
- External reference
  - identifying data type of, 4-18
  - identifying storage system of, 4-18
  - in DDIF front end, 2-1
  - in Text front end, 2-3
  - processing, 8-126
  - specifying description of the data type of, 4-18
  - specifying label for, 4-18
  - specifying treatment of, 4-20
- External reference aggregate, 4-18 to 4-20
  - See also DDIF\$\_ERF aggregate
  - See DTIF external reference aggregate
  - control item in, 4-20
  - items in, 4-18t
  - reference data type item in, 4-18
  - reference descriptor item in, 4-18
  - reference label item in, 4-18
  - storage item in, 4-18
- External reference computed content, 1-16
- ExternalReference syntax diagram, B-20, C-2
- ExternalRefIndex syntax diagram, B-23, C-3
- External restricted content aggregate, 4-21 to 4-22
  - See also DDIF\$\_EXT aggregate
  - data value descriptor item in, 4-21
  - direct reference item in, 4-21
  - encoding indicator item in, 4-22

External restricted content aggregate (Cont.)

- encoding length item in, 4-22
- indirect reference item in, 4-21
- items in, 4-21t

Ext-hour syntax diagram

- See ExpressionList syntax diagram

Ext-minute syntax diagram

- See ExpressionList syntax diagram

Ext-month syntax diagram

- See ExpressionList syntax diagram

Extract substring left construct

- See ExpressionList syntax diagram and CFE expressions, string

Extract substring right construct

- See ExpressionList syntax diagram and CFE expressions, string

Ext-second syntax diagram

- See ExpressionList syntax diagram

Ext-year syntax diagram

- See ExpressionList syntax diagram

## F

Factorial construct

- See ExpressionList syntax diagram and CFE expressions, transcendental

False construct

- See ExpressionList syntax diagram and CFE expressions, variables

FAX image

- See Image data unit aggregate, compression parameters item in, compression type item in

-f format qualifier

- for the dxvdoc command, 2-18
- for the vdoc command, 2-20

Field in table construct

- See ExpressionList syntax diagram and CFE expressions, choose and lookup

Field reference aggregate, 6-55

- field context item in, 6-55
- field path item in, 6-55
- items in, 6-55

Field-reference construct

- See ExpressionList syntax diagram and CFE expressions, field reference

File

- See also Text file
- closing, 8-6
- creating, 8-39
- opening, 8-130

File tag

- accessing, F-10
- creation of, F-1
- DDIF, F-1
- disposition by COPY command, F-4
- preserving, F-13
- requirement for, F-1
- use of, F-1

Fill area set

- controlling the rendition of, 4-23
- specifying the composite path of, 4-24

Fill area set content aggregate, 4-23 to 4-24

- See also DDIF\$\_FAS aggregate
- flags item in, 4-23
- items in, 4-23t

Fill area set content aggregate (Cont.)

- set path item in, 4-24

FillAreaSet syntax diagram, B-12

FIND DEFINITION routine, 8-79

FIND TRANSFORMATION routine, 8-83

\$FINSWD collating sequence, 5-49

\$FLOAT editstring-name, 5-49

FLOATING-POINT built-in primitive, defined, B-1

Floating-point construct

- See ExpressionList syntax diagram and CFE expressions, literals

Floating-point data type, defined, 1-4

Flush routine, 9-4

FLUSH STREAM routine, 8-86

FmtFlags syntax diagram, C-14

FmtPrec syntax diagram, C-13

\$FN tag, defined, 4-30, 4-90, 4-91

Font definition

- specifying for the defining segment, 4-25
- specifying name for, 4-25
- specifying private data for, 4-25

Font definition aggregate, 4-25

- See also DDIF\$\_FTD aggregate

- identifier item in, 4-25

- items in, 4-25t

- number item in, 4-25

- private data item in, 4-25

FontDefn syntax diagram, B-21

FontNumber syntax diagram, B-29

Foreground

- See pattern definition; Pattern definition aggregate; Image component space

Format information

- generic format attributes, 5-29

- guidelines for storing, 5-41

- inheritance of, 5-29

- specifying a format descriptor for, 5-40

- specifying a format type indicator for, 5-30

- specifying a language preference index in, 5-39

- specifying a numeric data type indicator for, 5-31

- specifying display width for, 5-39

- specifying format direction for, 5-40

- specifying window identifier for, 5-30

Format information aggregate, 5-29 to 5-42

- date format indicator item in, 5-36

- descriptor item in, 5-40

- display width item in, 5-39

- format direction item in, 5-40

- items in, 5-29

- language preference index item in, 5-39

- numeric format indicator item in, 5-31

- specifying a date edit string index for, 5-38

- specifying a date ESF format for, 5-38

- specifying a date standard order for, 5-37

- specifying a numeric edit string index for, 5-34

- specifying a numeric ESF format for, 5-34

- specifying a text edit string index for, 5-36

- specifying a text ESF format for, 5-36

- specifying a text format type indicator for, 5-35

- specifying borders for, 5-40

- specifying digits of precision for, 5-34

- specifying flags for, 5-38

- specifying format alignment for, 5-40

- specifying rounded or truncated values for, 5-34

- specifying standard date types for, 5-37

- specifying standard numeric types for, 5-32

Format information aggregate (Cont.)  
 specifying standard text types for, 5–35  
 specifying the number of fractional digits of precision for, 5–34  
 window identifier item in, 5–30

FormatInfo syntax diagram, C–10

/FORMAT qualifier  
 for the VIEW command, 2–13

Formats  
 See Format information

Format syntax diagram, B–21

FormattingPrimitive syntax diagram, B–10

FormatType syntax diagram, C–11

Frame  
 bounding box, 4–142  
 controlling presentation of, 4–140  
 fixed position, 4–147  
 galley, 4–149  
 inline position, 4–148  
 margin, 4–150  
 specifying a coordinate transformation for, 4–152  
 specifying attributes of, 4–139 to 4–152  
 specifying lower left corner x position of, 4–141  
 specifying lower left corner y position of, 4–141  
 specifying the clipping path of, 4–144  
 specifying the horizontal offset of the base for, 4–150  
 specifying the horizontal positioning of, 4–149  
 specifying the horizontal position of the lower left corner of, 4–151  
 specifying the outline path of, 4–143  
 specifying the vertical offset from the base for, 4–150  
 specifying the vertical offset of the origin of, 4–148  
 specifying the vertical positioning of the lower edge of, 4–149  
 specifying the x position of the origin of, 4–147  
 specifying the y position of the origin of, 4–147  
 specifying upper right corner x position of, 4–141  
 specifying upper right corner y position of, 4–141

FrameParameters syntax diagram, B–21

FRM  
 See Frame

Front end  
 DDIF, 2–1  
 ddif\$read\_format entry point, 11–10 to 11–14  
 dtif\$read\_format entry point, 11–10 to 11–14  
 entry point, 8–15  
 text, 2–2 to 2–3

\$F tag, defined, 4–91

FTD  
 See Font definition; Font definition aggregate

Function computed content, 4–96

FunctionLink syntax diagram, B–23

Functions  
 See CFE expressions

Future value construct  
 See ExpressionList syntax diagram and CFE expressions, financial

Future value of an annuity construct  
 See ExpressionList syntax diagram and CFE expressions, financial

Future value of a single sum construct  
 See ExpressionList syntax diagram and CFE expressions, financial

Fva syntax diagram  
 See ExpressionList syntax diagram

Fvpv syntax diagram  
 See ExpressionList syntax diagram

Fv syntax diagram  
 See ExpressionList syntax diagram

## G

Galley  
 specifying attributes for, 4–133  
 specifying bottom margin for, 4–27  
 specifying left margin for, 4–26  
 specifying right margin for, 4–27  
 specifying top margin for, 4–26

Galley attributes aggregate, 4–26 to 4–27  
 See also DDIF\$\_GLA aggregate  
 galley bottom margin item in, 4–27  
 galley left margin item in, 4–26  
 galley right margin item in, 4–27  
 galley top margin item in, 4–26  
 items in, 4–26t

GalleyAttributes syntax diagram, B–41

Galley-based layout, 4–102, 4–104

GalleyFrameParams syntax diagram, B–22

GalleyLabel syntax diagram, B–25

GalleyVerticalPosition syntax diagram, B–22

Generalized Time universal defined type, defined, B–2

General text content, 4–32

General text content aggregate, 4–32  
 See also DDIF\$\_GTX aggregate  
 text content item in, 4–32

Generic aggregate items  
 CFE\$\_AGGREGATE\_TYPE, 6–1  
 CFE\$\_USER\_CONTEXT, 6–1  
 DDIF\$\_AGGREGATE\_TYPE, 4–1  
 DDIF\$\_USER\_CONTEXT, 4–1  
 DTIF\$\_AGGREGATE\_TYPE, 5–1  
 DTIF\$\_USER\_CONTEXT, 5–1  
 ESF\$\_AGGREGATE\_TYPE, 7–1  
 ESF\$\_USER\_CONTEXT, 7–1

Generic layout  
 specifying descriptions of page templates and rules for, 4–42  
 specifying private data in, 4–42

Generic layout aggregate, 4–42  
 See also DDIF\$\_LG1 aggregate  
 items in, 4–42t  
 page descriptions item in, 4–42  
 private data item in, 4–42

GenericLayout syntax diagram, B–39

GenMeasure syntax diagram, B–42

GenSize syntax diagram, B–43

Geq syntax diagram  
 See ExpressionList syntax diagram

Get-Aggregate entry point, 11–5

GET AGGREGATE routine, 8–88

GET ARRAY SIZE routine, 8–94

GET DOCUMENT routine, 8–97

GET EXTERNAL ENCODING routine, 8–100

Get-Position entry point, 11–8

Get-position routine, 9–8

Get routine, 9–6, 9–7

GET STREAM POSITION routine, 8–103

GET TEXT POSITION routine, 8–106

## GLY

- See Galley; Galley attributes aggregate
- \$GO tag, defined, 4-91
- Graphics
  - controlling interior fill pattern for, 4-130
- Graphics discard option, 1-17
- GraphicsPrimitive syntax diagram, B-11
- Greater than construct
  - See ExpressionList syntax diagram and CFE expressions, Boolean and relational
- Greater than or equal to construct
  - See ExpressionList syntax diagram and CFE expressions, Boolean and relational
- \$GRP tag, defined, 4-91
- Gtr syntax diagram
  - See ExpressionList syntax diagram

## H

- Hard directive, 4-34
  - values for, 4-34t, 4-81t
- Hard directive aggregate, 4-34
  - See also DDIF\$\_HRD aggregate
  - hard directive item in, 4-35
- Hard value directive, 4-36 to 4-37
  - specifying escapement constant for, 4-37
  - specifying escapement ratio denominator for, 4-37
  - specifying escapement ratio numerator for, 4-37
  - specifying new variable value for, 4-37
  - specifying type of, 4-37
  - specifying variable to be reset by, 4-37
- Hard value directive aggregate, 4-36 to 4-37
  - See also DDIF\$\_HRV aggregate
  - directive choice item in, 4-37
  - escapement constant indicator in, 4-37
  - escapement ratio item in, 4-37
  - items in, 4-36t
  - reset value item in, 4-37
  - reset variable item in, 4-37
- Hlookup construct
  - See ExpressionList syntax diagram and CFE expressions, choose and lookup
- h paper-height qualifier
  - for the dxvdoc command, 2-19
  - for the vdoc command, 2-21

## I

- Identification constructs
  - See ExpressionList syntax diagram and CFE expressions, identification
- Identifier construct
  - See ExpressionList syntax diagram and CFE expressions, variables
- If-then-else construct
  - See ExpressionList syntax diagram and CFE expressions, Boolean and relational
- Image
  - specifying application-private lookup tables for, 4-136
  - specifying aspect ratio along the pixel path of, 4-134
  - specifying attributes for, 4-134 to 4-136
  - specifying correlation between physical image data and spectral components of, 4-135
  - specifying direction of scanline capture for, 4-134

## Image (Cont.)

- specifying line progression path aspect ratio for, 4-134
- specifying private data for, 4-134
- specifying the contents of, 4-41
- specifying the direction of pixel capture path for, 4-134
- specifying the physical format of the pixel grid of, 4-135
- specifying the representation of intensity levels in, 4-135
- specifying wavelength information for, 4-136
- ImageAttributes syntax diagram, B-16
- ImageCodingAttrs syntax diagram, B-15
- Image component space
  - specifying attributes for, 4-137 to 4-138
  - specifying number of bits used for each image in, 4-138
  - specifying number of data planes for pixel in, 4-138
  - specifying number of spectral components in, 4-138
  - specifying physical organization of, 4-137
  - specifying significance of data planes in, 4-138
- Image content aggregate, 4-41
  - image content item in, 4-41
  - items in, 4-41t
  - See also DDIF\$\_IMG aggregate, 4-41
- Image data
  - containing parameters for compression of, 4-39
  - indicating compression scheme for a plane of, 4-39
  - specifying actual values of, 4-40
  - specifying distance between pixels in, 4-39
  - specifying distance between scanlines in, 4-39
  - specifying number of pixels per scanline in, 4-38
  - specifying number of scanlines in, 4-39
  - specifying offset to first bit of, 4-39
  - specifying pixel order in, 4-40
  - specifying private data in, 4-38
  - specifying total number of bits per pixel in, 4-40
- Image data unit aggregate, 4-38 to 4-40
  - See also DDIF\$\_IDU aggregate
  - compression parameters item in, 4-39
  - compression type item in, 4-39
  - data offset item in, 4-39
  - items in, 4-38t
  - number of lines item in, 4-39
  - pixel order item in, 4-40
  - pixels per line item in, 4-38
  - pixel stride item in, 4-39
  - plane bits per pixel item in, 4-40
  - plane data item in, 4-40
  - private data item in, 4-38
  - scanline stride item in, 4-39
- Image discard processing option, 1-17
- ImagePrimitive syntax diagram, B-15
- Image resolution, 4-41
  - See Segment attributes aggregate, frame bounding box items
- ImgCmptSpcAttrs syntax diagram, B-17
- ImgLutData syntax diagram, B-17
- Index construct
  - See ExpressionList syntax diagram and CFE expressions, choose and lookup
- Inherit attributes processing option, 1-14

Inherit attributes processing option (Cont.)  
 initial values, 1–14  
 segment binding attributes, 1–14  
 segment initial values, 1–14  
 Initial values, 1–9, 1–14  
 Initial values, defined, 1–14  
 InlineFrameParams syntax diagram, B–21  
 Input formats, 2–1 to 2–3  
 INSERT AGGREGATE routine, 8–108  
 In-table syntax diagram  
 See ExpressionList syntax diagram  
 INTEGER built-in primitive, defined, B–1  
 Integer construct  
 See ExpressionList syntax diagram and CFE expressions, literals  
 \$INTEGER editstring-name, 5–49  
 Integrate construct  
 See ExpressionList syntax diagram and CFE expressions, series  
 Interest construct  
 See ExpressionList syntax diagram and CFE expressions, financial  
 Interest rate construct  
 See ExpressionList syntax diagram and CFE expressions, financial  
 /INTERFACE qualifier  
 for the VIEW command, 2–14  
 Internal rate of return construct  
 See ExpressionList syntax diagram and CFE expressions, financial  
 Int syntax diagram  
 See ExpressionList syntax diagram  
 Irr syntax diagram  
 See ExpressionList syntax diagram  
 Isblank syntax diagram  
 See ExpressionList syntax diagram  
 Isdate syntax diagram  
 See ExpressionList syntax diagram  
 Iserror syntax diagram  
 See ExpressionList syntax diagram  
 Isnot-avail syntax diagram  
 See ExpressionList syntax diagram  
 Isnot-calc syntax diagram  
 See ExpressionList syntax diagram  
 Isnull syntax diagram  
 See ExpressionList syntax diagram  
 Isnumber syntax diagram  
 See ExpressionList syntax diagram  
 Isref syntax diagram  
 See ExpressionList syntax diagram  
 Isstring syntax diagram  
 See ExpressionList syntax diagram  
 \$! tag, defined, 4–90, 4–91  
 Item  
 array-valued, 8–94  
 erasing, 8–76  
 finding definition of, 8–79  
 locating, 8–115  
 writing the contents of, 8–160  
 Item change list, 4–153  
 Item change list, defined, 1–3  
 Item data types, 1–1t  
 \$!X tag, defined, 4–30, 4–90

## K

Kerning  
 definition of, 4–127

## L

Label syntax diagram, B–25  
 Label types syntax diagram, B–24  
 LangPrefIndex syntax diagram, C–11  
 LangPrefTable syntax diagram, C–3  
 LanguageIndex syntax diagram, B–24  
 Language preference table aggregate, 5–46 to 5–50  
 collating sequence item in, 5–49  
 customized collating sequence item in, 5–49  
 data type format item in, 5–48  
 items in, 5–46  
 preference table index in, 5–46  
 presentation attributes item in, 5–47  
 private preference data item in, 5–46  
 Language preference tables  
 specifying an index to, 5–39  
 specifying collating sequence for, 5–49  
 specifying customized collating sequence for, 5–49  
 specifying data type format for, 5–48  
 specifying presentation attributes for, 5–47  
 specifying private data for, 5–46  
 Languages  
 specifying for processing, 4–14  
 Language preference tables  
 specifying the index for, 5–46  
 Latin1-String defined type, defined, B–4  
 Latin1-String syntax diagram, B–4  
 Latin1 text content, 4–164  
 Latin1 text content aggregate, 4–164  
 See also DDIF\$\_TXT aggregate  
 Layout, 4–102  
 forcing new line, galley, or page through, 4–46  
 galley-based, 4–102, 4–104  
 path-based, 4–102, 4–105  
 position-relative, 4–102, 4–109  
 selecting new galley for, 4–46  
 specifying amount of space after a segment in, 4–48  
 specifying amount of space before a segment in, 4–48  
 specifying indentation distance in, 4–47  
 specifying in-segment break condition in, 4–47  
 specifying leading space between lines in, 4–48  
 specifying new left indent in, 4–47  
 specifying new right indent in, 4–47  
 specifying post-segment break condition in, 4–47  
 specifying pre-segment break condition in, 4–46  
 specifying tab stops in, 4–49  
 text-position, 4–102, 4–111  
 Layout attributes aggregate, 4–45 to 4–49  
 See also DDIF\$\_LL1 aggregate  
 galley selection item in, 4–46  
 initial directive item in, 4–46  
 initial indent indicator item in, 4–47  
 in-segment break condition item in, 4–47  
 items in, 4–45t  
 leading ratio item in, 4–48  
 left indent indicator item in, 4–47  
 post-segment break condition item in, 4–47  
 pre-segment break condition item in, 4–46  
 right indent indicator item in, 4–47



- Layout attributes aggregate (Cont.)
  - space-after indicator item in, 4-48
  - space-before indicator item in, 4-48
  - tab stops item in, 4-49
- LayoutAttributes syntax diagram, B-42
- Layout galley
  - specifying bounding box information for, 4-29
  - specifying content streams for, 4-30
  - specifying flag parameters for, 4-30
  - specifying outline path for content in, 4-30
  - specifying reference label for, 4-29
  - specifying text overflow galley type in, 4-31
- Layout galley aggregate, 4-28 to 4-31
  - See also DDIF\$\_GLY aggregate
  - bounding box items for, 4-29
  - flags item in, 4-30
  - galley label item in, 4-29
  - galley outline item in, 4-30
  - galley streams item in, 4-30
  - galley successor item in, 4-31
  - items in, 4-28t
- LayoutGalley syntax diagram, B-40
- LayoutGalley type, B-40
- LayoutObjectType syntax diagram, B-37
- LayoutPrimitive syntax diagram, B-40
- LayoutPrimitive type, B-40
- \$LBL tag, defined, 4-91
- LeaderStyle syntax diagram, B-8
- Least squares construct
  - See ExpressionList syntax diagram and CFE expressions, series
- LEAVE SCOPE routine, 8-112
- Legend
  - See Content
- Legend attributes, 4-99
- LegendUnits syntax diagram, B-25
- Leq syntax diagram
  - See ExpressionList syntax diagram
- Less than construct
  - See ExpressionList syntax diagram and CFE expressions, Boolean and relational
- Less than or equal to construct
  - See ExpressionList syntax diagram and CFE expressions, Boolean and relational
- \$LE tag, defined, 4-91
- Line
  - specifying attributes for, 4-128 to 4-131
  - specifying denominator of miter ratio of, 4-130
  - specifying ending shape of, 4-129
  - specifying ending size of, 4-130
  - specifying mask pattern of, 4-129
  - specifying numerator of miter ratio of, 4-130
  - specifying pattern for, 4-128
  - specifying pattern size of, 4-129
  - specifying shape of joins of, 4-130
  - specifying shape of the endings of, 4-130
  - specifying width of, 4-128
- LineAttributes syntax diagram, B-12
- LineDefn syntax diagram, B-31
- LineEndNumber syntax diagram, B-14
- LineJoin syntax diagram, B-14
- Line-style definition
  - specifying line-style pattern in, 4-52
  - specifying private data for, 4-53
  - specifying reference number for, 4-52
- Line-style definition aggregate, 4-52 to 4-53
  - See also DDIF\$\_LSD aggregate
  - items in, 4-52t
  - line-style number item in, 4-52
  - line-style pattern item in, 4-52
  - line-style private data item in, 4-53
- LineStyleNumber syntax diagram, B-14
- Linking application images
  - on ULTRIX, 8-2
  - on VMS, 8-2
- Linking converter images
  - on ULTRIX, 11-2
  - on VMS, 11-2
- Linking viewer images
  - on ULTRIX, 13-2
  - on VMS, 13-2
- Lit-complex-float syntax diagram
  - See ExpressionList syntax diagram
- Lit-date syntax diagram
  - See ExpressionList syntax diagram
- Lit-false syntax diagram
  - See ExpressionList syntax diagram
- Lit-float syntax diagram
  - See ExpressionList syntax diagram
- Lit-integer syntax diagram
  - See ExpressionList syntax diagram
- Lit-pi syntax diagram
  - See ExpressionList syntax diagram
- Lit-scaled-integer syntax diagram
  - See ExpressionList syntax diagram
- \$LIT tag, defined, 4-91
- Lit-text syntax diagram
  - See ExpressionList syntax diagram
- Lit-true syntax diagram
  - See ExpressionList syntax diagram
- Lit-vtext syntax diagram
  - See ExpressionList syntax diagram
- LOCATE ITEM routine, 8-115
- Log, base 10 construct
  - See ExpressionList syntax diagram and CFE expressions, transcendental
- Log, base e construct
  - See ExpressionList syntax diagram and CFE expressions, transcendental
- Log10 syntax diagram
  - See ExpressionList syntax diagram
- Logest construct
  - See ExpressionList syntax diagram and CFE expressions, series
- Logical AND construct
  - See ExpressionList syntax diagram and CFE expressions, Boolean and relational
- Logical NOT construct
  - See ExpressionList syntax diagram and CFE expressions, Boolean and relational
- Logical OR construct
  - See ExpressionList syntax diagram and CFE expressions, Boolean and relational
- Logn syntax diagram
  - See ExpressionList syntax diagram
- Lookup table entry aggregate, 4-77
  - See also DDIF\$\_RGB aggregate
  - blue value item in, 4-77

Lookup table entry aggregate (Cont.)

green value item in, 4-77

index item in, 4-77

items in, 4-77t

red value item in, 4-77

Lsqr syntax diagram

See ExpressionList syntax diagram

Lss syntax diagram

See ExpressionList syntax diagram

\$L tag, defined, 4-91

## M

Mailing CDA documents

See Transferring CDA documents

MarginFrameParams syntax diagram, B-22

MarginHorizontalPosition syntax diagram, B-23

Marker

specifying attributes for, 4-132

specifying pattern for, 4-132

specifying size for, 4-132

specifying symbol used as, 4-132

MarkerAttributes syntax diagram, B-15

MarkerNumber syntax diagram, B-15, B-29

Matches construct

See ExpressionList syntax diagram and CFE expressions, choose and lookup

Maximum construct

See ExpressionList syntax diagram and CFE expressions, statistical

Max syntax diagram

See ExpressionList syntax diagram

\$MCS collating sequence, 5-49

Measurement

See Segment attributes aggregate, units per measure item in; Legend

Measurement enumeration, 1-3

Measurement imported, 1-16

MeasurementUnits syntax diagram, B-28

Measure syntax diagram, B-26

Messages

CDA\$\_ facility, G-1 to G-6

Minimum construct

See ExpressionList syntax diagram and CFE expressions, statistical

Min syntax diagram

See ExpressionList syntax diagram

Mirr syntax diagram

See ExpressionList syntax diagram

\$MN tag, defined, 4-30, 4-90

Modified internal rate of return construct

See ExpressionList syntax diagram and CFE expressions, financial

Modulo syntax diagram

See ExpressionList syntax diagram

Modulus construct

See ExpressionList syntax diagram and CFE expressions, Boolean and relational

\$MONEY editstring-name, 5-49

Month name construct

See ExpressionList syntax diagram and CFE expressions, date/time

Multiplication construct

See ExpressionList syntax diagram and CFE expressions, arithmetic

Multiply syntax diagram

See ExpressionList syntax diagram

## N

Name-daynum syntax diagram

See ExpressionList syntax diagram

Name-day syntax diagram

See ExpressionList syntax diagram

Named edit string aggregate, 5-51

edit string definition item in, 5-51

edit string name item in, 5-51

items in, 5-51

Named edit strings

\$DATE, 5-49

\$DATETIME, 5-49

\$FLOAT, 5-49

\$INTEGER, 5-49

\$MONEY, 5-49

\$PERCENT, 5-49

\$PHONE, 5-49

specifying the definition of, 5-51

specifying the name for, 5-51

\$TEXT, 5-49

\$TIME, 5-49

NamedEditString syntax diagram, C-4

Named number built-in operator, defined, B-3

Named parameter aggregate, 6-56

items in, 6-56

name item in, 6-56

value item in, 6-56

NamedParameter syntax diagram, D-2

Named range aggregate, 5-52

items in, 5-52

name range data item in, 5-52

Named range construct

See ExpressionList syntax diagram and CFE expressions, variables

Named ranges, 5-56

See also Ranges

Named-range syntax diagram

See ExpressionList syntax diagram

NamedRange syntax diagram, C-18

NamedValueList syntax diagram, B-29, C-15

NamedValue syntax diagram, B-28

NamedValueTag syntax diagram, B-35

Name-monthnum syntax diagram

See ExpressionList syntax diagram

Name-month syntax diagram

See ExpressionList syntax diagram

Negate syntax diagram

See ExpressionList syntax diagram

Negation construct

See ExpressionList syntax diagram and CFE expressions, arithmetic

Neq syntax diagram

See ExpressionList syntax diagram

Net present value construct

See ExpressionList syntax diagram and CFE expressions, financial

NEXT AGGREGATE routine, 8-120

\$NORWEG collating sequence, 5-49

Not available construct  
 See ExpressionList syntax diagram and CFE expressions, cell-related

Not-avail syntax diagram  
 See ExpressionList syntax diagram

Not-calc syntax diagram  
 See ExpressionList syntax diagram

Not calculable construct  
 See ExpressionList syntax diagram and CFE expressions, cell-related

Not equal to construct  
 See ExpressionList syntax diagram and CFE expressions, Boolean and relational

Not syntax diagram  
 See ExpressionList syntax diagram

Now construct  
 See ExpressionList syntax diagram and CFE expressions, date/time literals

/[NO]OUTPUT qualifier  
 for the VIEW command, 2-14

/[NO]OVERRIDE\_FORMAT qualifier  
 for the VIEW command, 2-14

/[NO]PAGE qualifier  
 for the VIEW command, 2-14

Npv syntax diagram  
 See ExpressionList syntax diagram

NULL built-in primitive, defined, B-1

Null construct  
 See ExpressionList syntax diagram and CFE expressions, cell-related

Number of periods given present value construct  
 See ExpressionList syntax diagram and CFE expressions, financial

Number of periods to achieve future value construct  
 See ExpressionList syntax diagram and CFE expressions, financial

NumericFmt syntax diagram, C-12

## O

ObjectDescriptor defined type, defined, B-4

ObjectDescriptor syntax diagram, B-4

Object identifier, defined, 1-3

OBJECT IDENTIFIER built-in primitive, defined, B-2

OBJECT ID TO AGGREGATE TYPE routine, 8-123

Occurrence definition  
 specifying permitted types of, 4-57  
 specifying structure definition in, 4-58

Occurrence definition aggregate, 4-57 to 4-58  
 See also DDIF\$\_OCC aggregate  
 items in, 4-57t  
 occurrence indicator item in, 4-57  
 structure element indicator item in, 4-58

OccurrenceDefn syntax diagram, B-34

OCTET STRING built-in primitive, defined, B-1

One's complement construct  
 See ExpressionList syntax diagram and CFE expressions, binary

Ones-cmp syntax diagram  
 See ExpressionList syntax diagram

-O options qualifier  
 for the cdoc command, 2-17  
 for the dxvdoc command, 2-19  
 for the vdoc command, 2-21

OPEN CONVERTER routine, 8-126

OPEN FILE routine, 8-130

OPEN STREAM routine, 8-138

OPEN TEXT FILE routine, 8-141

OPTIONAL built-in operator, defined, B-3

/OPTIONS qualifier  
 for the CONVERT/DOCUMENT command, 2-11  
 for the VIEW command, 2-14

Or syntax diagram  
 See ExpressionList syntax diagram

Output formats, 2-3 to 2-9

## P

PageDesclabel syntax diagram, B-25

Page description  
 including private data in, 4-59  
 specifying reference label for, 4-59  
 specifying the type of, 4-60

Page description aggregate, 4-59 to 4-60  
 See also DDIF\$\_PGD aggregate  
 indicator item in, 4-60  
 items in, 4-59t  
 label item in, 4-59  
 private data item in, 4-59

Page descriptions language discard option, 1-17

PageDescription syntax diagram, B-39

Page layout  
 specifying frame for, 4-63  
 specifying nominal measure for, 4-62  
 specifying orientation of, 4-63  
 specifying prototype for, 4-63  
 specifying reference label for, 4-62  
 specifying x shrink amount for, 4-62  
 specifying x stretch amount for, 4-62  
 specifying y nominal measurement for, 4-62  
 specifying y shrink amount for, 4-63  
 specifying y stretch amount for, 4-62

Page layout aggregate, 4-61 to 4-63  
 See also DDIF\$\_PGL aggregate  
 content item in, 4-63  
 items in, 4-61t  
 layout identifier item in, 4-62  
 nominal measure indicator item in, 4-62  
 orientation item in, 4-63  
 prototype item in, 4-63  
 x shrink indicator item in, 4-62  
 x stretch indicator item in, 4-62  
 y shrink indicator item in, 4-63  
 y stretch indicator item in, 4-62

PageLayoutLabel syntax diagram, B-25

PageLayout syntax diagram, B-40

Page selection  
 specifying page-side criteria for, 4-64  
 specifying selected layout for, 4-65

Page selection aggregate, 4-64 to 4-65  
 See also DDIF\$\_PGS aggregate  
 items in, 4-64t  
 page-side criteria item in, 4-64  
 select page layout indicator item in, 4-65

PageSet syntax diagram, B-39

Parenthesized construct  
 See ExpressionList syntax diagram and CFE expressions, miscellaneous

Parenthesized expression aggregate, 6-57  
 expression item in, 6-57  
 items in, 6-57

- text string item in, 6–57
- Path-based layout, 4–102, 4–105
- Path definition
  - specifying composite path in, 4–66
  - specifying private data for, 4–66
  - specifying reference number for, 4–66
- Path definition aggregate, 4–66
  - See also DDIF\$\_PHD aggregate
  - description item in, 4–66
  - items in, 4–66t
  - number item in, 4–66
  - private data item in, 4–66
- PathDefn syntax diagram, B–30
- PathNumber syntax diagram, B–29
- Pattern definition
  - selecting as either solid color or standard pattern, 4–68
  - selecting color type for, 4–68
  - specifying blue intensity for, 4–68
  - specifying color map for, 4–69
  - specifying green intensity for, 4–68
  - specifying image data unit for, 4–69
  - specifying private data for, 4–69
  - specifying red intensity for, 4–68
  - specifying reference number for, 4–67
  - specifying standard pattern number for, 4–68
- Pattern definition aggregate, 4–67 to 4–69
  - See also DDIF\$\_PTD aggregate
  - blue intensity item in, 4–68
  - colors item in, 4–69
  - definition indicator item in, 4–68
  - green intensity item in, 4–68
  - items in, 4–67t
  - number item in, 4–67
  - private data item in, 4–69
  - raster-pattern item in, 4–69
  - red intensity item in, 4–68
  - solid color indicator item in, 4–68
  - standard pattern number item in, 4–68
- PatternDefn syntax diagram, B–32
- PatternNumber syntax diagram, B–30
- Payback construct
  - See ExpressionList syntax diagram and CFE expressions, financial
- Payment per period given present value construct
  - See ExpressionList syntax diagram and CFE expressions, financial
- Payment per period to achieve future value construct
  - See ExpressionList syntax diagram and CFE expressions, financial
- \$PDL tag, defined, 4–90
- Percent construct
  - See ExpressionList syntax diagram and CFE expressions, arithmetic
- \$PERCENT editstring-name, 5–49
- Perfv syntax diagram
  - See ExpressionList syntax diagram
- Periods to achieve future value construct
  - See ExpressionList syntax diagram and CFE expressions, financial
- Perpmt syntax diagram
  - See ExpressionList syntax diagram
- Perpv syntax diagram
  - See ExpressionList syntax diagram
- \$PHONE editstring-name, 5–49
- PI construct
  - See ExpressionList syntax diagram and CFE expressions, variables
- Plus-days syntax diagram
  - See ExpressionList syntax diagram
- Plus-hours syntax diagram
  - See ExpressionList syntax diagram
- Plus-mins syntax diagram
  - See ExpressionList syntax diagram
- Plus-months syntax diagram
  - See ExpressionList syntax diagram
- Plus-secs syntax diagram
  - See ExpressionList syntax diagram
- Plus-weeks syntax diagram
  - See ExpressionList syntax diagram
- Plus-years syntax diagram
  - See ExpressionList syntax diagram
- Pmtfv syntax diagram
  - See ExpressionList syntax diagram
- Pmtpv syntax diagram
  - See ExpressionList syntax diagram
- Polarity
  - See Segment attributes aggregate, brightness polarity item in
- Polyline
  - controlling the drawing of line segments of, 4–44
  - controlling the rendition of, 4–43
  - specifying the layout of, 4–44
- Polyline content aggregate, 4–43 to 4–44
  - See also DDIF\$\_LIN aggregate
  - draw pattern item in, 4–44
  - flags item in, 4–43
  - items in, 4–43t
  - line path indicator item, 4–44
- PolyLinePath syntax diagram, B–31
- Polyline syntax diagram, B–11
- Position-relativelayout, 4–102, 4–109
- Position syntax diagram, B–26
- PostScript back end, 2–5 to 2–8
  - conversion restrictions, 2–5
  - data mapping in, 2–5
  - processing options in, 2–6
- Power syntax diagram
  - See ExpressionList syntax diagram
- p qualifier
  - for the vdoc command, 2–21
- Present value of an annuity construct
  - See ExpressionList syntax diagram and CFE expressions, financial
- Present value to achieve future value construct
  - See ExpressionList syntax diagram and CFE expressions, financial
- Primary aggregates, defined, 1–5
- Principal construct
  - See ExpressionList syntax diagram and CFE expressions, financial
- Private collating sequence, 5–49
- Private content aggregate, 4–74 to 4–75
  - See also DDIF\$\_PVT aggregate
  - external reference index item in, 4–75
  - items in, 4–74t
  - value indicator item in, 4–75
  - value name item in, 4–75

- Private data, 4-74
  - See CFE application private aggregate
  - See DTIF application private aggregate
  - See ESF application private aggregate
- PrivateFuncExpr syntax diagram, D-1
- Private-function construct
  - See ExpressionList syntax diagram and CFE expressions, private
- Processing options
  - attribute inheritance, 1-14
    - style guides, 1-15
    - type definitions, 1-14
    - type references, 1-14
  - discard segments, 1-17
  - evaluate content, 1-15
    - computed content, 1-16
    - content definitions, 1-16
    - content references, 1-16
  - for DDIF, 1-13, 8-47, 8-134
  - in PostScript back end, 2-6
  - in Text back end, 2-4
  - retain definitions, 1-15
- PRUNE AGGREGATE routine, 8-145
- PRUNE POSITION routine, 8-148
- \$P tag, defined, 4-91
- PUT AGGREGATE routine, 8-150
- PUT DOCUMENT routine, 8-153
- Put routine, 9-10
- Pva syntax diagram
  - See ExpressionList syntax diagram
- Pvfv syntax diagram
  - See ExpressionList syntax diagram

## R

- Raise to a power construct
  - See ExpressionList syntax diagram and CFE expressions, arithmetic
- Random number construct
  - See ExpressionList syntax diagram and CFE expressions, random number
- Random-u syntax diagram
  - See ExpressionList syntax diagram
- Range definition aggregate, 5-55 to 5-56
  - items in, 5-55
  - range name item in, 5-55
  - range region item in, 5-56
  - range type item in, 5-55
  - sort key item in, 5-56
- RangeDefn syntax diagram, C-17
- Ranges
  - cell ranges, 5-18
  - column ranges, 5-19
  - contiguous, 5-55
  - disjoint, 5-55
  - named ranges, 5-52
  - row ranges, 5-59
  - specifying a name for, 5-55
  - specifying a range region in, 5-56
  - specifying a sort key for, 5-56
  - specifying the type of, 5-55
  - specifying within a table window, 5-67
  - specifying within table metadata, 5-64
- Range syntax diagram, C-17

- Rate syntax diagram
  - See ExpressionList syntax diagram
- Ratio syntax diagram, B-26
- READ TEXT FILE routine, 8-156
- Receiving CDA documents
  - on an ULTRIX system, 3-3
  - on a VMS system, 3-2
- Record definition
  - specifying segments creating instances of, 4-76
  - specifying type identifier of, 4-76
  - specifying variables of, 4-76
- Record definition aggregate, 4-76
  - See also DDIF\$\_RCD aggregate
  - contents item in, 4-76
  - items in, 4-76t
  - tag item in, 4-76
  - type item in, 4-76
- RecordDefn syntax diagram, B-39
- RecordList syntax diagram, B-38
- Reference
  - processing external, 8-126
- Reference syntax diagram, B-32
- REMOVE AGGREGATE routine, 8-158
- RenditionCode syntax diagram, B-8
- Repeat aggregate, 7-24 to 7-26
  - items in, 7-24
  - repeat count item in, 7-24
  - single edit string indicator item in, 7-24
  - single edit string item in, 7-26
- Repeat syntax diagram, E-3
- Resolution
  - See Segment attributes aggregate, frame bounding box items
- Resolution of images, 4-41
- Restricted content
  - external, 4-21 to 4-22
    - describing data value of, 4-21
    - identifying data type of, 4-21
    - indicating encoding of, 4-22
    - specifying encoding length of, 4-22
  - private
    - identifying value of, 4-75
    - indicating type of data in, 4-75
    - specifying external reference index for, 4-75
- RestrictedContent syntax diagram, B-18
- Reverse image
  - See Segment attributes aggregate, brightness polarity item in
- RGB syntax diagram, B-19
- RightAngle syntax diagram, B-26
- Root aggregate, 4-12
  - See DTIF document root aggregate
  - creating, 8-45
  - deleting, 8-60
  - document content item in, 4-12
  - document descriptor item in, 4-12
  - document header item in, 4-12
  - items in, 4-12t
- Root segment, 4-78
- Rotation
  - See Transformation; Transformation aggregate; Path definition
- Round syntax diagram
  - See ExpressionList syntax diagram

Round to the nearest whole number construct  
See ExpressionList syntax diagram and CFE expressions, conversion

#### Routines

*Allocate/Deallocates*, 9–2

*cda\$aggregate\_type\_to\_object\_id*, 8–3

*cda\$close\_file*, 8–6

*cda\$close\_stream*, 8–9

*cda\$close\_text\_file*, 8–11

*cda\$convert*, 8–13

*cda\$convert\_aggregate*, 8–24

*cda\$convert\_document*, 8–28

*cda\$convert\_position*, 8–31

*cda\$copy\_aggregate*, 8–33

*cda\$create\_aggregate*, 8–36

*cda\$create\_file*, 8–39

*cda\$create\_root\_aggregate*, 8–45

*cda\$create\_stream*, 8–50

*cda\$create\_text\_file*, 8–54

*cda\$delete\_aggregate*, 8–58

*cda\$delete\_root\_aggregate*, 8–60

*cda\$enter\_scope*, 8–62

*cda\$erase\_item*, 8–76

*cda\$find\_definition*, 8–79

*cda\$find\_transformation*, 8–83

*cda\$flush\_stream*, 8–86

*cda\$get\_aggregate*, 8–88

*cda\$get\_array\_size*, 8–94

*cda\$get\_document*, 8–97

*cda\$get\_external\_encoding*, 8–100

*cda\$get\_stream\_position*, 8–103

*cda\$get\_text\_position*, 8–106

*cda\$insert\_aggregate*, 8–108

*cda\$leave\_scope*, 8–112

*cda\$locate\_item*, 8–115

*cda\$next\_aggregate*, 8–120

*cda\$object\_id\_to\_aggregate\_type*, 8–123

*cda\$open\_converter*, 8–126

*cda\$open\_file*, 8–130

*cda\$open\_stream*, 8–138

*cda\$open\_text\_file*, 8–141

*cda\$prune\_aggregate*, 8–145

*cda\$prune\_position*, 8–148

*cda\$put\_aggregate*, 8–150

*cda\$put\_document*, 8–153

*cda\$read\_text\_file*, 8–156

*cda\$remove\_aggregate*, 8–158

*cda\$store\_item*, 8–160

*cda\$write\_text\_file*, 8–168

*close* entry point, 11–3

*domain\$read\_format*, 11–10

*domain\$write\_format* entry point, 11–15

*Flush*, 9–4

*Get*, 9–6, 9–7

*Get-Aggregate* entry point, 11–5

*Get-position*, 9–8

*Get\_position*, 11–8

*Put* routine, 9–10

RowDefn syntax diagram, C–6

Row number construct

See ExpressionList syntax diagram and CFE expressions, variables

Row-num syntax diagram

See ExpressionList syntax diagram

RowNum syntax diagram, C–16

Row portion of cell name construct

See ExpressionList syntax diagram and CFE expressions, cell-related

Row range aggregate

See CFE row range aggregate

See DTIF row range aggregate

Row range construct

See ExpressionList syntax diagram and CFE expressions, variables

Row-range syntax diagram

See ExpressionList syntax diagram

RowRange syntax diagram, C–18

Rows

See Table rows

-r qualifier

for the dxvdoc command, 2–19

for the vdoc command, 2–21

## S

Scaled integer construct

See ExpressionList syntax diagram and CFE expressions, literals

Scaling

See Transformation; Transformation aggregate; Legend attributes

Secondary aggregates, defined, 1–5

Segment

binding attributes to, 4–79

identifying changed attributes in, 4–153

indicating category of the content of, 4–90

listing the variables bound to, 4–91

referencing a type definition for, 4–79

root, 4–78

specifying a reference label for, 4–78

specifying available content definitions for, 4–116

specifying available font definitions for, 4–112

specifying available line style definitions for, 4–115

specifying available path definitions for, 4–114

specifying available pattern definitions for, 4–113

specifying available type definitions for, 4–117

specifying content of, 4–80

specifying content streams for, 4–90

specifying generic layout for, 4–79

specifying language for, 4–98

specifying name for, 4–79

specifying private attributes for, 4–90

specifying processing characteristics for, 4–91

specifying specific layout for, 4–79

specifying the type of computed content in, 4–92

Segment attributes

units per measurement, 1–16

Segment attributes aggregate, 4–85 to 4–153

See also DDIF\$\_SGA aggregate

alternate presentation item in, 4–101

bits per component item in, 4–138

brightness polarity item in, 4–135

component space organization item in, 4–137

component wavelength indicator item in, 4–136

computed content indicator item in, 4–92

content category item in, 4–90

content definition item in, 4–116

content streams item in, 4–90

cross-reference index item in, 4–95

cross-reference segment label item in, 4–95

## Segment attributes aggregate (Cont.)

- cross-reference variable label item in, 4-95
- data plane significance item in, 4-138
- data-planes-per-pixel item in, 4-138
- fixed frame position items in, 4-147
- font definition item in, 4-112
- frame bounding box items, 4-141 to 4-142
- frame clipping path item in, 4-144
- frame content transformation item in, 4-152
- frame flags item in, 4-140
- frame outline item in, 4-143
- frame position item in, 4-146
- function name item in, 4-96
- function parameters item in, 4-96
- galley frame items in, 4-149
- galley layout item in, 4-104
- grid type item in, 4-135
- horizontal alignment item in, 4-107
- inline frame items in, 4-148
- item change list item in, 4-153
- items in, 4-86t
- language item in, 4-98
- layout format item in, 4-106
- layout indicator item, 4-103
- layout path item in, 4-105
- legend unit denominator item in, 4-99
- legend unit name item in, 4-99
- legend unit numerator item in, 4-99
- line end finish item in, 4-130
- line end size indicator item in, 4-130
- line end start item in, 4-129
- line interior pattern item in, 4-130
- line joint item in, 4-130
- line mask pattern item in, 4-129
- line pattern size item in, 4-129
- line progression item in, 4-134
- line progression path aspect ratio item in, 4-134
- line style definition item in, 4-115
- line style item in, 4-128
- line width indicator item in, 4-128
- lookup table item in, 4-136
- margin frame items in, 4-150 to 4-151
- marker mask pattern item in, 4-132
- marker size indicator item in, 4-132
- marker style item in, 4-132
- miter limit denominator item in, 4-130
- miter limit numerator item in, 4-130
- number of components item in, 4-138
- path definition item in, 4-114
- path orientation indicator item, 4-106
- pattern definition item in, 4-113
- pixel path aspect ratio item in, 4-134
- pixel path item in, 4-134
- private attributes item in, 4-90
- private data item in, 4-134
- reference index item in, 4-93
- reference target item in, 4-93
- relative horizontal character position item in, 4-109
- relative vertical character position item in, 4-109
- segment binding item in, 4-91
- segment tags item in, 4-91
- spectral component mapping item in, 4-135
- structure attributes items in, 4-97
- text character decimal alignment item in, 4-125
- text direction item in, 4-124
- text font item in, 4-120
- text kerning item in, 4-127

## Segment attributes aggregate (Cont.)

- text leader attribute items in, 4-126
- text mask pattern item in, 4-119
- text position indicator item in, 4-111
- text rendition item in, 4-122
- text size attribute items in, 4-123
- type definition item in, 4-117
- unit name item in, 4-100
- units per measurement item in, 4-100
- variable item in, 4-94
- vertical alignment item in, 4-108
- wrap attributes item in, 4-104
- SegmentAttributes syntax diagram, B-33
- Segment binding
  - specifying computed variable items in, 4-158
  - specifying counter variable items for, 4-156 to 4-157
  - specifying list variable items in, 4-159
  - specifying name of variable being defined in, 4-154
  - specifying type of variable value in, 4-155
- Segment binding aggregate, 4-154 to 4-159
  - See also DDIF\$\_SGB aggregate
  - computed variable items in, 4-158
  - counter variable items in, 4-156 to 4-157
  - items in, 4-154t
  - list variable items in, 4-159
  - variable name item in, 4-154
  - variable value indicator item in, 4-155
- Segment binding attributes, 1-14
- Segment definitions, 1-15
- Segment initial values, 1-14
- SegmentLabel syntax diagram, B-24
- SegmentPrimitive syntax diagram, B-7
- Segment processing, 1-15
- SegmentTag syntax diagram, B-35
- SegTypeDefn syntax diagram, B-33
- Selector list aggregate, 6-61
  - criteria item in, 6-61
  - items in, 6-61
  - selection item in, 6-61
- Sending CDA documents
  - on an ULTRIX system, 3-2
  - on a VMS system, 3-1
- Sequence
  - removing an aggregate from, 8-158
- SEQUENCE built-in constructor, defined, B-2
- SEQUENCE OF built-in constructor, defined, B-2
- s format qualifier
  - for the cdoc command, 2-16
- SGA
  - See Segment attributes aggregate
- Sigma construct
  - See ExpressionList syntax diagram and CFE expressions, series
- Sign construct
  - See ExpressionList syntax diagram and CFE expressions, sign
- Sine construct
  - See ExpressionList syntax diagram and CFE expressions, trigonometric
- Single syntax diagram, E-2
- Sin syntax diagram
  - See ExpressionList syntax diagram

## Size

See Measurement enumeration; Transformation;  
Transformation aggregate; Legend attributes

Size syntax diagram, B-26

Soft directive, 4-81

values for, 4-34t, 4-81t

Soft directive aggregate, 4-81

DDIF\$\_SFT aggregate

soft directive item in, 4-82

Soft value directive, 4-83 to 4-84

specifying escapement constant for, 4-84

specifying escapement ratio denominator for, 4-84

specifying escapement ratio numerator for, 4-84

specifying new variable value for, 4-84

specifying type of, 4-84

specifying variable to be reset by, 4-84

Soft value directive aggregate, 4-83 to 4-84

See also DDIF\$\_SFV aggregate

directive choice item in, 4-84

escapement constant indicator in, 4-84

escapement ratio item in, 4-84

items in, 4-83t

reset value item in, 4-84

reset variable item in, 4-84

Sort ranges, 5-56

See also Ranges

\$SPANSH collating sequence, 5-49

Specific layout

specifying type of layout for, 4-50

Specific layout aggregate, 4-50 to 4-51

See also DDIF\$\_LS1 aggregate

items in, 4-50t

layout indicator item in, 4-50

SpecificLayout syntax diagram, B-41

Sqrt syntax diagram

See ExpressionList syntax diagram

Square root construct

See ExpressionList syntax diagram and CFE  
expressions, Boolean and relational

\$S tag, defined, 4-91

Standard deviation construct

See ExpressionList syntax diagram and CFE  
expressions, statistical

StandardPattern syntax diagram, B-32

Starts syntax diagram

See ExpressionList syntax diagram

Stdev syntax diagram

See ExpressionList syntax diagram

StorageSystemTag syntax diagram, B-36, C-3, D-2

Stored semantics file attribute, F-1

See also File tag

STORE ITEM routine, 8-160

Storing format information, 5-41

Str-char syntax diagram

See ExpressionList syntax diagram

Str-code syntax diagram

See ExpressionList syntax diagram

Str-concat syntax diagram

See ExpressionList syntax diagram

Stream

closing, 8-6, 8-9

creating, 8-39, 8-50, 8-130

flushing contents of, 8-86

opening, 8-138

Stream (Cont.)

retrieving position in, 8-103

retrieving size of, 8-103

returning position in, 8-31

returning size of, 8-31

writing a document to, 8-153

writing aggregates to, 8-150

StreamTag syntax diagram, B-36

Str-extract syntax diagram

See ExpressionList syntax diagram

Str-find syntax diagram

See ExpressionList syntax diagram

Str-fixed syntax diagram

See ExpressionList syntax diagram

Str-format syntax diagram

See ExpressionList syntax diagram

String character code construct

See ExpressionList syntax diagram and CFE  
expressions, string

String character construct

See ExpressionList syntax diagram and CFE  
expressions, string

String concatenate construct

See ExpressionList syntax diagram and CFE  
expressions, string

StringExpression syntax diagram, B-38

String extract construct

See ExpressionList syntax diagram and CFE  
expressions, string

String find substring construct

See ExpressionList syntax diagram and CFE  
expressions, string

String fixed construct

See ExpressionList syntax diagram and CFE  
expressions, string

String format aggregate, 6-62

edit string item in, 6-62

items in, 6-62

value item in, 6-62

String format construct

See ExpressionList syntax diagram and CFE  
expressions, string

StringLayout syntax diagram, B-9

String length construct

See ExpressionList syntax diagram and CFE  
expressions, string

String lowercase construct

See ExpressionList syntax diagram and CFE  
expressions, string

String pretty aggregate, 6-63

flags item in, 6-63

items in, 6-63

string item in, 6-63

String pretty construct

See ExpressionList syntax diagram and CFE  
expressions, string

String proper construct

See ExpressionList syntax diagram and CFE  
expressions, string

String repeat construct

See ExpressionList syntax diagram and CFE  
expressions, string



- String replace construct
  - See ExpressionList syntax diagram and CFE expressions, string
- String reverse construct
  - See ExpressionList syntax diagram and CFE expressions, string
- String starts with construct
  - See ExpressionList syntax diagram and CFE expressions, string
- String trim construct
  - See ExpressionList syntax diagram and CFE expressions, string
- String uppercase construct
  - See ExpressionList syntax diagram and CFE expressions, string
- Str-left syntax diagram
  - See ExpressionList syntax diagram
- Str-length syntax diagram
  - See ExpressionList syntax diagram
- Str-lower syntax diagram
  - See ExpressionList syntax diagram
- Str-pretty syntax diagram
  - See ExpressionList syntax diagram
- Str-proper syntax diagram
  - See ExpressionList syntax diagram
- Str-repeat syntax diagram
  - See ExpressionList syntax diagram
- Str-replace syntax diagram
  - See ExpressionList syntax diagram
- Str-reverse syntax diagram
  - See ExpressionList syntax diagram
- Str-right syntax diagram
  - See ExpressionList syntax diagram
- Str-trim syntax diagram
  - See ExpressionList syntax diagram
- Structure attributes
  - specifying legal types of, 4–97
- StructureDefinition syntax diagram, B–34
- StructureElement syntax diagram, B–34
- Str-upper syntax diagram
  - See ExpressionList syntax diagram
- Style guides, 1–15
- Subtraction construct
  - See ExpressionList syntax diagram and CFE expressions, arithmetic
- Subtract syntax diagram
  - See ExpressionList syntax diagram
- Summation construct
  - See ExpressionList syntax diagram and CFE expressions, statistical
- Sum syntax diagram
  - See ExpressionList syntax diagram
- Symbols
  - specifying within table metadata, 5–64
- Syntax diagrams
  - Angle, B–25
  - AngleRef, B–25
  - ApplPrivate, B–5, C–14, E–3
  - Arc, B–12
  - ArcPath, B–30
  - ArrayDefn, C–8
  - ASCIIString, B–25, C–15
  - BeginSegment, B–7
- Syntax diagrams (Cont.)
  - Binding, B–36
  - BoundingBox, B–18
  - BreakCriteria, B–42
  - CategoryTag, B–35
  - CellCoord, C–16
  - CellData, C–6
  - CellRange, C–18
  - CellValue, C–7
  - Character-String, B–5
  - ColAttributes, C–9
  - ColNum, C–16
  - Color, B–19
  - ColRange, C–18
  - ComplexFloat, C–8
  - CompositePath, B–30
  - ComputeDefn, B–19
  - ConformanceTag, B–35
  - ContentDefn, B–24
  - ContentReference, B–18
  - ContentReferencePrimitive, B–18
  - CounterDefn, B–37
  - CounterStyle, B–38
  - CrossRef, B–20
  - CubicBezier, B–12
  - CubicBezierPath, B–31
  - Datatype, C–10
  - DateFmt, C–13
  - DateTime, C–14
  - DDIFDocument, B–5
  - DecimalString, D–9
  - Directive, B–10
  - DocumentDescriptor, B–5, C–1
  - DocumentHeader, B–6, C–2
  - Document root segment, B–6
  - DTIFDocument, C–1
  - EditStrBuff, E–1
  - EditStrIndex, C–12
  - EditString, D–9, E–1
  - Escapement, B–20
  - EscapementDirective, B–11
  - Expression, B–37, D–2
  - ExpressionList, D–3
  - ExternalReference, B–20, C–2
  - ExternalRefIndex, B–23, C–3
  - FieldRef, D–9
  - FillAreaSet, B–12
  - FmtFlags, C–14
  - FmtPrec, C–13
  - FontDefn, B–21
  - FontNumber, B–29
  - Format, B–21
  - FormatInfo, C–10
  - FormattingPrimitive, B–10
  - FormatType, C–11
  - FrameParameters, B–21
  - FunctionLink, B–23
  - GalleyAttributes, B–41
  - GalleyFrameParams, B–22
  - GalleyVerticalPosition, B–22
  - GenericLayout, B–39
  - GenMeasure, B–42
  - GenSize, B–43
  - GraphicsPrimitive, B–11
  - ImageAttributes, B–16
  - ImageCodingAttrs, B–15
  - ImagePrimitive, B–15

## Syntax diagrams (Cont.)

- ImgCmptSpcAttrs, B-17
- ImgLutData, B-17
- InlineFrameParams, B-21
- Label, B-25
- Label types, B-24
- LangPrefIndex, C-11
- LangPrefTable, C-3
- LanguageIndex, B-24
- Latin1-String, B-4
- LayoutAttributes, B-42
- LayoutGalley, B-40
- LayoutObjectType, B-37
- LayoutPrimitive, B-40
- LeaderStyle, B-8
- LegendUnits, B-25
- LineAttributes, B-12
- LineDefn, B-31
- LineEndNumber, B-14
- LineJoin, B-14
- LineStyleNumber, B-14
- MarginFrameParams, B-22
- MarginHorizontalPosition, B-23
- MarkerAttributes, B-15
- MarkerNumber, B-15, B-29
- Measure, B-26
- MeasurementUnits, B-28
- NamedEditString, C-4
- NamedParameter, D-2
- NamedRange, C-18
- NamedValue, B-28
- NamedValueList, B-29, C-15
- NamedValueTag, B-35
- NumericFmt, C-12
- ObjectDescriptor, B-4
- OccurrenceDefn, B-34
- PageDescription, B-39
- PageLayout, B-40
- PageSet, B-39
- ParenthesizedExpr, D-9
- PathDefn, B-30
- PathNumber, B-29
- PatternDefn, B-32
- PatternNumber, B-30
- Polyline, B-11
- PolyLinePath, B-31
- Position, B-26
- PrivateFuncExpr, D-1
- Range, C-17
- RangeDefn, C-17
- Ratio, B-26
- RecordDefn, B-39
- RecordList, B-38
- Reference, B-32
- RenditionCode, B-8
- Repeat, E-3
- RestrictedContent, B-18
- RGB, B-19
- RightAngle, B-26
- RowDefn, C-6
- RowNum, C-16
- RowRange, C-18
- SegmentAttributes, B-33
- SegmentPrimitive, B-7
- SegmentTag, B-35
- SegTypeDefn, B-33
- SelectorList, D-8

## Syntax diagrams (Cont.)

- Single, E-2
- Size, B-26
- SpecificLayout, B-41
- StandardPattern, B-32
- StorageSystemTag, B-36, C-3, D-2
- StreamTag, B-36
- StringExpression, B-38
- StringLayout, B-9
- StructureDefinition, B-34
- StructureElement, B-34
- TableDefn, C-4
- TableMD, C-5
- TabStop, B-43
- TabStopList, B-43
- Tag, B-35
- Text, D-8
- TextAttributes, B-7
- TextFmt, C-13
- TextLayout, B-9
- TextPrimitive, B-7
- Text-String, B-5
- Transformation, B-36
- ValueData, B-28, C-15
- ValueDirective, B-10
- VariableLabel, B-25
- VariableReset, B-11
- VaryingText, C-7, D-8
- WindowDefn, C-5
- WrapAttributes, B-41
- XCoordinate, B-27
- YCoordinate, B-28

## T

### Table construct

- See ExpressionList syntax diagram and CFE expressions, choose and lookup

### Table definition

- specifying descriptive information for, 5-61
- specifying number of columns for, 5-60
- specifying number of rows for, 5-61
- specifying private data for, 5-61
- specifying the rows in, 5-61
- specifying windows for, 5-61

### Table definition aggregate, 5-60 to 5-61

- descriptive table information item in, 5-61
- items in, 5-60
- number of columns item in, 5-60
- number of rows item in, 5-61
- private table data item in, 5-61
- rows item in, 5-61
- window item in, 5-61

### TableDefn syntax diagram, C-4

### Table discard option, 1-17

### TableMd syntax diagram, C-5

### Table metadata

- specifying a table identifier for, 5-62
- specifying a table name for, 5-62
- specifying column attributes for, 5-63
- specifying flags for, 5-63
- specifying formats for, 5-63
- specifying private data for, 5-62
- specifying ranges for, 5-64
- specifying symbols for, 5-64
- specifying table descriptor for, 5-62

### Table metadata aggregate, 5-62 to 5-64

## Table metadata aggregate (Cont.)

- column attributes item in, 5-63
- flags item in, 5-63
- format item in, 5-63
- items in, 5-62
- private table data item in, 5-62
- ranges item in, 5-64
- symbols item in, 5-64
- table descriptor item in, 5-62
- table identifier item in, 5-62
- table name item in, 5-62

## Table row aggregate, 5-57 to 5-58

- cell item in, 5-58
- flags item in, 5-58
- format item in, 5-57
- identifier item in, 5-57
- items in, 5-57
- private row data item in, 5-57

## Table rows

- specifying an identifier for, 5-57
- specifying cells for, 5-58
- specifying flags for, 5-58
- specifying formats for, 5-57
- specifying private data for, 5-57

## Tables

- See Table definition
- See Table metadata

## Table window aggregate, 5-66 to 5-68

- active location item in, 5-67
- cardinal number item in, 5-67
- flags item in, 5-67
- format item in, 5-67
- items in, 5-66
- private window data item in, 5-66
- ranges item in, 5-67
- window descriptor item in, 5-67
- window identifier item in, 5-66
- window name item in, 5-66

## Table windows

- referencing from format information, 5-30
- specifying a cardinal number for, 5-67
- specifying a descriptor item for, 5-67
- specifying a window identifier for, 5-66
- specifying a window name for, 5-66
- specifying flags for, 5-67
- specifying formats for, 5-67
- specifying private data for, 5-66
- specifying ranges in, 5-67
- specifying the active or current cell in, 5-67
- specifying within table definition, 5-61

## Tab stop

- specifying horizontal position of, 4-160
- specifying leader character for, 4-161
- specifying type of alignment for, 4-161

## Tab stop aggregate, 4-160 to 4-161

- See also DDIF\$\_TBS aggregate
- items in, 4-160t
- tab stop horizontal position indicator item in, 4-160
- tab stop leader item in, 4-161
- tab stop type item in, 4-161

## TabStopList syntax diagram, B-43

## TabStop syntax diagram, B-43

## Tag

- See File tag

## Tags

- See ESF edit strings

## Tags (Cont.)

- \$AN, 4-91
- content categories, 1-8
- content processing characteristics, 1-8
- content streams in layout, 1-9
- \$CRF, 4-91
- \$2D, 4-90
- \$DB, 4-30, 4-90
- \$E, 4-91
- \$EN, 4-30, 4-90
- \$F, 4-91
- \$FN, 4-30, 4-90, 4-91
- \$GO, 4-91
- \$GRP, 4-91
- \$I, 4-90, 4-91
- \$IX, 4-30, 4-90
- \$L, 4-91
- \$LBL, 4-91
- \$LE, 4-91
- \$LIT, 4-91
- \$MN, 4-30, 4-90
- \$P, 4-91
- \$PDL, 4-90
- \$S, 4-91
- See Edit string aggregate, 7-4
- \$T, 4-90
- \$TBL, 4-90
- \$TOC, 4-30, 4-90
- \$TTL, 4-91

## Tag syntax diagram, B-35

## Tangent construct

- See ExpressionList syntax diagram and CFE expressions, trigonometric

## Tan syntax diagram

- See ExpressionList syntax diagram

## \$TBL tag, defined, 4-90

## Text

- controlling kerning for, 4-127
- specifying alignment characters for, 4-125
- specifying amount of space used by the leader character in, 4-126
- specifying attributes for, 4-118 to 4-127
- specifying format of lines wrapped by the formatter, 4-55
- specifying format of lines wrapped by the user, 4-55
- specifying leader alignment in, 4-126
- specifying maximum consecutive hyphenated lines of, 4-56
- specifying maximum orphan size of, 4-56
- specifying maximum widow size of, 4-56
- specifying one or more renditions for, 4-121
- specifying pattern and color of glyphs in, 4-119
- specifying rules that affect hyphenation of, 4-55
- specifying string used to fill leader space in, 4-126
- specifying the direction of characters in, 4-124
- specifying the font used for, 4-120
- specifying the height of, 4-123
- specifying the ratio for character widths in, 4-123
- specifying type of leader to use in, 4-126

## Text attribute, 4-118

## TextAttributes syntax diagram, B-7

## Text back end, 2-4 to 2-5

- conversion restrictions, 2-4
- data mapping in, 2-4
- processing options in, 2-4

- Text choice aggregate, 6–64
  - items in, 6–64
  - text type indicator item in, 6–64
- Text content
  - general, 4–32
  - Latin1, 4–164
  - specifying general character set for, 4–32
  - specifying Latin1 character set for, 4–164
- Text content aggregate
  - See also DDIF\$\_TXT aggregate
  - item in, 4–164
- Text discard option, 1–17
- \$TEXT editstring-name, 5–49
- Text file
  - closing, 8–11
  - creating, 8–54
  - opening, 8–141
  - reading a line from, 8–156
  - returning position in, 8–106
  - returning size of, 8–106
  - writing a line to, 8–168
- TextFmt syntax diagram, C–13
- Text front end, 2–2 to 2–3
  - conversion restrictions, 2–3
  - data mapping in, 2–2
  - document syntax errors in, 2–3
  - external file references in, 2–3
- Text kerning, 4–127
- TextLayout syntax diagram, B–9
- Text-position layout, 4–102
- TextPrimitive syntax diagram, B–7
- Text string aggregate, 7–27
  - items in, 7–27
  - text string item in, 7–27
- Text-string construct
  - See ExpressionList syntax diagram and CFE expressions, literals
- Text-String defined type, defined, B–4
- Text-String syntax diagram, B–5
- \$TIME edit-string name, 5–49
- \$TOC tag, defined, 4–30, 4–90
- Today construct
  - See ExpressionList syntax diagram and CFE expressions, date/time literals
- Tomorrow construct
  - See ExpressionList syntax diagram and CFE expressions, date/time literals
- Transferring CDA documents, 3–1
- Transformation
  - returning information about, 8–83
  - specifying type of parameter specified for, 4–163
  - specifying value of the parameter for, 4–163
- Transformation aggregate, 4–162 to 4–163
  - See also DDIF\$\_TRN aggregate
  - items in, 4–162t
  - transformation parameter indicator item in, 4–163
- Transformation syntax diagram, B–36
- Trend construct
  - See ExpressionList syntax diagram and CFE expressions, series
- True construct
  - See ExpressionList syntax diagram and CFE expressions, variables
- Truncate construct
  - See ExpressionList syntax diagram and CFE expressions, conversion

- \$T tag, defined, 4–90
- \$TTL tag, defined, 4–91
- TXT
  - See Text
- Type definition
  - specifying parent for, 4–165
  - specifying private data for, 4–166
  - specifying reference label for, 4–165
  - specifying segment attributes for, 4–165
- Type definition aggregate, 4–165 to 4–166
  - See also DDIF\$\_TYD aggregate
  - attributes item in, 4–165
  - items in, 4–165t
  - label item in, 4–165
  - parent item in, 4–165
  - private data item in, 4–166
- Type definitions, 1–14
- Type references, 1–14

## U

- ULTRIX commands
  - cdoc, 2–16
  - dxvdoc, 2–18
  - vdoc, 2–20
- Unary-plus construct
  - See ExpressionList syntax diagram and CFE expressions, arithmetic
- User routines
  - Allocate/Deallocate*, 9–2
  - Flush*, 9–4
  - Get*, 9–6, 9–7
  - Get-position*, 9–8
  - Put routine*, 9–10

## V

- ValueData syntax diagram, B–28, C–15
- ValueDirective syntax diagram, B–10
- Variable
  - encoding of, 1–4
- Variable computed content, 4–94
- VariableLabel syntax diagram, B–24, B–25
- VariableReset syntax diagram, B–11
- Variance construct
  - See ExpressionList syntax diagram and CFE expressions, statistical
- Var syntax diagram
  - See ExpressionList syntax diagram
- Varying length text construct
  - See ExpressionList syntax diagram and CFE expressions, literals
- Varying text aggregate
  - See CFE varying text aggregate
  - See DTIF varying text aggregate
- VaryingText syntax diagram, C–7
- vdoc command, 2–20 to 2–21
  - f format qualifier, 2–20
  - h paper-height qualifier, 2–21
  - O options qualifier, 2–21
  - p qualifier, 2–21
  - r qualifiter, 2–21
  - w paper-width qualifier, 2–21
- VIEW command, 2–13 to 2–15
  - /FORMAT qualifier, 2–13

**VIEW command (Cont.)**

- /INTERFACE qualifier, 2-14
- /[NO]OUTPUT qualifier, 2-14
- /[NO]OVERRIDE\_FORMAT qualifier, 2-14
- /[NO]PAGE qualifier, 2-14
- /OPTIONS qualifier, 2-14
- /WIDTH qualifier, 2-14

**Viewer images**

- linking on ULTRIX, 13-2
- linking on VMS, 13-2

**Vlookup construct**

- See ExpressionList syntax diagram and CFE expressions, choose and lookup

**VMS**

- support for CDA in, F-1

**VMS commands**

- CONVERT/DOCUMENT, 2-11
- VIEW, 2-13

---

**W**

**/WIDTH qualifier**

- for the VIEW command, 2-14

**WindowDefn syntax diagram, C-5**

**Windows**

- See Table windows

**-w paper-width qualifier**

- for the dxvdoc command, 2-19
- for the vdoc command, 2-21

**Wrap attributes aggregate, 4-54 to 4-56**

- See also DDIF\$\_LW1 aggregate
- hyphenation flags item in, 4-55
- hyphenation lines item in, 4-56
- ifems in, 4-54t

- maximum orphan size item in, 4-56
- maximum widow size item in, 4-56
- quad format item in, 4-55
- wrap format item in, 4-55

**WrapAttributes syntax diagram, B-41**

**WRITE TEXT FILE routine, 8-168**

---

**X**

**XCoordinate syntax diagram, B-27**

**XRF**

- See Cross-reference computed content

---

**Y**

**YCoordinate syntax diagram, B-28**

**Yesterday construct**

- See ExpressionList syntax diagram and CFE expressions, date/time literals



## How to Order Additional Documentation

---

### Technical Support

If you need help deciding which documentation best meets your needs, call 800-343-4040 before placing your electronic, telephone, or direct mail order.

### Electronic Orders

To place an order at the Electronic Store, dial 800-DEC-DEMO (800-332-3366) using a 1200- or 2400-baud modem. If you need assistance using the Electronic Store, call 800-DIGITAL (800-344-4825).

### Telephone and Direct Mail Orders

| <b>Your Location</b>                  | <b>Call</b>  | <b>Contact</b>   |
|---------------------------------------|--------------|--|
| Continental USA,<br>Alaska, or Hawaii | 800-DIGITAL  | Digital Equipment Corporation<br>P.O. Box CS2008<br>Nashua, New Hampshire 03061  |
| Puerto Rico                           | 809-754-7575 | Local Digital subsidiary   |
| Canada                                | 800-267-6215 | Digital Equipment of Canada<br>Attn: DECdirect Operations KAO2/2<br>P.O. Box 13000<br>100 Herzberg Road<br>Kanata, Ontario, Canada K2K 2A6         |
| International                         | _____        | Local Digital subsidiary or<br>approved distributor  |
| Internal <sup>1</sup>                 | _____        | USASSB Order Processing - WMO/E15<br>or<br>U.S. Area Software Supply Business<br>Digital Equipment Corporation<br>Westminster, Massachusetts 01473 |

---

<sup>1</sup>For internal orders, you must submit an Internal Software Order Form (EN-01740-07).





# Reader's Comments

CDA Reference Manual  
Volume 1  
AA-PBD3A-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

| I rate this manual's:                      | Excellent                | Good                     | Fair                     | Poor                     |
|--|--------------------------|--------------------------|--------------------------|--------------------------|
| Accuracy (software works as manual says)   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Completeness (enough information)          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Clarity (easy to understand)               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Organization (structure of subject matter) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Figures (useful)                           | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Examples (useful)                          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Index (ability to find topic)              | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Page layout (easy to find information)     | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

I would like to see more/less \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

What I like best about this manual is \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

What I like least about this manual is \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

I found the following errors in this manual:

| Page  | Description |
|-------|-------------|
| _____ | _____       |
| _____ | _____       |
| _____ | _____       |
| _____ | _____       |

Additional comments or suggestions to improve this manual:  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

I am using **Version** \_\_\_\_\_ of the software this manual describes.

Name/Title \_\_\_\_\_ Dept. \_\_\_\_\_

Company \_\_\_\_\_ Date \_\_\_\_\_

Mailing Address \_\_\_\_\_

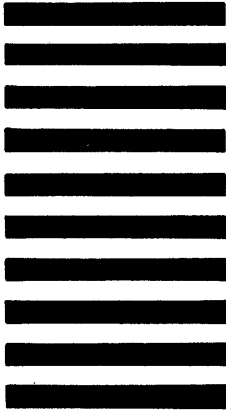
Phone \_\_\_\_\_

Do Not Tear - Fold Here and Tape

**digital**™



No Postage  
Necessary  
if Mailed  
in the  
United States



**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION  
Corporate User Publications—Spit Brook  
ZK01-3/J35  
110 SPIT BROOK ROAD  
NASHUA, NH 03062-9987



Do Not Tear - Fold Here

Cut Along Dotted Line

# Reader's Comments

CDA Reference Manual  
Volume 1  
AA-PBD3A-TE

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

| I rate this manual's:                      | Excellent                | Good                     | Fair                     | Poor                     |
|--|--------------------------|--------------------------|--------------------------|--------------------------|
| Accuracy (software works as manual says)   | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Completeness (enough information)          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Clarity (easy to understand)               | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Organization (structure of subject matter) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Figures (useful)                           | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Examples (useful)                          | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Index (ability to find topic)              | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Page layout (easy to find information)     | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

I would like to see more/less \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

What I like best about this manual is \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

What I like least about this manual is \_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

I found the following errors in this manual:

| Page  | Description |
|-------|-------------|
| _____ | _____       |
| _____ | _____       |
| _____ | _____       |
| _____ | _____       |
| _____ | _____       |

Additional comments or suggestions to improve this manual:  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_  
\_\_\_\_\_

I am using **Version** \_\_\_\_\_ of the software this manual describes.

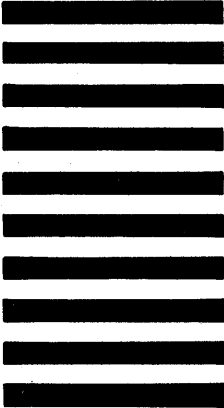
Name/Title \_\_\_\_\_ Dept. \_\_\_\_\_  
Company \_\_\_\_\_ Date \_\_\_\_\_  
Mailing Address \_\_\_\_\_  
\_\_\_\_\_ Phone \_\_\_\_\_

Do Not Tear - Fold Here and Tape

**digital**™



No Postage  
Necessary  
if Mailed  
in the  
United States



**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

DIGITAL EQUIPMENT CORPORATION  
Corporate User Publications—Spit Brook  
ZK01-3/J35  
110 SPIT BROOK ROAD  
NASHUA, NH 03062-9987



Do Not Tear - Fold Here

Cut Along Dotted Line