INTERNATIONAL STANDARD

ISO/IEC 8632-1

Second edition
1999-12-15

Information technology — Computer graphics — Metafile for the storage and transfer of picture description information —

Part 1:
Functional specification

Technologies de l'information — Infographie — Métafichier de stockage et de transfert des informations de description d'images —

Partie 1: Description fonctionnelle

Reference number
ISO/IEC 8632-1:1999(E)

© ISO/IEC 1999

Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

© ISO/IEC 1999

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 734 10 79
E-mail copyright@iso.ch
Web www.iso.ch

Printed in Switzerland

© ISO/IEC 1999 – All rights reserved

Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

Contents

Page

1 Scope ... 1
2 Conformance ... 1
3 Normative references ... 1
4 Terms and definitions ... 3
4.1 Definitions ... 3
5 Symbols and abbreviated terms ... 11
6 Concepts ... 12
6.1 Introduction ... 12
6.2 Delimiter elements ... 13
6.3 Metafile descriptor elements ... 13
6.3.1 Identification ... 14
6.3.2 Functional capability ... 14
6.3.3 Default metafile state ... 17
6.3.4 Fonts and character sets ... 17
6.3.5 Picture directory ... 22
6.4 Picture descriptor elements ... 22
6.4.1 Scaling mode ... 23
6.4.2 Colour selection mode ... 23
6.4.3 Specification modes ... 23
6.4.4 VDC extent ... 23
6.4.5 CGM tailoring ... 24
6.4.6 Background colour ... 26
6.4.7 Device viewport control ... 26
6.4.8 Representations ... 26
6.4.9 Definable attributes ... 27
6.4.10 Application structure directory ... 27
6.5 Control elements ... 27
6.5.1 VDC space and range ... 27

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

6.5.2 Clipping 27
6.5.3 Save and restore primitive context 28
6.5.4 Compound clipping and shielding 29
6.5.5 Generalized text path 32
6.5.6 Mitre limit 32
6.5.7 Transparent cell colour 32
6.6 Graphical primitive elements 32
6.6.1 Line elements 34
6.6.2 Marker elements 36
6.6.3 Text elements 37
6.6.4 Filled-area elements 38
6.6.5 Cell elements 39
6.6.6 Circular arc elements 44
6.6.7 Elliptical elements 44
6.6.8 Hyperbolic arc element 44
6.6.9 Parabolic arc element 45
6.6.10 Spline curve elements 49
6.6.11 Closed figures 51
6.6.12 Symbol elements 58
6.7 Attribute elements 58
6.7.1 Line attributes 60
6.7.2 Marker attributes 62
6.7.3 Text attributes 64
6.7.4 Filled-area attributes 87
6.7.5 Specification modes and transformation of aspects 90
6.7.6 Colour attributes 91
6.7.7 Pick identifier 93
6.7.8 Compound text path 93
6.7.9 Symbol Attributes 93
6.8 Escape elements 94
6.9 External elements 95

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

6.10 Segment elements 95
6.10.1 Introduction 95
6.10.2 Local and global segments 95
6.10.3 Delimiting and naming segments 96
6.10.4 Segment attributes 96
6.10.5 Copy segment and inheritance 97
6.11 Metafile states 102
6.12 Registration 113
6.13 Application Structure Elements 113
6.13.1 Introduction 113
6.13.2 Location of and access to Application Structures 114
6.13.3 Nesting of Application Structures 114
6.13.4 Graphical Context of Application Structures 114
6.13.5 Application Structure Attributes 114
6.13.6 Relationship between application structures and segments 115
7 Abstract specification of elements 118
7.1 Introduction 118
7.2 Delimiter elements 120
7.2.1 BEGIN METAFILE 120
7.2.2 END METAFILE 121
7.2.3 BEGIN PICTURE 121
7.2.4 BEGIN PICTURE BODY 122
7.2.5 END PICTURE 122
7.2.6 BEGIN SEGMENT 122
7.2.7 END SEGMENT 123
7.2.8 BEGIN FIGURE 123
7.2.9 END FIGURE 123
7.2.10 BEGIN PROTECTION REGION 124
7.2.11 END PROTECTION REGION 124
7.2.12 BEGIN COMPOUND LINE 124
7.2.13 END COMPOUND LINE 124

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

7.2.14 BEGIN COMPOUND TEXT PATH 125
7.2.15 END COMPOUND TEXT PATH 125
7.2.16 BEGIN TILE ARRAY 125
7.2.17 END TILE ARRAY 126
7.2.18 BEGIN APPLICATION STRUCTURE 127
7.2.19 BEGIN APPLICATION STRUCTURE BODY 127
7.2.20 END APPLICATION STRUCTURE 127
7.3 Metafile descriptor elements 128
7.3.1 METAFILE VERSION 128
7.3.2 METAFILE DESCRIPTION 128
7.3.3 VDC TYPE 128
7.3.4 INTEGER PRECISION 129
7.3.5 REAL PRECISION 129
7.3.6 INDEX PRECISION 129
7.3.7 COLOUR PRECISION 129
7.3.8 COLOUR INDEX PRECISION 130
7.3.9 MAXIMUM COLOUR INDEX 130
7.3.10 COLOUR VALUE EXTENT 130
7.3.11 METAFILE ELEMENT LIST 131
7.3.12 METAFILE DEFAULTS REPLACEMENT 132
7.3.13 FONT LIST 132
7.3.14 CHARACTER SET LIST 133
7.3.15 CHARACTER CODING ANNOUNCER 133
7.3.16 NAME PRECISION 134
7.3.17 MAXIMUM VDC EXTENT 135
7.3.18 SEGMENT PRIORITY EXTENT 135
7.3.19 COLOUR MODEL 135
7.3.20 COLOUR CALIBRATION 136
7.3.21 FONT PROPERTIES 138
7.3.22 GLYPH MAPPING 140
7.3.23 SYMBOL LIBRARY LIST 142

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

7.3.24 PICTURE DIRECTORY 142
7.4 Picture descriptor elements 142
7.4.1 SCALING MODE 142
7.4.2 COLOUR SELECTION MODE 143
7.4.3 LINE WIDTH SPECIFICATION MODE 143
7.4.4 MARKER SIZE SPECIFICATION MODE 144
7.4.5 EDGE WIDTH SPECIFICATION MODE 144
7.4.6 VDC EXTENT 145
7.4.7 BACKGROUND COLOUR 145
7.4.8 DEVICE VIEWPORT 146
7.4.9 DEVICE VIEWPORT SPECIFICATION MODE 146
7.4.10 DEVICE VIEWPORT MAPPING 147
7.4.11 LINE REPRESENTATION 147
7.4.12 MARKER REPRESENTATION 148
7.4.13 TEXT REPRESENTATION 148
7.4.14 FILL REPRESENTATION 149
7.4.15 EDGE REPRESENTATION 149
7.4.16 INTERIOR STYLE SPECIFICATION MODE 150
7.4.17 LINE AND EDGE TYPE DEFINITION 150
7.4.18 HATCH STYLE DEFINITION 151
7.4.19 GEOMETRIC PATTERN DEFINITION 152
7.4.20 APPLICATION STRUCTURE DIRECTORY 152
7.5 Control elements 153
7.5.1 VDC INTEGER PRECISION 153
7.5.2 VDC REAL PRECISION 153
7.5.3 AUXILIARY COLOUR 153
7.5.4 TRANSPARENCY 154
7.5.5 CLIP RECTANGLE 154
7.5.6 CLIP INDICATOR 155
7.5.7 LINE CLIPPING MODE 155
7.5.8 MARKER CLIPPING MODE 155

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

7.5.9 EDGE CLIPPING MODE 156
7.5.10 NEW REGION 156
7.5.11 SAVE PRIMITIVE CONTEXT 156
7.5.12 RESTORE PRIMITIVE CONTEXT 157
7.5.13 PROTECTION REGION INDICATOR 158
7.5.14 GENERALIZED TEXT PATH MODE 158
7.5.15 MITRE LIMIT 158
7.5.16 TRANSPARENT CELL COLOUR 159
7.6 Graphical primitive elements 159
7.6.1 POLYLINE 159
7.6.2 DISJOINT POLYLINE 159
7.6.3 POLYMARKER 160
7.6.4 TEXT 160
7.6.5 RESTRICTED TEXT 161
7.6.6 APPEND TEXT 162
7.6.7 POLYGON 163
7.6.8 POLYGON SET 164
7.6.9 CELL ARRAY 166
7.6.10 GENERALIZED DRAWING PRIMITIVE (GDP) 168
7.6.11 RECTANGLE 168
7.6.12 CIRCLE 168
7.6.13 CIRCULAR ARC 3 POINT 169
7.6.14 CIRCULAR ARC 3 POINT CLOSE 169
7.6.15 CIRCULAR ARC CENTRE 172
7.6.16 CIRCULAR ARC CENTRE CLOSE 172
7.6.17 ELLIPSE 173
7.6.18 ELLIPTICAL ARC 173
7.6.19 ELLIPTICAL ARC CLOSE 174
7.6.20 CIRCULAR ARC CENTRE REVERSED 175
7.6.21 CONNECTING EDGE 176
7.6.22 HYPERBOLIC ARC 176

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

7.6.23 PARABOLIC ARC 177
7.6.24 NON-UNIFORM B-SPLINE 177
7.6.25 NON-UNIFORM RATIONAL B-SPLINE 178
7.6.26 POLYBEZIER 178
7.6.27 POLYSYMBOL 179
7.6.28 BITONAL TILE 179
7.6.29 TILE 181
7.7 Attribute elements 183
7.7.1 LINE BUNDLE INDEX 183
7.7.2 LINE TYPE 183
7.7.3 LINE WIDTH 184
7.7.4 LINE COLOUR 184
7.7.5 MARKER BUNDLE INDEX 185
7.7.6 MARKER TYPE 185
7.7.7 MARKER SIZE 186
7.7.8 MARKER COLOUR 186
7.7.9 TEXT BUNDLE INDEX 186
7.7.10 TEXT FONT INDEX 187
7.7.11 TEXT PRECISION 187
7.7.12 CHARACTER EXPANSION FACTOR 188
7.7.13 CHARACTER SPACING 189
7.7.14 TEXT COLOUR 189
7.7.15 CHARACTER HEIGHT 189
7.7.16 CHARACTER ORIENTATION 190
7.7.17 TEXT PATH 190
7.7.18 TEXT ALIGNMENT 191
7.7.19 CHARACTER SET INDEX 191
7.7.20 ALTERNATE CHARACTER SET INDEX 192
7.7.21 FILL BUNDLE INDEX 192
7.7.22 INTERIOR STYLE 193
7.7.23 FILL COLOUR 193

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

7.7.24 HATCH INDEX 194
7.7.25 PATTERN INDEX 194
7.7.26 EDGE BUNDLE INDEX 195
7.7.27 EDGE TYPE 195
7.7.28 EDGE WIDTH 196
7.7.29 EDGE COLOUR 196
7.7.30 EDGE VISIBILITY 197
7.7.31 FILL REFERENCE POINT 197
7.7.32 PATTERN TABLE 198
7.7.33 PATTERN SIZE 198
7.7.34 COLOUR TABLE 199
7.7.35 ASPECT SOURCE FLAGS 199
7.7.36 PICK IDENTIFIER 200
7.7.37 LINE CAP 200
7.7.38 LINE JOIN 201
7.7.39 LINE TYPE CONTINUATION 202
7.7.40 LINE TYPE INITIAL OFFSET 202
7.7.41 TEXT SCORE TYPE 203
7.7.42 RESTRICTED TEXT TYPE 203
7.7.43 INTERPOLATED INTERIOR 204
7.7.44 EDGE CAP 205
7.7.45 EDGE JOIN 206
7.7.46 EDGE TYPE CONTINUATION 206
7.7.47 EDGE TYPE INITIAL OFFSET 207
7.7.48 SYMBOL LIBRARY INDEX 207
7.7.49 SYMBOL COLOUR 208
7.7.50 SYMBOL SIZE 208
7.7.51 SYMBOL ORIENTATION 208
7.8 Escape elements 209
7.8.1 ESCAPE 209
7.9 External elements 209

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

7.9.1 MESSAGE 209
7.9.2 APPLICATION DATA 210
7.10 Segment elements....210
7.10.1 COPY SEGMENT 210
7.10.2 INHERITANCE FILTER 211
7.10.3 CLIP INHERITANCE....212
7.10.4 SEGMENT TRANSFORMATION 212
7.10.5 SEGMENT HIGHLIGHTING 213
7.10.6 SEGMENT DISPLAY PRIORITY....213
7.10.7 SEGMENT PICK PRIORITY....213
7.11 Application structure descriptor elements....214
7.11.1 APPLICATION STRUCTURE ATTRIBUTE 214
8 Metafile defaults 215
9 Profiles and conformance 219
9.1 Introduction 219
9.1.1 Objectives 219
9.1.2 Scope....219
9.1.3 Concept and purpose of profiles for CGM....219
9.1.4 Purpose of the Model Profile 220
9.2 Conformance 220
9.2.1 Conformance of profiles....220
9.2.2 Conformance of metafiles 221
9.2.3 Conformance of metafile generators 221
9.2.4 Conformance of metafile interpreters....222
9.3 Criteria for designing profiles....222
9.3.1 Criteria on the profile in its entirety 222
9.3.2 Criteria for the technical content of the profile....222
9.4 Form and format of a profile 222
9.5 Profile rules, proforma, and model profile 223
9.5.1 Overview 223
9.5.2 General principles....224

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

9.5.3 Metafile rules **226**

9.5.4 Multi-element rules **226**

9.5.5 Individual element rules **231**

9.5.6 Generator implementation requirements **231**

9.5.7 Interpreter implementation requirements **233**

9.5.8 PPF Tables **236**

Annex A (normative) **Formal grammar of the functional specification of version 1 metafiles** **238**

A.1 Introduction **238**

A.2 Notation used **238**

A.3 Detailed grammar **238**

A.3.1 Metafile structure **238**

A.3.2 Metafile descriptor elements **239**

A.3.3 Picture descriptor elements **240**

A.3.4 Control elements **241**

A.3.5 Graphical elements **241**

A.3.6 Attribute elements **243**

A.3.7 Escape elements **246**

A.3.8 External elements **246**

A.4 Terminal symbols **247**

Annex B (normative) **Formal Grammar of the functional specification of version 2 metafiles** **251**

B.1 Introduction **251**

B.2 Notation used **251**

B.3 Detailed grammar **251**

B.3.1 Metafile structure **251**

B.3.2 Metafile descriptor elements **252**

B.3.3 Picture descriptor elements **254**

B.3.4 Control elements **256**

B.3.5 Graphical elements **256**

B.3.6 Attribute elements **259**

B.3.7 Closed figure element **261**

B.3.8 Escape elements **262**

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

B.3.9 External elements 262

B.3.10 Segment elements 262

B.4 Terminal symbols 264

Annex C (normative) **Formal grammar of the functional specification of version 3 metafiles** 271

C.1 Introduction 271

C.2 Definitions 271

C.2.1 Notation Used 271

C.2.2 Structured Data Records 271

C.3 Detailed Grammar 272

C.3.1 Metafile structure 272

C.3.2 Metafile descriptor elements 276

C.3.3 Picture descriptor elements 279

C.3.4 Control elements 281

C.3.5 Graphical elements 282

C.3.6 Attribute elements 285

C.3.7 Escape elements 289

C.3.8 External elements 289

C.3.9 Segment elements 290

C.4 Terminal symbols 292

Annex D (informative) **Guidelines for metafile generators and interpreters** 300

D.1 Introduction 300

D.2 Errors and degeneracies 300

D.2.1 Syntax errors 301

D.2.2 Geometrically degenerate primitives 301

D.2.3 Mathematical singularities and ambiguities 302

D.3 General guidelines 302

D.3.1 Indexes 302

D.3.2 Colour model 302

D.3.3 Order of metafile descriptor elements 305

D.3.4 Unsatisfied references 305

D.4 Guidelines for element classes 305

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

D.4.1 Delimiter elements 305
D.4.2 Metafile descriptor elements 305
D.4.3 Picture descriptor elements 305
D.4.4 Control elements 306
D.4.5 Graphical primitive elements 306
D.4.6 Attribute elements 309
D.4.7 Escape elements 311
D.4.8 External elements 311
D.4.9 Segment elements 311
Annex E (informative) Guidelines for private encodings 312
Annex F (informative) Reference models 313
Annex G (informative) Conversions between CIEXYZ reference colour space & metafile colour spaces 317
G.1 Introduction 317
G.2 CIELUV 317
G.2.1 Conversion from the CIEXYZ reference colour space to CIELUV 317
G.2.2 Conversion from CIELUV to the CIEXYZ reference colour space 318
G.3 CIELAB 318
G.3.1 Conversion from the CIEXYZ reference colour space to CIELAB 318
G.3.2 Conversion from CIELAB to the CIEXYZ reference colour space 319
G.4 RGB 320
G.4.1 Conversion from the CIEXYZ reference colour space to RGB 320
G.4.2 Conversion from RGB to the CIEXYZ reference colour space 321
G.5 RGB-related 322
G.6 CMYK 322
G.6.1 Conversion from CMYK to the CIEXYZ reference colour space 322
G.6.2 CMYK Calibration data 322
G.7 Bibliography 323
Annex H (normative) Formal grammar of the functional specification of version 4 metafiles 324
H.1 Introduction 324
H.2 Definitions 324
H.2.1 Notation used 324

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

H.2.2 Structured Data Records: 324
H.3 Detailed grammar 325
H.3.1 Metafile structure 325
H.3.2 Metafile descriptor elements 329
H.3.3 Picture descriptor elements 332
H.3.4 Control elements 335
H.3.5 Graphical elements 336
H.3.6 Attribute elements 339
H.3.7 Escape elements 343
H.3.8 External elements 343
H.3.9 Segment elements 343
H.3.10 Application structure descriptor elements 346
H.4 Terminal Symbols 346
Annex I (normative) Proforma tables and font metrics 355
I.1 Proforma tables 354
I.2 Font character codes and metrics 434
I.2.1 Introduction 434
I.2.2 Association of character code to glyph 434
I.2.3 Font metric tables 437

Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this part of ISO/IEC 8632 may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 8632-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 24, *Computer graphics and image processing*.

This second edition cancels and replaces the first edition (ISO/IEC 8632-1:1992), which has been technically revised. Note that the previous edition of ISO/IEC 8632-1, published in 1992, was a first edition but second edition was indicated by error on its cover page and in the foreword.

ISO/IEC 8632 consists of the following parts, under the general title *Information technology — Computer graphics — Metafile for the storage and transfer of picture description information*:

— *Part 1: Functional specification*

— *Part 3: Binary encoding*

— *Part 4: Clear text encoding*

Annexes A, B, C, H and I form a normative part of this part of ISO/IEC 8632. Annexes D to G are for information only.

NOTE In previous editions of ISO/IEC 8632, Part 2 defined a Character Encoding. Part 2 was withdrawn in 1998, due to its lack of implementation and use.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

Introduction

0.1 Purpose

The Computer Graphics Metafile provides a file format suitable for the storage and retrieval of picture information. The file format consists of a set of elements that can be used to describe pictures in a way that is compatible between systems of different architectures and devices of differing capabilities and design. This picture description includes the capability for describing static pictures. Static pictures are those where elements which may lead to dynamic effects (for example those leading to regeneration) are prohibited within the picture body.

0.2 Reasons for this International Standard

The main reasons for producing a standard computer graphics metafile are

a) to allow picture information to be stored in an organized way on a graphical software system;

b) to facilitate transfer of picture information between different graphical software system;

c) to enable picture information to be transferred between graphical devices;.

d) to enable picture information to be transferred between different computer graphics installations.

0.3 Design requirements

To reach these objectives, a number of design principles were adopted:

a) The metafile should provide a suitable set of elements for the transfer of a wide range of pictorial information.

b) The metafile should address the more usual and essential features found on graphical devices directly and should provide access to less common facilities via an escape mechanism.

c) The design of the metafile should not preclude extension of ISO/IEC 8632 at a later stage to cover facilities beyond those included in this version of the Standard. It should also not preclude further extensions to support future standards.

d) The metafile should be usable from GKS (Graphical Kernel System — ISO 7942) with both metafile input and metafile output functions. It should include the capability to support ISO 7942 (GKS) static picture capture.

e) ISO/IEC 8632 should address the needs of different applications that have conflicting requirements for size of metafile, speed of generation and interpretation, readability, editability and ease of transfer through different transport mechanisms.

0.4 Design criteria

The requirements of 0.3 were used to formulate the following criteria which were used to decide between different design possibilities.

a) Completeness: In any area of ISO/IEC 8632, the functionality specified by ISO/IEC 8632 should be complete in itself.

b) Conciseness: Redundant elements or parameters should be avoided.

c) Consistency: Contradictory elements should be avoided.

d) Extensibility: The ability to add new elements and generality to ISO/IEC 8632 should not be precluded.

e) Fidelity: The minimal results and characteristics of elements should be well defined.

Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

f) Implementability: An element should be able to be supported efficiently on most host systems and/or graphics hardware.

g) Orthogonality: The elements of the metafile should be independent of each other, or any dependencies should be structured and well defined.

h) Predictability: ISO/IEC 8632 should be such that the recommended or proper use of standard elements guarantees the results of using a particular element.

i) Standard practice: Only those elements that reflect existing practice, that are necessary to support existing practice, or that are necessary to support proposed standards should be standardized.

j) Usefulness: Functions should be powerful enough to perform useful tasks.

k) Well-structured: The assumptions that elements make about each other should be minimized. An element should have a well-defined interface and a simply stated unconditional purpose. Multipurpose elements and side effects should be avoided.

0.5 Access to a metafile

The metafile has been designed so that, although its main usage is anticipated as being with completely sequential access, non-sequential access is also possible. Once the basic environment of the metafile has been established, individual pictures may be accessible if the medium, the encoding and the implementation support this form of access.

0.6 Generation and interpretation of metafiles

The specific mechanisms of metafile generation and interpretation are not described by ISO/IEC 8632, although it does describe the intended result of such interpretation. The basic set of metafile elements includes a capability for the addition of application-dependent data, which do not have graphical meaning and for which no intended interpretation results are described.

0.7 Distinction between formal specification and encodings

The functionality provided by the metafile is separated from the specification of any particular encoding format. ISO/IEC 8632 provides for both standard and private encodings of the elements described in this part of ISO/IEC 8632. Guidelines for private encodings are specified in annex E; these guidelines do not form part of ISO/IEC 8632.

Standardized encodings are specified in parts 3 and 4 of ISO/IEC 8632. Each of the standardized encodings is capable of representing the full functionality described in this part of ISO/IEC 8632. Translation between the standardized encodings is possible without loss of picture information, although subsequent translation back into the original encoding may not result in precisely the same data stream, due to different quantizations of precisions in the different encodings.

The binary encoding specified in ISO/IEC 8632-3 provides an encoding that requires least effort to generate and interpret on many systems, and which is relatively compact as well.

The clear text encoding specified in ISO/IEC 8632-4 provides an encoding that can be created, viewed and edited with standard text editors. It is therefore also suitable for transfer through networks that support only transfer of text files.

0.8 Relationship to other International Standards

ISO/IEC 8632 draws extensively for its model of a graphics system on GKS (Graphical Kernel System — ISO 7942). In addition, ISO/IEC 8632 specifies a metafile that may be used as a static picture-capture metafile by GKS.

This part of ISO/IEC 8632 uses font concepts and the font architecture defined in ISO/IEC 9541-1 for defining CGM references to fonts and font resources. The font properties of ISO/IEC 9541-1 are adopted, where appropriate, to define CGM mechanisms to provide information useful for font substitution between parties interchanging Metafiles. This part of ISO/IEC 8632 includes the minimum font description subset defined in ISO/IEC 9541-2. Clause 3

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

contains a number of glossary definitions that are taken from, and are identical to, those found in ISO/IEC 9541-1. This part of ISO/IEC 8632 also defines access to extended families of glyph based on the principles and procedures of ISO/IEC 10036.

This part of ISO/IEC 8632 uses a colorimetrically precise reference colour space to allow for interchange of precise colour specifications. It uses concepts defined in ISO/IEC 8613/Amd.2 which are based on CIE publications. ISO/IEC 8613/Amd.2 provides tutorial material on relevant definitions and colour concepts, which is useful for understanding the material in this Standard but is not incorporated into this Standard in that amount of detail.

The binary encoding specified in ISO/IEC 8632-3 employs the mechanism for representing floating point numbers specified in ANSI/IEEE 754-1986.

For certain elements, the CGM defines value ranges of parameters as being reserved for registration. The meanings of these values will be defined using the established procedures (see subclause 6.12) of the ISO International Registration Authority for Graphical Items. These procedures do not apply to values and value ranges defined as being reserved for private use; these values and ranges are not standardized. There is a very close relationship between many of the elements in ISO 8632 and a subset of the functions in the CGI (Computer Graphics Interface — ISO/IEC 9636).

0.9 Versions

ISO/IEC 8632 defines several versions of the Computer Graphics Metafile. A version is defined by a formal grammar and additional specifications contained in clauses 6, 7, and 8.

The following versions are currently defined: Version 1 (one); Version 2 (two); Version 3 (three); and Version 4 (four).

NOTE 1 A valid Version 3 metafile is also a valid Version 4 metafile. A valid Version 2 metafile is also a valid Version 3 metafile. A valid Version 1 metafile is also a valid Version 2 metafile.

NOTE 2 Version 1 metafiles are as defined by the original CGM standard, which was designated ISO 8632:1987. Version 2 metafiles are as defined by the first amendment to the CGM standard, which was designated ISO 8632:1987/Amd.1:1990. Version 3 metafiles are as defined by an amendment to Version 2 of the CGM standard. This amendment was originally designated ISO/IEC 8632:1987/Amd.3:1991, but was never published as an amendment. Instead all documents were consolidated to produce the 1992 revision of ISO 8632; Versions 1, 2, and 3 were all defined by the 1992 revision. Version 4 metafiles were originally defined by ISO/IEC 8632:1992/Amd.2-1995, which has been integrated into this (1999) revision of ISO/IEC 8632; Versions 1, 2, 3, and 4 are all defined by this revision

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

Information technology — Computer graphics — Metafile for the storage and transfer of picture description information —

Part 1:
Functional specification

1 Scope

ISO/IEC 8632 provides a file format suitable for the storage and retrieval of picture description information. The file format consists of an ordered set of elements that may be used to describe pictures in a way that is compatible between systems of different architectures, compatible with devices of differing capabilities and design, and meaningful to application constituencies. This picture description includes the capability for describing static images.

The elements specified provide for the representation of a wide range of pictures on a wide range of graphical devices. The elements are organized into groups that delimit major structures (metafiles, pictures, and application structures), that specify the representations used within the metafile, that control the display of the picture, that perform basic drawing actions, that control the attributes of the basic drawing actions, that allow application-specific structuring to be overlaid on the graphical content, and that provide access to non-standard device capabilities.

The metafile is defined in such a way that, in addition to sequential access to the whole metafile, random access to individual pictures and individual context-independent application structures is well-defined. Applications which require random access to pictures and/or context-independent application structures within pictures may, within the metafile, define directories to these pictures and/or context-independent application structures. The metafile may then be opened and randomly accessed without interpreting the entire metafile.

In addition to a functional specification, two standard encodings of the metafile syntax are specified. These encodings address the needs of applications that require small metafile size plus minimum effort to generate and interpret, and maximum flexibility for a human reader or editor of the metafile.

This part of ISO/IEC 8632 describes the format using an abstract syntax. The remaining parts of ISO 8632 specify standardized encodings that conform to this syntax: ISO/IEC 8632-3 specifies a binary encoding; ISO/IEC 8632-4 specifies a clear text encoding.

2 Conformance

ISO/IEC 8632 defines specific criteria for conformance of metafiles, conformance of metafile generators, and conformance of metafile interpreters to this International Standard. Conformance is defined in terms of profiles. Specific conformance criteria for metafiles, generators, and interpreters are defined in clause 9. The criteria for valid, conforming profiles, plus a Model Profile, are specified in clause 9 and annex I. A methodology for defining valid profiles, and a pro forma (Profile Proforma, or PPF) for writing profiles, are defined in clause 9 and annex I.

3 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 8632. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this part of ISO/IEC 8632 are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO/IEC 646:1991, *Information technology — ISO 7-bit coded character set for information interchange.*

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

ISO 2022:1986, *Information processing — ISO 7-bit and 8-bit coded character sets — Code extension techniques.*

ISO 2375:1985, *Data processing — Procedure for registration of escape sequences.*

ISO 7942:1985, *Information processing systems — Computer graphics — Graphical Kernel System (GKS) functional description.*

ISO 8601:1988, *Data elements and interchange formats — Information interchange — Representation of dates and time.*

ISO 8859-1:1987, *Information processing — 8-bit single-byte coded graphic character sets — Part 1: Latin alphabet No. 1.*

ISO/IEC 9541-1:1991, *Information technology — Font information interchange — Part 1: Architecture.*

ISO/IEC 9541-2:1991, *Information technology — Font information interchange — Part 2: Interchange format.*

ISO/IEC 9636-1:1991, *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Functional specification — Part 1: Overview, profiles, and conformance.*

ISO/IEC 9636-2:1991, *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Functional specification — Part 2: Control.*

ISO/IEC 9636-3:1991, *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Functional specification — Part 3: Output.*

ISO/IEC 9636-4:1991, *Information technology — Computer graphics — Interfacing techniques for dialogues with graphical devices (CGI) — Functional specification — Part 4: Segments.*

ISO/IEC TR 9973:1988, *Information processing — Procedures for registration of graphical items.*

ISO/IEC 10036, *Information technology — Font information interchange — Procedure for registration of glyph and glyph collection identifiers.*

ISO/IEC TR 10000-1:1992, *Information technology — Framework and taxonomy of International Standardized Profiles — Part 1: Framework.*

ISO/IEC TR 10000-2:1992, *Information technology — Framework and taxonomy of International Standardized Profiles — Part 2: Taxonomy of OSI Profiles.*

ISO/IEC 10646-1:1993, *Information technology — Universal Multiple-Octet Coded Character Set (UCS) — Part 1: Architecture and Basic Multilingual Plane.*

ISO/IEC 10646-1:1993, Amd.2:1996, *Information technology — Universal Multiple-Octet Coded Character Set (UCS) — Part 1: Architecture and Basic Multilingual Plane, Amendment 2: UCS Transformation Format 8 (UTF-8).*

CCITT Recommendation T.4 (1988), *Standardization of group 3 facsimile apparatus for document transmission.*

CCITT Recommendation T.6 (1988), *Standardization of group 4 facsimile apparatus for document transmission.*

CIE Publication 15.2, *Colorimetry, 1986 (2nd edition).*

CIE Publication S002, *Colorimetric Observers, 1986 (1st edition).*

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

4 Terms and definitions

4.1 Definitions

For the purposes of this part of ISO/IEC 8632, the following definitions apply.

NOTE As far as possible, commonly accepted graphics terminology is used.

4.1.1
anisotropic mapping
A mapping in which the scale factors applied along each axis are not equal. This is often used in reference to the mapping from VDC to distance units on the physical display surface. With anisotropic mapping, the angle between any pair of non-parallel line segments can change; circles cease to be circles and become ellipses after such a transformation. See "isotropic mapping".

4.1.2
aspect ratio
The ratio of the width to the height of a rectangular area, such as a window or viewport. For example, an aspect ratio of 2.0 indicates an area twice as wide as it is high.

4.1.3
aspect source flag (ASF)
Indicator as to whether a particular attribute selection is to be individual or bundled.

4.1.4
aspects of primitives
Ways in which the appearance of a primitive can vary. Some aspects are controlled directly by primitive attributes; some are controlled indirectly through a bundle table.

4.1.5
attribute elements
Metafile elements that describe the appearance of graphical element.

4.1.6
boundary
The mathematical locus that defines, in abstract VDC space, the limits of a region to be filled (for fill primitives and closed figures). The visual appearance of interior style ‘hollow' consists of a depiction of the boundary obtained after clipping has been taken into account.

4.1.7
brightness
Attribute of a visual sensation according to which an area appears to emit more or less light.

4.1.8
bundle
Set of attributes associated with one of the following graphical element types: line, marker, text, and filled area.

4.1.9
bundle index
An index into a bundle table for a particular output primitive.

4.1.10
bundle table
An indexed table containing a set of attributes for each index.

4.1.11
character set
The set of displayable symbols mapped to individual characters in a TEXT, APPEND TEXT, or RESTRICTED TEXT string. This corresponds to the "G-set" defined in ISO 2022. A character set is independent of the font or typeface; examples of character sets are: ASCII (X3.4), German and Katakana.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

4.1.12
chroma
Colourfulness (chromaticness) of an area judged in proportion to the brightness of a similarly illuminated area that appears white or highly transmitting.

4.1.13
chromaticity
Ratio of each of a set of three tristimulus values to their sum. (As the sum of the three chromaticity coordinates equals 1, two of them are sufficient to define a chromaticity.)

4.1.14
CIELAB colour space
A CIE recommended, approximately uniform, colour space with rectangular coordinates , L^*, a^*, and b^*. L^* is the approximate correlate of lightness, a^* and b^* are used to calculate approximate correlates of hue and chroma. CIELAB uses the tristimulus ratios X/X_n, Y/Y_n, Z/Z_n instead of chromaticity coordinates as in CIELUV. X_n, Y_n, Z_n are the values of X, Y, Z for the appropriate chosen reference white. The reduced perceptual significance of a given difference in chromaticity as the colour becomes darker is incorporated by using the tristimulus ratios.

4.1.15
CIELUV colour space
A CIE recommended, approximately uniform, colour space with rectangular coordinates L^*, u^*, and v^*. L^* is the approximate correlate of lightness, u^* and v^* are used to calculate approximate correlates of hue and chroma. The colour stimulus is described by Y, u', v' and the reference white by Y_n, u_n', v_n'. A given difference in chromaticity is reduced in magnitude by the factor L^* as the colour becomes darker.

4.1.16
CIE tristimulus values
Amounts of the three reference colour stimuli, in a given trichromatic system, required to match the colour of the stimulus considered.

4.1.17
CIE uniform colour spaces
Two CIE recommended, approximately uniform colour spaces, CIELAB and CIELUV, are allowed in the CGM. These colour spaces are non-linear transformations of the CIE 1931 XYZ tristimulus space, into the approximate perceptual correlates of lightness, hue, and chroma. CIELAB and CIELUV closely approximate uniform colour spaces over small distances, and they each provide an approximately uniform measure of perceived colour differences. Both colour spaces allow for the perceptual effect that a given difference in chromaticity represents a smaller and smaller colour difference as the lightness is reduced.

4.1.18
clip indicator
Indicator as to whether metafile graphical elements are to be clipped at the limits of CLIP RECTANGLE.

4.1.19
clipping mode
A generic term referring to one of Line Clipping, Marker Clipping or Edge Clipping Modes. An object clipping may be either ‘locus', ‘shape' or ‘locus then shape'.

4.1.20
clip rectangle
A rectangle defined in VDC space which is used as a clipping boundary when the metafile graphical elements are to be clipped.

4.1.21
clipping
Removing parts of display elements that lie outside a given boundary.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

4.1.22
closed figure
A compound primitive that behaves as a fill primitive of more general shape. It is formed by delimiting a sequence of line or fill primitives, edge attributes, and certain control elements, with the elements BEGIN FIGURE and END FIGURE.

4.1.23
CMYK colour space
A colour space based on the subtractive colour mixture of Cyan (C), Magenta (M) and Yellow (Y) primaries with the inclusion of black (K).

4.1.24
colour component
One of the dimensions of a colour space.

4.1.25
colour model
A specification of a 3D colour coordinate system and a 3D subspace in the coordinate system within which each displayable colour is represented by a point. Some colour models include a fourth, redundant, dimension to allow the independent representation of black. For the purpose of ISO/IEC 8632 colour model refers to one of RGB, CIELAB, CIELUV, CMYK, or RGB-related, which may be designated as the colour space for metafile content, or CIEXYZ, which is used in this International Standard as a reference space for specifying colour transformations.

4.1.26
colour selection mode
Indicator as to whether colour selection is to be direct (by specifying a colour value) or indexed (by specifying an index into a table of colour values). See COLOUR VALUE.

4.1.27
colour space
See COLOUR MODEL.

4.1.28
colour stimulus
Visible radiation entering the eye and producing a sensation of colour.

4.1.29
colour table
A table for use in mapping from a colour index to the corresponding colour. See DIRECT COLOUR, INDEXED COLOUR.

4.1.30
colour value
Value of the n-tuple of components describing a colour in a given colour model.

4.1.31
compound primitive
A compound primitive is specified by a sequence of CGM elements, as opposed to primitives represented by a single element. Compound text and closed figures are examples of compound primitives in the CGM.

4.1.32
Computer Graphics Interface (CGI)
The specification for interface techniques for dialogues with graphical devices.

4.1.33
Computer Graphics Metafile (CGM)
The specification for a mechanism for storing and transferring picture description information.

Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

4.1.34
conjugate diameter pair (CDP)
A pair D,d of diameters of an ellipse such that a tangent to the ellipse at each endpoint of one diameter is parallel to the other diameter.

4.1.35
control elements
Metafile elements that specify address space, clipping boundaries, picture delimiters, and format descriptions of the Metafile elements.

4.1.36
data interface
An interface between software modules or devices comprising one or more packets containing opcodes and data (as contrasted with a subroutine or function call interface).

4.1.37
descriptor elements
Metafile elements that describe the functional content, format, default conditions, identification, and characteristics of the Metafile.

4.1.38
device coordinates
The coordinates native to a device; device-dependent coordinates; physical device coordinates.

4.1.39
device driver
The device-dependent part of a graphics implementation which supports a physical device. The device driver generates device-dependent output.

4.1.40
device viewport
A rectangular subset of the physical display surface into which VDC EXTENT is mapped. See "effective viewport".

4.1.41
direct colour
A colour selection scheme in which the colour values are specified directly, without requiring an intermediate mapping via a colour table. See COLOUR TABLE, INDEXED COLOUR.

4.1.42
display surface
That part of a graphics device upon which a visible image appears (for example, the screen of a display, the paper in a plotter).

4.1.43
edge
The rendering of the perimeter of a filled region, controlled by edge attributes. Edges are clipped after being applied to the boundary, as distinct from the rendition of the boundary obtained from interior style ‘hollow'. See "boundary".

4.1.44
effective viewport
The actual viewport resulting from forced isotropic mapping from the VDC extent to the viewport.

4.1.45
escape elements
Metafile elements that describe device- or system-dependent elements used to construct a picture, but that are not otherwise standardized.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

4.1.46
external elements
Metafile elements that communicate information not directly related to the generation of a graphical image.

4.1.47
escapement
During the rendering of text strings onto a display, the movement of the current position on the display surface after a glyph representation is imaged.

4.1.48
escapement point
A glyph metric; a point in the glyph coordinate system, to which the current position on the display surface is usually translated, after the glyph representation is imaged.

4.1.49
font
A collection of glyph images having the same basic design, e.g., *Courier Bold Oblique*.

4.1.50
font family
A collection of fonts of common design, e.g., *Courier, Courier Bold, Courier Bold Oblique*.

4.1.51
font metrics
The set of dimensions and positioning information in a font resource common to all glyph representations contained in that font resource.

4.1.52
font resource
A collection of glyph representations together with descriptive and font metric information which are relevant to the collection of glyph representations as a whole.

4.1.53
foreground colour
The colour used in the rendering process in which primitives are rendered on the display surface, as opposed to the BACKGROUND COLOUR or AUXILIARY COLOUR. The foreground colour is set separately for each class of primitive.

4.1.54
glyph
A recognizable abstract graphical symbol, which is independent of any specific design.

4.1.55
global segment
A segment that is defined in the Metafile Descriptor (see "segment"), in which case it may be referenced from within any picture.

4.1.56
graphical primitive elements
Metafile elements that describe images in the Metafile.

4.1.57
Graphical Kernel System (GKS)
An ISO standard application programmer's interface to graphics (ISO 7942).

4.1.58
graphics device
A device (for example, refresh display, storage tube display, or plotter) on which display images can be represented.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

4.1.59
graphic object
A graphical primitive or a compound primitive, together with the associated attributes.

4.1.60
hatch style
A format for filling closed figures. A hatch style consists of one or more sets of lines whose presence represents the interior of the figure in question.

4.1.61
hue
Attribute of a visual sensation according to which an area appears to be similar to one of the perceived colours, red, yellow, green, and blue, or to a combination of two of them.

4.1.62
indexed colour
A colour selection scheme in which the colour index is used to retrieve colour values from a colour table. See COLOUR TABLE, DIRECT COLOUR.

4.1.63
isotropic mapping
A mapping which is invariant with respect to direction; equal scaling in all orthogonal representational dimensions. It is often used to describe the mapping from VDC to distance units on the physical display surface. With isotropic mapping, the angle between any pair of non-parallel line segments remains unchanged; for example, circles remain circles. See "anisotropic mapping".

4.1.64
local segment
A segment which is defined in the picture descriptor or picture body, and whose definition is local to the picture in which it appears.

4.1.65
message
A string of characters used to communicate information to operators at Metafile interpretation time.

4.1.66
metafile
A mechanism for retaining and transporting graphical data and control information. This information contains a device-independent description of one or more pictures.

4.1.67
Metafile Descriptor (MD)
A metafile element that describes the format of the metafile (but not its encoding method) and the functionality expected of a metafile interpreter.

4.1.68
metafile element
A functional item that can be used to construct a picture or convey information.

4.1.69
metafile generator
The process or equipment that produces the Computer Graphics Metafile.

4.1.70
metafile interpreter
The process or equipment that reads the Computer Graphics Metafile and interprets the contents. An interpreter may be needed in order to drive a Computer Graphics Interface or other device interface to obtain a picture that resembles the intended picture as closely as possible.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

4.1.71
normalized device coordinates (NDC)
Coordinates specified in a device-independent coordinate system, normalized to some range (typically 0 to 1). See VDC EXTENT, VDC RANGE, VDC SPACE, VIRTUAL DEVICE COORDINATES.

4.1.72
object clipping
Object clipping is applied to a graphic object. For example, clipping is applied to a line after it has had the width attribute associated with it.

4.1.73
pattern style
A format for filling closed figures with patterns. A pattern style consists of an array of variously coloured or shaded cells

4.1.74
Picture Descriptor (PD)
A set of metafile elements used to set the interpretation modes of attribute elements for the entire picture.

4.1.75
pixel
The smallest element of a display surface that can be independently assigned colour.

4.1.76
posture
The extent to which the shape of a glyph or set of glyphs appear to incline, including any consequent design or form change.

4.1.77
primary colour stimuli
Three selected coloured lights used to specify the colour of any light represented by the amounts of the three lights that must be mixed additively to produce light matching the light presented. (Any three coloured lights may serve as primaries provided no one of them can be matched by a mixture of the other two. To achieve the maximum gamut of colours by additive mixture, saturated red, green, and blue primaries are commonly used.)

4.1.78
primary colorants
A small number of colorants (dyes or pigments) that may be mixed subtractively to produce a large gamut of colours. The most common primary colorants are cyan (greenish blue), magenta (purplish red), and yellow.

4.1.79
realized edge
The zero-width ideal boundary line of the filled-area if the edge is invisible, and the finite-width displayed line if the edge is visible.

4.1.80
realized interior
In a filled area element, that portion of the ideal interior as extending to and terminating at the realized edge.

4.1.81
region
In the context of closed figures or the POLYGON SET element, an area that is explicitly or implicitly closed, that is a subset of the full area being filled. Regions can be nested, disjoint or overlapping. The boundaries of all regions are considered together when applying the interior test for filling a closed figure or POLYGON SET.

4.1.82
RGB colour space
A colour space with colorimetric coordinates based on red, green and blue reference stimuli or primaries. Colour values may be negative in certain areas outside the gamut defined by the RGB primaries.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

4.1.83
RGB-related colour space
A colour space related to the RGB colour space through a linear transformation (3x3 matrix). (Examples are YUV (PAL, SECAM), YIQ (NTSC) or $\mathbf{YC_RC_B}$ (CCITT video codecs and ITU 601 studio standard)).

4.1.84
reference colour model
Basic colour model within CGM relative to which relationships to specifiable colour models (RGB, CMYK, CIELUV, CIELAB, and RBG-related) are calibrated. The reference colour model is defined by the CIE 1931 standard colorimetric system (XYZ).

4.1.85
saturation
Colourfulness of an area judged in proportion to its brightness.

4.1.86
segment
A collection of primitives, primitive attributes and some additional attributes associated with the segment as a whole. See "segment attribute".

4.1.87
segment attribute
An attribute associated with a segment as a whole rather than attributes of individual primitives.

4.1.88
size specification mode
A generic term for Line Width Specification Mode, Edge Width Specification Mode, or Marker Size Specification Mode.

4.1.89
skewed
Used to describe stroke precision text when the CHARACTER ORIENTATION vectors are non-perpendicular; CELL ARRAYs when the three defining points form a parallelogram which is not a rectangle; or a segment transformation that causes rectangles to become non-rectangular parallelograms.

4.1.90
symbol
A graphical object which is included by reference at some point in the metafile.

4.1.91
trichromatic system
System for specifying colour stimuli in terms of tristimulus values, based on matching colours by additive mixture of three suitable chosen reference colour stimuli.

4.1.92
view surface
See DISPLAY SURFACE.

4.1.93
virtual device
An idealized graphics device that presents a set of graphics capabilities to graphics software or systems via the Computer Graphics Interface.

4.1.94
virtual device coordinates (VDC)
The coordinates used to specify position in the VDC space. These are absolute two-dimensional coordinates. See VDC SPACE.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

4.1.95
VDC extent
A rectangular region of interest contained within the VDC range. See VDC RANGE, VDC SPACE.

4.1.96
VDC range
A rectangular region within VDC space consisting of the set of all coordinates representable in the declared coordinate type, precision, and encoding format of the metafile. See VDC EXTENT, VDC SPACE.

4.1.97
VDC space
A two-dimensional Cartesian coordinate space of infinite precision and extent. Only a subset of VDC space, the VDC range, is realizable in a metafile. See VDC EXTENT, VDC RANGE, VIRTUAL DEVICE COORDINATES.

4.1.98
weight
The ratio of a glyph's or set of glyphs' stem width to font height.

4.1.99
colour device
A device which offers more than two colours.

4.1.100
grey-scale device
A special case of colour device where hue and saturation are the same for all colours (generally saturation = 0).

4.1.101
geometric degeneracy
The degeneracy is intrinsic to the parameterization of the element. A degeneracy which results when parameterization for the geometry does not provide sufficient information to draw the intended primitive.

4.1.102
interoperability
The generator and the interpreter have the same understanding of the encodings, the syntax, and the semantics of the metafile.

4.1.103
monochrome device
A device which has only two colours, a foreground and a background colour. The background colour is the colour of the display surface after it has been cleared.

4.1.104
Profile Proforma (PPF)
A template consisting of profile specifications, which is used by writers of profiles for generating instances of a profile. A completed PPF specifies the rules and options of a profile of ISO/IEC 8632.

4.1.105
application structure
A sequence of metafile elements delimited by the BEGIN APPLICATION STRUCTURE and END APPLICATION STRUCTURE elements, containing one BEGIN APPLICATION STRUCTURE BODY element, and optionally containing one or more APPLICATION STRUCTURE ATTRIBUTE elements.

5 Symbols and abbreviated terms

The following abbreviations are used in all parts of ISO/IEC 8632.

APS Application Structure

ASF Aspect Source Flag

Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

CDP Conjugate Diameter Pair

CGI Computer Graphics Interface

CGM Computer Graphics Metafile

GDP Generalized Drawing Primitive

GKS Graphical Kernel System

ISP International Standardized Profile

MD Metafile Descriptor

NDC Normalized Device Coordinate(s)

PD Picture Descriptor

PPF Profile Proforma

VDC Virtual Device Coordinate(s)

6 Concepts

6.1 Introduction

The objective of the Computer Graphics Metafile (CGM) is to provide for the description, storage, and communication of graphical information together with related, application-specific information in a device-independent manner. To accomplish this, ISO/IEC 8632 defines the form (syntax) and functional behaviour (semantics) of a set of elements that may occur in the CGM. The following classes of elements are defined:

— Delimiter Elements, which delimit significant structures within the Metafile.

— Metafile Descriptor Elements, which describe the functional content, default conditions, identification, and characteristics of the CGM; and optionally, define a directory.

— Picture Descriptor Elements, which set the interpretation modes of attribute elements for each picture and optionally, define a directory to the application structures contained in each picture.

— Control Elements, which allow picture boundaries and coordinate representation to be modified.

— Graphical Primitive Elements, which describe the visual components of a picture in the CGM.

— Attribute Elements, which describe the appearance of graphical primitive elements.

— Escape Elements, which describes device- or system-dependent elements used to construct a picture; however, the elements are not otherwise standardized.

— External Elements, which communicate information not directly related to the generation of a graphical image.

— Segment Elements, which enable the grouping of graphic objects for graphical operations such as copying.

— Application structure elements, which enable the grouping of elements for retrieval, electronic linking, and other specific application-dependent operations.

A Computer Graphics Metafile is a collection of elements from this standardized set. The BEGIN METAFILE and END METAFILE elements each occur exactly once in a complete metafile; as many or as few of the elements in the other classes may occur as are needed. A metafile needs to be interpreted in order to display its pictorial content on a graphics device. The descriptor elements give the interpreter sufficient data to interpret metafile elements and to make informed decisions concerning the resources needed for display.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

Any CGM contains certain delimiter elements. In addition it may include control elements for metafile interpretation, Picture Descriptor elements for declaring parameter modes of attribute elements and defining the representations of certain attribut index values, graphical primitive elements for defining graphical entities, attribute elements for defining the appearance of the graphical primitive elements, escape elements for accessing non-standardized features of particular devices, and external elements for communication of information external to the definition of the pictures in the CGM. In Version 2, Version 3, and Version 4 metafiles, graphical output primitives, attributes, and control elements may be grouped in segments.

A minimal correct metafile consists of BEGIN METAFILE, a Metafile Descriptor consisting of METAFILE VERSION and METAFILE ELEMENT LIST, and END METAFILE. In Version 4 metafiles, graphical output primitives, attributes, control elements, and application structure attributes may be grouped in application structures.

6.2 Delimiter elements

Every metafile starts with a BEGIN METAFILE element and ends with an END METAFILE element. This allows multiple metafiles to be stored or transferred together.

Each picture starts with a BEGIN PICTURE element and ends with an END PICTURE element. Between these delimiters the Picture Descriptor is separated from the picture body by a BEGIN PICTURE BODY element.

Once the Metafile Descriptor has been read, access to individual pictures, on a random as opposed to sequential basis, may be safely accomplished if the encoding, access mechanism and implementation permit.

BEGIN METAFILE and BEGIN PICTURE both have parameters for a name by which the metafile and picture (respectively) can be identified.

In Version 2, Version 3 and Version 4 metafiles, primitives may be grouped together to form a composite primitive known as a closed figure. The primitives to be included in the closed figure being defined are delimited by the elements BEGIN FIGURE and END FIGURE.

In Version 2, Version 3 and Version 4 metafiles, groups of elements, called segments, are delimited by BEGIN SEGMENT and END SEGMENT. Each segment is uniquely identified by a segment identifier. Segments may be defined in the Metafile Descriptor, in the Picture Descriptor, or within picture bodies.

In Version 3 and Version 4 metafiles, a compound clipping or shielding region may be defined by line and filled-area elements occurring between BEGIN PROTECTION REGION and END PROTECTION REGION.

In Version 3 and Version 4 metafiles a compound path may be defined for drawing a compound line primitive. A compound line is defined by line primitive elements occurring between BEGIN COMPOUND LINE and END COMPOUND LINE elements. A compound path may also be defined for displaying text strings along an arbitrary text path. A compound text path is defined by line primitive elements occurring between BEGIN COMPOUND TEXT PATH and END COMPOUND TEXT PATH elements. The allowed elements and rules for definition are identical for the two types of paths.

In Version 3 and Version 4 metafiles, a tile array may be defined by tile array elements occurring between BEGIN TILE ARRAY and END TILE ARRAY.

The exact list of elements which may occur in any of these definition states will be found in the state table, Table 8a.

In Version 4 metafiles, application structures consisting of groups of elements within picture bodies, are delimited by BEGIN APPLICATION STRUCTURE and END APPLICATION STRUCTURE. Between these delimiters the BEGIN APPLICATION STRUCTURE BODY element separates the APPLICATION STRUCTURE ATTRIBUTE elements in the application structure descriptor from the other elements contained in the application structure body.

6.3 Metafile descriptor elements

The Metafile Descriptor (MD) is a group of elements that describes the functional capabilities required to interpret the CGM. These elements are

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

METAFILE VERSION	FONT LIST
METAFILE DESCRIPTION	CHARACTER SET LIST
VDC TYPE	CHARACTER CODING ANNOUNCER
INTEGER PRECISION	NAME PRECISION
REAL PRECISION	MAXIMUM VDC EXTENT
INDEX PRECISION	SEGMENT PRIORITY EXTENT
COLOUR PRECISION	COLOUR MODEL
COLOUR INDEX PRECISION	COLOUR CALIBRATION
MAXIMUM COLOUR INDEX	FONT PROPERTIES
COLOUR VALUE EXTENT	GLYPH MAPPING
METAFILE ELEMENT LIST	SYMBOL LIBRARY LIST
METAFILE DEFAULTS REPLACEMENT	PICTURE DIRECTORY

NOTE 1 Other elements, as defined in this part of ISO/IEC 8632, may appear within the Metafile Descriptor within the definition of a global segment.

In a particular metafile, the METAFILE ELEMENT LIST lists at least those standardized elements that occur in the metafile. The CGM interpreter is thus informed of the capabilities required to successfully interpret the Computer Graphics Metafile. The CGM contains a single Metafile Descriptor. The Metafile Descriptor immediately follows the BEGIN METAFILE element in a metafile (with the possible exception of intervening external and escape elements).

METAFILE VERSION and METAFILE ELEMENT LIST shall occur only once in the Metafile Descriptor for metafiles of all three versions.

NOTE 2 It is recommended that the following elements: METAFILE VERSION, METAFILE ELEMENT LIST and (possibly multiple occurrences of) METAFILE DESCRIPTION appear first in the Metafile Descriptor and in the order listed.

6.3.1 Identification

The identifying information includes declaration of the version of ISO/IEC 8632, by the METAFILE VERSION element, and descriptive information about the metafile (origin, owner, generation date, etc.), by the METAFILE DESCRIPTION element.

6.3.2 Functional capability

The contents of the Computer Graphics Metafile are defined by the METAFILE ELEMENT LIST element. This shall contain a list of the non-mandatory elements that are utilized in the metafile (see 6.1 for mandatory elements). Several shorthand names for CGM elements are also provided for use with the METAFILE ELEMENT LIST. The shorthand names shall not be considered macro names, nor shall they be construed to be levels of conformance

6.3.2.1 Drawing set

The drawing set includes the mandatory CGM elements (i.e., those that shall appear in every conforming CGM) and most of the graphical primitive elements and attribute elements. The drawing set is specified by the shorthand name DRAWING SET.

The elements included in the drawing set are:

BEGIN METAFILE	LINE TYPE
END METAFILE	LINE WIDTH
BEGIN PICTURE	LINE COLOUR
BEGIN PICTURE BODY	MARKER BUNDLE INDEX
END PICTURE	MARKER TYPE
METAFILE VERSION	MARKER SIZE
METAFILE DESCRIPTION	MARKER COLOUR
VDC TYPE	TEXT BUNDLE INDEX
METAFILE ELEMENT LIST	TEXT FONT INDEX
AUXILIARY COLOUR	TEXT PRECISION
TRANSPARENCY	CHARACTER EXPANSION FACTOR
CLIP RECTANGLE	CHARACTER SPACING
CLIP INDICATOR	TEXT COLOUR

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

VDC EXTENT
BACKGROUND COLOUR
COLOUR SELECTION MODE
POLYLINE
DISJOINT POLYLINE
POLYMARKER
TEXT
RESTRICTED TEXT
APPEND TEXT
POLYGON
POLYGON SET
CELL ARRAY
GENERALIZED DRAWING PRIMITIVE
RECTANGLE
CIRCLE
CIRCULAR ARC 3 POINT
CIRCULAR ARC 3 POINT CLOSE
CIRCULAR ARC CENTRE
CIRCULAR ARC CENTRE CLOSE
ELLIPSE
ELLIPTICAL ARC
ELLIPTICAL ARC CLOSE
LINE BUNDLE INDEX
CHARACTER HEIGHT
CHARACTER ORIENTATION
TEXT PATH
TEXT ALIGNMENT
FILL BUNDLE INDEX
INTERIOR STYLE
FILL COLOUR
HATCH INDEX
PATTERN INDEX
EDGE BUNDLE INDEX
EDGE TYPE
EDGE WIDTH
EDGE COLOUR
EDGE VISIBILITY
FILL REFERENCE POINT
PATTERN TABLE
PATTERN SIZE
COLOUR TABLE
ASPECT SOURCE FLAGS
ESCAPE
MESSAGE
APPLICATION DATA

6.3.2.2 Drawing plus control set

The drawing-plus-control set may be used to indicate all of the elements in the drawing set plus additional control, Metafile Descriptor, Picture Descriptor, and attribute elements. It is specified by the shorthand name DRAWING PLUS CONTROL SET.

The elements included in the drawing-plus-control set are all of the elements in the drawing set and the following elements:

INTEGER PRECISION
REAL PRECISION
INDEX PRECISION
COLOUR PRECISION
COLOUR INDEX PRECISION
MAXIMUM COLOUR INDEX
COLOUR VALUE EXTENT
METAFILE DEFAULTS REPLACEMENT
FONT LIST
CHARACTER SET LIST
CHARACTER CODING ANNOUNCER
VDC INTEGER PRECISION
VDC REAL PRECISION
SCALING MODE
LINE WIDTH SPECIFICATION MODE
MARKER SIZE SPECIFICATION MODE
EDGE WIDTH SPECIFICATION MODE
CHARACTER SET INDEX
ALTERNATE CHARACTER SET INDEX

NOTE The drawing-plus-control set essentially constitutes a "Version-1 set", however that designation was not defined in the original ISO 8632:1987 and therefore cannot be used in a Version 1 metafile for reasons of upward compatibility of versions.

6.3.2.3 Version 2 set

The Version-2 set includes all elements from the drawing-plus-control set and the following elements defined in Version 2 metafiles:

BEGIN SEGMENT
END SEGMENT
BEGIN FIGURE
END FIGURE
NAME PRECISION
MAXIMUM VDC EXTENT
SEGMENT PRIORITY EXTENT
DEVICE VIEWPORT
DEVICE VIEWPORT MAPPING
DEVICE VIEWPORT SPECIFICATION MODE
MARKER CLIPPING MODE
EDGE CLIPPING MODE
NEW REGION
SAVE PRIMITIVE CONTEXT
RESTORE PRIMITIVE CONTEXT
CIRCULAR ARC CENTRE REVERSED
CONNECTING EDGE
PICK IDENTIFIER
COPY SEGMENT
INHERITANCE FILTER

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

LINE REPRESENTATION
MARKER REPRESENTATION
TEXT REPRESENTATION
FILL REPRESENTATION
EDGE REPRESENTATION
LINE CLIPPING MODE
CLIP INHERITANCE
SEGMENT TRANSFORMATION
SEGMENT HIGHLIGHTING
SEGMENT DISPLAY PRIORITY
SEGMENT PICK PRIORITY

6.3.2.4 Extended primitives set

The extended-primitives set may be used to indicate those primitives which are available in Version 1 metafiles but are not defined in ISO 7942 (GKS). These elements are:

DISJOINT POLYLINE
RESTRICTED TEXT
APPEND TEXT
POLYGON SET
RECTANGLE
CIRCLE
CIRCULAR ARC 3 POINT
CIRCULAR ARC CENTRE
CIRCULAR ARC CENTRE CLOSE
CIRCULAR ARC CENTRE REVERSED
ELLIPSE
ELLIPTICAL ARC
ELLIPTICAL ARC CLOSE
CIRCULAR ARC 3 POINT CLOSE

6.3.2.5 Version 2 GKSM set

The Version-2-GKSM set is a set of Version 2 metafile elements for ISO 7942 (GKS) picture capture. The elements included in the Version-2-GKSM set are:

BEGIN METAFILE
BEGIN PICTURE
BEGIN PICTURE BODY
END PICTURE
BEGIN SEGMENT
END SEGMENT
END METAFILE
METAFILE VERSION
METAFILE DESCRIPTION
VDC TYPE
INTEGER PRECISION
REAL PRECISION
INDEX PRECISION
COLOUR PRECISION
COLOUR INDEX PRECISION
NAME PRECISION
MAXIMUM COLOUR INDEX
COLOUR VALUE EXTENT
METAFILE ELEMENT LIST
METAFILE DEFAULTS REPLACEMENT
FONT LIST
CHARACTER SET LIST
CHARACTER CODING ANNOUNCER
MAXIMUM VDC EXTENT
SEGMENT PRIORITY EXTENT
VDC EXTENT
DEVICE VIEWPORT
DEVICE VIEWPORT MAPPING
DEVICE VIEWPORT SPECIFICATION MODE
LINE REPRESENTATION
MARKER REPRESENTATION
TEXT REPRESENTATION
FILL REPRESENTATION
VDC INTEGER PRECISION
VDC REAL PRECISION
CLIP RECTANGLE
CELL ARRAY
GDP
LINE BUNDLE INDEX
LINE TYPE
LINE WIDTH
LINE COLOUR
MARKER BUNDLE INDEX
MARKER TYPE
MARKER SIZE
MARKER COLOUR
TEXT BUNDLE INDEX
TEXT FONT INDEX
TEXT PRECISION
CHARACTER EXPANSION FACTOR
CHARACTER SPACING
TEXT COLOUR
CHARACTER HEIGHT
CHARACTER ORIENTATION
TEXT PATH
TEXT ALIGNMENT
CHARACTER SET INDEX
ALTERNATE CHARACTER SET INDEX
FILL BUNDLE INDEX
INTERIOR STYLE
FILL COLOUR
HATCH INDEX
PATTERN INDEX
FILL REFERENCE POINT
PATTERN TABLE
PATTERN SIZE
COLOUR TABLE
ASPECT SOURCE FLAGS
PICK IDENTIFIER
ESCAPE
MESSAGE
APPLICATION DATA

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

POLYLINE
POLYMARKER
TEXT
POLYGON
SEGMENT TRANSFORMATION
SEGMENT HIGHLIGHTING
SEGMENT DISPLAY PRIORITY
SEGMENT PICK PRIORITY

6.3.2.6 Version 3 set

The Version-3 set may be used to indicate all elements in the Version-2 set and the elements:

BEGIN PROTECTION REGION
END PROTECTION REGION
BEGIN COMPOUND LINE
END COMPOUND LINE
BEGIN COMPOUND TEXT PATH
END COMPOUND TEXT PATH
BEGIN TILE ARRAY
END TILE ARRAY
COLOUR MODEL
COLOUR CALIBRATION
FONT PROPERTIES
GLYPH MAPPING
SYMBOL LIBRARY LIST
INTERIOR STYLE SPECIFICATION MODE
LINE AND EDGE TYPE DEFINITION
HATCH STYLE DEFINITION
GEOMETRIC PATTERN DEFINITION
PROTECTION REGION INDICATOR
GENERALIZED TEXT PATH MODE
MITRE LIMIT
TRANSPARENT CELL COLOUR
HYPERBOLIC ARC
PARABOLIC ARC
NON-UNIFORM B-SPLINE
NON-UNIFORM RATIONAL B-SPLINE
POLYBEZIER
POLYSYMBOL
BITONAL TILE
TILE
LINE CAP
LINE JOIN
LINE TYPE CONTINUATION
LINE TYPE INITIAL OFFSET
TEXT SCORE TYPE
RESTRICTED TEXT TYPE
INTERPOLATED INTERIOR
EDGE CAP
EDGE JOIN
EDGE TYPE CONTINUATION
EDGE TYPE INITIAL OFFSET
SYMBOL LIBRARY INDEX
SYMBOL COLOUR
SYMBOL SIZE
SYMBOL ORIENTATION

6.3.2.7 Version 4 set

The Version 4 set may be used to indicate all the elements in Version 3 and the elements:

PICTURE DIRECTORY
APPLICATION STRUCTURE DIRECTORY
BEGIN APPLICATION STRUCTURE
BEGIN APPLICATION STRUCTURE BODY
END APPLICATION STRUCTURE
APPLICATION STRUCTURE ATTRIBUTE

6.3.3 Default metafile state

The default state is the state to which the interpreter is returned at the start of each picture. The default states of all metafile elements are defined in clause 8. These default values may be selectively replaced by using the METAFILE DEFAULTS REPLACEMENT element. The correspondence between character set indexes and registered or private character sets, and the meaning assigned to text font indexes, are also established in the Metafile Descriptor.

6.3.4 Fonts and character sets

6.3.4.1 Font list and font resources

ISO/IEC 9541-1 defines an architecture for font resources, but does not define or standardize applications' use of the information in a font resource — ranging from gross or aggregate properties such as font posture to very specific and detailed properties such as individual glyph metrics. A metafile generator (with its associated application) will be a user of such font resource information. The application, in defining a picture which contains text strings, has knowledge of the properties of the font resource. It makes use of these properties to format or layout strings of text so that the complete strings have the desired characteristics.

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

CGM is used to transmit such pictures from a generating application to an interpreting application, possibly remote in time and space and possibly of very different architecture and resource availability. The font facilities of CGM are designed to provide a font referencing mechanism. Font referencing is the process of identifying or characterizing a font resource. Referencing may include identification of a specific font by name, or provide sufficient descriptive information to permit identification of a suitable font or substitute. This concept is described in ISO/IEC 9541-1, annex B.

The FONT LIST element of CGM allows the exact naming of a font resource. Such font resources may in the future be registered and given structured names under the mechanisms of ISO/IEC 9541. In the ideal case the metafile interpreter recognizes and has available the font resource named in the FONT LIST. For cases where the named font is not available to the interpreter, the CGM has elements (FONT PROPERTIES and GLYPH MAPPING) which allow generators to pass to interpreters additional descriptive information about desired fonts and font resources. An alternative font can be selected by an interpreter through this descriptive information if the specified one is not available.

6.3.4.2 Font properties

The FONT PROPERTIES element can be used to guide selection of a best fit font if an exact match is not available on a specific device. The font properties which may appear are those in the Minimum Font Description Subset of ISO/IEC 9541-2. Applications may use registered extensions to access additional properties from amongst the ISO/IEC 9541 font properties.

NOTE Registration of additional font properties is done using the registration procedures of the ISO International Register of Graphical Items, as described in ISO/IEC TR9973.

The element allows the importance of each property to be assigned a priority relative to the other properties. In the case that a font named in the FONT LIST is not available, the priorities of the properties instruct the interpreter of the relative importance of the various characteristics of the requested font. In some cases it may not even be necessary to get a particular font, but rather any font with certain characteristics — boldness, presence of serif, etc. The FONT PROPERTIES element enables generators to specify such concepts. The use of this information by interpreters is not standardized.

6.3.4.3 Character set repertoire in graphical text strings

The CHARACTER SET LIST element allows designation of registered character sets for use in the metafile. It also allows designation of private character sets, and in combination with the GLYPH MAPPING element allows designation of collections of glyphs which are registered (according to the procedures of ISO/IEC 10036).

Format effector control characters (NUL, CR, LF, BS, HT, VT, and FF) are permitted in parameters of type String but their interpretation is implementation dependent. Control characters used for character set invocation and designation are permitted according to the setting of CHARACTER CODING ANNOUNCER. The other control codes from the C0 and C1 sets are reserved for future standardization.

NOTE SI and SO are only defined and usable with 7-bit coding.

The mechanisms described in this section for character set definition, and the mechanisms for accessing the character sets (CHARACTER CODING ANNOUNCER, CHARACTER SET INDEX, and ALTERNATE CHARACTER SET INDEX) apply only to string parameters (S) of graphical text elements (TEXT, RESTRICTED TEXT, APPEND TEXT, and appropriate GDPs). They do not apply to non-graphical text strings (SF) such as the *metafile identifier* parameter of the BEGIN METAFILE element.

For designation of ISO registered character sets, two pieces of information are specified by the parameters: a character set type, and the tail of its designating sequence by which it is known in the ISO register.

There are five types of character sets: 94-character G-sets, 96-character G-sets, 94-character multibyte G-sets, 96-character multibyte G-sets, and character sets intended to be designated as "complete codes".

94-CHARACTER G-SETS. These character sets are designated by IS0 2022 escape sequences of the form <ESC> <I1> <I>(o) <F>. Here, <I1> is either 2/8, 2/9, 2/10, or 2/11; <I>(o) represents zero or more intermediate characters from column 2 of the code chart; and <F> is a final character from columns 3 through 7 of the code chart. If <F> is from column 3 of the code chart, the character set is a "private" character set. If <F> is from columns 4 through 7 of the code chart, the character set is a "standard" character set in the sense that it and its

AVA Documents Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

designating escape sequences are registered in the International Register Of Coded Character Sets To Be Used With Escape Sequences.

For 94-character G-sets, the character set declaration consists of ‘94-character G-set', followed by a string consisting of all characters in the IS0 2022 designating escape sequence except the first two characters, <ESC> <I1>.

For example, the G-set from the U.K.'s national 7-bit character set is registered in the International Register Of Coded Character Sets To Be Used With Escape Sequences. Its designating escape sequences are as follows:

<ESC>	2/8	4/1	{to designate it as G0}
<ESC>	2/9	4/1	{to designate it as G1}
<ESC>	2/10	4/1	{to designate it as G2}
<ESC>	2/11	4/1	{to designate it as G3}

Again, the French character set (1982 version, from the 1982 version of AFNOR NF Z 62-010) is registered in the International Register Of Coded Character Sets To Be Used With Escape Sequences. Its designating escape sequences are as follows:

<ESC>	2/8	6/6	{to designate it as G0}
<ESC>	2/9	6/6	{to designate it as G1}
<ESC>	2/10	6/6	{to designate it as G2}
<ESC>	2/11	6/6	{to designate it as G3}

Therefore, a CHARACTER SET LIST element could specify that the U.K. character set is to be referred to by character set index 1, and the French character set by character set index 2, as follows:

<CHARACTER-SET-LIST: U.K., French>

‘94-character G-set' 4/1
‘94-character G-set' 6/6

96-CHARACTER G-SETS. These character sets are similar to 94-character G-sets, but include the code positions 2/0 and 7/15, which are excluded from 94-character G-sets. Their IS0 2022 designating escape sequences take the form <ESC> <I1> <I>(o) <F>, where the first intermediate character <I1> is either 2/13, 2/14, or 2/15. The remainder of the escape sequence is similar to the escape sequences for 94-character G-sets: zero or more intermediate characters from column 2 of the code chart and a final character from columns 3 through 7 of the code chart.

For 96-character G-sets, the character set declaration consists of ‘96-character G-set', followed by a string consisting of all characters in the IS0 2022 designating escape sequence except the first two characters, <ESC> <I1>.

It is possible for interchanging parties to agree on a private 96-character G-set whose designating escape sequences would end with a character from column 3 of the code chart. For example, the following might be private escape sequences to designate such a G-set:

<ESC>	2/13	3/0	{to designate it as G1}
<ESC>	2/14	3/0	{to designate it as G2}
<ESC>	2/15	3/0	{to designate it as G3}

(96-character G-sets may not be designated as G0 sets.)

For example, the following CHARACTER SET LIST element establishes the U.K. 94-character G-set, the French 94-character G-set, and a private 96-character G-set as the character sets named by character set indexes 1, 2, and 3, respectively:

<CHARACTER-SET-LIST: U.K., French, private 96-character G-set>

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

‘94-character G-set' 4/1
‘94-character G-set' 6/6
‘96-character G-set' 3/0

94-CHARACTER MULTIBYTE G-SETS. A 94-character multibyte G-set can contain 94 to the Nth power characters, each coded as a sequence of N bytes from columns 2 through 7 of the code chart, not including the bytes 2/0 and 7/15, which are excluded from 94-character G-sets. For example, a 94-character 2-byte G-set can contain 8,836 characters.

The IS0 2022 designating escape sequences for 94-character multibyte G-sets have the following forms:

<ESC>	2/4	2/8	<F>	{to designate it as G0}
<ESC>	2/4	2/9	<F>	{to designate it as G1}
<ESC>	2/4	2/10	<F>	{to designate it as G2}
<ESC>	2/4	2/11	<F>	{to designate it as G3}

For compatibility with the first version of ISO 2022, when <f> is 4/0, 4/1, or 4/2 the byte 2/8 may be left out of the designating escape sequence as a GO set.

For 94-character multibyte G-sets, the character set declaration consists of ‘94-character multibyte G-set', followed by a string consisting only of the final character in the IS0 2022 designating escape sequence.

For example, a Japanese 2-byte character set of 6802 graphic characters has been registered in the International Register Of Coded Character Sets To Be Used With Escape Sequences, and its designating escape sequences have the form shown above, with the final character <F> being 4/0. Thus, the following CHARACTER SET LIST element could be used to specify that this 2-byte Japanese character set is to be referred to by character set index 1:

<CHARACTER-SET-LIST: Japanese 2-byte character set>

‘94-character multibyte G-set' 4/0

96-CHARACTER MULTIBYTE G-SETS. A 96-character multibyte G-set is similar to a 94-character multibyte G-set except that it can include the bytes 2/0 and 7/15. Thus, a 96-character 2-byte G-set could have 96 times 96 (or 9216) 2-byte character codes.

The ISO 2022 designating escape sequences for 96-character multibyte G-sets have the following forms:

<ESC>	2/4	2/13	<F>	{to designate it as G1}
<ESC>	2/4	2/14	<F>	{to designate it as G2}
<ESC>	2/4	2/15	<F>	{to designate it as G3}

It is not possible to designate a 96-character multibyte G-set as a G0 set.

The character set declaration for a 96-character multibyte G-set consists of ‘96-character multibyte G-set', followed by a string consisting only of the final character <F> in the character set's ISO 2022 designating escape sequence.

At the time of this publication, no 96-character multibyte G-sets have been registered in the International Register of Coded Character Sets To Be Used with Escape Sequences.

CHARACTER SETS INTENDED TO BE DESIGNATED AS COMPLETE CODES. Other character sets may not fit the ISO 2022 "G-set" structure. ISO 2022 provides an escape sequence format for invoking coding systems different from ISO 2022. The complete code escape sequences have the following form

<ESC> 2/5 <I>o <F>

where <I>o means "zero or more characters from column 2 of the code chart", and <F> is a final character from columns 3 through 7 of the code chart. If <F> is from column 3, the coding system is a private code. If <F> is from columns 4 through 7, it is a code for which a designating and invoking escape sequence has been registered in the International Register Of Coded Character Sets To Be Used With Escape Sequences.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

The character set declaration for a character set that would be invoked as a coding system different from ISO 2022 consists of 'complete code' followed by a string consisting only of those characters in the code's ISO 2022 escape sequence which come after the first two characters, <ESC> 2/5.

As well as using a registered complete code, a private code could be used. For example, suppose the interchanging parties have agreed on a private 8-bit code to be invoked by the following escape sequence:

<ESC> 2/5 3/0

The following CHARACTER SET LIST element would declare the French character set to have character set index 1 and that 8-bit private code to have character set index 2:

<CHARACTER-SET-LIST: French, private coding system>

'94-character G-set' 6/6
'complete code' 3/0

Information regarding the designation sequence tail parameter is found in the International Register of Coded Character Sets to be Used with Escape Sequences. This register is maintained by the Registration Authority for ISO 2375, which is the European Computer Manufacturers Association (ECMA), Rue du Rhone 114, CH-1204, Geneva, Switzerland.

6.3.4.4 Glyph repertoire

ISO/IEC 10036 specifies a procedure and a registrar (registering authority) for registering typographic glyph collections. There currently is no standard that associates codes (i.e., character codes) with these glyphs. However the registrar — the Association for Font Information Interchange, or AFII — assigns a unique 4-byte integer indentifier with each glyph.

This part of ISO/IEC 8632 defines a means to access these registered glyph collections for use in graphical text strings. The GLYPH MAPPING element associates the AFII 4-byte identifiers with single-byte or multi-byte codes. A set of such codes is defined as a collection, forming a locally defined character set for use within the metafile. The local character set is associated with an index, and within the body of the CGM the normal character set access and switching mechanisms (based upon, and adapted from, ISO 2022) may be used to access the AFII registered glyphs within CGM text strings.

NOTE1 The glyph complement is a property of a font resource in the ISO/IEC 9541 font architecture. When the separate mechanisms of ISO/IEC 8632 for font reference and glyph access are used there is potential for incompatibility between the specifications — the requested glyph complement may not be representable in the requested font.

6.3.4.5 Character set selection in non-graphical text strings

The mechanisms discussed in the preceding sections apply to graphical text strings — text strings which result in display as graphical entities. These are the parameters of TEXT, RESTRICTED TEXT, APPEND TEXT, and possibly GDP elements.

They do not apply to other, non-graphical text strings which occur as metafile parameters. See 7.1**,** type String Fixed (SF). For these strings, character set selection shall be accomplished only through use of embedded ISO 2022 controls, to switch sets within the string.

For the purposes of coding type SF parameters in ISO/IEC 8632 metafiles, 8-bit coding environment shall be used, by default. It is possible using ISO 2022 controls to select other coding environments, such as multi-byte complete codes.

In SF parameters, therefore, by default the two character sets which are currently designated as GO and G1 may be accessed by using respectively the code values 32..127 and 160..255.

BEGIN METAFILE causes ISO 8859-1 Left Hand Side to be designated as the G0 set and ISO 8859-1, Right Hand Side of Latin Alphabet Nr. 1 to be designated as the G1 set.

EXAMPLE 1 — If the following 4-octet ISO 2022 sequence were to occur as the first 4 octets of the 'metafile id' parameter of the BEGIN METAFILE element,

Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

ESC 02/05 02/15 04/09,

then the character set for subsequent type SF character data would be the variable length multi-byte UTF-8 transformation of Unicode (ISO 10646).

EXAMPLE 2 — The ISO 2022 designating sequence

ESC 2/8 4/2, for example, would designate

ISO 646, U.S. National Character Set (ASCII) as the G0 set (the default set).

EXAMPLE 3 — Similarly, the ISO 2022 designating sequence

ESC 2/13 4/2

would designate ASCII as the G1 set, superceding the default G1 set (ISO 8859-1) — character codes following an SI would then refer to ASCII, not ISO 8859-1.

Format effector control characters (NUL, CR, LF, BS, HT, VT, and FF) are permitted in a parameters of type String Fixed but their interpretation is implementation dependent. Control characters used for character set invocation and designation (SI, SO, ESC, SS2, and SS3) are permitted according to the setting of CHARACTER CODING ANNOUNCER. The other control codes from the C0 and C1 sets are reserved for future standardization.

6.3.4.6 Revision of character sets

ISO 2022:1986 allows for the revision of character sets. It states: ESC 2/6 F, when used, shall immediately precede a designating escape sequence and indicate a revision of a registered set. F will specify the revision number 1 to 63 by taking values 4/0 to 7/14 respectively. Revisions are only permitted to add a character or characters to a set and shall be submitted to the Registration Authority as required by ISO 2375, pointing out that the submission is a revision of a registered set.

6.3.5 Picture directory

The PICTURE DIRECTORY element contains locations of the pictures in a metafile. The picture directory contains a list of picture identifiers and their locations relative to the BEGIN METAFILE element. The picture directory may optionally contain the location of the APPLICATION STRUCTURE DIRECTORY element in each picture relative to the BEGIN METAFILE element. An interpreter wishing to randomly address a picture in a metafile with a picture directory need only interpret the Metafile Descriptor and skip to the locations of the picture specified in the PICTURE DIRECTORY. The next element read will be the desired BEGIN PICTURE.

6.4 Picture descriptor elements

Picture Descriptor elements include elements to declare the parameter modes of other elements for an entire picture, to configure that portion of coordinate space that is of interest in the picture, and to set the colour to which the view surface is cleared at the start of the picture.These elements are:

SCALING MODE	LINE REPRESENTATION
COLOUR SELECTION MODE	MARKER REPRESENTATION
LINE WIDTH SPECIFICATION MODE	TEXT REPRESENTATION
MARKER SIZE SPECIFICATION MODE	FILL REPRESENTATION
EDGE WIDTH SPECIFICATION MODE	EDGE REPRESENTATION
VDC EXTENT	INTERIOR STYLE SPECIFICATION MODE
BACKGROUND COLOUR	LINE AND EDGE TYPE DEFINITION
DEVICE VIEWPORT	HATCH STYLE DEFINITION
DEVICE VIEWPORT SPECIFICATION MODE	GEOMETRIC PATTERN DEFINITION
DEVICE VIEWPORT MAPPING	APPLICATION STRUCTURE DIRECTORY

In Version 1 metafiles, any Picture Descriptor elements within a picture shall appear after the BEGIN PICTURE element and before the BEGIN PICTURE BODY element. Escape and external elements are permitted in the Picture Descriptor.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

In Version 2, Version 3 and Version 4 metafiles, some of the Picture Descriptor elements may appear within the picture body. Table 8a indicates which Picture Descriptor elements may appear and where they may appear for each metafile version.

6.4.1 Scaling mode

VDC space may be either an abstract space, which may be mapped to an arbitrary size on a physical device, or a metric space, which is intended to be mapped to a particular size. Selection of the mode to be used can be made on a picture-by-picture basis by means of the SCALING MODE element. The scaling mode element provides a flag to select abstract space or metric space, and a scale factor which specifies the number of millimeters per VDC unit when metric space is selected.

6.4.2 Colour selection mode

COLOUR SELECTION MODE selects either indexed or direct colour specification and is described further under colour attributes, 6.7.6. For Version 1 metafiles, the selection is for the whole picture.

6.4.3 Specification modes

Line width, marker size, interiors of filled-area elements, and edge width may be specified in more than one way. The width of lines, for example, may be specified as either a measure in VDC units, a scale factor to be applied to a device-dependent nominal line width at interpretation time, a fraction of the device view surface, or a measure in millimetres. For each attribute element having such multiple modes, there is an associated control element that defines the mode of the parameter of the attribute element.

For each of the four values -- scaled, absolute, mm, fractional -- of the line width specification mode, there is an associated current value of the line width attribute in that mode. This current value is defined either by the last occurrence of a LINE WIDTH attribute element in that mode, or by the default line width for that mode. The same is true for markers, edges, and filled-area interiors -- there are conceptually four current values of each of these attributes, one associated with the each of the four values of the appropriate specification mode element.

6.4.4 VDC extent

There is a metafile element to define the VDC extent. The extent is set with the VDC EXTENT element by specifying the addresses (in VDC) of the lower-left corner and the upper-right corner of this extent as seen by the viewer of the picture. Specification of values outside the VDC extent is permitted in CGM elements.

It is intended that the visible portion of an image be contained within the VDC extent. It thus provides a frame for the region of interest in a picture. The values of the coordinates for either dimension may be either increasing or decreasing from the lower-left to the upper-right corner. For example, for devices with an upper-left origin, a picture may be described in coordinates that map directly to the device but still may be displayed correctly on a device with a lower-left origin. Figure 1 illustrates these concepts.

The VDC extent thus establishes the sense and orientation of VDC space (that is, the directions of the positive x (+x) and positive y (+y) axes, and whether the +y axis is 90° clockwise or 90° counterclockwise from the +x axis). In particular, VDC EXTENT establishes the direction of positive and negative angles as follows: positive 90° is defined to be the right angle from the positive x-axis to the positive y-axis (see figure 1).

Note that some attributes such as text attributes (for example, the directions of the ‘up' and ‘base' component vectors of CHARACTER ORIENTATION, and therefore the meaning of the enumerative values ‘right', ‘left', ‘up', ‘down') are intimately bound to these definitions.

The default state of the extent is specified in clause 8 and can be changed in the METAFILE DEFAULTS REPLACEMENT element in the Metafile Descriptor. VDC EXTENT returns to this default state at the beginning of each picture. MAXIMUM VDC EXTENT defines an extent which bounds the VDC extent values which may be found in the metafile. It may be, but need not be, a closest bound in the sense that it exactly equals the union of the extent rectangles in the metafile. This element may be used, for example, to map integer virtual device coordinates of the metafile to a unit square in a normalized device space.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

6.4.5 CGM tailoring

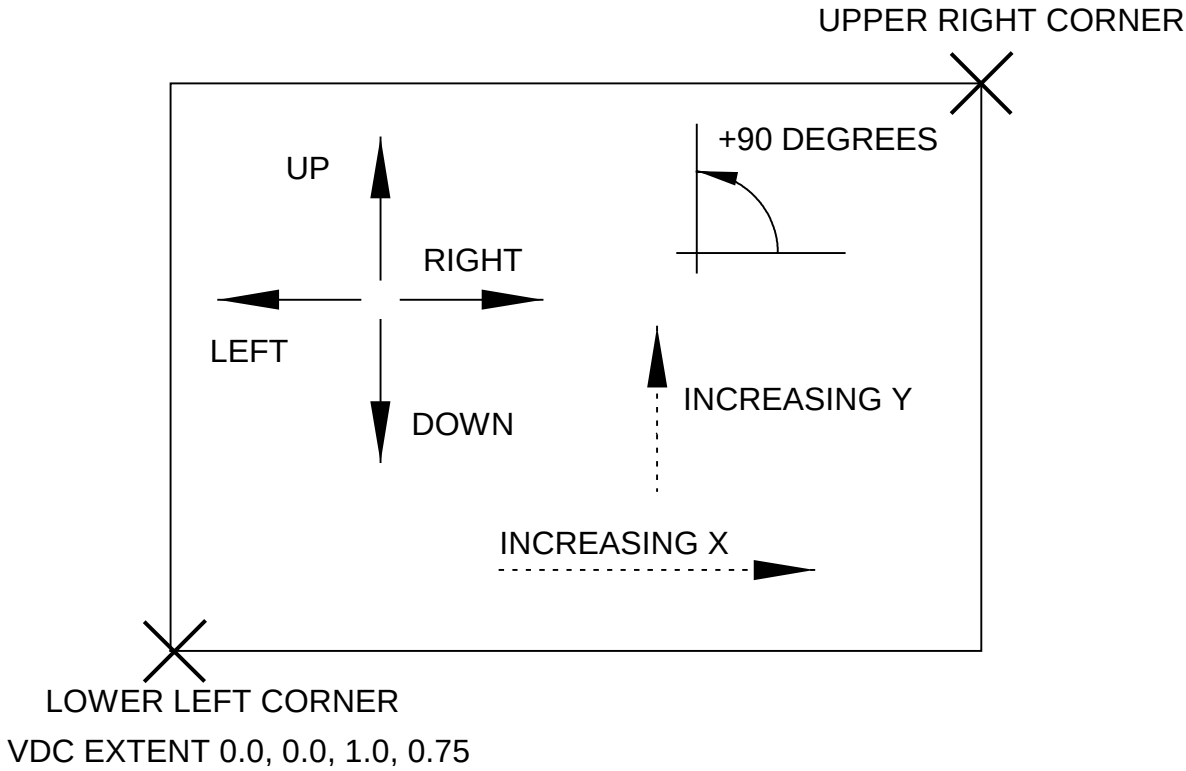
The ability to specify the VDC range and the VDC extent provides the flexibility to configure the metafile addressability in any way desired. It can be configured as an abstract, normalized address range for maximum device independence. It can also be configured to mimic the addressability of a particular target device in order to take advantage of particular device characteristics. The address range of such a device-specific metafile is just another normalized address range with the normalization limits inherent in the VDC-customizing element; therefore, device independence is maintained.

Such tailoring of the coordinates in a metafile can eliminate the need for transformation of coordinates at metafile interpretation time for the target device. The ability to specify the VDC extent thus allows for the exact registration of coordinates in a metafile with addressable points on the target graphics device.

The use of VDC EXTENT to directly encode world coordinates of large dynamic range and very small granularity will likely result in performance penalties at metafile interpretation time, and may result in decreased portability if such VDC extents exceed those compatible with less capable metafile interpreters.

In addition to VDC tailoring, a metafile generator can limit or tailor the functional content of a metafile to accomodate particular devices or applications, and announce such functional tailoring through the use of METAFILE ELEMENT LIST.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999



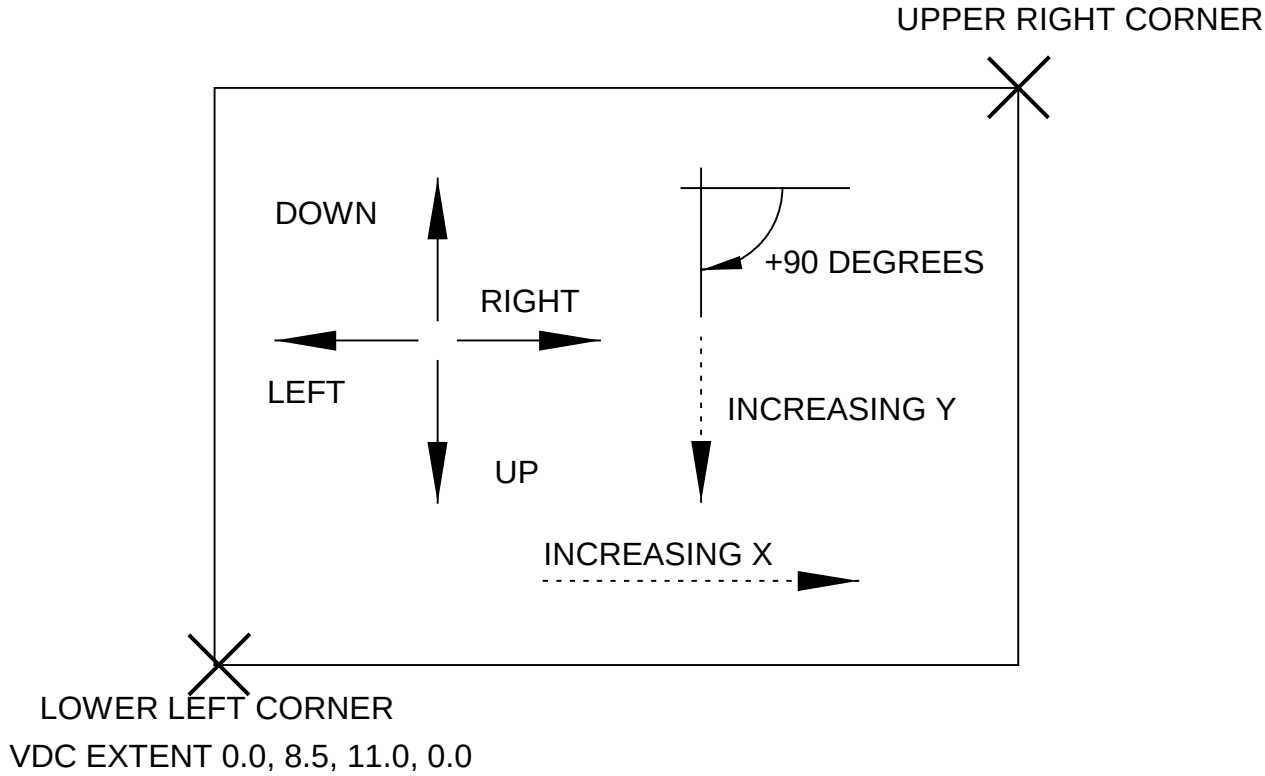


Figure 1 — VDC EXTENT establishes the direction of positive and negative angles

AVA Documents
Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

6.4.6 Background colour

Each picture defines a graphical image that is independent of the other images in a metafile.

The background colour of the image may be specified by the BACKGROUND COLOUR Picture Descriptor element. It may also be specified by the COLOUR TABLE definition of colour index 0 in the METAFILE DEFAULTS REPLACEMENT (metafiles of all versions), or by the COLOUR TABLE definition of colour index 0 in the Picture Descriptor (metafiles of Versions 2 and higher). Definition of colour index 0 within the picture body has no effect on the background colour, only on subsequent primitives displayed with colour index 0 — the background colour is determined and bound at the occurrence of BEGIN PICTURE BODY.

6.4.7 Device viewport control

The device viewport specifies the region of the device display surface into which the VDC extent is to be mapped on interpretation. VDC-to-Device mapping is determinied by the VDC extent, device viewport, and device viewport mapping.

The position of the device viewport is specified in one of three coordinate systems selected by the DEVICE VIEWPORT SPECIFICATION MODE element:

— by fraction [0.0 to 1.0] of the available display surface, which allows reasonable placement and relative sizing of the viewport;

— in millimetres times a scale factor, which allows absolute sizing of images;

— in physical device coordinates.

The device viewport is specified in terms of two points on the device display surface at diagonally opposite corners of the rectangle. Mirroring or 180° rotation of the image may be achieved by specifying the corners in some way other than the first as below and to the left of the second.

The DEVICE VIEWPORT MAPPING element may be used to force isotropic mapping even if the specified VDC extent and device viewport would not otherwise have led to one. In such a case, the VDC extent is mapped on to a subset of the specified device viewport on interpretation. This subset is defined by shrinking either the vertical or horizontal dimension of the specified viewport as needed to reach the required aspect ratio. This smaller "effective viewport" is then used to define the coordinate mapping from VDC to the device's coordinates. The placement of the effective viewport rectangle within the original one can be specified. This placement can be one of ‘left', ‘right' or ‘centred' when the shrinking is horizontal, and ‘top', ‘bottom' or ‘centred' when it is vertical. These meanings are relative to the display surface of the device.

The VDC-to-Device mapping maps the first point specifying the VDC extent on to the corner of the effective viewport corresponding to the first point specifying the device viewport, and similarly for the second point. The mapping is linear in each dimension, but is not necessarily isotropic (for example, a circle in VDC may not appear as a circle to the viewer).

Both the way VDC space is oriented relative to the display surface and the way the effective viewport is placed on the physical device may lead to mirroring and 180° rotation.

The behaviour of primitives and attributes with significance in VDC space under transformations is further described in 6.6.

If both device viewport and scaling mode appear in the same metafile then the last specified is used. If neither appears then the default values for device viewport take precedence.

6.4.8 Representations

The elements LINE REPRESENTATION, MARKER REPRESENTATION, TEXT REPRESENTATION, FILL REPRESENTATION and EDGE REPRESENTATION are used to set all of the attribute values in a bundle table entry at the same time. The attributes that may be bundled are described in 6.7.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

6.4.9 Definable attributes

A precise definition of line and edge types as well as hatch styles may be accomplished as described in 6.7. Moreover, geometric patterns may be defined (in addition to patterns which are arrays of colours confined to parallellogram-like cells) using the segment definition mechanism. See 6.7.

6.4.10 Application structure directory

The APPLICATION STRUCTURE DIRECTORY element contains the locations of the various application structures in a picture. The application structure directory contains a list of application structure identifiers and their locations relative to the BEGIN PICTURE element. APSs within a metafile containing multiple pictures can be accessed directly in two steps. First, the picture directory is used to locate a picture and possibly the application structure directory for the picture. Second, the APSs in the picture are located using the application structure directory.

6.5 Control elements

Control elements specify address space, clipping boundaries, and format descriptions of the CGM elements. Control of some of these format descriptions may be accomplished by Metafile Descriptor elements, while control of others is accomplished by control elements, which may appear in the picture bodies in the metafile. Those items in the former category are fixed for a given metafile, while those in the latter category are changeable; that is, they may change within a picture. Some of the control elements may appear in the Picture Descriptor if this is permitted by the formal grammar for the metafile version. See Table 8a.

6.5.1 VDC space and range

The graphical primitive elements of a metafile define virtual images. The coordinates of these elements (that is, the addresses of points in the virtual image) are absolute two-dimensional Virtual Device Coordinates (VDC). VDC space is a two-dimensional coordinate space of infinite precision and infinite extent. Only a subset of VDC space, the VDC range, is realizable. The VDC range comprises all coordinates representable in the format specified by the declared VDC TYPE and (depending on the type) the VDC INTEGER PRECISION or VDC REAL PRECISION.

The VDC range cannot be set directly; it is completely determined by VDC TYPE and either VDC INTEGER PRECISION or VDC REAL PRECISION elements in the metafile. These elements are controllable, some by dynamic elements in the metafile body and some by static elements in the Metafile Descriptor. Note that the VDC range thus defined (a rectangular subregion of the VDC space) does not enclose a continuum of values, but has a distinct granularity. Regardless of the aspect ratio of the VDC range and the granularity within the range, it is implicit that one VDC unit in the x-direction represents the same distance as one VDC unit in the y-direction in VDC space.

6.5.2 Clipping

In order to defer clipping of graphical primitive elements (particularly, expandable elements such as CIRCLE, CIRCULAR ARC 3 POINT, TEXT, etc.) until metafile interpretation time, a clipping control feature is provided in the CGM. Clipping control is achieved by defining CLIP RECTANGLE in VDC space. Whether clipping to the limits of CLIP RECTANGLE actually occurs at metafile interpretation time is controlled by the CLIP INDICATOR element that sets the mode of the metafile to ‘on' or ‘off'.

In Version 3 and Version 4 metafiles, primitives may also be clipped against more general regions as defined by BEGIN PROTECTION REGION and END PROTECTION REGION, and as controlled by PROTECTION REGION INDICATOR (see 6.5.4). Clipping effects in the remainder of this section are described in terms of the basic CLIP RECTANGLE capability, but they apply equally to general clip regions.

There are three different clipping modes for lines, markers and edges. The required clipping mode is recorded in the metafile with the elements: LINE CLIPPING MODE, MARKER CLIPPING MODE, and EDGE CLIPPING MODE. When the CLIP INDICATOR associated with a graphical primitive is ‘on', only those parts of a graphical primitive that are considered inside the effective clipping region are rendered on interpretation. The object clipping modes allow precise specification as to how clipping is applied to primitives on interpretation.

Clipping may be either ‘locus', ‘shape' or ‘locus then shape'. Conceptually, a locus is a mathematical object like a point or line segment, while a shape is an area in 2-dimensional space. Loci are 0-, 1- or 2-dimensional subsets of real-valued 2-space. For markers and text, they are points. For lines, they are the individual line segments or

AVA Documents Document Sharing Hub

Full document details: ISO_IEC 8632-1_1999

portions of arcs. The locus of an area is the shape and the boundary. Shapes reflect the realization of geometric attributes and are generally 2-dimensional subsets of real-valued 2-space.

‘Locus' clipping is applied for each portion of a graphical object based on its mathematical location and is independent of the area it will occupy after rendering. For example, no portion of a line segment is rendered if the ideal mathematical line lies outside the effective clipping region (even if its line width would carry some portion of the rendering of it into the clipping rectangle); no portion of a marker is rendered if its location lies outside the clipping rectangle.

If ‘locus' clipping is used, the rendering is applied to the locus of the graphical object after clipping. The resulting rendered shape areas may therefore extend outside the effective clipping region.

‘Shape' clipping is applied after the abstract rendering of shape in device coordinate space. The 2-dimensional point set associated with the graphical object is intersected with the effective clipping region, which has been transformed to device coordinate space.

‘Locus then shape' clipping allows the specification that both ‘locus' and ‘shape' clipping be applied to graphical objects as described above. In this case however, the rendered shape will not extend outside the effective clipping region. A thick line whose locus is outside the clip rectangle will not have any portion visible even if its line width would carry some portion of the rendering inside the clip rectangle.

Figure 2 shows some examples of the effect of the clipping modes.

When a width or size specification mode is ‘scaled', ‘fractional', or ‘mm', the rendering of shape proceeds in device coordinate space after application of the VDC-to-Device mapping.

When a width or size specification mode is ‘absolute', the rendering of shape proceeds, in VDC space before application of the copy transformation and before application of the segment transformation (if the primitive is in a copied segment), and before the VDC-to-Device mapping.

Fill and text primitives do not have associated object clipping modes (though the edge of a fill primitive and the boundary edges of a closed figure do). Clipping for fill primitives is always consistent with ‘shape' clipping (see 6.6.4). For text primitives, the type of clipping is determined by the associated text precision:

— For ‘string' precision text, clipping proceeds, on a per string basis, in a manner consistent with ‘locus' clipping.

— For ‘character' precision text, clipping proceeds, on a per character basis, in a manner consistent with ‘locus' clipping.

— For ‘stroke' precision text, the clipping always proceeds in a manner consistent with ‘shape' clipping.

NOTE It is valid for an interpreter to perform ‘shape' clipping of text in all cases, i.e., ‘stroke' text precision is a valid realization of ‘character' and ‘string' text precision (and ‘character' text precision is a valid realization of ‘string' text precision).

Clip rectangles applied to graphical primitive elements within segments may be subject to transformations in VDC space. Intersection of clip rectangles (untransformed or transformed) may result in polygonal clipping boundaries (see 6.10.5)

6.5.3 Save and restore primitive context

Two elements, SAVE PRIMITIVE CONTEXT and RESTORE PRIMITIVE CONTEXT, are provided to save and restore a context; that is, attributes and control elements as collections. This capability allows a list of attributes and control elements (see 7.5.11) to be stored in the metafile which can be referenced by name at a later point in the metafile. This capability can be used to save and restore attributes and control elements in conjunction with opening and closing segments.

The values for attributes controlled by specification or selection modes are saved in the mode in which they were last specified along with the value of the corresponding mode. In restoring a context, the current specification and selection modes are not changed.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

6.5.4 Compound clipping and shielding

The clipping and shielding elements consist of BEGIN PROTECTION REGION, END PROTECTION REGION, and PROTECTION REGION INDICATOR. The BEGIN PROTECTION REGION and END PROTECTION REGION elements are delimiter elements, and the PROTECTION REGION INDICATOR element is a control element.

Protection regions are identified by an index. Protection regions are constructed by the same primitive elements as closed figures. The interior of a given protection region is defined in the same way as the interior of a closed figure. Regions which are constructed by line elements are closed by NEW REGION, END PROTECTION REGION, or any filled area element. If the endpoints and beginning points of subsequent line elements are not identical they are implicitly connected by a straight line.

If a protection region is used within a segment (i.e., the PROTECTION REGION INDICATOR has the value ‘clip' or ‘shield' for that region inside the segment), it behaves as do clip rectangles with respect to transformations — it transforms by the copy transformation which is associated with the COPY SEGMENT element. Protection regions used in segments are also affected by the segment transformations of the segment.

The effective protection region for clipping and/or shielding, when protection regions are defined, or referenced, within segments, is determined by CLIP INHERITANCE in the same way as for the simpler rectangle clipping (see 6.10.5).

NOTE Inheritance of regions is complicated by the fact that, conceptually, the entire list of regions together with their indicators must be subject to the inheritance mechanism, down to the point in the segment hierarchy where reference to a region is made or the region must be applied for clipping or shielding (in the case of the ‘intersection' value of CLIP INHERITANCE).

If a protection region is defined within a segment, then its definition persists after the END SEGMENT terminating that segment definition. This is exactly as for clip rectangles, and primitive attributes. Thus the definition and activation of a region within a local segment in a picture body potentially has effect in the picture body following the segment. If a segment is copied into the picture body with a COPY SEGMENT function, then region definitions within the segment do not affect primitives following, again exactly as clip rectangle.

A protection region is associated with an index at definition time. If a region is defined with an index that is already in use, then the old definition is deleted. The associated protection region indicator for the newly defined region takes the initial value ‘off'.

When a protection region is set for clipping, only the portions of the graphical elements inside or on the boundary of the protection region are drawn. When a protection region is set for shielding, only the portions of the graphical elements outside the protection region and its boundary are drawn.

Several clip and shield regions may be in effect simultaneously. In this case, only portions of the graphical elements inside or on the boundary of the intersection of all individual clip regions and outside the union of all individual shield regions are drawn. See figure 3.

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999

© ISO/IEC

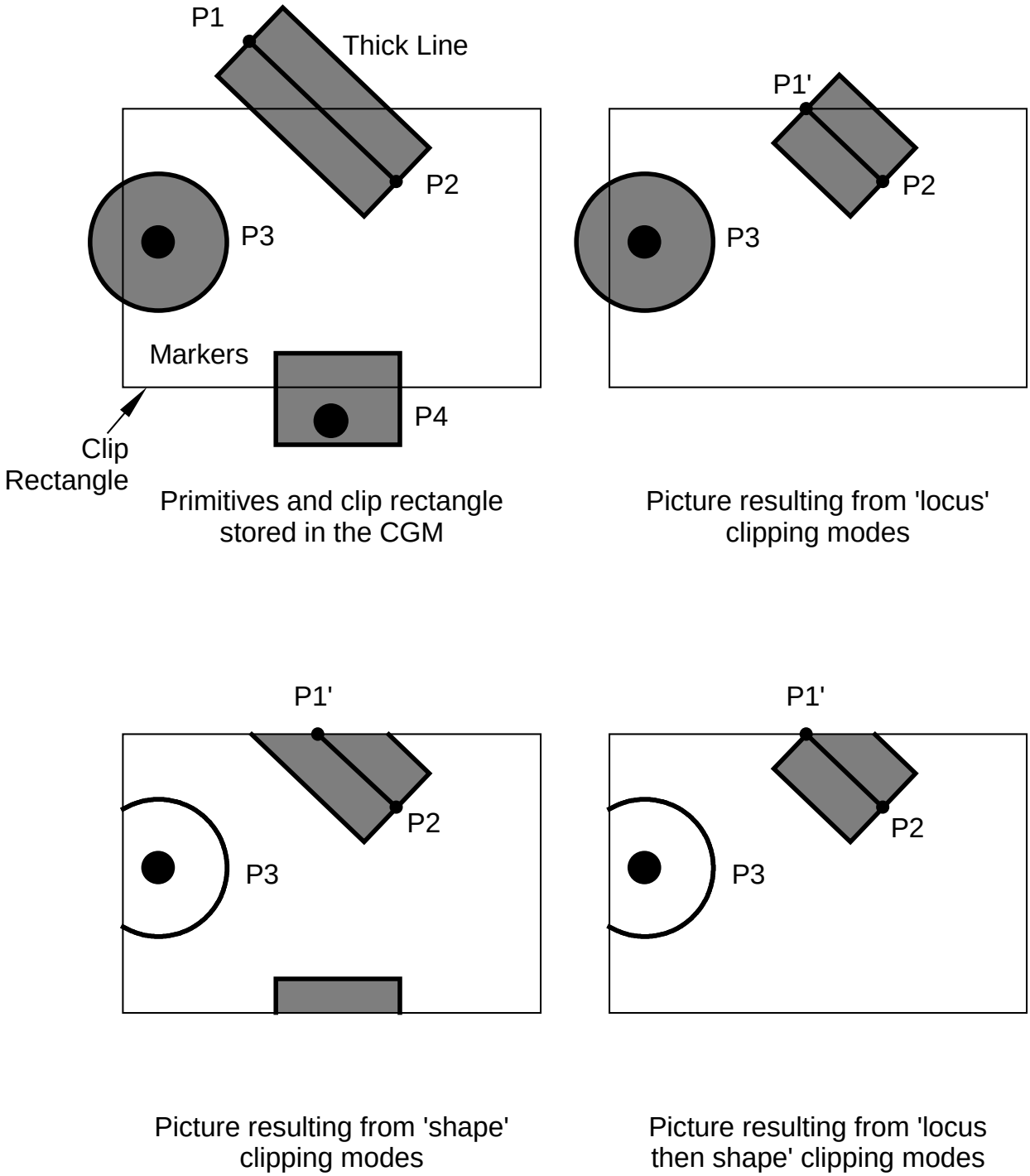


Figure 2 — Clipping modes

Documents
Document Sharing Hub
Full document details: ISO_IEC 8632-1_1999