INTERNATIONAL STANDARD

ISO 8632-1

First edition
1987-08-01

AMENDMENT 1
1990-11-01

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

Part 1 :
Functional specification

AMENDMENT 1

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

Partie 1: Description fonctionnelle

AMENDEMENT 1

Reference number
ISO 8632-1 : 1987/Amd. 1 : 1990 (E)

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.

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.

International Standard ISO 8632-1/Amd. 1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology.*

International Organization for Standardization
Case postale 56 • CH-1211 Genève 20 • Switzerland
Printed in Switzerland

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

Part 1:
Functional specification

AMENDMENT 1

Page 1

Add the following at the end of 0.1:

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.

Page 1

Sub-clause 0.3: Add the following at the end of item c):

It should also not preclude further extensions to support future standards.

Page 1

Sub-clause 0.3:Add the following at the end of item d):

It should include the capability to support ISO 7942 (GKS) static picture-capture.

Page 3

Add the following at the end of 0.8:

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 (currently a Draft International Standard)).

Page 4

Clause 1: Add the following at the end of the first paragraph:

This picture description includes the capability for describing static images.

Page 5

Clause 2: Add the following to the list of references:

ISO/IEC 9636 *Information processing systems - Computer Graphics - Interfacing techniques for dialogues with graphical devices (CGI).* Parts 1-6 (currently a Draft International Standard).

Page 6

Clause 3: Add the following to the list of definitions and abbreviations:

3.1.49 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 post-transformed ellipses. See "isotropic mapping".

3.1.50 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.

3.1.51 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.

3.1.52 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'.

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

3.1.54 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.

3.1.55 compound text: A compound text primitive is formed through the use of APPEND TEXT. There may be attribute changes between portions of the resulting complete text string.

3.1.56 device coordinates: The coordinates native to a device; device-dependent coordinates; physical device coordinates.

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

3.1.58 edge: The rendering of the perimiter 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".

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

3.1.60 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.

3.1.61 global segment: A segment that is defined in the Metafile Descriptor (see "segment"). It may be referenced from within any picture.

3.1.62 graphic object: A graphic object is a graphic primitive, including a compound primitive, together with the associated attributes.

3.1.63 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".

3.1.64 local segment: A segment whose definition is local to the picture in which it appears.

3.1.65 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.

3.1.66 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.

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

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

3.1.69 size specification mode: A generic term for Line Width Specification Mode, Edge Width Specification Mode, or Marker Size Specification Mode. A size specification mode may be 'absolute' or 'scaled', the latter being referenced to a nominal size in device coordinate space.

3.1.70 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.

Page 7

Sub-clause 3.1.26: Definition of graphical elements

Insert "primitive" between "graphical" and "element".

Page 9

Sub-clause 4.1: Add the following at the end of the list of classes of elements:

Segment Elements, which enable the grouping and manipulation of elements.

Page 9

Sub-clause 4.1: Add the following after the third paragraph:

Graphical output primitives and attributes may be grouped in segments. Segment attribute elements control the appearance of segments.

Page 10

Sub-clause 4.2: Add the following at the end:

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.

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 or within picture bodies.

Page 10

Sub-clause 4.3: Add the following to the list after the first paragraph:

NAME PRECISION
MAXIMUM VDC EXTENT
SEGMENT PRIORITY EXTENT

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

Page 10

Add the following paragraph at the end of 4.3:

METAFILE VERSION and METAFILE ELEMENT LIST shall occur only once in the Metafile Descriptor for version 2 metafiles. It is recommended that they shall only appear once in version 1 metafiles.

NOTE - 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.

Page 10

Sub-clause 4.3.2 : Change the start of the third sentence from "Two shorthand names....." to:

Several shorthand names.....

Page 11

Add the following after 4.3.2.2:

4.3.2.3 Version 2 set

The Version-2 set may be used to indicate all the elements in the drawing-plus-control set and all the additional elements defined in this part of ISO/IEC 8632.

4.3.2.4 Extended primitives set

The extended-primitives set may be used to indicate those primitives which 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 3 POINT CLOSE
CIRCULAR ARC CENTRE
CIRCULAR ARC CENTRE CLOSE
CIRCULAR ARC CENTRE REVERSED
ELLIPSE
ELLIPTICAL ARC
ELLIPTICAL ARC CLOSE
CONNECTING EDGE

4.3.2.5 Version 2 GKSM set

The Version-2-GKSM set includes 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
POLYLINE
POLYMARKER
TEXT
POLYGON
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
SEGMENT TRANSFORMATION
SEGMENT HIGHLIGHTING
SEGMENT DISPLAY PRIORITY
SEGMENT PICK PRIORITY

Page 12

Sub-clause 4.4. Add the following text at the end of the first paragraph:

Some of the picture descriptor elements may appear outside the Picture Descriptor if this is permitted by the formal grammar for the metafile version. In such a case they do not set parameter values to apply for the entire picture.

Page 12

Sub-clause 4.4.2. Change the text to the following:

COLOUR SELECTION MODE selects either indexed or direct (RGB) colour specification and is described further under colour attributes. For version 1 metafiles the selection is for the whole picture.

Page 12

Add the following paragraph at the end of 4.4.4:

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.

Page 14

Add the following after 4.4.6:

4.4.7 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^{o} 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^{o} rotation.

The behaviour of primitives and attributes with significance in VDC space under transformations is further described in 4.6.

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

4.4.8 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 4.7.

Page 14

Add the following at the end of 4.5:

Some of the control elements may appear in the Picture Descriptor if this is permitted by the formal grammar for the metafile version.

Page 15

Add the following text at the end of 4.5.2:

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 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 graphic 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 graphic 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 graphic 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 graphic 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 1a shows some examples of the effect of the clipping modes.

When a width or size specification mode is 'scaled', 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, conceptually, in VDC space before application of the copy transformation, before application of the segment transformation 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 4.6.4.5). 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 - 'shape' clipping for all text precisions is always allowed by this part of ISO/IEC 8632.

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 4.12.5).

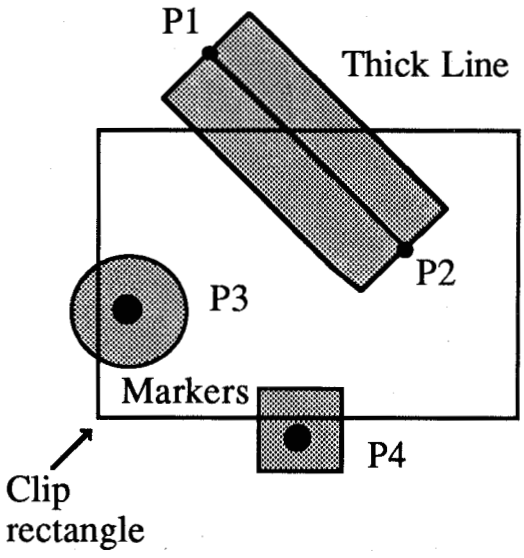
Page 15

Add the following after 4.5.2

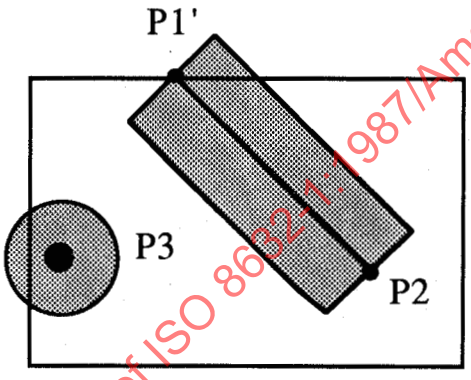
4.5.3 Save and restore primitive context

Two elements 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 5.5.11) 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.

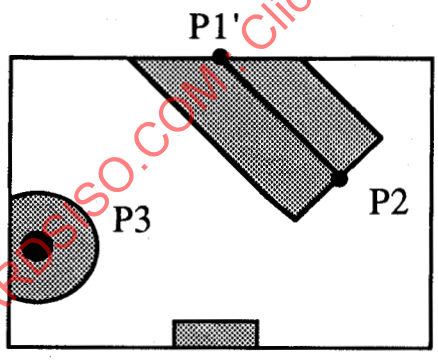
The values for attributes controlled by specification or selection modes are saved in the mode in which they were last specified along with the value of the corresponding mode. In restoring a context the current specification and selection modes are not changed.



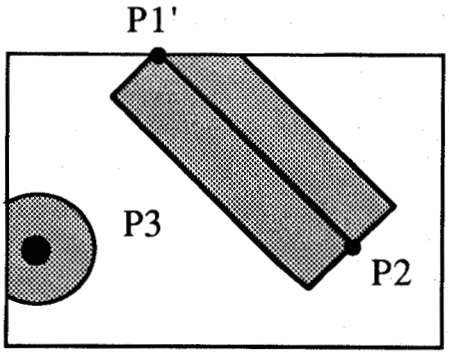
Primitives and clip rectangle stored in the CGM



Picture resulting from 'locus' clipping modes



Picture resulting from 'shape' clipping modes



Picture resulting from 'locus then shape' clipping modes

Figure 1a - Examples of the effects of object clipping modes

Page 15

Add the following to the list of graphical primitive elements and to the list of line elements in sub-clause 4.6:

CIRCULAR ARC CENTRE REVERSED
CONNECTING EDGE

Page 16

Add the following before sub-clause 4.6.1:

In addition to the graphical primitive elements listed above, this part of ISO/IEC 8632 defines elements permitting the definition of 'compound primitives' from several of the other graphical primitives. The following classes of compound primitives are defined: 'compound text' and 'closed figures'. The elements that may be used to specify compound primitives are listed in table 1a.

Table 1a - Contributing primitives to compound primitives

Compound Primitive	First Element	Primitives Included	Other Elements	Final Element
Compound Text	TEXT RESTRICTED TEXT (Note 1)	APPEND TEXT (Note2) GDP (Note 5)		APPEND TEXT (Note 3) GDP (Note 5)
Closed Figure	BEGIN FIGURE	Line Primitives Fill Primitives (Note 4) GDP (Note 5)	NEW REGION	END FIGURE

NOTES

1 The final/not final flag is 'not final'; the primitive defines the reference point of the entire compound text primitive; the text of the primitive is accumulated.

2 The final/not final flag is 'not final'.

3 The final/not final flag is 'final'; the text of the primitive is accumulated before the compound primitive is closed.

4 All primitives of the identified classes may be included.

5 Whether a GDP may contribute to compound text or closed figures, and whether or how it specifies that the compound text state or closed figure state be opened, maintained or closed, is specified with the definition of the GDP in the International Register of Graphical Items.

Graphical primitive elements and compound primitive elements may be subject to transformation in VDC space (segment and copy transformation, see 4.12.4.2 and 4.12.5). Such a transformation may change the shape of some primitives. If there is a skew, a primitive initially specified as a rectangle may become a parallelogram. If there is an anisotropic scaling, a primitive initially specified as a circle may become an ellipse. Note that the shape of markers is not affected by such transformations. Anisotropic transformation will change the angle at which non-parallel lines intersect; isotropic transformation will preserve the angle at which non-parallel lines intersect.

Page 16

Sub-clause 4.6.1.1. Add the following text to the paragraph describing CIRCULAR ARC xxx:

A reverse direction arc can also be specified; see 5.6.20.

Page 16

Add the following at the end of 4.6.1.1:

CONNECTING EDGE A line segment connecting the last point of the preceding line element to the next point is generated during the construction of a closed figure. The next point is either the first point of the next line element or the current closure point.

Page 16

Add the following at the end of 4.6.1.3:

In version 2 metafiles, line clipping is controlled by the LINE CLIPPING MODE element, which can have one of the following values: 'locus', 'shape', or 'locus then shape'. However, clipping applies only if the CLIP INDICATOR is 'on'.

For 'locus' clipping, the mathematical locus of the line is clipped at the intersection with the clip rectangle before shape rendering is applied. Hence, part of the shape of a clipped line may appear outside the clip rectangle.

For 'shape' clipping, the shape of the rendered line is clipped to the intersection with the clip rectangle; that is, nothing is drawn outside the clip rectangle. A portion of a widened line may appear inside the clip rectangle even though the mathematical locus of the line itself may be entirely outside the clip rectangle.

For 'locus then shape' clipping, the mathematical locus of the line is clipped, as with locus clipping, and then subsequently the rendered shape of the clipped locus is again clipped. Note that, since the mathematical locus of the line may have changed as a result of locus clipping, subsequent shape rendering and clipping may produce a different appearance of a line from either of the other two clipping modes.

If the line width is measured in VDC units it is subject to the VDC-to-Device mapping (4.4.7) as well as to both segment and copy transformation (4.12.4.2 and 4.12.5). Note that the entire locus of an arc is subject to these transformations. In the case of an anisotropic mapping or transformation the rendered width of the line will change with the direction of the line segment. If the line width is specified as a scale factor it is not affected by any transformations.

Page 17

Add the following before the first paragraph of 4.6.2.3:

The following discussion applies to version 1 metafiles.

Page 17

Sub-clause 4.6.2.3: at the end of the first paragraph change "is not standardized." to the following:

is not standardized for version 1 metafiles.

Page 17

Add the following at the end of 4.6.2.3:

In version 2 metafiles, marker clipping is controlled by the MARKER CLIPPING MODE element, which can have one of the following values: 'locus', 'shape' or 'locus then shape'. However, clipping applies only if the CLIP INDICATOR is 'on'.

For 'locus' clipping, the specifying points of each marker are clipped at the intersection with the clip rectangle before shape rendering is applied. The marker is only visible if its specifying point is within the clip rectangle. Hence, part of the shape of a marker may appear outside the clip rectangle providing its specifying point is within the clip rectangle.

For 'shape' clipping, the shape of the rendered marker symbols are clipped to the intersection with the clip rectangle; that is, nothing is drawn outside the clip rectangle. Portions of the marker symbol may appear inside the clip rectangle even if the marker's position is outside.

For 'locus then shape' clipping, the clipping is first applied to the specifying points of each marker, as with 'locus' clipping, and then subsequently the rendered shape of the markers are again clipped.

If the marker size is measured in VDC units, it is subject to the VDC-to-Device mapping (4.4.7) as well as to both segment and copy transformation (4.12.4.2 and 4.12.5). The shape of markers is never affected by transformations; for example, a circle used as a marker type shall always appear as a circle. Only the marker size may be transformed. For this purpose, conceptually, vectors with length equal to the marker size and arbitrary orientations are transformed; the resulting marker size is determined by the orientation of the vector which maximizes the length under the transformation.

If the marker size is specified as a scale factor it is not affected by any transformations.

Page 18

Add the following at the end of 4.6.3.3:

Clipping of text strings is described in 4.7.6.

The vectors specified by the CHARACTER ORIENTATION element (4.7.6) are subject to the VDC-to-Device mapping (4.4.7) as well as to both segment and copy transformation (4.12.4.2 and 4.12.5).

Page 19

Add the following at the end of 4.6.4.5:

Edge clipping is controlled by the EDGE CLIPPING MODE element, which has the same enumerations as LINE CLIPPING MODE. Edges are clipped in the same way that lines are clipped; see 4.6.1.3.

Page 19

Add the following after 4.6.4.5:

4.6.4.6 Transformation

The entire mathematical locus of rectangles, circular and elliptical filled-area elements is subject to the VDC-to-Device mapping (4.4.7), segment transformations (2.12.4.2) and copy transformations (4.12.5). Because anisotropic transformation does not preserve angles between non-parallel lines, rectangles may become parallelograms and circles may become ellipses.

The vectors of the PATTERN SIZE element are subject to all transformations.

The edge widths are treated in exactly the same way as line widths (4.6.1.3).

Under certain conditions the clip rectangle is subject to the copy transformation (4.12.5).

Page 20

Add the following after 4.6.7

4.6.8 Closed figures

4.6.8.1 Construction of closed figures

A closed figure is a fill type compound object which commences with a BEGIN FIGURE element, followed by an ordered sequence of line and fill primitives (and optionally attributes and NEW REGION elements), and followed by END FIGURE. Edge attribute values are associated with the edge portions of the closed figure and fill attribute values are associated with the complete graphic object. BEGIN FIGURE and END FIGURE elements are delimiter elements; NEW REGION is a control element. The entire fill object is considered as a single unit on interpretation.

4.6.8.1.1 Closure point

The first point of the first line primitive in a new region is the closure point for that region. On interpretation this closure point is retained for use in closing the region. When the region is closed (with a NEW REGION or END FIGURE element, or by a fill primitive which begins a new region) an implicit boundary portion from the last point of the last line primitive in the region to this closure point is added to the closed figure on interpretation, unless these points are already coincident.

4.6.8.1.2 Regions

A closed figure consists of one or more regions. A region has a closed boundary which may be concave, convex, or self intersecting. A region is formed either by invoking a fill primitive inbetween BEGIN FIGURE and END FIGURE elements (FIGURE OPEN state; see 4.10) which closes the last region and contributes one or more complete regions, by invoking NEW REGION to start new regions to be formed from line primitives, or by a final invocation of END FIGURE. A closed figure constructed from only line primitives without use of NEW REGION consists of a single region.

The NEW REGION element may occur at any time during the closed figure construction. If the current region is closed, the element is ignored on interpretation. If the current region is open, an implicit boundary portion is added from the last point of the last primitive to the current closure point unless CONNECTING EDGE has been invoked after the last line primitive, in which case, an explicit boundary portion and edge portion is added by the CONNECTING EDGE line primitive.

4.6.8.2 Boundaries and edges

The boundary of each region consists of a combination of implicit boundary portions and edge portions.

4.6.8.2.1 Explicit boundary portions

Explicit boundary portions and edge portions are those added by the inclusion of primitives during closed figure construction. These are generated in the following situations:

- For fill primitives other than POLYGON SET, the complete edge becomes an explicit boundary portion and edge portion in the closed figure.
- For line primitives, those portions which would be rendered outside closed figure construction become explicit boundary portions and edge portions. In particular for DISJOINT POLYLINE, only the segments from the first point to the second point, from the third point to the fourth point, and so on, become explicit boundary portions and edge portions when incorporated into closed figures.
- A CONNECTING EDGE primitive which precedes an action which would normally have added an implicit boundary portion to the closed figure either to close a region (including closing the closed figure itself) or to connect two line primitives results in the portion added being an explicit boundary portion and edge portion. CONNECTING EDGE preceding or following DISJOINT POLYLINE or POLYGON SET does not affect the interpretation of those elements with respect to boundaries and edges.

Edge portions have associated edge attribute values taken from the current attribute values on interpretation. These values can be changed between the line and fill primitives that result in edge portions in a closed figure, and hence each edge portion has a distinct set of attribute values associated with it.

4.6.8.2.2 Implicit boundary portions

Edge attributes are never associated with implicit boundary portions. Implicit boundary portions are only rendered on interpretation for interior style HOLLOW and are a special representation of the interior, not a representation of any portion of the edge.

Implicit boundary portions are added on interpretation to the closed figure definition under the following circumstances:

- When NEW REGION, END FIGURE, or a fill primitive is interpreted and the current region has not been explicitly closed and CONNECTING EDGE has not occurred since the last line primitive, an implicit boundary portion is added from the last point of the last primitive to the current closure point to close the region.
- When the last point of the preceding line primitive is not coincident with the first point of the current line primitive, an implicit boundary portion is created to connect the last point of the preceding line primitive to the first point of the current line primitive.
- When portions of a DISJOINT POLYLINE primitive would not normally be rendered (i.e. from the second point to the third point, from the fourth point to the fifth point, and so on), implicit boundary portions are added between these points. (These are additional to the ones which may be added to connect to a preceding or following line primitive or to effect region closure after the disjoint polyline.)
- The portions of a POLYGON SET primitive as described below.

4.6.8.2.3 Conditions under which no boundary or edge is added

No boundary or edge portion is ever created connecting two regions, regardless of how those regions were created or closed.

4.6.8.3 Contribution of primitive elements to the closed figure

4.6.8.3.1 Contribution of line elements to the closed figure

For line primitives, the 'first point' of a line primitive is connected to the 'last point' of the preceding line primitive, and the connecting implicit boundary portion becomes part of the boundary of the closed figure on interpretation. For each of the line primitives the first and last points are defined to be as follows:

POLYLINE p1, p2, ..., pn:
p1 is the first point; pn is the last point.

DISJOINT POLYLINE p1, p2, ..., pn:
p1 is the first point; pn is the last point.

CIRCULAR ARC 3 POINT p1, p2, p3:
p1 is the first point; p3 is the last point.

CIRCULAR ARC CENTRE:
CIRCULAR ARC CENTRE REVERSED:
The first point is the intersection of the circle with the ray (dx start, dy start) from the centre point (i.e. the clockwise end of the arc for CIRCULAR ARC CENTRE, the anti-clockwise end of the arc for CIRCULAR ARC CENTRE REVERSED); the last point is the intersection of the circle with the ray (dx end, dy end) from the centre point (i.e. the anti-clockwise end of the arc for CIRCULAR ARC CENTRE, the clockwise end of the arc for CIRCULAR ARC CENTRE REVERSED).

ELLIPTICAL ARC:
The first point is the intersection of the ellipse with the ray (dx start, dy start) from the centre point; the last point is the intersection of the ellipse with the ray (dx end, dy end) from the centre point.

GENERALIZED DRAWING PRIMITIVE:
For GDPs which generate line primitives, the first point is the first point of the point list; and the last point is the last point of the point list, as defined in the in the GDP registration and associated documentation.

CONNECTING EDGE:
If the region is open, the start point of the connecting edge is the last point of the last line primitive, and the end point of the connecting edge is either the first point of the following primitive or the current closure point as described above. If the connecting edge would be of zero length (i.e. if the two points it connects are coincident), the element is ignored on interpretation. The current modal values of the edge attributes are associated with any edge portion generated by this element.

If the current region is not open, invocations of the CONNECTING EDGE elements encoutered are ignored on interpretation (i.e. CONNECTING EDGE shall not be used to connect regions).

Invoking CONNECTING EDGE multiple times after a line primitive results in the first instance (with its associated attributes) being used on interpretation.

On interpretation the theoretical definitions of the line primitives, not their renditions on the display surface, are used to define the explicit boundary portions of the closed figure. In particular, clipping does not apply to the construction of the closed figure, and the gaps or spaces of the edge type or the rendered width of the edge width do not affect the definition of the boundary of the closed figure.

4.6.8.3.2 Contribution of fill elements to the closed figure

Each fill primitive contributes a complete region to the figure (POLYGON SET may contribute more than one), after first closing the current region if one is open. On interpretation, an implicit NEW REGION is performed before and after a fill primitive (i.e. the new region resulting from a fill primitive is closed, and the next primitive begins a new region.)

The unclipped boundary of each fill primitive contributes to the unclipped boundary of the closed figure.

POLYGON SET primitives contribute to closed figure construction as follows:

- A POLYGON SET is considered to contribute one or more complete regions. If the current region has not been closed, an implicit NEW REGION is performed before the POLYGON SET is added to the figure definition. If the POLYGON SET does not end with a point whose edge-out flag is 'close visible' or 'close invisible', an implicit NEW REGION is performed after the POLYGON SET.

- Sequences of points with edge-out flag 'visible' are treated as if they were polylines, terminating with the first point with a different edge-out flag. Each such polyline becomes an edge portion of the boundary of the figure. The edge attribute values (including EDGE VISIBILITY) in effect when POLYGON SET occurs are associated on interpretation with any edge portion added in this way.

- Sequences of points with edge-out flag 'invisible' contribute implicit boundary portions which are polylines joining the points in the sequence, but not edges. Edge attribute values are not associated with these.

- Points with edge-out flag 'close invisible' generate the equivalent of a NEW REGION, generating an implicit boundary portion from this point to the current closure point if these are not coincident, and closing the current region.

- Points with edge-out flag 'close visible' generate the equivalent of a CONNECTING EDGE followed by a NEW REGION, resulting in an edge portion from this point to the current closure point if these are not coincident. The edge attribute values (including EDGE VISIBILITY) in effect when POLYGON SET is invoked are associated with any edge portion added in this way.

4.6.8.3.3 Contribution of GDPs to the closed figure

A GDP which is defined as a line primitive shall specify which is the first point and the last point in its point list, with respect to closed figure construction. Such GDPs are assumed to contribute to a closed figure a boundary corresponding to the unclipped locus which would be rendered on interpretation if the element occurred outside closed figure construction. Any other behaviour shall be as documented explicitly in the GDP description. A GDP which is defined as being a fill primitive is treated as described in the previous section. Any variation or special handling for closed figure construction shall be documented explicitly in the GDP description.

4.6.8.5 Examples of closed figures

Examples of closed figures are shown in Figure 2a, 2b, 2c, 2d, 2e and 2f.

The POLYGON SET example shown in Figure 13 may also be obtained using the closed figure:

```
EDGE VISIBILITY (ON)
BEGIN FIGURE
        POLYLINE (P3, P1, P2)
        NEW REGION {Note 1}
        POLYLINE (P4, P5, P6, P4)
END FIGURE
```

NOTE

1 Invisible implicit boundary portion P3..P2 generated.

Figure 2a shows the closed figure resulting from interpretation of the elements listed below.

```
EDGE VISIBILITY (ON)
BEGIN FIGURE
        POLYLINE (P1, P2)
        CIRCULAR ARC 3 POINT (P2, P3, P4)
        POLYLINE (P4, P5)
        CIRCULAR ARC 3 POINT (P5, P6, P1)
END FIGURE
```

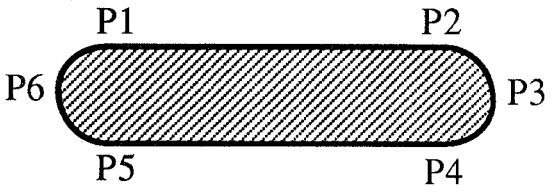


Figure 2a

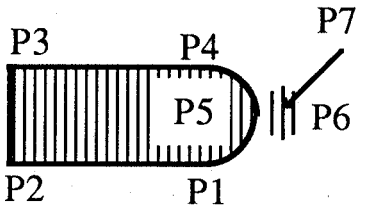


Figure 2b

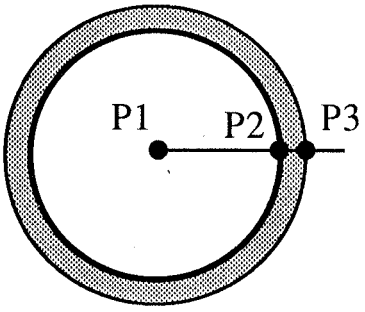


Figure 2c

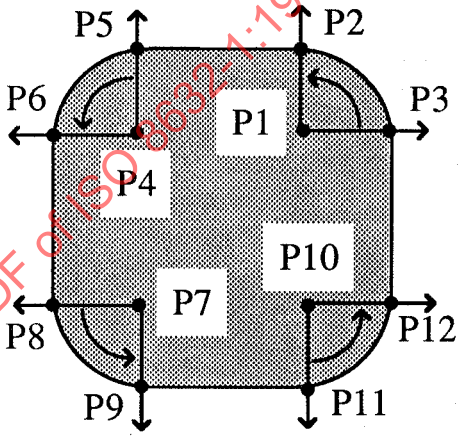


Figure 2d

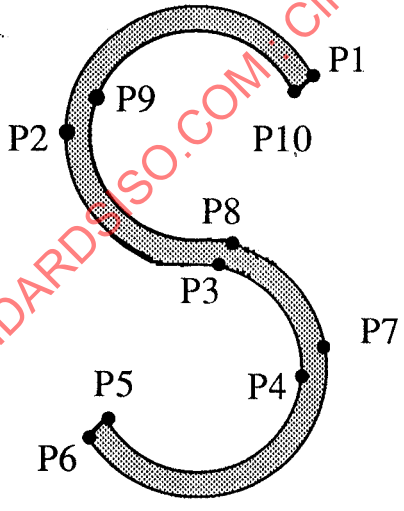


Figure 2e

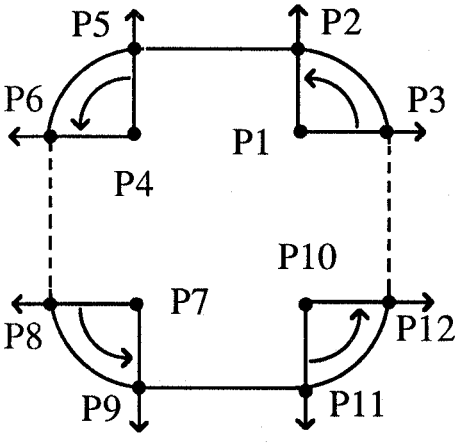


Figure 2f

Figures 2a, 2b, 2c, 2d, 2e, 2f - Examples of closed figures

Figure 2a could also be the result of interpreting the following sequence of elements which include CONNECTING EDGE.

```
EDGE VISIBILITY (ON)
BEGIN FIGURE
        CIRCULAR ARC 3 POINT (P2, P3, P4)
        CONNECTING EDGE
        CIRCULAR ARC 3 POINT (P5, P6, P1) {Note 1}
        CONNECTING EDGE
END FIGURE {Note 2}
```

NOTES

1 Visible edge portion P4..P5 generated.
2 Visible edge portion P1..P2 generated.

Figure 2b shows the closed figure resulting from interpretation of the elements listed below.

```
EDGE VISIBILITY (ON)
BEGIN FIGURE
        POLYLINE (P1, P2, P3, P4)
        CIRCULAR ARC 3 POINT (P4, P5, P1)
        EDGE VISIBILITY (OFF)
        NEW REGION
        P7 = P5 + (P6-P5)/2
        CIRCULAR ARC CENTRE (P7, 1, 0, 1, 0, |P7 - P5|)
END FIGURE
```

Figure 2c shows the closed figure resulting from interpretation of the elements listed below.

```
BEGIN FIGURE
        CIRCULAR ARC CENTRE (P1, 1, 0, 1, 0, |P3 - P1|)
        NEW REGION
        CIRCULAR CENTRE (P1, 1, 0, 1, 0, |P2 - P1|)
END FIGURE
```

Figure 2c could also be the result of interpreting the following sequence of elements which include fill area elements.

```
BEGIN FIGURE
        CIRCLE (P1, |P3 - P1|)
        CIRCLE (P1, |P2 - P1|)
END FIGURE
```

Figure 2d shows the use of ELLIPTICAL ARC to draw a box with rounded corners and is the result of interpreting the sequence of elements shown below.

```
EDGE VISIBILITY (ON)
BEGIN FIGURE
        ELLIPTICAL ARC (P1, P2, P3, (1,0), (0,1))
        CONNECTING EDGE
        ELLIPTICAL ARC (P4, P5, P6, (0,1), (-1,0)) {Note 1}
        CONNECTING EDGE
        ELLIPTICAL ARC (P7, P8, P9, (-1,0), (0,-1))
        CONNECTING EDGE
        ELLIPTICAL ARC (P10, P11, P12, (0,-1), (1,0))
        CONNECTING EDGE
END FIGURE {Note 2}
```

NOTES

1 Visible edge portion P2..P5 generated; edge portions P6..P8 and P9..P11 are drawn with the next two arcs.
2 Visible edge portion P12..P3 generated.

Figure 2e shows the use of CIRCULAR ARC 3 POINT to create an 'S' shape and is the result of interpreting the sequence of elements shown below.

```
EDGE VISIBILITY (ON)
BEGIN FIGURE
        CIRCULAR ARC 3 POINT (P1, P2, P3)
        CIRCULAR ARC 3 POINT (P3, P4, P5)
        CONNECTING EDGE
        CIRCULAR ARC 3 POINT (P6, P7, P8) {Note 1}
        CIRCULAR ARC 3 POINT (P8, P9, P10)
        CONNECTING EDGE
END FIGURE {Note 2}
```

NOTES

1 Visible edge portion P5..P6 generated.
2 Visible edge portion P10.P1 generated.

Figure 2f shows the closed figure resulting from interpretation of the elements listed below. It is similar to figure 2d, but makes use of changing the edge attributes between successive occurrences of CONNECTING EDGE.

```
EDGE VISIBILITY (ON)
BEGIN FIGURE
        ELLIPTICAL ARC(P1, P2, P3, (1,0), (0,1))
        ELLIPTICAL ARC(P4, P5, P6, (0,1), (-1,0)) {Note 1}
        EDGE TYPE (DASHED)
        CONNECTING EDGE
        ELLIPTICAL ARC(P7, P8, P9, (-1,0), (0,-1)) {Note 2}
        EDGE TYPE (SOLID)
        CONNECTING EDGE
        ELLIPTICAL ARC(P10, P11, P12, (0,-1), (1,0))
        EDGE TYPE(DASHED)
        CONNECTING EDGE
END FIGURE {Note 3}
```

NOTES

1 No edge portion P2..P5 generated.
2 Visible (dashed) edge portion P6..P8 generated; solid edge portion P9..P11 drawn with the next arc.
3 Visible (dashed) edge portion P12..P3 generated.

Page 39

Add the following after 4.7.8:

4.7.9 Pick identifier

The pick identifier is associated with graphical primitive elements within segments (see 4.12). It is the only attribute element which does not affect the appearance of a graphical primitive element. It merely establishes a means of identification of primitives within segments at metafile interpretation. The PICK IDENTIFIER element has no graphical effect.

Page 40

Add the following after 4.11:

4.12 Segment elements

4.12.1 Introduction

In the CGM graphic objects may be grouped in segments, each segment being identified by a unique segment identifier. Segments may have the attributes:

a) transformation;

b) highlighting;

c) display and pick priority.

These may be defined at segment definition time, before the first primitives of the segment, and shall not be changed thereafter.

Only elements inside segments are affected by the segment attributes.

The segment elements are:

COPY SEGMENT
INHERITANCE FILTER
CLIP INHERITANCE
SEGMENT TRANSFORMATION
SEGMENT HIGHLIGHTING
SEGMENT DISPLAY PRIORITY
SEGMENT PICK PRIORITY

Segments are delimited by BEGIN SEGMENT and END SEGMENT.

4.12.2 Local and global segments

There are two types of segments: local segments and global segments. Both contain primitives and attributes that can be manipulated in the manner described above. Local segments have no existence beyond the bounds of the picture body in which they are defined. Defining a local segment in a picture automatically includes that segment in the picture's image. In contrast, global segments can be referenced by any of the pictures in the metafile in which they are defined.

4.12.2.1 Location of, and access to, global segments.

A global segment is delimited by the BEGIN SEGMENT and END SEGMENT elements. Global segments are defined in the Metafile Descriptor. They are not a part of any picture within the metafile. They shall be accessed from within individual pictures by the COPY SEGMENT (4.12.5) element. The COPY SEGMENT element incorporates the segment into the open picture in the same way for both local and global segments.

4.12.2.2 Permitted segment-related elements in the Metafile Descriptor

BEGIN SEGMENT is the only segment-related element that is allowed within the Metafile Descriptor State (MDS) (see 4.10). BEGIN SEGMENT changes the state to Global Segment State (GSS).

4.12.2.3 References to global segments

Within pictures, no elements are allowed that would modify the contents or default appearance of global segments. This restriction preserves the logical independence of pictures and the ability to randomly access pictures. The only references to global segments within pictures shall be by using the COPY SEGMENT element.

4.12.2.4 Association of control and attribute elements with primitives inside segments

The current modal values of control and attribute elements are associated with the primitives inside local segments. The modal values established by setting control or attribute elements within a segment remain in effect outside the segment until they are explicitly changed.

Control and attribute elements are bound in global segments as they are in local segments. Upon the occurrence of BEGIN METAFILE, every element that is modally defined and bound to primitives (Metafile Descriptor elements defining modes and precisions, Picture Descriptor elements, Control elements, Attribute elements and Segment Control elements) has a default value. Conceptually the set of all of these define a "modal state list".

The Metafile Descriptor (MD) is processed sequentially. Throughout the Metafile Descriptor, modal MD elements modify the MD entries in the state list and occurrences (possibly multiple) of the METAFILE DEFAULTS REPLACEMENT element allow manipulation (outside of GSS state) of the rest of the modal elements (as well as explicitly changing the defaults). Within GSS state the allowable modal (control, attribute, and segment attribute) elements also alter the contents of the modal state list. The values of modal elements that are in effect upon BEGIN PICTURE are the default values for that picture, whether they are implicit (defined in ISO/IEC 8632) or explicit (that is, by values set in the Metafile Defaults Replacement).

4.12.3 Delimiting and naming segments

The contents of a segment are delimited by the elements BEGIN SEGMENT and END SEGMENT which are delimiter elements. The elements in between these two delimiters are a part of that segment. Each segment has an identifier associated with it. No two global segments shall have the same identifier and no local segment shall have an identifier which is the same as either a local segment in the same picture or the same as a global segment.

4.12.4 Segment attributes

4.12.4.1 Introduction

The segment attributes associated with each segment control its display. Segment attributes shall be set only after the segment has been opened with the BEGIN SEGMENT element. When a segment is opened the segment's attributes are set to their default values. Segment attributes, if set, shall be set immediately after the BEGIN SEGMENT element and before any other type of element. This structure is shown below:

BEGIN SEGMENT (Segment identifier)
Segment attributes
Allowed primitives, attributes and control elements in any order
END SEGMENT

4.12.4.2 Segment transformation

The segment transformation is a coordinate transformation associated with each segment and applies to all graphical objects in the identified segment and will be used on interpretation. Clipping rectangles are not transformed by the segment transformation. It allows scaling, translation, and rotation of segments to be defined during segment definition.

The segment transformation is a transformation of VDC space to VDC space and is distinct from the VDC-to-Device mapping which is a transformation of VDC space to device coordinate space.

The transformation attribute of a segment may be defined by the SEGMENT TRANSFORMATION element during the segment definition. A segment transformation is represented by a 2 x 3 matrix, comprising a 2 x 2 scaling and rotation portion, and a 2 x 1 translation portion. If the SEGMENT TRANSFORMATION element is not stored in the metafile, then all coordinate data is mapped using only the VDC-to-Device mapping. If the SEGMENT TRANSFORMATION is stored in the metafile, it is applied before the application of the VDC-to-Device mapping.

The use of segment transformations may produce coordinates that cannot be expressed within the VDC range. This is handled in an interpretation dependent way.

4.12.4.3 Segment highlighting

Segment highlighting can take one of two values, NORMAL or HIGHLIGHTED. The setting of this attribute selects one of these two states for the segment.

4.12.4.4 Segment display priority

The display priority attribute of a segment determines how overlapping segments are displayed. During interpretation segments with higher display priorities will be displayed as if they were in front of segments with lower display priorities. The segment display priority may be normalized to the continuous range of real numbers, zero to one, by applying the minimum extent and maximum extent values provided by the Metafile Descriptor element SEGMENT PRIORITY EXTENT.

4.12.4.5 Segment pick priority

The pick priority attribute of a segment is used to resolve the picking of segments which overlap. The segment pick priority may be normalized to the continuous range of real numbers, zero to one, by applying the minimum extent and maximum extent values provided by the Metafile Descriptor element SEGMENT PRIORITY EXTENT. Interpretation of SEGMENT PICK PRIORITY has no graphical effect.

4.12.5 Copy segment and inheritance

The COPY SEGMENT element inserts the elements of the referenced segment into the picture at the point of occurrence of the element.

The elements copied may be altered in a variety of ways:

a) The inheritance filter mechanism controls whether individual attribute values are reapplied to the elements.

b) The clip inheritance mechanism controls whether the primitives in the segment are clipped to the current clip rectangle or to a combination of the current and the segment clipping rectangles.

c) The primitive elements are transformed by the copy transformation and optionally by the segment transformation of the copied segment according to the rules for transformation.

COPY SEGMENT has a transformation matrix as a parameter. The copy transformation is applied to graphical objects before they are copied. This also applies to clipping rectangles in the segment (see below). Graphical objects may be transformed to alter their location, size, and orientation.

A segment may be referenced by the COPY SEGMENT element, either within a picture or in a global segment. The attributes associated on interpretation can be those bound to the segment being copied or can be imposed by the inclusion of the INHERITANCE FILTER element.

The clipping associated with a segment can be that associated with the picture at the time of the copy or can be a combination of the current clipping and the segment clipping when the CLIP INHERITANCE element is used.

The inheritance filter mechanism allows the use of the current values of attributes and controls to be associated with the copied segment in place of the attributes and controls bound to the primitives when the segment was created. The attributes and controls to be associated with the segment can be all attributes or can be a subset of attributes. The attributes and controls are selected using the INHERITANCE FILTER element. The attributes and controls can be selected using individual or group names for attributes, controls and ASFs. The elements that can be selected are shown in table 3a for attributes and controls (both individual element names and group names) and in table 3b for ASFs.

If an attribute or group of attributes designated in the filter selection list is set to 'state list', graphic objects inherit that attribute or group of attributes from the current modal values when a segment is copied.

If an attribute or group of attributes designated in the filter selection list is set to 'segment', that attribute or group of attributes is unaffected (in all graphic objects employing them) by the corresponding current state list when a segment is copied.

The default inheritance filter setting value is 'segment' for all attributes and controls.

Table 3a - Inheritance filter selection names for attributes

Attribute Group Name	Individual Attribute Name
LINE ATTRIBUTES	LINE BUNDLE INDEX
	LINE TYPE
	LINE WIDTH
	LINE COLOUR
	LINE CLIPPING MODE
MARKER ATTRIBUTES	MARKER BUNDLE INDEX
	MARKER TYPE
	MARKER SIZE
	MARKER COLOUR
	MARKER CLIPPING MODE
TEXT PRESENTATION AND PLACEMENT ATTRIBUTES	TEXT BUNDLE INDEX
	TEXT FONT INDEX
	TEXT PRECISION
	CHARACTER EXPANSION FACTOR
	CHARACTER SPACING
	TEXT COLOUR
TEXT PLACEMENT AND ORIENTATION ATTRIBUTES	CHARACTER HEIGHT
	CHARACTER ORIENTATION
	TEXT PATH
	TEXT ALIGNMENT
FILL ATTRIBUTES	FILL BUNDLE INDEX
	INTERIOR STYLE
	FILL COLOUR
	HATCH INDEX
	PATTERN INDEX
EDGE ATTRIBUTES	EDGE BUNDLE INDEX
	EDGE TYPE
	EDGE WIDTH
	EDGE COLOUR
	EDGE VISIBILITY
	EDGE CLIPPING MODE
PATTERN ATTRIBUTES	FILL REFERENCE POINT
	PATTERN SIZE
OUTPUT CONTROL	AUXILIARY COLOUR
	TRANSPARENCY
PICK IDENTIFIER	PICK IDENTIFIER
ALL ATTRIBUTES AND CONTROL	All attributes and control elements
ALL	All attributes, control elements and ASFs

Table 3b - Inheritance filter selection names for Aspect Source Flags

ASF Group Name	Individual ASF Name
LINE ASFS	LINE TYPE ASF
	LINE WIDTH ASF
	LINE COLOUR ASF
MARKER ASFS	MARKER TYPE ASF
	MARKER SIZE ASF
	MARKER COLOUR ASF
TEXT ASFS	TEXT FONT INDEX ASF
	TEXT PRECISION ASF
	CHARACTER EXPANSION FACTOR ASF
	CHARACTER SPACING ASF
	TEXT COLOUR ASF
FILL ASFS	INTERIOR STYLE ASF
	FILL COLOUR ASF
	HATCH INDEX ASF
	PATTERN INDEX ASF
EDGE ASFS	EDGE TYPE ASF
	EDGE WIDTH ASF
	EDGE COLOUR ASF
ALL ASFS	All aspect source flags

An example of the COPY SEGMENT element with the INHERITANCE FILTER element is as follows:

```
BEGIN METAFILE "..."
  .
  .
BEGIN SEGMENT (1)
        LINE COLOUR (blue)
        POLYLINE                                              blue solid line
END SEGMENT

BEGIN DEFAULTS REPLACEMENT
        LINE TYPE (dash)
END DEFAULTS REPLACEMENT

BEGIN SEGMENT (2)
        LINE COLOUR (red)
        INHHERITANCE FILTER (LINE ATTRIBUTES,STATE LIST)
        COPY SEGMENT (1)                                      red dashed line
        POLYLINE                                              red dashed line
        INHERITANCE FILTER (LINE ATTRIBUTES,SEGMENT)
        COPY SEGMENT (1)                                      blue solid line
        POLYLINE                                              red dashed line
END SEGMENT

BEGIN PICTURE "..."
BEGIN PICTURE BODY
LINE COLOUR (green)
INHERITANCE FILTER (LINE ATTRIBUTES,SEGMENT)
COPY SEGMENT (2)                                              red dashed line
                                                              red dashed line
                                                              blue solid line
                                                              red dashed line
POLYLINE                                                      green dashed line
INHERITANCE FILTER (LINE ATTRIBUTES,STATE LIST)
COPY SEGMENT (2)                                              green dashed line
                                                              green dashed line
                                                              green dashed line
                                                              green dashed line

BEGIN SEGMENT (3)
        LINE COLOUR (red)
        COPY SEGMENT (1)                                      red dashed line
        INHERITANCE FILTER (LINE ATTRIBUTES,SEGMENT)
        COPY SEGMENT (1)                                      blue solid line
END SEGMENT
  .
LINE COLOUR (green)
COPY SEGMENT (3)                                              red dashed line
                                                              blue solid line
INHERITANCE FILTER (LINE ATTRIBUTES,STATE LIST)
COPY SEGMENT (3)                                              green dashed line
                                                              green dashed line
  .
  .
  .
END PICTURE
END METAFILE
```

Clipping is not included in the INHERITANCE FILTER. There is a separate element that controls clipping behaviour - CLIP INHERITANCE. Its values may be either 'state list' or 'intersection'.

If the value is 'state list', then the clip rectangle associated with primitives in the copied segment is that of the last CLIP RECTANGLE encountered during interpretation in the metafile element sequence prior to the COPY SEGMENT element, that is, the value in the "modal state list".

If the value is 'intersection' and if both the modal state list clip indicator and the clip indicator associated with the primitives of the copied segment are 'on', then the resulting clipping boundary is the intersection of the modal state list clip rectangle with the clipping boundary resulting from the application of the copy transformation to the clip rectangle associated with the primitives. If either indicator is 'off', then there is no contribution from its associated clip rectangle. To illustrate: if TA and TB are copy transformations:

```
BEGIN SEGMENT A
CLIP INDICATOR(ON)
CLIP RECTANGLE R1
POLYLINE P1
END SEGMENT

CLIP INHERITANCE (INTERSECTION)
CLIP INDICATOR(ON)
CLIP RECTANGLE R2
POLYLINE P2
COPY SEGMENT (A,TA)
POLYLINE P3
```

P2 and P3 are clipped by R2, P1 is clipped by R2 (intersected with) TA(R1). This clipping region may turn out to be an 8-sided convex polygon, if TA causes rotation and skewing.

The composition of clipping rectangles continues however many levels the segment hierarchy is nested. For example:

```
BEGIN SEGMENT A
CLIP RECTANGLE R0
POLYLINE P0
CLIP RECTANGLE  R1
POLYLINE P1
END SEGMENT

BEGIN SEGMENT B
CLIP RECTANGLE R2
POLYLINE P2
CLIP INHERITANCE (INTERSECTION)
COPY SEGMENT (A,TA)
END SEGMENT

CLIP RECTANGLE R3
CLIP INHERITANCE (INTERSECTION)
COPY SEGMENT (B,TB)
POLYLINE P3
```

The effective clipping "rectangles" are:

for P0: TB(R2 intersection TA(R0)) intersection R3
for P1: TB(R2 intersection TA(R1)) intersection R3
for P2: TB(R2) intersection R3
for P3: R3

From this example it can be seen that the effective clipping "rectangle" can in fact be an arbitrary convex polygon. Annex D contains recommended fallback procedures for interpreters which cannot perform such clipping.

Segment Transformations are never applied to clipping boundaries. The default value for CLIP INHERITANCE is 'state list'.

Page 39

Sub-clause 4.10: Change the text in the third paragraph, sixth line from "figure 12" to:

figures 12 and 12a

Page 40

Add the following at the end of 4.10:

The states in which each element is allowed for version 2 metafiles are also described in table 3c.

Page 41

Change the title of Figure 12 to be:

Figure 12 - State diagram for version 1 metafiles

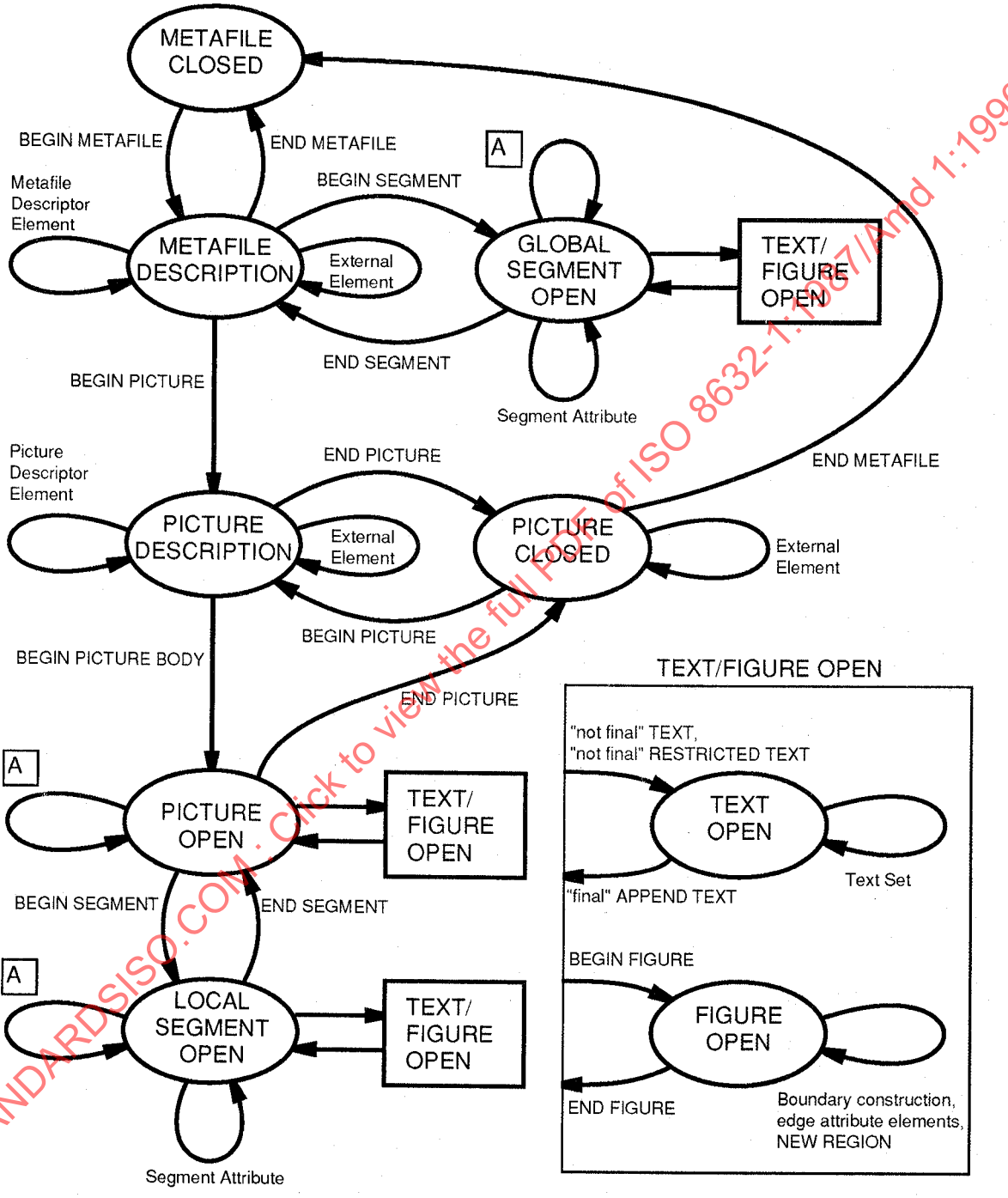
Page 41

Add the following text after the state diagram

NOTE - Many elements allowed in state PO can also occur in the METAFILE DEFAULTS REPLACEMENT.

Page 41

Add figure 12a



A All Control, Graphical Primitive, Attribute, Escape and External Elements except NEW REGION, CONNECTING EDGE, APPEND TEXT, "not final" TEXT, "not final" RESTRICTED TEXT plus COPY SEGMENT, INHERITANCE FILTER and CLIP CONTROL

Text Set: "not final" APPEND TEXT, TEXT BUNDLE INDEX, TEXT FONT INDEX, TEXT PRECISION, CHARACTER EXPANSION FACTOR, CHARACTER SPACING, TEXT COLOUR, CHARACTER HEIGHT, CHARACTER SET INDEX, ALTERNATE CHARACTER SET INDEX, AUXILLIARY COLOUR, TRANSPARENCY

Figure 12a - State diagram for version 2 metafiles

Page 41

Add the following table after Figure 12:

Table 3c - CGM Elements by their allowed states

CGM Element	CGM States								
	PCS	MDS	DR (4)	GSS	PDS	POS	TOS	LSS	FOS
BEGIN METAFILE (1)									
BEGIN PICTURE	X	X							
BEGIN PICTURE BODY					X				
END PICTURE						X			
BEGIN SEGMENT		X				X			
END SEGMENT				X				X	
BEGIN FIGURE				X		X		X	
END FIGURE									X
END METAFILE	X								
METAFILE VERSION		X							
METAFILE DESCRIPTION		X							
VDC TYPE		X							
INTEGER PRECISION		X							
REAL PRECISION		X							
INDEX PRECISION		X							
COLOUR PRECISION		X							
COLOUR INDEX PRECISION		X							
NAME PRECISION		X							
MAXIMUM COLOUR INDEX		X							
COLOUR VALUE EXTENT		X							
METAFILE ELEMENT LIST		X							
METAFILE DEFAULTS REPLACEMENT		X							
FONT LIST		X							
CHARACTER SET LIST		X							
CHAR CODING ANNOUNCER		X							
METAFILE CATEGORY		X							
MAXIMUM VDC EXTENT		X							
SEGMENT PRIORITY EXTENT		X							
SCALING MODE			X		X				
COLOUR SELECTION MODE			X	X	X	X		X	
LINE WIDTH SPECIFICATION MODE			X	X	X	X		X	
MARKER SIZE SPECIFICATION MODE			X	X	X	X		X	
EDGE WIDTH SPECIFICATION MODE			X	X	X	X		X	
VDC EXTENT			X		X				
BACKGROUND COLOUR			X		X				
DEVICE VIEWPORT			X		X				
DEVICE VIEWPORT MAPPING			X		X				
DEVICE VIEWPORT SPECIFICATION MODE			X		X				

Table 3c (continued)

CGM Element	CGM States								
	PCS	MDS	DR	GSS	PDS	POS	TOS	LSS	FOS
LINE REPRESENTATION			X		X				
MARKER REPRESENTATION			X		X				
TEXT REPRESENTATION			X		X				
FILL REPRESENTATION			X		X				
EDGE REPRESENTATION			X		X				
VDC INTEGER PRECISION			X	X		X		X	X
VDC REAL PRECISION			X	X		X		X	X
AUXILIARY COLOUR			X	X		X	X	X	X
TRANSPARENCY			X	X		X	X	X	X
CLIP RECTANGLE			X	X		X		X	
CLIP INDICATOR			X	X		X		X	
LINE CLIPPING MODE			X	X		X		X	
MARKER CLIPPING MODE			X	X		X		X	
EDGE CLIPPING MODE			X	X		X		X	
NEW REGION									X
SAVE PRIMITIVE CONTEXT				X		X		X	
RESTORE PRIMITIVE CONTEXT				X		X		X	
POLYLINE				X		X		X	X
DISJOINT POLYLINE				X		X		X	X
POLYMARKER				X		X		X	
TEXT				X		X		X	
RESTRICTED TEXT				X		X		X	
APPEND TEXT							X		
POLYGON				X		X		X	X
POLYGON SET				X		X		X	X
CELL ARRAY				X		X		X	
GDP				X		X		X	X
RECTANGLE				X		X		X	X
CIRCLE				X		X		X	X
CIRCULAR ARC 3 POINT				X		X		X	X
CIRCULAR ARC 3 POINT CLOSE				X		X		X	X
CIRCULAR ARC CENTRE				X		X		X	X
CIRCULAR ARC CENTRE CLOSE				X		X		X	X
ELLIPSE				X		X		X	X
ELLIPTICAL ARC				X		X		X	X
ELLIPTICAL ARC CLOSE				X		X		X	X
CIRCULAR ARC CENTRE REVERSED				X		X		X	X
CONNECTING EDGE									X
LINE BUNDLE INDEX			X	X		X		X	
LINE TYPE			X	X		X		X	
LINE WIDTH			X	X		X		X	
LINE COLOUR			X	X		X		X	

Table 3c (concluded)

CGM Element	CGM States								
	PCS	MDS	DR	GSS	PDS	POS	TOS	LSS	FOS
MARKER BUNDLE INDEX			X	X		X		X	
MARKER TYPE			X	X		X		X	
MARKER SIZE			X	X		X		X	
MARKER COLOUR			X	X		X		X	
TEXT BUNDLE INDEX			X	X		X	X	X	
TEXT FONT INDEX			X	X		X	X	X	
TEXT PRECISION			X	X		X	X(3)	X	
CHARACTER EXPANSION FACTOR			X	X		X	X	X	
CHARACTER SPACING			X	X		X	X	X	
TEXT COLOUR			X	X		X	X	X	
CHARACTER HEIGHT			X	X		X	X	X	
CHARACTER ORIENTATION			X	X		X		X	
TEXT PATH			X	X		X		X	
TEXT ALIGNMENT			X	X		X		X	
CHARACTER SET INDEX			X	X		X	X	X	
ALTERNATE CHARACTER SET INDEX			X	X		X	X	X	
FILL BUNDLE INDEX			X	X		X		X	
INTERIOR STYLE			X	X		X		X	
FILL COLOUR			X	X		X		X	
HATCH INDEX			X	X		X		X	
PATTERN INDEX			X	X		X		X	
EDGE BUNDLE INDEX			X	X		X		X	X
EDGE TYPE			X	X		X		X	X
EDGE WIDTH			X	X		X		X	X
EDGE COLOUR			X	X		X		X	X
EDGE VISIBILITY			X	X		X		X	X
FILL REFERENCE POINT			X	X		X		X	
PATTERN TABLE			X		X	X			
PATTERN SIZE			X	X		X		X	
COLOUR TABLE			X		X	X			
ASPECT SOURCE FLAGS			X	X		X		X	X(2)
PICK IDENTIFIER			X	X		X		X	
ESCAPE	X	X	X	X	X	X	X	X	X
MESSAGE	X	X	X	X	X	X		X	X
APPLICATION DATA	X	X	X	X	X	X		X	X
COPY SEGMENT				X		X		X	
INHERITANCE FILTER			X	X		X		X	
CLIP INHERITANCE			X	X		X		X	
SEGMENT TRANSFORMATION				X				X	
SEGMENT HIGHLIGHTING				X				X	
SEGMENT DISPLAY PRIORITY				X				X	
SEGMENT PICK PRIORITY				X				X	

Abbreviations used in table 3c:

PCS	Picture Closed State
MDS	Metafile Description State
DR	Defaults Replacement Mode
GSS	Global Segment State
PDS	Picture Description State
POS	Picture Open State
TOS	Text Open (Partial text) State
LSS	Local Segment State
FOS	Figure Open State

NOTES

1 BEGIN METAFILE is the only element allowed in the state 'Metafile Closed'
2 Edge ASFs are the only ASFs allowed in Figure Open State
3 Use of TEXT PRECISION in text open state is permitted; however, the intended result is not well defined and such usage is likely to lead to unpredictable results.
4 Defaults replacement mode is not actually a metafile state, but it is included in this table for the convenience of the implementor of this standard.

Page 42

Sub-clause 5.1: Add the following after the ninth paragraph which starts with the sentence: "The External Elements....":

The segment elements (see 5.10) provide for the grouping and manipulation of elements.

Page 42

Sub-clause 5.1: Add the following at the end of the table of abbreviations of data type names:

N	Name	Identifier for a segment, pick or context. Realization is integer: range is dependent on NAME PRECISION
VC	Viewport Coordinate	Single real or integer value as determined by the DEVICE VIEWPORT SPECIFICATION MODE: R fraction [0..1] of default viewport I millimetres (scaled) I native device units
VP	Viewport Point	Two VC values representing the x and y coordinates of a point in viewport specification space

Page 46

Add the following after 5.2.5:

5.2.6 BEGIN SEGMENT

Parameters:

segment identifier (N)

Description:

This is the first element of a segment. All subsequent elements until the next END SEGMENT will belong to this segment.

References:

4.2
4.12.3

5.2.7 END SEGMENT

Parameters:

None

Description:
Subsequent elements will no longer belong to a segment.

References:
4.2
4.12.3

5.2.8 BEGIN FIGURE

Parameters:

none

Description:
This is the first element of a closed figure. All subsequent elements until the next END FIGURE will be part of the closed figure.

Reference:
4.6.8

5.2.9 END FIGURE

Parameters:

none

Description:
This element terminates the current closed figure.

If the current region has not yet been closed by a preceding NEW REGION element and if the last point of the last line element is not coincident with the current closure point, then the current subregion is closed by a line segment connecting the last point of the preceding line element to the current closure point. This line becomes a part of the implicit boundary specification. If the END FIGURE was preceded by a CONNECTING EDGE element, which was itself preceded by a line primitive, then this line also becomes part of the edge specification. If the region which has been previously closed is empty, or if the last point of the last line element is coincident with the current closure point, or if the last element was a filled-area primitive, then no line segment is generated by this element.

Reference:
4.6.8

Page 47

Add the following at the end of the "Description" in 5.3.1:

The CGM as defined in ISO 8632 : 1987/Am.1 : 1990 is version two (2).

METAFILE VERSION shall appear only once in the Metafile Descriptor for version 2 metafiles. It is recommended that it shall appear only once in version 1 metafiles.

Page 50

Sub-clause 5.3.11: Add the following shorthand names at the end of the list given in the second paragraph of the "Description":

VERSION 2 SET
EXTENDED PRIMITIVES SET
VERSION 2 GKSM SET

Page 50

Sub-clause 5.3.11: Add the following at the end of the "Description":

METAFILE ELEMENT LIST shall appear only once in the Metafile Descriptor for version 2 metafiles. It is recommended that it shall appear only once in version 1 metafiles.

Page 55

Add the following sub-clauses after 5.3.15:

5.3.16 NAME PRECISION

Parameters:

The form of the parameter depends on the specific encoding.

Description:

The precision for operands of data type name (N) is specified for subsequent data of type N. The precision is defined as the field width measured in units applicable to the specific encoding.

Reference:

4.3

5.3.17 MAXIMUM VDC EXTENT

Parameters:

first corner (P)
second corner (P)

Description:

The two corners define a rectangular extent in VDC space which bounds the values of the VDC EXTENT elements 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.

References:

4.3
4.4.4

5.3.18 SEGMENT PRIORITY EXTENT

Parameters:

minimum priority extent (I)
maximum priority extent (I)

Description:

The parameters represent an extent which bounds the segment display and pick priority values which will be encountered in the metafile. It need not represent the exact priorities in the metafile. The lowest display priority value is zero.

References:
4.3
4.12.4.4
4.12.4.5

Page 56

Add the following note at the end of 5.4.1 (SCALING MODE):

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

Page 58

Add the following sub-clauses after 5.4.7:

5.4.8 DEVICE VIEWPORT

Parameters:

first corner (VP)
second corner (VP)

Description:
The two parameters define the opposite corners of a rectangular viewport on the device's display surface. These parameters are specified by the unit system selected by DEVICE VIEWPORT SPECIFICATION MODE.

The effective viewport is that area of the display surface onto which the VDC extent rectangle is mapped. If the current DEVICE VIEWPORT MAPPING forces isotropic mapping, and the aspect ratio is not equal to that of the device viewport, the effective viewport will be smaller than the specified viewport on one or the other axis (but not both).

If the current DEVICE VIEWPORT MAPPING does not force isotropic mapping, the effective viewport will be the same as the specified viewport. If the Device Viewport exceeds the available display surface, the Device Viewport is still used to determine the VDC-to-Device mapping.

Mirroring or 180^{o} rotation of the image may be achieved by specifying the corners in some way other than that first is below and to the left of the second.

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

Reference:
4.4.7

5.4.9 DEVICE VIEWPORT SPECIFICATION MODE

Parameters:

VC specifier (one of: fraction of display surface,
millimetres with scalefactor,
physical device coordinates)(E)

metric scale factor (R)

Description:
This element determines how subsequent elements using the data type VC (viewport coordinate) or VP (viewport point) will be defined.

These parameters may be specified in one of three modes: fraction of display surface; millimetres with scale factor; or physical device coordinates.

When the VC specifier is 'fraction of display surface ', the value (0.0, 0.0) corresponds to the lower left corner and the value (1.0, 1.0) corresponds to the upper right corner of the default device viewport. (The default device viewport is the largest unrotated rectangular area visible on the display surface). Numbers outside the range [0.0 to 1.0] may be specified (see 5.4.8). When the VC specifier is 'fraction of display surface' the metric scale factor is ignored.

When the VC specifier is 'millimetres with scalefactor', the metric scale factor parameter represents the distance (in millimetres) on the display surface corresponding to one unit in VC space. One unit in VC space represents one millimetre multiplied by the metric scale factor. The value (0,0) corresponds to the lower left corner and the values increase positively to the right and upwards.

When the VC specifier is 'physical device coordinates', the native units and handedness of the physical device are used. The metric scale factor is ignored.

Metric scaling with a scale factor provides a device-independent means of generating output at a known size. In metric mode, a scale factor of 1.0 indicates that the VC are in units of millimetres; a scale factor of 0.0254 would imply a VC of one thousand per inch.

Reference:
4.4.7

5.4.10 DEVICE VIEWPORT MAPPING

Parameters:

isotropy flag (one of: not forced, forced)(E)
horizontal alignment flag (one of: left, centre, right)(E)
vertical alignment flag (one of: bottom, centre, top)(E).

Description:
This element determines how the coordinate mapping is derived from the VDC EXTENT and the specified DEVICE VIEWPORT. The remaining parameters are significant only if isotropy is forced by the first parameter. If so, the effective viewport is generally smaller than the specified viewport, and these parameters determine how it will be positioned within the specified viewport. 'Left' and 'bottom' are interpreted as being towards the "first corner" of the specified DEVICE VIEWPORT, regardless of any mirroring or rotation of the viewport on the physical device.

Reference:
4.4.7

5.4.11 LINE REPRESENTATION

Parameters:

line bundle index (IX)
line type (IX)
line width specifier
    if line width specification mode is 'absolute',
    absolute line width (VDC)

    if line width specification mode is 'scaled',
    line width scale factor (R)

line colour specifier
    if the colour selection mode is 'indexed',
    line colour index (CI)

    if the colour selection mode is 'direct',
    line colour value (CD)

Description:

In the line bundle table, the given line bundle index is associated with the specific parameters.

Line type is specified and behaves as indicated in the LINE TYPE attribute element.

Line width is defined in the current LINE WIDTH SPECIFICATION MODE and is stored in the bundle table along with that mode. Thus, the definition is immune to subsequent changes in the specification mode.

Line colour is defined in the current COLOUR SELECTION MODE and is stored in the bundle table along with that mode. Thus, the definition is immune to subsequent changes to the selection mode.

Which aspects are used depends on the corresponding ASFs; see the ASPECT SOURCE FLAG element.

Reference:

4.4.8

5.4.12 MARKER REPRESENTATION

Parameters:

marker bundle index (IX)
marker type (IX)
marker size specifier
    if marker size specification mode is 'absolute',
    absolute marker size (VDC)

    if marker size specification mode is 'scaled',
    marker size scale factor (R)

marker colour specifier
    if the colour selection mode is 'indexed',
    marker colour index (CI)

    if the colour selection mode is 'direct',
    marker colour value (CD)

Description:

In the marker bundle table, the given marker bundle index is associated with the specified parameters.

Marker type is specified and behaves as indicated in the MARKER TYPE attribute element.

Marker size is defined in the current MARKER SIZE SPECIFICATION MODE and is stored in the bundle table along with that mode. Thus, the definition is immune to subsequent changes in the specification mode.

Marker colour is defined in the current COLOUR SELECTION MODE and is stored in the bundle table along with that mode. Thus, the definition is immune to subsequent changes to the selection mode.

Which aspects are used depends on the corresponding ASFs; see the ASPECT SOURCE FLAG element.

Reference:

4.4.8

5.4.13 TEXT REPRESENTATION

Parameters:

text bundle index (IX)
font index (IX)
text precision (one of: string, character, stroke) (E)
character spacing (R)
character expansion factor (R)
text colour specifier
    if the colour selection mode is 'indexed',
    text colour index (CI)

    if the colour selection mode is 'direct',
    text colour value (CD)

Description:

In the text bundle table, the given text bundle index is associated with the specified parameters.

Font index is specified and behaves as indicated in the TEXT FONT INDEX attribute element.

Text precision is specified and behaves as indicated in the TEXT PRECISION attribute element.

Character spacing is specified and behaves as indicated in the CHARACTER SPACING attribute element.

Character expansion factor is specified and behaves as indicated in the CHARACTER EXPANSION FACTOR attribute element.

Text colour is defined in the current COLOUR SELECTION MODE and is stored in the bundle table along with that mode. Thus, the definition is immune to subsequent changes to the selection mode.

Which aspects are used depends on the corresponding ASFs; see the ASPECT SOURCE FLAG element.

Reference:

4.4.8

5.4.14 FILL REPRESENTATION

Parameters:

fill bundle index (IX)
interior style (one of: hollow, solid, pattern, hatch, empty)(E)
fill colour specifier
    if the colour selection mode is 'indexed',
    fill colour index (CI)

    if the colour selection mode is 'direct',
    fill colour value (CD)

hatch index (IX)
pattern index (IX)

Description:

In the fill bundle table, the given fill bundle index is associated with the specified parameters.

Interior style is specified and behaves as indicated in the INTERIOR STYLE attribute element.

Fill colour is defined in the current COLOUR SELECTION MODE and is stored in the bundle table along with that mode. Thus, the definition is immune to subsequent changes to the selection mode.

Hatch index is specified and behaves as indicated in the HATCH INDEX attribute element.

Pattern index is specified and behaves as indicated in the PATTERN INDEX attribute element.

Which aspects are used depends on the corresponding ASFs; see the ASPECT SOURCE FLAG element.

Reference:
4.4.8

5.4.15 EDGE REPRESENTATION

Parameters:

edge bundle index (IX)
edge type (IX)
edge width specifier
if edge width specification mode is 'absolute',
absolute edge width (VDC)

if edge width specification mode is 'scaled',
edge width scale factor (R)

edge colour specifier
if the colour selection mode is 'indexed',
edge colour index (CI)

if the colour selection mode is 'direct',
edge colour value (CD)

Description:
In the edge bundle table, the given edge bundle index is associated with the specified parameters.

Edge type is specified and behaves as indicated in the EDGE TYPE attribute element.

Edge width is defined in the current EDGE WIDTH SPECIFICATION MODE and is stored in the bundle table along with that mode. Thus, the definition is immune to subsequent changes in the specification mode.

Edge colour is defined in the current COLOUR SELECTION MODE and is stored in the bundle table along with that mode. Thus, the definition is immune to subsequent changes to the selection mode.

Which aspects are used depends on the corresponding ASFs; see the ASPECT SOURCE FLAG element.

Reference:
4.4.8

Page 61

Add the following after 5.5.6:

5.5.7 LINE CLIPPING MODE

Parameters:

mode (one of: locus, shape, locus then shape) (E)

Description:
The Line Clipping Mode is set to the value specified.

References:
4.5.2
4.6.1.3

5.5.8 MARKER CLIPPING MODE

Parameters:

mode (one of: locus, shape, locus then shape) (E)

Description:
The Marker Clipping Mode is set to the value specified.

References:
4.5.2
4.6.2.3

5.5.9 EDGE CLIPPING MODE

Parameters:

mode (one of: locus, shape, locus then shape) (E)

Description:
The Edge Clipping Mode is set to the value specified.

References:
4.5.2
4.6.4.5

5.5.10 NEW REGION

Parameters:

none

Description:
This element is used for control of subregion construction within closed figures.

If the current region has not yet been closed by a preceding NEW REGION element and if the last point of the last line element is not coincident with the current closure point, then the current subregion is closed by a line segment connecting the last point of the preceding line element to the current closure point. This line becomes a part of the implicit boundary specification. If the NEW REGION was preceded by a CONNECTING EDGE element, which was itself preceded by a line primitive, then this line also becomes part of the edge specification. If the region which has been previously closed is empty, or if the last point of the last line element is coincident with the current closure point, or if the last element was a filled-area primitive then no line segment is generated by this element.

The first point of the next line element following a NEW REGION element becomes the new closure point, starting a new subregion.

Reference:
4.6.8

5.5.11 SAVE PRIMITIVE CONTEXT

Parameters:

context name (N)

Description:
This element allows for the grouping and identification of the set of current values of the attribute and control elements listed in the list below as a single named entity.

Groups of elements may be saved in a picture or segment (local or global) using the context name.

The attribute and control elements which may be saved by SAVE PRIMITIVE CONTEXT and restored by RESTORE PRIMITIVE CONTEXT are:

LINE BUNDLE INDEX
LINE TYPE
LINE WIDTH (Note 1)
LINE COLOUR (Note 1)
LINE CLIPPING MODE

MARKER BUNDLE INDEX
MARKER TYPE
MARKER SIZE (Note 1)
MARKER COLOUR (Note 1)
MARKER CLIPPING MODE

TEXT BUNDLE INDEX
TEXT FONT INDEX
TEXT PRECISION
CHARACTER EXPANSION FACTOR
CHARACTER SPACING
TEXT COLOUR (Note 1)
CHARACTER HEIGHT
CHARACTER ORIENTATION
TEXT PATH
TEXT ALIGNMENT
CHARACTER SET INDEX
ALTERNATE CHARACTER SET INDEX

FILL BUNDLE INDEX
INTERIOR STYLE
FILL COLOUR (Note 1)
HATCH INDEX
PATTERN INDEX
EDGE BUNDLE INDEX
EDGE TYPE
EDGE WIDTH (Note 1)
EDGE COLOUR (Note 1)
EDGE VISIBILITY
EDGE CLIPPING MODE
FILL REFERENCE POINT (Note 2)
PATTERN SIZE

PICK IDENTIFIER

CLIP INDICATOR
CLIP RECTANGLE (Note 2)
AUXILIARY COLOUR (Note 1)
TRANSPARENCY

ASPECT SOURCE FLAGS

NOTES

1 The corresponding specification mode or selection mode in which this value was last set is also recorded., this will not cause an implicit change of mode on interpretation of RESTORE PRIMITIVE CONTEXT (see 4.5.3).

2 The VDC TYPE in effect when these values are saved is also recorded.

Reference:
4.5.3

5.5.12 RESTORE PRIMITIVE CONTEXT

Parameters:

context name (N)

Description:
The attribute and control set recorded in the metafile in the saved context name set by the SAVE PRIMITIVE CONTEXT element is recalled on interpretation.

Reference:
4.5.3

Page 63

Add the following text to the end of the second paragraph of 5.6.3

These instructions for the actual displayed portions of a marker apply only to MARKER CLIPPING MODE 'locus'.

Page 65

Sub-clause 5.6.6: In the NOTE replace the words "into the PICTURE OPEN state" by:

"back to the state that pertained when the text element initiating the string occurred."

Page 78

Add the following after 5.6.19:

5.6.20 CIRCULAR ARC CENTRE REVERSED

Parameters:

centrepoint (P)
DX_start, DY_start, DX_end, DY_end (4VDC)
radius (VDC)

Description:

A circular arc is drawn which is defined as follows:

DX_start and DY_start define a start vector, and DX_end and DY_end define an end vector. The tails of these vectors are placed on the centrepoint. A start ray and end ray are derived from the start and end vectors. The start and end rays are semi-infinite lines from the centrepoint in the directions of the start and end vectors respectively.

The specified radius and centrepoint define a circle. The arc is drawn in the negative angular direction (as defined by VDC EXTENT) from the intersection of the circle and the start ray (as obtained by measuring a distance 'radius' along the start ray from the centrepoint) to the intersection of the circle and the end ray.

The arc is displayed with current line element attributes.

Valid values of the vector components are those which produce vectors of non-zero length.

Valid values of 'radius' are non-negative VDC.

If the start ray and end ray are coincident, it is ambiguous whether the defined arc subtends 0^{o} or 360^{o} of central angle (see the specifications for the CIRCULAR ARC CENTRE in annex D).

Reference:

4.6

5.6.21 CONNECTING EDGE

Parameters:

none

Description:

During the construction of a closed figure a line segment connecting the last point of the preceding line element and the next point is added to the boundary and edge definitions. The next point may be either:

1) the first point of the next line element, or

2) the current closure point (in cases where CONNECTING EDGE is followed by either NEW REGION or END FIGURE).

The appearance of the connecting edge is fully determined by the edge attributes including EDGE VISIBILITY.

Reference:
4.6.8

Page 98

Add the following after 5.7.35:

5.7.36 PICK IDENTIFIER

Parameters:

pick identifier (N)

Description:
The pick identifier value is associated with all of the graphical primitive elements of a segment until the next PICK IDENTIFIER element. Usage of the PICK IDENTIFIER on interpretation is dependent upon the application.

Reference:
4.7.9

Page 100

Add the following after 5.9:

5.10 Segment elements

5.10.1 Segment control elements

5.10.1.1 COPY SEGMENT

Parameters:

segment identifier (N)
copy transformation matrix:
scaling and rotation portion (2 x 2) (R)
translation portion (2 x 1) (VDC)
segment transformation application (one of: no, yes) (E)

Description:
The segment which is indicated by the segment identifier is referenced at this point in the metafile for copying into the picture, or into a segment when referenced from a segment, on interpretation. The identified segment is referred to as the copied segment. With the possible exception of the segment transformation associated with the copied segment the segment attributes of the copied segment are ignored. The segment attributes of a segment in which the COPY SEGMENT may occur are unchanged by this element.

The copy transformation is applied to all graphic objects of the copied segment before they are copied into the picture or into the segment. The copy transformation is also applied to clipping rectangles under some circumstances.

The INHERITANCE FILTER element allows for control of the control and attribute values which are used when copying segments. This filter controls whether values of individual attribute and control elements are reapplied to the graphic objects. The effects of INHERITANCE FILTER are described in 4.12.5. The way in which clipping is applied to primitives within a copied segment is controlled by CLIP INHERITANCE (see 4.12.5).

The 'segment transformation application' parameter controls whether or not the segment transformation associated with the copied segment will be applied as an effect of the copy process. In no case is the segment transformation applied to a clip rectangle associated with a copied graphic object. In case the 'segment transformation application' is 'yes', the segment transformation is applied prior to the copy transformation.

References:
4.12.1
4.12.5

5.10.1.2 INHERITANCE FILTER

Parameters:

filter selection list (list of elements or groups from:
LINE BUNDLE INDEX
LINE TYPE
LINE WIDTH
LINE COLOUR
LINE CLIPPING MODE
MARKER BUNDLE INDEX
MARKER TYPE
MARKER SIZE
MARKER COLOUR
MARKER CLIPPING MODE
TEXT BUNDLE INDEX
TEXT FONT INDEX
TEXT PRECISION
CHARACTER EXPANSION FACTOR
CHARACTER SPACING
TEXT COLOUR
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
EDGE CLIPPING MODE
FILL REFERENCE POINT
PATTERN SIZE
AUXILIARY COLOUR
TRANSPARENCY
LINE ATTRIBUTES
MARKER ATTRIBUTES
TEXT REPRESENTATION AND PLACEMENT ATTRIBUTES
TEXT PLACEMENT AND ORIENTATION ATTRIBUTES
FILL ATTRIBUTES
EDGE ATTRIBUTES
PATTERN ATTRIBUTES
OUTPUT CONTROL
PICK IDENTIFIER
ALL ATTRIBUTES AND CONTROL
ALL
LINE TYPE ASF
LINE WIDTH ASF
LINE COLOUR ASF
MARKER TYPE ASF
MARKER SIZE ASF

Page 127

Add the following text at the end of the APPEND TEXT recommendations:

Changing the TEXT PRECISION in Text Open (partial text) state is likely to lead to unpredictable results. Generators are discouraged from doing this. Interpreters that can otherwise implement text attribute changes in partial text state should ignore this element in that state as a fallback.

Page 128

Sub-clause D.4.5: Add the following text between CIRCULAR ARC CENTRE CLOSE and Elliptical elements:

CIRCULAR ARC CENTRE REVERSED

If the start ray and end ray coincide, it is recommended that the interpreter draw the full circle.

Page 132

Add the following after D.4.8:

D.4.9 Segment elements

The restriction that segment attributes be set only immediately after the BEGIN SEGMENT element and before any other element avoids any dynamic effects.

SEGMENT DISPLAY PRIORITY

If the output device cannot adjust segment display priority on interpretation, segments should be displayed in order of occurrence.

COPY SEGMENT with CLIP INHERITANCE 'intersection'

If the interpreters cannot handle clip rectangles transformed by a copy transformation with non-zero off-diagonal elements (resulting in a parallelogram) the suggested fallback is to clip to an effective clip rectangle which is the smallest axis-aligned rectangle that contains the transformed clip rectangle. Similarly, in the case where multiple parallelograms might be composed (by intersection) to form a general convex polygon, interpreters should intersect the circumscribing rectangles to derive an effective clip rectangle.

Page 133

Sub-clause D5. Change the words in the first sentence from "..the capabilities shown in table 5" to:

"..the capabilities listed in tables 5a and 5b, appropriate to the version of the metafile they are supporting"

Page 133

Sub-clause D5. Change the title for Table 5 to:

Table 5a - Suggested minimum capabilities for version 1 metafiles.

Page 133

Sub-clause D.5 Add the following table after Table 5a:

Table 5b - Suggested additional minimum capabilities for version 2 metafiles.

Capability	Minimum Suggested Interpreter Support
DEVICE VIEWPORT SPECIFICATION MODE	fraction of display surface
DEVICE VIEWPORT MAPPING	not forced, forced
	left, centre, right
	bottom, centre, top
LINE REPRESENTATION	5 entries
MARKER REPRESENTATION	5 entries
TEXT REPRESENTATION	2 entries
FILL REPRESENTATION	5 entries
EDGE REPRESENTATION	5 entries
LINE CLIPPING MODE	locus, shape, locus then shape
MARKER CLIPPING MODE	locus, shape, locus then shape
EDGE CLIPPING MODE	locus, shape, locus then shape
SAVE/RESTORE PRIMITIVE CONTEXT	5 simultaneously saved control and attribute element sets
closed figure	an arbitrary mix containing at least one of the eligible graphical primitives, with POLYGON (SET) supporting at least 128 vertices
segments	64 simultaneously existing segments

Annexes F,G and H are new Annexes and are to be inserted after page 144

Annex F

Formal Grammar of the functional specification of version 2 metafiles

(normative)

F.1 Introduction

This grammar is a formal definition of a standard CGM extended syntax for version 2 metafiles. The encoding-independent and the encoding-dependent productions are separated, and there are subsections showing the syntax of each of the standardized encoding schemes. Details on the encoding of terminal symbols can be found in parts of this Standard that deal with the particular encoding schemes.

F.2 Notation used

<symbol>	- nonterminal
<SYMBOL>	- terminal
<symbol>*	- 0 or more occurrences
<symbol>+	- 1 or more occurrences
<symbol>o	- optional (0 or 1 occurrences)
<symbol>(n)	- exactly n occurrences, n=2,3,...
<symbol-1> ::= <symbol-2>	- symbol-1 has the syntax of symbol-2
<symbol-1> | <symbol-2>	- symbol-1 or alternatively symbol-2
<symbol: meaning>	- symbol with the stated meaning
{comment}	- explanation of a symbol or a production

F.3 Detailed grammar

F.3.1 Metafile structure

```
<metafile>              ::= <BEGIN METAFILE>
                                <metafile identifier>
                            <metafile descriptor>
                            <metafile contents>*
                            <END METAFILE>

<metafile identifier>   ::= <string>

<metafile contents>     ::= <extra element>*
                            <picture>
                            <extra element>*

<extra element>         ::= <external element>
                          | <escape element>

<picture>               ::= <BEGIN PICTURE>
                                <picture identifier>
                            <picture descriptor element>*
                            <BEGIN PICTURE BODY>
                            <picture content>*
                            <END PICTURE>

<picture identifier>    ::= <string>

<picture content>       ::= <picture element>
                          | &lt;segment&gt;
```

<picture element> ::= <control element>
| <graphical element>
| <closed figure>
| <primitive attribute element>
| <pattern table element>
| <colour table element>
| <specification element>
| <segment control element>
| <extra element>

<segment> ::= <BEGIN SEGMENT>
<segment identifier>
<segment attribute element>*
<eligible picture element>*
<END SEGMENT>

<segment identifier> ::= <name>

<eligible picture element> ::= <control element>
| <graphical element>
| <closed figure>
| <primitive attribute element>
| <specification element>
| <segment control element>
| <extra element>

F.3.2 Metafile descriptor elements

<metafile descriptor> ::= <<optional descriptor element>*
<version>
<optional descriptor element>*
<element list>
<optional descriptor element>*>
| <<optional descriptor element>*
<element list>
<optional descriptor element>*
<version>
<optional descriptor element>*>

<version> ::= <METAFILE VERSION>
<integer>

<element list> ::= <METAFILE ELEMENT LIST>
<element name>*
| <element name shorthand enumerated>*

<element name shorthand enumerated> ::= <DRAWING SET>
| <DRAWING PLUS CONTROL SET>
| <VERSION 2 SET>
| <EXTENDED PRIMITIVES SET>
| <VERSION 2 GKSM SET>

<optional descriptor element> ::= <description>
| <VDC TYPE>
<vdc type enumerated>
| <MAXIMUM COLOUR INDEX>
<colour index>
| <COLOUR VALUE EXTENT>

```
                                     <red green blue>(2)
                               | <METAFILE DEFAULTS REPLACEMENT>
                                     <element default>+
                               | <FONT LIST>
                                     <font name>+
                               | <CHARACTER SET LIST>
                                     <character set definition>+
                               | <CHARACTER CODING ANNOUNCER>
                                     <coding technique enumerated>
                               | <scalar precision>
                               | <MAXIMUM VDC EXTENT>
                                     <point> (2)
                               | &lt;SEGMENT PRIORITY EXTENT&gt;
                                     <minimum extent>
                                     <maximum extent>
                               | &lt;segment&gt;
                               | <extra element>

<description>                ::= <METAFILE DESCRIPTION>
                                     <string>

<vdc type enumerated>        ::= <INTEGER>
                               | <REAL>

<element default>            ::= <control element>
                               | <picture descriptor element>
                               | <primitive attribute element>
                               | <extra element>

<font name>                  ::= <string>

<character set definition>   ::= <char set enumerated>
                                 <designation sequence>

<index>                      ::= <standard index value>
                               | <private index value>

<standard index value>       ::= <positive integer>
<non-negative integer>       ::= <integer>    {greater or equal to 0}
<positive integer>           ::= <integer>    {greater than 0}
<private index value>        ::= <negative integer>
<negative integer>           ::= <integer>    {less than 0}
<positive index>             ::= <positive integer>

<character set enumerated>   ::= <94 CHAR>
                               | <96 CHAR>
                               | <MULTI-BYTE 94 CHAR>
                               | <MULTI-BYTE 96 CHAR>
                               | <COMPLETE CODE>

<coding technique enumerated> ::= <BASIC 7-BIT>
                               | <BASIC 8-BIT>
                               | <EXTENDED 7-BIT>
                               | <EXTENDED 8-BIT>

<designation sequence>       ::= <string>

<scalar precision>           ::= <INTEGER PRECISION>
                                     <integer precision value>
                               | <REAL PRECISION>
                                     <real precision value>
```

| <INDEX PRECISION>
<index precision value>
| <COLOUR PRECISION>
<colour precision value>
| <COLOUR INDEX PRECISION>
<colour index precision value>
| <NAME PRECISION>
<name precision value>
{these elements have encoding}
{dependent parameters}

<point> ::= <vdc value> (2)

<minimum extent> ::= <integer>

<maximum extent> ::= <integer>

F.3.3 Picture descriptor elements

<picture descriptor element> ::= <SCALING MODE>
<scaling specification mode enumerated>
<metric scale factor>
| <VDC EXTENT>
<point> (2)
| <DEVICE VIEWPORT>
<viewport point>(2)
| <DEVICE VIEWPORT SPECIFICATION MODE
<VC specifier enumerated>
<metric scale factor>
| <DEVICE VIEWPORT MAPPING>
<isotropy flag enumerated>
<horizontal alignment flag enumerated>
<vertical alignment flag enumerated>
| <BACKGROUND COLOUR>
<red green blue>
| <specification element>
| <representation element>
| <pattern table element>
| <colour table element>
| <extra element>

<specification element> ::= <COLOUR SELECTION MODE>
<colour selection mode enumerated>
| <LINE WIDTH SPECIFICATION MODE>
<specification mode enumerated>
| <MARKER SIZE SPECIFICATION MODE>
<specification mode enumerated>
| <EDGE WIDTH SPECIFICATION MODE>
<specification mode enumerated>

<colour selection mode enumerated> ::= <INDEXED>
| <DIRECT>

<scaling specification mode enumerated> ::= <ABSTRACT>
| <METRIC>

<metric scale factor> ::= <real>

<isotropy flag enumerated> ::= <NOT FORCED>

```
                                | <FORCED>

<horizontal alignment flag
  enumerated>                   ::= <LEFT>
                                  | <CENTRE>
                                  | <RIGHT>

<vertical alignment flag
  enumerated>                   ::= <BOTTOM>
                                  | <CENTRE>
                                  | <TOP>

<specification mode enumerated> ::= <ABSOLUTE>
                                  | <SCALED>

<viewport point>                ::= <vc value> (2)

<VC specifier enumerated>       ::= <FRACTION OF DISPLAY SURFACE>
                                  | <MILLIMETRES WITH SCALE FACTOR>
                                  | <PHYSICAL DEVICE COORDINATES>

<representation element>        ::= <LINE REPRESENTATION>
                                        <positive index>
                                        <index>  {line type}
                                        <size value>  {line width}
                                        <colour>
                                  | <MARKER REPRESENTATION>
                                        <positive index>
                                        <index>  {marker type}
                                        <size value>
                                        <colour>
                                  | <TEXT REPRESENTATION>
                                        <positive index>
                                        <positive index>  {font}
                                        <text precision enumerated>
                                        <real>  {character spacing}
                                        <real>  {expansion factor}
                                        <colour>
                                  | <FILL REPRESENTATION>
                                        <positive index>
                                        <interior style enumerated>
                                        <colour>
                                        <index> {hatch index}
                                        <positive index>{pattern index}
                                  | <EDGE REPRESENTATION>
                                        <positive index>
                                        <index>  {edge type}
                                        <size value>  {edge width}
                                        <colour>

<size value>                    ::= <non-negative vdc value>
                                  | <non-negative real>

<non-negative vdc value>        ::= <vdc value>  {greater or equal to 0}

<non-negative real>             ::= <real>  {greater or equal to 0}

<colour>                        ::= <colour index>
                                  | <red green blue>
```

<text precision enumerated> ::= <STRING>
| <CHARACTER>
| <STROKE>

<interior style enumerated> ::= <HOLLOW>
| <SOLID>
| <PATTERN>
| <HATCH>
| <EMPTY>

F.3.4 Control elements

<control element> ::= <vdc precision>
| <AUXILIARY COLOUR>
<colour>
| <TRANSPARENCY>
<on-off indicator enumerated>
| <CLIP RECTANGLE>
<point>(2)
| <CLIP INDICATOR>
<on-off indicator enumerated>
| <LINE CLIPPING MODE>
<clip mode enumerated>
| <MARKER CLIPPING MODE>
<clip mode enumerated>
| <EDGE CLIPPING MODE>
<clip mode enumerated>
| <SAVE PRIMITIVE CONTEXT>
<context name>
| <RESTORE PRIMITIVE CONTEXT>
<context name>

<on-off indicator enumerated> ::= <ON>
| <OFF>

<vdc precision> ::= <VDC INTEGER PRECISION>
<vdc integer precision value>
| <VDC REAL PRECISION>
<vdc real precision value>
{these elements have encoding}
{dependent parameters}

<clip mode enumerated> ::= <LOCUS>
| <SHAPE>
| <LOCUS THEN SHAPE>

<context name> ::= <name>

F.3.5 Graphical elements

<graphical element> ::= <polypoint element>
| <text element>
| <cell element>
| <gdp element>
| <rectangle element>
| <circular element>
| <elliptical element>
| <pointless element>

```
<polypoint element>          ::= <POLYLINE>
                                     <point pair>
                                     <point list>
                               | <DISJOINT POLYLINE>
                                     <point pair>
                                     <point pair list>
                               | <POLYMARKER>
                                     <point>
                                     <point list>
                               | <POLYGON>
                                     <point>(3)
                                     <point list>
                               | <POLYGON SET>
                                     <point edge pair>(3)
                                     <point edge pair list>

<point list>                 ::= <point>*

<point pair list>            ::= <point pair>*

<point pair>                 ::= <point>(2)

<point edge pair>            ::= <point><edge out flag>

<point edge pair list>       ::= <point edge pair>*

<edge out flag>              ::= <INVISIBLE>
                               | <VISIBLE>
                               | <CLOSE INVISIBLE>
                               | <CLOSE VISIBLE>

<text element>               ::= <TEXT>
                                     <point>
                                     <text tail>
                               | <restricted text element>

<restricted text element>    ::= <RESTRICTED TEXT>
                                     <extent>
                                     <point>
                                     <text tail>

<extent>                     ::= <vdc value>(2)

<text tail>                  ::= <final character list>
                               | <nonfinal character list>

<final character list>       ::= <FINAL>
                                     <string>

<nonfinal character list>    ::= <NOT FINAL>
                                     <string>
                                     <partial text attribute element>*
                                     <spanned text>

<spanned text>               ::= <APPEND TEXT>
                                     <text tail>

<cell element>               ::= <CELL ARRAY>
                                     <point>(3)
                                     <integer>(2)
                                     <local colour precision>
```

<colour>(integer1 x integer2)

{this element has an encoding}
{dependent parameter}

<local colour precision> ::= <colour precision value>
| <colour index precision value>
| <default colour precision indicator>

<gdp element> ::= <GDP>
<gdp identifier>
<point list>
<data record>

<gdp identifier> ::= <integer>

<rectangle element> ::= <RECTANGLE>
<point pair>

<circular element> ::= <CIRCLE>
<point>
<radius>
| <CIRCULAR ARC 3 POINT>
<point>(3)
| <CIRCULAR ARC 3 POINT CLOSE>
<point>(3)
<close type>
| <CIRCULAR ARC CENTRE>
<point>
<vdc value>(4)
<radius>
| <CIRCULAR ARC CENTRE CLOSE>
<point>
<vdc value>(4)
<radius>
<close type>
| <CIRCULAR ARC CENTRE REVERSED>
<point>
<vdc value>(4)
<radius>

<radius> ::= <non-negative vdc value>

<close type> ::= <PIE>
| <CHORD>

<elliptical element> ::= <ELLIPSE>
<point>(3)
| <ELLIPTICAL ARC>
<point>(3)
<vdc value>(4)
| <ELLIPTICAL ARC CLOSE>
<point>(3)
<vdc value>(4)
<close type>

<pointless element> ::= <CONNECTING EDGE>

F.3.6 Attribute elements

<primitive attribute element> ::= <line attribute element>
| <marker attribute element>
| <text attribute element>
| <filled-area attribute element>
| <aspect source flags>
| <pick identifier>

<line attribute element> ::= <LINE BUNDLE INDEX>
<positive index>
| <LINE TYPE>
<index>
| <LINE WIDTH>
<size value>
| <LINE COLOUR>
<colour>

<marker attribute element> ::= <MARKER BUNDLE INDEX>
<positive index>
| <MARKER TYPE>
<index>
| <MARKER SIZE>
<size value>
| <MARKER COLOUR>
<colour>

<partial text attribute element> ::= <TEXT FONT INDEX>
<positive index>
| <TEXT PRECISION>
<text precision enumerated>
| <CHARACTER EXPANSION FACTOR>
<real>
| <CHARACTER SPACING>
<real>
| <TEXT COLOUR>
<colour>
| <CHARACTER HEIGHT>
<non-negative vdc value>
| <CHARACTER SET INDEX>
<positive index>
| <ALTERNATE CHARACTER SET INDEX>
<positive index>
| <TEXT BUNDLE INDEX>
<positive index>
| <AUXILIARY COLOUR>
<colour>
| <TRANSPARENCY>
<on-off indicator enumerated>

<text attribute element> ::= <TEXT BUNDLE INDEX>
<positive index>
| <TEXT FONT INDEX>
<positive index>
| <TEXT PRECISION>
<text precision enumerated>
| <CHARACTER EXPANSION FACTOR>
<real>
| <CHARACTER SPACING>
<real>
| <TEXT COLOUR>

```
                    <colour>
                | <CHARACTER HEIGHT>
                    <non-negative vdc value>
                | <CHARACTER ORIENTATION>
                    <vdc value>(4)
                | <TEXT PATH>
                    <path enumerated>
                | <TEXT ALIGNMENT>
                    <horizontal alignment enumerated>
                    <vertical alignment enumerated>
                    <continuous alignment value> (2)
                | <CHARACTER SET INDEX>
                    <positive index>
                | <ALTERNATE CHARACTER SET INDEX>
                    <positive index>

<path enumerated>                   ::= <RIGHT>
                                      | <LEFT>
                                      | <UP>
                                      | <DOWN>

<horizontal alignment
   enumerated>                      ::= <NORMAL HORIZONTAL>
                                      | <LEFT>
                                      | <CENTRE>
                                      | <RIGHT>
                                      | <CONTINUOUS HORIZONTAL>

<vertical alignment enumerated>     ::= <NORMAL VERTICAL>
                                      | <TOP>
                                      | <CAP>
                                      | <HALF>
                                      | <BASE>
                                      | <BOTTOM>
                                      | <CONTINUOUS VERTICAL>

<continuous alignment value>        ::= <real>

<filled-area attribute element>     ::= <FILL BUNDLE INDEX>
                                          <positive index>
                                      | <INTERIOR STYLE>
                                          <interior style enumerated>
                                      | <FILL COLOUR>
                                          <colour>
                                      | <HATCH INDEX>
                                          <index>
                                      | <PATTERN INDEX>
                                          <positive index>
                                      | <EDGE BUNDLE INDEX>
                                          <positive index>
                                      | <EDGE TYPE>
                                          <index>
                                      | <EDGE WIDTH>
                                          <size value>
                                      | <EDGE COLOUR>
                                          <colour>
                                      | <EDGE VISIBILITY>
                                          <on-off indicator enumerated>
                                      | <FILL REFERENCE POINT>
                                          <point>
```

| <PATTERN SIZE>
<vdc value>(4)

<colour table element> ::= <COLOUR TABLE>
<starting index>
<red green blue>+

<pattern table element> ::= <PATTERN TABLE>
<positive index>
<integer>(2)
<local colour precision>
<colour>(integer1 x integer2)
{this element has an encoding}
{dependent parameter}

<starting index> ::= <colour index>

<●ect source flags> ::= <ASPECT SOURCE FLAGS>
<asf pair>+

<asf pair> ::= <asf type enumerated>
<asf enumerated>

<●type enumerated> ::= <LINE TYPE ASF>
| <LINE WIDTH ASF>
| <LINE COLOUR ASF>
| <MARKER TYPE ASF>
| <MARKER SIZE ASF>
| <MARKER COLOUR ASF>
| <TEXT FONT ASF>
| <TEXT PRECISION ASF>
| <CHARACTER EXPANSION FACTOR ASF>
| <CHARACTER SPACING ASF>
| <TEXT COLOUR ASF>
| <INTERIOR STYLE ASF>
| <FILL COLOUR ASF>
| <HATCH INDEX ASF>
| <PATTERN INDEX ASF>
| <EDGE TYPE ASF>
| <EDGE WIDTH ASF>
| <EDGE COLOUR ASF>

<asf enumerated> ::= <INDIVIDUAL>
| <BUNDLED>

<pick identifier> ::= <PICK IDENTIFIER>
<name>

F.3.7 Closed figure element

<closed figure> ::= <BEGIN FIGURE>
<eligible elements within closed figures>
<END FIGURE>

<eligible elements within closed figures> ::= <VDC REAL PRECISION>
| <VDC INTEGER PRECISION>
| <AUXILIARY COLOUR>
| <TRANSPARENCY>
| <NEW REGION>

```
                                | <POLYLINE>
                                | <DISJOINT POLYLINE>
                                | <POLYGON>
                                | <POLYGON SET>
                                | <GDP>
                                | <RECTANGLE>
                                | <CIRCLE>
                                | <CIRCULAR ARC 3 POINT>
                                | <CIRCULAR ARC 3 POINT CLOSE>
                                | <CIRCULAR ARC CENTRE>
                                | <CIRCULAR ARC CENTRE CLOSE>
                                | <CIRCULAR ARC CENTRE REVERSED>
                                | <ELLIPSE>
                                | <ELLIPTICAL ARC>
                                | <ELLIPTICAL ARC CLOSE>
                                | <CONNECTING EDGE>
                                | <EDGE BUNDLE INDEX>
                                | <EDGE TYPE>
                                | <EDGE WIDTH>
                                | <EDGE COLOUR>
                                | <EDGE VISIBILITY>
                                | <EDGE TYPE ASF>
                                | <EDGE WIDTH ASF>
                                | <EDGE COLOUR ASF>
                                | <ESCAPE>
                                | <MESSAGE>
                                | <APPLICATION DATA>
```

F.3.8 Escape elements

```
<escape element>              ::= <ESCAPE>
                                      <identifier>
                                      <data record>

<identifier>                  ::= <integer>
```

F.3.9 External elements

```
<external element>            ::= <MESSAGE>
                                      <action flag enumerated>
                                      <string>
                                | <APPLICATION DATA>
                                      <integer>
                                      <data record>

<action flag enumerated>      ::= <YES>
                                | <NO>
```

F.3.10 Segment elements

```
&lt;segment control element&gt;     ::= <COPY SEGMENT>
                                      &lt;segment identifier&gt;
                                      <copy transformation matrix>
                                      &lt;segment transformation application&gt;
                                | <INHERITANCE FILTER>
                                      <filter selection list enumerated>*
                                      <selection setting enumerated>
                                | <CLIP INHERITANCE>
```

<clip inheritance enumerated>

<segment attribute element> ::= <SEGMENT TRANSFORMATION>
<segment identifier>
<transformation matrix>
| <SEGMENT HIGHLIGHTING>
<segment identifier>
<highlighting enumerated>
| <SEGMENT DISPLAY PRIORITY>
<segment identifier>
<segment display priority>
| <SEGMENT PICK PRIORITY>
<segment identifier>
<segment pick priority>

<copy transformation matrix> ::= <transformation matrix>

<transformation matrix> ::= <2 x 2 matrix of reals>
<2 x 1 matrix of vdcs>

<segment transformation application> ::= <NO>
| <YES>

<filter selection list enumerated> ::= <attribute and control name enumerated>
| <attribute and control group enumerated>
| <asf name enumerated>
| <asf group enumerated>

<attribute and control name enumerated> ::= <LINE BUNDLE INDEX>
| <LINE TYPE>
| <LINE WIDTH>
| <LINE COLOUR>
| <LINE CLIPPING MODE>
| <MARKER BUNDLE INDEX>
| <MARKER TYPE>
| <MARKER SIZE>
| <MARKER COLOUR>
| <MARKER CLIPPING MODE>
| <TEXT BUNDLE INDEX>
| <TEXT FONT INDEX>
| <TEXT PRECISION>
| <CHARACTER EXPANSION FACTOR>
| <CHARACTER SPACING>
| <TEXT COLOUR>
| <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>

```
                                 | <EDGE CLIPPING MODE>
                                 | <FILL REFERENCE POINT>
                                 | <PATTERN SIZE>
                                 | <AUXILIARY COLOUR
                                 | <TRANSPARENCY>

<attribute and control
    group enumerated>          ::= <LINE ATTRIBUTES>
                                 | <MARKER ATTRIBUTES>
                                 | <TEXT PRESENTATION AND PLACEMENT ATTRIBUTES>
                                 | <TEXT PLACEMENT AND ORIENTATION ATTRIBUTES>
                                 | <FILL ATTRIBUTES>
                                 | <EDGE ATTRIBUTES>
                                 | <PATTERN ATTRIBUTES>
                                 | <OUTPUT CONTROL>
                                 | <PICK IDENTIFIER>
                                 | <ALL ATTRIBUTES AND CONTROL>
                                 | <ALL>

<selection setting enumerated> ::= <STATE LIST>
                                 | &lt;SEGMENT&gt;

<asf name enumerated>          ::= <LINE TYPE ASF>
                                 | <LINE WIDTH ASF>
                                 | <LINE COLOUR ASF>
                                 | <MARKER TYPE ASF>
                                 | <MARKER SIZE ASF>
                                 | <MARKER COLOUR ASF>
                                 | <TEXT FONT INDEX ASF>
                                 | <TEXT PRECISION ASF>
                                 | <CHARACTER EXPANSION FACTOR ASF>
                                 | <CHARACTER SPACING ASF>
                                 | <TEXT COLOUR ASF>
                                 | <INTERIOR STYLE ASF>
                                 | <FILL COLOUR ASF>
                                 | <HATCH INDEX ASF>
                                 | <PATTERN INDEX ASF>
                                 | <EDGE TYPE ASF>
                                 | <EDGE WIDTH ASF>
                                 | <EDGE COLOUR ASF>

<asf group enumerated>         ::= <LINE ASFS>
                                 | <MARKER ASFS>
                                 | <TEXT ASFS>
                                 | <FILL ASFS>
                                 | <EDGE ASFS>
                                 | <ALL ASFS>

<clip inheritance enumerated>  ::= <STATE LIST>
                                 | <INTERSECTION>

<highlighting enumerated>      ::= <NORMAL>
                                 | <HIGHLIGHTED>

&lt;segment display priority&gt;   ::= <integer>

&lt;segment pick priority&gt;      ::= <integer>
```

F.4 Terminal symbols

The following are the terminals in this grammar. Their representation is dependent on the encoding scheme used. In annex A of the subsequent parts of this Standard, these encoding-dependent symbols are further described.

<element name>
<integer>
<real>
<vdc value>
<string>
<colour index>
<red green blue>
<integer precision value>
<real precision value>
<index precision value>
<colour precision value>
<colour index precision value>
<name precision value>
<default colour precision indicator>
<vdc integer precision value>
<vdc real precision value>
<data record>
<name>
<vc value>
<2 x 2 matrix of reals>
<2 x 1 matrix of vdcs>

The CGM extended opcodes are encoding dependent. A complete list of them can be found in the productions for <element name enumerated> below.

The enumerated types are:

<INTEGER>
<REAL>
<ON>
<OFF>
<INDEXED>
<DIRECT>
<ABSTRACT>
<METRIC>
<ABSOLUTE>
<SCALED>
<94 CHAR>
<96 CHAR>
<MULTI-BYTE 94 CHAR>
<MULTI-BYTE 96 CHAR>
<COMPLETE CODE>
<BASIC 7-BIT>
<BASIC 8-BIT>
<EXTENDED 7-BIT>
<EXTENDED 8-BIT>
<FRACTION OF DISPLAY SURFACE>
<MILLIMETRES WITH SCALE FACTOR>
<PHYSICAL DEVICE COORDINATES>
<NOT FORCED>
<FORCED>
<LEFT>
<RIGHT>
<CENTRE>
<BOTTOM>

<TOP>
<LOCUS>
<SHAPE>
<LOCUS THEN SHAPE>
<INVISIBLE>
<VISIBLE>
<CLOSE INVISIBLE>
<CLOSE VISIBLE>
<PIE>
<CHORD>
<FINAL>
<NOT FINAL>
<INDIVIDUAL>
<BUNDLED>
<HOLLOW>
<SOLID>
<PATTERN>
<HATCH>
<EMPTY>
<STRING>
<CHARACTER>
<STROKE>
<UP>
<DOWN>
<NORMAL HORIZONTAL>
<CONTINUOUS HORIZONTAL>
<NORMAL VERTICAL>
<CAP>
<HALF>
<BASE>
<CONTINUOUS VERTICAL>
<YES>
<NO>
<LINE TYPE ASF>
<LINE WIDTH ASF>
<LINE COLOUR ASF>
<MARKER TYPE ASF>
<MARKER SIZE ASF>
<MARKER COLOUR ASF>
<TEXT FONT ASF>
<TEXT PRECISION ASF>
<CHARACTER EXPANSION FACTOR ASF>
<CHARACTER SPACING ASF>
<TEXT COLOUR ASF>
<INTERIOR STYLE ASF>
<HATCH INDEX ASF>
<PATTERN INDEX ASF>
<FILL COLOUR ASF>
<EDGE TYPE ASF>
<EDGE WIDTH ASF>
<EDGE COLOUR ASF>
<LINE ATTRIBUTES>
<MARKER ATTRIBUTES>
<TEXT PRESENTATION AND PLACEMENT ATTRIBUTES>
<TEXT PLACEMENT AND ORIENTATION ATTRIBUTES>
<FILL ATTRIBUTES>
<EDGE ATTRIBUTES>
<PATTERN ATTRIBUTES>
<OUTPUT CONTROL>
<ALL ATTRIBUTES AND CONTROL>
<ALL>

<LINE BUNDLE INDEX>
<LINE TYPE>
<LINE WIDTH>
<LINE COLOUR>
<LINE CLIPPING MODE>
<MARKER BUNDLE INDEX>
<MARKER TYPE>
<MARKER SIZE>
<MARKER COLOUR>
<MARKER CLIPPING MODE>
<TEXT BUNDLE INDEX>
<TEXT FONT INDEX>
<TEXT PRECISION>
<CHARACTER EXPANSION FACTOR>
<CHARACTER SPACING>
<TEXT COLOUR>
<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>
<EDGE CLIPPING MODE>
<FILL REFERENCE POINT>
<PATTERN SIZE>
<AUXILIARY COLOUR>
<TRANSPARENCY>
<STATE LIST>
<INTERSECTION>

<LINE ASFS>
<MARKER ASFS>
<TEXT ASFS>
<FILL ASFS>
<EDGE ASFS>
<ALL ASFS>
<NORMAL>
<HIGHLIGHTED>
<DRAWING SET>
<DRAWING PLUS CONTROL SET>
<VERSION 2 SET>
<EXTENDED PRIMITIVES SET>
<VERSION 2 GKSM SET>

<element name enumerated> ::= <BEGIN METAFILE>
| <END METAFILE>
| <BEGIN PICTURE>
| <BEGIN PICTURE BODY>
| <END PICTURE>
| <BEGIN SEGMENT>
| <END SEGMENT>
| <BEGIN FIGURE>

| <END FIGURE>
| <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>
| <SCALING MODE>
| <COLOUR SELECTION MODE>
| <LINE WIDTH SPECIFICATION MODE>
| <MARKER SIZE SPECIFICATION MODE>
| <EDGE WIDTH SPECIFICATION MODE>
| <VDC EXTENT>
| <BACKGROUND COLOUR>
| <DEVICE VIEWPORT>
| <DEVICE VIEWPORT SPECIFICATION MODE>
| <DEVICE VIEWPORT MAPPING>
| <LINE REPRESENTATION>
| <MARKER REPRESENTATION>
| <TEXT REPRESENTATION>
| <FILL REPRESENTATION>
| <EDGE REPRESENTATION>
| <VDC INTEGER PRECISION>
| <VDC REAL PRECISION>
| <AUXILIARY COLOUR>
| <TRANSPARENCY>
| <CLIP RECTANGLE>
| <CLIP INDICATOR>
| <LINE CLIPPING MODE>
| <MARKER CLIPPING MODE>
| <EDGE CLIPPING MODE>
| <NEW REGION>
| <SAVE PRIMITIVE CONTEXT>
| <RESTORE PRIMITIVE CONTEXT>
| <POLYLINE>
| <DISJOINT POLYLINE>
| <POLYMARKER>
| <TEXT>
| <RESTRICTED TEXT>
| <APPEND TEXT>
| <POLYGON>
| <POLYGON SET>
| <CELL ARRAY>
| <GDP>
| <RECTANGLE>
| <CIRCLE>
| <CIRCULAR ARC 3 POINT>
| <CIRCULAR ARC 3 POINT CLOSE>
| <CIRCULAR ARC CENTRE>

| <CIRCULAR ARC CENTRE CLOSE>
| <CIRCULAR ARC CENTRE REVERSED>
| <ELLIPSE>
| <ELLIPTICAL ARC>
| <ELLIPTICAL ARC CLOSE>
| <CONNECTING EDGE>
| <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>
| <EDGE BUNDLE INDEX>
| <EDGE TYPE>
| <EDGE WIDTH>
| <EDGE COLOUR>
| <EDGE VISIBILITY>
| <FILL REFERENCE POINT>
| <PATTERN TABLE>
| <PATTERN SIZE>
| <COLOUR TABLE>
| <ASPECT SOURCE FLAGS>
| <PICK IDENTIFIER>
| <COPY SEGMENT>
| <INHERITANCE FILTER>
| <CLIP INHERITANCE>
| <SEGMENT TRANSFORMATION>
| <SEGMENT HIGHLIGHTING>
| <SEGMENT DISPLAY PRIORITY>
| <SEGMENT PICK PRIORITY>
| <ESCAPE>
| <MESSAGE>
| <APPLICATION DATA>

Annex G
Formal grammar of the functional specification of version 1 metafiles

(normative)

G.1 Introduction

This grammar is a formal definition of a standard CGM syntax for version 1 metafiles. The encoding-independent and the encoding-dependent productions are separated, and there are subsections showing the syntax of each of the standardized encoding schemes. Details on the encoding of terminal symbols can be found in parts of this Standard that deal with the particular encoding schemes.

G.2 Notation used

<symbol>	- nonterminal
<SYMBOL>	- terminal
<symbol>*	- 0 or more occurrences
<symbol>+	- 1 or more occurrences
<symbol>o	- optional (0 or 1 occurrences)
<symbol>(n)	- exactly n occurrences, n=2,3,...
<symbol-1> ::= <symbol-2>	- symbol-1 has the syntax of symbol-2
<symbol-1> | <symbol-2>	- symbol-1 or alternatively symbol-2
<symbol: meaning>	- symbol with the stated meaning
{comment}	- explanation of a symbol or a production

G.3 Detailed grammar

G.3.1 Metafile structure

```
<metafile>                  ::= <BEGIN METAFILE>
                                <metafile identifier>
                                <metafile descriptor>
                                <metafile contents>*
                                <END METAFILE>

<metafile identifier>       ::= <string>

<metafile contents>         ::= <extra element>*
                                <picture>
                                <extra element>*

<extra element>             ::= <external element>
                              | <escape element>

<picture>                   ::= <BEGIN PICTURE>
                                <picture identifier>
                                <picture descriptor element>*
                                <BEGIN PICTURE BODY>
                                <picture element>*
                                <END PICTURE>

<picture identifier>        ::= <string>

<picture element>           ::= <control element>
```

| <graphical element>
| <primitive attribute element>
| <pattern table element>
| <colour table element>
| <extra element>

G.3.2 Metafile descriptor elements

<metafile descriptor> ::= <<optional descriptor element>*
<version>
<optional descriptor element>*
<element list>
<optional descriptor element>*>
| <<optional descriptor element>*
<element list>
<optional descriptor element>*
<version>
<optional descriptor element>*>

<version> ::= <METAFILE VERSION>
<integer>

<element list> ::= <METAFILE ELEMENT LIST>
<element name>*
| <element name shorthand enumerated>*

<element name shorthand enumerated> ::= <DRAWING SET>
| <DRAWING PLUS CONTROL SET>

<optional descriptor element> ::= <description>
| <VDC TYPE>
<vdc type enumerated>
| <MAXIMUM COLOUR INDEX>
<colour index>
| <COLOUR VALUE EXTENT>
<red green blue>(2)
| <METAFILE DEFAULTS REPLACEMENT>
<element default>+
| <FONT LIST>
<font name>+
| <CHARACTER SET LIST>
<character set definition>+
| <CHARACTER CODING ANNOUNCER>
<coding technique enumerated>
| <scalar precision>
| <extra element>

<description> ::= <METAFILE DESCRIPTION>
<string>

<vdc type enumerated> ::= <INTEGER>
| <REAL>

<element default> ::= <control element>
| <picture descriptor element>
| <primitive attribute element>
| <extra element>

<font name> ::= <string>

```
<character set definition>      ::= <character set enumerated>
                                    <designation sequence>

<index>                         ::= <standard index value>
                                  | <private index value>

<standard index value>          ::= <positive integer>
<non-negative integer>          ::= <integer>    {greater or equal to 0}
<positive integer>              ::= <integer>    {greater than 0}
<private index value>           ::= <negative integer>
<negative integer>              ::= <integer>    {less than 0}
<positive index>                ::= <positive integer>

<character set enumerated>      ::= <94 CHAR>
                                  | <96 CHAR>
                                  | <MULTI-BYTE 94 CHAR>
                                  | <MULTI-BYTE 96 CHAR>
                                  | <COMPLETE CODE>

<coding technique enumerated>   ::= <BASIC 7-BIT>
                                  | <BASIC 8-BIT>
                                  | <EXTENDED 7-BIT>
                                  | <EXTENDED 8-BIT>

<designation sequence>          ::= <string>

<scalar precision>              ::= <INTEGER PRECISION>
                                        <integer precision value>
                                  | <REAL PRECISION>
                                        <real precision value>
                                  | <INDEX PRECISION>
                                        <index precision value>
                                  | <COLOUR PRECISION>
                                        <colour precision value>
                                  | <COLOUR INDEX PRECISION>
                                        <colour index precision value>
                                    {these elements have encoding}
                                    {dependent parameters}
```

G.3.3 Picture descriptor elements

```
<picture descriptor element>    ::= <SCALING MODE>
                                        <scaling specification mode enumerated>
                                        <metric scale factor>
                                  | <VDC EXTENT>
                                        <point> (2)
                                  | <BACKGROUND COLOUR>
                                        <red green blue>
                                  | <specification element>
                                  | <extra element>

<specification element>         ::= <COLOUR SELECTION MODE>
                                        <colour selection mode enumerated>
                                  | <LINE WIDTH SPECIFICATION MODE>
                                        <specification mode enumerated>
                                  | <MARKER SIZE SPECIFICATION MODE>
                                        <specification mode enumerated>
                                  | <EDGE WIDTH SPECIFICATION MODE>
                                        <specification mode enumerated>
```

<colour selection mode enumerated> ::= <INDEXED>
| <DIRECT>

<scaling specification mode enumerated> ::= <ABSTRACT>
| <METRIC>

<metric scale factor> ::= <real>

<specification mode enumerated> ::= <ABSOLUTE>
| <SCALED>

<point> ::= <vdc value> (2)

G.3.4 Control elements

<control element> ::= <vdc precision>
| <AUXILIARY COLOUR>
<colour>
| <TRANSPARENCY>
<on-off indicator enumerated>
| <CLIP RECTANGLE>
<point>(2)
| <CLIP INDICATOR>
<on-off indicator enumerated>

<on-off indicator enumerated> ::= <ON>
| <OFF>

<colour> ::= <colour index>
|<red green blue>

<vdc precision> ::= <VDC INTEGER PRECISION>
<vdc integer precision value>
| <VDC REAL PRECISION>
<vdc real precision value>
{these elements have encoding}
{dependent parameters}

G.3.5 Graphical elements

<graphical element> ::= <polypoint element>
| <text element>
| <cell element>
| <gdp element>
| <rectangle element>
| <circular element>
| <elliptical element>

<polypoint element> ::= <POLYLINE>
<point pair>
<point list>
| <DISJOINT POLYLINE>
<point pair>
<point pair list>
| <POLYMARKER>
<point>
<point list>
| <POLYGON>

```
                                <point>(3)
                                <point list>
                            | <POLYGON SET>
                                <point edge pair>(3)
                                <point edge pair list>

<point list>            ::= <point>*

<point pair list>       ::= <point pair>*

<point pair>            ::= <point>(2)

<point edge pair>       ::= <point><edge out flag>

<point edge pair list>  ::= <point edge pair>*

<edge out flag>         ::= <INVISIBLE>
                          | <VISIBLE>
                          | <CLOSE INVISIBLE>
                          | <CLOSE VISIBLE>

<text element>          ::= <TEXT>
                                <point>
                                <text tail>
                          | <restricted text element>

<restricted text element> ::= <RESTRICTED TEXT>
                                <extent>
                                <point>
                                <text tail>

<extent>                ::= <vdc value>(2)

<text tail>             ::= <final character list>
                          | <nonfinal character list>

<final character list>  ::= <FINAL>
                                <string>

<nonfinal character list> ::= <NOT FINAL>
                                <string>
                                <partial text attribute element>*
                                <spanned text>

<spanned text>          ::= <APPEND TEXT>
                                <text tail>

<cell element>          ::= <CELL ARRAY>
                                <point>(3)
                                <integer>(2)
                                <local colour precision>
                                <colour>(integer1 x integer2)

                            {this element has an encoding}
                            {dependent parameter}

<local colour precision> ::= <colour precision value>
                           | <colour index precision value>
                           | <default colour precision indicator>

<gdp element>           ::= <GDP>
```

<gdp identifier>
<point list>
<data record>

<gdp identifier> ::= <integer>

<rectangle element> ::= <RECTANGLE>
<point pair>

<circular element> ::= <CIRCLE>
<point>
<radius>
| <CIRCULAR ARC 3 POINT>
<point>(3)
| <CIRCULAR ARC 3 POINT CLOSE>
<point>(3)
<close type>
| <CIRCULAR ARC CENTRE>
<point>
<vdc value>(4)
<radius>
| <CIRCULAR ARC CENTRE CLOSE>
<point>
<vdc value>(4)
<radius>
<close type>

<radius> ::= <non-negative vdc value>

<non-negative vdc value> ::= <vdc value> {greater than or equal to 0}

<close type> ::= <PIE>
| <CHORD>

<elliptical element> ::= <ELLIPSE>
<point>(3)
| <ELLIPTICAL ARC>
<point>(3)
<vdc value>(4)
| <ELLIPTICAL ARC CLOSE>
<point>(3)
<vdc value>(4)
<close type>

G.3.6 Attribute elements

<primitive attribute element> ::= <line attribute element>
| <marker attribute element>
| <text attribute element>
| <filled-area attribute element>
| <aspect source flags>

<line attribute element> ::= <LINE BUNDLE INDEX>
<positive index>
| <LINE TYPE>
<index>
| <LINE WIDTH>
<size value>
| <LINE COLOUR>
<colour>

<size value> ::= <non-negative vdc value>
| <non-negative real>

<non-negative real> ::= <real> {greater than or equal to 0}

<marker attribute element> ::= <MARKER BUNDLE INDEX>
<positive index>
| <MARKER TYPE>
<index>
| <MARKER SIZE>
<size value>
| <MARKER COLOUR>
<colour>

<partial text attribute element> ::= <TEXT FONT INDEX>
<positive index>
| <TEXT PRECISION>
<text precision enumerated>
| <CHARACTER EXPANSION FACTOR>
<real>
| <CHARACTER SPACING>
<real>
| <TEXT COLOUR>
<colour>
| <CHARACTER HEIGHT>
<non-negative vdc value>
| <CHARACTER SET INDEX>
<positive index>
| <ALTERNATE CHARACTER SET INDEX>
<positive index>
| <TEXT BUNDLE INDEX>
<positive index>
| <AUXILIARY COLOUR>
<colour>
| <TRANSPARENCY>
<on-off indicator enumerated>

<text attribute element> ::= <TEXT BUNDLE INDEX>
<positive index>
| <TEXT FONT INDEX>
<positive index>
| <TEXT PRECISION>
<text precision enumerated>
| <CHARACTER EXPANSION FACTOR>
<real>
| <CHARACTER SPACING>
<real>
| <TEXT COLOUR>
<colour>
| <CHARACTER HEIGHT>
<non-negative vdc value>
| <CHARACTER ORIENTATION>
<vdc value>(4)
| <TEXT PATH>
<path enumerated>
| <TEXT ALIGNMENT>
<horizontal alignment enumerated>
<vertical alignment enumerated>
<continuous alignment value> (2)
| <CHARACTER SET INDEX>
<positive index>

| <ALTERNATE CHARACTER SET INDEX>
<positive index>

<text precision enumerated> ::= <STRING>
| <CHARACTER>
| <STROKE>

<path enumerated> ::= <RIGHT>
| <LEFT>
| <UP>
| <DOWN>

<horizontal alignment enumerated> ::= <NORMAL HORIZONTAL>
| <LEFT>
| <CENTRE>
| <RIGHT>
| <CONTINUOUS HORIZONTAL>

<vertical alignment enumerated> ::= <NORMAL VERTICAL>
| <TOP>
| <CAP>
| <HALF>
| <BASE>
| <BOTTOM>
| <CONTINUOUS VERTICAL>

<continuous alignment value> ::= <real>

<filled-area attribute element> ::= <FILL BUNDLE INDEX>
<positive index>
| <INTERIOR STYLE>
<interior style enumerated>
| <FILL COLOUR>
<colour>
| <HATCH INDEX>
<index>
| <PATTERN INDEX>
<positive index>
| <EDGE BUNDLE INDEX>
<positive index>
| <EDGE TYPE>
<index>
| <EDGE WIDTH>
<size value>
| <EDGE COLOUR>
<colour>
| <EDGE VISIBILITY>
<on-off indicator enumerated>
| <FILL REFERENCE POINT>
<point>
| <PATTERN SIZE>
<vdc value>(4)

<interior style enumerated> ::= <HOLLOW>
| <SOLID>
| <PATTERN>
| <HATCH>
| <EMPTY>

<colour table element> ::= <COLOUR TABLE>