

XML Import Definition



ROFIN-SINAR Laser GmbH • Neufeldstr. 16 • D-85232 Günding

Proprietary Information

The information in this document is the intellectual property of Rofin-Sinar Laser GmbH. No part of this document may be reproduced or used in any form or by any means, including, but not limited to, graphic, electronic or mechanical storage or retrieval systems (i. e., photocopy, recordings, tapes, or databases), without the written permission of Rofin-Sinar Laser GmbH.

Revision History

Rev.	Date	Author	Comment
1.0	28-09-00	Andrei Khruzin	Initial release
2.0	13-10-00	Andrei Khruzin	XML format
3.0	11-12-00	Andrei Khruzin	Modified XML format
4.0	06-05-03	Andrei Khruzin	Bitmap, polyline, and arc objects added
5.0	08-01-04	Peter Egger	Group and barcode objects added Hatch parameters added to several objects
6.0 preliminary	01-06-04	Peter Egger	DataMatrix, QR-Code added (preliminary description: pages 21-23, 43-46)
7.0 preliminary	10-12-04	Kitsantas Konstantinos	InfoGlyph Code added (preliminary description: pages 24-25, 47-48)
8.0	05-05-2010	Ariane Apfel	A layer table can be defined; Layer attributes : name, Z-axis height, color

Table of Contents

1. INTRODUCTION	4
2. OVERVIEW.....	4
3. SYNTAX	8
4. SEMANTICS.....	24
5. EXAMPLES	40
6. CURRENT IMPLEMENTATION	50
7. REFERENCES	50

1. Introduction

This document defines the syntax and semantics of XML drawing representing marking data in a structured text form which can be converted automatically into VLM layouts.

2. Overview

The current implementation of XML drawing supports primitive marking objects including text, logo, bitmap, polyline, barcode and circle arc as well as matrix and group objects. Matrix objects may use static or dynamic binning. Also supported is the ability to reference an entry in a database rather than specify the matrix explicitly.

A drawing to be marked may contain one or more matrix objects. Associated with a matrix object are one or more cell layouts. In case of several cell layouts, they are distinguished via bin numbers. Each cell layout can contain only primitive marking objects. Matrix object can be defined either explicitly or implicitly by a reference to an entry in the leadframe or tray database.

Drawing objects have attributes describing their placement in a container, dimensions and other properties. Objects are positioned in a container by selecting a reference point and specifying its position in the container coordinate system. Container is a notion used to describe the structure of the drawing. The top-most container is the galvo field (or marking field). The galvo field represents the entire marking area. It is not possible to mark outside the galvo field. The next container down the hierarchy is the drawing representing all marking objects. A group is a simple container for marking objects. Another important container is matrix cell containing a cell layout. The figure 1 shows object placement in a container.

Matrix object contains one or more cells arranged in a rectangular grid. Associated to each grid point is a cell reference point. It is possible to define interleaved grids as well as grids with grouping. The figures 2, 3 and 4 show a regular grid, an interleaved grid, and a grid with grouping. Note that interleave and grouping are possible in both directions, X and Y. Note also that positions of grid points in the matrix object are defined indirectly by the position of cell reference points.

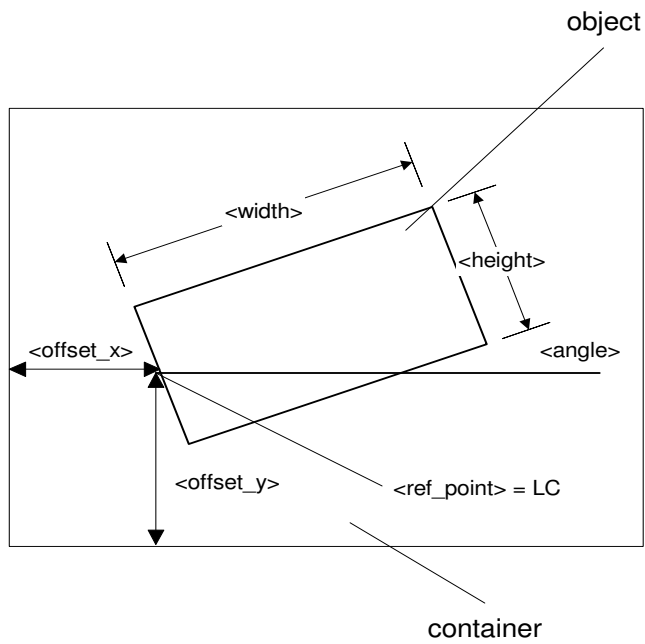


Figure 1.

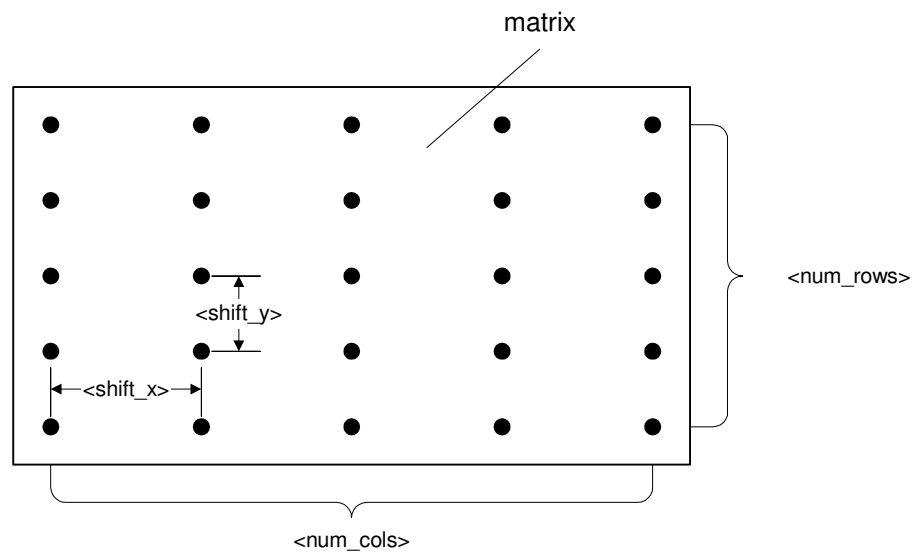


Figure 2.

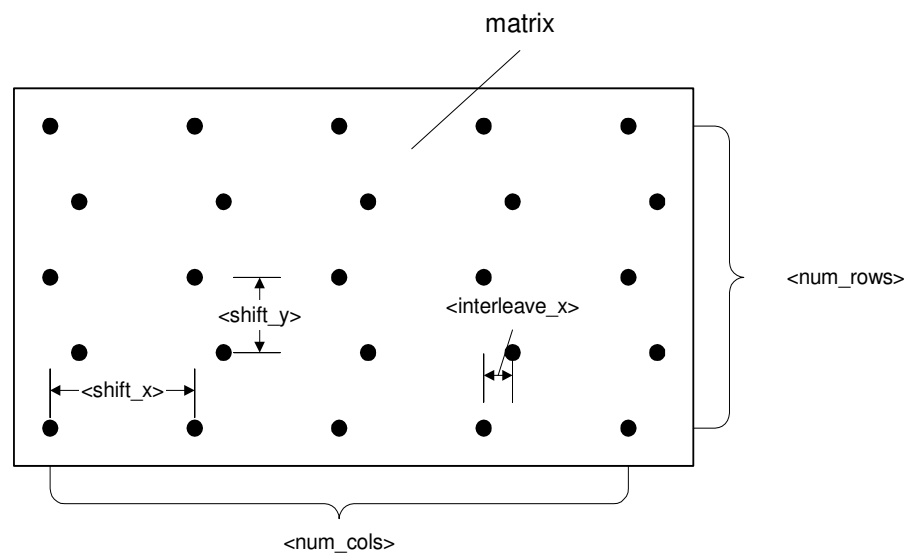


Figure 3.

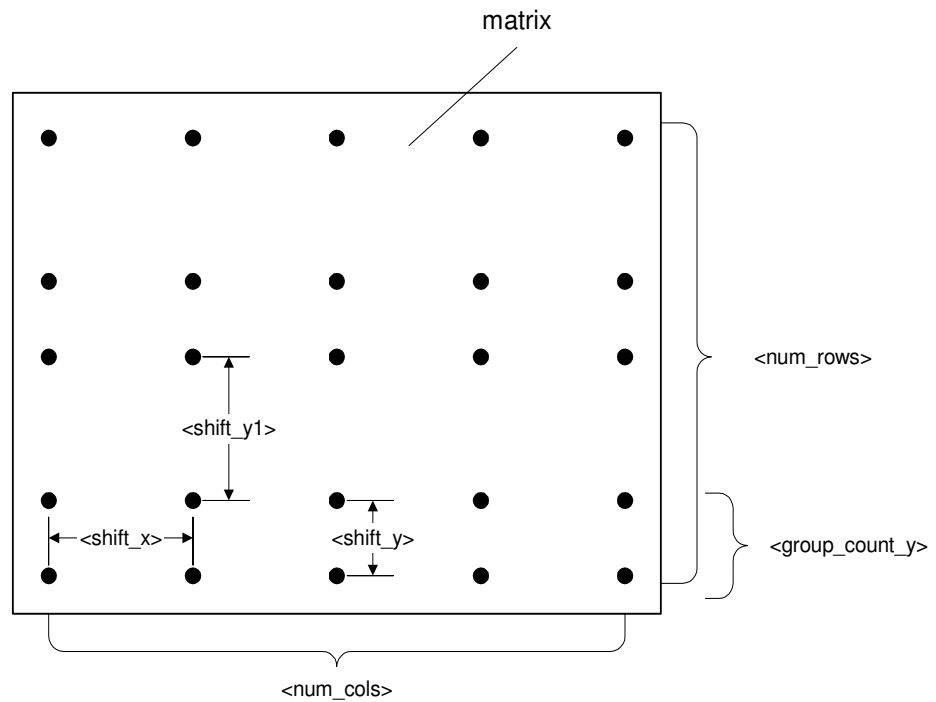


Figure 4.

Besides what is said above, container has a property worth considering here. This property, represented by REFLECT attribute, is reflection of the container content relative to either X- or Y-axis of the container coordinate system. The container position remains unchanged. Figure 5 illustrates reflection of a text container. The same applies to other containers. Note that reflection can be combined with rotation.

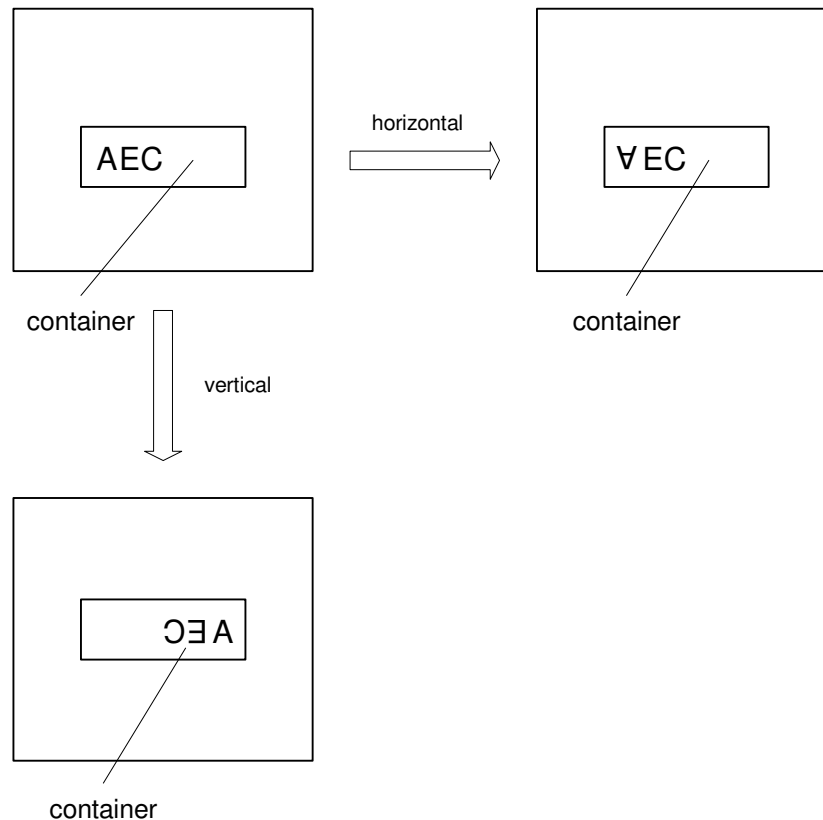


Figure 5.

3. Syntax

In the current implementation, XML drawing is a well-formed XML document whose syntax is described by a Document Type Definition .:

Drawing.dtd

```
<?xml version="1.0"?>

<!ENTITY % PRIMITIVE "TEXT | LOGO | BITMAP | POLYLINE | ARC | BARCODE |
DATAMATRIX | QRCODE | INFOGLYPH | PDF417 | BURST | TEXTRING">

<!ENTITY % FUNCTION "DXF_IMPORT">

<!ENTITY % CENTER "POINT">

<!ELEMENT DRAWING (ROOT | (SURFACE, ROOT) | (LAYERS, ROOT))>

<!ATTLIST DRAWING
    UNIT                (MM | INCH) "MM">

<!ELEMENT LAYERS (LAYER)+>
<!ELEMENT LAYER (#PCDATA) >
<!ATTLIST LAYER
    NAME                CDATA          #REQUIRED
    HEIGHT_Z_AXIS       CDATA          #IMPLIED
    COLOR               CDATA          #IMPLIED>

<!ELEMENT ROOT (GROUP | MATRIX | VLM_REFERENCE | %FUNCTION; |
%PRIMITIVE; )+>
<!ATTLIST ROOT
    ID                  CDATA          #IMPLIED
    WIDTH               CDATA          #IMPLIED
    HEIGHT              CDATA          #IMPLIED
    OFFSET_X            CDATA          #IMPLIED
    OFFSET_Y            CDATA          #IMPLIED>

<!ELEMENT VLM_REFERENCE (#PCDATA)>
<!ATTLIST VLM_REFERENCE
    ID                  CDATA          #IMPLIED
    WIDTH               CDATA          #IMPLIED
    HEIGHT              CDATA          #IMPLIED
    OFFSET_X            CDATA          #REQUIRED
    OFFSET_Y            CDATA          #REQUIRED
```


ANGLE	CDATA	#IMPLIED	
REFLECT		(N H V)	"N"
REF_POINT		(LB CB RB LC CC RC LT CT RT)	
"LB"			
LAYER	CDATA	#IMPLIED	
DESCRIPTION	CDATA	#IMPLIED	
SCALE_X	CDATA	#IMPLIED	
SCALE_Y	CDATA	#IMPLIED	
USE_BOX		(N Y)	"N"
HATCH		(N Y)	"N"
HP	CDATA	#IMPLIED	
LP	CDATA	#IMPLIED>	

<!ELEMENT GROUP (GROUP | MATRIX | VLM_REFERENCE | %PRIMITIVE;)+>

<!ATTLIST GROUP

| | | | |
|---------------------------|---------|--|------|
| ID | CDATA | #IMPLIED | |
| WIDTH | CDATA | #REQUIRED | |
| HEIGHT | CDATA | #REQUIRED | |
| OFFSET_X | CDATA | #REQUIRED | |
| OFFSET_Y | CDATA | #REQUIRED | |
| ANGLE | CDATA | #IMPLIED | |
| REFLECT | | (N H V) | "N" |
| REF_POINT | | (LB CB RB LC CC RC LT CT RT) | |
| "LB" | | | |
| LAYER | CDATA | #IMPLIED | |
| DESCRIPTION | CDATA | #IMPLIED | |
| USE_BOX | | (N Y) | "N" |
| HATCH | | (N Y) | "N" |
| HATCH_DIRECTION | | (UNI BI MEANDER ON_THE_FLY) | "BI" |
| HATCH_ON_THE_FLY_DISTANCE | CDATA | #IMPLIED | |
| HATCH_ANGLE | CDATA | #IMPLIED | |
| HATCH_NO_OUTLINE | (N Y) | | "N" |
| HATCH_INVERSE | (N Y) | | "N" |
| HATCH_TO_OUTLINE | (N Y) | | "N" |
| HATCH_OUTLINE_CORR | (N Y) | | "N" |
| HATCH_AUTO_LINE_WIDTH | (N Y) | | "N" |
| HATCH_OPTIMIZE_JUMPS | (N Y) | | "N" |
| HATCH_FROM_OUTLINE | CDATA | #IMPLIED | |

| | | |
|---------------|-------|-----------|
| HATCH_OVERLAP | CDATA | #IMPLIED |
| HP | CDATA | #IMPLIED |
| LP | CDATA | #IMPLIED> |

<!ELEMENT MATRIX ((FRAME | FRAME_REFERENCE), (DEF_CELL_LAYOUT?, CELL_LAYOUT*))>

<!ATTLIST MATRIX

| | | |
|---------|-------------|------|
| BINNING | (N S D) | "N"> |
|---------|-------------|------|

<!ELEMENT FRAME_REFERENCE EMPTY>

<!ATTLIST FRAME_REFERENCE

| | | |
|----------|-------|-----------|
| DATABASE | CDATA | #REQUIRED |
| KEY | CDATA | #REQUIRED |
| TRACK | CDATA | #IMPLIED> |

<!ELEMENT FRAME EMPTY>

<!ATTLIST FRAME

| | | |
|---------------|--|-----------|
| ID | CDATA | #IMPLIED |
| WIDTH | CDATA | #REQUIRED |
| HEIGHT | CDATA | #REQUIRED |
| OFFSET_X | CDATA | #REQUIRED |
| OFFSET_Y | CDATA | #REQUIRED |
| ANGLE | CDATA | #IMPLIED |
| REFLECT | (N H V) | "N" |
| REF_POINT | (LB CB RB LC CC RC LT CT RT) | "LB" |
| NUM_ROWS | CDATA | #REQUIRED |
| NUM_COLS | CDATA | #REQUIRED |
| SHIFT_X | CDATA | #REQUIRED |
| SHIFT_Y | CDATA | #REQUIRED |
| INTERLEAVE_X | CDATA | #IMPLIED |
| INTERLEAVE_Y | CDATA | #IMPLIED |
| GROUP_COUNT_X | CDATA | #IMPLIED |
| GROUP_COUNT_Y | CDATA | #IMPLIED |
| GROUP_SHIFT_X | CDATA | #IMPLIED |
| GROUP_SHIFT_Y | CDATA | #IMPLIED |
| DESCRIPTION | CDATA | #IMPLIED |

| | | |
|---------------|-------|-----------|
| CUSTOM | CDATA | #IMPLIED |
| LAYER | CDATA | #IMPLIED |
| CELL_WIDTH | CDATA | #REQUIRED |
| CELL_HEIGHT | CDATA | #REQUIRED |
| CELL_OFFSET_X | CDATA | #REQUIRED |
| CELL_OFFSET_Y | CDATA | #REQUIRED |
| CELL_ANGLE | CDATA | #IMPLIED> |

<!ELEMENT DEF_CELL_LAYOUT (GROUP | VLM_REFERENCE | %PRIMITIVE;)*>

<!ELEMENT CELL_LAYOUT (GROUP | VLM_REFERENCE | %PRIMITIVE;)*>

<!ATTLIST CELL_LAYOUT

| | | |
|-----|-------|------------|
| BIN | CDATA | #REQUIRED> |
|-----|-------|------------|

<!ELEMENT TEXT (#PCDATA)>

<!ATTLIST TEXT

| | | |
|--------------|--|-----------|
| ID | CDATA | #IMPLIED |
| WIDTH | CDATA | #IMPLIED |
| HEIGHT | CDATA | #IMPLIED |
| OFFSET_X | CDATA | #REQUIRED |
| OFFSET_Y | CDATA | #REQUIRED |
| ANGLE | CDATA | #IMPLIED |
| REFLECT | (N H V) | "N" |
| REF_POINT | (LB CB RB LC CC RC LT CT RT) | "LB" |
| LAYER | CDATA | #IMPLIED |
| DESCRIPTION | CDATA | #IMPLIED |
| FONT | CDATA | #REQUIRED |
| FONT_HEIGHT | CDATA | #REQUIRED |
| FONT_RATIO | CDATA | #REQUIRED |
| FONT_ITALIC | CDATA | #IMPLIED |
| CHAR_SPACING | CDATA | #IMPLIED |
| LINE_SPACING | CDATA | #IMPLIED |
| ALIGNMENT | (LB CB RB LC CC RC LT CT RT) | "LB" |
| MONOSPACE | (N Y) | "N" |
| CROSSLESS | (N Y) | "N" |
| USE_BOX | (N Y) | "N" |

| | | |
|---------------------------|-----------------------------------|-----------|
| HATCH | (N Y) | "N" |
| HATCH_DIRECTION | (UNI BI MEANDER ON_THE_FLY) | "BI" |
| HATCH_ON_THE_FLY_DISTANCE | CDATA | #IMPLIED |
| HATCH_ANGLE | CDATA | #IMPLIED |
| HATCH_NO_OUTLINE | (N Y) | "N" |
| HATCH_INVERSE | (N Y) | "N" |
| HATCH_TO_OUTLINE | (N Y) | "N" |
| HATCH_OUTLINE_CORR | (N Y) | "N" |
| HATCH_AUTO_LINE_WIDTH | (N Y) | "N" |
| HATCH_OPTIMIZE_JUMPS | (N Y) | "N" |
| HATCH_FROM_OUTLINE | CDATA | #IMPLIED |
| HATCH_OVERLAP | CDATA | #IMPLIED |
| HP | CDATA | #IMPLIED |
| LP | CDATA | #IMPLIED> |

<!ELEMENT TEXTRING (#PCDATA)>

<!ATTLIST TEXTRING

| | | |
|--------------|--|-----------|
| ID | CDATA | #IMPLIED |
| WIDTH | CDATA | #IMPLIED |
| HEIGHT | CDATA | #IMPLIED |
| OFFSET_X | CDATA | #REQUIRED |
| OFFSET_Y | CDATA | #REQUIRED |
| ANGLE | CDATA | #IMPLIED |
| REFLECT | (N H V) | "N" |
| REF_POINT | (LB CB RB LC CC RC LT CT RT) | |
| "LB" | | |
| LAYER | CDATA | #IMPLIED |
| DESCRIPTION | CDATA | #IMPLIED |
| FONT | CDATA | #REQUIRED |
| FONT_HEIGHT | CDATA | #REQUIRED |
| FONT_RATIO | CDATA | #REQUIRED |
| FONT_ITALIC | CDATA | #IMPLIED |
| CHAR_SPACING | CDATA | #IMPLIED |
| LINE_SPACING | CDATA | #IMPLIED |
| ALIGNMENT | (LB CB RB LC CC RC LT CT RT) | |
| "LB" | | |
| MONOSPACE | (N Y) | "N" |
| CROSSLESS | (N Y) | "N" |

```

USE_BOX                (N | Y)                "N"
HATCH                  (N | Y)                "N"
HATCH_DIRECTION        (UNI | BI | MEANDER | ON_THE_FLY)  "BI"
HATCH_ON_THE_FLY_DISTANCE  CDATA            #IMPLIED
HATCH_ANGLE            CDATA            #IMPLIED
HATCH_NO_OUTLINE       (N | Y)            "N"
HATCH_INVERSE          (N | Y)            "N"
HATCH_TO_OUTLINE       (N | Y)            "N"
HATCH_OUTLINE_CORR     (N | Y)            "N"
HATCH_AUTO_LINE_WIDTH  (N | Y)            "N"
HATCH_OPTIMIZE_JUMPS   (N | Y)            "N"
HATCH_FROM_OUTLINE     CDATA            #IMPLIED
HATCH_OVERLAP          CDATA            #IMPLIED
HP                     CDATA            #IMPLIED
LP                     CDATA            #IMPLIED
RADIUS      CDATA      #IMPLIED
START_ANGLE CDATA #IMPLIED
IS_CCW      CDATA      #IMPLIED
FIT_TO_ANGLE_SECTOR CDATA #IMPLIED
ANGLE_SECTOR CDATA #IMPLIED>

<!ELEMENT LOGO (#PCDATA)>
<!ATTLIST LOGO
    ID          CDATA      #IMPLIED
    WIDTH       CDATA      #IMPLIED
    HEIGHT      CDATA      #IMPLIED
    OFFSET_X    CDATA      #REQUIRED
    OFFSET_Y    CDATA      #REQUIRED
    ANGLE        CDATA      #IMPLIED
    REFLECT      (N | H | V) "N"
    REF_POINT    (LB | CB | RB | LC | CC | RC | LT | CT | RT)
    "LB"
    LAYER        CDATA      #IMPLIED
    DESCRIPTION  CDATA      #IMPLIED
    SCALE_X      CDATA      #IMPLIED
    SCALE_Y      CDATA      #IMPLIED
    USE_BOX      (N | Y)      "N"
    HATCH        (N | Y)      "N"

```

| | | |
|---------------------------|-----------------------------------|-----------|
| HATCH_DIRECTION | (UNI BI MEANDER ON_THE_FLY) | "BI" |
| HATCH_ON_THE_FLY_DISTANCE | CDATA | #IMPLIED |
| HATCH_ANGLE | CDATA | #IMPLIED |
| HATCH_NO_OUTLINE | (N Y) | "N" |
| HATCH_INVERSE | (N Y) | "N" |
| HATCH_TO_OUTLINE | (N Y) | "N" |
| HATCH_OUTLINE_CORR | (N Y) | "N" |
| HATCH_AUTO_LINE_WIDTH | (N Y) | "N" |
| HATCH_OPTIMIZE_JUMPS | (N Y) | "N" |
| HATCH_FROM_OUTLINE | CDATA | #IMPLIED |
| HATCH_OVERLAP | CDATA | #IMPLIED |
| HP | CDATA | #IMPLIED |
| LP | CDATA | #IMPLIED> |

<!ELEMENT BITMAP (#PCDATA)>

<!ATTLIST BITMAP

| | | |
|-------------|--|-----------|
| ID | CDATA | #IMPLIED |
| WIDTH | CDATA | #IMPLIED |
| HEIGHT | CDATA | #IMPLIED |
| OFFSET_X | CDATA | #REQUIRED |
| OFFSET_Y | CDATA | #REQUIRED |
| ANGLE | CDATA | #IMPLIED |
| REFLECT | (N H V) | "N" |
| REF_POINT | (LB CB RB LC CC RC LT CT RT) | |
| "LB" | | |
| LAYER | CDATA | #IMPLIED |
| DESCRIPTION | CDATA | #IMPLIED |
| DENSITY_X | CDATA | #IMPLIED |
| DENSITY_Y | CDATA | #IMPLIED |
| USE_BOX | (N Y) | "N" |
| INVERTED | (N Y) | "N" |
| LP | CDATA | #IMPLIED> |

<!ELEMENT POLYLINE ((POINT, POINT+) | (POLYPOINT, POLYPOINT+))>

<!ATTLIST POLYLINE

| | | |
|--------|-------|----------|
| ID | CDATA | #IMPLIED |
| WIDTH | CDATA | #IMPLIED |
| HEIGHT | CDATA | #IMPLIED |

| | | | |
|---------------------------|-------|--|------|
| OFFSET_X | CDATA | #IMPLIED | |
| OFFSET_Y | CDATA | #IMPLIED | |
| ANGLE | CDATA | #IMPLIED | |
| REFLECT | | (N H V) | "N" |
| REF_POINT | | (LB CB RB LC CC RC LT CT RT) | |
| "LB" | | | |
| LAYER | CDATA | #IMPLIED | |
| DESCRIPTION | CDATA | #IMPLIED | |
| USE_BOX | | (N Y) | "N" |
| HATCH | | (N Y) | "N" |
| HATCH_DIRECTION | | (UNI BI MEANDER ON_THE_FLY) | "BI" |
| HATCH_ON_THE_FLY_DISTANCE | CDATA | #IMPLIED | |
| HATCH_ANGLE | CDATA | #IMPLIED | |
| HATCH_NO_OUTLINE | | (N Y) | "N" |
| HATCH_INVERSE | | (N Y) | "N" |
| HATCH_TO_OUTLINE | | (N Y) | "N" |
| HATCH_OUTLINE_CORR | | (N Y) | "N" |
| HATCH_AUTO_LINE_WIDTH | | (N Y) | "N" |
| HATCH_OPTIMIZE_JUMPS | | (N Y) | "N" |
| HATCH_FROM_OUTLINE | CDATA | #IMPLIED | |
| HATCH_OVERLAP | CDATA | #IMPLIED | |
| HP | CDATA | #IMPLIED | |
| LP | CDATA | #IMPLIED> | |

<!ELEMENT ARC (POINT, POINT, %CENTER;, DIRECTION)>

<!ATTLIST ARC

| | | | |
|-------------|-------|--|-----|
| ID | CDATA | #IMPLIED | |
| WIDTH | CDATA | #IMPLIED | |
| HEIGHT | CDATA | #IMPLIED | |
| OFFSET_X | CDATA | #IMPLIED | |
| OFFSET_Y | CDATA | #IMPLIED | |
| ANGLE | CDATA | #IMPLIED | |
| REFLECT | | (N H V) | "N" |
| REF_POINT | | (LB CB RB LC CC RC LT CT RT) | |
| "LB" | | | |
| LAYER | CDATA | #IMPLIED | |
| DESCRIPTION | CDATA | #IMPLIED | |
| USE_BOX | | (N Y) | "N" |

| | | |
|---------------------------|-----------------------------------|-----------|
| HATCH | (N Y) | "N" |
| HATCH_DIRECTION | (UNI BI MEANDER ON_THE_FLY) | "BI" |
| HATCH_ON_THE_FLY_DISTANCE | CDATA | #IMPLIED |
| HATCH_ANGLE | CDATA | #IMPLIED |
| HATCH_NO_OUTLINE | (N Y) | "N" |
| HATCH_INVERSE | (N Y) | "N" |
| HATCH_TO_OUTLINE | (N Y) | "N" |
| HATCH_OUTLINE_CORR | (N Y) | "N" |
| HATCH_AUTO_LINE_WIDTH | (N Y) | "N" |
| HATCH_OPTIMIZE_JUMPS | (N Y) | "N" |
| HATCH_FROM_OUTLINE | CDATA | #IMPLIED |
| HATCH_OVERLAP | CDATA | #IMPLIED |
| HP | CDATA | #IMPLIED |
| LP | CDATA | #IMPLIED> |

<!ELEMENT BARCODE (#PCDATA)>

<!ATTLIST BARCODE

| | | |
|-----------------------|---|--------------|
| ID | CDATA | #IMPLIED |
| WIDTH | CDATA | #IMPLIED |
| HEIGHT | CDATA | #IMPLIED |
| OFFSET_X | CDATA | #REQUIRED |
| OFFSET_Y | CDATA | #REQUIRED |
| ANGLE | CDATA | #IMPLIED |
| REFLECT | (N H V) | "N" |
| REF_POINT | (LB CB RB LC CC RC LT CT RT) | "LB" |
| LAYER | CDATA | #IMPLIED |
| DESCRIPTION | CDATA | #IMPLIED |
| USE_BOX | (N Y) | "N" |
| BARCODE_TYPE | (CODE39_EXT CODE39 CODE93 CODE128 EAN_UCC128 2_OF_5 UPC-A UPC-E EAN-8 EAN-13 BOOKLAN CODABAR BC412) | "CODE39_EXT" |
| BARCODE_ELEMENT_WIDTH | CDATA | #REQUIRED |
| BARCODE_ELEMENT_RATIO | CDATA | #IMPLIED |
| BARCODE_QUIET_ZONE | (N Y) | "N" |
| BARCODE_CHECK_DIGIT | (N Y) | "N" |
| BARCODE_INVERT | (N Y) | "N" |
| BARCODE_KEEP_SIZE | (BOX ELEMENT) | "BOX" |

| | | |
|---------------------------|-----------------------------------|-----------|
| HATCH | (Y) | "Y" |
| HATCH_DIRECTION | (UNI BI MEANDER ON_THE_FLY) | "BI" |
| HATCH_ON_THE_FLY_DISTANCE | CDATA | #IMPLIED |
| HATCH_ANGLE | CDATA | #IMPLIED |
| HATCH_NO_OUTLINE | (N Y) | "Y" |
| HATCH_INVERSE | (N Y) | "N" |
| HATCH_TO_OUTLINE | (N Y) | "Y" |
| HATCH_OUTLINE_CORR | (N Y) | "Y" |
| HATCH_AUTO_LINE_WIDTH | (N Y) | "Y" |
| HATCH_OPTIMIZE_JUMPS | (N Y) | "N" |
| HATCH_FROM_OUTLINE | CDATA | #IMPLIED |
| HATCH_OVERLAP | CDATA | #IMPLIED |
| HP | CDATA | #IMPLIED |
| LP | CDATA | #IMPLIED> |

<!ELEMENT DATAMATRIX (#PCDATA)>

<!ATTLIST DATAMATRIX

| | | |
|-------------------------|--|-----------------|
| ID | CDATA | #IMPLIED |
| WIDTH | CDATA | #IMPLIED |
| HEIGHT | CDATA | #IMPLIED |
| OFFSET_X | CDATA | #REQUIRED |
| OFFSET_Y | CDATA | #REQUIRED |
| ANGLE | CDATA | #IMPLIED |
| REFLECT | (N H V) | "N" |
| REF_POINT | (LB CB RB LC CC RC LT CT RT) | "LB" |
| LAYER | CDATA | #IMPLIED |
| DESCRIPTION | CDATA | #IMPLIED |
| USE_BOX | (N Y) | "N" |
| DATAMATRIX_TYPE | (ECC00 ECC50 ECC80 ECC100 ECC130 ECC140 ECC200_SQUARE ECC200_RECT) | "ECC200_SQUARE" |
| DATAMATRIX_CELL_SIZE | CDATA | #IMPLIED |
| DATAMATRIX_FORMAT_ID | (NONE NUM_500 ALPHA_500 ALPHA_PUNC_500 ALPHA_NUM_500) | "NONE" |
| DATAMATRIX_CELL_DENSITY | (MINIMUM 10X10 12X12 14X14 16X16 18X18 20X20 22X22 24X24 26X26 32X32 36X36 40X40 44X44 48X48 52X52 64X64 72X72 88X88 96X96 102X102 132X132 144X144 8X18 8X32 12X26 12X36 16X36 16X48 9X9 11X11 13X13 15X15 17X17 19X19 21X21 23X23 25X25 27X27 29X29 | |

31X31 | 33X33 | 35X35 | 37X37 | 39X39 | 41X41 | 43X43 | 45X45 | 47X47 |
49X49) "MINIMUM"

| | | |
|---------------------------|-----------------------------------|-----------|
| DATAMATRIX_BURST_MODE | (N Y) | "N" |
| DATAMATRIX_KEEP_SIZE | (BOX CELL) | "BOX" |
| HATCH | (Y) | "Y" |
| HATCH_DIRECTION | (UNI BI MEANDER ON_THE_FLY) | "BI" |
| HATCH_ON_THE_FLY_DISTANCE | CDATA | #IMPLIED |
| HATCH_ANGLE | CDATA | #IMPLIED |
| HATCH_NO_OUTLINE | (N Y) | "Y" |
| HATCH_INVERSE | (N Y) | "N" |
| HATCH_TO_OUTLINE | (N Y) | "Y" |
| HATCH_OUTLINE_CORR | (N Y) | "Y" |
| HATCH_AUTO_LINE_WIDTH | (N Y) | "Y" |
| HATCH_OPTIMIZE_JUMPS | (N Y) | "N" |
| HATCH_FROM_OUTLINE | CDATA | #IMPLIED |
| HATCH_OVERLAP | CDATA | #IMPLIED |
| HP | CDATA | #IMPLIED |
| LP | CDATA | #IMPLIED> |

<!ELEMENT QRCODE (#PCDATA)>

<!ATTLIST QRCODE

| | | |
|--------------------|--|-----------|
| ID | CDATA | #IMPLIED |
| WIDTH | CDATA | #IMPLIED |
| HEIGHT | CDATA | #IMPLIED |
| OFFSET_X | CDATA | #REQUIRED |
| OFFSET_Y | CDATA | #REQUIRED |
| ANGLE | CDATA | #IMPLIED |
| REFLECT | (N H V) | "N" |
| REF_POINT | (LB CB RB LC CC RC LT CT RT) | |
| "LB" | | |
| LAYER | CDATA | #IMPLIED |
| DESCRIPTION | CDATA | #IMPLIED |
| USE_BOX | (N Y) | "N" |
| QRCODE_MODEL | (1 2 MICRO_QR) | "2" |
| QRCODE_ECC_LEVEL | (L M Q H) | "M" |
| QRCODE_CELL_PITCH | CDATA | #IMPLIED |
| QRCODE_CELL_ADJUST | CDATA | #IMPLIED |
| QRCODE_QUIET_ZONE | CDATA | #IMPLIED |

```

QRCODE_BURST_MODE (N | Y)          "N"
QRCODE_KEEP_SIZE  (CELL)           "CELL"
HATCH              (Y)              "Y"
HATCH_DIRECTION    (UNI | BI | MEANDER | ON_THE_FLY)  "BI"
HATCH_ON_THE_FLY_DISTANCE  CDATA      #IMPLIED
HATCH_ANGLE        CDATA            #IMPLIED
HATCH_NO_OUTLINE   (N | Y)          "Y"
HATCH_INVERSE       (N | Y)          "N"
HATCH_TO_OUTLINE   (N | Y)          "Y"
HATCH_OUTLINE_CORR (N | Y)          "Y"
HATCH_AUTO_LINE_WIDTH (N | Y)       "Y"
HATCH_OPTIMIZE_JUMPS (N | Y)        "N"
HATCH_FROM_OUTLINE CDATA            #IMPLIED
HATCH_OVERLAP      CDATA            #IMPLIED
HP                 CDATA            #IMPLIED
LP                 CDATA            #IMPLIED>

<!ELEMENT INFOGLYPH (#PCDATA)>
<!ATTLIST INFOGLYPH
    ID                CDATA            #IMPLIED
    WIDTH             CDATA            #IMPLIED
    HEIGHT            CDATA            #IMPLIED
    OFFSET_X          CDATA            #REQUIRED
    OFFSET_Y          CDATA            #REQUIRED
    ANGLE             CDATA            #IMPLIED
    REFLECT            ( N | H | V )    "N"
    REF_POINT         ( LB | CB | RB | LC | CC | RC | LT | CT | RT )
    "LB"
    LAYER             CDATA            #IMPLIED
    DESCRIPTION        CDATA            #IMPLIED
    USE_BOX            ( N | Y )        "N"

    INFOGLYPH_BLOCKDIM ( 2X2 | 2X3 | 2X4 | 2X5 |
                        3X2 | 3X3 | 3X4 | 3X5 | 3X6 | 3X7
| 3X8 | 3X9 |
                        4X2 | 4X3 | 4X4 | 4X5 | 4X6 | 4X7
|
                        5X2 | 5X3 | 5X4 | 5X5 |

```

```

        6X3 | 6X4 |
        7X3 | 7X4 |
        8X3 |
        9X3 )          "3X3"

INFOGLYPH_ERRORCOR ( ERRORCORRECTION_20 |
                    ERRORCORRECTION_30 |
                    ERRORCORRECTION_40 |
                    ERRORCORRECTION_50 |
                    ERRORCORRECTION_60 |
                    ERRORCORRECTION_70 |
                    ERRORCORRECTION_80 |
                    ERRORCORRECTION_90 )

"ERRORCORRECTION_50"

INFOGLYPH_CODESET ( CODESET_NOMAPPING |
                    CODESET_8BIT |
                    CODESET_7BIT_ASCII_US |
                    CODESET_7BIT_ASCII_GER |
                    CODESET_6BIT_UPPERLOWER_DOT |
                    CODESET_6BIT_UPPERLOWER_SLASH |
                    CODESET_6BIT_UPPER_SPECIAL |
                    CODESET_6BIT_LOWER_SPECIAL |
                    CODESET_5BIT_UPPER |
                    CODESET_5BIT_LOWER |
                    CODESET_4BIT_NUMBERS )
"CODESET_6BIT_UPPERLOWER_DOT"

INFOGLYPH_BURST_MODE ( N | Y )          "N"

INFOGLYPH_KEEP_SIZE      ( BOX | CELL )    "BOX"


HATCH                      ( Y )          "Y"

HATCH_DIRECTION          ( UNI | BI | MEANDER |
ON_THE_FLY )          "BI"

HATCH_ON_THE_FLY_DISTANCE  CDATE          #IMPLIED

HATCH_ANGLE              CDATE          #IMPLIED

HATCH_NO_OUTLINE          ( N | Y )          "Y"

HATCH_INVERSE              ( N | Y )          "N"

HATCH_TO_OUTLINE          ( N | Y )          "Y"

HATCH_OUTLINE_CORR        ( N | Y )          "Y"

HATCH_AUTO_LINE_WIDTH     ( N | Y )          "Y"

```

| | | |
|----------------------|-----------|------------|
| HATCH_OPTIMIZE_JUMPS | (N Y) | "N" |
| HATCH_FROM_OUTLINE | CDATA | #IMPLIED |
| HATCH_OVERLAP | CDATA | #IMPLIED |
| HP | CDATA | #IMPLIED |
| LP | CDATA | #IMPLIED > |

<!ELEMENT PDF417 (#PCDATA)>

<!ATTLIST PDF417

| | | |
|---------------------------|--|-----------|
| ID | CDATA | #IMPLIED |
| WIDTH | CDATA | #IMPLIED |
| HEIGHT | CDATA | #IMPLIED |
| OFFSET_X | CDATA | #REQUIRED |
| OFFSET_Y | CDATA | #REQUIRED |
| ANGLE | CDATA | #IMPLIED |
| REFLECT | (N H V) | "N" |
| REF_POINT | (LB CB RB LC CC RC LT CT RT) | "LB" |
| LAYER | CDATA | #IMPLIED |
| DESCRIPTION | CDATA | #IMPLIED |
| USE_BOX | (N Y) | "N" |
| PDF417_MODULE_WIDTH | CDATA | #REQUIRED |
| PDF417_MODULE_HEIGHT | CDATA | #IMPLIED |
| PDF417_QUIET_ZONE | (N Y) | "N" |
| PDF417_TRUNCATED | (N Y) | "N" |
| PDF417_INVERT | (N Y) | "N" |
| PDF417_COMPRESSION | (AUTO_EXC_BIN AUTO_EXC_BIN_NUM BIN | |
| EXC NUM) | "AUTO_EXC_BIN" | |
| PDF417_ERR_CORR_LEVEL | (AUTO 0 1 2 3 4 5 6 7 8) | "AUTO" |
| PDF417_ERR_CORR_OVERHEAD | CDATA | #IMPLIED |
| PDF417_SIZE_MAX_ROWS | CDATA | #IMPLIED |
| PDF417_SIZE_MAX_COLS | CDATA | #IMPLIED |
| PDF417_KEEP_SIZE | (BOX ELEMENT) | "BOX" |
| HATCH | (Y) | "Y" |
| HATCH_DIRECTION | (UNI BI MEANDER ON_THE_FLY) | "BI" |
| HATCH_ON_THE_FLY_DISTANCE | CDATA | #IMPLIED |
| HATCH_ANGLE | CDATA | #IMPLIED |
| HATCH_NO_OUTLINE | (N Y) | "Y" |

| | | |
|-----------------------|---------|-----------|
| HATCH_INVERSE | (N Y) | "N" |
| HATCH_TO_OUTLINE | (N Y) | "Y" |
| HATCH_OUTLINE_CORR | (N Y) | "Y" |
| HATCH_AUTO_LINE_WIDTH | (N Y) | "Y" |
| HATCH_OPTIMIZE_JUMPS | (N Y) | "N" |
| HATCH_FROM_OUTLINE | CDATA | #IMPLIED |
| HATCH_OVERLAP | CDATA | #IMPLIED |
| HP | CDATA | #IMPLIED |
| LP | CDATA | #IMPLIED> |

<!ELEMENT BURST (BURST_DOT, BURST_DOT+)>

<!ATTLIST BURST

| | | |
|-------------|-------|-----------|
| ID | CDATA | #IMPLIED |
| LAYER | CDATA | #IMPLIED |
| DESCRIPTION | CDATA | #IMPLIED |
| LP | CDATA | #IMPLIED> |

<!ELEMENT POINT (#PCDATA)>

<!ELEMENT POLYPOINT (#PCDATA)>

<!ATTLIST POLYPOINT

| | | |
|------|--------------|---------|
| TYPE | (LINE ARC) | "LINE"> |
|------|--------------|---------|

<!ELEMENT DIRECTION (#PCDATA)>

<!ELEMENT BURST_PARAMID (#PCDATA)>

<!ELEMENT BURST_DOT (POINT, BURST_PARAMID)>

<!ELEMENT DXF_IMPORT (#PCDATA)>

<!ATTLIST DXF_IMPORT

| | | |
|-----------|--|----------|
| ID | CDATA | #IMPLIED |
| WIDTH | CDATA | #IMPLIED |
| HEIGHT | CDATA | #IMPLIED |
| OFFSET_X | CDATA | #IMPLIED |
| OFFSET_Y | CDATA | #IMPLIED |
| ANGLE | CDATA | #IMPLIED |
| REFLECT | (N H V) | "N" |
| REF_POINT | (LB CB RB LC CC RC LT CT RT) | "LB" |

| | | | |
|--------------------|---------|-----------|--|
| LAYER | CDATA | #IMPLIED | |
| DESCRIPTION | CDATA | #IMPLIED | |
| SPLINE_AS_POLYLINE | (N Y) | "Y" | |
| ASPECT_RATIO | (N Y) | "Y" | |
| TOLERANCE | CDATA | #IMPLIED | |
| HATCH | (N Y) | "N" | |
| HP | CDATA | #IMPLIED | |
| LP | CDATA | #IMPLIED> | |

<!ELEMENT SURFACE (CYLINDER | PLANE)>

<!ATTLIST SURFACE

| | | |
|------------------------|---------|------|
| USE_SURFACE_CORRECTION | (N Y) | "N" |
| USE_Z_CORRECTION | (N Y) | "N"> |

<!ELEMENT CYLINDER (AXIS, ORIGIN)>

<!ATTLIST CYLINDER

| | | |
|-------------|---------|-----------|
| RADIUS | CDATA | #REQUIRED |
| FOCUS_DEPTH | CDATA | #IMPLIED |
| CONVEX | (Y N) | "Y" |
| HEAD | (1 2) | "1"> |

<!ELEMENT AXIS (#PCDATA)>

<!ELEMENT ORIGIN (#PCDATA)>

<!ELEMENT PLANE (NORMAL, ORIGIN)>

<!ATTLIST PLANE

| | | |
|-------------|---------|----------|
| FOCUS_DEPTH | CDATA | #IMPLIED |
| HEAD | (1 2) | "1"> |

<!ELEMENT NORMAL (#PCDATA)>

4. Semantics

The root element of the XML drawing is called DRAWING. It has one attribute, UNIT, with the meaning:

- UNIT sets unit of measurement (mm or inch).

The element ROOT represents a drawing (=collection of objects to be marked). The drawing reference point is always its left bottom corner. The attributes have the following meaning:

- ID is a name of file to save the generated VLM layout in,
- WIDTH is the width of the whole drawing,
- HEIGHT is the height of the whole drawing,
- OFFSET_X is the x-coordinate of the drawing reference point in the galvo field,
- OFFSET_Y is the y-coordinate of the drawing reference point in the galvo field.

The element GROUP represents a group object. A group object is used as a container for other marking elements or groups. Each group has its own coordinate system. So the marking object positions of the included objects must be referenced to the left bottom corner of the group element, not to the reference point of the drawing!

The attributes have the following meaning:

- ID is the name of the group element,
- WIDTH is the width of the group,
- HEIGHT is the height of the group,
- OFFSET_X is the X-coordinate of the group reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the group reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the group is rotated (counterclockwise if positive) about its reference point in the container coordinate system,

- REFLECT defines whether the contents of the group is reflected horizontally (value "H"), vertically (value "V"), or not reflected (value "N"),
- REF_POINT defines the group reference point, allowed values being left bottom "LB", center bottom "CB", right bottom "RB", left center "LC", center center "CC", right center "RC", left top "LT", center top "CT" and right top "RT",
- LAYER is the drawing layer to place the group at, (not implemented yet!)
- DESCRIPTION is a text string associated with the group element which may be used at run-time to control the marking process,
- USE_BOX sets / resets the property "Preserve aspect ration" of the group element,
- HATCH defines whether the group is hatched,
- HATCH_DIRECTION defines the hatch direction of the group element. For group elements only the directions "UNI", unidirectional and "BI", bidirectional are available. Default value is "BI"
- HATCH_ON_THE_FLY_DISTANCE not available for group element,
- HATCH_ANGLE is the angle in degrees for the hatch lines,
- HATCH_NO_OUTLINE defines if the outline of the hatched object is marked or not,
- HATCH_INVERSE inverts the hatching of the object,
- HATCH_TO_OUTLINE not available for group element,
- HATCH_OUTLINE_CORR not supported by current hatch algorithm,
- HATCH_AUTO_LINE_WIDTH not available for group element,
- HATCH_OPTIMIZE_JUMPS optimizes the galvo jumps for hatch lines,
- HATCH_FROM_OUTLINE not supported by current hatch algorithm,
- HATCH_OVERLAP line distance for hatch lines in percent. A negative value will increase the distance between the hatch lines, positive values will decrease the distance between the hatch lines,

The element MATRIX represents a matrix object. It has one attribute with the meaning:

- BINNING defines whether the matrix is capable of binning or not, the value “N” designating no binning, “S” static binning, “D” dynamic binning.

The element FRAME represents matrix frame. It has the following attributes (see Figure 2):

- ID is a name of the matrix object,
- WIDTH is the width of the matrix,
- HEIGHT is the height of the matrix,
- OFFSET_X is the X-coordinate of the matrix reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the matrix reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the matrix is rotated (counterclockwise if positive) about its reference point in the container coordinate system,
- REFLECT defines whether the contents of the matrix is reflected horizontally (value “H”), vertically (value “V”), or not reflected (value “N”),
- REF_POINT defines the matrix reference point, allowed values being left bottom “LB”, center bottom “CB”, right bottom “RB”, left center “LC”, center center “CC”, right center “RC”, left top “LT”, center top “CT”, and right top “RT”,
- NUM_ROWS is the number of rows in the matrix,
- NUM_COLS is the number of columns in the matrix,
- SHIFT_X is the distance between reference points of two consecutive cells along the x-axis,
- SHIFT_Y is the distance between reference points of two consecutive cells along the y-axis,
- INTERLEAVE_X is the additional shift along the y-axis applied to every even cell in the matrix (see Figure 3),
- INTERLEAVE_Y is the additional shift along the x-axis applied to every even cell in the matrix (see Figure 3),

- GROUP_COUNT_X is the number of cells in a group along the x-axis (see Figure 4),
- GROUP_COUNT_Y is the number of cells in a group along the y-axis (see Figure 4),
- GROUP_SHIFT_X is the distance along the x-axis between two consecutive groups measured from the reference point of the last cell in one group to that of the first cell in the next group (see Figure 4),
- GROUP_SHIFT_Y is the distance along the y-axis between two consecutive groups measured from the reference point of the last cell in one group to that of the first cell in the next group (see Figure 4),
- DESCRIPTION is a text string associated with the matrix which can be used at run-time to control the marking process,
- CUSTOM is a text string with the purpose similar to that of DESCRIPTION,
- LAYER is the drawing layer to place the matrix at, (not implemented yet!)
- CELL_WIDTH is the width of the matrix cell,
- CELL_HEIGHT is the height of the matrix cell,
- CELL_OFFSET_X is the X-coordinate of the cell reference point in the matrix coordinate system,
- CELL_OFFSET_Y is the Y-coordinate of the cell reference point in the matrix coordinate system,
- CELL_ANGLE is the angle in degrees by which each cell is rotated (counterclockwise if positive) about its reference point in the matrix coordinate system.

Note that cell reference point is its center and cannot be changed.

The element FRAME_REFERENCE represents a reference to a matrix frame in either the leadframe or tray database. It has the following attributes:

- DATABASE is the name of the database to use,
- KEY is the name of the key to use,
- TRACK is the symbolic name of the track to put the matrix at.

Any key which is unique within the database can be used to select an entry. Current implementation uses key VARIATION to make a selection from both the leadframe and tray databases. The TRACK attribute is used to specify symbolically offsets for matrix (or matrices in case of a dual-track system) and takes the values of the form TRACK1, TRACK2, and so on. When a VLM layout is generated, TRACK value is transformed into the matrix ID so that the matrix at TRACK1 gets the name MATRIX1, at TRACK2 the name MATRIX2, and so on. The value of KEY attribute is transformed into the DESCRIPTION property of the matrix object. This information can be used at run-time to position matrices correctly.

The element DEF_CELL_LAYOUT represents a collection of primitive marking objects used as matrix cell layout. It has no attribute. Currently DEF_CELL_LAYOUT may contain either text or logo objects only. The DEF_CELL_LAYOUT element can be used in two cases, a matrix with no binning or a matrix with dynamic binning. For static binning, CELL_LAYOUT element should be used instead which has one attribute:

- BIN is the bin number.

CELL_LAYOUT may contain either text or logo objects only.

The element TEXT represents a text object. Its content is the text itself. Its attributes have the meaning:

- ID is the name of the element,
- WIDTH is the width of the text box,
- HEIGHT is the height of the text box,
- OFFSET_X is the X-coordinate of the text box reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the text box reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the text box is rotated (counterclockwise if positive) about its reference point in the container coordinate system,
- REFLECT defines whether the contents of the text box is reflected horizontally (value "H"), vertically (value "V"), or not reflected (value "N"),
- REF_POINT defines the text box reference point, allowed values being left bottom "LB", center bottom "CB", right bottom "RB", left center "LC", center center "CC", right center "RC", left top "LT", center top "CT", and right top "RT",
- LAYER is the drawing layer to place the text element at, (not implemented yet!)
- DESCRIPTION is a text string associated with the text element which may be used at run-time to control the marking process,
- FONT is the name of a font used,
- FONT_HEIGHT is the desired height of the characters,
- FONT_RATIO is the ration of character width to its height,
- CHAR_SPACING is the distance between bounding boxes of adjacent characters,
- LINE_SPACING is the distance between base lines of consecutive lines of text,
- ALIGNMENT defines text alignment in the text box. It has no effect if USE_BOX is set to "N",
- MONOSPACE defines whether the font used should be mono spaced,
- CROSSLESS defines if the text is marked cross less, (available for Rofin fonts "*.fnt" only!)
- USE_BOX defines whether the bounding box should be used to bound the text,

- HATCH defines whether the characters are hatched (available for true type fonts only),
- HATCH_DIRECTION defines the hatch direction of the text element. For text elements only the directions “UNI”, unidirectional and “BI”, bidirectional are available. Default value is “BI”
- HATCH_ON_THE_FLY_DISTANCE not available for text element,
- HATCH_ANGLE is the angle in degrees for the hatch lines,
- HATCH_NO_OUTLINE defines if the outline of the hatched object is marked or not,
- HATCH_INVERSE inverts the hatching of the object,
- HATCH_TO_OUTLINE not available for text element,
- HATCH_OUTLINE_CORR not supported by current hatch algorithm,
- HATCH_AUTO_LINE_WIDTH not available for text element,
- HATCH_OPTIMIZE_JUMPS optimizes the galvo jumps for hatch lines,
- HATCH_FROM_OUTLINE not supported by current hatch algorithm,
- HATCH_OVERLAP line distance for hatch lines in percent. A negative value will increase the distance between the hatch lines, positive values will decrease the distance between the hatch lines,
- LP is the name of an entry in the global laser parameter table.

Note that width and/or height of the text box can be set to 0 meaning the actual box dimensions are determined by the text box contents and font attributes.

The element LOGO represents a logo. Its content is a file name containing the logo. It has the attributes:

- ID is the element name,
- WIDTH is the width of the logo box,
- HEIGHT is the height of the logo box,
- OFFSET_X is the X-coordinate of the logo box reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the logo box reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the logo box is rotated (counterclockwise if positive) about its reference point in the container coordinate system,
- REFLECT defines whether the contents of the logo box is reflected horizontally (value "H"), vertically (value "V"), or not reflected (value "N"),
- REF_POINT defines the logo box reference point, allowed values being left bottom "LB", center bottom "CB", right bottom "RB", left center "LC", center center "CC", right center "RC", left top "LT", center top "CT", and right top "RT",
- LAYER is the drawing layer to place the logo element at, (not implemented yet!)
- DESCRIPTION is a text string associated with the logo element which may be used at run-time to control the marking process,
- SCALE_X is the scaling factor along the x-axis,
- SCALE_Y is the scaling factor along the y-axis,
- USE_BOX defines whether the bounding box associated with the logo element should be used to bound the logo,
- HATCH defines whether the logo is hatched,
- HATCH_DIRECTION defines the hatch direction of the logo element. For logo elements only the directions "UNI", unidirectional and "BI", bidirectional are available. Default value is "BI"
- HATCH_ON_THE_FLY_DISTANCE not available for logo element,
- HATCH_ANGLE is the angle in degrees for the hatch lines,
- HATCH_NO_OUTLINE defines if the outline of the hatched object is marked or not,
- HATCH_INVERSE inverts the hatching of the object,
- HATCH_TO_OUTLINE not available for logo element,

- HATCH_OUTLINE_CORR not supported by current hatch algorithm,
- HATCH_AUTO_LINE_WIDTH not available for logo element,
- HATCH_OPTIMIZE_JUMPS optimizes the galvo jumps for hatch lines,
- HATCH_FROM_OUTLINE not supported by current hatch algorithm,
- HATCH_OVERLAP line distance for hatch lines in percent. A negative value will increase the distance between the hatch lines, positive values will decrease the distance between the hatch lines,
- LP is the name of an entry in the global laser parameter table.

Note that width and/or height of the logo box can be set to 0 meaning the actual logo dimensions are determined by the original dimensions of the logo.

The element BITMAP represents a bitmap. Its content is a file name containing the bitmap. It has the attributes:

- ID is the element name,
- WIDTH is the width of the bitmap box,
- HEIGHT is the height of the bitmap box,
- OFFSET_X is the X-coordinate of the bitmap box reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the bitmap box reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the bitmap box is rotated (counterclockwise if positive) about its reference point in the container coordinate system,
- REFLECT defines whether the content of the bitmap box is reflected horizontally (value "H"), vertically (value "V"), or not reflected (value "N"),
- REF_POINT defines the bitmap box reference point, allowed values being left bottom "LB", center bottom "CB", right bottom "RB", left center "LC", center center "CC", right center "RC", left top "LT", center top "CT", and right top "RT",
- LAYER is the drawing layer to place the bitmap element at, (not implemented yet!)
- DESCRIPTION is a text string associated with the bitmap element which can be used at run-time to control the marking process,
- DENSITY_X is the density in dpi along the x-axis,
- DENSITY_Y is the density in dpi along the y-axis,
- USE_BOX defines whether the bounding box associated with the bitmap should be used to bound the bitmap,
- INVERTED defines whether the bitmap should be marked inverse (display is not affected),
- LP is the name of an entry in the global laser parameter table.

The element POLYLINE represents a polyline. Its content is a sequence of at least two POINT elements. It has the attributes:

- ID is the element name,
- WIDTH is the width of the polyline box,
- HEIGHT is the height of the polyline box,
- OFFSET_X is the X-coordinate of the polyline box reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the polyline box reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the polyline box is rotated (counterclockwise if positive) about its reference point in the container coordinate system,
- REFLECT defines whether the content of the polyline box is reflected horizontally (value "H"), vertically (value "V"), or not reflected (value "N"),
- REF_POINT defines the polyline box reference point, allowed values being left bottom "LB", center bottom "CB", right bottom "RB", left center "LC", center center "CC", right center "RC", left top "LT", center top "CT", and right top "RT",
- LAYER is the drawing layer to place the element at, (not implemented yet!)
- DESCRIPTION is a text string associated with the element which may be used at run-time to control the marking process,
- USE_BOX defines whether the bounding box associated with the polyline should be used to bound it,
- HATCH defines whether the polyline is hatched,
- HATCH_DIRECTION defines the hatch direction of the polyline element. For polyline elements only the directions "UNI", unidirectional and "BI", bidirectional are available. Default value is "BI"
- HATCH_ON_THE_FLY_DISTANCE not available for polyline element,
- HATCH_ANGLE is the angle in degrees for the hatch lines,
- HATCH_NO_OUTLINE defines if the outline of the hatched object is marked or not,
- HATCH_INVERSE inverts the hatching of the object,
- HATCH_TO_OUTLINE not available for polyline element,
- HATCH_OUTLINE_CORR not supported by current hatch algorithm,
- HATCH_AUTO_LINE_WIDTH not available for polyline element,

- HATCH_OPTIMIZE_JUMPS optimizes the galvo jumps for hatch lines,
- HATCH_FROM_OUTLINE not supported by current hatch algorithm,
- HATCH_OVERLAP line distance for hatch lines in percent. A negative value will increase the distance between the hatch lines, positive values will decrease the distance between the hatch lines,
- LP is the name of an entry in the global laser parameter table.

The element ARC represents a circle arc. Its content is a sequence containing a start POINT, an end POINT, a center POINT, and arc DIRECTION. It has the attributes:

- ID is the element name,
- WIDTH is the width of the arc box,
- HEIGHT is the height of the arc box,
- OFFSET_X is the X-coordinate of the arc box reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the arc box reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the arc box is rotated (counterclockwise if positive) about its reference point in the container coordinate system,
- REFLECT defines whether the content of the arc box is reflected horizontally (value "H"), vertically (value "V"), or not reflected (value "N"),
- REF_POINT defines the arc box reference point, allowed values being left bottom "LB", center bottom "CB", right bottom "RB", left center "LC", center center "CC", right center "RC", left top "LT", center top "CT", and right top "RT",
- LAYER is the drawing layer to place the element at, (not implemented yet!)
- DESCRIPTION is a text string associated with the element which may be used at run-time to control the marking process,
- USE_BOX defines whether the bounding box associated with the arc should be used to bound it,
- HATCH defines whether the arc is hatched,
- HATCH_DIRECTION defines the hatch direction of the arc element. For arc elements only the directions "UNI", unidirectional and "BI", bidirectional are available. Default value is "BI"
- HATCH_ON_THE_FLY_DISTANCE not available for arc element,
- HATCH_ANGLE is the angle in degrees for the hatch lines,
- HATCH_NO_OUTLINE defines if the outline of the hatched object is marked or not,
- HATCH_INVERSE inverts the hatching of the object,
- HATCH_TO_OUTLINE not available for arc element,
- HATCH_OUTLINE_CORR not supported by current hatch algorithm,

- HATCH_AUTO_LINE_WIDTH not available for arc element,
- HATCH_OPTIMIZE_JUMPS optimizes the galvo jumps for hatch lines,
- HATCH_FROM_OUTLINE not supported by current hatch algorithm,
- HATCH_OVERLAP line distance for hatch lines in percent. A negative value will increase the distance between the hatch lines, positive values will decrease the distance between the hatch lines,
- LP is the name of an entry in the global laser parameter table.

The element POINT represents a point in two-dimensional space. Its exact syntax is left unspecified. The current implementation assumes the POINT is given by a sequence of two numbers delimited by one or more spaces. Each number must be a valid float number according to the C language syntax. The dot is used as separator.

The element DIRECTION represents one of the two possible arc orientations, clock-wise or counterclock-wise. The exact syntax is again not specified by the DTD. The current implementation assumes the DIRECTION is either "CW" or "CCW" string.

The element BARCODE represents a barcode object. It has the attributes:

- ID is the element name,
- WIDTH is the width of the barcode, used only if attribute BARCODE_KEEP_SIZE is set to value "BOX".
- HEIGHT is the height of the barcode,
- OFFSET_X is the X-coordinate of the barcode reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the barcode reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the barcode is rotated (counterclockwise if positive) about its reference point in the container coordinate system,
- REFLECT defines whether the content of the barcode is reflected horizontally (value "H"), vertically (value "V"), or not reflected (value "N"),
- REF_POINT defines the barcode reference point, allowed values being left bottom "LB", center bottom "CB", right bottom "RB", left center "LC", center center "CC", right center "RC", left top "LT", center top "CT", and right top "RT",
- LAYER is the drawing layer to place the element at, (not implemented yet!)
- DESCRIPTION is a text string associated with the element which may be used at run-time to control the marking process,
- USE_BOX sets / resets the property "Preserve aspect ration" of the barcode element,
- BARCODE_TYPE selects the type of the barcode element. Allowed types are CODE39_EXT, CODE39, CODE93, CODE128, EAN_UCC128, 2_OF_5, UPC-A, UPC-E, EAN-8, EAN-13, BOOKLAN, CODABAR. Default is CODE39_EXT. Code PDF417 is not supported yet!
- BARCODE_ELEMENT_WIDTH element width of the barcode bars in mm,
- BARCODE_ELEMENT_RATIO ratio of the barcode bars,
- BARCODE_QUIET_ZONE insert quiet zone into barcode,
- BARCODE_CHECK_DIGIT insert check digit into barcode. Check digits are not available for all barcode types!
- BARCODE_INVERT invert barcode marking,

- **BARCODE_KEEP_SIZE** defines if the width of the barcode is calculated out of the element width of the barcode ("ELEMENT"), or if the element width is adjusted so, that the barcode fits into the dimension, defined with attribute **WIDTH** ("BOX").
- **HATCH** this value always must be "Y" for barcodes, as barcodes consist of hatched rectangles,
- **HATCH_DIRECTION** defines the hatch direction of the barcode element. For barcode elements all directions are available. Default value is "BI".
- **HATCH_ON_THE_FLY_DISTANCE** distance for on the fly marking of barcodes in mm, therefore the **HATCH_DIRECTION** must be set to "ON_THE_FLY",
- **HATCH_ANGLE** is the angle in degrees for the hatch lines,
- **HATCH_NO_OUTLINE** defines if the outline of the hatched object is marked or not,
- **HATCH_INVERSE** not available for barcode element,
- **HATCH_TO_OUTLINE** disables the distance of the hatch lines to the outline of the bars. The distance is usually the half beam width of the laser,
- **HATCH_OUTLINE_CORR** the bars of the barcode are corrected with the width of the laserbeam, to get correct outline dimensions of the bars,
- **HATCH_AUTO_LINE_WIDTH** the distance between the hatch lines is adjusted to the width of the barcode bars,
- **HATCH_OPTIMIZE_JUMPS** not available for barcode objects,
- **HATCH_FROM_OUTLINE** not available for barcode objects,
- **HATCH_OVERLAP** line distance for hatch lines in percent. A negative value will increase the distance between the hatch lines, positive values will decrease the distance between the hatch lines,
- **LP** is the name of an entry in the global laser parameter table.

The element DATAMATRIX represents 2D matrix code object. It has the attributes:

- ID is the element name,
- WIDTH is the width of the matrix code,
- HEIGHT is the height of the matrix code,
- OFFSET_X is the X-coordinate of the matrix code reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the matrix code reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the matrix code is rotated (counterclockwise if positive) about its reference point in the container coordinate system,
- REFLECT defines whether the content of the matrix code is reflected horizontally (value "H"), vertically (value "V"), or not reflected (value "N"),
- REF_POINT defines the matrix code reference point, allowed values being left bottom "LB", center bottom "CB", right bottom "RB", left center "LC", center center "CC", right center "RC", left top "LT", center top "CT", and right top "RT",
- LAYER is the drawing layer to place the element at.
- DESCRIPTION is a text string associated with the element which may be used at run-time to control the marking process,
- USE_BOX sets / resets the property "Preserve aspect ration" of the matrix code element,
- DATAMATRIX_TYPE selects the type of matrix code. Allowed square types are: ECC00, ECC50, ECC80, ECC100, ECC130, ECC140 and ECC200_SQUARE.

Allowed rectangular types are: ECC200_RECT

Default setting is ECC200_SQUARE,

- DATAMATRIX_CELL_SIZE sets the cell size in [mm],
- DATAMATRIX_FORMAT_ID sets the format id of the matrix code. This is only used for types ECC00 – ECC140. Allowed values are: NONE, NUM_500, ALPHA_500, ALPHA_PUNC_500, ALPHA_NUM_500. Default value is NONE.
- DATAMATRIX_CELL_DENSITY sets the cell density of the matrix code. If set to MINIMUM the minimum cell density will be calculated. Allowed values are: MINIMUM, 10X10, 12X12, 14X14, 16X16, 18X18, 20X20, 22X22, 24X24, 26X26, 32X32, 36X36, 40X40, 44X44, 48X48,

52X52, 64X64, 72X72, 88X88, 96X96, 102X102, 132X132, 144X144, 8X18, 8X32, 12X26, 12X36, 16X36, 16X48, 9X9, 11X11, 13X13, 15X15, 17X17, 19X19, 21X21, 23X23, 25X25, 27X27, 29X29, 31X31, 33X33, 35X35, 37X37, 39X39, 41X41, 43X43, 45X45, 47X47, 49X49. Default setting is MINIMUM.

- DATAMATRIX_BURST_MODE sets if the matrix code will be marked in burst mode. Allowed values are: N, Y. Default setting is N.
- DATAMATRIX_KEEP_SIZE defines if the size of the matrix code is calculated out of the cell size ("CELL"), or if the cell size is adjusted so, that the matrix code fits into the dimension, defined with attribute WIDTH / HEIGHT ("BOX"). Default setting is "BOX"
- HATCH_DIRECTION defines the hatch direction of the matrix code element. For matrix code elements all directions are available. Default value is "BI".
- HATCH_ON_THE_FLY_DISTANCE distance for on the fly marking of matrix codes in mm, therefore the HATCH_DIRECTION must be set to "ON_THE_FLY",
- HATCH_ANGLE is the angle in degrees for the hatch lines,
- HATCH_NO_OUTLINE defines if the outline of the hatched object is marked or not,
- HATCH_INVERSE not available for matrix code element,
- HATCH_TO_OUTLINE disables the distance of the hatch lines to the outline of the cells. The distance is usually the half beam width of the laser,
- HATCH_OUTLINE_CORR the cells of the matrix code are corrected with the width of the laserbeam, to get correct outline dimensions of the cells,
- HATCH_AUTO_LINE_WIDTH the distance between the hatch lines is adjusted to the width of the matrix code cells,
- HATCH_OPTIMIZE_JUMPS not available for matrix code objects,
- HATCH_FROM_OUTLINE not available for matrix code objects,
- HATCH_OVERLAP line distance for hatch lines in percent. A negative value will increase the distance between the hatch lines, positive values will decrease the distance between the hatch lines,
- LP is the name of an entry in the global laser parameter table.

The element QRCODE represents 2D QR - matrix code object. It has the attributes:

- ID is the element name,
- WIDTH is the width of the QR code,
- HEIGHT is the height of the QR code,
- OFFSET_X is the X-coordinate of the QR code reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the QR code reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the QR code is rotated (counterclockwise if positive) about its reference point in the container coordinate system,
- REFLECT is not available for QR code objects. Allowed is only not reflected (value "N"),
- REF_POINT defines the matrix code reference point, allowed values is left bottom "LB" other reference points are not supported yet,
- LAYER is the drawing layer to place the element at.
- DESCRIPTION is a text string associated with the element which may be used at run-time to control the marking process,
- USE_BOX sets / resets the property "Preserve aspect ration" of the QR code element,
- QRCODE_MODEL selects the type of QR code. Allowed types are Type 1 (1), Type 2 (2) and micro QR (MICRO_QR). Default setting is value "2"
- QRCODE_ECC_LEVEL sets the QR code ECC level. Allowed values are L (7%), M (15%), Q (25%) and H(30%).Default setting is "M"
- QRCODE_CELL_PITCH sets the QR code cell pitch in [mm]
- QRCODE_CELL_ADJUST sets the QR code cell adjust in [mm]
- QRCODE_QUIET_ZONE sets the number cells for the quiet zone of the QR code
- QRCODE_BURST_MODE sets if the QR code will be marked in burst mode. Allowed values are: N, Y. Default setting is N.
- QRCODE_KEEP_SIZE size of QR code objects will always be calculated out of the cell size. Only value "CELL" is allowed.
- HATCH_DIRECTION defines the hatch direction of the QR code element. For QR code elements all directions are available. Default value is "BI".

- HATCH_ON_THE_FLY_DISTANCE distance for on the fly marking of QR codes in mm, therefore the HATCH_DIRECTION must be set to "ON_THE_FLY",
- HATCH_ANGLE is the angle in degrees for the hatch lines,
- HATCH_NO_OUTLINE defines if the outline of the hatched object is marked or not,
- HATCH_INVERSE not available for QR code element,
- HATCH_TO_OUTLINE disables the distance of the hatch lines to the outline of the bars. The distance is usually the half beam width of the laser,
- HATCH_OUTLINE_CORR the cells of the QR code are corrected with the width of the laserbeam, to get correct outline dimensions of the cells,
- HATCH_AUTO_LINE_WIDTH the distance between the hatch lines is adjusted to the width of the QR code cells,
- HATCH_OPTIMIZE_JUMPS not available for QR code objects,
- HATCH_FROM_OUTLINE not available for QR code objects,
- HATCH_OVERLAP line distance for hatch lines in percent. A negative value will increase the distance between the hatch lines, positive values will decrease the distance between the hatch lines,
- LP is the name of an entry in the global laser parameter table.

The element INFOGLYPH represents a 2D matrix code object. It has the attributes:

- ID is the element name,
- WIDTH is the width of the Infoglyph code,
- HEIGHT is the height of the Infoglyph code,
- OFFSET_X is the X-coordinate of the Infoglyph code reference point in the container coordinate system,
- OFFSET_Y is the Y-coordinate of the Infoglyph code reference point in the container coordinate system,
- ANGLE is the angle in degrees by which the Infoglyph code is rotated (counterclockwise if positive) about its reference point in the container coordinate system,
- REFLECT defines whether the content of the Infoglyph code is reflected horizontally (value "H"), vertically (value "V"), or not reflected (value "N"),
- REF_POINT defines the Infoglyph code reference point, allowed values being left bottom "LB", center bottom "CB", right bottom "RB", left center "LC", center center "CC", right center "RC", left top "LT", center top "CT", and right top "RT",
- LAYER is the drawing layer to place the element at.
- DESCRIPTION is a text string associated with the element which may be used at run-time to control the marking process,
- USE_BOX sets / resets the property "Preserve aspect ration" of the Infoglyph code element,
- INFOGLYPH_BLOCKDIM selects the dimension of Infoglyph code.
Allowed dimensions are: 2X2 , 2X3, 2X4, 2X5, 3X2 , 3X3, 3X4, 3X5, 3X6, 3X7, 3X8, 3X9, 4X2 , 4X3, 4X4, 4X5, 4X6, 4X7, 5X2, 5X3, 5X4, 5X5 ,6X3, 6X4, 7X3, 7X4, 8X3 , 9X3).
Default setting is 3X3
- INFOGLYPH_ERRORCOR sets the applied ErrorCorrection.
Allowed values are: ERRORCORRECTION_20,
ERRORCORRECTION_30, ERRORCORRECTION_40,
ERRORCORRECTION_50, ERRORCORRECTION_60,
ERRORCORRECTION_70, ERRORCORRECTION_80,
ERRORCORRECTION_90
Default setting is ERRORCORRECTION_50
- INFOGLYPH_CODESET sets the CodeSet (Bit Depth) for the InfoGlyph code. Allowed values are: CODESET_NOMAPPING,

CODESET_8BIT, CODESET_7BIT_ASCII_US,
CODESET_7BIT_ASCII_GER,
CODESET_6BIT_UPPERLOWER_DOT,
CODESET_6BIT_UPPERLOWER_SLASH,
CODESET_6BIT_UPPER_SPECIAL,
CODESET_6BIT_LOWER_SPECIAL, CODESET_5BIT_UPPER,
CODESET_5BIT_LOWER, CODESET_4BIT_NUMBERS

- INFOGLYPH _BURST_MODE sets if the InfoGlyph code will be marked in burst mode. Allowed values are: N, Y. Default setting is N.
- INFOGLYPH _KEEP_SIZE defines if the size of the InfoGlyph code is calculated out of the cell size ("CELL"), or if the cell size is adjusted so, that the InfoGlyph code fits into the dimension, defined with attribute WIDTH / HEIGHT ("BOX"). Default setting is "BOX"
- HATCH_DIRECTION defines the hatch direction of the InfoGlyph code element. For InfoGlyph code elements all directions are available. Default value is "BI".
- HATCH_ON_THE_FLY_DISTANCE distance for on the fly marking of InfoGlyph codes in mm, therefore the HATCH_DIRECTION must be set to "ON_THE_FLY",
- HATCH_ANGLE is the angle in degrees for the hatch lines,
- HATCH_NO_OUTLINE defines if the outline of the hatched object is marked or not,
- HATCH_INVERSE not available for InfoGlyph code element,
- HATCH_TO_OUTLINE disables the distance of the hatch lines to the outline of the cells. The distance is usually the half beam width of the laser,
- HATCH_OUTLINE_CORR the cells of the InfoGlyph code are corrected with the width of the laserbeam, to get correct outline dimensions of the cells,
- HATCH_AUTO_LINE_WIDTH the distance between the hatch lines is adjusted to the width of the InfoGlyph code cells,
- HATCH_OPTIMIZE_JUMPS not available for InfoGlyph code objects,
- HATCH_FROM_OUTLINE not available for InfoGlyph code objects,
- HATCH_OVERLAP line distance for hatch lines in percent. A negative value will increase the distance between the hatch lines, positive values will decrease the distance between the hatch lines,
- LP is the name of an entry in the global laser parameter table.

5. Examples

The first sample XML drawing consists of a single matrix object.

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<!-- Sample matrix -->
<!DOCTYPE DRAWING SYSTEM '..\system\drawing.dtd'>
<DRAWING>
  <ROOT WIDTH='120.0' HEIGHT='120.0'>
    <MATRIX>
      <FRAME WIDTH='104.0' HEIGHT='90.0'
        OFFSET_X='8.0' OFFSET_Y='15.0'
        NUM_ROWS='4' NUM_COLS='5'
        SHIFT_X='20.0' SHIFT_Y='20.0'
        INTERLEAVE_X='0.0' INTERLEAVE_Y='4.0'
        GROUP_COUNT_X='0' GROUP_COUNT_Y='2'
        GROUP_SHIFT_X='20.0' GROUP_SHIFT_Y='30.0'
        CELL_WIDTH='16.0' CELL_HEIGHT='16.0'
        CELL_OFFSET_X='10.0' CELL_OFFSET_Y='10.0' />
      <DEF_CELL_LAYOUT>
        <TEXT WIDTH='0.0' HEIGHT='0.0'
          OFFSET_X='1.31' OFFSET_Y='7.0'
          FONT='Chrset00' FONT_HEIGHT='3.0'
          FONT_RATIO='0.8'>
          text box
        </TEXT>
      </DEF_CELL_LAYOUT>
    </MATRIX>
  </ROOT>
</DRAWING>
```

The corresponding VLM layout is shown below in Figure 6.

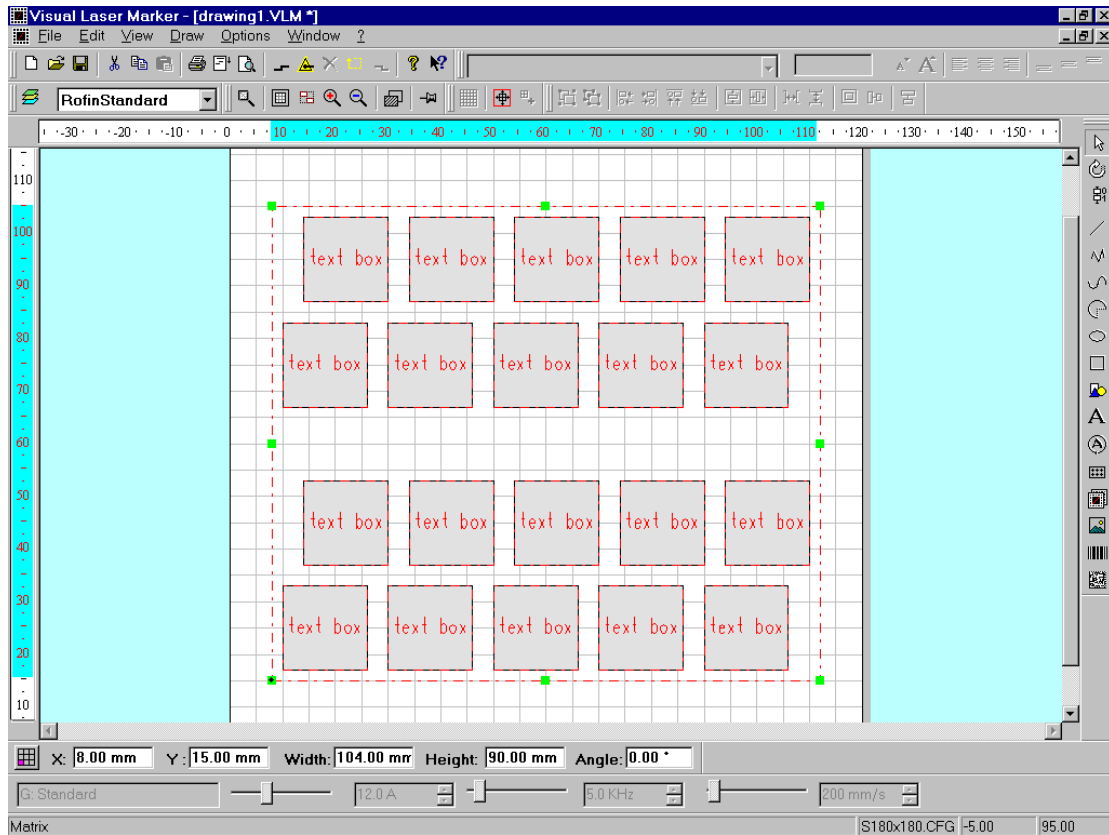


Figure 6.

The second sample consists of two elements, an arc and a polyline.

```
<?xml version='1.0' encoding='ISO-8859-1'?>
<!-- Sample recipe file: text and logo -->
<!DOCTYPE DRAWING SYSTEM '..\..\system\drawing.dtd'>
<DRAWING>
<ROOT ID='test' WIDTH='120.0' HEIGHT='120.0'>
  <ARC ID='ARC01'>
    <POINT> 10.0 10.0 </POINT>
    <POINT> 20.0 10.0 </POINT>
    <POINT> 15.0 5.0 </POINT>
    <DIRECTION> CCW </DIRECTION>
  </ARC>
  <POLYLINE ID='POLY01' REF_POINT='LB' DESCRIPTION='MyLine' LP='Wood'
    REFLECT='N' HATCH='Y' USE_BOX='N'>
    <POINT> 10.0 10.0 </POINT>
    <POINT> 35.0 20.0 </POINT>
    <POINT> 45.0 10.0 </POINT>
    <POINT> 10.0 10.0 </POINT>
  </POLYLINE>
</ROOT>
</DRAWING>
```


The corresponding VLM layout is shown below in Figure 7.

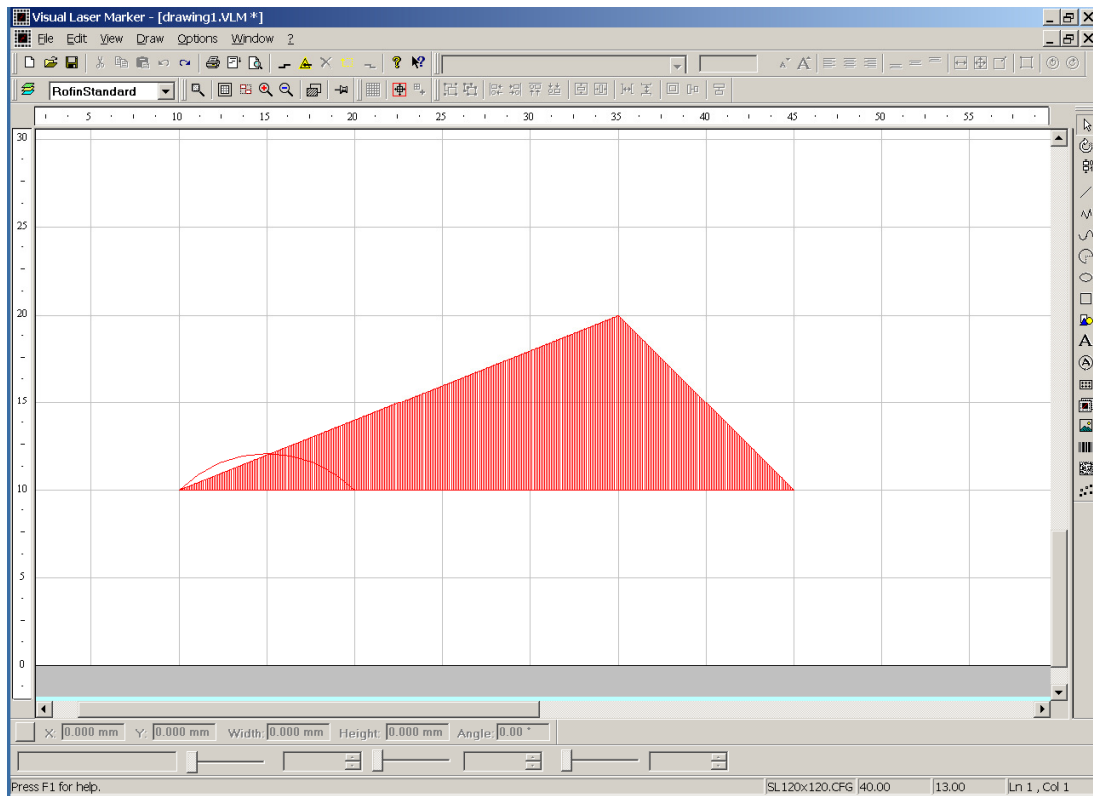


Figure 7

6. Current implementation

The current implementation has the following limitations:

- LAYER attribute is not implemented yet,
- matrix cell reflection is not implemented,
- reflection of objects other than matrix cell is done relative to the axes through the object reference point, so reflection changes object position,
- only “Variation” key can be used to access database entries.
- some hatch attributes are dependent to the used element and may not be available or may not be supported by the hatch algorithm for the used element.
- Code PDF417 is not supported yet in Barcode element

7. References

Extensible Markup Language (XML) 1.0 (Second Edition). W3C Recommendation, 6 October 2000. 59pp. <http://www.w3.org/TR/2000/REC-xml-20001006>

User’s Manual, Visual Laser Marker, version 2.5. Rofin-Sinar Laser GmbH, 2000. 93pp.