Goal: describe geometrical object like lines and rectangles mainly for a barcode drawing library

1 `ga` grammar

A graphic data specification format called '`ga`' *graphic alchemy*, or if you want *generic graphic assembler*.

```
ga<DIM, UINT> := generic graphic assembler
    <DIM>  := numeric type parameter for dimension, for example f64 or i32
    <UINT> := numeric type parameter for quantity, an unsigned integer, i.e. u8

ga<DIM, UINT> := +Elem<DIM, UINT>

Elem<DIM, UINT> := Code<u8> +Args<DIM, UINT>

Code<u8> := End<u8> | State<u8> | Object<u8> | Fn<u8>

End<u8>    :=   0        -- end sequence symbol (reserved for string serialization)
State<u8>  :=   1 ->  31 -- graphic properties
Object<u8> :=  32 -> 239 -- graphic object
Fn<u8>     := 240 -> 255 -- functions

Args<DIM, UINT> : <x: DIM> | <e: u8> | <n: UINT>
   <x: DIM> a dimension value of type DIM
   <e: u8> an enumeration value of type u8 (unsigned byte)
   <n: UINT> an unsigned integer for multiplicity
```

2 Properties

Colors, linecap style etc...

OpCode	Mnemonic key	Graphic property	Operands
1	pen_thick	Line thick	<w: DIM>
2	pen_cap_style	Line cap style	<e: u8>
8	color	...	
30	start_bbox_group	Stop to check the bounding box	-
31	end_bbox_group	Set a bounding box and restart to check	<x1: DIM> <y1: DIM> <x2: DIM> <y2: DIM>

3 Objects

3.1 Lines

A segment that starts from point P1 (x1, y1) and ends in P2 (x2, y2).

OpCode	Mnemonic key	Graphic object	Operands
32	line	Line	<x1: DIM> <y1: DIM> <x2: DIM> <y2: DIM>
33	line_thick	Line with a thick	<w: DIM> <x1: DIM> <y1: DIM> <x2: DIM> <y2: DIM>
36	vbar	Vertical bars	<y1: DIM> <y2: DIM> <b: UINT> <x1: DIM> <t1: DIM> ...
37	hbar	Horizontal bars	<x1: DIM> <x2: DIM> <b: UINT> <y1: DIM> <t1: DIM> ...
38	polyline	Open polyline	<n: UINT> <x1: DIM> <y1: DIM> <x2: DIM> <y2: DIM> ...
39	c_polyline	Closed polyline	<n: UINT> <x1: DIM> <y1: DIM> <x2: DIM> <y2: DIM> ...

3.2 Rectangles

OpCode	Mnemonic key	Graphic object	Operands
48	rect	Rectangle	<x1: DIM> <y1: DIM> <x2: DIM> <y2: DIM>
49	f_rect	Filled rectangle	<x1: DIM> <y1: DIM> <x2: DIM> <y2: DIM>
50	rect_size	Rectangle	<x1: DIM> <y1: DIM> <w: DIM> <h: DIM>
51	f_rect_size	Filled rectangle	<x1: DIM> <y1: DIM> <w: DIM> <h: DIM>

3.3 Text

OpCode	Mnemonic key	Graphic object/Operands
130	text	A text with several gliphs <ax: FLOAT> <ay: FLOAT> <x: DIM> <y: DIM> <s: STRING>
131	text_spaced	A text line with gliphs equally spaced <ax: FLOAT> <ay: FLOAT> <x: DIM> <y: DIM> <gap: DIM> <s: STRING>
132	start_text_group	Texts on the same baseline
133	gtext	A sequence of gliphs sharing a baseline <s: STRING>
134	gtext_spaced	Gliphs separated by a fixed distance sharing a baseline <gap: DIM> <s: STRING>
135	gtext_space	A distance between to consecutive text sharing a baseline <gap: DIM>
140	end_text_group	End text group <ax: FLOAT> <ay: FLOAT> <x: DIM> <y: DIM>

3.4 Function

OpCode	Mnemonic key	Function	Operands
240	move	Translate objects	<n: UINT> <dx: DIM> <dy: UINT>
241	copy	Copy object	<n: UINT> <c: UINT> <dx1: DIM> <dy1: UINT> ...
242	and_copy	Place and copy objects	<n: UINT> <c: UINT> <dx1: DIM> <dy1: UINT> ...
243	grid	Copy next n objects on a grid	<n: UINT> <col: UINT> <row: UINT> <dx: DIM> <dy: DIM>
244	sl_grid		
250	mirror		
255	comment	A string comment	<s: STRING>