

# Open Command-oriented Geometric Graphics Generator

OpenCG<sup>3</sup> Specification Version 0.3.1

Dong Nai-Jia <sup>1</sup>   Lin Yong-Hsiang <sup>2</sup>

<sup>1</sup>National Chiao Tung University  
Department of Computer Science

<sup>2</sup>National Taiwan University  
Department of Agricultural Chemistry

August 19, 2017

## Regular Expressions

$\mathbb{N} := \{ \alpha \mid \alpha \in [0-9]^+ \}$

$\mathbb{R} := \{ \alpha \mid \alpha \in [+ \backslash -]^? ([0-9]^* [.] )^? [0-9]^+ \}$

$\Rightarrow \mathbb{R} \supset \mathbb{N}$

$\mathbb{S} := \{ \alpha \mid \alpha \in '(. * ?)' \mid [. 0-9 A-Z a-z + \backslash -]^+ \}$

$\Rightarrow \mathbb{S} \supset \mathbb{R}$

$\mathbb{W} := \{ \alpha \mid \alpha \in [ \backslash t ]^* \}$

whitespace

## Descriptions

- The matching mechanism abides by the maximal munch rule.
- Each command is whitespace-insensitive except being quoted by a pair of single quotation marks (').

## Context-Free Expansions

$$\mathbf{C} \rightarrow \mathbf{AC} \mid ; \mid \text{EOL}$$

$$\mathbf{A} \rightarrow \mathbf{T(A)} \mid \mathbf{V(A)} \mid \mathbf{S(A)} \mid \mathbf{L(A)} \mid \mathbf{L(A, A, \dots, A)} \mid \mathbf{N} \mid \mathbf{R} \mid \mathbf{S}$$

$$\begin{array}{l} \mathbf{T(\Pi)} \equiv \Pi : n \rangle \rightarrow ( \Sigma(\Pi, n) ) \\ \mathbf{V(\Pi)} \equiv \Pi : n \rangle \rightarrow < \Sigma(\Pi, n) > \\ \mathbf{S(\Pi)} \equiv \Pi : n \} \rightarrow \{ \Sigma(\Pi, n) \} \\ \mathbf{L(\Pi_1, \Pi_2, \dots, \Pi_{n-1}, \Pi_n)} \equiv \mathbb{L} [ \Pi_1 \Pi_2 \dots \Pi_{n-1} \Pi_n ] \rightarrow [ \Pi_1 \dots \Pi_n ] \end{array} \quad \left\| \quad \begin{array}{l} \Sigma(\Pi, n) \rightarrow \overbrace{\Pi \dots \Pi}^{n \text{ items}} \quad (\text{identical}) \\ \mathbf{L(\Pi)} \equiv \mathbb{L} [ \Pi : n ] \rightarrow [ \Sigma(\Pi, n) ] \end{array}$$

## Descriptions

- Each command starts from  $\mathbf{C}$  and ends with a  $;$  or an EOL.
- Non-terminal symbol expansions are prior than function expansions.

## Escape Sequence

- `\x` is an escape sequence.
- If `x` is `\`, then it is treated as a single backslash.
- If `x` is EOL which may vary from platforms, then the sequence is omitted.
- Otherwise, the sequence is ignored and triggers a warning by default.

## Error Handling

- Physical lines are separated by an EOL.
- Logical lines are separated by either a semicolon or an unescaped EOL.
- If the command cannot be parsed by the grammar, then all the characters on the same logical line will be discarded.

## Definitions

- The whole system are divided into four fields and several classes:
  - ① field e-(nviron.): includes class window and class camera.
  - ② field p-(rimitive): includes class point, class circle, etc.
  - ③ field c-(ompound): includes class line, class triangle, class polygon, etc.
  - ④ field a-(uxiliary): includes class attrib and class group.

## Notations

- $\text{class}^x$  denotes the name of a class in the field  $x$ .
- $\text{label}^x$  denotes the globally unique name of the object from a class in the field  $x$ .

## Prototypes

- Argument prototypes are written in a mixture of types and names with underlines.
- Each type with an asterisk indicates that the brackets are used for cross-referencing.
- Cross-reference is a feature for manipulating multiple objects in a single command.
- Each name with a plus/minus/ampersand implies that the given name is used for creating new objects/deleting existed objects/cross-referencing among objects, etc.

# Perspective Projection

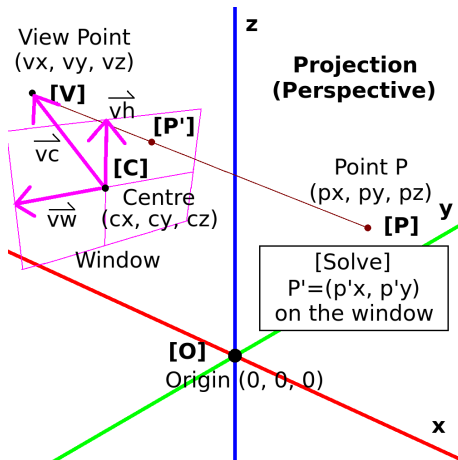


Figure: Projection in Euclidean  $\mathbb{R}^3$  Space

## Command

`create window § +labele` (1)

## Parameters

- label<sup>e</sup> : the name of the object from the class window

## Examples

`create window main`

## Command

```
delete window $ -labele $ string (2)
```

## Parameters

- label<sup>e</sup> : the name of the object from the class window
- string : the text printed right after exiting the current session

## Examples

```
delete window main  
delete window main 'Have a nice day.'
```



# Properties of a Window

Group	Property	Value Format	Example
Digital	dots-per-cm*	$\mathbb{R}$	28.346472
BG-Color	background-rgb*	$\mathbb{N} \in [0, 255] : 3)$	(0 0 0)

## Command

create camera S +label<sup>e</sup>  $\mathbb{R} : 3$ ) center  $\mathbb{R} : 3$  : 2) plane  $\mathbb{R} : 3$  > sight (3)

## Parameters

- label<sup>e</sup> : the name of the object from the class camera
- center : the world coordinate  $(c_x, c_y, c_z)$  of the center of the viewport
- plane : the horizontal and the vertical vectors  $(\vec{v}_w, \vec{v}_h)$  of the viewport
- sight : the reverse line of sight  $\vec{v}_c$  from center to the camera

## Examples

create camera z-top (0 0 1) (<1 0 0> <0 1 0>) <0 0 1>

## Command

```
select camera S &label1 S &label2 (4)
```

## Parameters

- label<sub>1</sub> : the name of the object from the class camera
- label<sub>2</sub> : the name of the object from the class window

## Examples

```
select camera z-top main
```

## Command

```
remove camera S &label1 S &label2 (5)
```

## Parameters

- label<sub>1</sub> : the name of the object from the class camera
- label<sub>2</sub> : the name of the object from the class window

## Examples

```
remove camera z-top main
```

## Command

```
delete camera § -labele (6)
```

## Parameters

- label<sup>e</sup> : the name of the object from the class camera

## Examples

```
delete camera z-top
```

# Properties of a Camera

Group	Property	Value Format	Example
Axis	axis-enable*	$\mathbb{S} \in \{\text{true}, \text{false}\}$	true
Axis-Color	axis-x-rgb*	$\mathbb{N} \in [0, 255] : 3$	(255 0 0)
	axis-y-rgb*	$\mathbb{N} \in [0, 255] : 3$	(0 255 0)
	axis-z-rgb*	$\mathbb{N} \in [0, 255] : 3$	(0 0 255)
Axis-Width	axis-width*	$\mathbb{L} [ \mathbb{R} \in [0, \infty) \quad \mathbb{S} \in \{\text{px}, \text{cm}\} ]$	[2 px]
Grid	grid-enable*	$\mathbb{S} \in \{\text{true}, \text{false}\}$	true
Grid-Color	grid-rgb*	$\mathbb{N} \in [0, 255] : 3$	(191 191 191)
Grid-Width	grid-width*	$\mathbb{L} [ \mathbb{R} \in [0, \infty) \quad \mathbb{S} \in \{\text{px}, \text{cm}\} ]$	[1 px]
Grid-Intvl	grid-interval*	$\mathbb{R} \in [0, \infty)$	10

## Command

`create point`     $\frac{\mathbb{S} \text{ +label}^P : \}^*}{\mathbb{R} : 3} \text{ coord}$     (7)

`create point`     $\frac{\mathbb{S} \text{ +label}^P : \geq n)^*}{\mathbb{R} : 3} \text{ coord} : n)^*$     (8)

## Parameters

- label<sup>P</sup> : the name of the object from the class point
- coord : the world coordinate  $(p_x, p_y, p_z)$  of the object named label<sup>P</sup>

## Examples

```
create point 'origin'    (0 0 0)
create point {X-1 X-2}  (1 0 0)
create point (Y-1 Y-2) ((0 1 0))
create point (Z D1 D2) ((0 1 0) (1 1 1))
```

## Command

`delete point`     $\S$   $-\text{label}^P$  :  $\}$ <sup>\*</sup> (9)

## Parameters

- $\text{label}^P$  : the name of the object from the class point

## Examples

```
delete point 'origin'  
delete point {Z D1 D2}
```



# Properties of a Point

Group	Property	Value Format	Example
In-Type	type*	$\mathbb{S} \in \{\text{dot}, \text{square}\}$	dot
In-Size	radius*	$\mathbb{L} [ \mathbb{R} \in [0, \infty) \ \mathbb{S} \in \{\text{px}, \text{cm}\} ]$	[3 px]
In-Color	fill-rgb*	$\mathbb{N} \in [0, 255] : 3 )$	(0 0 0)
	fill-rgba	$\mathbb{L} [ \mathbb{N} \in [0, 255] : 3 ) \ \mathbb{R} \in [0, 1] ]$	[(31 0 63) 0.5]
	fill-hsv	$\mathbb{L} [ \mathbb{R} \in [0, 360] \ \mathbb{R} \in [0, 1] : 2 ) ]$	[240 (0.8 1.0)]
	fill-hsl	$\mathbb{L} [ \mathbb{R} \in [0, 360] \ \mathbb{R} \in [0, 1] : 2 ) ]$	[240 (1.0 0.6)]
Ex-Type	border-type*	$\mathbb{S} \in \{\text{none}, \text{mid}, \text{in}, \text{out}\}$	none
Ex-Width	border-width*	$\mathbb{L} [ \mathbb{R} \in [0, \infty) \ \mathbb{S} \in \{\text{px}, \text{cm}\} ]$	[1 px]
Ex-Color	border-rgb*	same as fill-rgb	(127 127 127)
	border-(...)	same as fill-rgba/hsv/hsl	(skipped)

## Command

`create line`      $\frac{\$ \text{+label}^c : \}^*}{\$ \text{\&label}^p : 2\}$      (10)

`create line`      $\frac{\$ \text{+label}^c : \geq n)^*}{\$ \text{\&label}^p : 2\} : n)^*$      (11)

## Parameters

- $\text{label}^c$  : the name of the object from the class line
- $\text{label}^p$  : the name of the object from the class point

## Examples

`create line`    `seg-1`            `{X-1 Y-1}`

`create line`   `{seg-2 seg-3}`   `{X-2 Y-2}`

# Delete Line Segments

## Command

`delete line`     $\S$  -label<sup>c</sup> : }<sup>\*</sup> (12)

## Parameters

- label<sup>c</sup> : the name of the object from the class line

## Examples

```
delete line seg-1  
delete line {seg-2 seg-3}
```

## Command

`create attrib $ +labela : }* L [ L [ $ classpc L [ $ prop $ value : ] ] : ]* (13)`

`create attrib $ +labela : }* L [ L [ $ classpc L [ $ prop $ value : ] ] : ]* (14)`

## Parameters

- label<sup>a</sup> : the name of the object from the class `attrib`
- class<sup>pc</sup> : the name of a class in the field primitive or compound
- prop : the property of the object from class<sup>pc</sup>
- value : the value of prop in the designated format

## Examples

```
create attrib (magenta dashed-and-translucent-line) \  
[[point fill-hsv '[300 (1.0 1.0)]]' ] \  
[line [style dashed] [fill-rbga '[(0 255 0) 0.5]]'] ] ]
```

## Command

`attach attrib  $\S$  &labela : )*  $\S$  &labelpc : }*` (15)

`attach attrib  $\S$  &labela : )*  $\S$  &labelpc : )*` (16)

## Parameters

- label<sup>a</sup> : the name of the object from the class attrib
- label<sup>pc</sup> : the name of the object from a class in the field primitive or compound

## Examples

```
attach attrib red point-0
attach attrib (red large) point-1
attach attrib blue {point-2 rect-0}
attach attrib (5px black) {point-3 circ-0}
attach attrib (red thick) (point-4 line-0 triangle-0)
```

## Command

`detach attrib`  $\frac{\$ \&\text{label}^a : \}^*}{\$ \&\text{label}^{\text{pc}} : \}^*}$  (17)

`detach attrib`  $\frac{\$ \&\text{label}^a : )^*}{\$ \&\text{label}^{\text{pc}} : )^*}$  (18)

## Parameters

- $\text{label}^a$  : the name of the object from the class `attrib`
- $\text{label}^{\text{pc}}$  : the name of the object from a class in the field primitive or compound

## Examples

```
detach attrib red point-0
detach attrib {red large} point-1
detach attrib blue {point-2 rect-0}
detach attrib {5px black} {point-3 circ-0}
detach attrib (red thick) (point-4 line-0 triangle-0)
```

## Command

`delete attrib § -labela : }*` (19)

## Parameters

- label<sup>a</sup> : the name of the object from the class attrib

## Examples

```
delete attrib red  
delete attrib {5px large}
```

## Command

`assign operat S action S class N repeat [=  $\infty$ ]` (20)

## Parameters

- action : the name of the corresponding action of class
- class : the name of a class
- repeat : the amount of the commands emitting operation names

## Examples

```
assign operat create point 2
x-axis (1 0 0)
y-axis (0 1 0)
// Back To Normal
```