

svg_bulk_export 0.7.0

Installation in draw.io

Execute following steps to install and use the plugin:

1. [Download](#) and install draw.io (plugin is developed and tested with [version 20.3.0](#))
2. Start draw.io with the command line argument **--enable-plugins**
 - You can add it to the link of draw.io
 - The argument is needed to add the plugin and also to use the plugin
3. Choose the plugin file in the open file dialog *Extras|Plugins...|Add|External Plugins|Select File...*
4. Draw.io adds the plugin file to the folder **C:\Users\\$(User)\AppData\Roaming\draw.io\plugins**
 - Add the file *svg_bulk_export.config.json* to this folder
5. After restart of draw.io three new menu items should be added to the bottom of menu *Extras*
 - SVG bulk export – Single symbol
 - SVG bulk export – All symbols
 - SVG bulk export - Create zip file with folder structure

Config file *svg_bulk_export.config.json*

| Property | Type | Description |
|---|---------|---|
| addDrawIOFilenameAsPrefix | boolean | Prepends the filename of the opened drawio file followed by a hyphen. (also look at SVG Filename) |
| openDevTools | boolean | Opens the Developer Tools (which are also accessible in the menu Help->Open Developer Tools) for debug purpose at startup. |
| writeToFileSystem | boolean | Writes svg files to the file system. Due to missing permission to create folders inside a drawio plugin the user has to create folders (defined by the property 'exportPath' in <i>layerGroups</i>) manually in advance. |
| zipFilePath | string | Exports all svg files to a single zip file including folders as defined by the property 'exportPath' in <i>layerGroups</i> . This bypasses the missing permission to create folders inside a drawio plugin (have a look at property <i>writeToFileSystem</i>). |
| cleanup | object | Performs some cleanup algorithms like resolve of transforms, element conversions, centering of origo layer, etc. |
| <ul style="list-style-type: none">• enabled | boolean | Enables the cleanup algorithms overall |

| Property | Type | Description |
|---|---------|--|
| <ul style="list-style-type: none"> unitScaling | object | Contains properties for the unit scaling algorithm |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> enabled | boolean | Enables unit scaling. The scaling factor is defined by properties of the symbol layer (Dimension (Width, Height), UoM (mm), Value) and the width and height of the symbol layer. Only the unit of measure (UoM) 'mm' is supported! |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> millimetersModelToDisplay | number | Defines the relation between one unit in the svg model and the display size (attributes width & height of the svg element). Width and height are computed from the viewBox by this common factor such that the display sizes of different svg files match to each other. |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> dimensions | object | Defines parameters to generate the dimension layer if the parameter 'DimensionLayer' is set in a layer group (see below). |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> color | string | The stroke and text color of the dimension layer (see below). |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> padding | object | Padding which is added to the viewBox of the svg (or the rounded viewBox if the parameter 'roundViewBox' is set to true, (see below)). |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> x | number | Padding in horizontal direction (added to left and right). |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> y | number | Padding in vertical direction (added to top and bottom). |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> strokeWidth | number | Defines the attribute 'stroke-width' of elements of the symbol layer. Furthermore the vector effect 'non-scaling-stroke' is removed from elements of the symbol layer. |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> roundViewBox | boolean | After unit scaling the viewBox is rounded down to integer to get better aligned auto-generated grid and dimension layers (see below). |
| <ul style="list-style-type: none"> origoCircle | object | Replaces the origo layer by a circle (as ellipse element). |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> radius | number | The radius of the origo circle. |

| Property | Type | Description |
|-----------------------|--------|--|
| ◦ color | string | The stroke color of the origo circle (see below). |
| • connectionCircle | object | Replaces each connection in the connection layer by a circle (as ellipse element). |
| ◦ radius | number | The radius of the connection circle. |
| ◦ color | string | The stroke color of the connection circle (see below). |
| ◦ direction | object | Adds a direction indicator (small circle) on the stroke of each connection. |
| ▪ radius | number | The radius of the direction indicator. |
| ▪ fill | string | The fill color of the direction indicator (see below). |
| ellipseAsPath | object | If enabled, all ellipses of the symbol layer are replaced by a path. |
| sendSymbolLayerToBack | object | If enabled, the symbol layer is <i>send to back</i> such that they it is drawn first and thus will show up behind all other layers. If disabled, other layers like the origo or connection layers can be overlapped by the symbol layer. |
| grid | object | Defines parameters to generate the grid layer if the parameter 'GridLayer' is set in a layer group (see below). |
| • padding | object | Padding which is added to the viewBox of the dimension layer (if exists) or otherwise the viewBox of the svg (or the rounded viewBox if the parameter 'roundViewBox' is set to true, (see below)). |
| ◦ x | number | Padding in horizontal direction (added to left and right). |
| ◦ y | number | Padding in vertical direction (added to top and bottom). |

| Property | Type | Description |
|---|---------|---|
| <ul style="list-style-type: none"> lineGroups | object | Contains one or several lineGroups. Each lineGroup defines a grid at a specific level of detail. The key of a lineGroup is the layer name of its grid (<i>g</i> element). In detail a lineGroup is an object itself and can have following properties: |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> distance | number | The horizontal and vertical distance between two adjacent lines. |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> stroke | object | Contains some properties which define the visual presentation of the grid lines. |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> width | number | The width of grid lines (svg attribute <i>stroke-width</i>) |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> color | number | The color of grid lines (svg attribute <i>stroke</i> , (see below)) |
| <ul style="list-style-type: none"> <ul style="list-style-type: none"> <ul style="list-style-type: none"> opacity | number | The opacity of the grid lines (svg attribute <i>stroke-opacity</i>) |
| layerGroups | object | Contains one or several layerGroups. Each layerGroup can have its own <i>enabledLayers</i> such that svg files are generated with different combinations of enabled layers. In detail a layerGroup is an object itself and can have following properties: |
| <ul style="list-style-type: none"> filenamePostfix | string | Overrides the last part of the filename. (also look at SVG Filename) |
| <ul style="list-style-type: none"> exportPath | string | By default svg files are exported to the folder containing the opened drawio file. This <i>exportPath</i> overrides or changes the default export path. It can be an absolute path which completely overrides the default path. Or it can be a path relative to the default path. |
| <ul style="list-style-type: none"> exportPerOptionLayer | boolean | If enabled a svg file is exported for each option layer. Option layers are defined by starting with <i>Option</i> . For each option layer a svg file will be exported by enabling the option layer and the <i>enabledLayers</i> . |
| <ul style="list-style-type: none"> enabledLayers | array | An array containing the layers which shall be exported for this <i>layerGroup</i> . |

| Property | Type | Description |
|--|--------|---|
| <ul style="list-style-type: none"> GridLayer | string | The name of the auto-generated grid layer. |
| <ul style="list-style-type: none"> DimensionLayer | string | The name of the auto-generated dimension layer. |

Color properties

All color properties above can be defined like a [color CSS property](#) (keyword, hexadecimal string notation, RGB functional notation, ...), e.g.:

- "red" (keyword)
- "#ff0000" (hexadecimal string notation)
- rgb(255, 0, 0) (RGB functional notation)

Padding

Different spaces (like padding) can be added between the layers from outer to inner in this order:

- *cleanup.grid.padding* (the grid padding is part of the grid layer)
 - dimension layer
 - *cleanup.unitScaling.dimensions.padding*
 - *some space added by viewBox rounding (config.cleanup.)*
 - other layers (symbol layer, label layer, origo layer, connection layer, ...)

SVG Filename

The filename of an exported svg file has following format:

{drawio_filename}-{symbol}_{layer}

The *{layer}* is the name of the option layer if *exportPerOptionLayer* has been enabled or the name of the layerGroup otherwise. The property *filenamePostfix* completely overrides *{layer}*.

The filename can be influenced by following properties:

- addDrawIOFilenameAsPrefix
 - Prepends the filename of the opened drawio file followed by a hyphen: *{drawio_filename}-*
- filenamePostfix
 - overrides the last part of the filename: *_{layer}* (including the prefixed underscore)

Changelog

0.7.0

- the dimension properties *width* and *height* for unit scaling are now set directly on the symbol layer (named *Width* and *Height*). The old way to get the values from the dimension layer (named *W* and *H*) still works for testing but is marked as deprecated and will be removed in a future version

- added alert for the case that the UoM (unit of measure) is not equal to 'mm' which is the only supported unit
- the origo is rounded to the nearest integer
- the attributes *stroke-linecap* and *stroke-linejoin* are set to *round* to get smoother line ending and line connections
- json config file *svg_bulk_export.config.json*
 - added the array property *_comments* which contains several comments for the *json config file*
 - added many new properties for the new features which are explained in detailed [above](#) and non-detailed below
 - *cleanup.unitScaling.roundViewBox*
 - the viewBox is rounded down to integer after unit scaling
 - *cleanup.ellipseAsPath*
 - ellipses of the symbol layer are converted to pathes
 - *cleanup.origoCircle* and *cleanup.connectionCircle*
 - origo and connections are replaced by circles (as ellipse element)
 - *cleanup.sendSymbolLayerToBack*
 - the symbol layer can be send to back
 - *cleanup.unitScaling.dimensions* and *layerGroup.DimensionLayer*
 - the dimension layer is auto-generated with an optional direction indicator (shown as a *dot* on the stroke of the connection circle)
 - *cleanup.grid* and *layerGroup.GridLayer*
 - the grid layer is auto-generated
 - *cleanup.unitScaling.millimetersModelToDisplay*
 - attributes *width* and *height* of the svg element are computed by a common factor such that the display sizes of different svg files match to each other
 - *cleanup.unitScaling.strokeWidth*
 - sets the attribute *stroke-width* on elements of the symbol layer and removes the vector effect 'non-scaling-stroke'

0.6.0

- added unit scaling by dimension and symbol layer
 - can be enabled by the new config property 'cleanup.unitScaling.enabled'
- moved boolean 'cleanup' to 'cleanup.enabled' in the config file
- added spinner for waiting

0.5.0

- cleaned the console log
- warning alerts (message boxes) pop up only once instead of multiple times

0.4.0

- In the config file (*svg_bulk_export.config.json*) the new flag 'cleanup' has been introduced to enable/disable the cleanup algorithm.
- Major Features:
 - resolve of transforms (as far as possible, e.g. not for ellipse/text elements)
 - transforms for ellipse element are simplified

- element conversions
 - circle -> ellipse
 - rect -> path
- origo layer is the center of the svg
- resolve of switch elements including foreignObject and text element
- Some remarks:
 - Image elements are not supported and are removed with a warning message box (e.g. the gear image in symbol 'ND0033').
 - The unit scaling is implemented but not tested. Thus it is disabled at the moment.
 - The export takes some seconds without indication of the progress.

0.3.0

- improvements:
 - files can be exported into a single zip file instead of folders
 - workaround for creating folders manually before export
 - introduced two new settings to select folder/zip export
 - writeToFileSystem (default: true)
 - zipFilePath: file path to a zip file which contains the exported files
 - added menu entry 'SVG bulk export - Create zip file with folder structure' to menu 'Extras'
 - creates the zip file 'svg_bulk_export_folders.zip' which contains the folders (without files) to export to file system (see setting 'writeToFileSystem' above)
 - so the folders do not have to be created manually
- bugfixes:
 - Export by 'All symbols' caused error 'Nothing is selected' if the draw.io file contained only one page/tab and no files have been exported
 - Unknown error - 'Uncaught TypeError: l.startsWith is not a function'
 - instead of the real layer name the name '[object Element]' could be exported

0.2.0

- Layers are restored to the state before export. Please tell me if you need a different behavior.
- The active symbol (tab) is restored after export.
- The attribute 'exportPath' has been introduced for layer groups in the config file to optionally export to a different path (see sample in config file)
- Due to missing permissions in the plugin, I was not able to create folders during export. So the 'exportPath' has to exist before export. An error message is shown if the path does not exist. I hope this restriction is not too disturbing.