



Patternodes 2

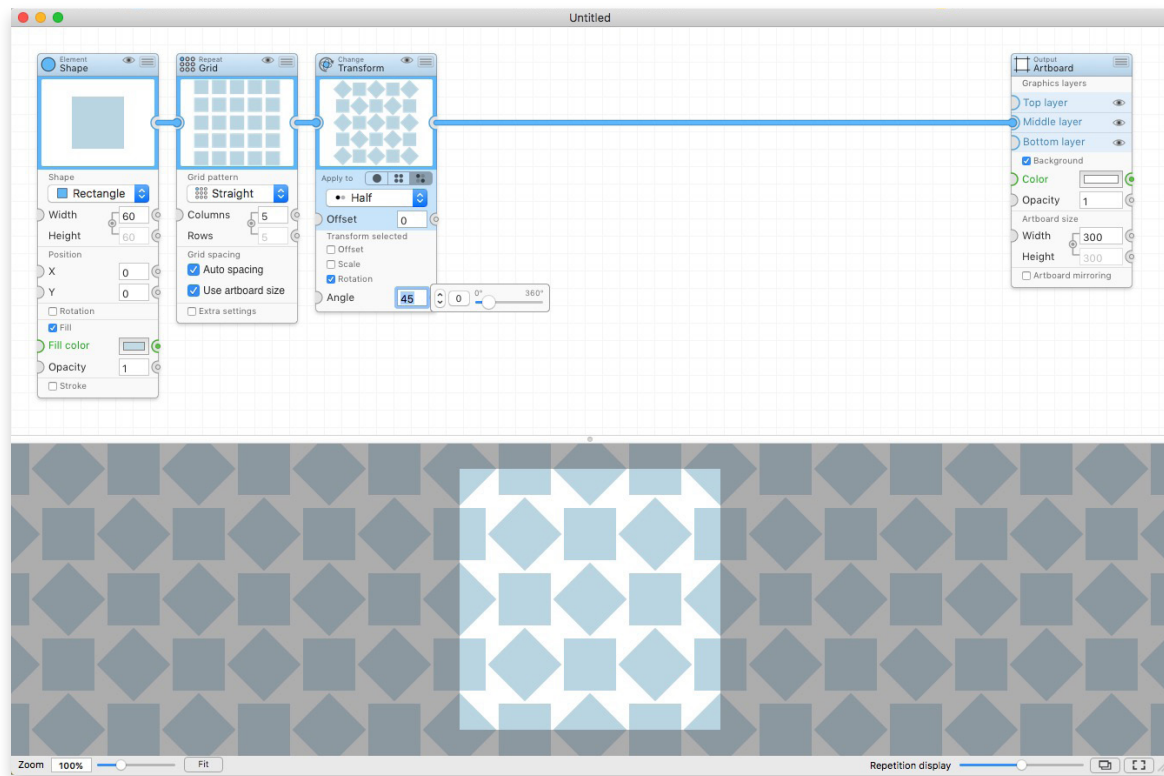
Parametric vector patterns and animations

Version 2.1.x

Manual

Contents

1 Getting started	1	4 Output preview	17
1.1 Introduction	1	4.1 Output display scaling	17
1.2 Nodes and connections	2	4.2 Repetition display	17
1.3 Interface layout	3	4.3 Full screen preview	17
1.4 Built in help features	4	4.4 Separate preview panel	18
2 Nodes	5	5 Animation	19
2.1 Adding, moving and removing nodes	5	5.1 Animation control	19
2.2 Editing node parameters	6	5.2 Animated values	19
2.3 Element nodes	8	5.3 Animation export	19
2.4 Repeater nodes	9	6 Saving and exporting	20
2.5 Change nodes	10	6.1 Saving Patternodes .pnds files	20
2.6 Group operation nodes	10	6.2 Copying the pattern to the pasteboard	20
2.7 Value nodes	11	6.3 File export (bitmap or vectors)	21
2.8 Color nodes	11	7 Sample files	22
2.9 Other nodes	11	8 Buying a license	23
3 Connections	12	9 Distribution and credits	24
3.1 Sockets and connections	12	Licenses	24
3.2 Making and breaking connections	12	Review requests, feedback and bug reports	24
3.3 Connection types	13	Credits	24
3.4 Graphics connection groups	14	Version history	25
3.5 Connection requests and advanced use	15		



1 Getting started

1.1 Introduction

Patternodes is a tool for creating graphical patterns, illustrations or animations using a parametric node-based interface. It lets you construct a graphical result using a sequence of linked **nodes**, each representing an action like creating, changing or repeating graphical **elements**.

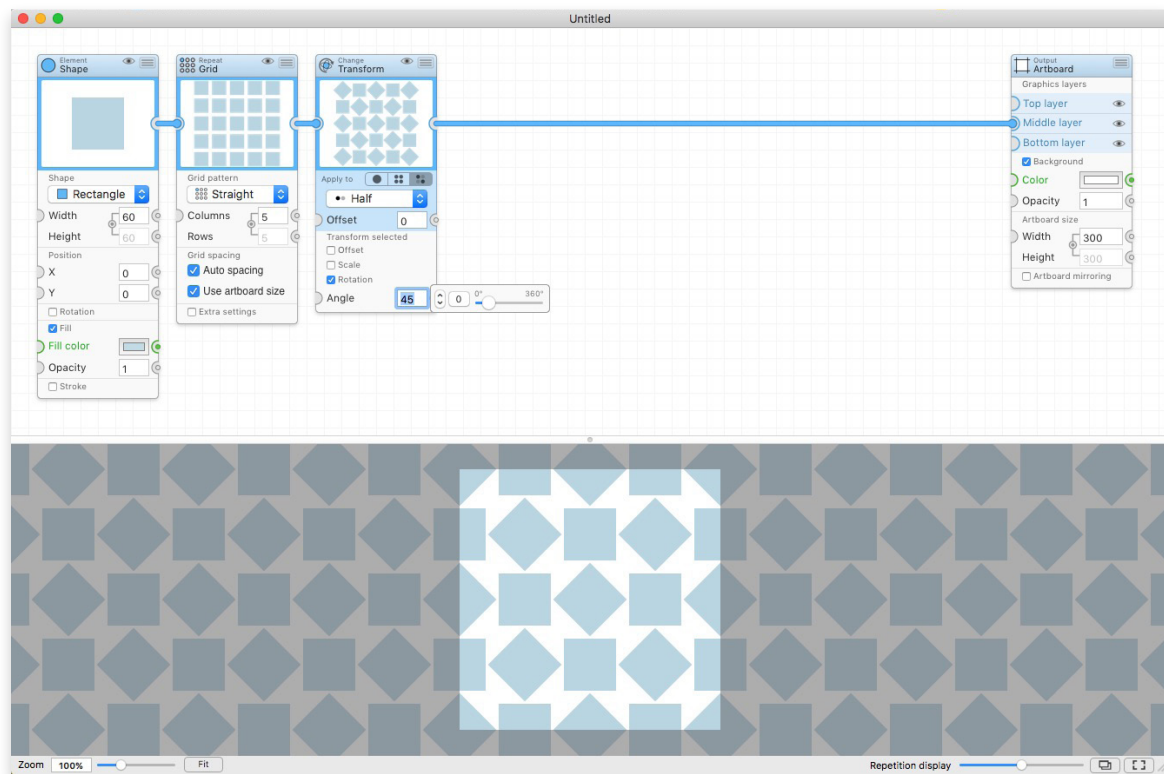
While this requires you to work in a slightly different way from traditional illustration software, it has several big advantages:

First of all it makes it easy to tweak and try making adjustments to any aspect the sequence at any time and see the end result instantly, without having to undo/redo a lot of steps.

Second, it makes a lot of tedious steps in pattern creation easy, like complex multi-level repetitions, randomizing repeated elements, repeating edge elements to make the pattern seamless, replacing repeated elements or creating symmetries.

Third, once you have the parameters describing the result set up you can easily animate them.

Together this means that Patternodes is probably not for every job, but for creating patterns or repeating or animating graphics it provides a powerful new way of working with graphics.



1.2 Nodes and connections

In Patternodes the pattern is defined by a sequence of *nodes*, or little boxes with parameters, each representing a step in creating the result.

Flow from left to right

The basic premise is that the linked nodes represent a flow starting with a node that creates a graphical *element* (like a square, line, image etc). Different operations are then done in sequence to this *element*, like rotating, scaling, repeating, changing color etc. This is done by connecting the graphics socket of the element node to the input of the next node in the sequence. Finally the sequence ends in the one of the layers of the Artboard output node, which adds the graphics to the output.

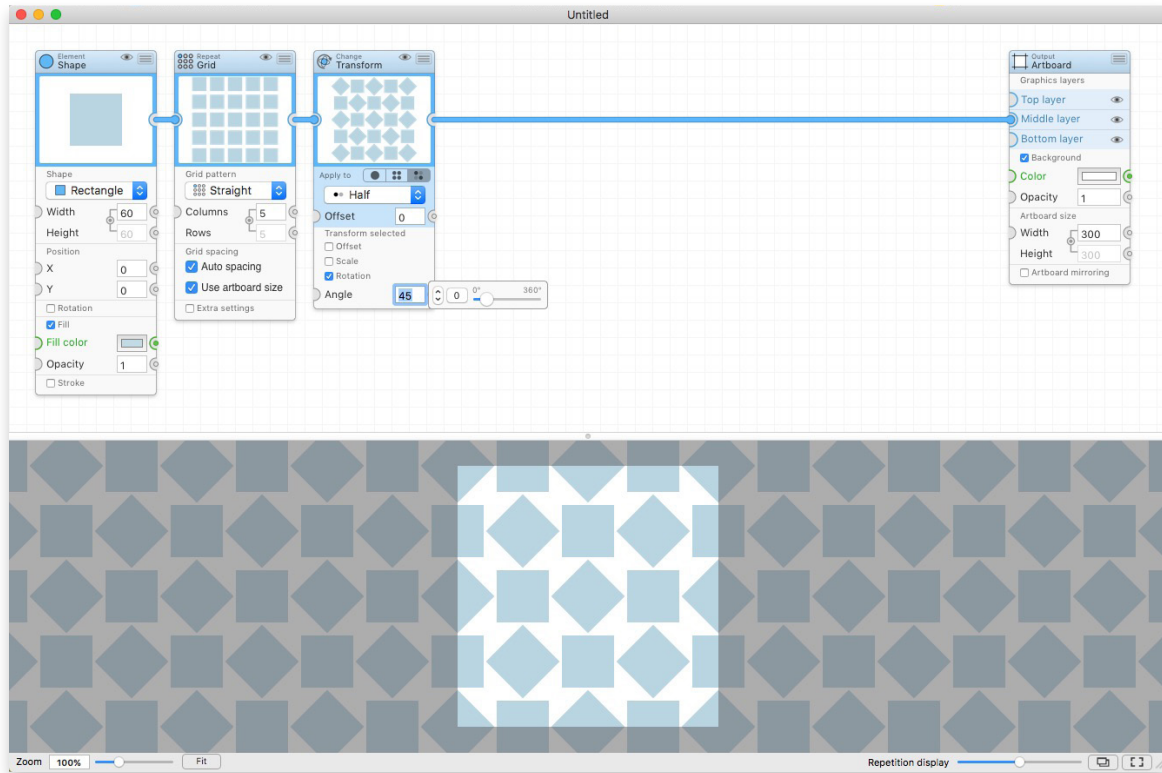
Parameters

Most nodes have parameters that control their functionality, these can be either numeric *values* (gray), *colors* (green) or *graphics* (blue). You can edit the colors or values by clicking on their fields, or for values using the mouse scroll wheel or controls that appear when you hover over a field. When a field is active you can also use the arrow keys to step up/down. The step can be changed by holding shift or option.

Connections

The blue graphics connections define the sequence of the actions, from creating elements to adding them to the artboard.

Most number and color parameters also have input (left side) and/or output sockets (right side), which means you can connect these parameter to a parameter of the same type in another node. This lets you re-use values and establish relationships between different parameters.



1.3 Interface layout

The interface is a single window with a split view for each document, with the node editor at the top and a preview of the output at the bottom.

Node editor

At the top is the node editor, a scrolling view containing all the nodes making up the current node sequence. This is where you work with the pattern by adding and removing nodes, connecting them to each other and adjusting their parameters.

To add new nodes, right-click (or control-click) in the view to bring up a context menu or use the Nodes menu. You can also add nodes on an existing connection or after an existing node to automatically connect them.

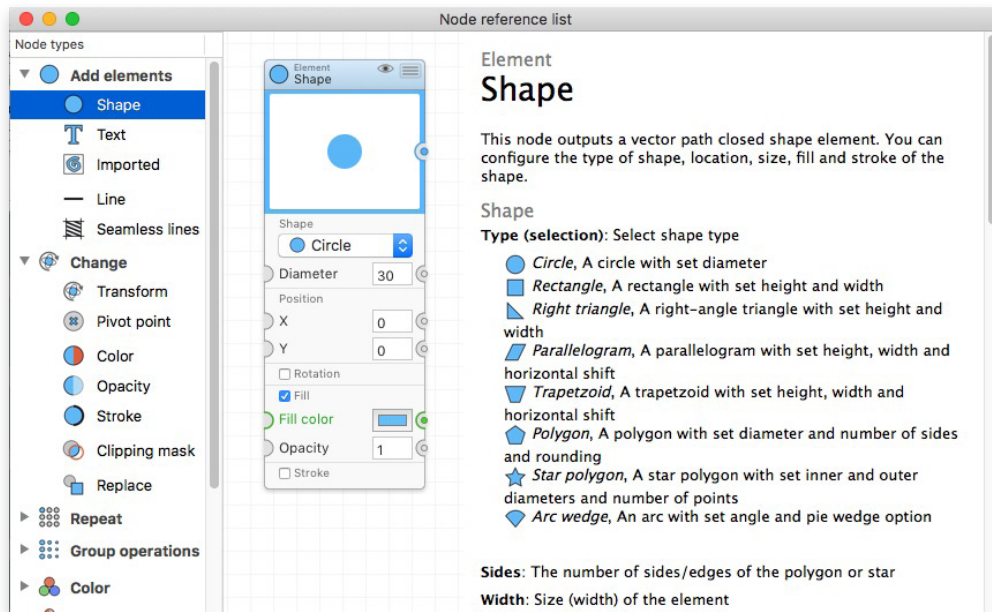
To scroll use the scroll bars or hold space and click and drag.

Output preview

At the bottom is the output preview, showing a live updating tiled preview of the output described by the nodes above.

If this is enabled in the settings, the preview also shows an outline around the elements in the output created or directly modified by the nodes selected in the node editor.

There are also two sliders below this view that control the display of the pattern. To the left is a zoom slider letting you quickly see the pattern at a larger or smaller scale. To the right is a slider that lets you adjust the opacity of the pattern repetition to help if you're working with a seamless pattern.



1.4 Built in help features

The best way of learning to use Patternodes is simply to play around with it and see what happens. To support this there are a couple of built-in help features (and this manual).

Interface tool tips

The main help feature in Patternodes is that most interface elements (buttons, menu items, etc.) have tool tips, little descriptive messages that show up when you hover the mouse cursor over them for a while. By using these you can quickly get a hint at what a parameter represents, what a node does, or what will happen if you press a button.

Node Reference list

Accessible from the Help menu there is also a reference list of all available nodes with information about how they work and what the parameters represent. For each node there is also a list of Sample files using the node where you can see it used.

You can also access reference help for a node from the context menu of the node.

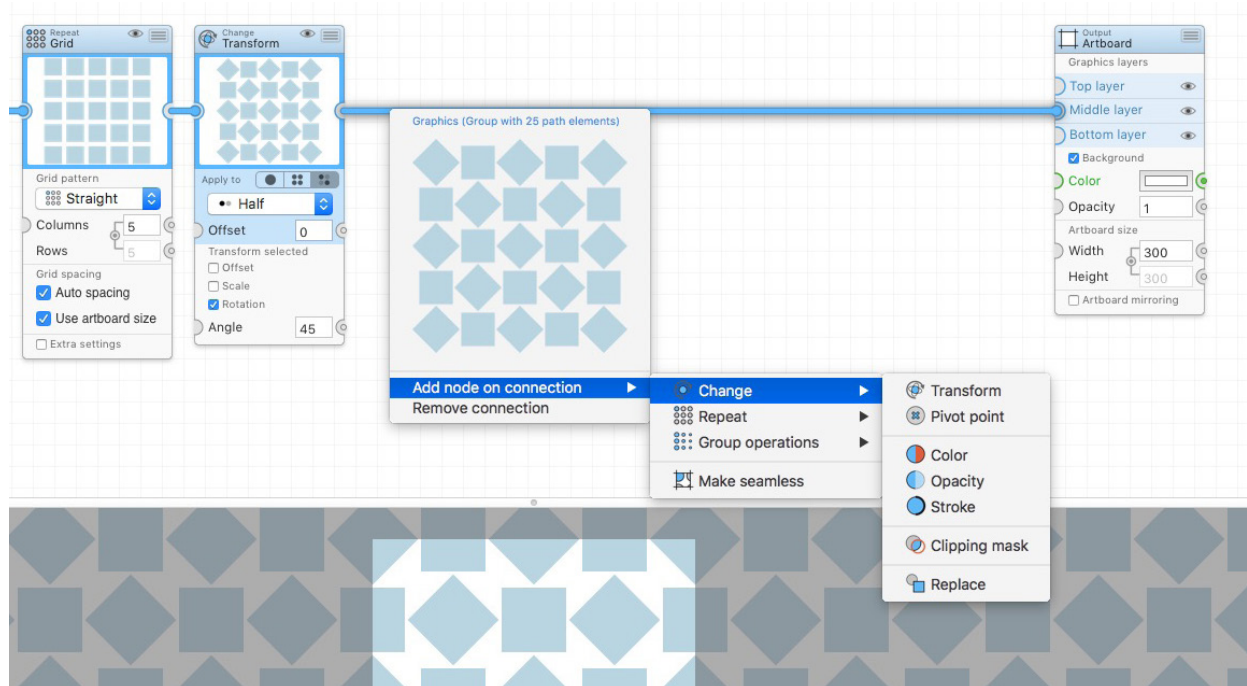
Warning messages

As a complement to this there is a system with simple warning messages that show up when some nodes are not connected or configured properly. This can be a further help to let you understand how the nodes are supposed to work, and why they do not in some cases.

Nodes with warnings get a little yellow triangle icon in the header, and a little box with the number of warnings is shown in the top right corner of the editor view. Clicking either one brings up a list of the warning messages. Clicking on a message in the list selects the offending node if there is one.

2 Nodes

There are a lot of different nodes that you can use in Patternodes, each with a different functionality. Try them out and check the tool tips in the application to learn more.



2.1 Adding, moving and removing nodes

Adding new nodes

To add new nodes, right-click or control-click in the editor view to bring up the "Add node" menu, or use the Add node item in the Nodes menu. The newly created node will then need to be connected to the other nodes to be included in the pattern creation.

To do this quickly you can create new nodes by right-clicking/control clicking the on an existing connection to add a new node on this connection, or on a node to add a new node connected after the node.

Replacing nodes

You can also use the node context menu to change an existing node to another type by using the Replace node menu item. This will change the node but leave it connected in the same place.

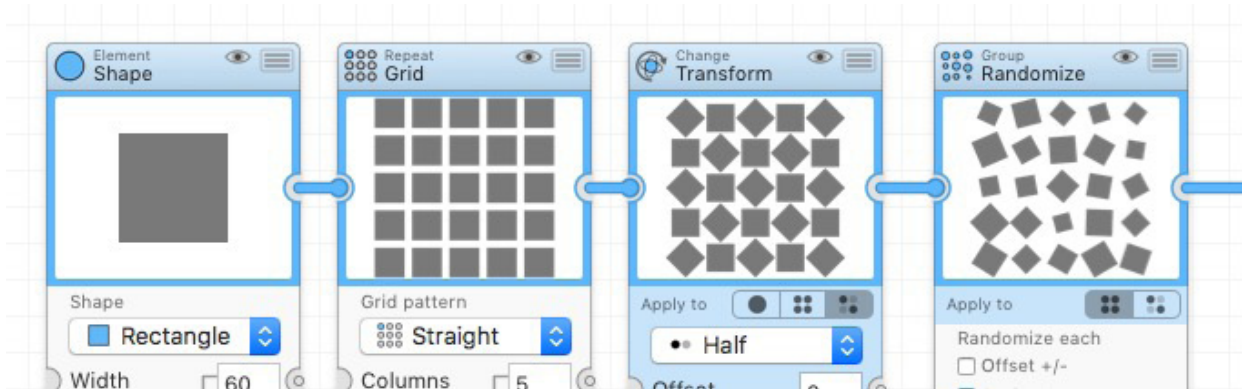
Selecting, moving and copying nodes

You can select nodes by clicking them, or by holding down the mouse and dragging to make a selection box to select multiple nodes. You can then click and drag nodes to move them, copy, cut and paste single or multiple selected nodes using the standard edit menu commands. You can also duplicate nodes by holding option and dragging to create a new copy.

The position of the nodes has no effect on the output, only the connections between them.

2.2 Editing node parameters

The nodes have different parameters controlling them, each represented by controls in the node interface.



Graphics previews

In addition to the main output preview at the bottom of the window all nodes that create or change graphic elements have a little preview in them so that you can see what change this step produces.

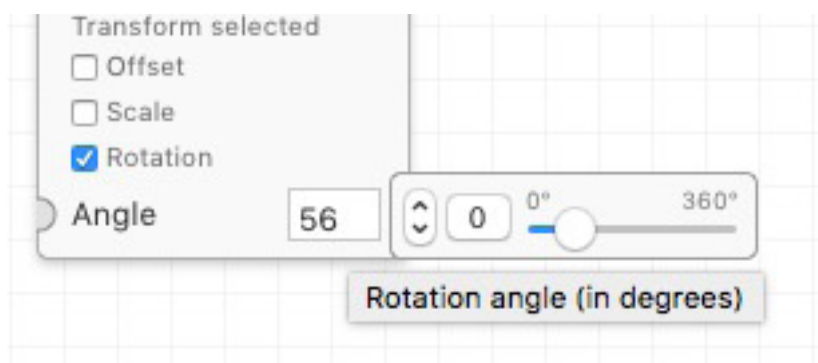
Numeric parameters

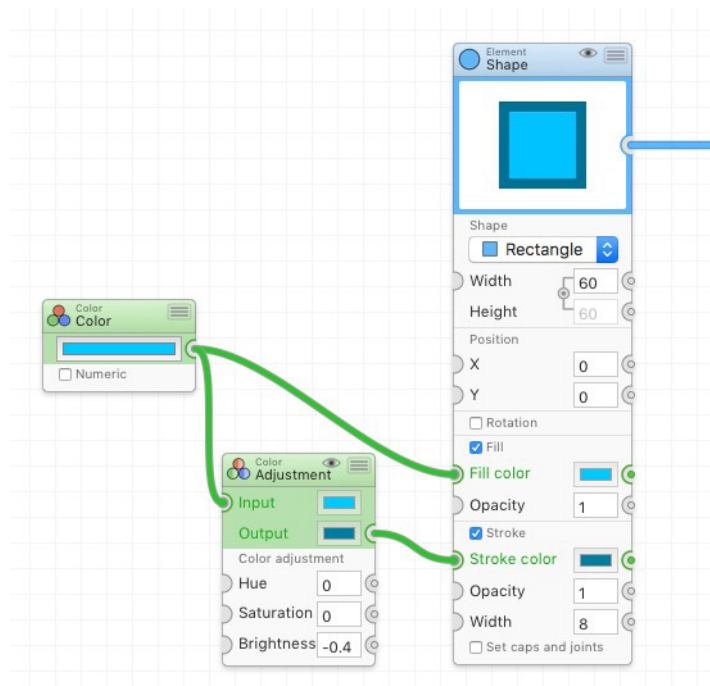
Numeric value parameters are the most common and shown in a field where you can edit the value. When you hover the mouse over such a value field a slider and a stepper is shown next to the field, allowing you to quickly adjust the value. For some parameters this hover-interface also provides a quick reset-button to set the value to 0 or 1.

In addition to this, when the field is active you can also use the arrow keys to step the value up/down fixed increments. Or, if you have this enabled in the settings, you can use your mouse scroll wheel.

In addition to this numeric parameters for X/Y values, like offset, position or scale, in some nodes can be edited by selecting the node and clicking and dragging in the pattern preview.

When this is possible a little blue four-arrow icon will be shown next to the X and Y fields as well as a small icon in the top right corner of the pattern preview.





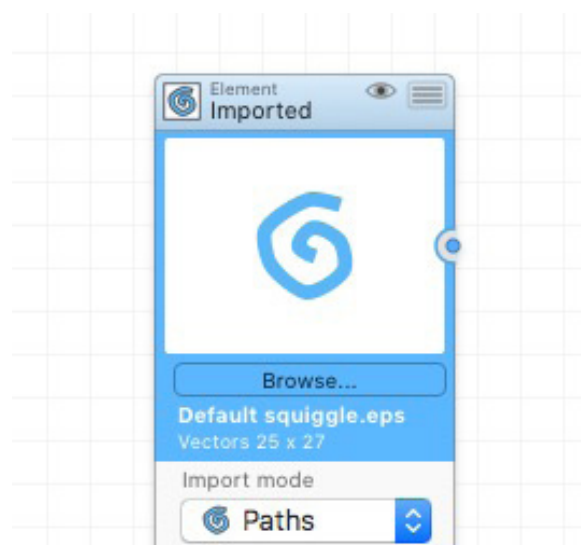
Color parameters

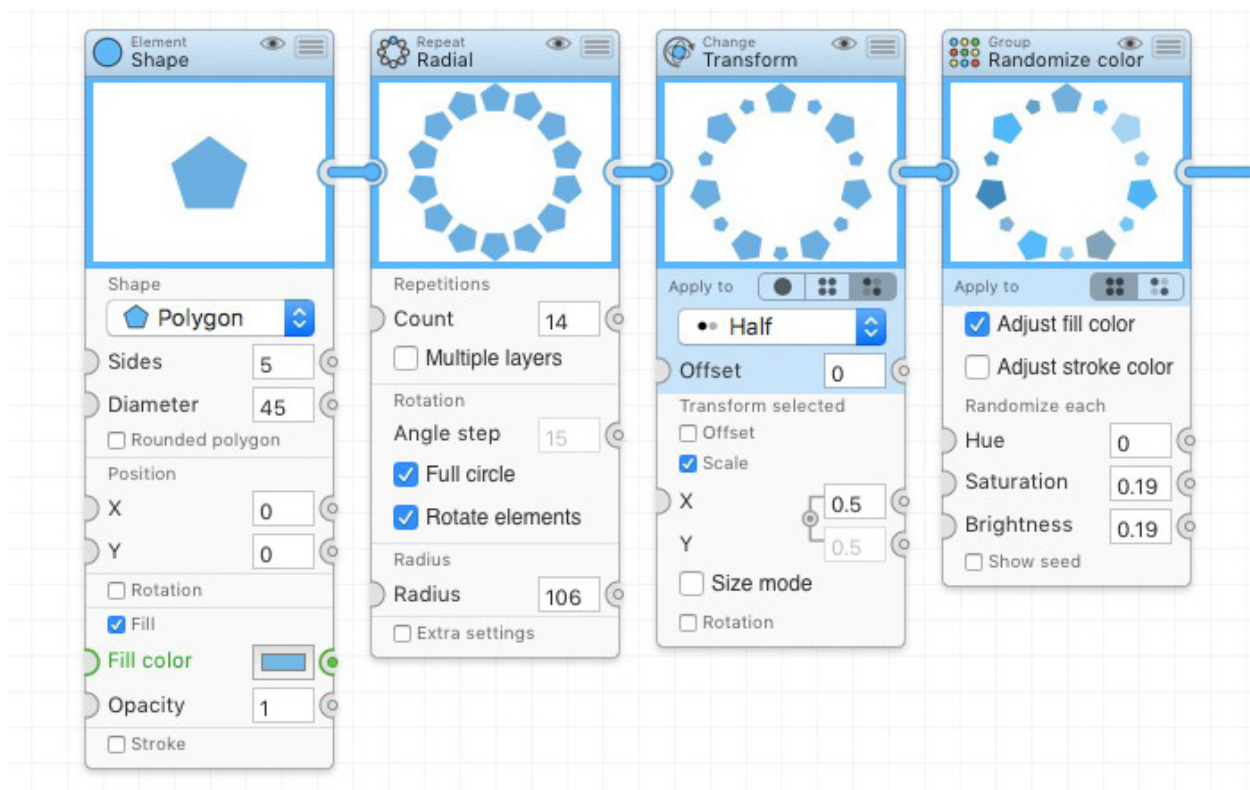
Color parameters are represented by a color well that shows the current color. Clicking it brings up a color picker interface where you can select a new color.

Image/file graphics parameters

Imported element node also let you specify an image file to use. In this case there is a browse button that lets you select a file to use. You can also simply select the node and paste in an image or graphic that you've copied in another application.

When pasting in vector graphics, make sure you paste them in from a document that is in RGB color mode for best results. PDF and EPS format vector data is supported, and SVG partially.





This manual will not in detail describe all the different individual nodes, but below are descriptions of the different types of nodes.

For more specific information about nodes and parameters you can read the tool tips in the application interface and use the built-in node reference list available from the Help menu.

2.3 Element nodes

These nodes create one or more **graphical elements**, the building blocks of the design or pattern that will be output, like vector shapes, lines or images. Since these nodes are the only ones that add new content they are usually the starting point of the graphical node sequence.

The graphics these nodes add are either defined completely defined by the node parameters (like a shape or text) or by importing pasted or file data from another application describing an image or vector graphics shape.

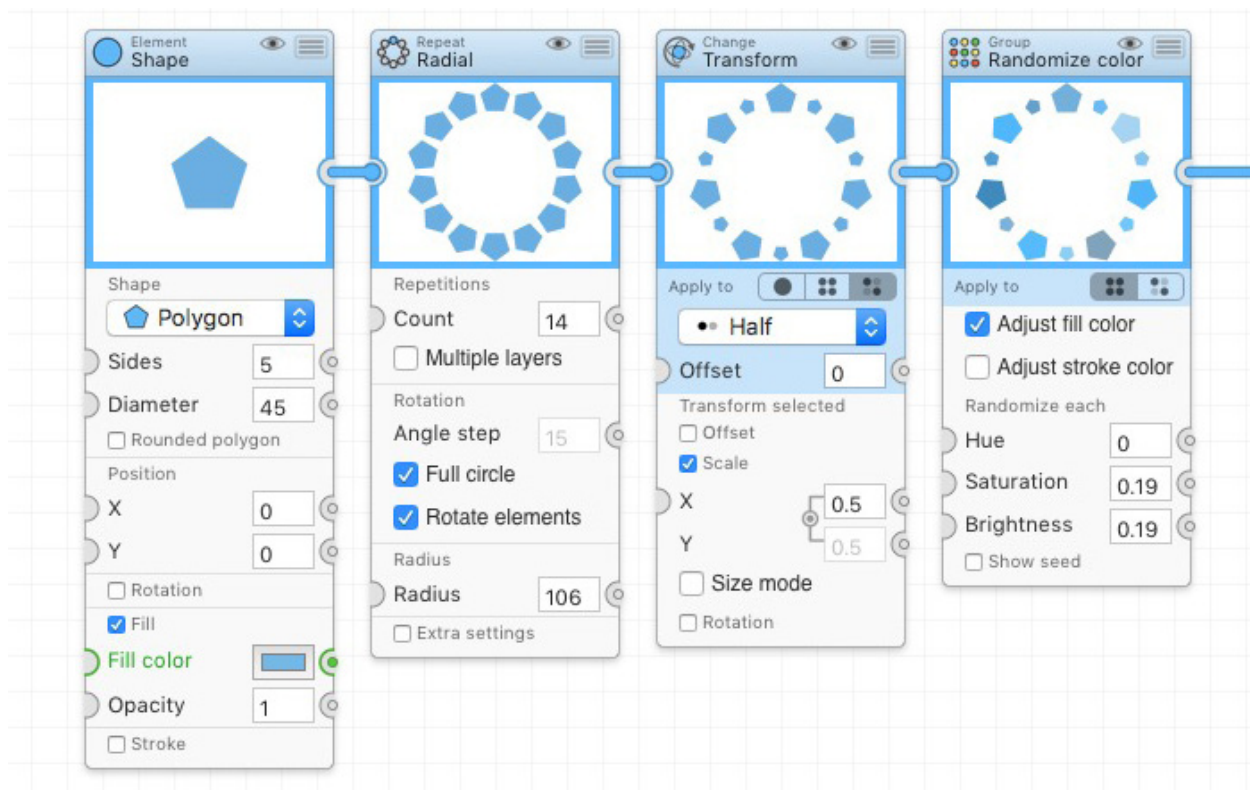
Apart from the different parameters describing the different element node shapes the element nodes have the following parameters in common:

Position (X,Y) - This represents the center position of the added element in the artboard. By selecting the node you can click and drag in the output preview to change this.

Rotation - An optional initial rotation of the element.

Fill - An optional fill color and fill opacity of the element. In imported image elements this can be used to tint the imported bitmap image using a color.

Stroke - An optional stroke color and stroke width and stroke opacity of the element.



2.4 Repeater nodes

Repeater nodes represent a repetition of the input graphical elements, usually making a series of altered copies of the original input in some pattern and outputting this group of elements. So if for example the input is a single shape, the output will be a group with multiple copies of this shape.

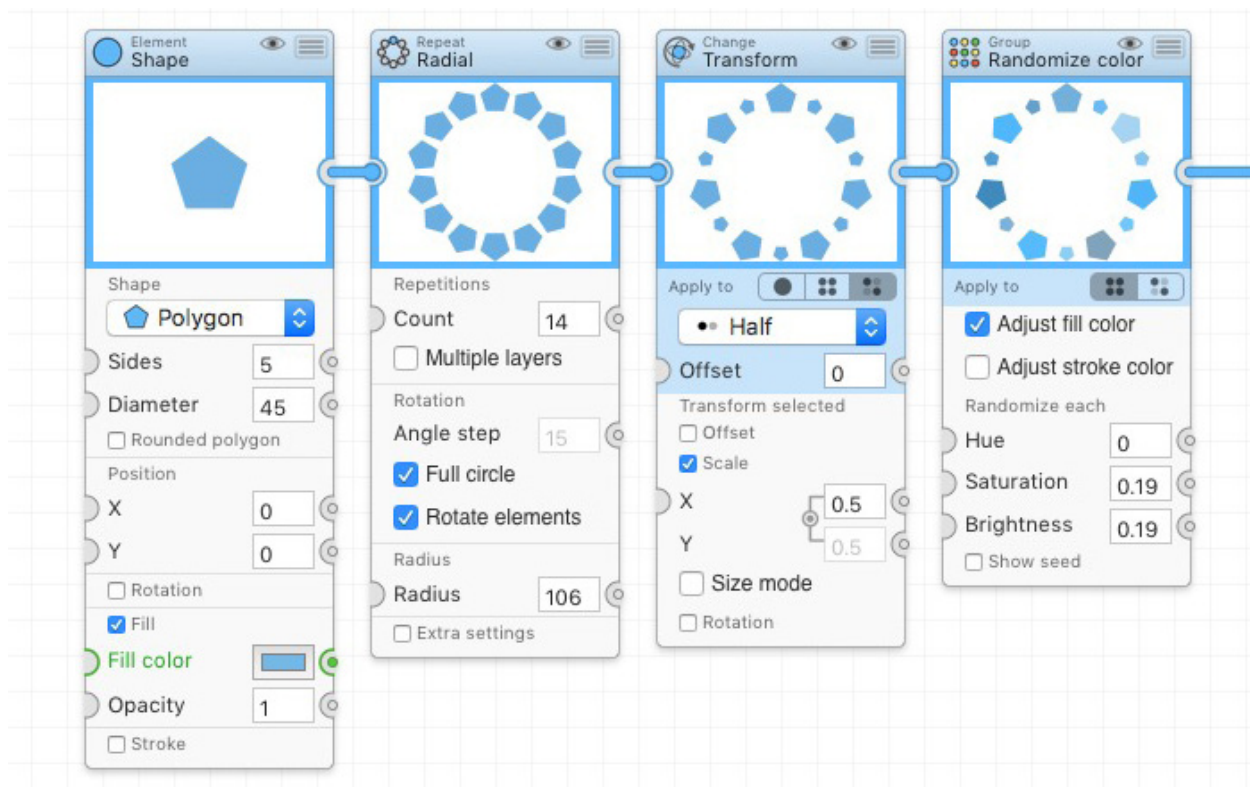
While in the example above the input of the repetition is a single shape, the input can just as well be a group of elements, made by combining several elements connected to the input or by a previous repeater node, letting you make repetitions of repetitions just as easily.

The different repeater nodes repeat or copy the input in different patterns and offer different parameters to control the number of repetitions, spacing and other aspects of the pattern.

Dynamic elements mode

A basic analogy for repeater nodes is that they represent a sort of copy and paste operation, based on whatever is connected as the input. However, for some more advanced applications most repeater nodes also have a "Dynamic elements" option you can toggle on under the "Extra settings" heading.

When dynamic elements is on, the repeater node doesn't just make copies of the original input element and repeat them in the defined pattern. Instead, for each repetition (for example each position in a grid) the repeater nodes requests a new input element to place there. If the input element is defined with some dynamic properties, like having size or rotation be a random value, a different element will be generated for each repetition.



2.5 Change nodes

These nodes represent a change of graphical elements connected to it, without creating and new copies. For example moving, scaling, or changing the color of the elements.

Depending on the graphics connected most change nodes can be configured to apply the change (for example rotation) either to the connected elements as a group, each individual element in a group or just some selected elements in the group. This is done with the "Apply to" setting in the blue field below the node preview.

The first "Apply to" option represents treating the input as a single element.

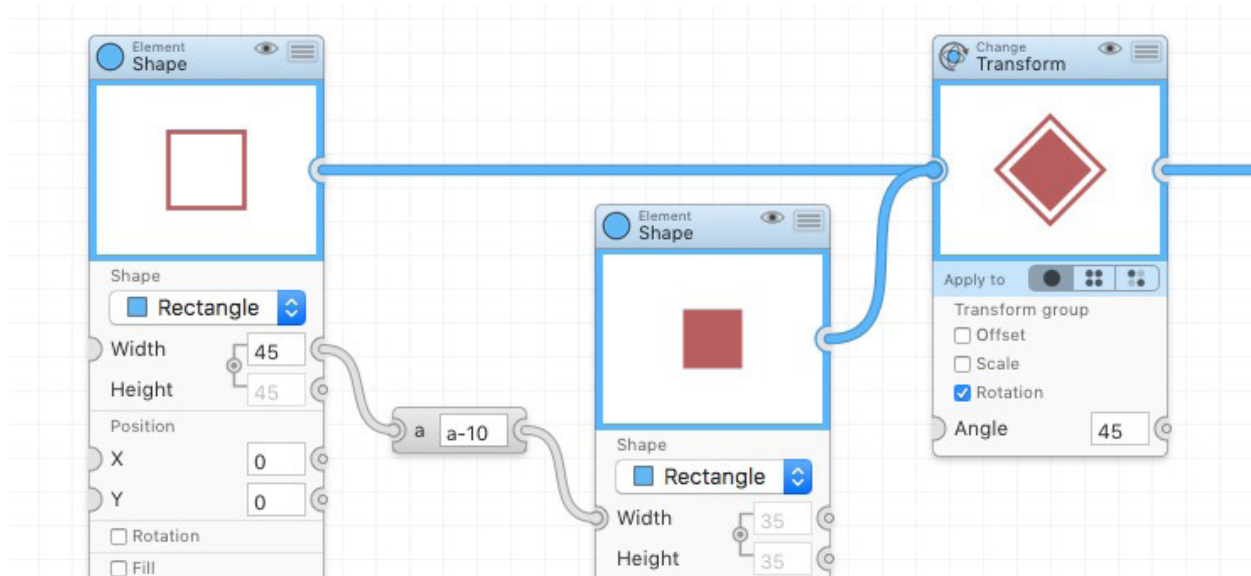
The second and third options are only available if the input is a group of elements and apply the change to each individual element in the group or just some selected individual elements. In these individual element modes the transformations (like scale or rotation) are done based on these individual element pivot points.

For example scaling down every other shape in the illustration above.

2.6 Group operation nodes

These group nodes represent some sort of operation on a group of elements. This might for example be randomizing the elements or applying some gradual change over the elements in the group. There are also some utility nodes of this type for selecting some elements in a group and merging/flattening groups or rearranging the order of the elements.

For example randomizing the color of the shapes the illustration above.



2.7 Value nodes

Nodes representing a numeric value or operations on such values.

These can be useful for setting up values you want to reuse in multiple nodes and for creating relationships between parameters in different nodes. So that if you change one the other changes in the correct way.

To generate basic patterns you seldom need to use value nodes, but if you want to do something more complex or unlock the more powerful capabilities of Patternodes, they are key. For example the random value node, group index value node and value list node can combined with the math nodes let you set up very complex relationships between different elements and repetitions. This is especially useful if you then want to start animating your graphics, which is done by defining animation function for values (More information in section 5).

2.8 Color nodes

Nodes representing a color value, or operations on colors such as adjusting the hue of a color value. This again lets you set up relationships between elements, like saying that one should have a lighter version of another's fill color. Or the same stroke color as another's fill color etc.

2.9 Other nodes

Finally, there are a couple of special nodes that do not fit into any other categories above.

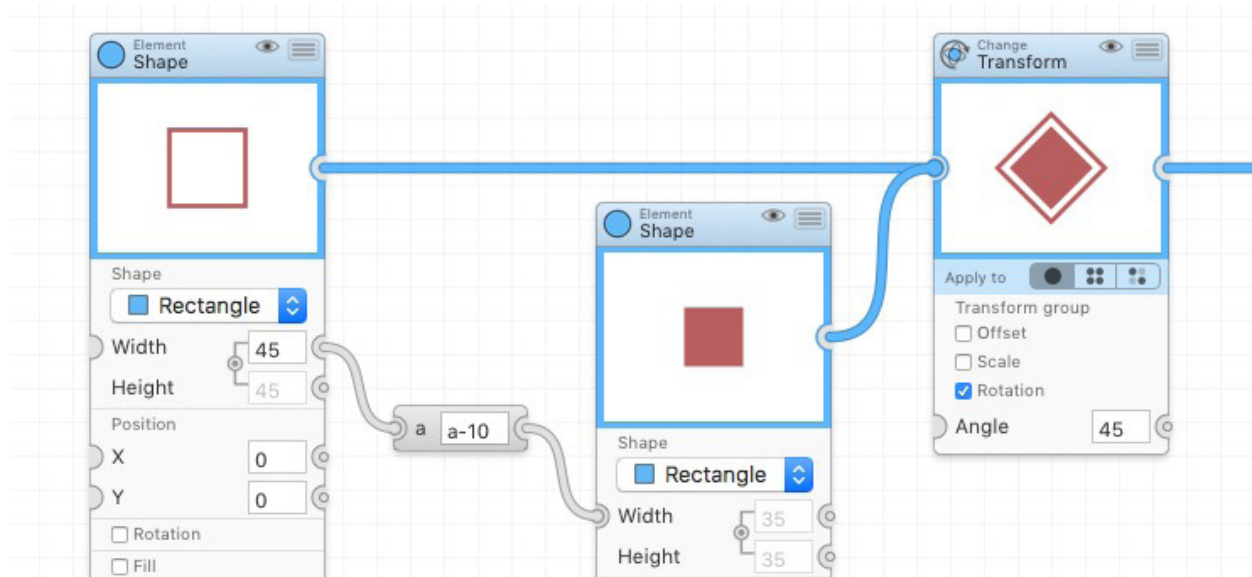
The Artboard output node. This node defines the artboard size and all graphics must in the end be connected to this node's output layers to be added to the output.

The Animation info node. This node is automatically added when you add animated values and let you set up duration and frame rate for the animation. It also provides interface for playing and pausing and changing the animation preview time.

Note nodes. These convenience nodes doesn't affect the pattern, just let you add small text boxes in the node editor for comments or reminders.

3 Connections

The end result is defined by the way the nodes are connected to each other. By connecting nodes together in smart ways you can also make complex multi-step adjustments very easy to try out and tweak with a single value.



3.1 Sockets and connections

Input sockets and output sockets

Most parameters in the nodes have sockets that let you connect input values to set the parameter, or output sockets that let you connect their values to other nodes.

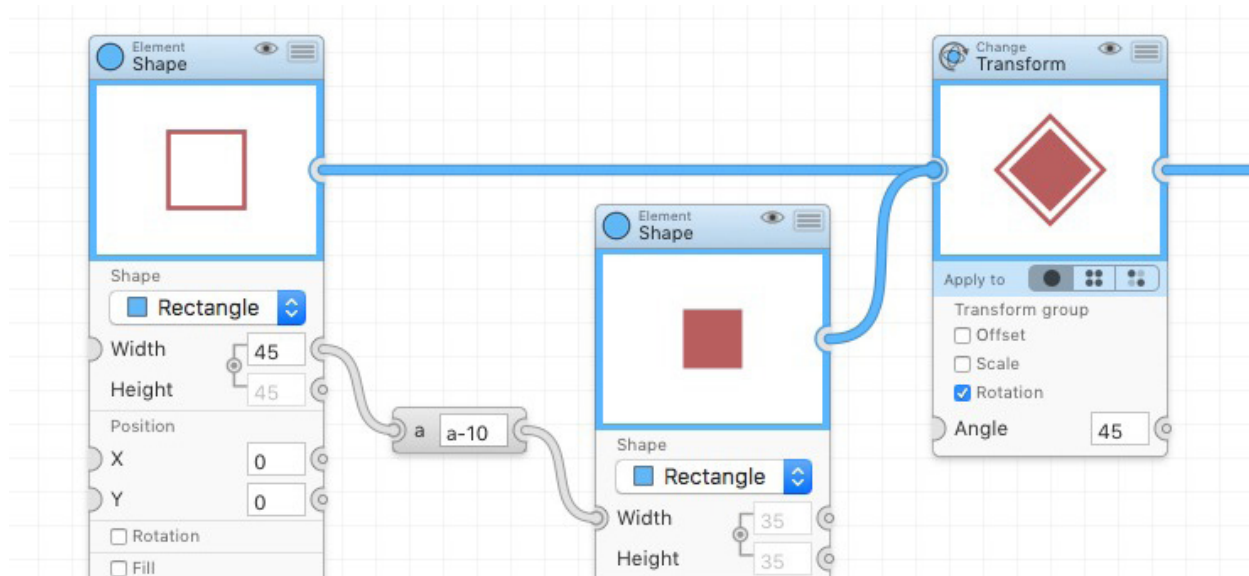
The sockets to the left are input sockets, and when something is connected to the input socket of a parameter it takes its value from this connection.

The sockets to the right are output sockets, letting you use the value of the parameter as input for another node.

3.2 Making and breaking connections

The basic way of making a new connection is simply to click and drag from a socket to another socket that can accept the socket. To break a connection, click and drag an existing connection end and let go of it away from the socket.

To make a new connection from a socket that already has connections, hold shift and click and drag. This will create a new connection instead of grabbing the existing connection end.



3.3 Connection types

There are three different types of connections, each with a different color. Sockets can only be connected to sockets of the same type.

Graphics connections (Blue)

The graphics connections control the sequence flow of how the pattern elements are transformed and repeated.

Graphics sockets can have multiple connections connected to the same output socket (in effect making multiple copies of the element).

Graphics sockets can also have multiple connections connected to a single input socket (in effect combining multiple elements to a group). In this case the order matters, with the order of the graphical elements being added in the same order as the connections. You can change the order disconnecting and then reconnecting the graphics in the order you want them, or use the context menu Connections -> Reverse order option.

Numeric value connections (Gray)

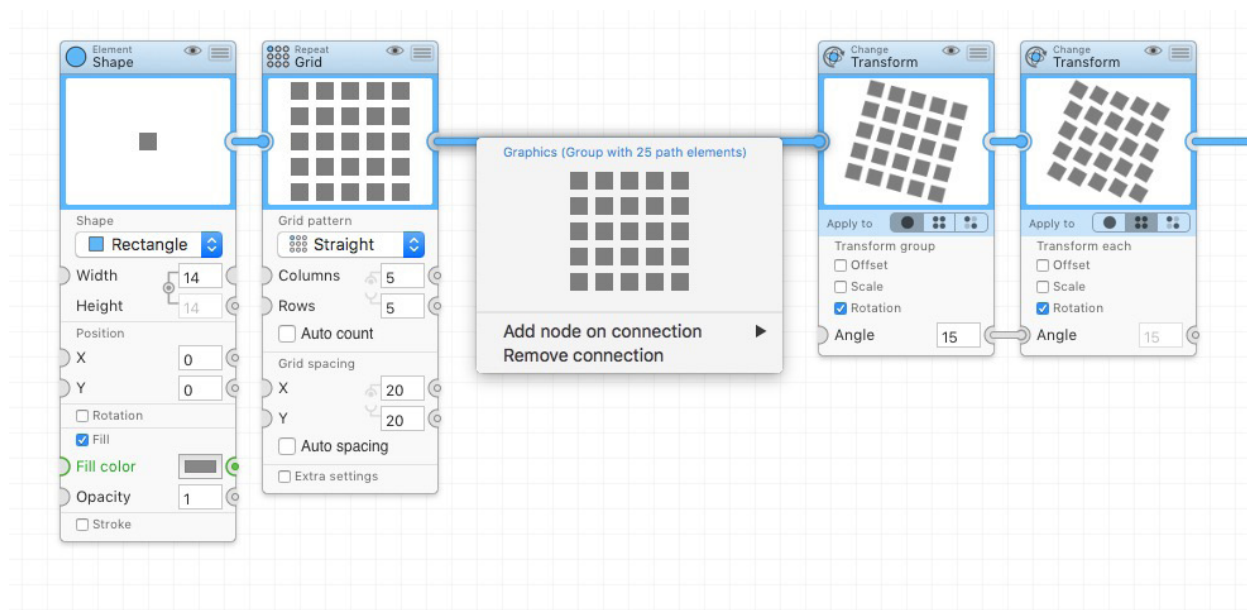
Numeric value connections lets a node take a numeric value for a parameter from another node. In some cases the value may not be possible to use in the node (for example negative values for repetition counts), which will lead to a warning and a changed value.

A value output socket can have multiple connections, to use the same value for multiple other parameters, but only one connection can be connected to a single value input socket.

Color connections (Green)

Similar to the numeric value connections these connections represent colors, and can be used to share and modify the colors parameters of different nodes.

A color output socket can have multiple connections, to use the same color for multiple other parameters, but only one connection can be connected to a single color input socket.



3.4 Graphics connection groups

This section discusses some of the more advanced aspects of how the graphics connections work when multiple elements are combined or repeated.

Single elements

In the basic case the graphics connections represent just a single element in the pattern, either a vector shape or an image. In this case it's easy to see how this element (defined in an element node) is then transformed and modified (for example scaled or rotated) before being added to a layer in the Artboard node.

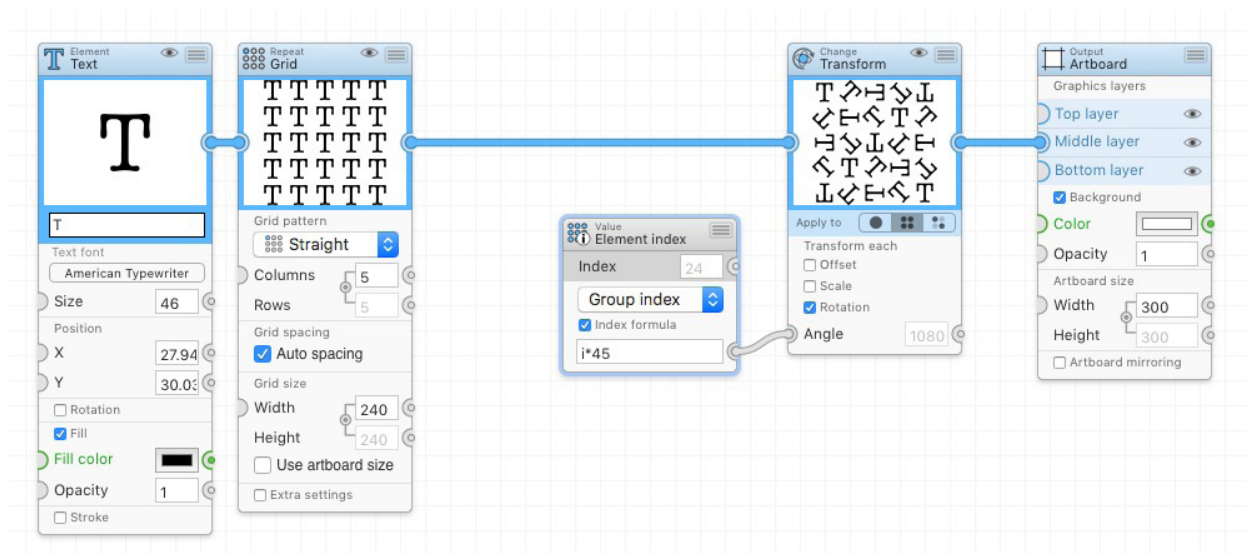
Groups of elements

When an element is repeated (by a repeater node) or several different graphics connections are connected to the same graphics input of a node, these elements are joined together into a group of elements. You can think of this as grouping the elements in Illustrator.

When connected to a Change node you can configure this node to either act on the entire group or each individual element. For example randomly rotating or scaling the elements in a grid, each around its own pivot point. These are some of the most powerful features of Patternodes. Especially as you can link the node parameters of nodes to the node index value node, to base the values on the index of the elements being transformed (see 3.5 for more).

Groups of groups

Just like groups in Illustrator, you can have multiple levels of groups. Each time something is repeated or multiple graphics connections are combined in a node a new level of "grouping" is added. For Change nodes this "groups within a group" are then what is acted on if the change node is set to Each or Selected mode.



3.5 Connection requests and advanced use

This section discusses some of the more advanced aspects of how connections and the generation process works to enable use of random or index based values.

Connection request model

The node sequence can be understood in simple terms as a sequence from start at an Element node to finish in the Artboard node. However, to understand and use some of the most powerful features in patternodes, a deeper understanding is required.

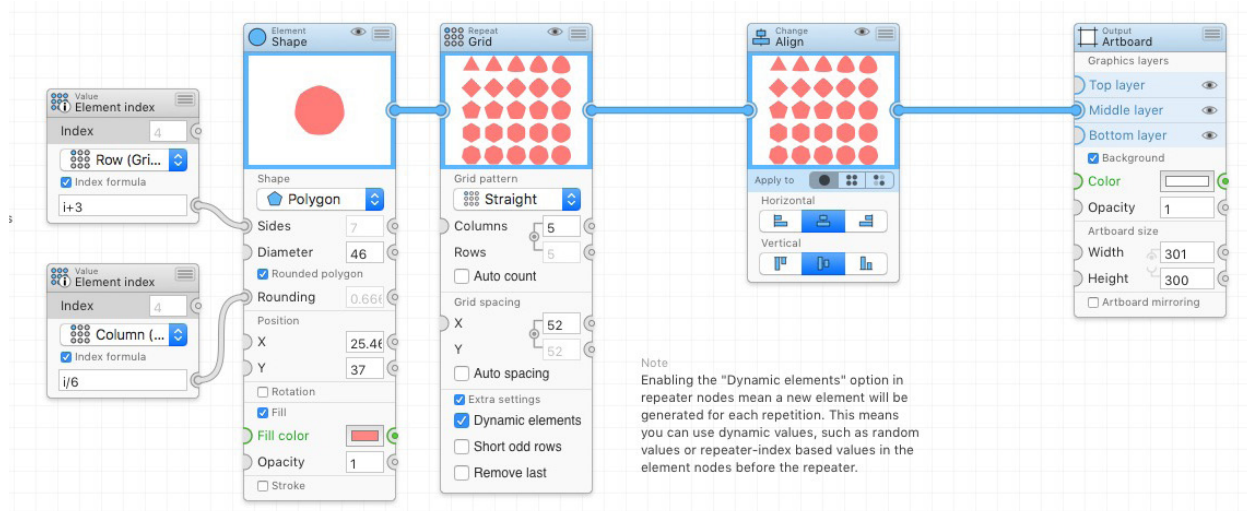
A technically more accurate way of describing the process of generating the output is that it actually goes the other way: Starting with a request from the Artboard node, which requests input from the previous connected node in the sequence. This node in turn sends requests for input from the connected nodes before it to generate its output. For simple non-branching sequences this makes no difference, but for a branching node sequence it can help you understand the process better.

Random values and the element indexes

This request models is especially helpful to understand how you can use random values and values based on the index of an element in its group for parameters. The key is that in "Each" or "Selected" apply to mode, **change nodes** and **group nodes** will send a new request for connected values for each element. In doing this the current "Group index" will also be updated to be the index number of this element in the group.

This means that you can use the Element index value node and make a calculation based on the index like in the example above, or a random value node, and connect it to a parameter (like rotation above). And in this case the node will ask for a new rotation value for each element it processes in the group.

This lets you set up calculations where you can have parameters vary in random ways or be based on calculations involving the index of the element in its group.

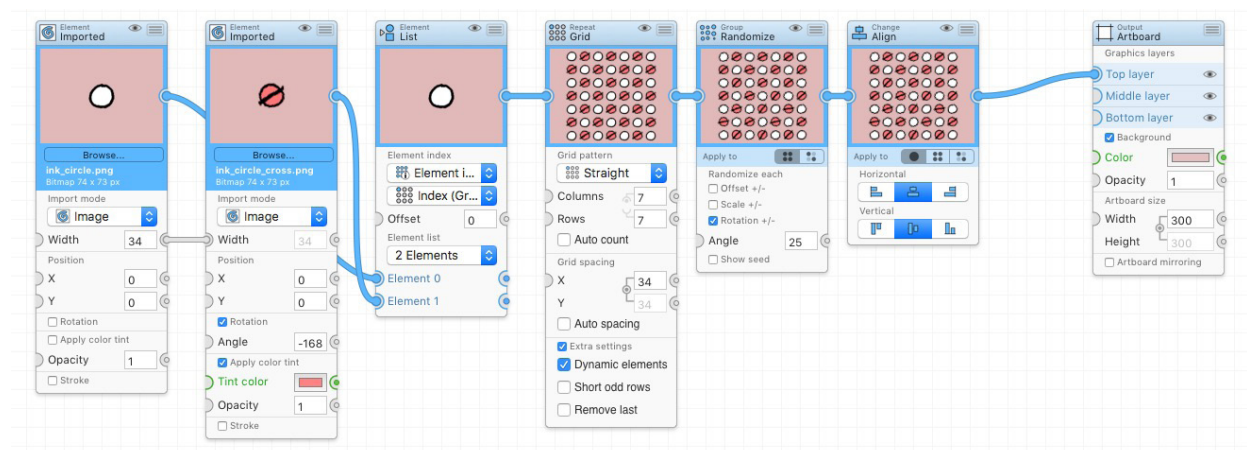


Dynamic repeater elements and repeater indexes

Similar to the Group index value used by group and change nodes, repeaters can give you access to different repeater indexes when they have *Dynamic elements* enabled.

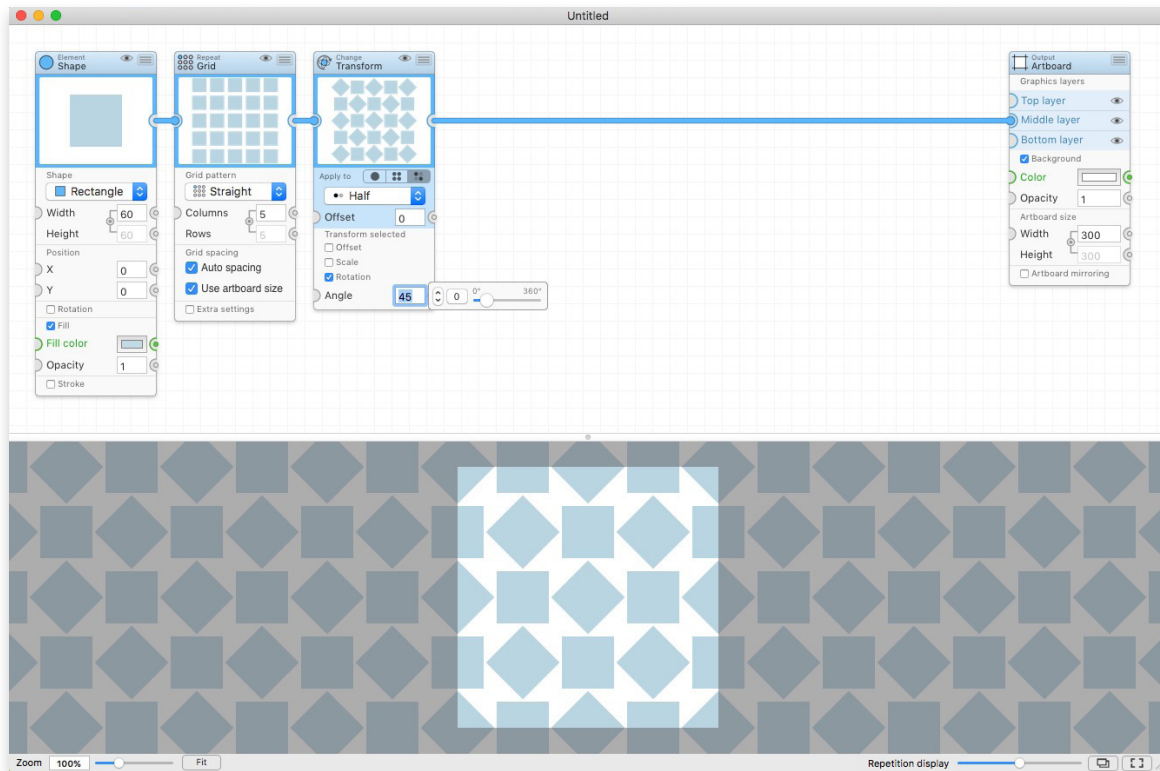
In normal mode a repeater just takes the input element and copies it in the repeater pattern. But with dynamic elements enabled the repeater actually requests a new element for each repetition. In doing this it also updates some indexes for this repeater, for example row, column and index for the Grid repeater. This lets you use either these types of index values or random values to control element parameters in a repeated pattern.

Besides the Element index node that lets you access these indexes, the Value list, Color list and Element list nodes all have built-in index modes that let you output values, colors or elements based directly on a selected index.



4 Output preview

At the bottom of the document window the output preview view shows a live updated preview of the current output.



4.1 Output display scaling

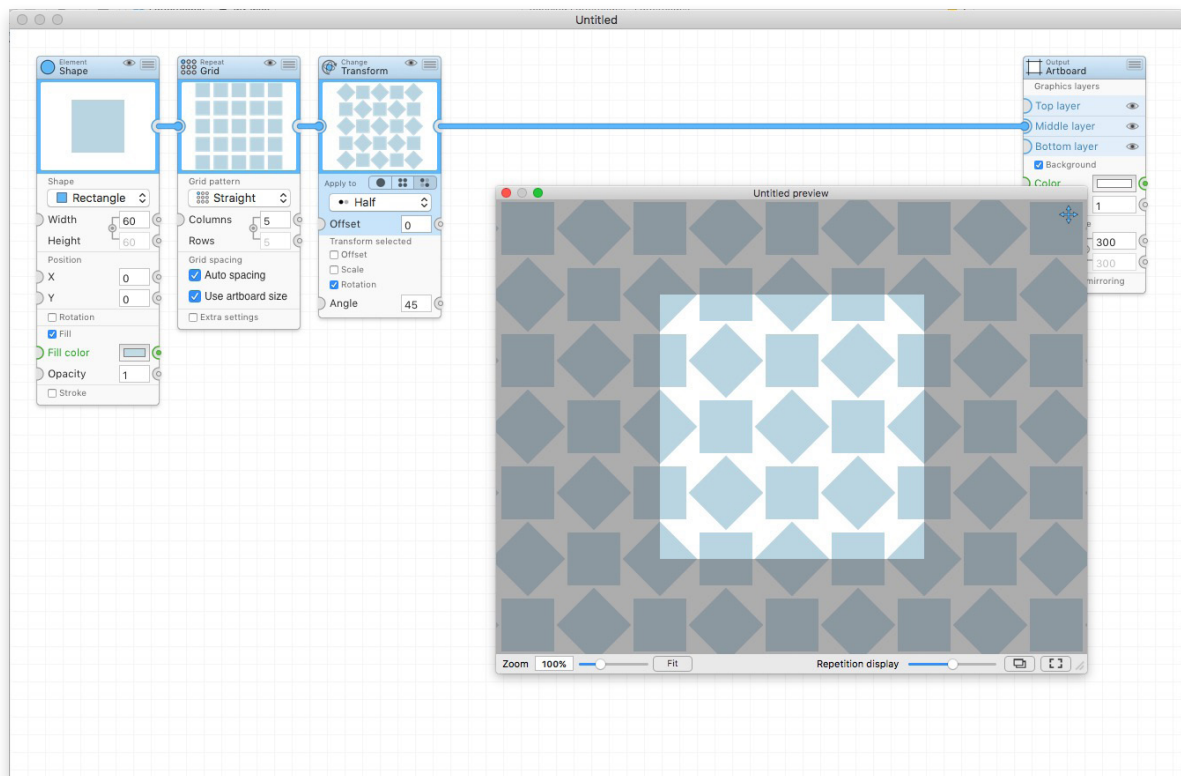
To the bottom left there is a slider that lets you quickly scale the pattern up and down to see how it looks larger or smaller. This display scaling can also be used in the exported results.

4.2 Repetition display

At the bottom right there is a second slider that lets you control the repetition display, changing the display of the repeated pattern artboard from full opacity to no opacity. This is to let you see either the individual tile to see where the edges are, or to see it repeated (hopefully seamlessly).

4.3 Full screen preview

At the bottom right of the preview panel there is a full screen button, pressing this (or using the corresponding menu item Command-F) instantly brings up a full screen display of the pattern, hiding all other interface elements to let you quickly evaluate the pattern on a larger area.

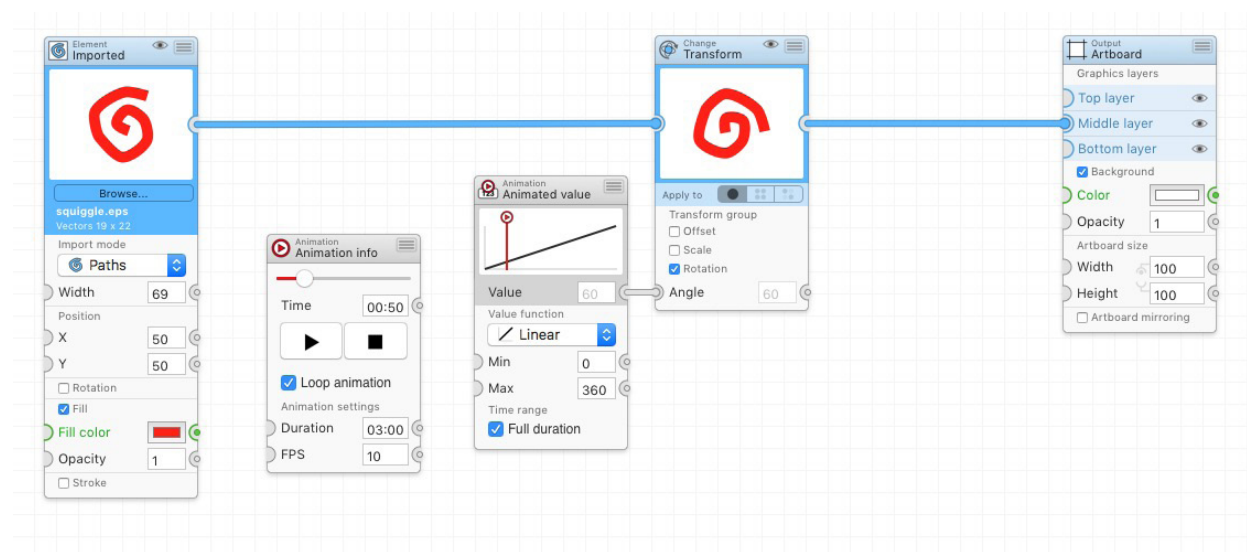


4.4 Separate preview panel

Next to the full screen button there is a button to toggle the preview into a separate window panel. This can give you more space in the main node editor view, and is especially useful if you have multiple monitors, as you can move the separate preview panel to another monitor.

Using the full screen preview (4.3) shows the full screen preview on the monitor the preview panel is currently on. So by moving the separate preview panel to a second monitor you can also enable the full screen preview on the entire second monitor while you work with the node editor view.

5 Animation



5.1 Animation control

Like most other aspects of your document the animation is controlled by nodes. The main animation controller is the Animation info node. This sets the duration and frame rate of the animation, and also lets you preview play/pause the animation and control the current animation time.

Technically you could just take the time value from the Animation info node and use it to change some parameter as it changes during the animation. However, in most cases an animated value node is easier to use.

5.2 Animated values

While you can only have one Animation info node to control the animation you can have any number of Animated value nodes. Each uses the current animation time (which will vary over the animation) and translates in using a function you define. You can then connect this value to whatever node parameter you want to animate.

For example you could have an animated value that changes between 0 and 360 and connect this to a rotation angle to have an element rotate one rotation over the animation.

5.3 Animation export

When you have your animation set up you can export it either to an animation format (gif, mov or mpeg4) or to numbered files for import in some other system (png, tiff, or jpg).

You can also export the animation to numbered pdf files, which may be useful if you want to generate multiple randomized variations of some vector artwork for example.

6 Saving and exporting

When you're done working with your pattern or illustration, you can easily save or export the pattern either as a pixel based image (jpg, png or tiff) or in a vector format (pdf or eps).



6.1 Saving Patternodes .pnds files

The internal Patternodes file format (.pnds) lets you save your node structure (and indirectly of course the resulting pattern or illustration) for later use.

All images or custom vector files are embedded in the saved file.

Additionally, the files also have embedded preview-data letting you see previews of the repeated pattern in Finder using Quick look.

6.2 Copying the pattern to the pasteboard

The easiest way to export if you want to keep working with the pattern in some other illustration software is to use the "Copy pattern vectors" item of the edit menu. This puts a vector representation of the pattern tile on the pasteboard, and you can just paste it into other software.

Note that this copied vector data will be in RGB-color, so if you paste it into a CMYK-color document (like the Illustrator standard) the colors will appear dulled down.

6.3 File export (bitmap or vectors)

Bitmap image file export

You can also chose "Export" in the file menu and save the pattern tile as an image file (JPG, TIFF, GIF or PNG). When doing this pixels will be used as units, so a 200x200 pattern tile will be 200x200 pixels.

For GIF export of images with transparent background you can also specify a Matte color. This will be used as a blending color for semi-transparent pixels to compensate the fact that GIF only supports 1-bit transparency. This will make the image look better on a specified background.

NOTE: Since bitmap images can't have half pixels, make sure your tile size doesn't, or you might have problems with the output not being seamless.

Vector image file export

You can also export your pattern to vector formats, either PDF or EPS. In these cases the units will be millimeters, so a 200x200 pattern tile will be 200x200 mm. PDF is usually by far the better option if you intend to use it in Apple or Adobe applications.

NOTE: EPS does not support transparency blending, so if you have transparency in your pattern the elements will be rasterized in the EPS file.

Export again feature

Once you've exported your pattern once you can use the "Export again" item in the file menu to quickly export the current pattern to the same file with the same format again, skipping the export dialogue. This can be especially useful when working with linked files in some other application, for example in web development.

WARNING: As the Export again feature skips the save dialogue it will not alert you when overwriting files.

Export animations

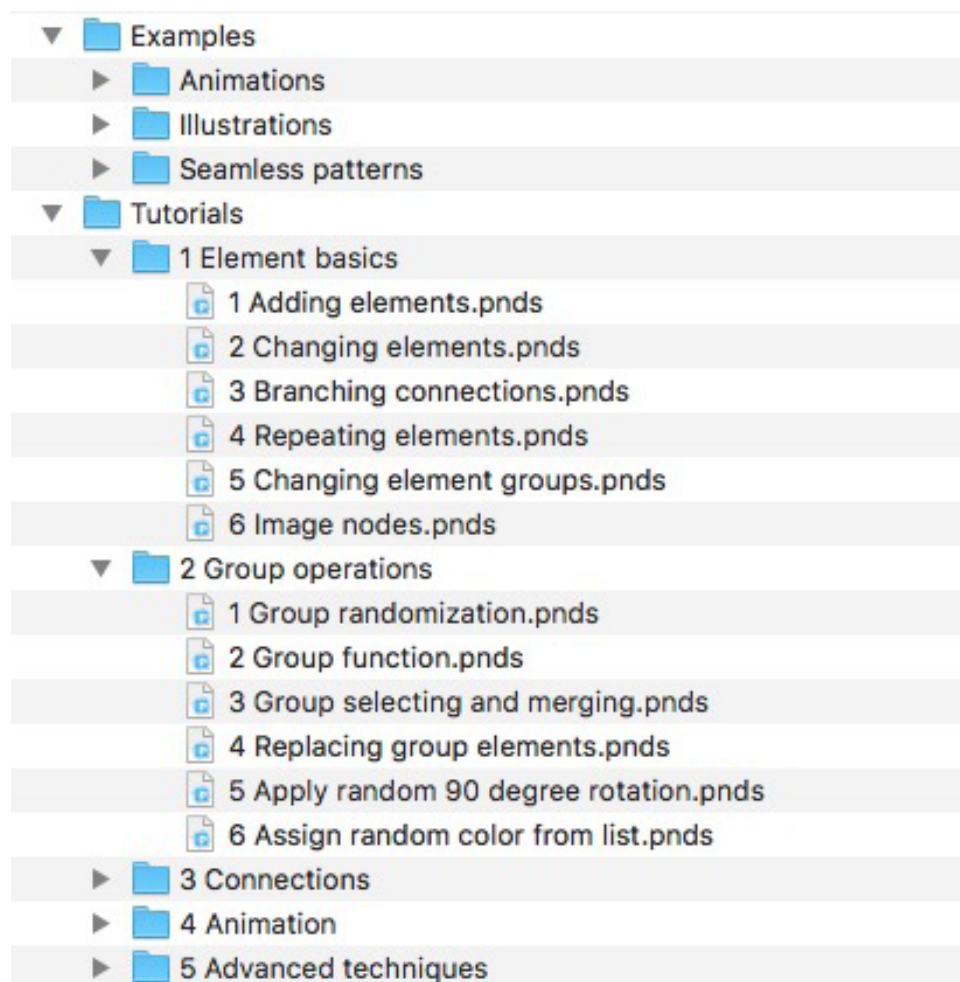
If you have an animation defined in your document (see section 5) you can also save the entire animation using the "Export animation" menu item. This lets you save the animation as numbered bitmap images (jpg or png), numbered pdf files or an animation GIF, Quicktime MOV or mpeg4 file. The settings for size and GIF matte color are the same as in single image export.

7 Sample files

Included in the Patternodes package is a large number of sample files to help you get started and understand how the software works.

Depending on how you like to do things you might want to start with the Tutorial files, with introductions to different features one by one, or you might want to look at the Example patterns, illustrations and animations to see how the software can be used to produce different results.

You can access these files from the Help menu



8 Buying a license

Patternodes is distributed as shareware. This means that you can download it and try it out for free, but with some limitations before you buy a license.



To remove the limitation, you buy a license for Patternodes online. This will give you a license code that you can use to unlock the full version. The code is entered by pressing the Enter license code button in the Demo mode window that shows each time you start up Patternodes in demo mode. The license entry dialogue can also be reached via the About Patternodes item in the Patternodes menu.

To buy Patternodes, go to lostminds.com/patternodes2

For schools and other educational institutions there is also a very good Educational site license offer, which lets you get Patternodes for all students and staff at your school at a very good price. More information can be found at lostminds.com/education

User licenses

The standard and upgrade licenses are personal to one user. In other words you can use the same license code on several computers as long as you are the only user. You may not share this license with other users, so if several people need to use the software you have to buy one license per user.

9 Distribution and credits

Patternodes is distributed as shareware, meaning you can download and try it out for free with some limitations. To unlock the full version you have to buy a license online (see the previous page).

For more information and to download the latest version go to: lostminds.com/patternodes2

If you wish to re-distribute Patternodes you can do so, as long as you do not alter the bundle in any way and try to make sure you distribute the latest version.

Review requests, feedback and bug reports

If you have a bug to report, a suggestion for a new feature or some other feedback, please use the "Send feedback..." option in the Help menu, as this will also give you access to some technical information that can be helpful in understanding what's going on.

Some common problems (like lost license codes) can be solved at lostminds.com/support

If you wish to review Patternodes for a site or magazine, please contact:
lars.gafvert@lostminds.com

Credits

Concept and development

- Lars Gäfvert

Math expression parsing code based on Expression

- Alex Nichol [<https://github.com/unixpickle/Expressions>] Copyright (c) 2014

Sparkle update code

- By the Sparkle Project, more information at <https://sparkle-project.org>

IJSVG svg rendering code

- By Curtis Hard / IconJar, more information at <https://github.com/IconJar/IJSVG/>

Beta testing, feedback and ideas

- Dimitry Chamy, Peter Danckwerts, Philippe Intraligi, Noah Katz, Jeff Karolski, Michal Macura, Alex McLaren, Roel De Meester, Kevin Mooney Jr, Matt Mower, Mac Myers, Bryan Rieger, Felix Petrescu, Stefan Schneller, Nico Silesi, Sebastiaan de Stigter

Version history

For full version history, go to lostminds.com/patternodes2 and click on Version history. Or select "Go to version history" in the Help menu in Patternodes.

Version 2.0.0 - 2018-03-26

Interface and usability improvements

- **Fewer more versatile nodes.** Many node types have been combined into more capable ones. This will let you do more with fewer nodes and make it easier to try new things and play around with the software without having to replace or add new nodes.
- **New artboard node with layers.** The old document and output nodes have been combined into a single Artboard output node with multiple output layers along with artboard size and settings.
- **Change node apply-modes.** Most of the nodes changing graphics (like changing color or transformation) now have built in support to toggle between applying their change to the group, each individual element in the group, or a selected subset of elements in the group. This replaces many array operation nodes and removes the need for a lot of splitting and combining you had to do in Patternodes 1.
- **Language and naming changes.** Many concepts have been renamed to be more in line with industry standards and/or common language to make them easier to understand. For example, the old Transform nodes are now called Change nodes and the old Array Elements are now simply called Groups.
- **Interface improvements.** Many small improvements to make editing node values and parameters quicker and easier to understand, like new context menus, improved connection editing and value editing hover controls.

Element creation

- **New Element Shape node,** combining the old circle, square, polygon and arc nodes and adding new parallelogram and trapezoids, as well as shortcuts for easily making specific rectangles and right-angle triangles.
- **New Element Import node,** combining the old Image and Custom Vectors node, making it easier to import graphical elements from file or the pasteboard and to configure how you want to use them.

Element change nodes

- **New Change Transform node,** combining the old rotate, scale, offset, transform and array transform nodes.
- **New Change Pivot node,** letting you quickly set the transformation pivot point of elements to specified edges or corners.
- **New Change Color and Change Opacity nodes,** combining the set color, set opacity and array adjust color nodes.

Element repeater nodes

- **Improved Make seamless node**, making it easier to make seamlessly repeating patterns tiles by automatically trimming and repeating edge elements.
- **Improved Repeat Grid node**, combining the old Grid and Alternating grid nodes and adding new shifted brick grids and flexible auto count mode to fill an area.

Color nodes

- **Parameter layout and naming changes**, to make the color nodes more consistent and easier to understand and use.
- **Built in random modes**, the Color List and Color Blend now have built in random-modes to make it easier to generate random colors from a list or in a gradient without extra nodes.

Value nodes and calculations

- **New Math node**, replacing many old value calculation nodes with a new generic interface for entering math expressions to be evaluated.
- **Joint Element info node**, offering value access to size, position and element count of a connected element.

Animation

- **New CoreVideo-based animation exporting**, with greatly improved memory management and parallel file writing, allowing for export of longer high-resolution animations to Quicktime MOV, MPEG4, GIF or jpg/png image files.
- **Numbered PDF animation exporting**, you can now export animations to numbered pdf files as well. Useful if you want to produce a large number of randomized versions of some vector output.

Other changes and improvements.

- You can now tap space to quickly play/pause animations.
- You can now use your mouse scroll wheel to quickly edit value fields.
- Improved Seamless lines node, with more options to set spacing, angle, count and calculating a seamless repeating artboard size.
- New value round node.
- Connection editing is now more consistent, dragging from an output socket always creates a new connection, existing connections can be grabbed from the end socket connection.
- More math functions, including acos, asin, round, ceil, floor and abs are now supported and properly evaluated when entered in any value field or math node expression.
- Patternodes is now self-contained with the manual and all sample files residing in the application bundle instead of as external files.
- Improved color matching on export, preview rendering and color picker eyedropper selection.
- Updated Sparkle software update features.
- Fixed element size calculations for clipped elements.
- Many more bug-fixes large and small.

For full version history please go to lostminds.com/patternodes2