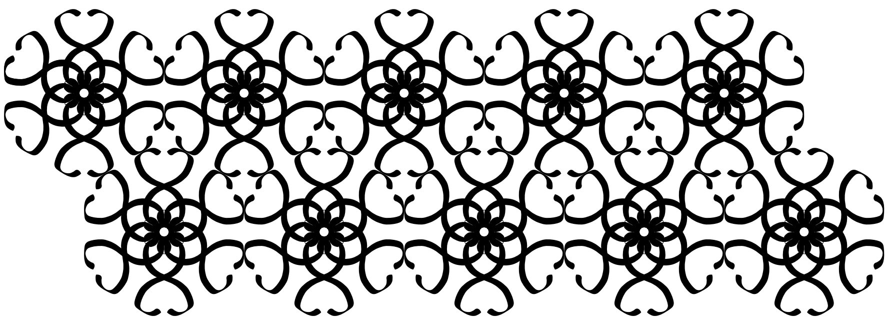
Chapter 14. Tiling

Tiling or tessellation is the covering of a surface with the repeated use of the same shape tile. A typical example is the tiling in a bathroom. In Inkscape, this concept is expanded to include a multitude of options including progressively changing the tile size, spacing, and orientation.

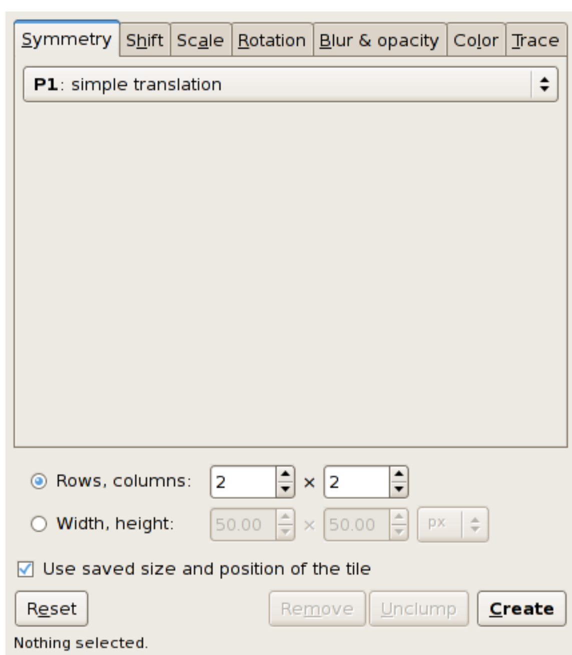
The tiles are in reality just clones of the source tile or object. Thus the same methods that apply to clones apply to tiles. (See the section called “Clones” in Chapter 5, *Editing Basics*.)

While random use of the *Tile Clones* dialog can produce exquisite patterns, it is useful to understand the fundamentals of tessellation in order to have more control over the final design.



An example of using the *Tile Clones* dialog with a simple calligraphic stroke and the *P6M* symmetry group (see text).

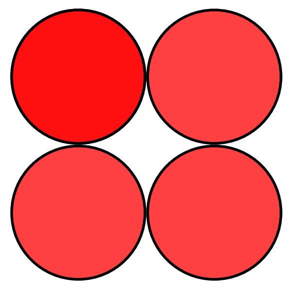
To construct a tiling, open up the *Create Tiled Clones* dialog (Edit → Clone → Create Tiled Clones...).



The Tile Clones dialog with no objects selected.

At the bottom of the dialog is a fixed section where you can choose the size of the tiling either by the number of rows and columns or by the width and height of the area you wish to cover. The terms *Rows* and *Columns* are only really appropriate for tiling of rectangular tiles (see below). Checking the “Used saved size and position of the tile” forces the tiling to use the size and position of the base tile at the last time the tile was used in a tiling. Clicking on the *Reset* button resets most of the entries under the tabs to their default values. The *Remove* button can be used to undo a tiling when the base tile is selected. The *Unclump* button can be used to spread out the clones in a somewhat random fashion (can be repeated). And, finally, the *Create* button creates the tiling.

With a circle and the default values (P1 symmetry, two rows and two columns), you will get the following tiling:



The simple tiling of a circle. The symmetry is "P1" and there are two rows and two columns.

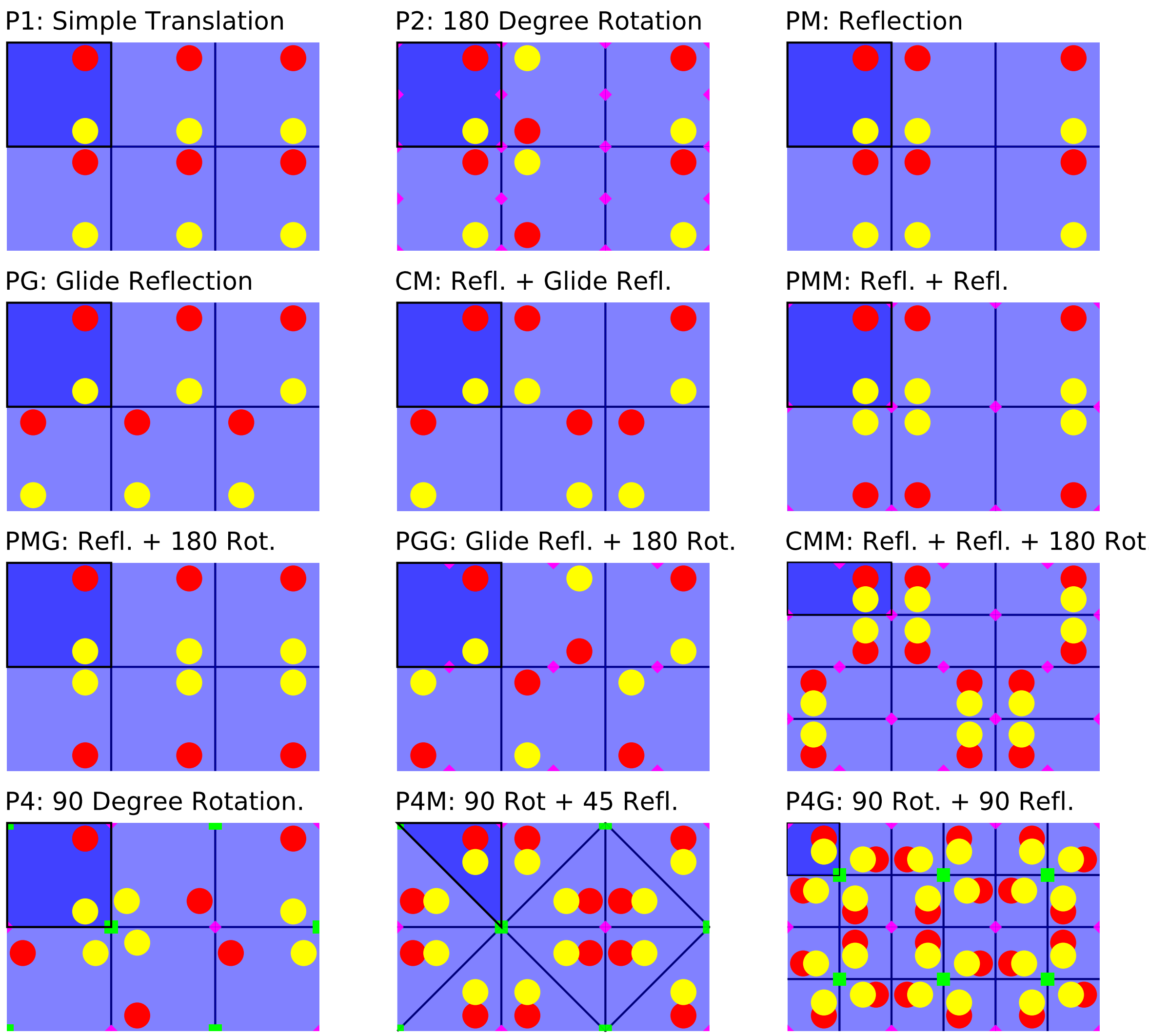
The circle has been replicated four times in two rows and two columns. The original circle is still there, under the top-left cloned circle. The *bounding box* of the circle has been used as the base tile size.

This example is not so interesting, but there are many options under the dialog's tabs that can produce many interesting effects. Each tab will be covered in turn in the following sections.

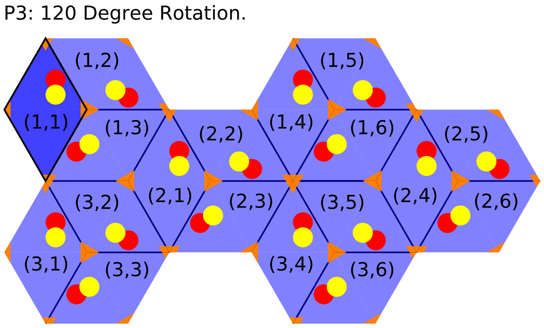
Symmetry Tab

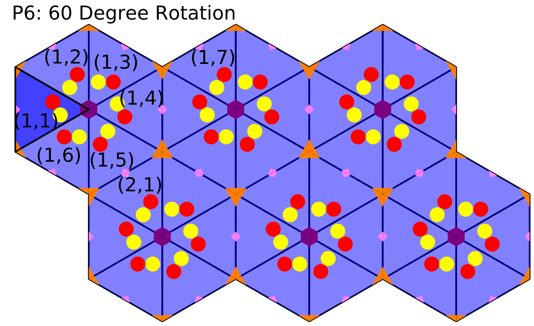
The *Symmetries* tab is at the heart of the tiling process. Understanding the different symmetries is necessary to have full control over the outcome of a tiling. The symmetry of the tiling is selected from the pull-down menu under the *Symmetries* tab (see above figure).

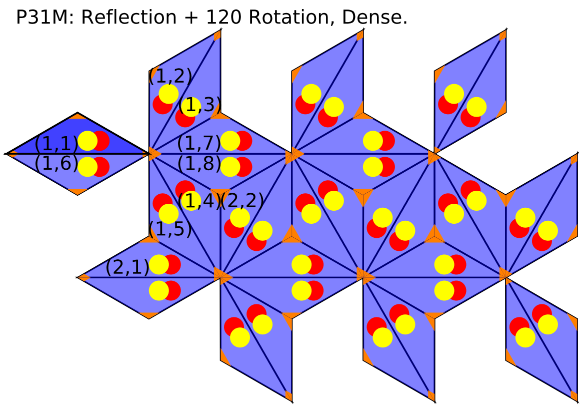
There are three regular geometric shapes that can be replicated to cover a surface completely (without gaps or overlaps). These shapes are: triangles, rectangles (parallelograms), and hexagons. A complete set of tiling symmetries requires taking these shapes and adding rotations and reflections. It is known that there are 17 such tiling symmetries. (See: Wikipedia entry [http://en.wikipedia.org/wiki/Wallpaper_group].) All 17 symmetries are included in the Inkscape *Create Tiled Clones* dialog. The symmetries are shown next.

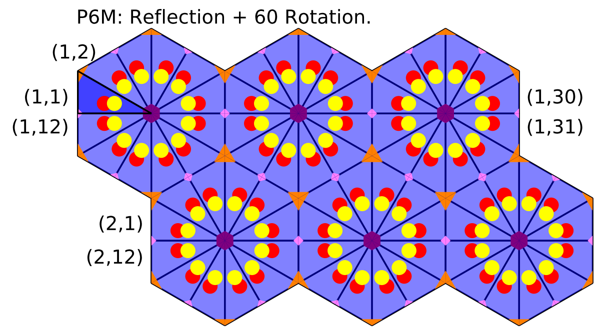


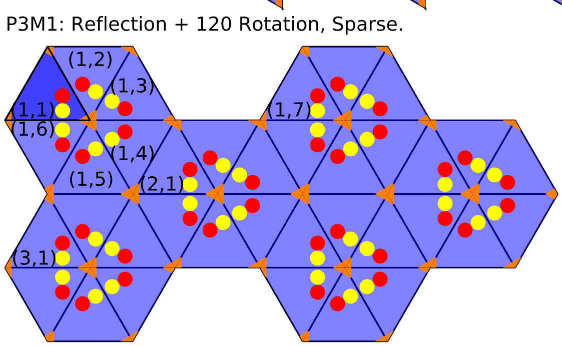
Tilings based on a rectangle tile (or 45-45-90 degree triangle). The outlined dark blue tile is the basic unit. Red and yellow dots show the reflection and rotation symmetries. Points of two-fold and four-fold rotational symmetry are shown by pink diamonds and green squares, respectively. The P1 and P2 symmetries also work with parallelograms.





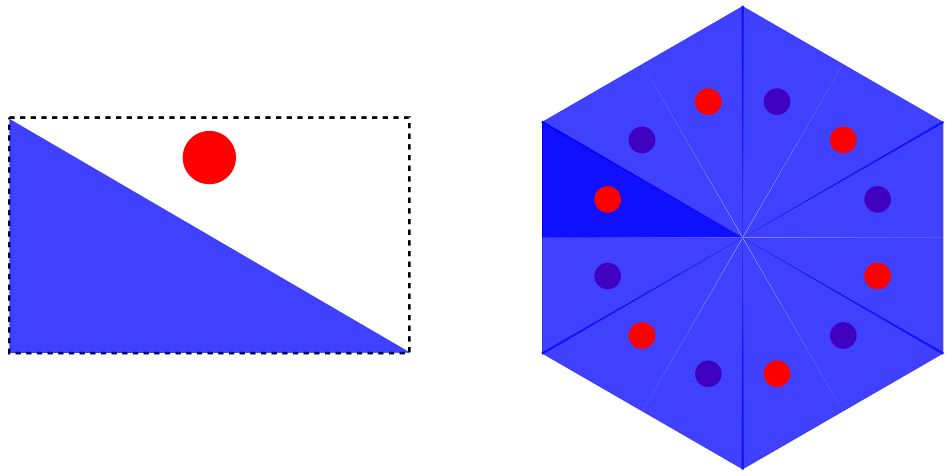






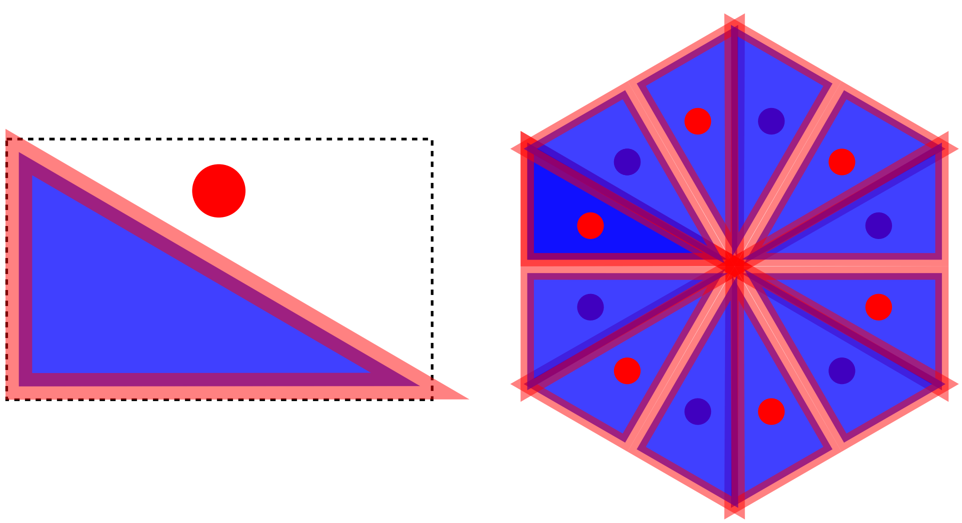
Tiling based on regular subdivisions of a hexagonal. The outlined dark blue tile is the basic unit. All tiling have points of three-fold rotational symmetry (orange triangles). Two also have two-fold and six-fold rotational symmetries (pink diamonds and purple hexagons). The pairs of numbers indicate the row and column numbers.

The basic tile for each of the 17 symmetries is shown in dark blue in the preceding figures. Inkscape uses the *bounding box* of an object to determine the basic tile size. For rectangular base tiles, the *bounding box* corresponds to the base tile. However, for triangular base tiles, the base tile covers only part of the *bounding box* area. This can result in tiles “overlapping” if an object extends outside the base tile shape (but is still within the *bounding box*) as in the tiling in the introduction to this chapter. Overlapping can also occur if the base tile is altered after the tiles are positioned.



On the left is a triangle and circle that are grouped together. The triangle corresponds to the base tile for a P6M symmetry. Note that the red circle is outside the base triangle but is still within the *bounding box* of the group (and triangle). On the right is a P6M tiling with the triangle and circle. Note how the red circle ends up above some but below other triangles as determined by the order in which the tiling is made.

Recall that when the *bounding box* is calculated, the width of any stroke is included. For rectangular tiles, this may not be a problem. But for triangular tiles, it can lead to undesired effects. In the example below, the *bounding box* is extended by the width of the stroke assuming a *Round Join* style. The *bounding box* no longer corresponds to that for a 30-60-90 degree triangle base tile. This leads to inconsistent overlaps between the tiles. Also, if the *Join* style is *Miter*, the stroked part of the path sticks out.

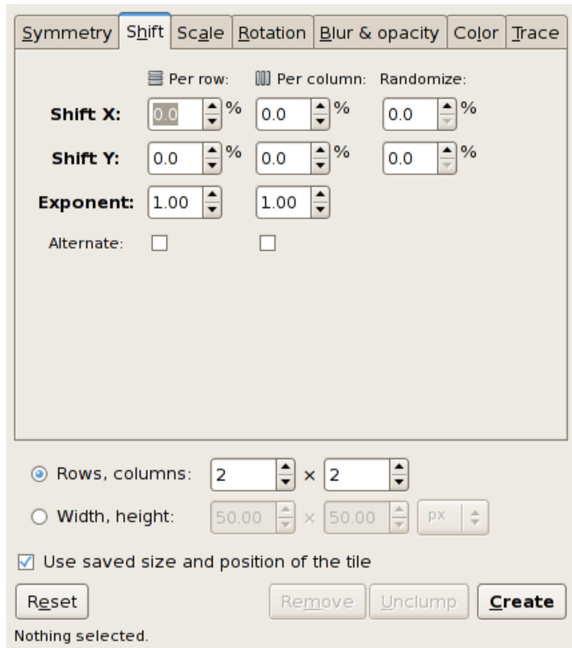


On the left is a 30-60-90 triangle with a wide red (partially transparent) stroke. The *bounding box* does not, however, correspond to a 30-60-90 degree triangle, which is required for tiling with the P6M symmetry. This results in inconsistent overlaps between the tiles as shown on the right. Note also how the mitered strokes stick out.

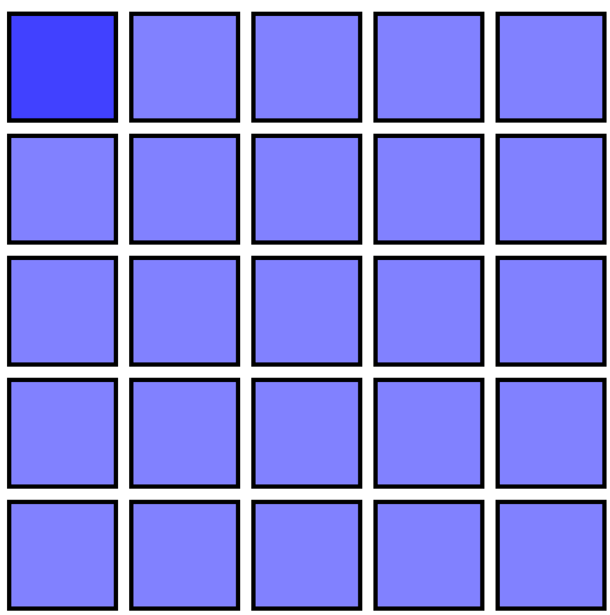
A solution to this problem is to define your tiles *without* a stroke and then add one to the base tile after tiling. Another solution is to edit the lines in the *XML* file with the *XML Editor* dialog that control the tile size: “inkscape:tile-h” and “inkscape:tile-w” (these will appear after you have cloned the object and are used only if the *Use saved size and position of the tile* button is checked).

Shift Tab

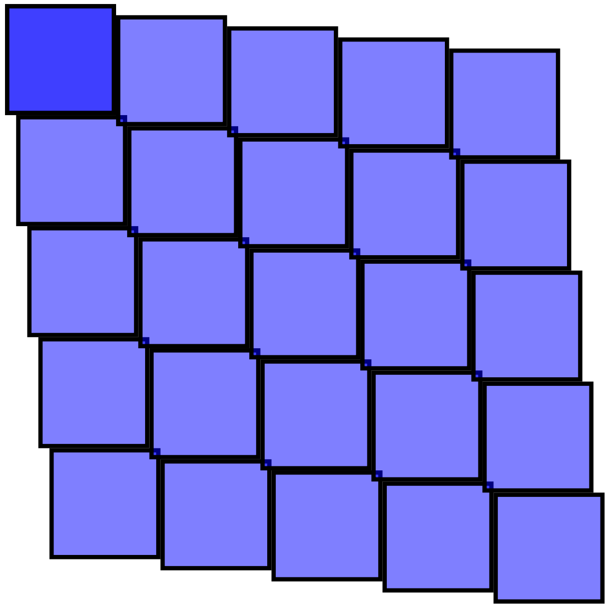
The *Shift* tab allows one to add or subtract space between the tiles. The amount of this shift is specified in a percentage of the width or height of the base tile *bounding box*. An *Exponent* factor z can be added to the shift. The position of each tile is then x (or y)= $(1 + \text{shift in percent})^z$. The shift can also *Alternate* between a positive and negative value and a *Randomizer* factor can be added.



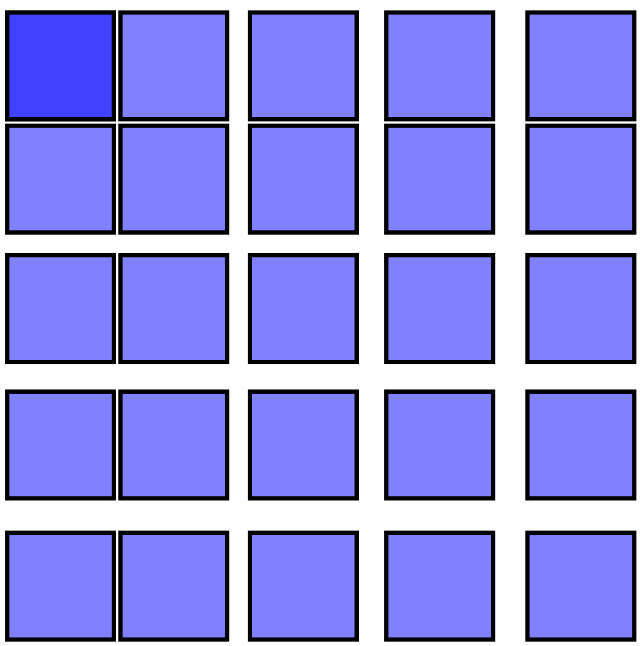
The Shift tab of the Tiling dialog.



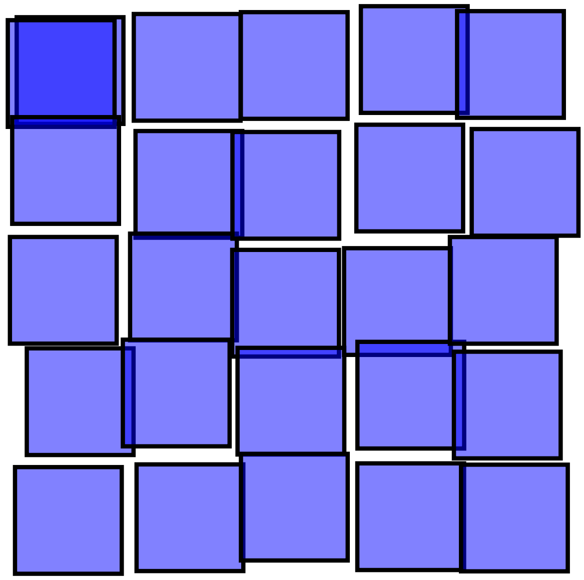
A P1 symmetry tiling with a constant shift of 10% (of the *bounding box*). There is an x shift for each column and a y shift for each row.



A P1 symmetry tiling with a constant shift of 10% (of the *bounding box*). There is a *y* shift for each column and an *x* shift for each row.

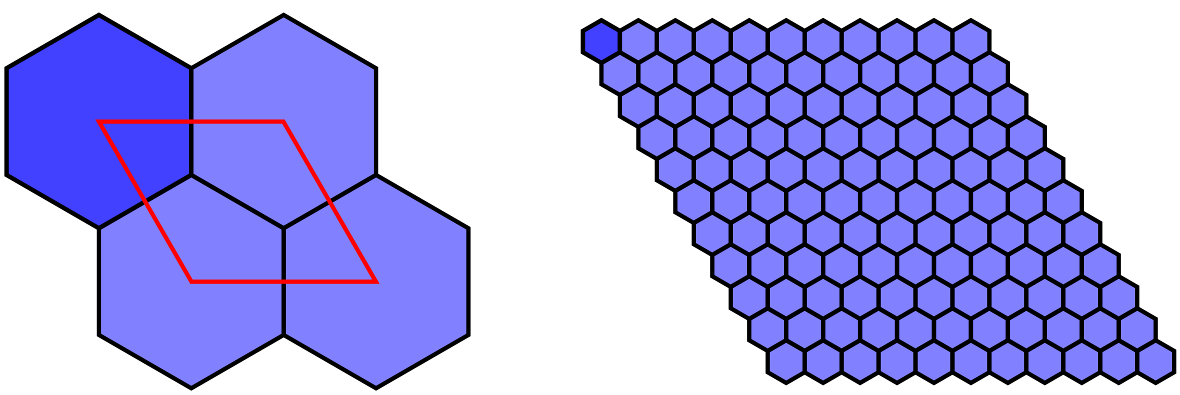


A P1 symmetry tiling with an exponential shift of 1.1 (2% shift in *x* and *y*).



A P1 symmetry tiling with a random shift of 10% (of the *bounding box*) in both *x* and *y*.

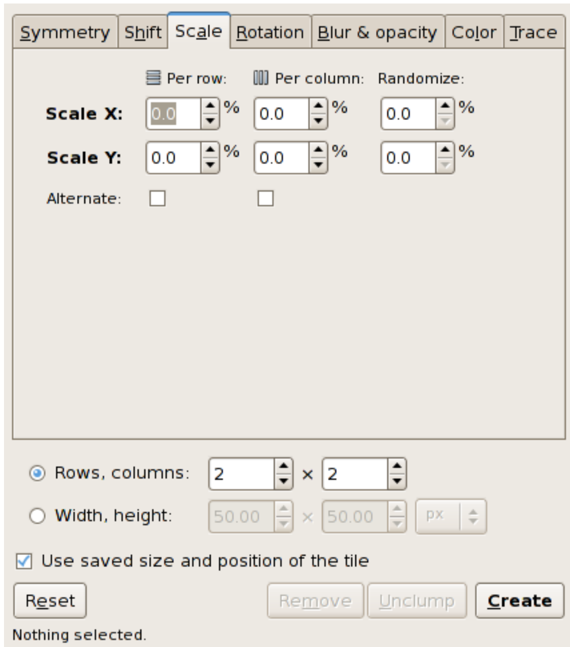
Question: What is the symmetry of closely packed hexagons? The answer is P1 as can be seen below. One can use this fact to trivially generate the board for the game Hex [http://en.wikipedia.org/wiki/Hex_%28board_game%29] invented independently by the mathematicians Piet Hein and John Nash.



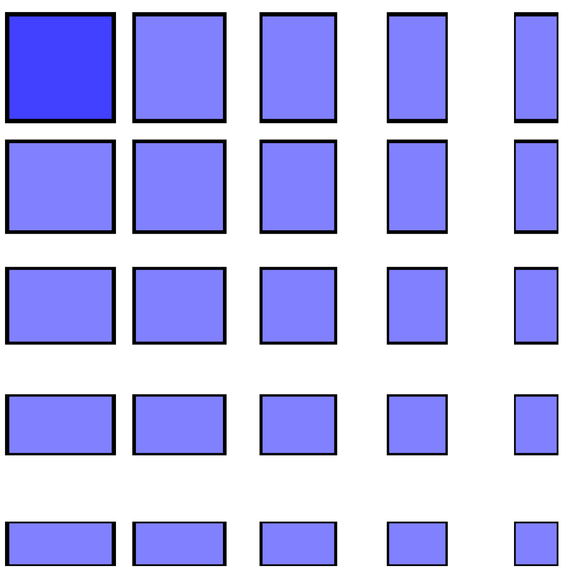
Closely packed hexagons have a P1 symmetry tiling as shown on the left. On the right is the board for the game Hex. To generate both tilings, a hexagon was tiled using a shift in *x* of 50% and a shift in *y* of -25% per row.

Scale Tab

The *Scale* tab allows one to increase or decrease the size of the tiles depending on the row and column position. The scaling is specified in a percentage of the width or height of the base tile *bounding box*. The scaling can be specified to *Alternate* between a positive and negative value. A *Randomizer* factor can also be specified.



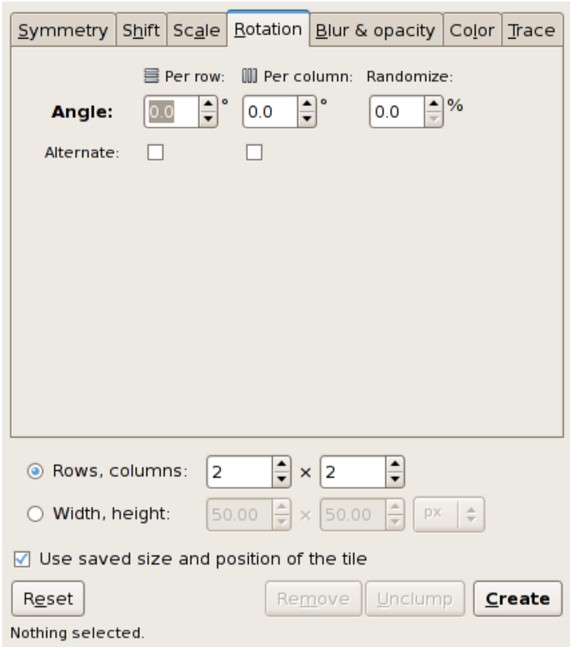
The Scale tab of the Tiling dialog.



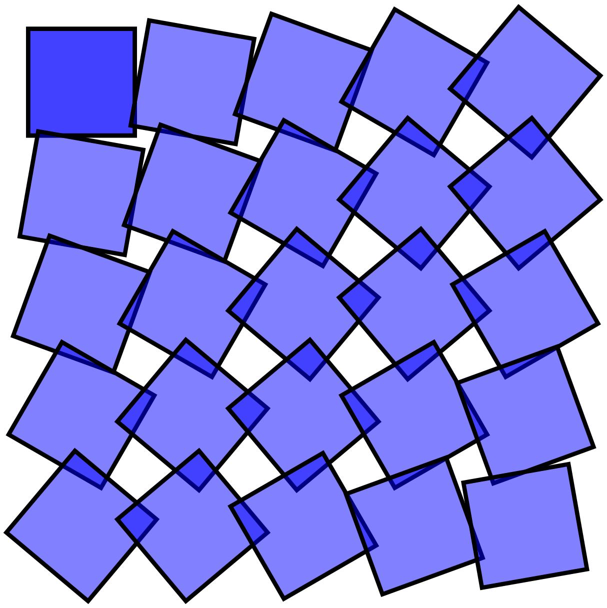
A P1 symmetry tiling with a negative scaling. There is an -15% *x* scaling for each column and a -15% *y* scaling for each row. The scaling is a percentage of the base tile *bounding box*. The spacing between the center of adjacent tiles remains fixed.

Rotation Tab

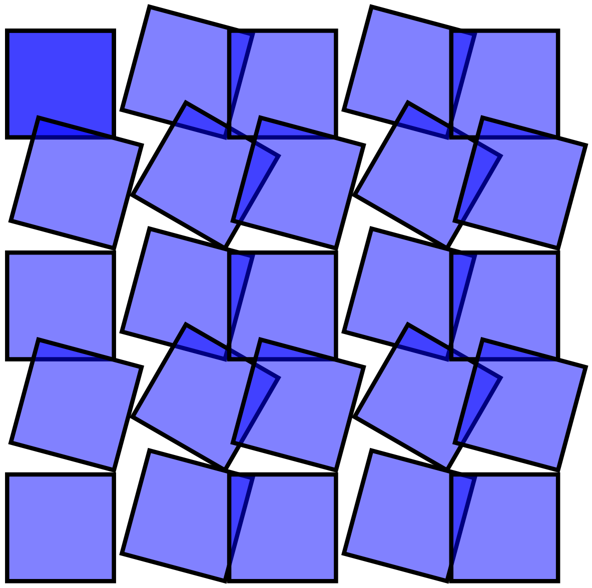
The *Rotation* tab allows one to rotate the tiles depending on the row and column position. The rotation is specified in degrees. The rotation can be specified to *Alternate* between a positive and negative value. A *Randomizer* factor can also be specified.



The Rotation tab of the Tiling dialog.



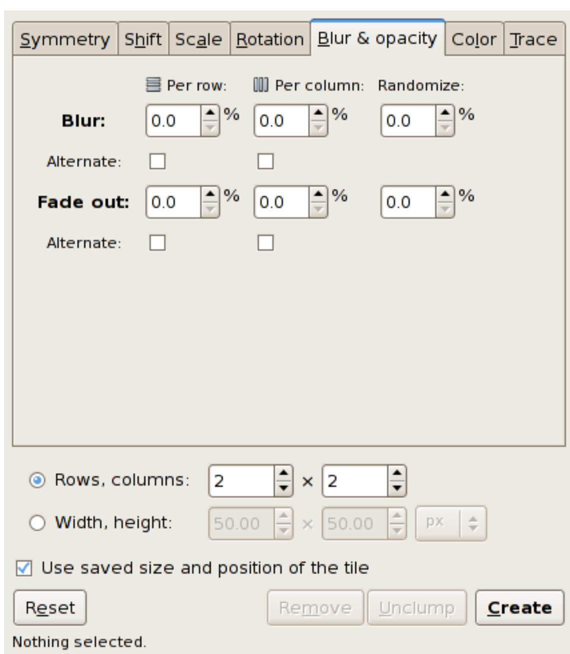
A P1 symmetry tiling with a 10 degree rotation for each row and column.



A P1 symmetry tiling with a 15 degree alternating rotation for each row and column.

Blur and Opacity Tab

The *Blur and opacity* tab allows one to change the *blur* and/or *transparency* of each tile depending on the row and column position.



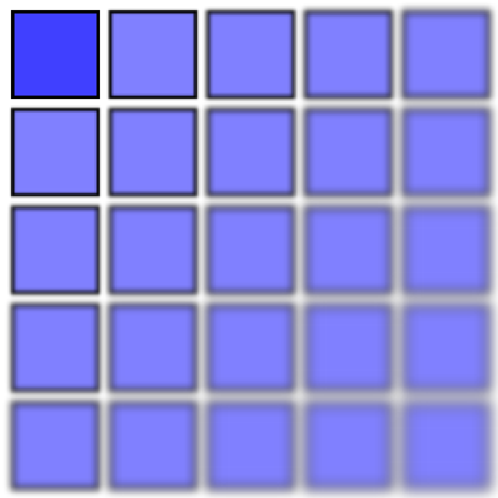
The Blur and opacity tab of the Tiling dialog.

Blur

New in v0.45.

A *Gaussian Blur* filter can be applied to each clone with different blurring values.

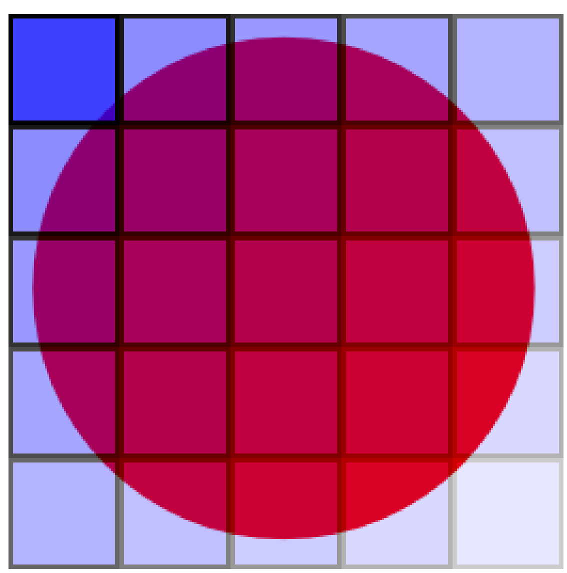
The blur change is specified in percent. The change in blur can be specified to *Alternate* between a positive and negative value; however, a negative blur value can be entered in the *Per row* and *Per column* boxes. A *Randomizer* factor can also be specified.



A P1 symmetry tiling with a 2% increase in blur for each row and column.

Opacity

The opacity change is specified in percent. The change in opacity can be specified to *Alternate* between a positive and negative value. A *Randomizer* factor can also be specified.



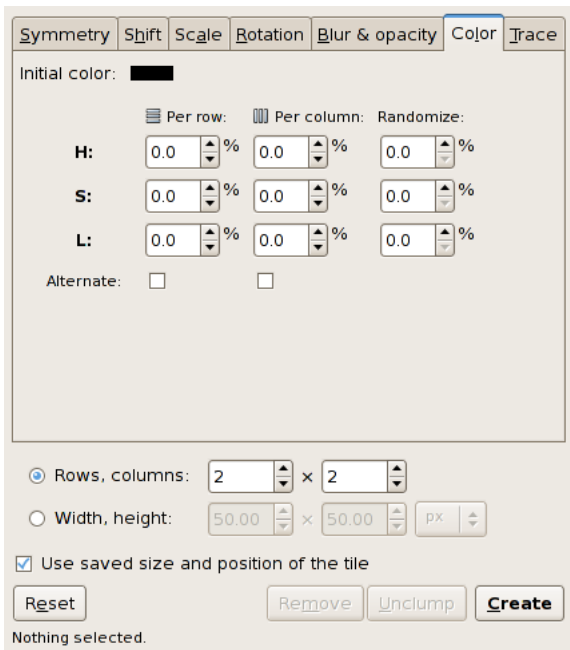
A P1 symmetry tiling with a 10% decrease in opacity for each row and column.
A red circle has been placed under the tiling to illustrate the changes in opacity.

Color Tab

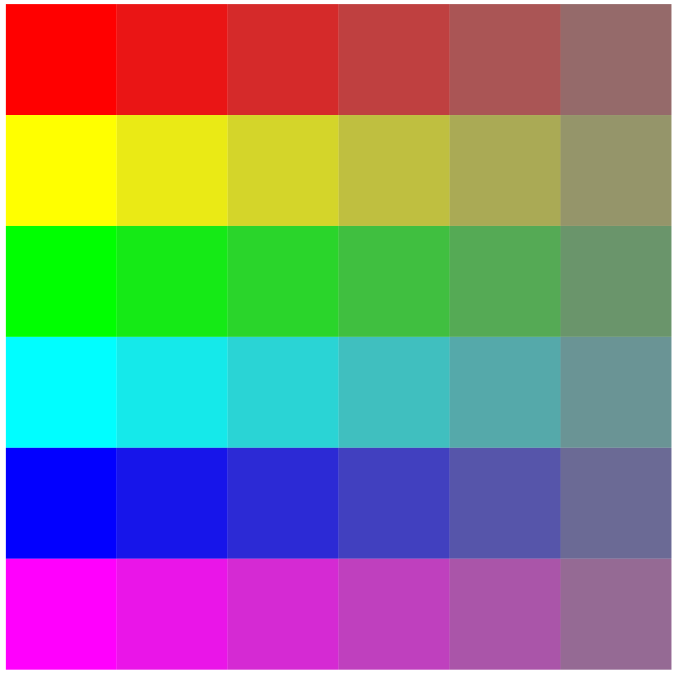
The *Color* tab allows one to change the *Color* of each tile depending on the row and column position. The color change is specified in percent for each of the three components of a color specified with the *HSL* standard (see the section called "HSL"). The *Hue* repeats itself after a change of 100%. The full scale for *Saturation* and *Lightness* components are each 100%. The changes in the three parameters can be specified to *Alternate* between a positive and negative change. A *Randomizer* factor can also be specified.

Two key points: First, the *Fill* and/or *Stroke paint* must be specified as *Unset* (?) (see the section called "Fill and Stroke Paint"). Second, an *Initial color* must be specified by using the *Initial color of tiled clones* dialog accessible by clicking on the color button next to the *Initial Color* label.

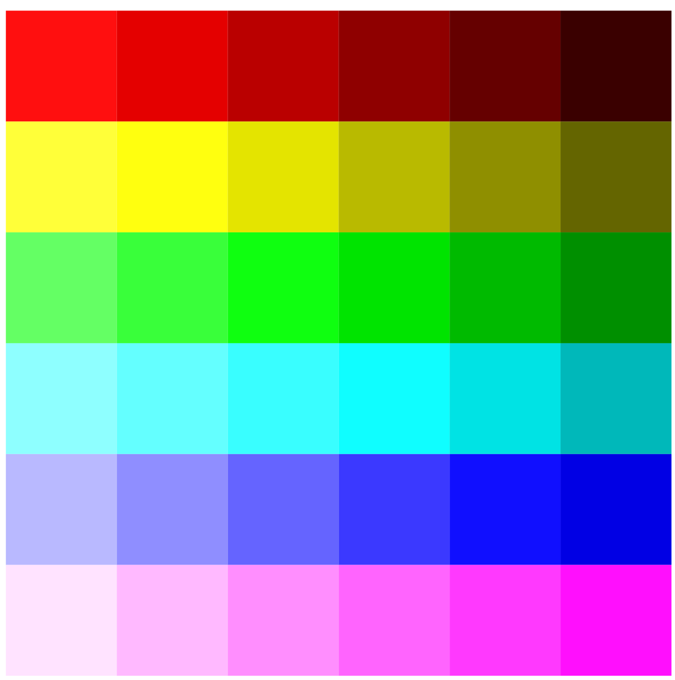
Note that it is meaningless to have only a shift in *Hue* with a starting color of black or white. This is like trying to walk east from the north pole.



The Color tab of the Tiling dialog.



A P1 symmetry tiling with a 16.7% change in *Hue* per row and a -16.7% change in *Saturation* per column. The starting color is a red with 100% *Saturation* and 50% *Lightness*.



A P1 symmetry tiling with an 8.3% change in *Lightness* per row and a -8.3% change in *Lightness* per column. The starting color is a red with 100% *Saturation* and 50% *Lightness*.

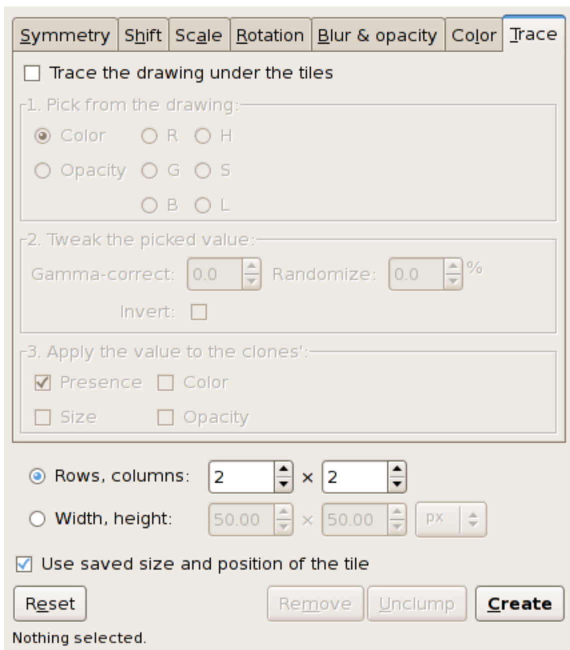
Trace

The *Trace* tab allows one to set the color, size, and transparency of the tiles by the color or transparency of the objects (including bitmaps) that are placed under the location of the tiling. To enable this feature, the *Trace the drawing under the tiles* box must be checked.

The *Trace* tab has three sections. At the top is a section for specifying what property of the underlying drawing should be used for input. Options include the color, one of the *RGB* components, or one of the *HSL* components. There is also the option to use the *Opacity*, which is the sum of the opacities (*Alpha*) of all objects under the tile.

In the middle of the tab is a section to modify the input value. One can specify a *Gamma*[1] correction or add a randomization factor to the input. One can also invert the input.

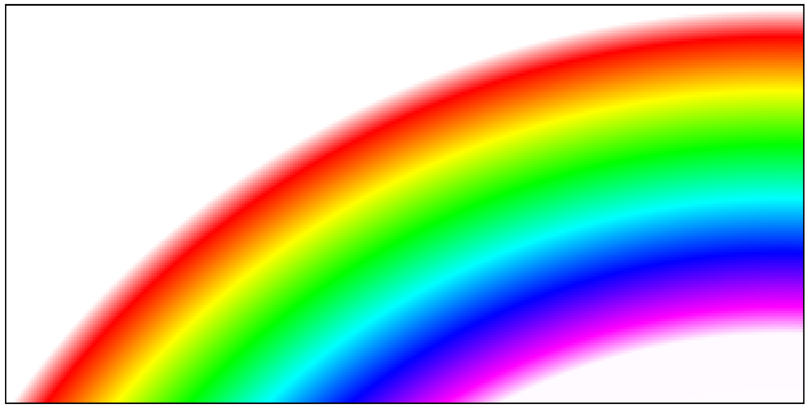
The bottom section is for specifying what should be affected by the input. Options include *Presence* (the probability that a given tile will be drawn), color, size, and opacity. The color will only be changed for regions of the base tile that have *Unset* fill.



The Trace tab of the Tiling dialog.

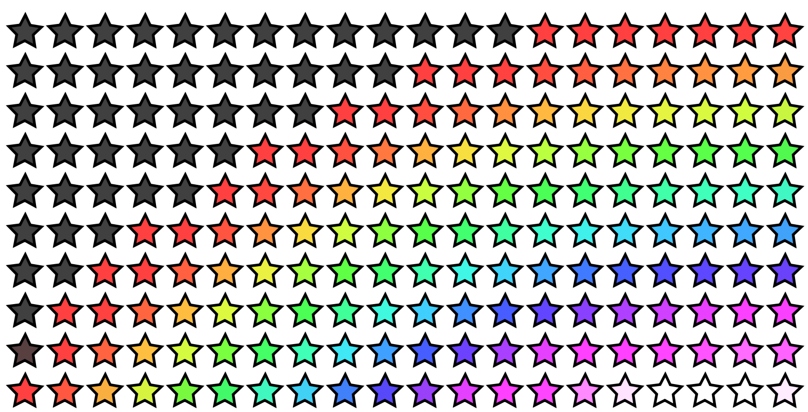
The following figures show the effect of some of the possible combinations of input and output options. All the figures use the first rainbow figure as the input drawing. The rainbow is a *radial gradient* with multiple stops. The inside of the rainbow is defined as a white gradient stop with zero *Alpha*. The last outside stop is defined with a red color and with zero *Alpha*. For most figures, a star inside an unfilled rectangle is used as the base tile. The star has been given an *Unset* fill when color is selected in the output.

[1] See appendix for definition of *Gamma*.

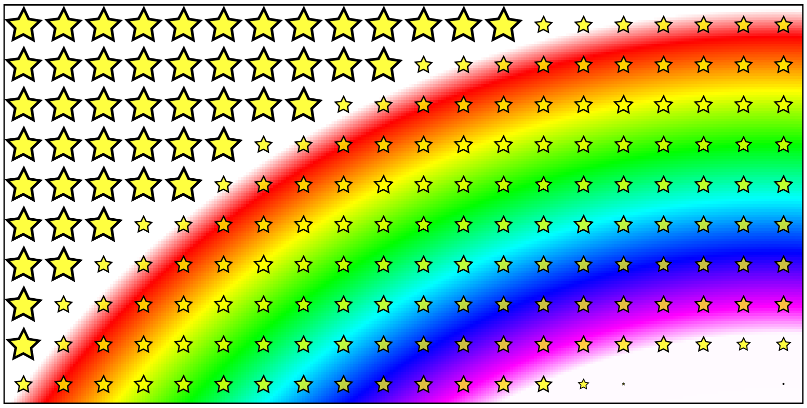


The rainbow pattern used for the background (a radial gradient).

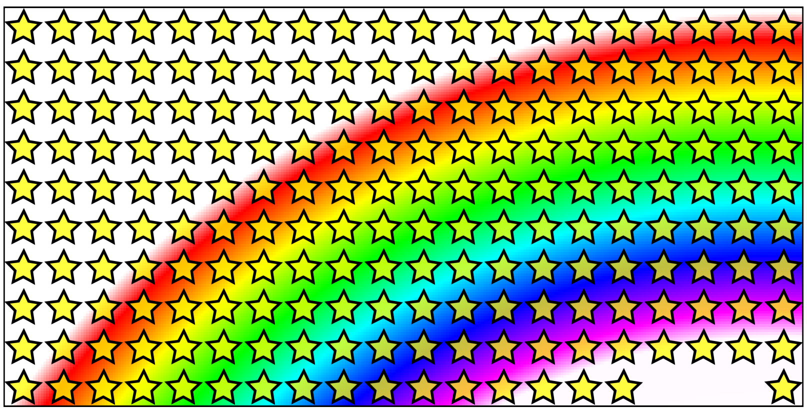
Input: Color. Output: Presence.



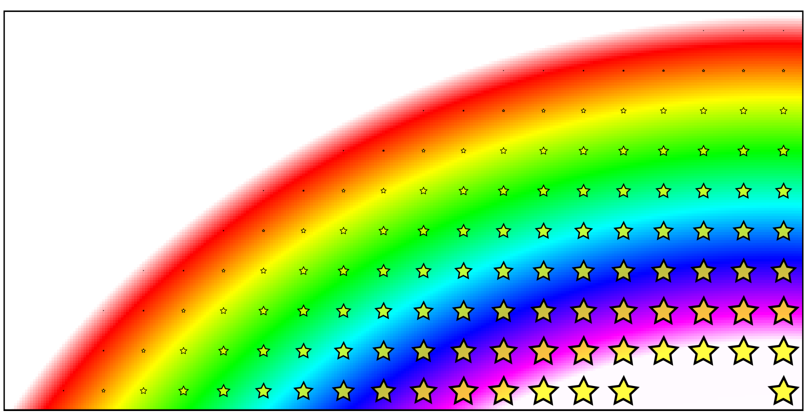
Input: Color. Output: Color. Background rainbow has been removed.



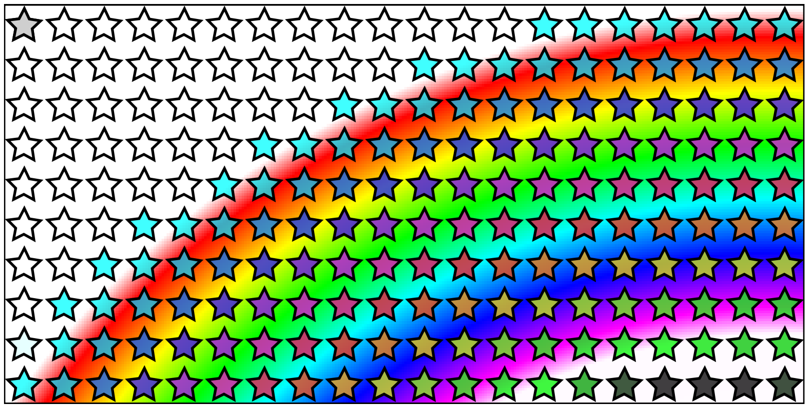
Input: Color. Output: Size.



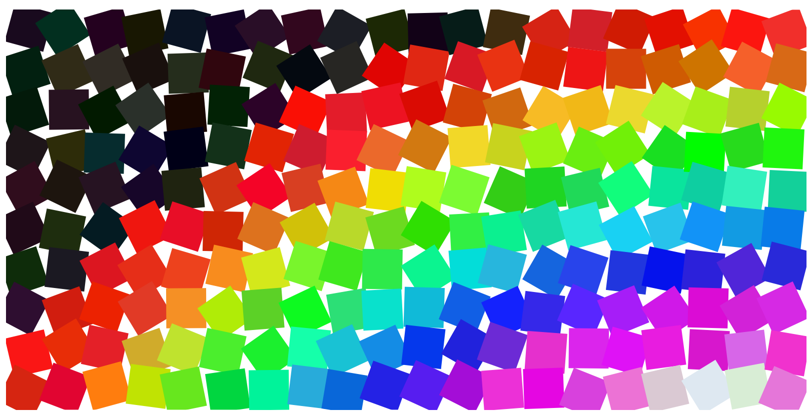
Input: Color. Output: Opacity.



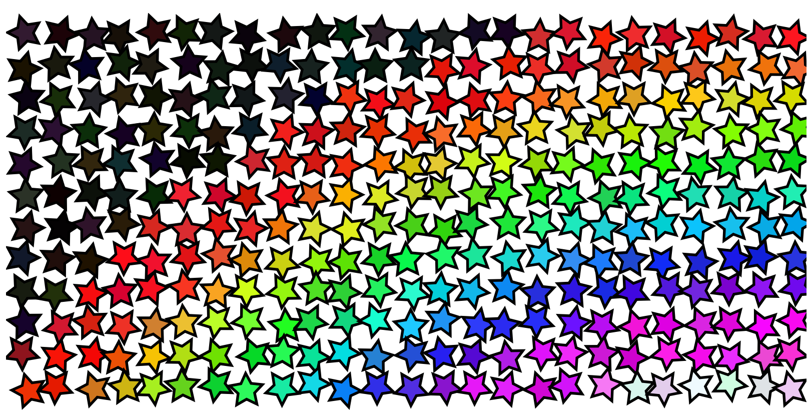
Input: Hue. Output: Size. Note how the red has a hue of zero and purple has the maximum value.



Input: Hue, inverted. Output: Color.



Input: Color, 10% random gamma. Output Color. Changes made to other tabs: Shift: random 10%, Rotation: random 20%. A square base tile with *Unset* fill has been used. The background rainbow has been deleted.

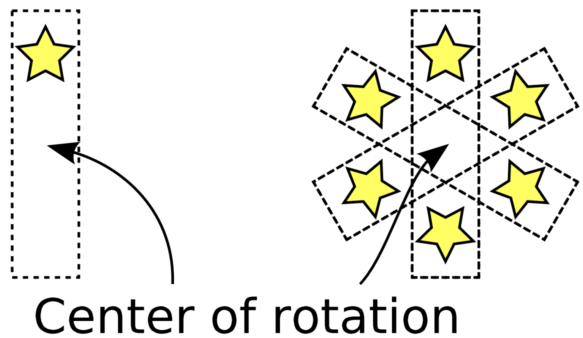


Input: Color, 10% random gamma. Output Color. Changes made to other tabs: Shift: -20%, random 10%, Rotation: random 20%. The number of rows and columns has been increased to compensate for the shift. The background rainbow has been deleted.

Tricks

It is possible to exploit the *Tiling* dialog to produce a number of useful effects. The most interesting is placing tiles along an arc or spiral.

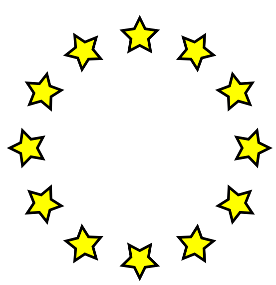
The key to placing tiles along an arc or spiral is to group the “tile” or object you wish to place along an arc with a larger rectangular box and then use the P1 symmetry with one row of tiles, shifting each tile back to the center of the circle or spiral. To see how this is done, look at the next figure.



The base tile is drawn on the left, showing the *bounding box* of the tile. On the right is after a P1 tiling with a per column shift of -100% and a rotation of 60%.

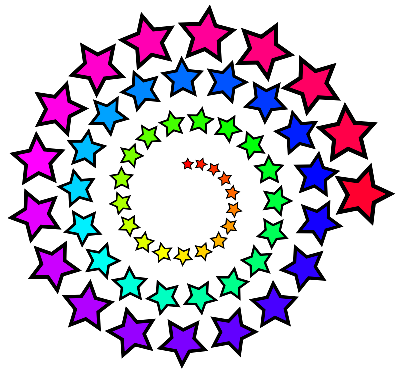
You can see above that the center of rotation is the center of the *bounding box*. Each column is shifted back in *x* by -100%.

The next figure shows how 12 stars can be put in a circle. This would have been an alternative way of placing the stars in the European Union flag if the stars did not need to be placed with one of their points straight up.

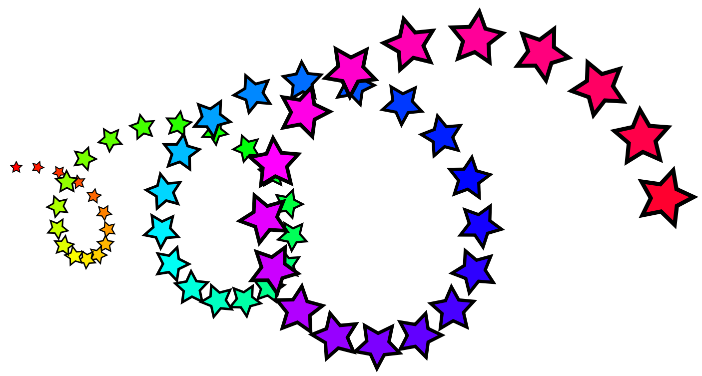


Twelve stars in a circle.

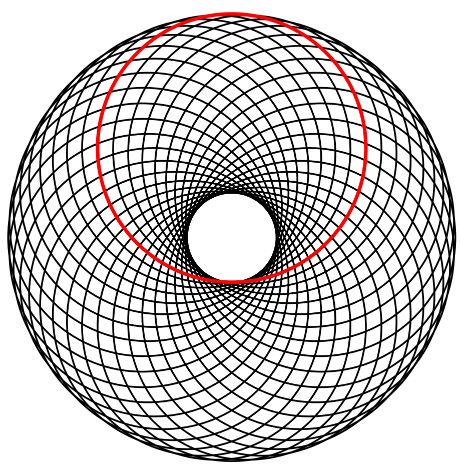
This trick can also place objects along a spiral by specifying that the tile should get larger with each column. Unfortunately, the current version of Inkscape only allows for a uniform increase in size, which results in the tiled object running into each other after a couple of loops.



Stars on a constant spiral. The tiling size is increased by 7% for each column.



Stars on a moving spiral. The tiling size is increased by 7% for each column and the shift per column has been set to -60%.



A circle tiled on an arc. The red circle (grouped with an invisible rectangle) was the source tile.

One could avoid the need to group an object with a rectangle to define the center of rotation for tiling on an arc by using the *XML Editor* dialog to add the following Inkscape private attributes to an object: *inkscape:tile-cx*, *inkscape:tile-cy*, *inkscape:tile-h*, and *inkscape:tile-x*. In this case, make sure the *Use saved size and position of the tile* option is checked.