



Version 1.02

A tutorial by John Polhemus, email: fretsawyer@verizon.net
With the oversight and incredible patience of Brian Damgaard
This document is released into the public domain

Introduction

Each figure in YASC is a square image, i.e., Floor, Wall, Pusher, Box, Goal, Pusher on Goal and Box on Goal.

The figure images are assembled in a specific “row and column” arrangement as shown in image 1, Skin Diagram.

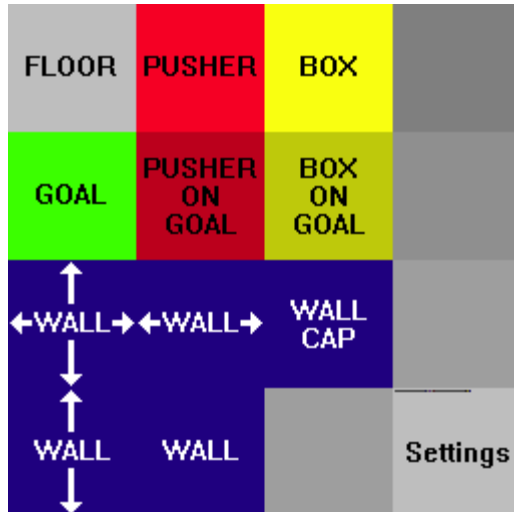


Image 1, Skin Diagram

Any simple image editor can be used to create the figure images and assemble them into the final skin. I used “*mspaint*” (comes with Windows operating systems) to make all the images in this tutorial.

A square measuring 64 x 64 pixels for each figure is recommended as it allows for fairly detailed drawing and is well suited for the way computers deal with pictures.

After the individual figure squares are assembled their combined size will be 256 x 256 pixels and is saved as a single 24-bit bitmap image. That combined image is the skin.

Note: it is important that the saved skin image not have any extra space around the combined figure images. Make sure to resize the image area in the image editor before saving it!

The square labeled “Settings” is for skin specific YASC settings embedded into the image. Additional squares can also be added for animation effects during game play and the solution. Those subjects will be discussed in later chapters.

Note: all the “pictures” in this document (except the cover) are high quality bitmap images. You can zoom in to full magnification with full clarity.

You can also click on any image, copy it and then paste it into your image editor.

Image 1 can also be used as skin. It’s interesting to see and shows clearly how YASC uses a skin to construct what you see on-screen. Copy it, paste it into your image editor, size it to 256 x 256 pixels, and save it as a 24-bit bitmap. Put it in your common skins folder and give it a try. If you’re not familiar with that folder, here’s the path to it:

... \Documents\Sokoban\Sokoban YASC\Skins\Common Skins.

Let’s get started

We’ll make the skin for levels to appear on-screen as shown in image 2. In the process, a good basic understanding of how skins are made will be gained.

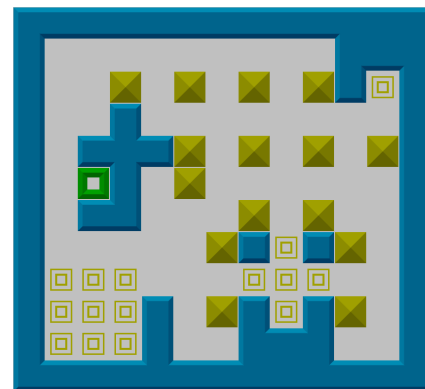


Image 2

The level shown above is useful for testing a skin to see how the various figures look in specific situations, e.g., grouped boxes, pusher or box on goal or in a pocket of walls. If you’d like to use it, highlight and copy the skin demo level text below. Open YASC and import it by clicking on the clipboard button beneath the “Open” button.

Skin Demo Level

```
#####
#-----###
#--$-$-$-$#.#
#--#-----#
#-###$-$-$-$#
#-@#-$-----#
#-##---$-$--#
#----$#.#$-#
#...---...--#
#...#-$#.#$-#
#...#--###--#
#####
```

If you would like to try out the skin we'll be making, copy image 3. Name it "Demo Skin" and save it in your common skins folder.

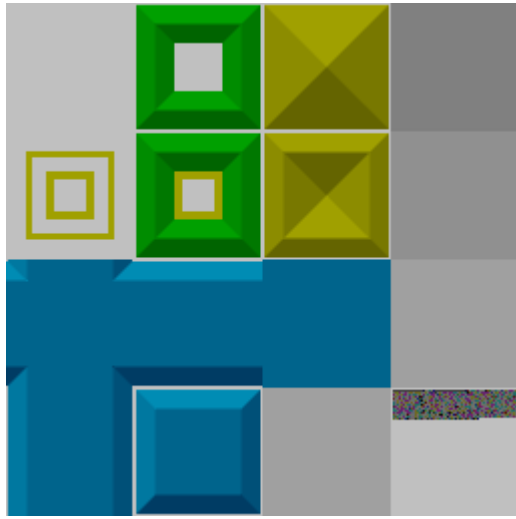


Image 3

Note: don't forget to resize the "Demo Skin" to 256 x 256 pixels if your image editor doesn't automatically do that!

Now, back to making the skin

Open your image editor and pick four colors, one each for the floor, wall, pusher and box. Pick a fifth color if you want the goal to be a different color than the box. Draw a filled 64 x 64 pixel square in the chosen color for each figure.

Image 4 shows the colors I chose to use. If you'd like, you can use them with your image editor's pick color tool for making your figure images.



Image 4

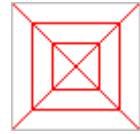
Note: I should mention that I am among the small percentage of the male population having a certain type of genetic color perception deficiency. Please feel free to use colors you prefer instead!

There was some loss in the RGB values bringing the images into my PDF program from my image editor. That may also happen when you copy from this PDF into your image editor. I put the RGB values in the color images so you can readjust their colors when you paste them into your image editor.

Note: if you're not familiar with the term RGB value, it refers to the amount of red, green and blue that make up a given color.

When you're designing a skin you need to have an idea of what you want the figures and finished look to be. For this skin I made the figure template shown in image 5 to use as the starting point for each of the figures. You could just copy and paste it of course but it would be a better learning experience to draw it yourself.

Image 5



Draw a 64 x 64 pixel unfilled square using the floor color. Using a contrasting color, draw a line from the opposing corners to make an X inside it, leaving the outer floor colored rim intact. Pure red (RGB value 255, 0, 0) is a good choice, unless of course you've chosen red for the floor color! If so, pick another color for the contrasting line. Obviously, don't use one of the colors chosen for the other figure images either. From here on in the tutorial, when I refer to doing this or that to the red lines it is these lines that I mean. Of course, if you use a different color you'll have to mentally substitute that color.

Draw a red unfilled 44 x 44 pixel square and red unfilled 24 x 24 pixel square. Center them inside the 64 x 64 pixel square using the red X. You'll be making copies of your figure template throughout the tutorial, so save it as a separate bitmap image.

Making the Goal Image

Make a copy of your figure template (step 1) and modify it by erasing the parts of the red lines to make it look like step 2 in image 6.

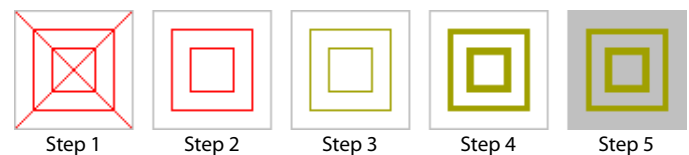


Image 6

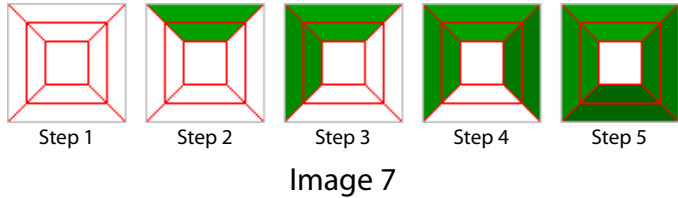
Since the idea of Sokoban is to put the boxes on the goals it makes sense (to me anyway) to make the goal image the same color as the box, as long as the design in mind for the goal won't make that confusing. So, use the pick color tool on the color image for the box, then fill the red lines (step 3).

A line that only is one pixel wide won't always show up on-screen, so, add two lines of pixels to the inside of the larger square and three lines of pixels to the inside of the smaller square in the goal

image (step 4). Finish the goal image by filling the interior spaces of the figure template with the floor color (step 5).

Making the Pusher and Pusher on Goal Images

Modify another copy of your figure template as shown in step 1, image 7.



Fill the top two sections with the color for the pusher (step 2).

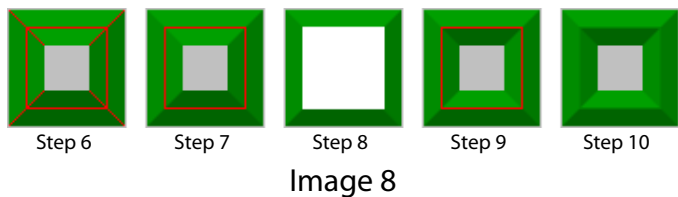
The RGB values for the green I used are 0, 160, 0. The pusher is given a 3D look by using different RGB values on each side for light and shadow effects.

The left side is filled with 0, 140, 0. Note the 20 point darker difference between the two sides (step 3).

The right side is filled with 0, 120, 0. Again, 20 points darker (step 4).

The bottom is filled with 0, 100, 0, another 20 points darker (step 5).

Fill the small center square and red line with the floor color (step 6, image 8).



The red corner lines get filled with an RGB value half way between the RGB values of the two sides forming each corner (step 7).

The top left red corner line is filled with 0, 150, 0, half way between the top sides value of 0, 160, 0 and the left sides value of 0, 140, 0.

Following the same "halfway" rule of thumb:
 Fill the upper right red corner line with 0, 140, 0.
 Fill the lower left red corner line with 0, 120, 0.
 Fill the lower right red corner line with 0, 110, 0.

Use the select tool to select the center area and the red line surrounding it, then cut it (step 8).

Paste the center area back in and rotate it 180° (step 9).

The pusher image is finished by filling the red line on the ridge with an RGB value of 0, 130, 0 which is halfway between the RGB values of the two sections forming each side (step 10).

The pusher on goal image is made by simply drawing the four lines of pixels of the center of the goal inside the center of a copy of the finished pusher image as shown in image 9.

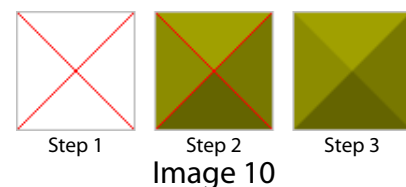


The single line of floor colored pixels surrounding the finished pusher images provides a bit of space for separation between the pusher and boxes and when the pusher is against a wall. The single line of floor colored pixels surrounds the box, box on goal and wall images as well for the same reason.

Using a single line of floor colored pixels in this way is a common practice, but not a rule. If you wanted the pusher and boxes to be flush against each other and the walls (like the skin diagram) the images would be drawn using the full 64 x 64 pixel square.

Making the Box Image

Modify another copy of your figure template as shown in step 1, image 10.



Fill the top side with the color for the box (top of step 2).

Fill the left side with an RGB value 20 points darker (left side of step 2).

Fill the right side with an RGB value 20 points darker than the left side (right side of step 2).

Fill the bottom side with an RGB value 20 points darker than the right side (bottom of step 2).

The box image is finished by splitting the RGB values of the adjoining sides and filling the red corner lines with those RGB values (step 3).

Making the Box on Goal Image

Modify another copy of your figure template as shown in step 1, image 11.

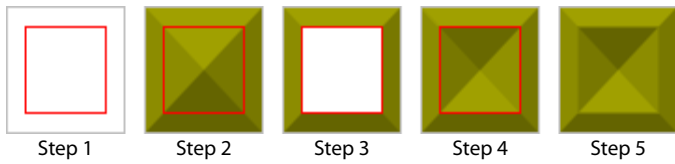


Image 11

Make a copy of the finished box image and align the modified figure template over it (step 2).

Select the center up to, but not including the red line and cut it (step 3).

Paste the center back in and rotate it 180° (step 4).

Fill the red lines with the RGB value half way between the sides RGB values (step 5).

Making the Wall Image

Modify another copy of your figure template as shown in step 1, image 12.

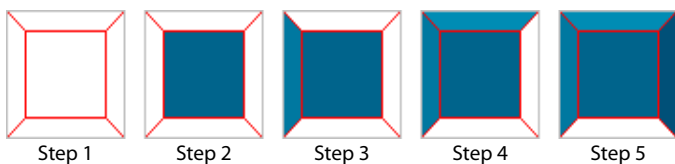


Image 12

Fill the top surface with the wall color (step 2).

The wall side surfaces are filled a little differently than the pusher and box images because it has a top surface that they don't.

Fill the left side surface with an RGB value 20 points lighter than the top surface (step 3).

Fill the top side surface with an RGB value 20 points lighter than the left side surface (step 4).

Fill the right side surface with an RGB value 20 points darker than the center surface (step 5).

Fill the bottom side surface with an RGB value 20 points darker than the right side (step 1, image 13).

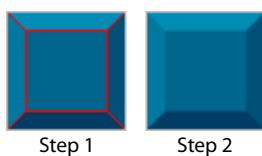


Image 13

That leaves the last step that you should be an expert at by now. Fill the red lines with RGB values half way between their adjoining surfaces, completing the wall image (step 2).

Making the Horizontal and Vertical Wall Images

Modify another copy of your figure template as shown in step 1, image 14.

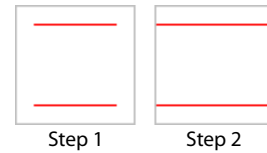


Image 14

Extend the red lines to the inside edges of the figure template (step 2).

Make a copy of the modified figure template and rotate it 90° for making the vertical wall image.

Place a copy of the wall image between the modified figure templates (steps 1, 2 & 3, image 15).

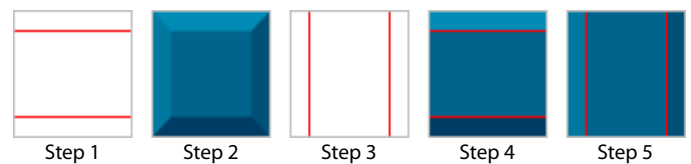


Image 15

Fill the three surfaces of the figure templates with the color of the corresponding surfaces of the wall image (steps 4 & 5).

Well ...you know what comes next! Fill the red lines with the RGB value half way between the surfaces RGB values (steps 1 & 2, image 16).

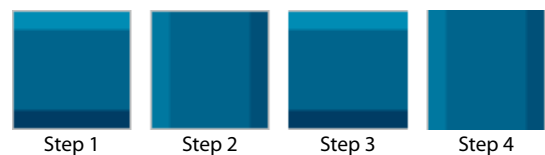


Image 16

The final touches to finish the horizontal and vertical wall images is to extend the colors into the 1 pixel floor color in their respective ends (steps 3 & 4). Zoom in closely and you'll see the 1 pixel floor color remaining on their sides, but not their ends.

Making the Inside Corners Image

To show how YASC uses the inside corners image to complete the inside corners where walls meet, I put a white square where the wall-cross image should be in a copy of the tutorial skin (image 17).

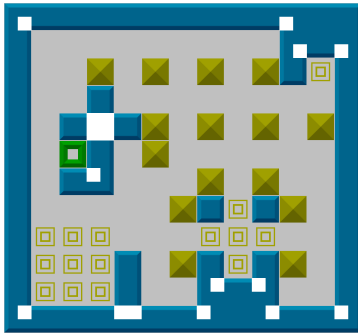


Image 17

If a square of the wall color were used the corners would look like image 18. The inside corners image will make them look as they should.

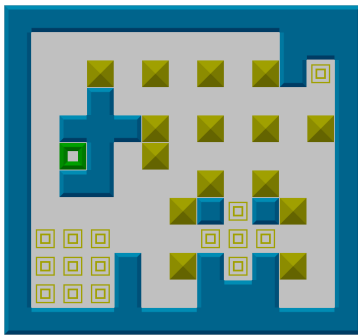


Image 18

To make the inside corners image modify another copy of your figure template as shown in step 1, image 19.

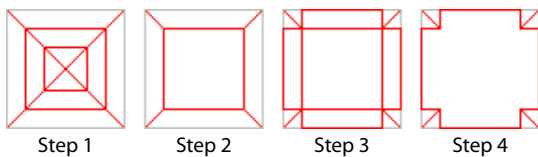


Image 19

Progressively modify it as shown in steps 2, 3 and 4. Zoom in and pay particular attention to steps 3 and 4.

Fill the center with the wall color (step 1, image 20).

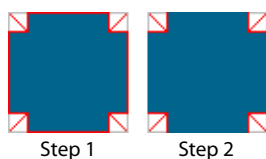


Image 20

Fill the parts of the red lines on each edge with the wall color. Zoom in for a close look (step 2).

I placed four copies of the wall image around the wall-cross image, then drew arrows to illustrate which wall image side surface color goes in each wall-cross image corner section. See image 21.

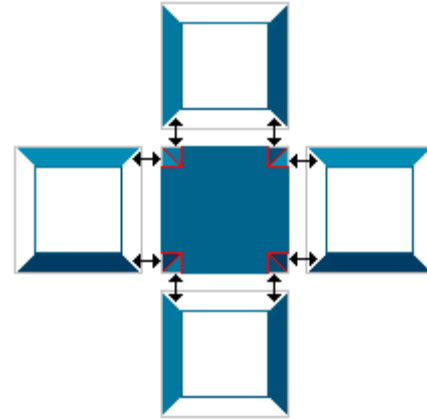


Image 21

Once the corner sections have been filled, change the floor colored pixels on their edges to the color of the section. Make sure to leave the single floor colored pixel in each corner! Zoom in for a close look (step 1, image 22).

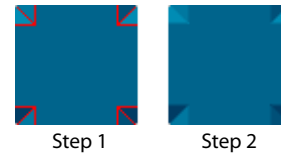


Image 22

Finish the inside corners image by filling the red lines in each corner section with the RGB value half way between the RGB values of the two sections making up each corner.

Now that all the figure images have been drawn they can be assembled according to the skin diagram (image 1). You can use a square of any color you wish for the four unnamed squares and the settings square.

Name it "Tutorial Skin" and save it as a 256 x 256 pixel, 24 bit bitmap image in your common skins folder.

The next thing we'll do is make another version of our skin, this time with a directional pusher.

Making the Directional Pusher Images

Make a copy of the "Tutorial Skin" and name it "Tutorial Skin With Directional Pusher".

To draw the directional pusher, begin by making another copy of your figure template. I have used blue lines to show how to progressively modify it as shown in image 23.

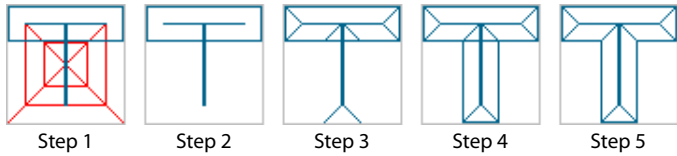


Image 23

Zoom in for a close look at step 1. Draw the blue lines as shown in your copy of the figure template.

Erase the red lines as shown in step 2.

Draw the diagonal lines as shown in step 3.

Draw the bottom and side lines of the vertical section as shown in step 4.

Erase the portion of the blue line where the horizontal and vertical sections meet as shown in step 5.

The figure template surfaces are now ready to be filled in with color.

Make a copy of the figure template and rotate it 90°.

Make a copy of that one and rotate it 90°.

Copy that one and rotate it 90° as well, giving you four figure templates facing up, left, down and right.

Make a copy of the four figure templates and place them over copies of the four sided pusher image as shown in image 24.

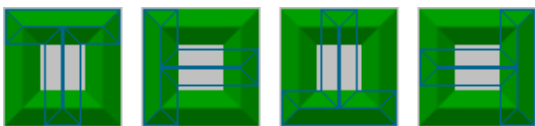


Image 24

The aligned copies show which colors go in the end sections and pushing surface of each directional pusher image.

Fill those sections in the first set of figure templates with those colors as shown in image 25.

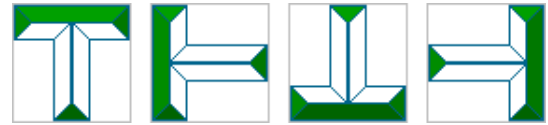


Image 25

I've left just the color for the surface opposite the pushing surface showing in image 26.



Image 26

Fill those sections in the first set of figure templates with those colors as shown in image 27.



Image 27

The top and left sides of the four sided pusher image are used for the colors of the trailing portions of the directional pushers. Fill the corresponding surfaces of the trailing portions of the pusher images as shown in image 28.



Image 28

Fill the blue lines on the directional pusher images with the RGB values halfway between their adjoining surfaces.

Align them to each other as shown in image 29. These will become the row of pusher on floor images.



Image 29

Fill the blue lines on the directional pusher images with the RGB values halfway between their adjoining surfaces.

Make a copy of the row of pusher on floor images to use for making the row of pusher on goal images.

Make four copies of the goal image and align them in a row as shown in image 30.



Image 30

Place the copied row of pusher on floor images on top of the row of goal images, finishing the row of pusher on goal images as shown in image 31.



Image 31

Align the row of pusher on floor images above the row of pusher on goal images as shown in image 32.



Image 32

Fill the inside of the row of pusher images with the floor color as shown in image 33.



Image 33

Add them to the copy of the tutorial skin named "Tutorial Skin With Directional Pusher" as shown in image 34.

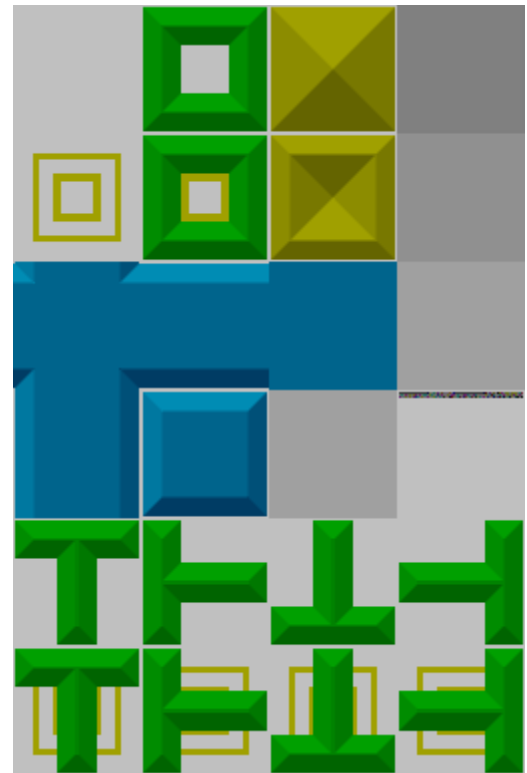


Image 34

Note: there is no need to replace the four sided pusher images in the base layout. The directional pusher images will override them.

Now let's make a version of our skin with a directional pusher and animation.

Animation Images

YASC uses four rows and up to twenty-four columns for animations, making numerous ideas for animation effects possible.

The animations are used for three purposes:

- Highlighting the pusher's legal moves.
- Highlighting boxes legal moves.
- "Celebration" when a level is solved.

The content and order of the animation rows is as follows:

- Row 1 - pusher image on floor
- Row 2 - pusher image on goal
- Row 3 - box image on floor
- Row 4 - box image on goal

Because there is no hard and fast rule as to what size figure images must be, YASC has to guess how many rows and columns a skin image has using a mathematical formula and the skin image dimensions.

This causes a mathematical conundrum where certain row and column combinations aren't recognized properly.

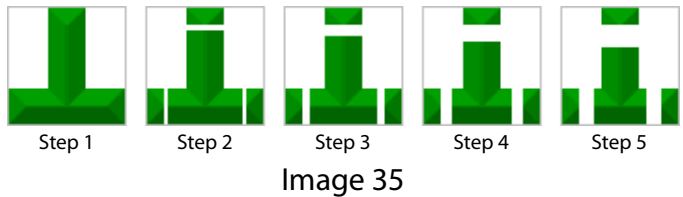
In testing several figure image sizes a general rule of thumb emerged showing that an odd number of columns almost always works. If a skin you are designing fails to be recognized properly, adding or deleting one column of animation will usually fix the problem.

Making the Directional Pusher Animation Images

The animation effect I have in mind is to make the pusher appear to shrink into itself a little bit, then expand back out.

Progressively modifying the down pusher image accomplishes this effect.

Start with a copy of the down pusher image (step 1, image 35).



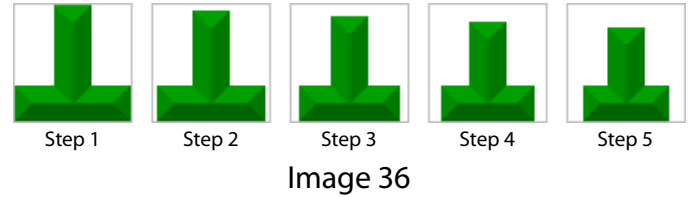
Make a copy of step 1 and use the line tool set to white to "cut" 3 pixels from the top and 2 pixels on each side as shown in step 2.

Copy step 2 and cut another 3 pixels from the top and 2 pixels on each side as shown in step 3.

Copy step 3 and cut another 3 pixels from the top and 2 pixels on each side as shown in step 4.

Copy step 4 and cut another 3 pixels from the top and 2 pixels on each side as shown in step 5.

Select the ends in each step and move them inward, reattaching them as shown in image 36.



Make a copy of steps 1, 2, 3 and 4. Line them up after step 5, in reverse order as shown in image 37.

Make a copy of the row of pusher on floor animation images to use for making the row of pusher on goal animation images.

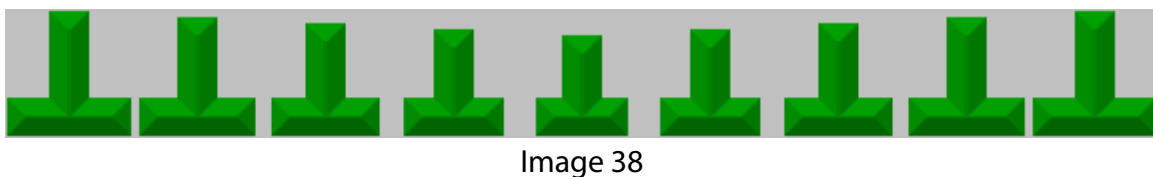
Fill the background of the first set on pusher on floor animation images with the floor color as shown in image 38.

That will be row 1 of the 4 animation rows.

Make nine copies of the goal image and align them in a row as shown in image 39.

Place the copy of the pusher on floor animation images on top of the row of goal images as shown in image 40, page 9.

That will be row 2 of the 4 animation rows.



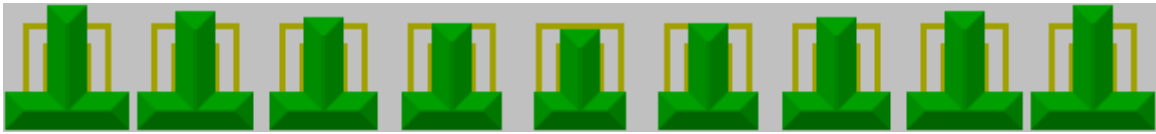


Image 40

Making the Box on Floor and Box on Goal Animation Images

A nice animation effect for the boxes is to highlight one section of the box image at a time in a clockwise rotation. This is accomplished by filling the highlighted section with an RGB value of 200, 200, 0.

Make nine copies of the box on floor image and align them in a row as shown in image 41.

Fill the sides with the highlight color as shown in image 42.

That will be row 3 of the 4 animation rows.

Make nine copies of the box on goal image and align them in a row as shown in image 43.

Erase the sections as shown in image 44.

Make nine copies of the goal image and align them in a row as shown in image 45.

Place the row of box on goal images with the erased sections on top of the row of goal images as shown in image 46.



Image 41



Image 42



Image 43



Image 44



Image 45



Image 46

Fill the erased section with the highlight color as shown in image 47.

That will be row 4 of the 4 animation rows.

Make a copy of the "Tutorial Skin With Directional Pusher" and name it "Tutorial Skin With Directional Pusher and Animation".

Align the 4 animation rows and add them to the "Tutorial Skin With Directional Pusher and Animation" skin as shown in image 48.



Image 47

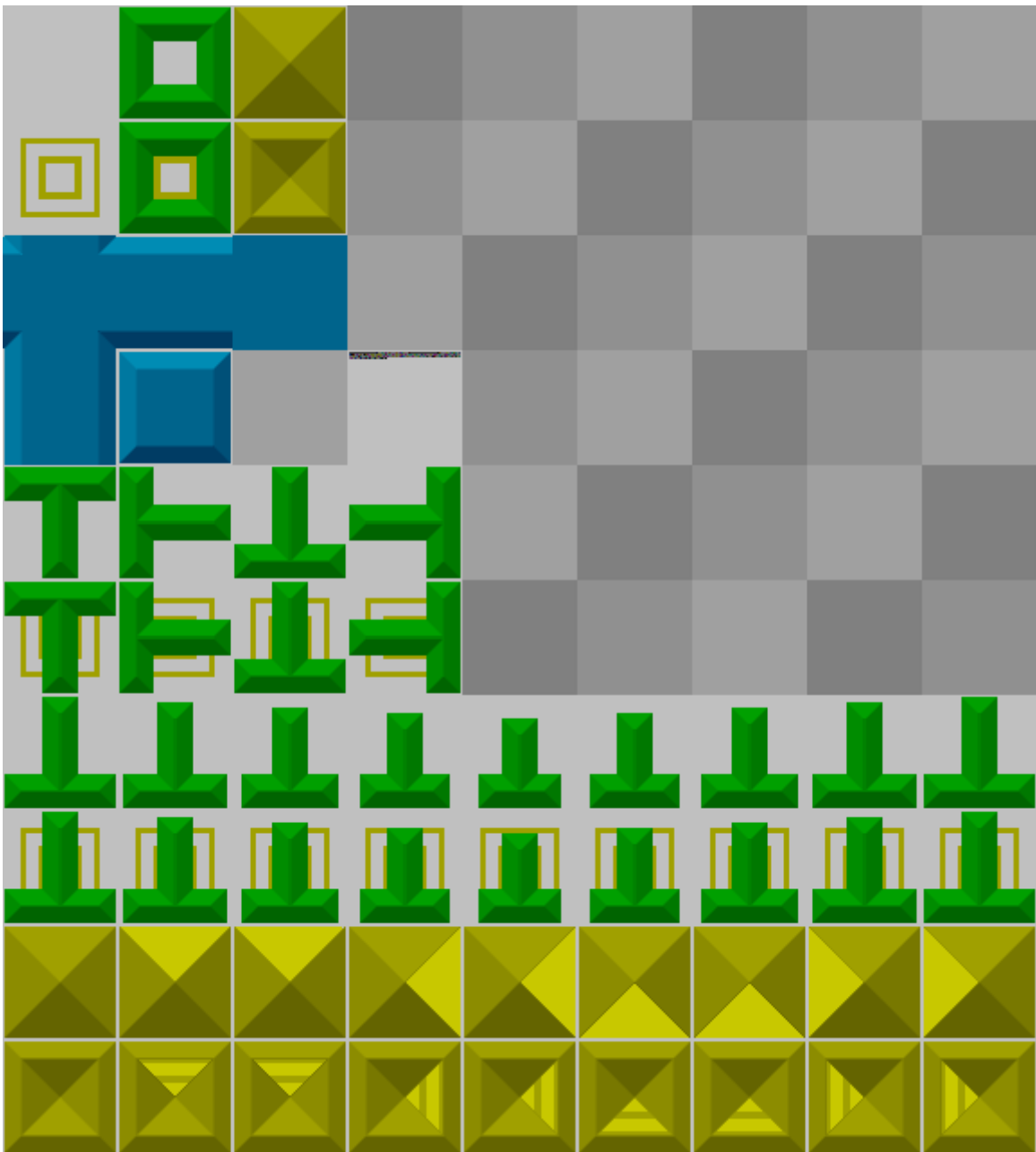


Image 48

Now for our last skin, the "Tutorial Skin With Animation":

As mentioned earlier, we only need to make the pusher on floor and pusher on goal animation images using the four sided pusher images in the tutorial skin.

Making the Four Sided Pusher Animation Images

The animation effect I have in mind is a simple modification of the pusher on floor and pusher on goal images in the tutorial skin, so this will be fast and easy.

I thought we could make the centers of the pusher and pusher on goal images appear to pulsate.

Make and align nine copies of the pusher image above and attached to a row of nine copies of the pusher on goal image as shown in image 49.

The left-most pusher images centers are the floor color. The center of the second pusher images are 20 RGB points darker. The center of the third, fourth and fifth pusher images are each progressively 20 points darker still.

The centers of remaining pusher images get 20 points progressively lighter until the centers of the last pusher images are back to the floor color as shown in image 50.

Align and add these two rows to a copy of the box animation rows we made earlier as shown in image 51.

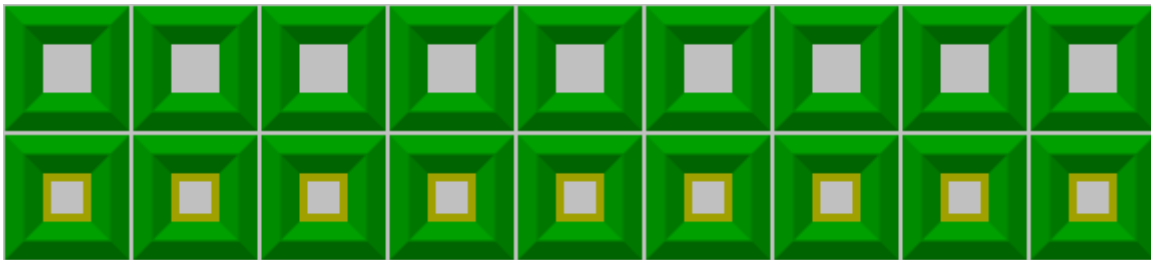


Image 49

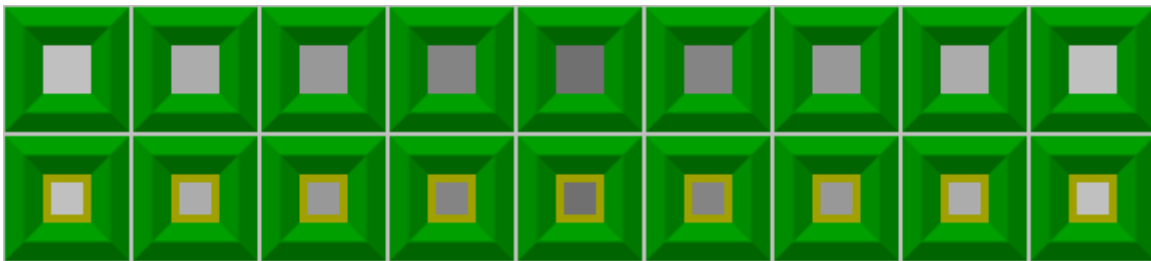


Image 50

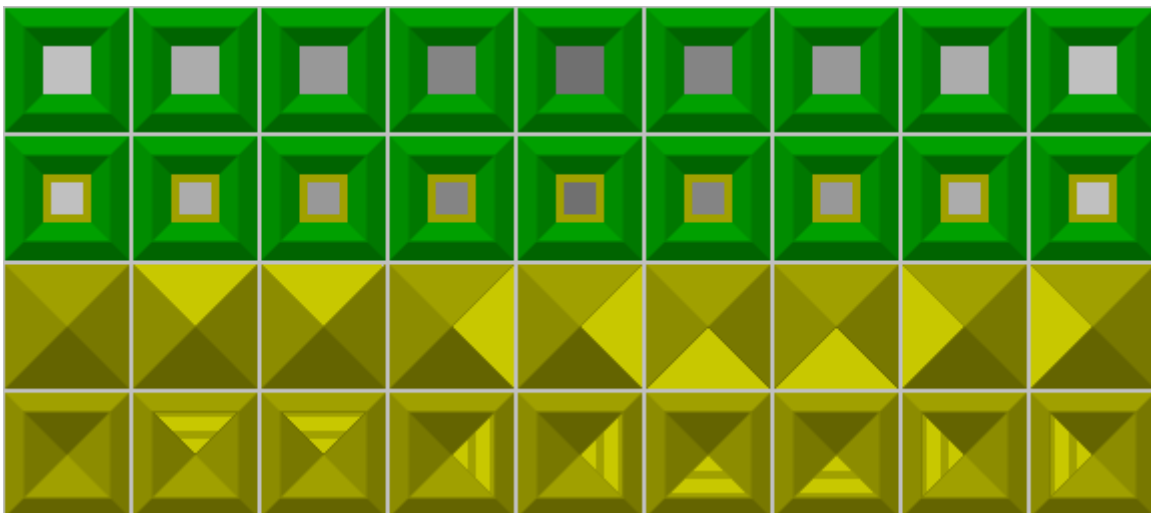


Image 51

Make a copy of the "Tutorial Skin" and name it "Tutorial Skin With Animation".

Add the 4 animation rows as shown in image 52.

Put all four of the tutorial skins in your common skins folder.

We'll wrap things up now with a look at the settings square.

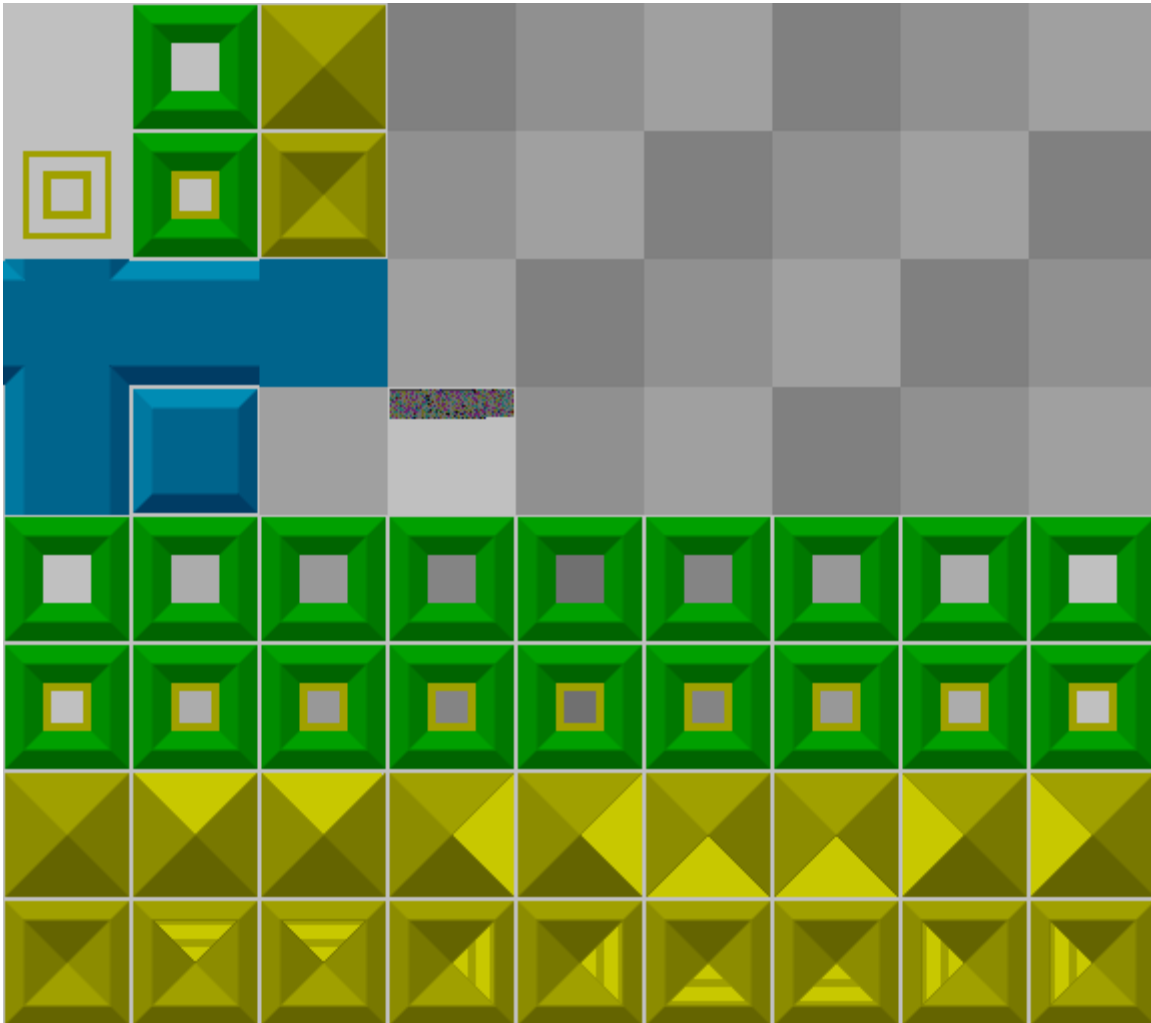


Image 52

Settings Square

Settings necessary for the skin to function properly need to be embedded here.

If needed, settings to fine-tune the appearance of the skin on-screen are also embedded.

Personal preference settings can also be embedded. That way, when you switch from one skin to another you won't have to manually reset them in the settings window.

Note: skins released to the public should only have necessary skin specific settings embedded in them because embedded settings override user settings.

Image 53 is an enlargement of the settings square from the tutorial skin image back on page 2.

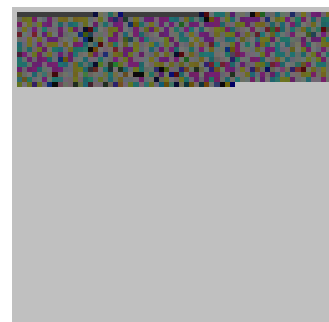


Image 53

The colored pixels visible in the settings square are embedded settings.

Here's the settings in the tutorial skin in text form:

```
=====
Tutorial Skin
=====

[Graphics - Board - Figures - Player on goal square - Image]
Section={Tile 2,1}
Mask image using background color=Yes
Background color={BackgroundColor Bottom Left}

[Graphics - Board - Figures - Wall - Outer wall razor, pixels]
Left=3
Top=3
Right=3
Bottom=3

end
```

The skin's name between the dashed lines is the header.

The word "end" denotes the end of the embedded settings

The header and the word "end" aren't required for the skin to work, but can be thought of as good "housekeeping" or "syntax etiquette".

The lines in between the header and the word "end" are the settings.

The "Tutorial Skin With Animation" has the same settings (except for the name in the header of course).

The "Tutorial Skin With Directional Pusher" doesn't need the first group of settings because of the directional pusher. Its settings look like this:

```
=====
Tutorial Skin With Directional Pusher
=====

[Graphics - Board - Figures - Wall - Outer wall razor, pixels]
Left=3
Top=3
Right=3
Bottom=3

end
```

The settings in the "Tutorial Skin With Directional Pusher and Animation" differs from the "Tutorial Skin With Directional Pusher" by the addition of two needed settings to work properly:

```
Directional player=Yes
Wall type=3x2
```

It is placed right after the header, and before the rest of the settings.

The "3x2" in the line refers to the 3 columns and 2 rows the wall images use in the skin diagram back on page 1.

Skins with directional pusher images and animation images need those settings to help YASC differentiate between our skin and an old, seldom seen "Wall type=4x4" type of skin.

There's more in those colored pixels in image 51 than just the settings. There's also the explanation of what the settings do and why they are there.

I've embedded that information in image 3 on page 2 instead of putting it here in the tutorial to use it for illustrating what the next chapter explains, how to view and edit embed settings.

If you didn't make a copy of image 3 at the beginning of the tutorial, do so now. Name it "Demo Skin" and save it in your common skins folder.

How to View and Edit Embedded Settings

Open YASC and click on the "Settings" button.

Under the "Skins" menu, select "Load skin".

Locate and click once on the "Demo Skin" in the "Files" pane to select it.

Click on the "Menu" button and select "Image text".

The text pane in the window that opens is where you'll see the settings and explanations. Take a few minutes to read them, then we'll continue.

Before you can change or add embedded settings you need to set the text pane to edit mode.

To do that, right-click in the text pane and select "Edit" from the pop-up menu. If you bring up the pop-up menu again there should now be a check mark next to "Edit", indicating that the text pane is in edit mode.

If you want to add or edit any settings, you can do so now that the text pane is in edit mode. When finished, click "OK" to save any changes you've made, embedding them in the skin image. The text pane will close and you'll return to the previous window.

Embedded Settings for the Tutorial Skins

To complete the tutorial skins you've made they need to have the appropriate settings embedded in them.

Some of the settings are essential to make a particular tutorial skin work correctly, whereas other settings are only there for polishing, e.g., the outer wall razoring.

Here is a complete list of embedded settings for each of the tutorial skins.

Copy the settings for each skin and paste them into the skin having the same name that you've made during this tutorial.

```
=====
Tutorial Skin
=====
```

```
[Graphics - Board - Figures - Player on goal square - Image]
Section={Tile 2,1}
Mask image using background color=Yes
Background color={BackgroundColor Bottom Left}
```

```
[Graphics - Board - Figures - Wall - Outer wall razor,pixels]
Left=3
Top=3
Right=3
Bottom=3
```

end

```
=====
Tutorial Skin With Directional Pusher
=====
```

```
[Graphics - Board - Figures - Wall - Outer wall razor,pixels]
Left=3
Top=3
Right=3
Bottom=3
```

end

```
=====
Tutorial Skin With Directional Pusher and Animation
=====
```

```
Directional player=Yes
Wall type=3x2
```

```
[Graphics - Board - Figures - Wall - Outer wall razor,pixels]
Left=3
Top=3
Right=3
Bottom=3
```

end

```
=====
Tutorial Skin With Animation
=====
```

```
[Graphics - Board - Figures - Player on goal square - Image]
Section={Tile 2,1}
Mask image using background color=Yes
Background color={BackgroundColor Bottom Left}
```

```
[Graphics - Board - Figures - Wall - Outer wall razor,pixels]
Left=3
Top=3
Right=3
Bottom=3
```

end

Maximum Zoom Factor

I would like to mention one last setting that I didn't put in the "Demo Skin".

Skins are scaled up for small levels because YASC automatically makes the level fit the window size.

A skin may not have enough details to tolerate being enlarged this way without getting pixellated, i.e., the individual pixels in the image become visible.

Embedding the following setting restricts on-screen display to the actual size the figure images were drawn in:

```
[Graphics - Board - Figures]
Maximum zoom factor (%)=100
```

Conclusion

If not for the oversight and patience of Brian Damgaard, this tutorial would not exist. Its purpose has been to build a basic understanding of how to make a working skin.

My motivation for writing this tutorial was threefold:

- I hoped documenting my experience of learning the basics of how to make a skin will speed up the process and straighten out the learning curve for others wanting to do the same.
- I'm hoping there will be a wave of new skins by budding designers that we all can enjoy.
- I wanted to contribute something to say thank you to Brian Damgaard for the YASC program.

Enjoy, John Polhemus