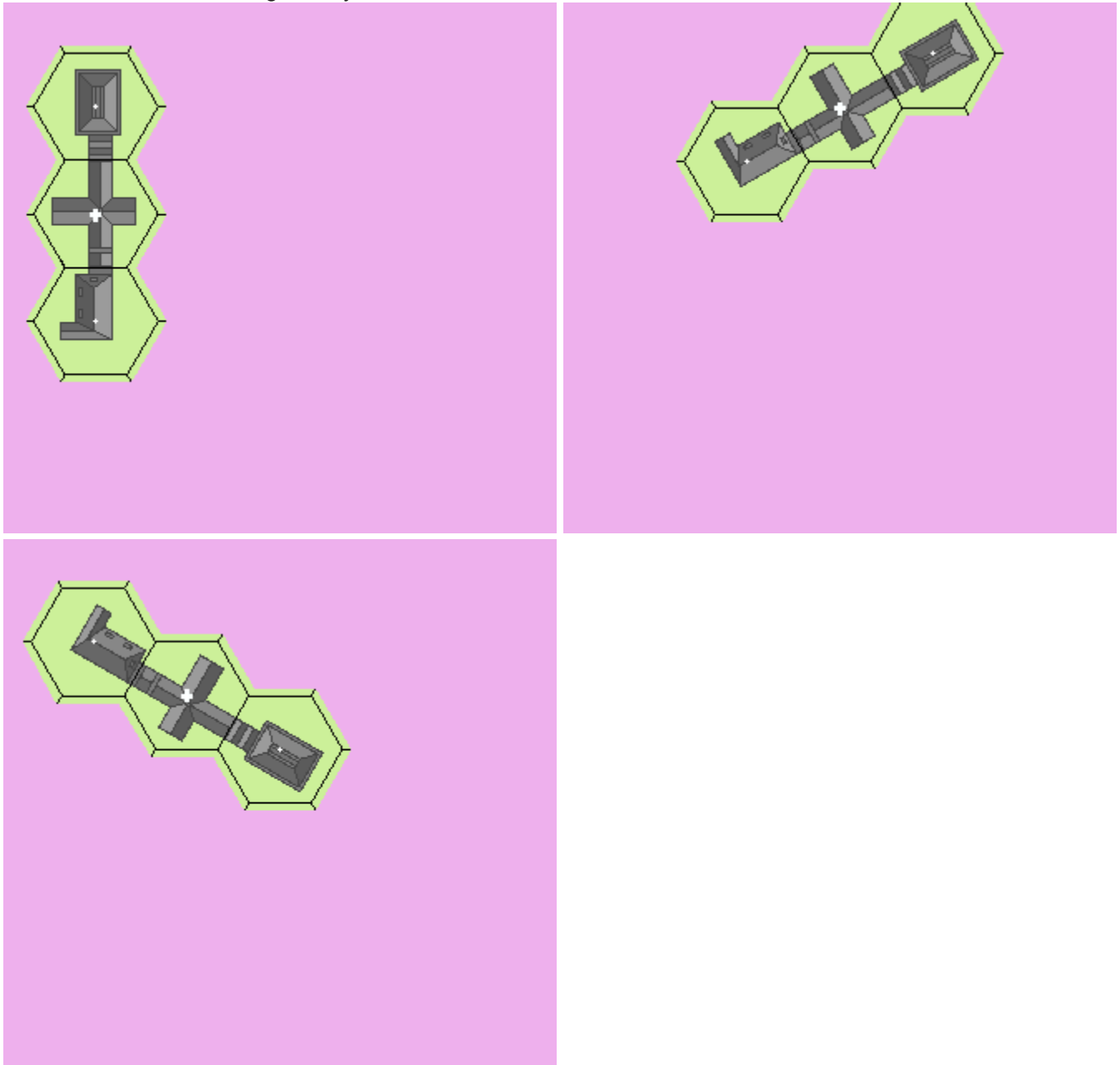


Making VASL overlays

Overlay files are found in the VASL\Boards\Overlays folder. An overlay zipfile contains a group of GIFs and a Data file that tells how VASL how to place them based on the hex coord or coords you enter at setup. For the standard overlay files, a single zipfile may contain many overlay images—for example, the ovrX building overlay file contains all the GIFs for all the building overlays.



Most overlays have six GIFs apiece in the zipfile, one for each possible orientation. Above from left to right are ovrx24a.gif, ovrx24b.gif, and ovrx24c.gif. The a-f suffix of the filename indicates the position of overlay hex 2 relative to hex 1: in ovrx24a, hex 2 is “north” of hex 1, in ovrx24b it’s northeast, and so on clockwise around the rosette to ovrx24f which has hex 2 northwest of hex 1. These designators are required, and “a” is always north.

Some overlays use fewer than six GIFs, since they aren’t intended to be rotated. Overlays 1-6 and most of the single-hex overlays are examples.

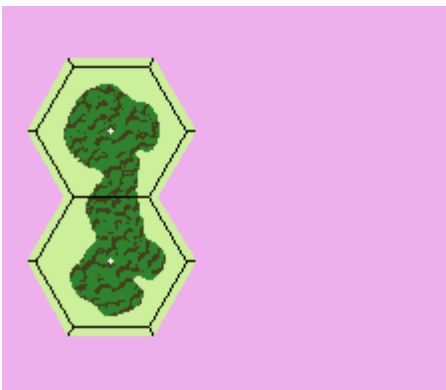
Positioning

Data is a text file that tells VASL approximately where the 1 hex of the overlay is in relation to the upper left corner of the rectangular image. Here's the Data file from the woods overlay file ovrWD:

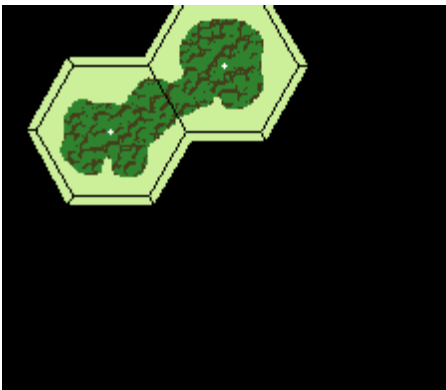
```
wd1 b2;b1;c1;d1;d2;c3
wd2 b2;b1;c1;d1;d2;c3
wd3 b2;b1;c1;d1;d2;c3
wd4 b2;b1;c1;d1;d2;c3
wd5 b2;b1;c1;d1;d2;c3
```

```
version 1.0
```

Each of the five woods overlays has a line in the file. A hex coordinate is given for each of the six orientations (a-f) of each overlay. These coordinates tell what the coordinate of overlay hex 1 would be if the upper left corner of the overlay were the midpoint of the north hexside of A1 (the upper left corner of a standard board). So if you were to align the upper left corner of the overlay with the upper left corner of a board, the coord is where the overlay's hex 1 would fall on the board.



In ovrwd2a, the 1 hex is at the south end of the woods. Looking at the Data for wd2, “a” orientation, you see the coordinate “b2” listed. This means that if you extended the hexgrid to the edges of the image and numbered the half-hex at the upper left corner A1, the 1 hex would be B2.



Now think about the next rotation of overlay 2, ovr2b. In this GIF, if the imaginary half-hex at the upper left were A1, the overlay's 1 hex would be B1. This is noted in the Data file for the “b” rotation of wd2.

Notice that the Data for wd1 has six coord entries, even though only the first is ever used—this single-hex woods overlay is never rotated, and it has only an “a” GIF in the zipfile. The last five coord entries could be replaced with spaces. (Up through VASL 3.01, dummy data was required to get some overlays to be recognized, which is probably what was done here.) For overlays such as ovr1 that have only two valid orientations, you'll see coords in the “a” and “d” positions and spaces in the others.

The Data coordinate gives only the approximate placement of the GIF, within one hex. To fine-tune the placement, you widen the transparent borders at the top and at the left. My procedure is to start with each GIF cropped as small as possible (i.e. minimum transparent area), write the Data file, build the zipfile, and do a test layout for the “a” orientations of the overlays on a desert board. If the data file's correct, the overlays will be slightly northwest of their intended positions. I save a PPM image of this test layout and open it in Paint Shop Pro.

Then I use the rulers to determine how many pixels east and how many pixels south each overlay needs to move. With this information, I open each of the “a” GIFs and add the appropriate widths of transparent border using Size Canvas. The process is repeated for the “b” images, and so on.

File naming conventions

Overlay GIF filenames such as ovrwd2a.gif should be entirely lower case. If there are capitals in these, VASL will have problems placing the overlay.

Zipfile names should be mixed case—ovrX, ovrWD, ovrOG, for example.

Creating a custom overlay file

To make custom overlays, first select a name that’s different from the standard ones. I’ll use the designator ZZ for this exercise. When I build the zipfile, it’ll be called ovrZZ. My overlay, a two-hex building, will have six GIFs for the different orientations which will be named ovrzz1a through ovrzz1f.

Start by getting a blank hexgrid, saving it as ovrZZ1a, and drawing the building outline and the outside boundary of the overlay. In this “a” orientation, hex 2 should be north of hex 1.

It’d be nice if there were a way to just paint this image and then rotate it to make the other five images, but it’s not that easy, at least for me. When you rotate an image at a nonorthographic angle like 60°, Paint Shop Pro antialiases several new colors, which goof up terrain change in VASL.

After saving this, copy it to ovrzz1b and ovrzz1c. Then remove the hexgrid from these two images with a color change. Rotate the “b” version 60° to the right, and rotate the “c” version 120°. Then clean up the dithered areas. (If anyone knows how to prevent Paint Shop Pro from antialiasing rotated images, I’d like to hear how it’s done.)

Paint the “a”, “b”, and “c” images (don’t forget to make the background transparent), then reapply the hexgrid and crop them. The “d”, “e”, and “f” images are copies of the first three, rotated 180°; make these.

Now write the Data file. For this two-hex overlay, the coords will be identical to other two-hex overlays. The file should look like this:

```
zz1 b2;b1;c1;d1;d2;c3
```

You can put a version designator at the bottom if you want, but it’s not necessary. The file should be called simply Data, with no filename extension.

Next, make the zipfile ovrZZ.zip. Add the six GIFs ovrzz1a-ovrzz1f and the Data file to it; make sure Compression is set to None when you do this. Close the WinZip window, then remove the .zip extension from the filename so it’s just called ovrZZ. Drag this file to the VASL\Boards\Overlays folder.

Now it’s time to do the initial test image. Start VASL and do a new scenario. Place board 30, then go to the Add Overlays box. Place ZZ1 in each of its six orientations: B2-B1, F2-G2, J2-K3, N2-M3, R2-Q2. Save a board image as test.ppm, then save the setup as test.vasl and exit VASL.

Open test.ppm in Paint Shop Pro, and turn on the rulers if you don’t already have them showing. Zoom in on the B2-B1 area, which shows the “a” image. Use the pointer tool and the rulers to determine how many pixels east and how many pixels south the overlay needs to move to align with the Board 30 hexgrid. (If it needs to move north or west, the overlay image may need to be cropped more closely, or you may need to change the coord for this orientation in the Data file.)

Now open ovrzz1a. Set your background brush to the transparent color, then select the Size Canvas command. Increase the overall width and height of the image by the proper number of pixels, and increase the left margin and top margin widths so that the pixels are added to these edges.

Correct all six GIFs like this, then zip them and the Data file into a new ovrZZ zipfile. Drag this to VASL\Boards\Overlays, overwriting the old file. Start VASL and load your test.vasl scenario. If you’re lucky, all of the overlays will align (almost) perfectly; if not, go back and tweak the GIFs a little more.