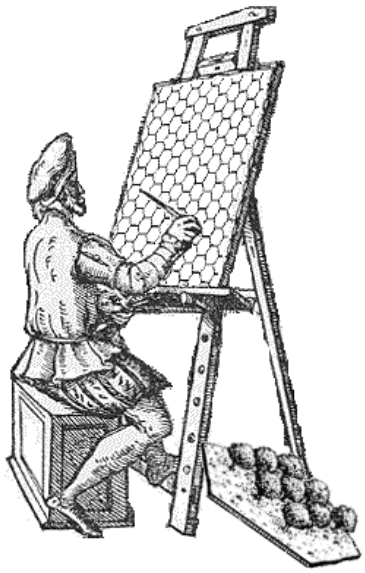
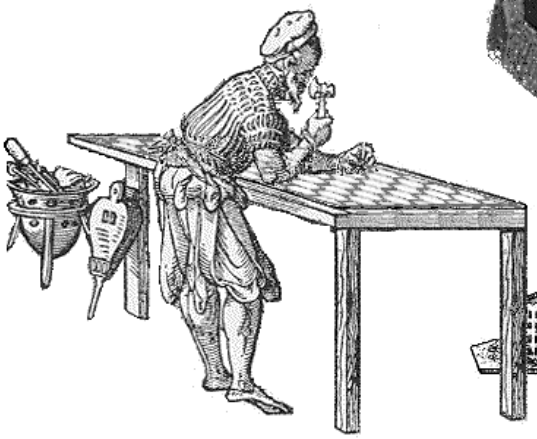
Ye Olde Booke



VASL MAP CABAL



of

VASL Mapmaking Secrets

Quick Start

1. Go to the VASL Map Cabal website or the VASL Map Cabal Yahoo Group (see p.2 for the URL's) and download the terrain libraries, color palette, and blank hexgrid for the geomorphic boards.

2. Go to Appendix C of this document and keep those pages close at hand while painting your map. Understand which colors are used for which terrain features.

3. Fire up your favorite paint program. See the section on "How to Paint a VASL Geomorphic Map" for a list of capabilities your paint program should have.

4. Open up the **HexgridForGeo.gif** file you downloaded in Step 1 and save it as another name like **boardX.gif**. Paint your map, making sure to save often and keep backup copies. Don't worry about coloring over the hexgrid; you'll restore it later. You may take Gordon Molek's enthusiastic advice to "paint naked!". Or not.

5. Open up the **HexgridForGeo.gif** file again. Copy and paste it onto your newly-painted map, restoring the hexgrid you painted over. See page 9 for details on how to do this.

6. Create a minimal **data** file, as described on p. 27. Save the **data** file and your new map together in an *uncompressed* zip file, then rename the file from something like **boardX.zip** to **bdX**. If Windows complains that terrible things can result if you remove the **.zip** extension, tell it you're really OK with what you're doing. Put the **bdX** file in your VASL **boards** directory.

7. Check your board. Fire up VASL, select "New Scenario" from the File menu, and call up your board. If VASL cannot find your board, you probably need to check your **bdX** file to make sure the board gif was indeed saved without compression and with the proper case. The gif inside the **bdX** file should be named **bdX.gif**, with the "bd" and "gif" in lower-case. Once VASL can find your board, check your colors by clicking the TerrainSSR button in the New Scenario dialog panel and run through all of the allowed terrain changes, making sure that your board changes color correctly. Go back and repaint your board as necessary, updating the **bdX.gif** file in your **bdX** zip archive.

8. Once your board is colored correctly, rotate it 180 degrees and save that version in your **bdX** zip archive as **bdrX.gif**.

9. Create the appropriate overlays which your board will need for making terrain changes like changing all Woods-Roads to Paths in the PTO, removing Bridges, etc. See Section 3 for details. You will also have to create the text files (**overlaySSR** and **SSRControls**) which control when and where the overlays are used; see page 28 for details. Check the overlays' positions and colors using VASL. Be sure to create reversed overlays as well. Save the overlay gifs in your **bdX** zip archive and check them in VASL like you did in Step 7 above.

10. Once you are satisfied with your board, send it out to other members of the VASL Map Cabal for their double-checking. Once everybody is satisfied, release the board to the raucous adoration of the ASL World. You go, Painter Dude.

Ye Old Booke of VASL Mapmaking Secrets

Table of Contents

1. How VASL Does That Voodoo It Does So Well 3
- 1.1. Java and Color Changes 3
- 1.2 The colors and colorSSR Files 4
- 1.3. The SSRControls and overlaySSR files 4

2. How to Paint a VASL Map 5
- 2.1. Your Paint Program 5
- 2.2. Stuff to Note 5
- 2.3. Working with the VASL Terrain Art Libraries 8
- 2.4 Scanning a Real Map and Working From There 8
- 2.5. Painting a Deluxe ASL Board 9
- 2.6. Restoring the Hexgrid Once You're Done Painting 9
- 2.7. Painting a HASL Map 9

3. Making Overlays 11
- 3.1. Making Board-Specific Overlays 11
- 3.2. The VASL Coordinate System 21
- 3.3. Making Generic VASL Overlays 22
- 3.4. Generic Overlay File Name Conventions 24
- 3.5. Creating a Generic Overlay 24

4. Putting It All Into A VASL bd File 27
- 4.1. Anatomy of a bdX File 27
- 4.2. Writing the data file 27
- 4.3. Writing the overlaySSR File 28
- 4.4. Writing the SSRControls File 28

5. Deep Background on How VASL Works 30
- 5.1. How VASL Creates a Board Configuration 30
- 5.2. Overlay or Color Change? Two Case Studies 30
- 5.3. Special Overlay Precedence 31
- 5.4 Uncle Rodney's Bigass List of VASL Terrain Changes 33
- 5.5. Making Your Own Terrain Changes 35
- 5.6. Master's Degree Thesis Projects 35

6. Printing VASL Maps 36
- 6.1. Converting to Full-Sized ASL Hexes 36
- 6.2. PDF Conversion 36
- 6.3 Greyscale Values 36

7. How to Make VASL Counters 38
- 7.1. Making Counters in VASL v3 38
- 7.2. The VASLdata File 40
- 7.3. Nationality Colour Chart 43
- 7.4. Making Counters in VASL v4 43

Appendix A - VASL Credits 44

Appendix B - The Old VASL Geo Palette 45

Appendix C - The New VASL Geo Palette 49

Appendix D - TJ Fachko's Geo Workshop 56

Appendix E - Historical Note: The Perl/Postscript Map Generator 70

Welcome

VASL is one of the coolest ASL products ever, in part because of the beauty of the hand-drawn maps which Bruce Mansfield unveiled to an appreciative VASL community in the late 1990's. After some initial resistance, the VASL Powers That Were finally got with the program and took on the challenge of enhancing the VASL experience with hand-drawn maps and overlays. VASL version 3 set the ASL world on its ear with stunning graphics that allowed terrain changes to be shown directly on the map instead of requiring the players to remember them.

Thus the VASL Map Cabal was born, with the purpose of serving the ASL Community with new geomorphic and HASL boards as they are published. This document is intended as a repository of the secrets of the VASL Mapmaker's Guild; it is passed down from greybeard to novice in the hope of preserving the mondo coolness of VASL for future generations.

Ars longa, vita brevis
(Art is long, life is short)

- motto of the VASL Map Cabal

Conventions

File names are given in **bold**.

Text appearing in various data files is given in
`Courier font on a light grey background.`

Figure captions are given in ***bold italics***.

Transparent colors in images are shown as white, not the pink color of Transparent.

Figures that show images in their entirety are enclosed in thin black borders, mostly to show the extent of the entire image, including transparent sections.

Figures that show only one part of a larger image are shown without a border.

Web Links

VASL
http://www.vasl.org

The VASL Map Cabal
http://www.communique.se/dreadtom/cabal/

The VASL Map Cabal Yahoo Discussion Group
http://groups.yahoo.com/group/vaslmapcabal
You must Sign In to Yahoo to join this group.

Ye Old Booke of VASL Mapmaking Secrets
version 6 November 2001

"Why yes, my dear… I <u>am</u> with the VASL Map Cabal"

Creative minds have always been known to survive any kind of bad training
- Anna Freud

VASL Mapmaking Secrets

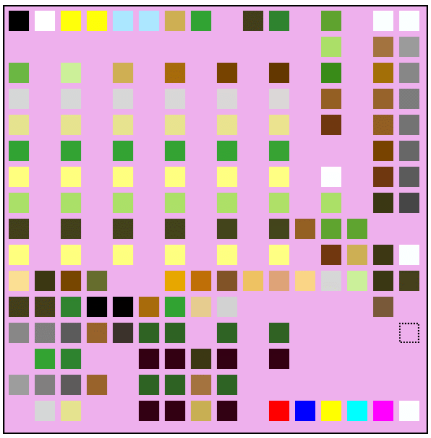
1. How VASL Does That Voodoo It Does So Well

VASL is able to do two things to alter the boards that are displayed for the user: it can change the color of the displayed board image and it can add overlay images on top of the basic board image. We try to use whichever method makes the most sense for a particular situation. When an effect can be achieved by global color changes, we do that. When the desired effect is too complicated or impossible to do this way, we simply create one or more overlays[1].

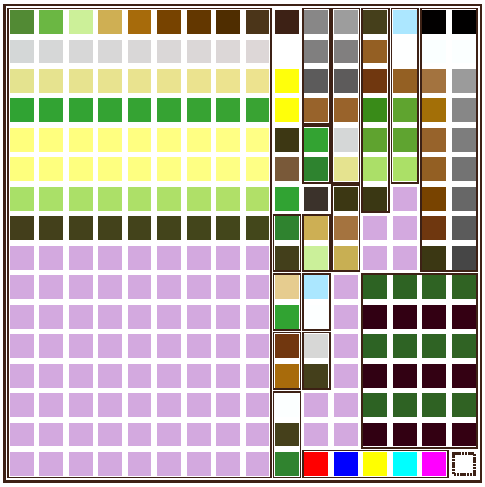
1.1. Java and Color Changes

Java (the language VASL is written in) has built-in classes which allow us to select pixels having certain Red/Green/Blue (RGB) triplets and operate on them in various ways, including changing them to other RGB triplets for display. This allows us to change the colors of the displayed VASL maps[2] to achieve various terrain changes mandated by SSR or weather, but it requires a standardized palette to be established for the colors drawn on VASL maps[3]. Every Level 3 Paved Road has to have the same RGB triplet on every board in the system; same goes for every Level -1 Open Ground hex, every Level 0 Marsh, etc.

When VASL v3 was released in the late 1990's, the standard palette[4] was the one shown at right. This is the *old geo palette*, which covered all of the terrain types then present in the system in a somewhat haphazard and unorganized, yet good-hearted, fashion. The old palette was used for geomorphic boards numbered 1 to roughly 48 and is described in detail in Appendix B. Note that the unused color slots are shown in light pink because white didn't seem like a good option; the only *real* pink color is the bright one in the lower-right corner.



The old geo palette



The new geo palette

Around the release of geo board number 49, the luminaries at the VASL Map Cabal decided to expand and reorganize the geo palette to make it easier to use while allowing for a greater range of terrain types. Thus was born the *new geo palette* seen at left and explained in detail in Appendix C. While the old palette covered levels -1 to 4 and got rather disorganized near the bottom, the new palette can handle terrain levels -2 to 6 and has a lot of room for easy future expansion. Life is good with the new palette.

Note again that the unused color slots are shown in pinkish-purple here; the only real pink color is again at the lower-right.

---

[1] While the full-color board files can be several hundreds of K in size, a mostly-transparent overlay can be only 10K in size, even if it covers the entire board. This enables some nice effects to be achieved while keeping the board archive files as small as possible.

[2] The map files that are saved on disk are not altered; only the *displayed* version of the maps are altered.

[3] Note that VASL will display maps just fine if they don't use the standard palette, but the terrain changes produced by swapping RGB values will simply not work.

[4] Also known as the "drawing palette" since it dictates which colors a VASL map should be drawn with, as opposed to the "display palette", which is all the colors that VASL may wind up displaying on the screen after it does the various color changes requested by the user. The drawing palette is limited to 256 colors since the gif file format is limited in this way; the display palette is limited only by the number of colors the user's screen can display.

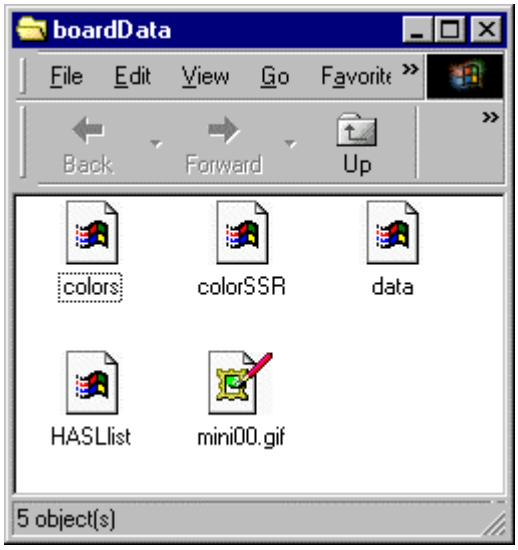
1.2. The colors and colorSSR Files

Inside the **boardData** archive in your VASL/boards directory are two files named **colors** and **colorSSR**.[5] The **colors** file tells VASL what the drawing palette colors are; here we learn that all Level 3 Paved Roads have the RGB value of 218, 215, 215 and Level -1 Open Ground is 107, 183, 67.[6] The **colors** file also defines the colors in the display palette, specifying a color named "MudPavedRdL3" as 178, 175, 175. The **colorSSR** file tells VASL when to swap one color for another; here we learn that under "Mud" conditions, the RGB triplet specified by "PavedRdL3" should be displayed as "MudPavedRdL3". Thus, when Mud is in effect, VASL finds all pixels with RGB values of 218, 215, 215 and displays them as 178, 175, 175, which is the desired color for Level 3 Paved Roads under Mud. Ta-daah!

Go ahead and poke around the **boardData** archive and examine the structure of these files. First, copy the file from your VASL/boards directory into somewhere safe. Then add a .zip extension to this file and unzip it. You'll see the folder looks like that on the right. The **colors**, **colorSSR**, **data**, and **HASLlist** files are text files without the .txt extension. We'll talk more later about the **boardData** archive and how you create the files inside it.

We'll get into more detail on all this stuff later; for now, all you need to see is that the gif and text files in the **bdX** and **boardData** files tell VASL how to alter the boards that are displayed for the user, achieving those mondo kewl effects that VASL users love. It is not absolutely necessary to have this stuff in there; you can create any old gif using any old colors and slap it in the **boards** folder with a name like **bd999.gif** and VASL will display it just fine[7]. But putting it all together correctly to take full advantage of all that VASL can do is very challenging and highly rewarding.

It is the goal of this Booke to show you how to do this. Listen well, Young Skywalker.



Contents of the boardData archive

1.3. The SSRControls and overlaySSR files

Like we said above, VASL achieves some of its effects by using overlays instead of color changes. The **SSRControls** and **overlaySSR** files tell VASL what overlays to put down on the board under what conditions. Like the color changes, these overlay changes are transparent to the user; all he has to do is specify a terrain SSR like "Dense PTO" and all of the necessary color and overlay changes are done for him. We'll talk more about these files in Sections 4.3 and 4.4; we just wanted to mention them here for completeness.

[5] Everything in the VASL zip archives is case-sensitive. This is a huge pain in the ass, but that's java for you. If you're having trouble getting VASL to read your board or implement the changes you want, odds are that something somewhere is spelled with the wrong case. Aargh.

[6] Each value in a RGB triplet is in the range 0-255, with 0,0,0 = black and 255,255,255 = white.

[7] VASL v4 only, actually; v3 requires that you create a minimal **data** file and add it to the **bdX** archive, as described on p. 27.

VASL Mapmaking Secrets

2. How to Paint a VASL Map

2.1. Your Paint Program

We're not going to spend time in this document giving tips for any specific paint programs - that could involve us in a morass of legalistic wrangling between the makers of graphics applications looking for free product endorsements, and the Cabal lawyers tell us they're tired of "dragging our sorry artist asses out of the fire," whatever that means.

Still, there are some capabilities your paint program should have. It should be able to read and write PPM and gif files, do pattern fills, handle transparent colors, readily show the RGB value for some pixel on the screen, easily handle working with a color palette and cutting and pasting from one image to another, and globally replace one color with another. Well, actually, *you* should be able to *make it* do these things, because they'll come in handy.

Suggested applications:
Windows: PaintShop Pro (Jasc Software), Photoshop (Adobe)
Macintosh: Photoshop (Adobe), GraphicConverter (Lemkesoft)

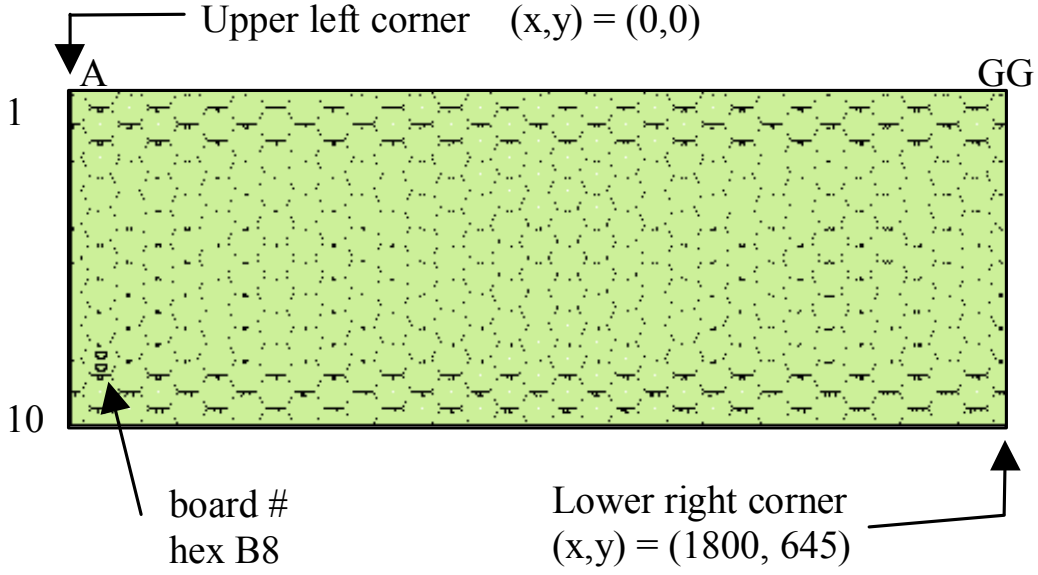
2.2. Stuff to Note

As you paint, keep this Booke open to Appendix C because you'll be referring to it constantly to make sure you're using the right colors. Before you venture forth, it's a good idea to read TJ Fachko's Geo Workshop in Appendix D to get an idea of the task you'll be facing. Aye, sure'n but it's a labor of love, it is…

2.2.1. Geo Board Details

Standard-sized VASL geo maps are 1800 x 645 pixels in size, 33 x 10 hexes. There is a 1-pixel-wide black frame around the board. The geomorphic parts are:

- Woods half-hexes along the top (F0, H0, J0, L0, V0, X0, Z0, BB0),
  bottom (F10, H10, J10, L10, V10, X10, Z10, BB10),
  left side (A4, A7), and
  right side (GG4, GG7)
- Roads entering I1, Q1, Y1, I10, Q10, Y10, and A5/A6, GG5/GG6 (the last two are along the hexside)
- For river boards, rivers entering along A4-A7 and GG4-GG7



VASL Geo Board Dimensions in x,y pixel coordinates

VASL only works with long orientation boards like that shown above. At one point, Rodney did work up the ability to work with tall orientation (rotating the long orientation by 90 degrees), but the Mac Cabaleers reported very slow execution speed, so it was dropped.

The **HexgridForGeo.gif** file available on the Cabal's websites already has the basics of the geo board, with the New Geo Palette already loaded and Open Ground Level 0 already filled in. It doesn't have the geomorphic woods and road hexes, but those are up to the board artist to fill in anyway.

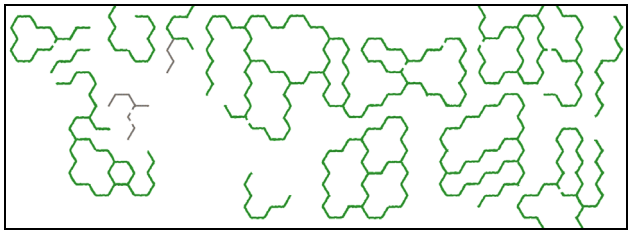
VASL Mapmaking Secrets

2.2.2. Working With Layers

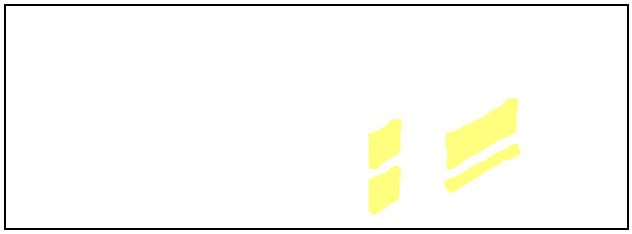
Layers are invaluable for painting maps. They let you organize your painting process and allow you to make mistakes in one type of terrain without affecting the other terrain lying nearby or below it. Saves a *lot* of erasing and redrawing. You **need** to use layers. Get comfortable with them. Find a web tutorial on them. Ask other Cabaleers about them. But *don't* paint a map without them. Trust us.

Here are the nine layers used to draw Ian Daglish's Db1 Bocage map, and the final map with the layers merged:

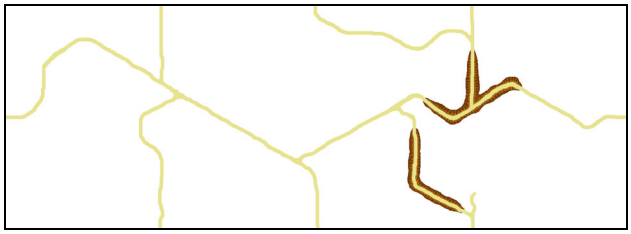
Buildings



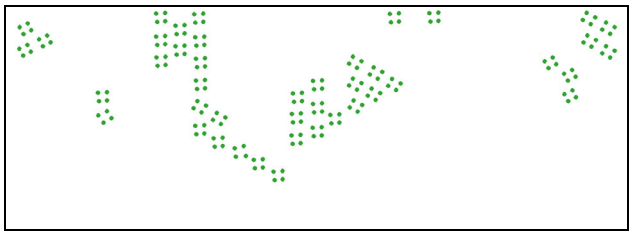
Walls and Hedges



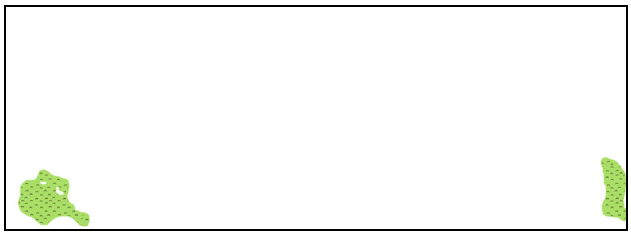
Grain



Roads

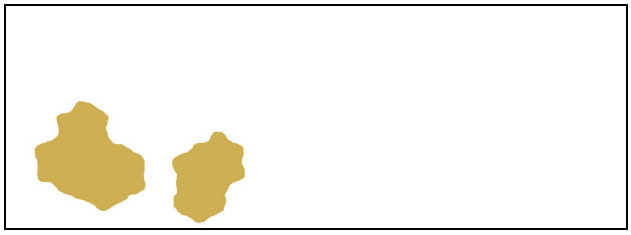


Orchards

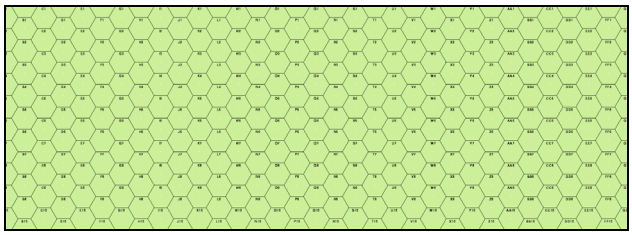


Brush

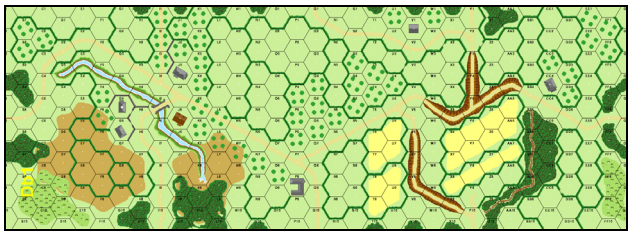
Woods



Hills



Background



All Layers Merged

2.2.3. Miscellaneous and Sundry

When doing a VASL version of a real hardcopy board, you need to try to be true to the real board in order to make LOS as equal as possible between the real board and its VASL cousin. In that vein, working from a scanned board (or a CAD version of a scanned board) can be a real boon. Remember, too, that ASL'ers care a lot about whether terrain touches a hexside or not, so watch those hexsides.

It's also big fun to draw your own geo maps. If you want people to actually *play* on them, though, you will have to make the map playable. That is, watch out for creating nonstandard terrain types like paths through grain, roads through crag, hexside gullies, and the like. While such things are neat to create and push the envelope of the system in often cool ways, remember that ASL'ers are going to want a clear understanding of how to play on your new terrain types. Don't just assume that it's obvious; support your new terrain with a text file explaining all of the details that one needs to know: stacking and movement limitations, Bog checks, how the terrain behaves in Mud and Snow, etc.

Even if you use completely standard ASL terrain, your geo board is going to be scrutinized for unrealistic behavior. Why does the road go *that way* when it would have been far less effort to build the road over *there*? Does water really flow that way? Would somebody really build a farmhouse *there* or plow *that* patch of ground when it's so much easier to go over *there*? Does that look like a realistic road net for a town? Believe it or not, people really do care about these things.

You should also watch for dominating terrain and choke points. Do you really want one hex or one building to dominate the local area or the whole board? It's OK if you do (think building 3M2), but you need to be aware of it. Too many choke points, can make your board dull to play on - try to create several ways to get across the board vertically and horizontally. Remember, too, that less is more - if somebody wants to add something to your board, they can always use an overlay. Feel free to use the Cabal clip art, especially for buildings.

2.3. Working with the VASL Terrain Art Libraries

For those of us who can't draw worth a lick, the VASL Terrain Art libraries are *highly* recommended. There are libraries of walls, hedges, orchards, buildings, railroads, and other stuff to save you all kinds of time putting together a map that actually looks good. There are also files that contain nothing but the woods, brush, marsh, and grain patterns to make it easy to draw pattern-filled terrain. The libraries can be downloaded from either Cabal website.

Be sure to watch the transparency colors when you cut and paste from the library file into your map file; the transparency colors should be the same on both files in order to ensure that the clip-art pastes properly into your map with only the colors you want showing.

2.4. Scanning a Real Map and Working From There

When making a VASL map of a hardcopy board, some people like to work from a scan of the map. One advantage of this is that the buildings and other terrain are more likely to be positioned correctly in the hex, making the VASL map's LOS more like the real board's. One downside of scanning is that large constant-color areas like grain and open ground give you many more colors than you'd expect, and there can be a "screening" effect that has to be removed.

Advice from Carl Pfeiffer:

> I've had good results scanning at 100 dpi. I figure VASL board resolution is about 83 dpi, since a geo board hex is a hair over 3/4", call it 25/32", and a VASL geo board is 645 pixels tall.
>
> I've always stitched scans together with a VASL hexgrid overlay as the first step in the process, but that's because I like to trace the terrain boundaries in a layered image rather than painting freehand. Doing it that way, I lose any detail gained by scanning at higher resolution when I reduce the image size to fit the hexgrid, so it doesn't make sense for me to scan at higher resolution. If you're painting freehand and keeping your scan open in a second window for reference, though, it might help to retain more detail when you zoom in tight on your reference image.

Art is a step from what is obvious and well-known toward what is arcane and concealed - Kahlil Gibran

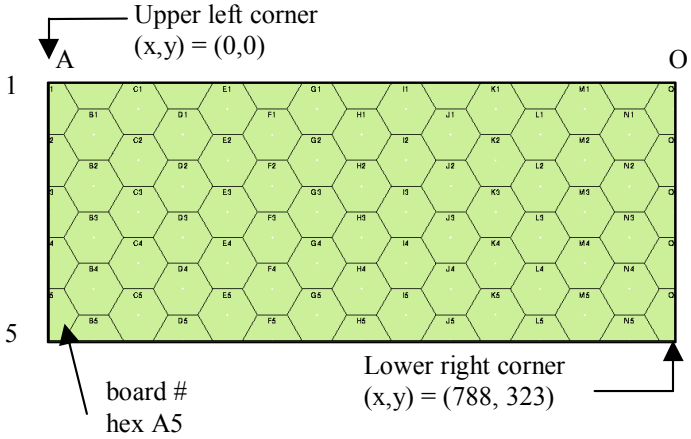
VASL Mapmaking Secrets

2.5. Painting a Deluxe ASL Board

DASL boards are essentially miniature geo boards, but of course there's differences. The DASL gif size is 788 x 323 pixels with a 1-pixel wide black border along the outer edge, 5x15 hexes (hexrows A-O). While it's not a requirement, DASL boards are generally named **bddxX.gif**, with the added "dx" in there just to help us remember that it's DASL. When VASL opens the board archive, it reads the board size directly from the board gif itself, so there's nothing special you need to do to tell VASL to read a DASL-sized (or HASL-sized, for that matter) board.

The geomorphic parts of a DASL board are:

- Woods half-hexes along the top (F0, J0), bottom (F5, J5), left side (A2, A4), and right side (O2, O4)
- Roads entering A3, H0, H5, and O3.



VASL DASL Board Dimensions in x,y pixel coordinates

2.6. Restoring the Hexgrid Once You're Done Painting

Open up the **HexgridForGeo** gif file again and make sure that the open ground color in that image is set to transparent. Make sure that the transparent color on your newly-painted map is the same as that for the blank hexgrid. Copy the entire blank hexgrid from the **HexgridForGeo** file and paste it onto your newly-painted map. While the transparent hexgrid is still selected, drag it so that it aligns perfectly with the hexgrid on your new map - the best way is to align the hex ID numbers. Un-select the transparent hexgrid you just pasted and it will be overlaid on your new map, restoring the parts of the hexgrid that you painted over.

2.7. Painting a HASL Map

Don't sweat the geo palette unless you want your HASL board to be usable for other scenarios outside the scope of your HASL work where those color-based terrain changes might come into effect. On the other hand, unless your artistic enough to come up with your own good color scheme, there's something to be said for keeping to the standard colors that VASL players are used to.

HASL map archives are just regular **bdX** files.

Advice from Carl Pfeiffer:
The first step is always to get the hexgrid you need. That's usually done by taking a copy of one of the big hexgrids (from one of the VASL Map Cabal websites) and cropping it to size. Name that copy comething like **bdHB.gif** and stick it in your VASL/Boards folder, then load it up in VASL and move some counters around on it (see footnote 7 on page 4). If it looks like the counters align properly at the four corners (they should, if you're using a standard hexgrid), you can just start painting on the hexgrid. Keep a blank copy of the hexgrid and paste it back in periodically to repair the grid.

Some things I can think of that are specific to HASL maps:

- Multitoned terrain. Mixing two or three subtly different shades in nonpatterned terrain such as OG or roads gives a more interesting coloration IMO, and it lets you suggest contours. I think Gordon Molek did this first, on BRT. I stole it shamelessly for Cheneux, using it to soften the edges of the roads as well as on the OG (hard to see with the dark color scheme, though). For a really spiffy application of this technique, check out BDF in snow colors--TJ Fachko used a light grey to give what looks to me like ditches or snowbanks alongside the roads.

- Banners and title blocks. They look cool, and can be a handy place to include a HASL-specific table or rules cite along with the map title and copyright information.

- Unusual Terrain Changess, mostly by overlay. RB gutted factories, flooding the river in Fortress Cassino, Mist color change in the KGP boards (didn't work all that well, though), a couple of terrain variants in Baraque de Fraiture.

- Shadowing of buildings to give a 3D appearance. A little of this can lend a lot of character to a map. This can be done with woods and orchard trees, too. Look at Baraque de Fraiture, Cheneux, and Stoumont for some different approaches.

More advice from TJ Fachko:
There's not much more to a HASL board than what you do on a geo board - the *big* difference is that you're less constrained by convention, i.e. you can use whatever colors you want and even go so far as to change the look of terrain features to suit your tastes. I find it a great excercise in creativity. Take the Devil's Den map for instance - there are terrain features that I can represent with geo art as I've done with the rocky/rough (crag/hammada) parts yet others I'll have to draw on my own (like the wooden fence). The colors aren't your standard geo tones so there's a bit of a challenge for me in choosing colors that look right while maintaining the spirit of the paper map, something I like to do. Another individual might want to dream up his/her own colors; doing HASLs allows this.

Carl Pfeiffer receives divine inspiration for the Cheneux map

An artist cannot speak about his art any more than a plant can discuss horticulture
- Jean Cocteau

VASL Mapmaking Secrets

3. Making Overlays

There are two types of overlays - board-specifc overlays that only go on one particular board and are kept in the **bdX** archive, and generic overlays (like the hardcopy ones that come with the ASL modules themselves) that go with any board and are kept in the VASL/boards/overlays directory.

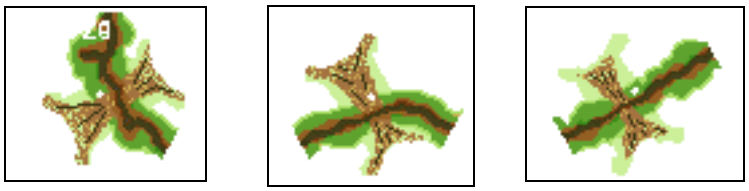
3.1. Making Board-Specific Overlays

Once you paint a VASL board, you're not done. You need to make the overlays that VASL needs to implement various terrain changes like "Dense Jungle PTO", "All Bridges are Fords", etc. Most boards only need a few of these board-specific overlays, but if your board has bridges, roads, sunken roads, woods, woods-roads, cliffs, or brush, then you need to think about making board-specific overlays:

- Ford
- NoBridge
- NoHills
- Stupid Road Tricks (SRTs) (VASL v4 only)
- RoadsToPaths
- NoWoodsRoads
- NoCliffs (VASL v4 only)
- Dense Jungle (VASL v4 only)
- Bamboo (VASL v4 only)
- Palm Trees (VASL v4 only)

3.1.1. Ford

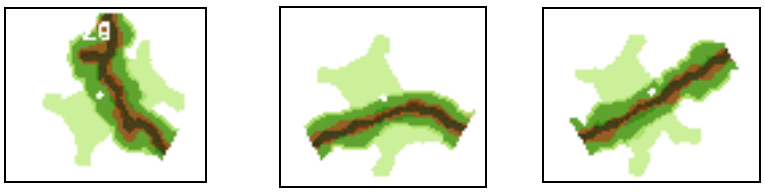
Under PTO conditions, Bridges turn into Fords. Use Path colors for the vaguely hourglass-shaped Ford. Like the NoBridge overlays, the best way to work up a Ford overlay is to draw the bridge over the Gully or Stream, then copy the bridge area and save it as a Ford overlay gif (remembering to note the x,y coordinates of the upper-left-hand corner like Section 3.1.11 says). Change all of the bridge colors to the correct Open Ground color (usually level 0; note the level 0 light green areas in the three Ford overlays below which will cover up the bridge art when overlaid on the bridge hex) and then fill in the Gully/Stream art with the Ford going across it.



The Ford overlays for board 5: z9 (left), y8 (center), and ee2(right)

3.1.2. NoBridge

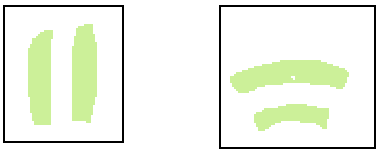
Just like Fords, except NoBridge is only invoked by SSR.



The NoBridge overlays for board 5: z9 (left), y8 (center), and ee2(right)

3.1.3. NoHills

NoHills overlays are needed when sunken or elevated roads occur on any level besides Level 0. That's because when the user selects the "No Hills" Terrain SSR, VASL changes all hill and valley colors to Level 0, effectively changing the non-Level-0 sunken or elevated road to a Level 0 sunken or elevated road, which is not correct. The fix is to put down special NoHills overlays that cover the sunken or elevated roads on hill or valley levels with level 0 open ground. See Section 5.2 for more details on how this works on board 41.



W7 *AA5*

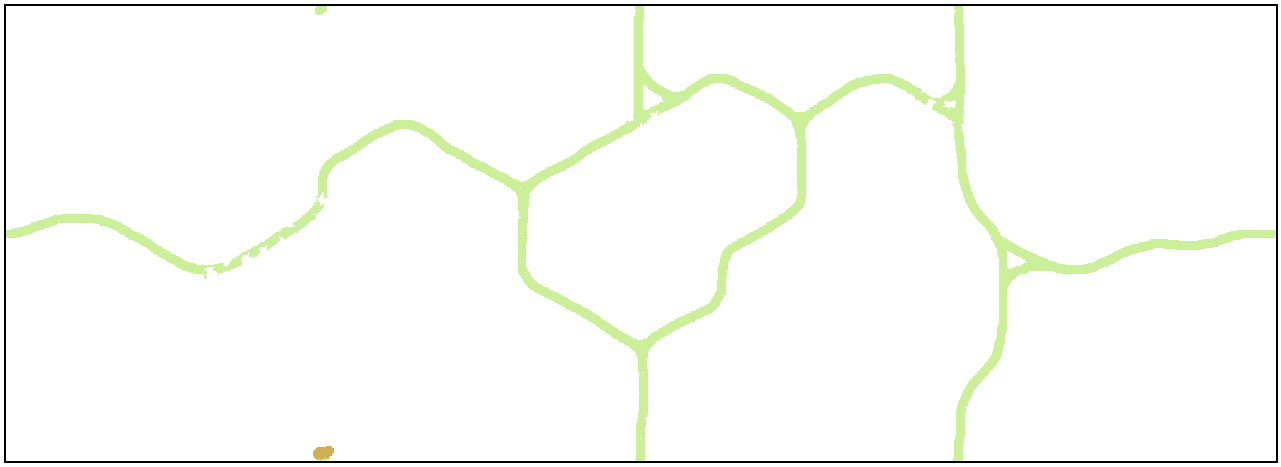
bd41 NoHills Overlays

3.1.4. Stupid Road Tricks (SRT's)

Stupid Road Tricks is an example of an *indulgence* - something that only one of the Cabal members really wants, but it's just so near and dear to his heart that the rest of the group just lets him go for it. SRT's allow the user to keep only certain sections of the road net so as to enable those scenarios that say "Only the road from A to B exists". There was much discussion about the best way to implement SRT's, but what eventually won out was the following scheme.

When the user wants to keep the road section from hex A to hex B, VASL first wipes out all of the roads on the board with a board-specific NoRoads overlay, then wipes out all Woods-Roads with the same overlay used to eliminate Woods-Roads under PTO conditions and then puts the desired road section back onto the map with a special SRT overlay that has only the desired road section on it. See Section 5.2 for more details on why this convoluted scheme is done the way it is. To keep a road from A to B to C, then, one keeps the road sections A-B and B-C.

To make SRT overlays, one needs to flense (see Section 3.1.12) the board so that only the road net is showing. Save this file for later. Then change the colors on the road net to their level-specific open ground equivalents and save this file as **SRTbdX.gif**. This is the board-specific NoRoads overlay mentioned above. Here's what the **SRTbd03.gif** file looks like, with nearly all of the roads replaced by their level 0 open ground equivalent; the little road blip going up to level 1 in hex I10 is colored in Level 1 open ground. The mangled-up roads around H5 are from Shellholes.



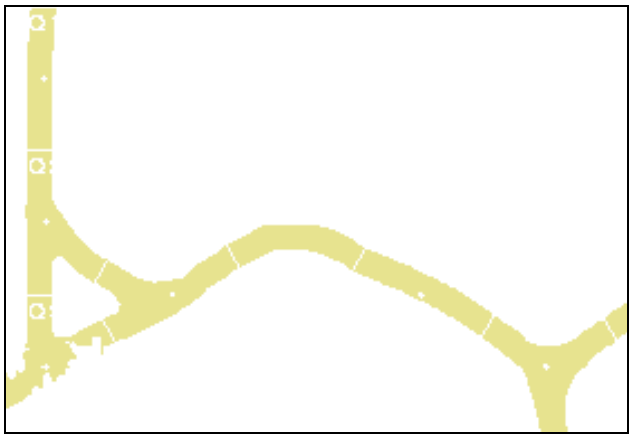
SRTbd03.gif

Now open up the original road net file that you saved earlier and chop up the road net into individual SRT overlays, one for each piece of road going from one intersection to the next. Save these overlays with names like **SRTA_B.gif**, where A and B are the hex numbers of the start and end points of that piece of the road net. Urban areas of semi-urban boards like boards 10, 12, and 46 can be combined into one big chunk called something like **SRTurban.gif**; use your own judgement about where the "urban" part of the road net meets the "rural" part. Entirely urban boards like 1, 20, 21, 23, 45, and 51 are simply too hard to do SRT's in, so we don't even bother.

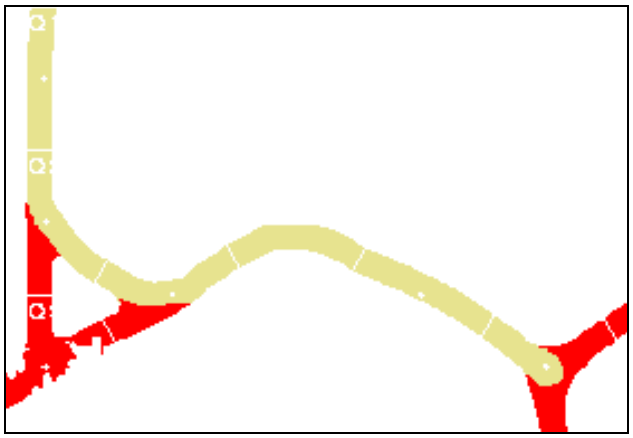
I am enough of an artist to draw freely upon my imagination. Imagination is more important than knowledge. Knowledge is limited. Imagination encircles the world - Albert Einstein

VASL Mapmaking Secrets

Next, you're going to have to tweak these overlays because the roads in each SRT overlay can only go to the center of each intersection, and the files you created above have these little spurs running out from the either side of the intersection. So open up each SRT overlay file and prune off the unwanted pieces so that the SRT road goes to the intersection point and stops. Yeah, the final reconstructed SRT road won't look as natural as the original road, but them's the breaks.

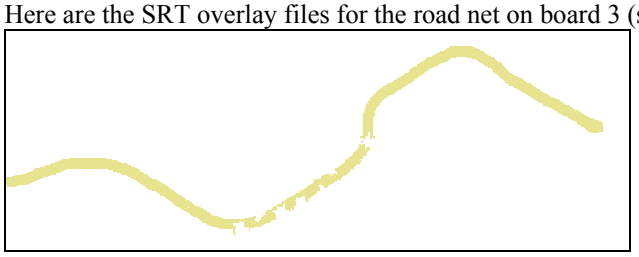


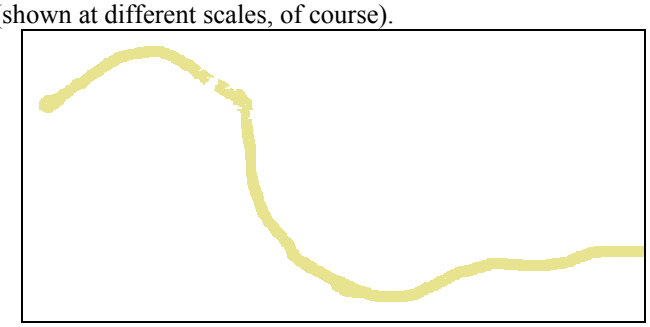
Section from 3Q1 to 3U3 as cut from the flensed road net

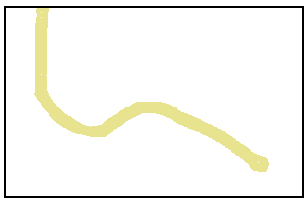


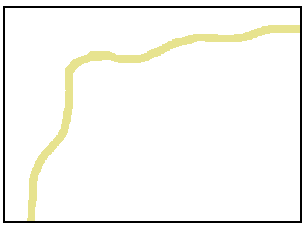
Erase the unwanted parts (shown in red here) to create the SRTq1_u3.GIF file for board 3

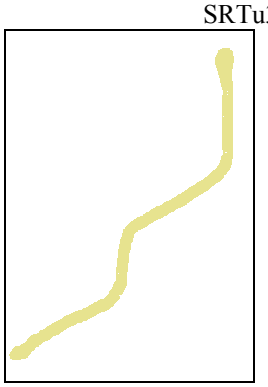
Here are the SRT overlay files for the road net on board 3 (shown at different scales, of course).

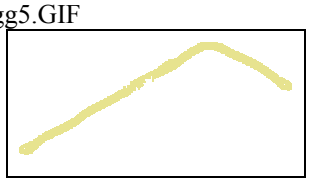

SRTa5_n4.GIF

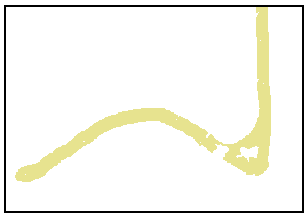

SRTu3_gg5.GIF

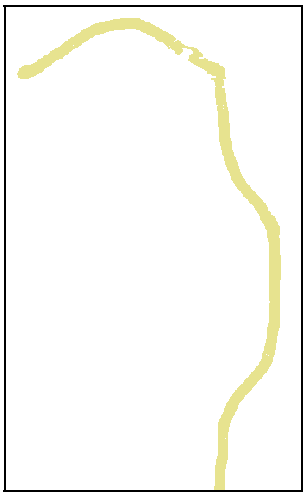

SRTq1_u3.GIF

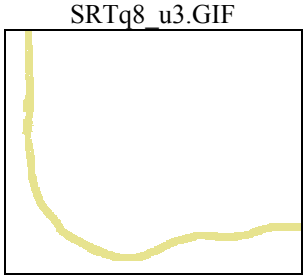

SRTy10_gg5.GIF

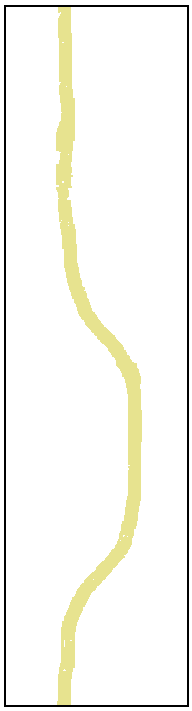

SRTq8_u3.GIF

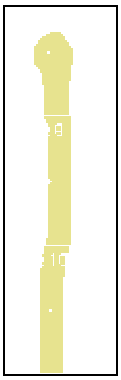

SRTn4_u3.GIF

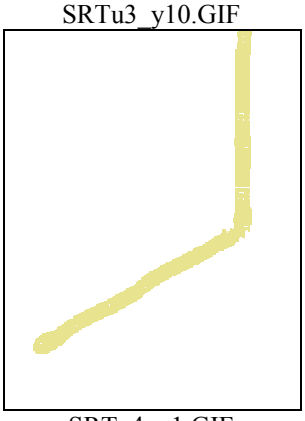

SRTu3_y1.GIF

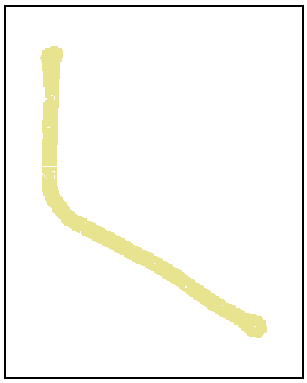

SRTu3_y10.GIF


SRTy1_gg5.GIF


SRTy1_y10.GIF


SRTq8_q10.GIF


SRTn4_q1.GIF


SRTn4_q8.GIF

The first prerogative of an artist in any medium is to make a fool of himself
- Pauline Kael

VASL Mapmaking Secrets

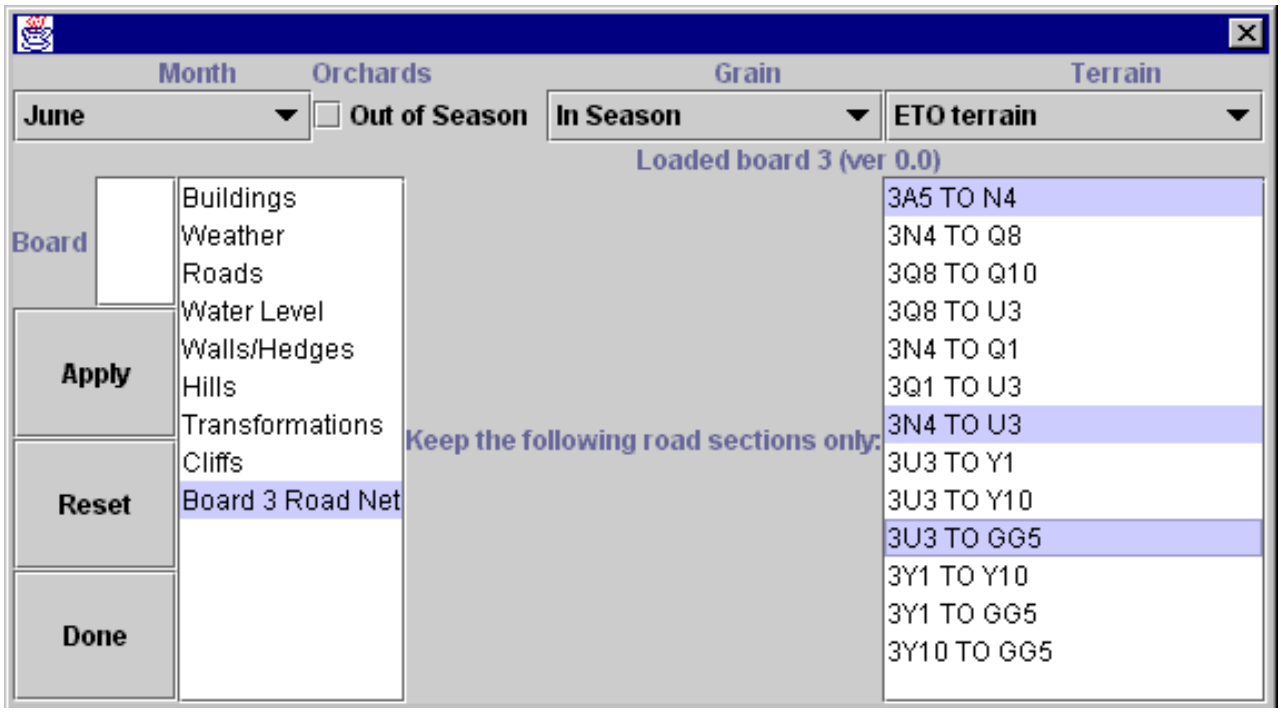
As explained in Sections 4.3 and 4.4, the **overlaySSR** and **SSRControls** files tell VASL how and where to put down any kind of overlay, including SRT overlays. The SRT sections of these files for board 3 are given below:

```
SRTa5_n4
SRTa5_n4.GIF 0,144

SRTn4_q8
SRTn4_q8.GIF 696,225

SRTq8_q10
SRTq8_q10.GIF 876,461

SRTq8_u3
SRTq8_u3.GIF 888,133

SRTn4_q1
SRTn4_q1.GIF 700,0

SRTq1_u3
SRTq1_u3.GIF 860,0

SRTn4_u3
SRTn4_u3.GIF 704,80

SRTu3_y1
SRTu3_y1.GIF 1104,0

SRTu3_y10
SRTu3_y10.GIF 1100,84

SRTu3_gg5
SRTu3_gg5.GIF 1096,81

SRTy1_y10
SRTy1_y10.GIF 1296,0

SRTy1_gg5
SRTy1_gg5.GIF 1312,0

SRTy10_gg5
SRTy10_gg5.GIF 1304,288

EraseRoads
SRTbd03.GIF 0,0
```

The SRT section of the overlaySSR file for board 3. Note the overlay coordinates as specified in Section 3.1.11.

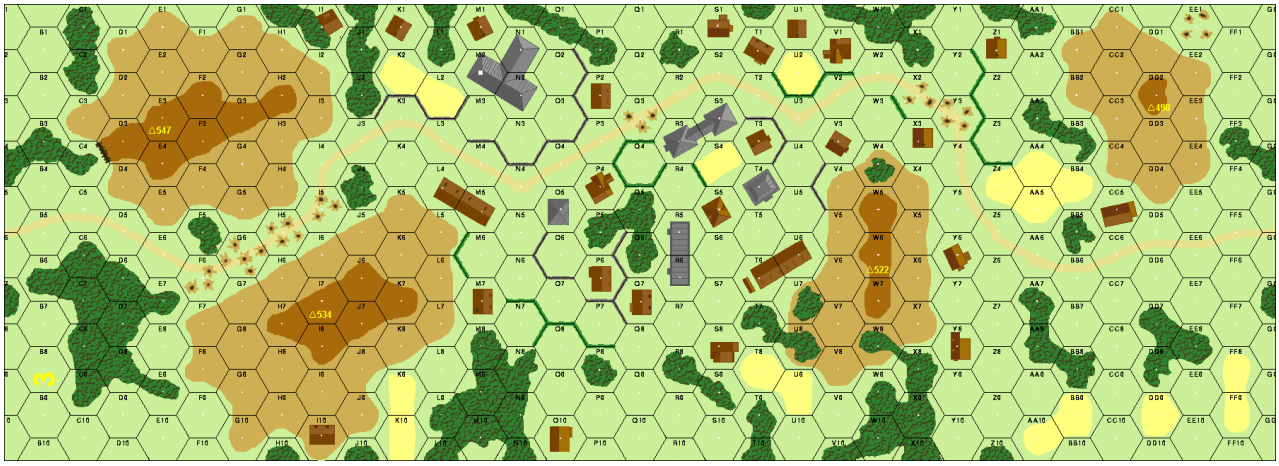
```
<SSRControls>
<Option name = "Cliffs">
<Checkbox>
<entry name = "No Cliffs" rule="NoCliffs" text="Remove all cliffs"/>
</Checkbox>
</Option>
   <Option name="Board 3 Road Net">
      <ScrollList name="Keep the following road sections only:" >
<entry name="3A5 TO N4" rule="EraseRoads,SRTa5_n4" text=" 3A5-N4 "/>
<entry name="3N4 TO Q8" rule="EraseRoads,SRTn4_q8" text=" 3N4-Q8 "/>
<entry name="3Q8 TO Q10" rule="EraseRoads,SRTq8_q10" text=" 3Q8-Q10 "/>
<entry name="3Q8 TO U3" rule="EraseRoads,SRTq8_u3" text=" 3Q8-U3 "/>
<entry name="3N4 TO Q1" rule="EraseRoads,SRTn4_q1" text=" 3N4-Q1 "/>
<entry name="3Q1 TO U3" rule="EraseRoads,SRTq1_u3" text=" 3Q1-U3 "/>
<entry name="3N4 TO U3" rule="EraseRoads,SRTn4_u3" text=" 3N4-U3 "/>
<entry name="3U3 TO Y1" rule="EraseRoads,SRTu3_y1" text=" 3U3-Y1 "/>
<entry name="3U3 TO Y10" rule="EraseRoads,SRTu3_y10" text=" 3U3-Y10 "/>
<entry name="3U3 TO GG5" rule="EraseRoads,SRTu3_gg5" text=" 3U3-GG5 "/>
<entry name="3Y1 TO Y10" rule="EraseRoads,SRTy1_y10" text=" 3Y1-Y10 "/>
<entry name="3Y1 TO GG5" rule="EraseRoads,SRTy1_gg5" text=" 3Y1-GG5 "/>
<entry name="3Y10 TO GG5" rule="EraseRoads,SRTy10_gg5" text=" 3Y10-GG5 "/>
      </ScrollList>
   </Option>
</SSRControls>
```

The SSRControls file for board 3, showing the rules invoked for creating SRT's.
The SRT section of the dialog box created by this XML code is shown below.



Selecting the desired road sections to keep on board 3 (VASL v4)

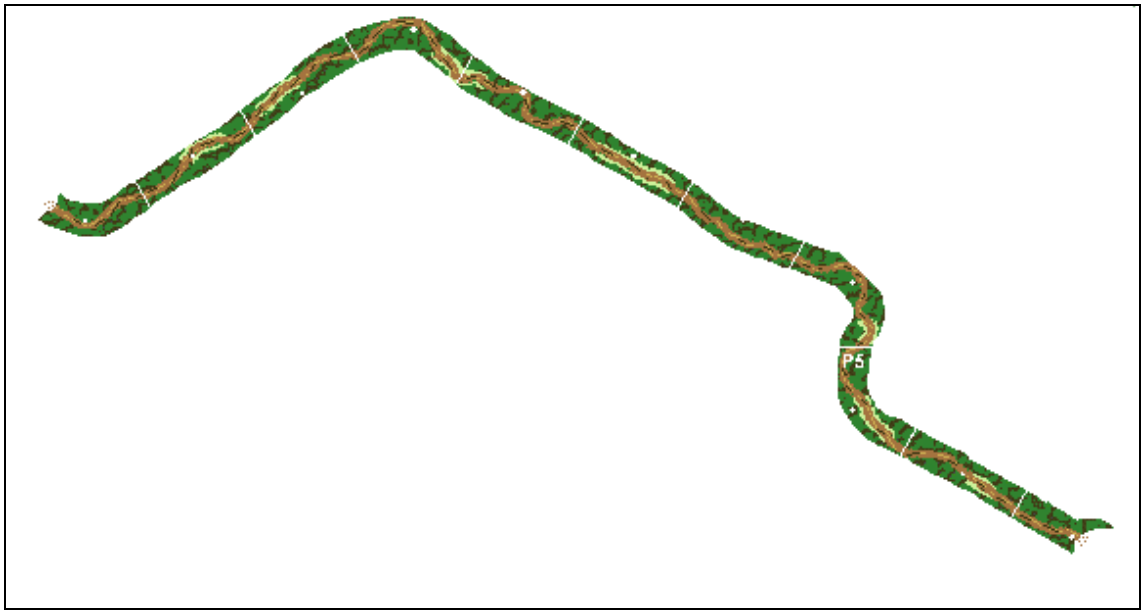
PAINTING, n. The art of protecting flat surfaces from the weather and exposing them to the critic - Ambrose Bierce



Board 3, keeping only the road from A5-N4-U3-GG5

3.1.5. RoadsToPaths

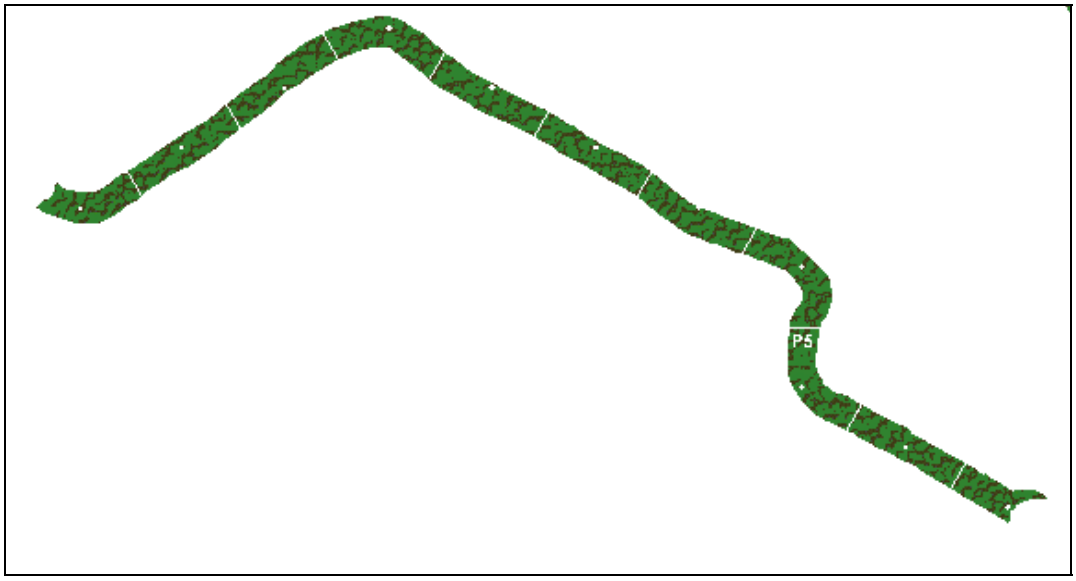
(Woods) Roads To Paths is invoked automatically in the PTO and by SSR in other scenarios. You have to flense out the road net like you did for SRT's and then change the art on the woods-road sections to paths. We typically do this with separate overlays for each woods-road section on the board, but nothing really says that you can't just lay down a full-board-sized overlay to do this. The path itself shouldn't be as wide as the road it's replacing, so you're going to fill in the rest of the overlaid-road with woods pattern or even a bit of open ground.



Overlay 05_RoadsToPaths_i4.gif

3.1.6. NoWoodsRoads

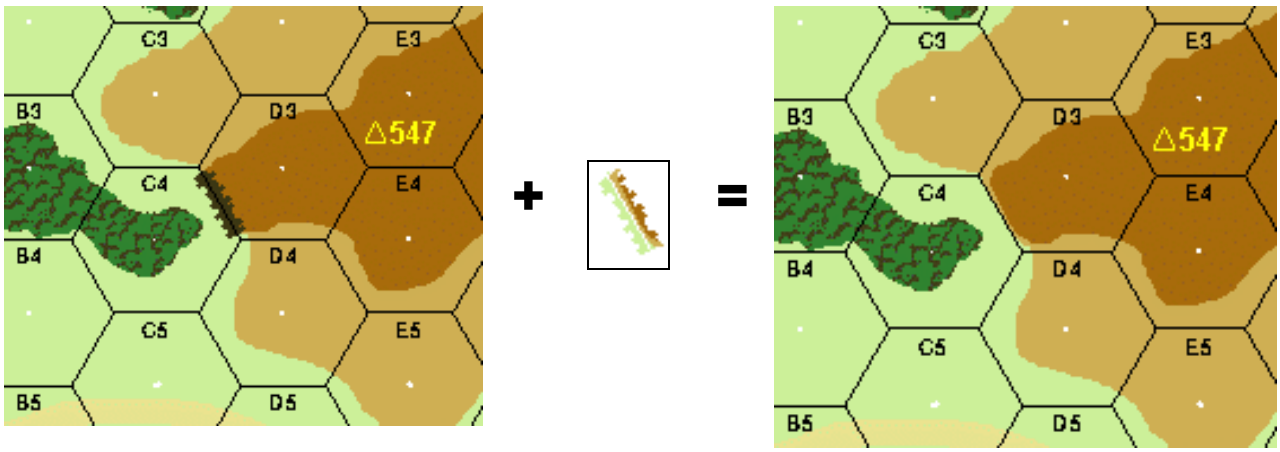
This is automatically invoked under Mud conditions in addition to SSR. The procedure is the same as for RoadsToPaths above, except that the entire road area to be replaced can be filled in with Woods pattern.



Overlay 05_NoRoads_i4.gif

3.1.7. NoCliffs

Removing Cliffs is done with a simple overlay; a color change would have necessitated recoloring all of the VASL v3 boards with level-specific cliff colors, which really didn't seem worth it. The cliff artwork itself is technically at the lower level of the cliff, so lower levels of the transformed cliff have to be accounted for as a multi-elevation change within the width of the cliff artwork. Unlike SRT's that have to specifically keep certain sections of the road net while eliminating others, if NoCliffs is in effect for a board, generally all of the cliffs are gone, so it's OK to just create one big mostly-transparent full-board-sized NoCliffs overlay that covers all of the cliffs on a board.



Board 3 Cliff plus 03_NoCliffs_c4d3.gif equals cliff gone!

3.1.8. Dense Jungle

A long time ago, we looked at various options for how to do dense jungle. Filling the entire hex with the woods pattern is ugly. Coloring the hexgrid in dense jungle hexes a green color is also ugly. What we decided on is the overlay idea. In order to implement Dense Jungles, one needs to create an overlay of the entire board which "densifies" all of the woods hexes. (Inherent Bamboo is part of the "PTO" transformation, separate from Dense Jungle because PTO does not necessarily imply Dense Jungle.) The dense jungle overlay is the size of the entire board and is transparent except for the fringes of the woods hexes which have been made to extend to the entire hex because dense jungle is inherent terrain[8].

When you do a dense jungle hex (or dense jungle overlay), you are allowed to take some artistic license. Don't just fill the entire hex with the woods pattern. Let the woods spill over the hexgrid a little here and there. Put holes and empty spots in the hex to break it up. The idea of "inherent terrain" just means that the terrain depiction should touch each of a hex's six sides at SOME points, but it's surprising how much variation you can put in there while still touching all six sides. Have part of the jungle touch at the vertex (and spill over a bit in a natural way) and you've fulfilled the "inherent" requirement for those two hexsides. You're free to let the jungle line do whatever you want for the remainder of the length of those hexsides. With adjacent jungle hexes, you can have even more leeway, because there's tons of ways to have the jungle cross over the adjoining hexside while still having a pretty natural and interesting tree line.

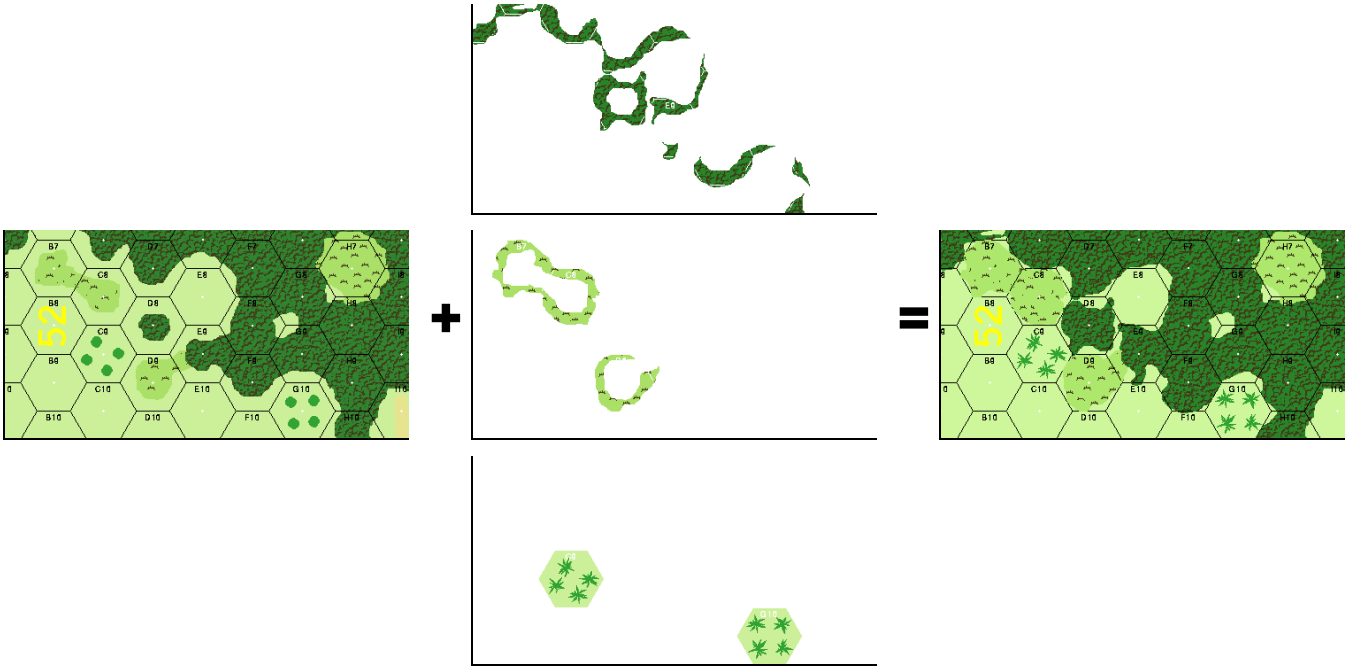
3.1.9. Bamboo

Doing Bamboo is essentially the same as doing Dense Jungle since they're both Inherent Terrain, except you need to watch the level-specific Brush colors. When you encounter a brush hexside that's part of a Woods hex (such as 52D9-E9 in the figure below), there's no real guidance about whether to make that hexside Jungle or Bamboo in Dense PTO. Use your own judgement.

3.1.10. PalmTrees

In the PTO, orchards go to Palm Trees, and Gordon Molek came up with some super-nifty Palm Tree art that's very much worth putting into a Palm Tree overlay. Thing is, the Palm Tree overlay has to completely cover up the green orchard circles while also showing the spiky palm tree art. There are two ways to do this. The first way (shown in the figure below) is to fill the entire hex with the appropriate level-specific open ground color, then put the palm trees in the hex. The second way is to make the orchard art go to the correct level-specific color with a color change entry in the **colorSSR** file, then put down the palm tree overlay that contains only palm tree art with no Open Ground colored background to fill the hex. The problem with the second way is that the orchard trees (the squiggly lines that are left in the hex under Orchards Out of Season conditions - color #105 in the new palette) have only one color for all levels of orchard, so you can only do this when all of the orchards on your board are at the same level.

[8] Alternatively, you can just redraw the entire jungle hex as inherent terrain, like TJ Fachko does in his Geo Workshop in Appendix D.



Board 52…. *plus Dense Jungle (top), Bamboo (middle), and Palm Tree (bottom) overlays…* *equals Dense PTO Board 52*

3.1.11. Positioning Board-Specific Overlays

When VASL puts down these board-specific overlays, it looks in the **overlaySSR** file for the x,y coordinates of the *upper-left-hand (ULH) corner of the overlay*. Therefore, all full-board-sized overlays will be put down at coordinates 0,0, since they cover the entire board (see the placement of the **SRTbd03.gif** overlay in the SRT section above). When you make an overlay out of a certain area of a board, then, you must carefully note the x,y coordinates of the ULH corner when you select that area. Most good paint programs will echo the x,y coordinates of your pointer to the screen. Copy those coordinates immediately to the **overlaySSR** file, which you'll probably want to have open while you're working on creating the overlays.

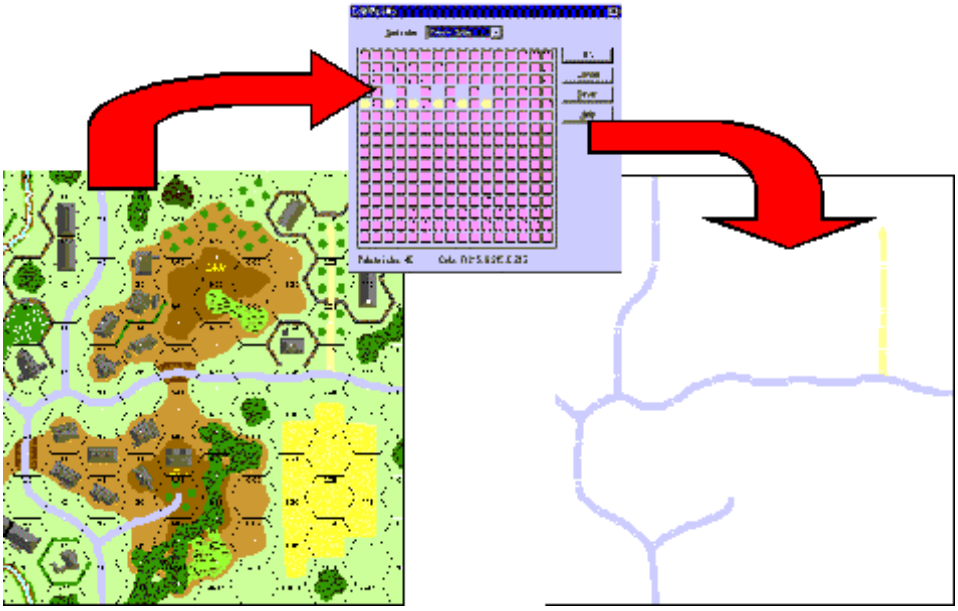
3.1.12. Flensing

And now a word from Mister Vocabulary Man:

> **flense** (flens) vt. flensed, flens'ing [from the Dutch *vlensen* or Danish *flense*]. 1. To cut blubber or skin from a whale, seal, etc. 2. To remove all unwanted colors from a VASL board.

Flensing is really useful, in addition to being cute jargon. Oftentimes while creating VASL overlays, you'll want to remove all parts of a board except certain terrain features like bridges, roads, or cliffs. You could get out your paint bucket tool and tediously repaint the board with transparent color, but being the smart Cabaleer that you are, you'll save time by simply importing another color palette into the board gif so that the only non-transparent colors are the ones you want to keep. You'll want to make sure you maintain the color indices from one palette to the next so that your paint program doesn't try to find alternate colors to use in the absence of the old ones. You really *want* those old colors to be changed to transparent; you don't want your paint program to substitute new colors for you.

Knowing is not enough; we must apply. Willing is not enough; we must do
- Johann von Goethe



Flensing in action: Take a piece of board 41 (left), import a palette that contains only road colors with everything else transparent (middle, for the old geo palette), and voila - a flensed board with only roads remaining (right).

3.2. The VASL Coordinate System

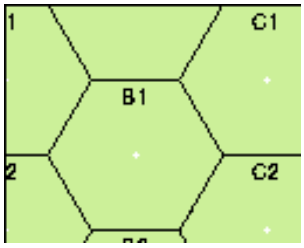
There are times when the user communicates with VASL using the standard ASL hexgrid coordinate system (hexrows A-GG, 0-10 for a geomorphic board), like when he places an overlay while making a new scenario. VASL also communicates with the user in hexgrid coordinates when it auto-reports moves in the text window like "*FF3: 4-5-8 moves FF4". Of course, internally, VASL only responds to mouse clicks and drag-and-drops at certain x,y locations on the board image, so there needs to be a conversion between x,y coordinates on the board and hexgrid coordinates. These conversions are:

$$\text{hex row (A - GG)} = \frac{x - x0}{dx}$$

$$\text{hex number (0 - 10)} = \frac{y - y0}{dy} + \begin{cases} \frac{1}{2} & \text{for even - numbered hexrows like B, D, etc.} \\ 1 & \text{for odd - numbered hexrows like A, C, etc.} \end{cases}$$

Negative numbers for the hexrow are converted to "-A, -B", etc., and hexrow AA comes after hexrow Z, as expected. The values of x0, y0, dx, and dy can be specified in the **boardData** file so as to allow some variety in VASL board graphics and to enable the non-standard hexgrid scheme used by some maps, where the top-left hex on the map is labeled A0, not A1.

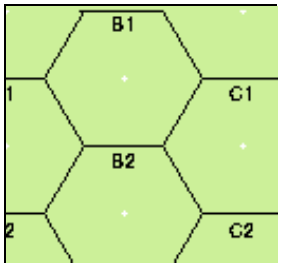
For standard geomorphic maps where the upper left-hand corner is hex A1 like this:



we have:

x0 = 0
y0 = 32
dx = 56
dy = 64

For maps where the upper left-hand corner is hex A0 like this:



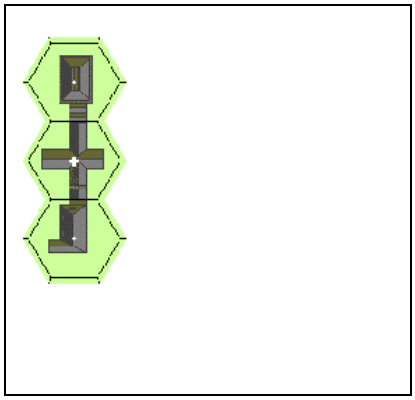
we have:

x0 = 0
y0 = 0
dx = 56
dy = 64

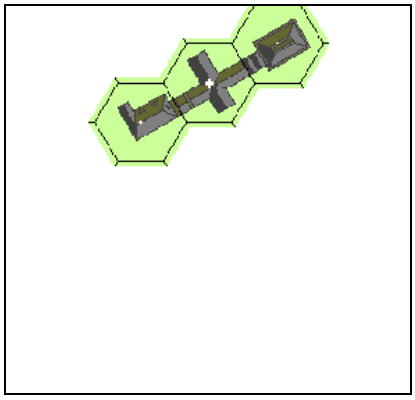
3.3. Making Generic VASL Overlays

Generic overlay files are found in the VASL\Boards\Overlays folder. An overlay zip file contains a group of gifs and a **data** file that tells how VASL how to place them based on the hex coordinate or coordinates you enter at setup. For the standard overlay files, a single zip file may contain many overlay images—for example, the **ovrX** building overlay file contains all the gifs for all the building overlays.

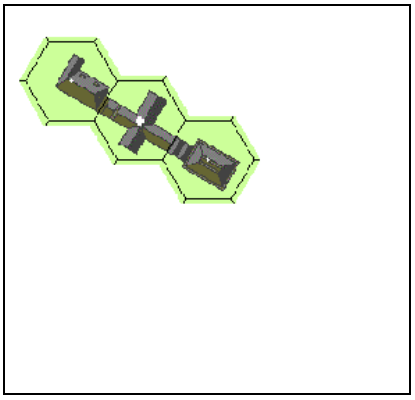
Most overlays have six gifs apiece in the zipfile, one for each possible orientation. In the figure below from left to right are **ovrx24a.gif**, **ovrx24b.gif**, and **ovrx24c.gif**. The a-f suffix of the filename indicates the position of overlay hex 2 relative to hex 1: in **ovrx24a**, hex 2 is "north" of hex 1, in **ovrx24b** it's northeast, and so on clockwise around the six hexagonal directions to **ovrx24f** which has hex 2 northwest of hex 1. These designators are required, and "a" is always north. Some overlays like overlays 1-6 and most of the single-hex overlays use fewer than six gifs, since they aren't intended to be rotated.



ovrx24a.gif



ovrx24b.gif



ovrx24c.gif

3.3.1. Positioning Generic Overlays

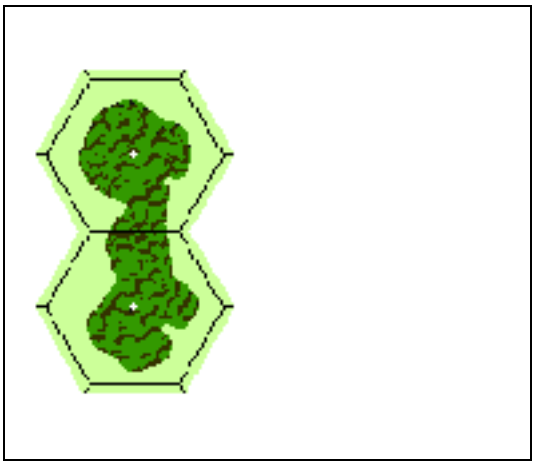
The **data** file in the overlay zip archive is a text file that tells VASL approximately where the "1" hex of the overlay is in relation to the upper left corner of the rectangular image. Here's the **data** file from the woods overlay file **ovrWD**:

```
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 coordinate is where the overlay's hex 1 would fall on the board.

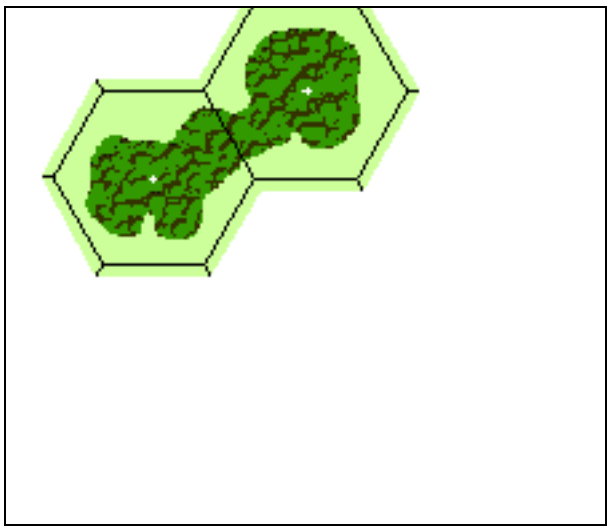
VASL Mapmaking Secrets

In ovrwd2a, the 1 hex is at the south end of the woods. Looking at the **data** file entry 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.



ovrwd2a

Now think about the next rotation of overlay 2, ovrwd2b. 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.



ovrwd2b

Notice that the data for wd1 has six coordinate 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 coordinate entries could be replaced with spaces[9]. For overlays such as ovr1 that have only two valid orientations, you'll see coordinates in the "a" and "d" positions and spaces in the others.

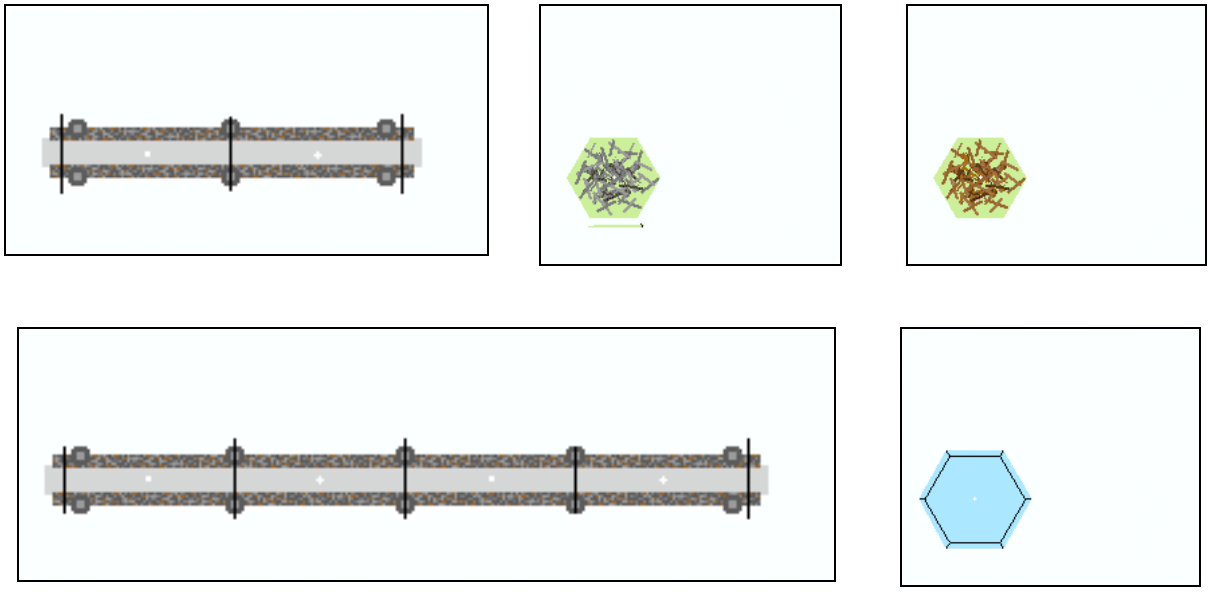
The data coordinate gives only the approximate placement of the gif, within one hex. To fine-tune the placement, you widen the transparent borders at the top and at the left. A good procedure is to start with each gif cropped as small as possible (i.e. minimum transparent area), write the **data** file, build the zipfile, and do a test layout for the "a" orientations of the overlays on a desert board. If the data file is correct, the overlays will be slightly northwest of their intended positions. Save an image of this test layout and open it in your paint app. Then use the rulers to determine how many pixels east and how many pixels south each overlay needs to move. With this information, 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.

[9] Up through VASL 3.01, dummy data was required to get some overlays to be recognized, which is probably what was done here for the "wd1" line in the **ovrWD data** file on the previous page.

3.3.2. Listing of the Generic Overlay Files

The table below lists the generic overlay files available for download from the VASL website. Note that the "Special" overlays do not have any hardcopy equivalent in the system and can be highly worth the effort to print out in hardcopy for use in face-to-face games. The Rubble and Bridge overlays shown below are particularly helpful for eliminating counter clutter while making the real board look nice.

Desert		Ocean		Other				Special	
D	Deir	OC	Ocean	1-6	1-6	DX	Deluxe	SH	Shellholes
E	Escarpment	BE	Beach	B	Brush	RR	Railroad	V	Vineyards
H	Hillock	EF	Effluent	G	Grain	X	Building	SR	Stone Rubble
S	Sand			M	Marsh	ST	Stream	WR	Wood Rubble
SD	Sand Dune			O	Orchard	WD	Woods	BR	Bridge
W	Wadi			OG	Open Ground	HI	Hill	WT	Water
				OW	Orchard-Woods	RP	Rice Paddy	RE	ToT2 Railway Embankment
				P	Pond			HD	Tactiques Hedges



Special Overlays. Clockwise from upper-left: 2-hex Bridge, 1-hex Stone Rubble, 1-hex Wood Rubble, 1-hex Water, and 4-hex Bridge

3.4. Generic Overlay File Name Conventions

Generic overlay gif file names such as **ovrwd2a.gif** should be entirely lower case. If there are capitals in these, VASL will have problems placing the overlay. This is not the case with board-specific overlays, which can have upper- and lower-cases in the name; as long as the board-specific overlay file name is spelled consistently between the **overlaySSR** file, the **SSRControls** file, and the file name itself, everything's OK.

Generic overlay zip file names should be mixed case—**ovrX**, **ovrWD**, **ovrOG**, for example.

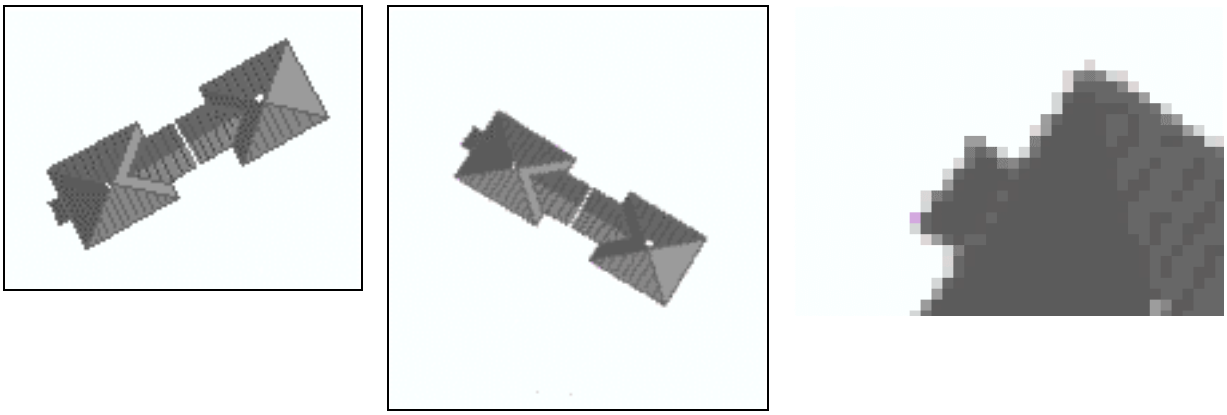
3.5. Creating a Generic Overlay

To make your own generic overlay, first select a name that's different from the standard ones. We'll use the designator ZZ for this exercise. When we build the zipfile, it'll be called **ovrZZ**. Our overlay, a two-hex building, will have six gifs for the different orientations which will be named **ovrzz1a.gif** through **ovrzz1f.gif**.

VASL Mapmaking Secrets

Start by getting a blank hexgrid, saving it as **ovrZZ1a.gif**, 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. When you rotate an image at a nonorthographic angle like 60°, your graphics app will probably create several new colors through antialiasing, which goofs up terrain changes in VASL:



Antialiasing in action: Original building from board 3 (left), with 5 colors. Center: building after rotating by 60 degrees - 28 colors are now used, artifacts of antialiasing. Right: close-up of the building, showing the plethora of colors. Since VASL requires very specific RGB values, the antialiased rotated building will not behave properly when it comes time to make terrain SSR color changes.

After saving this, copy it to **ovrzz1b.gif** and **ovrzz1c.gif**. 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 antialiased areas. (If anyone knows how to prevent Paint Shop Pro from antialiasing rotated images, we'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 coordinates 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.gif-ovrzz1f.gif** 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.

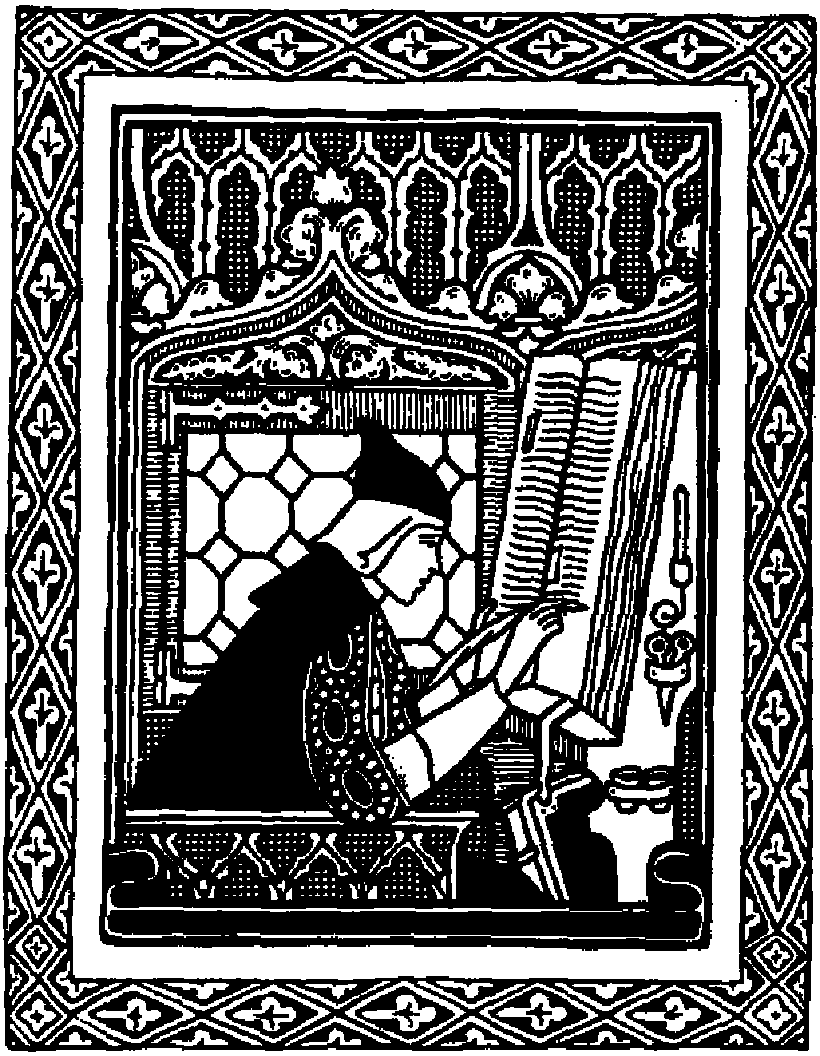
Now it's time to do the initial test image. Start VASL and do a new scenario. Place board 30, then go to the Add Overlays box. Place ZZ1 in each of its six orientations: B2-B1, F2-G2, J2-K3, N2-M3, R2-Q2. Save a board image as **test.ppm**[10], then save the setup as **test.vasl** and exit VASL.

Open **test.ppm** in your graphics application 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 coordinate for this orientation in the **data** file.)

[10] PPM files are for VASL v3, of course; VASL v4 writes gifs, so save this as **test.gif** in VASL v4.

Now open **ovrzz1a**. Set your background brush to the transparent color, then resize your image. 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.



SeparatedMarshBrown…. SeparatedMarshbrown….
SeparatedFuckingMarshBrown…..

To me, a painter, if not the most useful, is the least harmful member of our society
- Man Ray

4. Putting It All Into A VASL bd File

4.1. Anatomy of a bdX File

The files in the VASL boards[11] directory are all uncompressed .zip archives; if you add a .zip file name extension and unzip them, you'd see a folder like the one for board 41 shown at right.

Each **bdX** file contains almost everything VASL needs to display that board and implement various terrain changes such as those imposed by Scenario Special Rule (SSR), weather, or overlays. The board and overlay image files are in Graphical Image File (gif) format; the board files themselves are the largest files (159Kb for board 41 at right) in the folder while the overlays are smaller[12].

There are also several text files which can be in the **bdX** folder. The **data** file should always be in there; it contains the board version number and whatever other information seems useful, like the revision history and artist credits. The **overlaySSR** and **SSRControls** files are needed to tell VASL which overlays to put down under what conditions, and where. There can also be a **colorSSR** file which gives VASL special coloring instructions for that board, but that's generally only for HASL maps. The coloring instructions for all of the geomorphic maps are essentially the same, so that information is kept in the **colors** and **colorSSR** files contained in the **boardData** file, where it is accessible to all of the geo boards in the system.

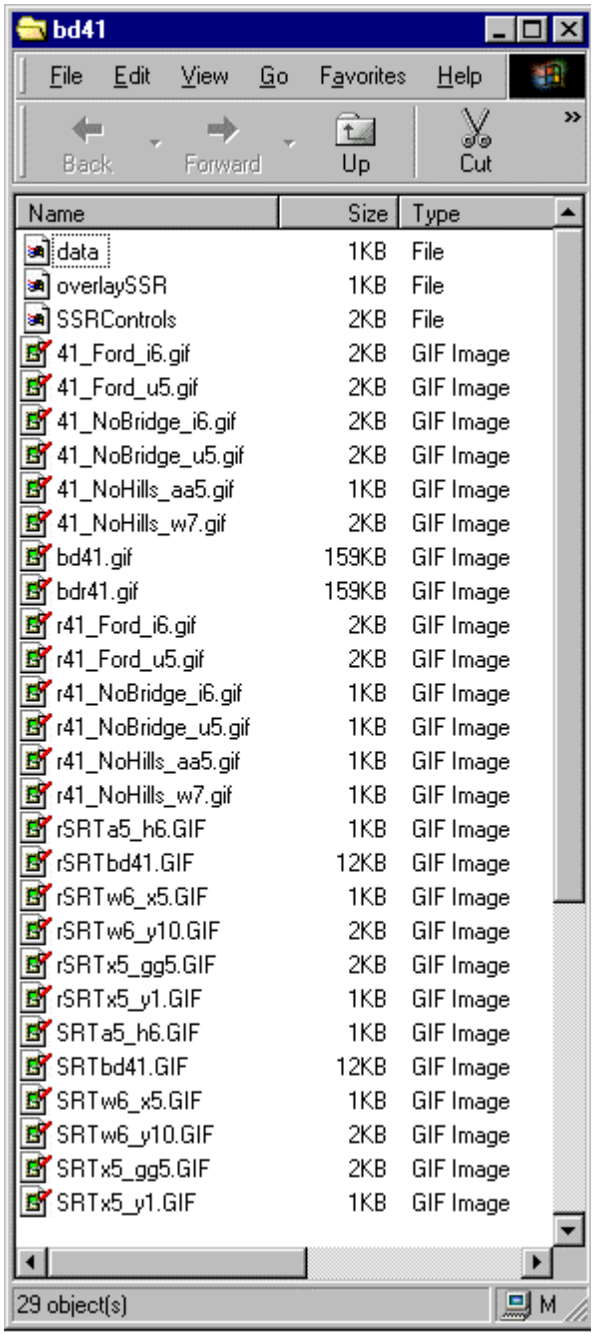
- Case Sensitive!

- The text files in a **bdX** archive don't have to be in MS-DOS format with Control-Linefeeds at the end of every line. Since Java is touted as being cross-platform, VASL should be able to read Unix and Macintosh formats as well.

Reversed versions of boards and overlays!

Making it downloadable from the net by double-zipping it

HASL bdX files - need to be in the HASL list?



The bd41 file, unzipped.
Not all overlay gifs are shown.

4.2. Writing the data file

[11] File names in this document are given in **bold**. Underscores in the file names are really there; Java does not like spaces in file names. The same goes for certain VASL variable names you will encounter later.

[12] Note that board and overlay gifs come in normal and reversed (rotated by 180 degrees) versions; the reversed versions have an "r" in front of the board number in the file name. We'll discuss this later.

The **data** file must be in the **bdX** archive in order for VASL to know that this board uses the standard color palette and can do SSR Terrain Changes for VASL v3 and above. The bare minimum for a **data** file is one line:

```
version 1.0
```

That's it. Other lines can (and should!) be used for other information, like a revision history of the board, including board artist, dates, and changes. Those other lines can be in any format; no special comment characters required - VASL just ignores any entry it doesn't recognize.

4.3. Writing the overlaySSR File

4.4. Writing the SSRControls File

The **SSRControls** file is in a format known as XML, for Extensible Markup Language. XML files are most easily manipulated with an XML-editing program, but they can also be manipulated as plain text in any text editor. A good, simple, free XML editing program is Microsoft's XML Notepad, available through searching Microsoft's website. Despite much effort, we haven't yet found an equivalent for Macintosh, so text editing is the best bet for you guys who Think Different.

4.4.1. XML Syntax

An XML file consists of "elements", each of which has "attributes" and may contain other elements. A typical **SSRControls** file looks like this:

```
<SSRControls>
   <Option ...>
      <Menu ...>
         <entry ... />
         <entry…/>
         <entry …/>        etc. for however many entries you have
      </Menu>
      <ScrollList ...>
         <entry ... />
         <entry ... />
         <entry ... />     etc. for however many entries you have
      </ScrollList>
      <Checkbox ...>
         <entry ... />
         <entry ... />     etc. for however many entries you have
      </Checkbox>
   </Option>
</SSRControls>
```

The `<SSRControls>` represents beginning of an element and the `</SSRControls>` represents the end of that element. If an element does not contain any other elements, like the <entry> elements, you can place a slash before the > instead of requiring a second `</entry>` line. Thus, a **SSRControls** XML file consists of one `SSRControls` element which may contain any number of `Option` elements, each which may contain any number of `Menu`, `ScrollList`, or `Checkbox` elements, each of which may contain `entry` elements. An XML editing program allows you to add and name elements through its interface, or you can enter the ASCII text directly in any text editor.

4.4.2. Building an SSRControls File

The **SSRControls** file tells VASL to place certain GUI controls into the Terrain SSR window and to use those controls to set terrain rules. The three types of countrols are: Checkbox, Menu, and ScrollList. Regardless of the type of GUI control used, the behavior is determined by an `entry` line in the XML file. For example:

```
<entry name="No Valley" rule="Level_1ToLevel0," text="no valley"/>
```

The `name` attribute of the entry element is displayed as text within the GUI controls. The `rule` attribute names the rule which will be invoked by selecting this entry. The ruled named in the `rule` attribute must appear in either the **colorSSR** or **overlaySSR** file. The `rule` attribute can contain multiple rules separated by a comma; this is often necessary when multiple terrain changes need to be made to put a certain rule into effect, as we saw above in the SSRControls file for board 5 on page 31. Note that the rules themselves cannot have spaces in them, or VASL will not be able to parse them and they will not show up in the TerrainSSR dialog box. The `text` attribute is used to construct a text message informing the user of the terrain transformation that VASL has performed on a board.

The same format for an `entry` line can be used in any of the available GUI controls. For example, the three lines

```
<Checkbox name="No Valley">
<entry name="No Valley" rule="Level_1ToLevel0," text="no valley"/>
</Checkbox>
```

result in a checkbox appearing in the terrain SSR window. The checkbox would be labeled "No Valley" and checking the checkbox would apply the "Level_1ToLevel0" rule and give the user a notification "All boards: no valley" once VASL had completed the transformation.

Menus and ScrollLists can have more than one entry. For example, the lines

```
<Menu name="Grain">
  <entry name="In Season"/>
  <entry name="Out of Season" rule="NoGrain" text="grain out of season"/>
  <entry name="Plowed Fields" rule="PlowedFields" text="plowed fields"/>
</Menu>
```

result in a pull-down menu appearing in the terrain SSR window. The menu would be labeled "Grain" and would contain three entries. The first, "In Season" has no effect if selected. The second, "Out of Season" results in the rule "NoGrain" being applied and the message "All boards: grain out of season". The third, "Plowed Fields", results in the rule "PlowedFields" being applied and the message "All boards: plowed fields".

The only difference between a ScrollList and a Menu is that the user is able to select more than one entry from a ScrollList. For example, the lines

```
<ScrollList name="Keep the following road sections only:" >
 <entry name="50A5 TO I10" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTa5_i10" text =" 50A5-I10 "/>
 <entry name="50I10 TO I11" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTi10_i11" text=" 50I10-I11 "/>
</ScrollList>
```

give the user the option of selecting both road sections A5-I10 and I10-I11. In this case, we are also invoking multiple rules per entry (EraseRoads, NoWoodsRoads, and either SRTa5_i10 or SRTi10_i11). It does not matter that the same rule may be invoked by both entries. Each rule will only be applied once, in the order specified by the **overlaySSR** file (back to front, of course). By the way, the text in the paragraph above is in a smaller font because the text in the **SSRControls** file should not spill over from one line to the next. All of the stuff between the <> brackets really should be on one line. Other than that, though, linebreaks won't matter.

Multiple GUI controls can be grouped together by placing them within the same `Option` element. For example, in the VASL v4 default controls (the **SSRControls** file inside the **VASLdata** file), the "Hills" option contains a Menu named "Hill Levels", a Checkbox named "No Valley" and a Checkbox named "No Cliffs".

5. Deep Background on How VASL Works

5.1. How VASL Creates a Board Configuration

When the user creates a board layout for a scenario, VASL loads the required board and overlay gifs (the former as directed by the Scenario Maker dialog box, the latter as directed by the **overlaySSR** file in that board's **bdX** archive) and then does terrain changes by performing the following tasks in this order:

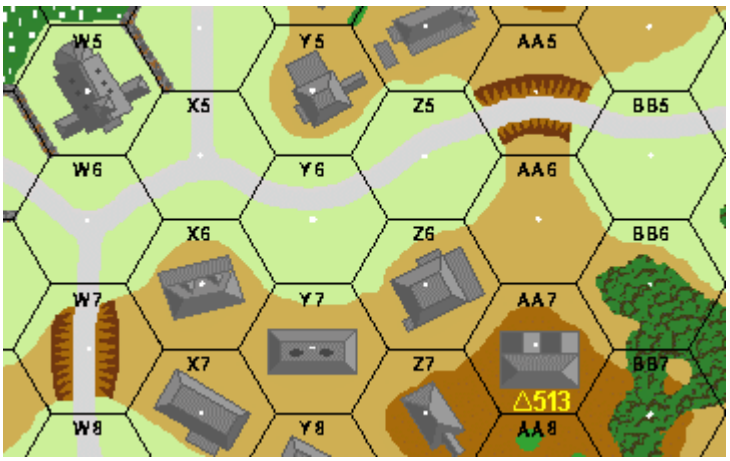
a) Apply color changes to the board-specific overlays.
b) Apply color changes to the generic overlays[13].
c) Apply color changes to the boards[14].
d) Place the board-specific overlays on the board.
e) Place the generic overlays on the board.

Understanding this sequence will help you understand why the boards and overlays are made the way they are. With this sequence, certain board changes become difficult or impossible to do with color changes, so the solution is to create an overlay. And vice-versa; certain things are easier to do with color changes than with overlays.

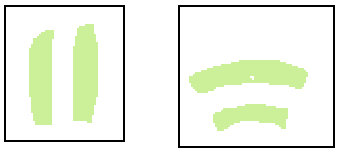
Nobody knows how or why Rodney did things this way. It's rumored that placing the overlays on the board can alter their pixels' RGB values slightly, messing up the color change system, but it's all lost in the mists of time.

5.2. Overlay or Color Change? Two Case Studies

As an example, consider the problem of trying to implement the "No Hills" rule on board 41, where there are Sunken Roads cutting through hills in hexes W7 and AA5. In general, the "No Hills" rule is implemented with a simple color change - every level-specific terrain color (Grain, Brush, Roads, Orchards, and Open Ground) is converted to its Level 0 equivalent. However, this would leave some ugly Sunken Road terrain art straddling the roads in W7 and AA5, so two overlays were created to cover the Sunken Road art with Open Ground Level 0 color:



Trouble at the Sunken Roads on board 41



W7 ***AA5***

bd41 NoHills Overlays

[13] Those that are stored in the VASL/boards/overlays folder and are generally the same overlays that are printed in hardcopy for use with the real hardcopy maps, like the Wd1-5 Woods overlays, etc.

[14] When applying color changes, board-specific rules (listed in the **colorSSR** file that may be in each board's **bdX** archive) are applied first, then the generic rules that are listed in the **colorSSR** file in the **boardData** archive. In general, the only boards with their own **colorSSR** file are HASL maps; geo boards should not need their own **colorSSR** files.

The Artisan is he who merely applies the rules which others have detected
- Henry David Thoreau

When "NoHills" is invoked for board 41, the **overlaySSR** file for bd41 directs VASL to load two overlays that are kept in the **bd41** archive: **41_NoHills_w7.gif** and **41_NoHills_aa5.gif**. Steps a-e are then performed, with these effects:

a) No effect. The colors on these overlays are already correct for Level 0.
b) If any ordinary overlays have been called for and read in by VASL, their level-specific colors will be changed to Level 0.
c) Level-specific colors on board 41 are changed to Level 0.
d) The **41_NoHills_w7.gif** and **41_NoHills_aa5.gif** overlays are placed on board 41 in the correct positions[15].
e) Any ordinary overlays are placed on board 41 in the correct positions[16].

As another example, consider what has to happen to implement Stupid Road Tricks (SRT's). SRT's allow us to implement SSR's that say "Only the road from A to B exists". One way to do this would be to wipe out all of the roads on a board with the "NoRoads" color change, then put down a "RoadAToB" overlay on the board, restoring the road that the user wants. Unfortunately, though, this won't work because of the a-e sequence above. When we implement the "NoRoads" color change, the road on the "RoadAToB" overlay will also change to open ground in step a, and it won't do any good to put it down on the board in step d. One could get around this by slightly changing the color of the road on the "RoadAToB" overlay to something that looks indistinguishable from the normal road color but is slightly different (and thus would not respond to the "NoRoads" color change), but then that road would not respond to *other* color changes that one might want it to, like changing to a lighter color in Winter.

Therefore, the way SRT's were implemented in VASL v4 was to wipe out all of the roads on a board not with a color change, but with a "NoRoads" overlay that replaces a road with its open ground equivalent. Then the "RoadAToB" overlay that the user wants is put down on top of *that*, restoring the desired road. Of course, since the roads on these overlays are the same color as the roads on the normal boards, they undergo the correct color changes due to weather like Winter or Mud (step a) before they are put on the board (step d).

5.3. Special Overlay Precedence

This touches on another issue, that of special overlay precedence. Some terrain changes require more than one board-specific overlay, and you need to know which one gets put down first and which gets put down last. The **SSRControls** file will tell you which overlays get put down for a certain terrain change, but they get put down in the *reverse order* in which they appear in the **overlaySSR** file. For instance, this is the SRT section of the **SSRControls** file for the VASL v4 version of board 5:

```
<Option name="Stupid Road Tricks">
<ScrollList name="Keep the following road sections only:" >
<entry name="5I10 TO I9" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTi10_i9" text=" 5I10-I9 "/>
<entry name="5Q10 TO Y10" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTq10_y10" text=" 5Q10-Y10 "/>
<entry name="5Q1 TO R1" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTq1_r1" text=" 5Q1-R1 "/>
<entry name="5A5 TO H3" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTa5_h3" text=" 5A5-H3 "/>
<entry name="5I1 TO H3" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTi1_h3" text=" 5I1-H3 "/>
<entry name="5H3 TO Y10" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTh3_y10" text=" 5H3-Y10 "/>
<entry name="5Y10 TO Y11" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTy10_y11" text=" 5Y10-Y11 "/>
<entry name="5Y10 TO EE6" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTy10_ee6" text=" 5Y10-EE6 "/>
<entry name="5EE6 TO GG5" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTee6_gg5" text=" 5EE6-GG5 "/>
<entry name="5Y1 TO EE6" rule="EraseRoads,NoWoodsRoads,NoBridge,SRTy1_ee6" text=" 5Y1-EE6 "/>
</ScrollList>
</Option>
```

The SRT section of the SSRControls file for board 5 (VASL v4 version)

[15] The x,y position of the upper left-hand corner of the overlay is given by the two numbers next to the name of the overlay in the **overlaySSR** file, where 0,0 is the upper left-hand corner of the board gif itself. See page 21 for more details on the VASL coordinate system.
[16] As indicated by the hex ID's given by the user in the "Add Overlays" dialog box.

Getting past the Extensible Markup Language (XML) formatting, you see on lines 3-12 that each SRT entry implements the EraseRoads, NoWoodsRoads, NoBridge, and SRT*hexa_hexb* rules. When more than one SRT entry is selected, VASL collects all of the referenced rules and parses the list to remove duplicates. For example, if the user wants to keep only the road running from 5A5 to 5I1, he will select the "5A5 TO H3" and "5I1 TO H3" rules, and VASL will parse the rule list to implement the "EraseRoads", "NoWoodsRoads", "NoBridge", "SRTi1_h3", and "SRTa5_h3" rules.

However, the order in which these rules are implemented (i.e., the order in which the corresponding overlays are placed on the board) is determined by the *reverse order* in which the rules appear in the **overlaySSR** file[17]. Looking at board 5's **overlaySSR** file for the rules that are relevant to keeping that 5A5-5I1 road section, we find:

```
SRTa5_h3
SRTa5_h3.gif  0,76

SRTi1_h3
SRTi1_h3.gif  360,0

NoWoodsRoads
05_NoRoads_i4.gif 409,116

NoBridge
05_NoBridge_y8.gif 1310,450
05_NoBridge_z9.gif 1370,547
05_NoBridge_ee2.gif 1645,65

EraseRoads
SRTbd05.gif  0,0
```

Part of the overlaySSR file for board 5 (VASL v4)

Thus, in keeping the 5A5-5I1 road, the overlays are put down in the following order[18]:

1)	**SRTbd05.gif**	*Covers up all roads on the board with their open ground color.*
2a)	**05_NoBridge_y8.gif**	*These three cover up the bridges in y8, z9, and ee2 with "NoBridge"*
2b)	**05_NoBridge_z9.gif**	*overlays which replace the bridge artwork with Gully artwork.*
2c)	**05_NoBridge_ee2.gif**	
3)	**05_NoRoads_i4.gif**	*Covers up the long woods-road with a woods pattern.*
4)	**SRTi1_h3.gif**	*Puts the H3-I1 road back on the board.*
5)	**SRTa5_h3.gif**	*Puts the A5-H3 road back on the board*[19].

As you can see from the description in the right-hand column above, this process does indeed result in (only) the correct road sections being displayed for the user. Ugly, you say? Convoluted? Weh-heh-heh-hell young whippersnapper, you're absolutely right. Still, it works, and having it work under all circumstances is pretty much the bottom line here.

[17] Why? Because RK did it that way. He's a sick, sick puppy.

[18] The position of the overlay on the board is determined by the x,y coordinates given immediately after the overlay file's name in the **overlaySSR** file. See Section 3.1.11 for more details on positioning board-specific overlays.

[19] SRT road overlays have the road go to the middle of a road intersection and stop, so as to connect to the next piece of the road, wherever it may lead. See Section 3.1.4 for details.

VASL Mapmaking Secrets

5.4. Uncle Rodney's Bigass List of VASL Terrain Changes

When the VASL Map Cabal first glommed on to the potential to make terrain changes, they went through their scenario books to find the most popular terrain changes called for by weather or SSR and worked on those first. As the years have passed, more and more terrain changes have been implemented as the Map Cabal members have grown increasingly desperate for things to do. The following is a list of the terrain changes which are possible in VASL, with brief explanations of how they're done. Note that some terrain changes call others, like when "Mud" implements the "NoWoodsRoads" rule in addition to its own color changes.

All of the color changes are listed in the **colorSSR** file in the **boardData** archive; all of the overlay changes are listed in the **overlaySSR** file in each board's own **bdX** archive. This is seemingly an efficient way to organize things; since color changes apply equally to every board in the system, they can be kept in the **boardData** file which is used by every board, while overlays implementing terrain changes are different for each individual board, so they should be kept in the board's own **bdX** file.

Frighteningly, this guy looks a <u>lot</u> like Rodney Kinney.

5.4.1. VASL v3 Terrain Changes

Color Changes

- AllPaved & AllDirt - Change Paved Roads on all levels to their Dirt Road equivalents, and vice-versa.
- NoDirt - Converts Dirt Roads on all levels to their level-specific Open Ground colors.
- NoStairwells - Changes the stairwells in wood and stone buildings to medium-darkness wood and stone building colors.
- RowhouseBarsToBuildings - Changes the rowhouse bars in wood and stone buildings to medium-darkness wood and stone building colors.
- AllStone - Changes the seven Wood building colors to their corresponding seven Stone building colors. Also changes the wood stairwell and rowhouse bar colors to their stone equivalents.
- AllWood - Same as AllStone, but changing Stone colors to Wood.
- XToY - Called by the "Transformations" dialog panel, "X" and "Y" can be "Woods", "Grain", "Brush", or any Open Ground color from level from -1 to 4. When converting to Brush or Woods, only the green of those colors is shown, not the darker indicator color (this is fixed in VASL v4). When converting from "Grain", the "Furrows" colors are also changed to "Y".
- BrushToOpenGround - Brush and BrushBlack at all levels are converted to their level-specific Open Ground colors.
- SwampToMarsh - Swamp Green and Brown colors are converted to their Marsh equivalents. Used when???
- MarshToOpenGround - Marsh and Separated Marsh Green and Brown colors are converted to their level-specific Open Ground colors.
- GulliesToStreams - Gully Green and Brown colors are converted to their Stream Green and Brown equivalents. Gully Black is converted to StreamWater.
- Flooded - Marsh Green and Brown become Water. Water White and Stream White become Water. StreamBrown and StreamGreen3 (the two colors closest to the water interior of the stream) become Water.
- DryStreams - MarshGreens become Mudflat (a nice muddy brown) while Stream colors either go to Level 0 Open Ground or become DryStream1 or DryStream2, two brownish colors.
- OrchardOutofSeason - Orchard colors revert to their level-specific Open Ground colors. OrchardTrees become black.
- NoGrain - Grain and Furrows colors revert to their level-specific Open Ground colors.

- NoSunkElevRoads - SunkRoad colors go to Level 0 Open ground.
- NoRoads - Paved and Dirt Road colors (not Paved/Dirt Road Bridge colors, though) convert to their level-specific Open Ground equivalents.
- NoHillNums - HillNum color goes to Level 0.
- HedgesOnly - Wall Browns and Greys convert to one of the Hedge Green colors.
- WallsOnly - Hedge Greens convert to one of the Wall Greys.
- WallsToBocage - Wall Browns and Greys convert to HedgeGreen (for pixels next to the hexside) or PathBrown (for pixels on the outside of the HedgeGreen pixels).
- HedgesToBocage - HedgeGreen goes to PathBrown on the outside of the hedge.
- PlowedFields - Grain colors at all levels go to PlowedFields brown, level-specific Furrows colors go to the dark brown Furrows color.
- RiverToValley - Water and WaterWhite go to Level -1 Open Ground
- IrrigatedPaddies - Interior of Paddies goes to Water
- InSeasonPaddies - Interior of Paddies goes to Grain
- Winter - a whole host of color changes. NoGrain and OrchardsOutOfSeason are also called.
- Extreme Winter - same as Winter, but also calling BrushToOpenGround.
- Mud - another whole host of color changes. NoDirtRoads is also called.

Overlay Changes

- NoBridge - Covers up a bridge with artwork depicting the gully or stream underneath it.
- BridgeToFord - Essentially the same as NoBridge, but also adds the the artwork indicating a Ford across the gully or stream.
- NoWoodsRoads - Covers up a woods-road with the woods pattern.
- RoadsToPaths - Covers up a woods-road with path artwork.
- Level1ToLevel0 - This covers up the Sunken Roads that cut through the level 1 hills, like you get on boards 39 and 41.

5.4.2. VASL v4 Additions

- Stupid Road Tricks (SRT's) - allow only certain sections of the road net to be displayed.
- PTO Terrain - Removes roads, turns woods-roads into paths, puts down Bamboo overlays to make Bamboo into inherent terrain, and puts down Palm Tree overlays on top of Orchards.
- PTO Dense Jungle - Same as PTO but also puts down Dense Jungle overlays to make Jungle into inherent terrain.
- "Transformations" like Woods to Brush now include the correct indicator, not just the background color, using underlays.
- NoCliffs
- Steppe Terrain

Art washes away from the soul the dust of everyday life
- Pablo Picasso

5.5. Making Your Own Terrain Changes

Once you understand how VASL uses the **colors**, **colorSSR**, and **overlaySSR** files to implement terrain changes, you too can play that game. VASL v4 is more amenable to this kind of thing than v3, but it's possible in both versions.

Step One is deciding what change you want to make and what's the best way to implement it - with a color change or with an overlay. As an example, let's say you're working on a HASL project using the geomorphic boards and your knowledge of the historical terrain indicates that the hills in the area were rocky and barren, while the lower-lying areas were more fertile. You may want to fix up your boards so that there are no Grainfields or Brush above Level 1. While this could be done with overlays, it's more easily done with color changes since the drawing palette provides different level-specific colors for Grain and Brush. You're going to want to change all Grain, Furrows, Brush, and Brush Indicator[20] colors at Levels 2 and above to their level-specific Open Ground colors.

Now let's try a different one. Let's assume you want to implement some kind of Dense Woods rule where the outer edge of large Woods masses are normal Woods hexes but the interior Woods hexes are "Dense Woods" that maybe have higher MF costs, lower stacking capacities, or whatever. You want your terrain change to make the Dense Woods look darker than the regular Woods hexes. Clearly this cannot be done by simple color changes, since VASL cannot look at a board gif and know from the pixels' RGB values which hexes are regular Woods and which are Dense Woods - the drawing palette in Appendix C lists all Woods hexes as having the same two colors (dark brown on green), with no distinction made between regular Woods and Dense Woods hexes, or even between Woods hexes on different Levels. Therefore, this is a case where you must make an overlay and tell VASL how to implement it.

5.6. Master's Degree Thesis Projects

In order to demonstrate your mastery of The VASL Way, here is a list of Officially Sanctioned Hard Things that the senior members of the VASL Map Cabal either didn't have the time or the energy to do themselves. Think of them as Quests for the journeyman board artist. Completion of any one of these projects will earn you the status of Master Apprentice To The Cabal, Second Class (Probationary).

- Implement a good color change scheme for Night rules. Starshells, Illuminating Rounds, etc.[21]
- Implement a good color change scheme for printing hardcopy VASL maps that look closer to the colors that the real ASL boards have.
- Do the VASL map for some ASL (or "ASL-compatible") product that nobody's ever done before.
- Do up an overlay to cover up the runway on board 14 (RK's personal irritation) or the big building on board 9.
- Do up a blank hexgrid that has hex ID numbers that look nicer than what we're using now.
- Go to the TR maps <http://www.mindspring.com/~tqr> and create all of the necessary board-specific overlays for the maps that don't already have them.
- Bring us… a shrubbery! [22]

[20] An "indicator" is the dark colored shape that is part of the Woods, Brush, and Marsh patterns. The actual colors are named WoodsBlack, BrushBlack, and MarshBrown. VASL v3's Transformations (Woods to Brush, etc.) could only lay down the background color of the transformed-to terrain type, without the proper indicator. The "underlay" concept in v4 fixes this.

[21] Not as easy as it sounds. Ostensibly you want to make the board somewhat dark and then lighten it up in the illuminated regions. Thing is, if the board is too dark, it gets hard to play on - people can't tell whether buildings are wood or stone, or what level a hill is. For a good simulation of this, open up a VASL board and then turn down the brightness on your monitor. Yowch.

[22] A nice one. And not too tall. With a little path running down the center. Ni!

6. Printing VASL Maps

When you select "Save Map Image" from the VASL menu, VASL saves the image for later printing. VASL v3 saves in Portable PixMap (PPM) format, while VASL v4 saves in gif format. PPM files are huge, as each line of the file is one RGB triplet of one pixel; a single VASL board can run 3 Mb in size. Fortunately, these files shrink down nicely when converted to gif or any other format. Several graphics programs can read and convert PPM files, including PaintShop Pro and xnView for Windows and GraphicConverter for the Macintosh.

6.1. Converting to Full-Sized ASL Hexes

It is indeed way cool to print a VASL map as a standard-sized ASL map. After you have converted the VASL map to the more workable gif format, you can import that file into a page layout program or just send all or some of it directly to a printer. The colors will probably not look as good as the colors you're used to seeing on a standard ASL board because the VASL colors were never intended to look good in print, only on a computer screen. Also, there are always variations in color tones produced by different printers. It remains a Cabal Master's Degree Thesis project for some fair lad to figure out what good RGB values are for the printed maps and set up those color changes as a user-selectable option in VASL.

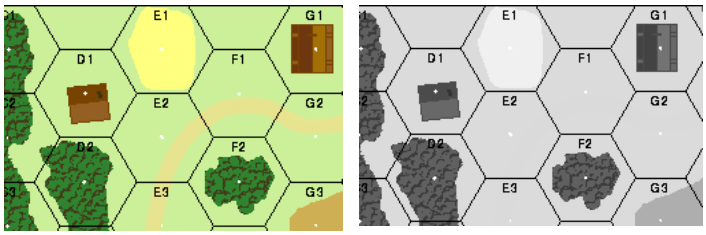
When outputting a VASL map to full-hex size, the desired percentage is 86.95% because VASL hexes are 23mm across, while real board hexes are 20mm across, a ratio of 115%, or 1/86.95%.

6.2. PDF Conversion[23]

There are several methods available for people who wish to print out VASL maps as full-sized ASL maps in Postscript and Portable Document Format (PDF) files. The way that works best is to import the original VASL map into a page layout program, then output at the desired size (86.95%) in postscript form, then convert the postscript document to PDF. When converting the postscript document to PDF ("distilling"), make sure that all downsampling of images is set to 300 dots per inch (dpi) for best results.

6.3. Greyscale Values

When printing VASL maps on a greyscale printer, you may notice that some terrrain features do not show up well. In particular, grain and roads seem to wash out. This is because those terrain types' RGB values translate into greyscale values which are very close to that of Level 0 Open Ground.



A section of board 2 in color and greyscale. Where'd the road from E3 to G2 go?

There are many ways to translate RGB color values to greyscale, but the one which your greyscale printer probably uses is this:

Grey value = 0.299R + 0.587G + 0.114B

With this equation, you can see how a Level 0 dirt road would be nearly indistinguishable from Level 0 Open Ground when printed on a greyscale printer. The dirt road's RGB triplet of 231,227,143 translates to a greyscale value of 219, while the Open Ground's RGB triplet of 204,240,153 translates to the exact same value of 219. Level 0 grain (RGB of 255,255,127) is slightly easier to distinguish from Level 0 Open Ground, since its greyscale value is 240.

This is unfortunate, but was not foreseen by the VASL Map Cabal when the VASL v3 colors were being standardized. Given that nobody wants to go back and repaint all of the VASL maps just to make them print better on greyscale printers, the solution is to come up with a set of color changes which could be invoked by the user in the TerrainSSR dialog box to yield colors that still look nice but are more readily distinguishable when printed in greyscale.

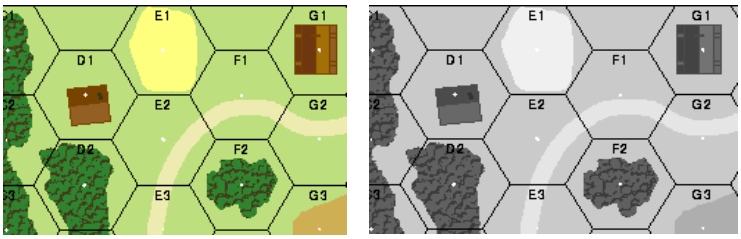
[23] This section steals liberally from notes supplied by Steve Svare.

Our delight in any particular study, art, or science rises and improves in proportion to the application which we bestow upon it. Thus, what was at first an exercise becomes at length an entertainment - Joseph Addison

VASL Mapmaking Secrets

For starters, though, the basic problem is separating roads from Level 0 Open Ground. To do this, one might change the former color to an RGB triplet of 238,235,176 (greyscale 229) and the latter to 189,223,125 (greyscale 200). Here's what that section of board 2 looks like in greyscale under those color changes: NIIICE.

{One of the senior Cabaleers already has this on his ToDo list}



The same section of board 2 with greyscale-tweaked colors.

7. How to Make VASL Counters

Those who love to make VASL counters are a funky sub-genus of the standard VASL Map Cabaleer. Eager to give equal time to the Counter Cabaleers, this section is proudly offered[24].

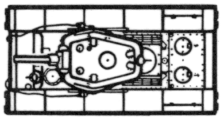
7.1. Making Counters in VASL v3

7.1.1. What you need:

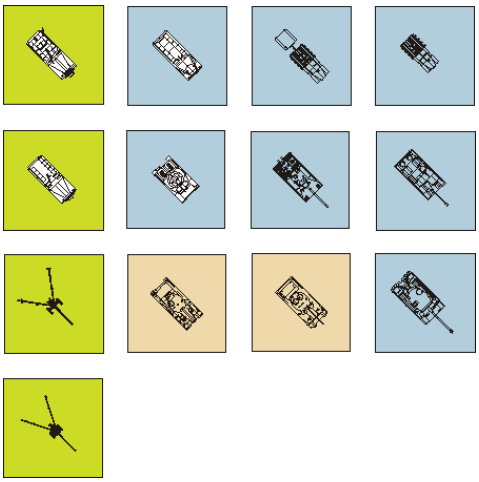
-Corel Draw (or any other vector based program) to draw/trace the Silhouettes
-PhotoShop or PaintShopPro to convert to BMP and gif
-a zipping program
-lots of time and patience

7.1.2. Step-by-Step:

1. Create your counter image and save it as a bitmap graphic (i.e. TIFF or BMP)

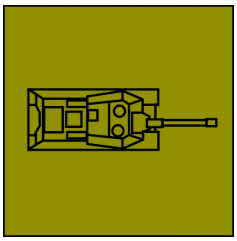


2. Import it into your Vector Program (i.e. Corel Draw) and trace the image



3. Place the image on a square with the appropriate nationality colour (see the Nationality Colour Chart)

4. Export the image as BMP (here you will need some fine tuning - I will provide more info later)

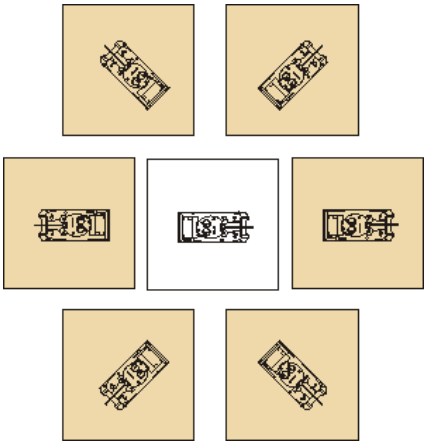


[24] This section is lifted almost verbatim from "VASL Counters – a first how-to ver. 0.1" by Klaus Fischer. Thus the funky "u" in "colour". Don't be alarmed, American Cabaleers.

A successful work of art is not one which resolves contradictions in a spurious harmony, but one which expresses the idea of harmony negatively by embodying the contradictions, pure and uncompromised, in its innermost structure - Theodor W Adorno

VASL Mapmaking Secrets

You need to export one image per counter facing!



5. Open the image in Photoshop or Paintshop Pro and resize it to the appropriate size (i.e. 60x60 pixels for 5/8" counters like AFV and Ordnance).

6. Now the tricky part: make everything except for the Silhouette itself transparent

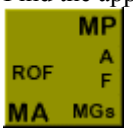
7. Save/export as gif

8. Watch the naming convention!
   Currently VASL has the following classes (per nation)
   Example is American (am)
   - amart1-6 Artillery
   - amdozer1-6 Dozer, unarmored
   - amhvAA1-6 heavy AA
   - amhvAT1-6 heavy AT
   - aminf1-6 INF Gun
   - amjeep1-6 Jeep
   - amlt0-6 (0 is the wreck picture) light Tank
   - amltAA1-6 light AA
   - amltAT1-6 light AT
   - amlvt0-6 LVT
   - ammt0-6 medium Tank
   - ammtr1-6 Mortar
   - amotac0-6 open-topped AC
   - amotag0-6 open-topped AG
   - amotht0-6 heavy Tank
   - amotlvt0-6 open-topped LVT
   - amotsaa0-6 open-topped SPAA
   - amotspa0-6 open-topped SPA
   - omottd0-6 open-topped TD
   - amtruck0-6 Truck

   this list is NOT exhaustive as other nations have RCL, SPAA, TD ect.

9. OK, now you've got the AFV/Gun Silhouette, let's do the Counter Information

10. Find the appropriate PSD template and open it up



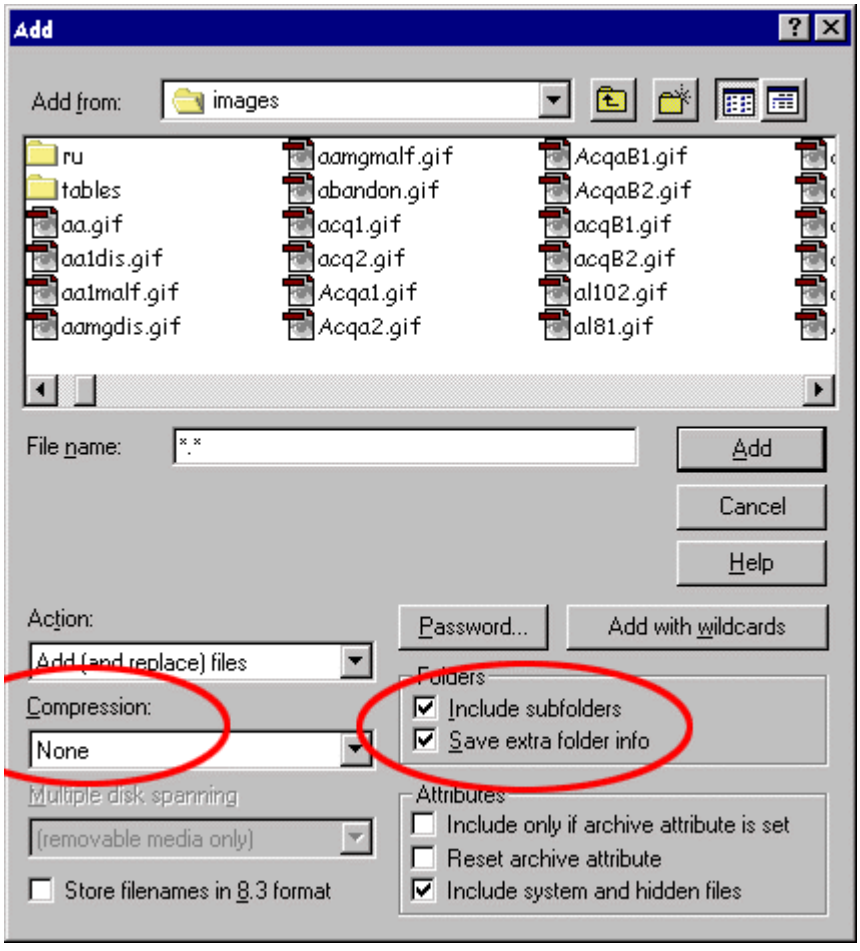
11. Check the “Layer” menu and show/hide the necessary layers.

12. Fill with the info required

13. Save/export as gif

7.2. The VASLdata File

Once you have the images needed place them in the appropriate directory (see Directory Structure) and create a ZIP file of all the images. IMPORTANT!! Make sure your ZIP program has the following options checked:

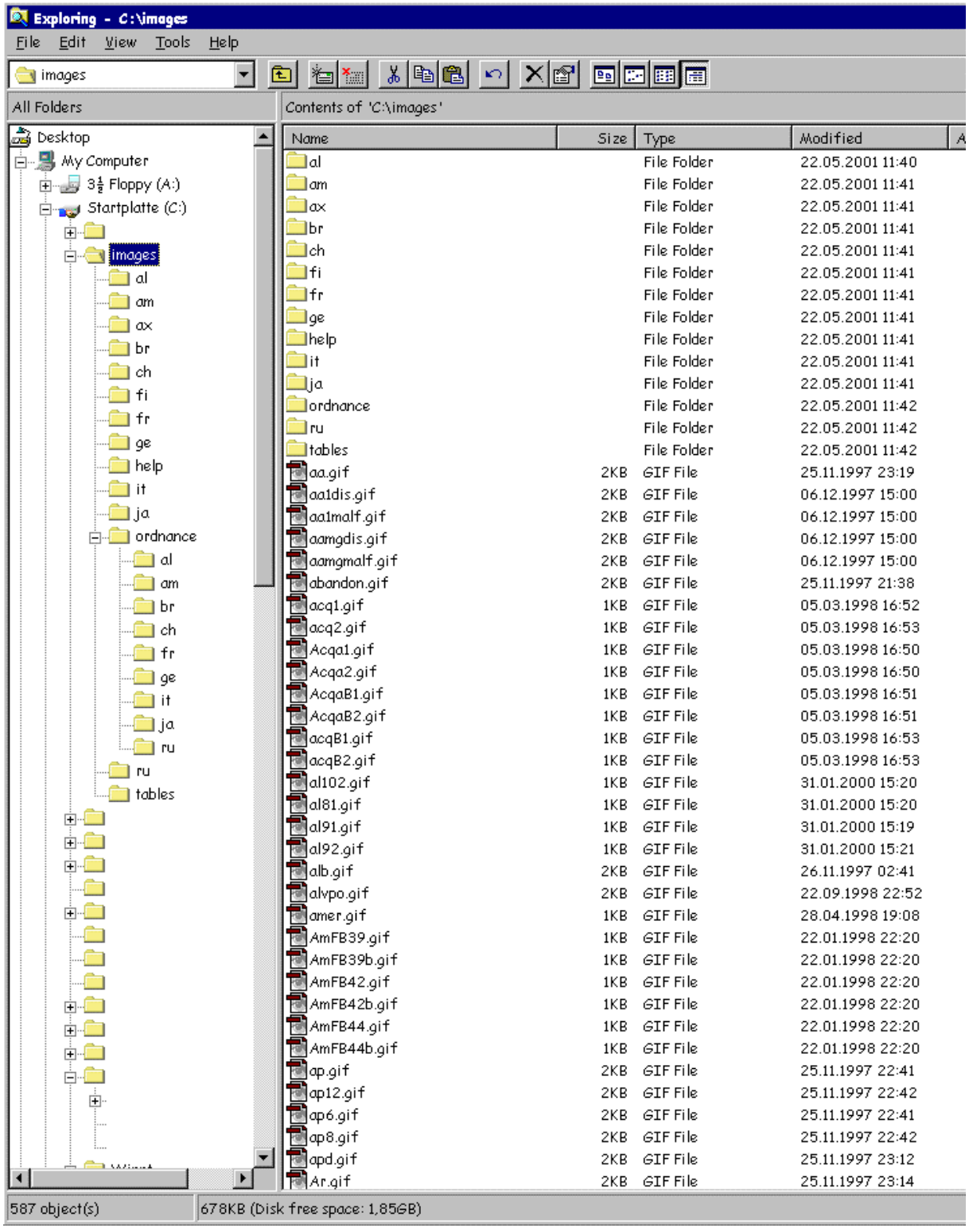


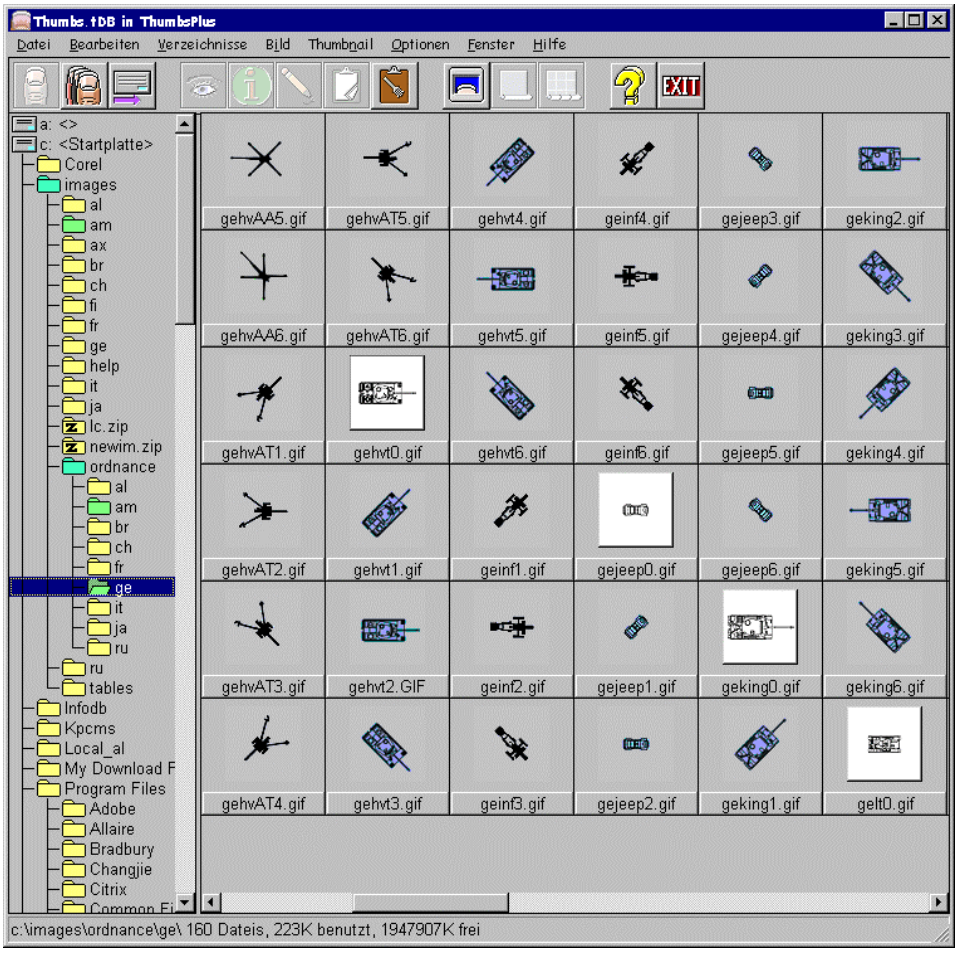
Name the file **VASLdata.zip**

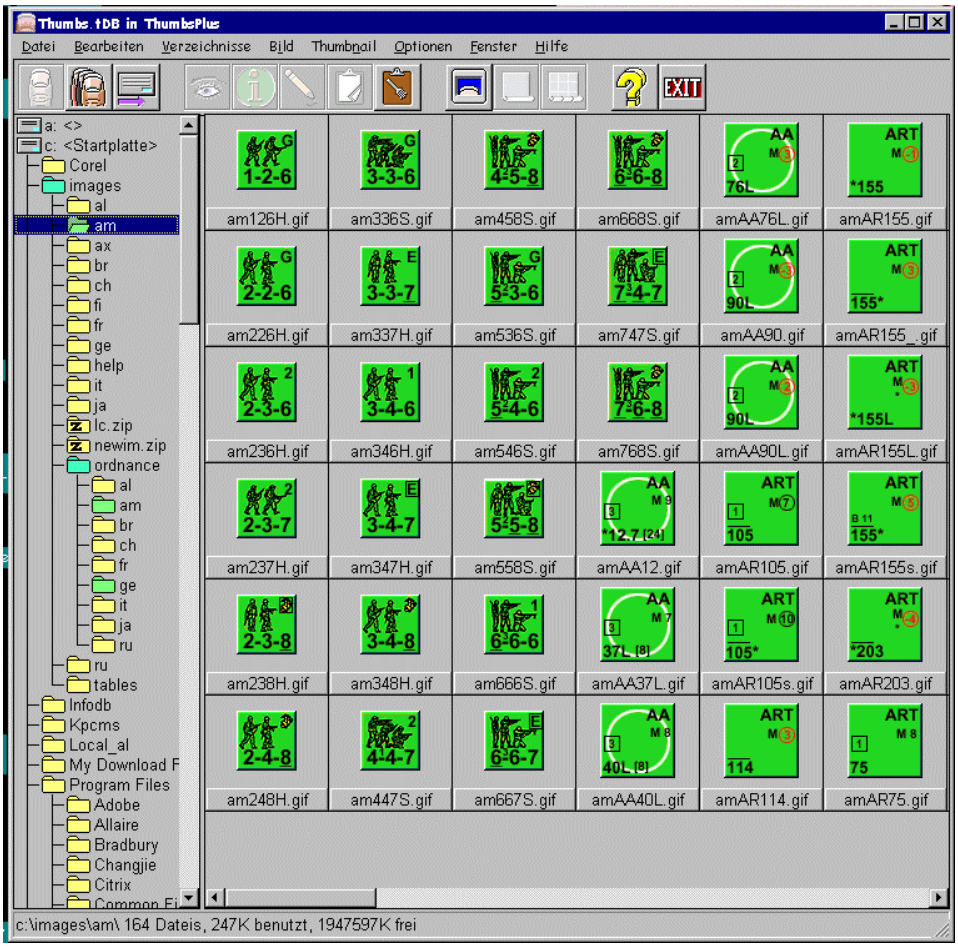
VASL Mapmaking Secrets

Backup the original file called **VASLdata**
Copy your **VASLdata.zip** into the VASL directory
Rename the file to **VASLdata** – no extension no matter what Windows says

7.2.1. Directory Structure for "IMAGES"




Thumbs.tDB in ThumbsPlus
Datei Bearbeiten Verzeichnisse Bild Thumbnail Optionen Fenster Hilfe
EXIT
a: <>
c: <Startplatte>
Corel
images
al
am
ax
br
ch
fi
fr
ge
help
it
ja
lc.zip
newim.zip
ordnance
al
am
br
ch
fr
ge
it
ja
ru
ru
tables
Infodb
Kpcms
Local_al
My Download F
Program Files
Adobe
Allaire
Bradbury
Changjie
Citrix
gehwAA5.gif gehwAT5.gif gehvt4.gif geinf4.gif gejeep3.gif geking2.gif
gehwAA6.gif gehwAT6.gif gehvt5.gif geinf5.gif gejeep4.gif geking3.gif
gehwAT1.gif gehvt0.gif gehvt6.gif geinf6.gif gejeep5.gif geking4.gif
gehwAT2.gif gehvt1.gif geinf1.gif gejeep0.gif gejeep6.gif geking5.gif
gehwAT3.gif gehvt2.GIF geinf2.gif gejeep1.gif geking0.gif geking6.gif
gehwAT4.gif gehvt3.gif geinf3.gif gejeep2.gif geking1.gif gelt0.gif
c:\images\ordnance\ge\ 160 Dateis, 223K benutzt, 1947907K frei



7.3. Nationality Colour Chart

Since the question comes up again and again, here are the Pantone Color Codes for the ASL counters. Since the VASL values are different from the original counter colors I'm going to list these too.

Nationality	Pantone	RGB Value	VASL RGB Value
German	277	184 232 255	106 184 255
British	155	255 232 184	229 206 160
U.S.	390	207 255 43	102 204 0
Russian	124	240 169 0	145 145 0
Italian	Cool Grey 6	158 158 158	166 173 178
Allied Minor	351	102 255 54	130 237 189
Axis Minor	361	217 255 217	29 226 86
Japanese	115	255 232 54	255 219 0
French	297	125 255 255	0 140 255

7.4. Making Counters in VASL v4

This section temporarily left blank until VASL v4 comes out. V4 will allow the user to make and use his own counters; we'll discuss that procedure here at that time.

Appendix A - The v5 Geo Board Color Palette

The v5 color palette sprung up from a desire to improve the look of the basic color scheme – the "cool mint ice" color of v4's Open Ground Level 0 (RGB 204,240,153) was slowly burning a hole in people's retinas. With that as a starting point, the entire palette received a makeover, generally taking on a more drab and real-world tone. Along the way, the concept of Cut & Paste artwork came up, with the number of colors dedicated to C&P terrain being determined by how many colors it took to look real enough. Although the standard geo board set does not go down to Level -2 or up to Level 6 as of this writing, the Korea module in development does contain Level 5 hills, so we feel like we've got the terrain space bounded.

As noted in Section XXX, this palette is only for standard geo boards – when you color a HASL map, you are free to use whatever palette you see fit, since that HASL board probably won't need to accommodate VASL color changes. On the other hand, even if you want to deviate from this standard set for your own geo boards, VASL lets you do that by listing your own colors in the board's **colors** file.

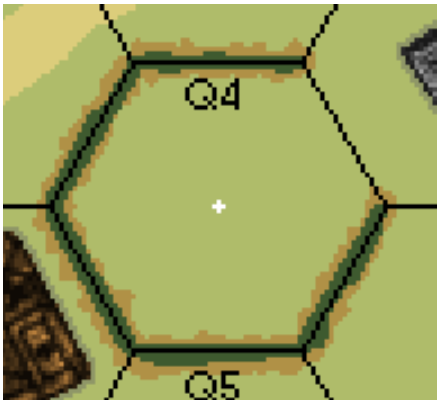
See the following pages for a listing of the various colors' RGB values and their corresponding terrain types. Each color name is followed by its RGB triplet. The color names given here are exactly the same as those that are in the **colors** and **colorSSR** files. Where a terrain type has level-specific colors, remember that the "L_2" and "L_1" colors are for valley levels below level 0; the colors for levels 1 and 2 have names with "L1" and "L2" suffixes.

In a break from the past, we aren't specifying the Palette Indices anymore – we aren't saying which color occupies Position X of the 256-color palette for a standard gif. While indices were useful in their day, the State of the Art seems to have passed them by, and we get along without them just fine without all the baggage that goes along with them. Also, it was misleading to specify "the standard geo palette", as if VASL gifs could only have those 256 specific colors – in reality, VASL gifs can use any 256 colors, as long as VASL is told how to handle them. These "standard" colors are simply the ones that VASL comes pre-equipped to handle through its global **boardData** files.

A final note - rather than trying to draw some of this terrain by hand, we recommend you use the terrain feature and pattern libraries available online at the VASL Map Bazaar. Save your creativity for designing new boards instead of mucking around trying to make those walls and hedges look just right.

Ye Old Booke

Bocage	R	G	B
BocageGreen	62	90	47
BocageBrownL0	176	145	70
BocageBrownL1	149	113	33
BocageBrownL2	123	92	27

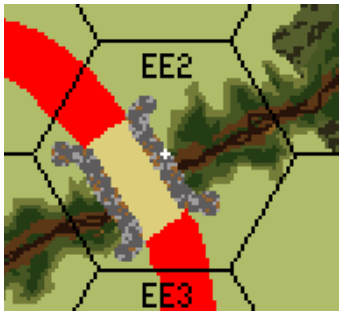


To make a Wall/Hedge To Bocage overlay, select all of the Wall or Hedge colors on the map. Then Fill Selection with BocageBrown. Then shrink the selection by two pixels and fill the selection with BocageGreen. Then edit by hand to make it look good.

The various BocageBrown colors are used to stand out well against their respective Hill levels. For aesthetic purposes, though, I like to use just one BocageBrown for an entire board, even if it contains Bocage at several different levels.

Bridges	R	G	B
BridgeBrown Also used for Canals	153	99	43
BridgeGrey1	159	157	157
BridgeGrey2	93	91	91
DirtRdBridge	220	206	123
PavedRdBridge	192	193	207

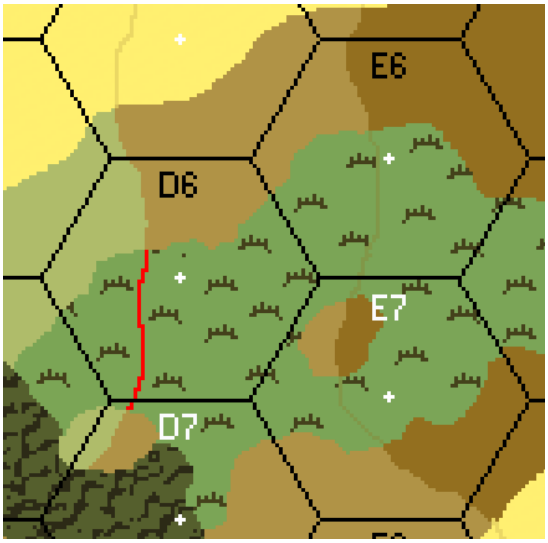
The Paved/Dirt RdBridge colors are the colors of the road going over the bridge inside the bridge itself, not the parts of the road leading up to the bridge in that hex - use Paved/Dirt Rd colors for that. The separate colors exist to enable us to remove the general road net without affecting the Bridges.



The DirtRdL0 color is shown in red here; the DirtRdBridge color is the light brown part of the road that actually crosses over the bridge.

Brush	R	G	B
BrushL_2	123	165	83
BrushL_1	123	165	84
BrushL0	123	165	85
BrushL1	123	165	86
BrushL2	123	165	87
BrushL3	123	165	88
BrushL4	123	165	89
BrushL5	123	165	90
BrushL6	123	165	91
BrushBlackL_2	68	62	27
BrushBlackL_1	68	63	27
BrushBlackL0	68	64	27
BrushBlackL1	68	65	27
BrushBlackL2	68	66	27
BrushBlackL3	68	67	27
BrushBlackL4	68	68	27
BrushBlackL5	68	69	27
BrushBlackL6	68	70	27
BrushCrestL_1	142	173	91
BrushCrestL0	142	173	92
BrushCrestL1	129	151	71
BrushCrestL2	129	151	72
BrushCrestL3	129	151	73
BrushCrestL4	129	151	74
BrushCrestL5	129	151	75
BrushCrestL6	129	151	76

BrushCrests are brownish-greens that show the crestlines underneath Brush; they are an often-preferable alternative to creating "holes" in the Brush field to show the underlying crestlines.



BrushCrestLevel1 shown in red above; BrushCrestLevel2 is in hexes E6 and E7.

Building Shadows	R	G	B
BldgShadowL_1	101	123	63
BldgShadowL0	143	151	103
BldgShadowL1	149	125	59
BldgShadowL2	127	95	27
BldgShadowL3	96	71	16

Building Shadows are two pixels wide for DASL boards and one pixel wide for standard geo boards. The easy way to create them is to duplicate the entire Building layer to a new layer underneath. In that layer: select all, deselect the transparent color, grow the selection one pixel, and then fill with the BldgShadow color. Lumberyards get these too.

Cliffs	R	G	B
Cliff	35	20	10
GullyCliff	36	20	10

Crags	R	G	B
CragGrey	184	184	176
CragBlack	68	68	67

Black is the outline of the crag; grey is the color inside the black.

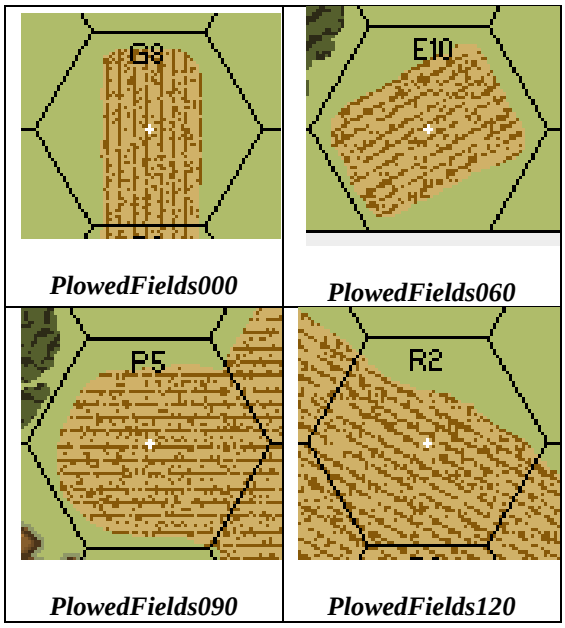
Dirt Roads	R	G	B
DirtRdL_2	220	205	121
DirtRdL_1	220	205	122
DirtRdL0	220	205	123
DirtRdL1	220	205	124
DirtRdL2	220	205	125
DirtRdL3	220	205	126
DirtRdL4	220	205	127
DirtRdL5	220	205	128
DirtRdL6	220	205	129
DirtRdBridge	220	206	123

Grain	R	G	B
GrainL_2	255	240	109
GrainL_1	255	240	110
GrainL0	255	240	111
GrainL1	255	240	112
GrainL2	255	240	113
GrainL3	255	240	114
GrainL4	255	240	115
GrainL5	255	240	116
GrainL6	255	240	117
FurrowsL_2	255	239	109
FurrowsL_1	255	239	110
FurrowsL0	255	239	111
FurrowsL1	255	239	112
FurrowsL2	255	239	113
FurrowsL3	255	239	114
FurrowsL4	255	239	115
FurrowsL5	255	239	116
FurrowsL6	255	239	117
GrainCrestL_1	235	216	98
GrainCrestL0	235	216	99
GrainCrestL1	235	216	100
GrainCrestL2	235	216	101
GrainCrestL3	235	216	102
GrainCrestL4	235	216	103
GrainCrestL5	235	216	104
GrainCrestL6	235	216	105

Furrows	135	89	9
PlowedFields	208	177	104

Furrows are the parts of grainfields that become dark brown lines when Plowed Fields are in effect. The Furrows and PlowedFields colors are suggestions for HASL maps.

There are four Grain patterns available online at the VASL Map Bazaar. They are named according to the angle the furrows make with respect to direction 1 (the hexside with the hex ID) with Plowed Fields:



PlowedFields000 ***PlowedFields060***

PlowedFields090 ***PlowedFields120***

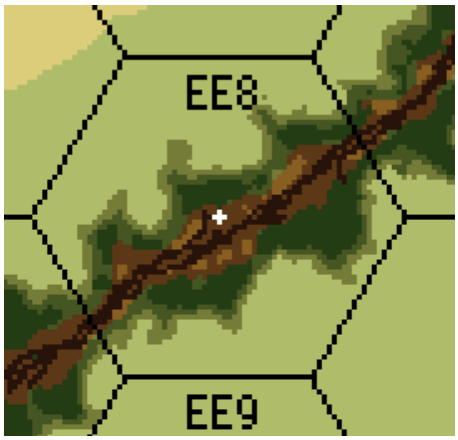
GrainCrests function like BrushCrests; they show the crestline under an unbroken field of grain as a nice alternative to creating "holes" in the grain. See the BrushCrest picture above; hex D5 contains GrainCrestL1.

Graveyard	**R**	**G**	**B**
GraveyardWhite1	204	215	171
GraveyardWhite2	164	164	132
GraveyardGreen	38	82	40
GraveyardShadow	53	62	15

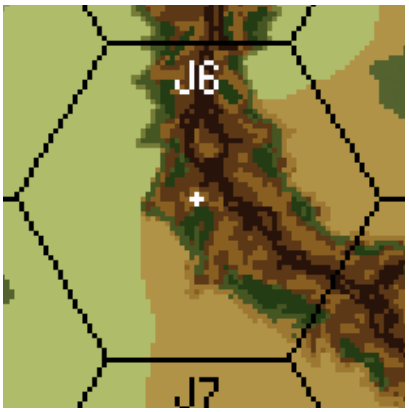
Graveyards look much better with Cut & Paste, rather than drawing from scratch, and there is also a Graveyard pattern available online. Paths through Graveyards use the normal Path colors.

Gullies	**R**	**G**	**B**
GullyGreen1	117	124	56
GullyGreen2	70	86	35
GullyGreen3	35	60	20
GullyBrown1 (for hill gullies)	156	120	51
GullyBrown2	136	95	33
GullyBrown3	93	56	22
GullyBlack	40	19	10

Gullies look good with Cut & Paste techniques, although it is often necessary to tweak things at length in order to get good-looking results.



From the outer edge inward, the Gully colors are: GullyGreen1, Green2, Green3, Brown2, Brown3, and Black.



Note the use of GullyBrown1 at the outer edge of a Hill Gully.

Hedge	R	G	B
HedgeGreen	62	91	47

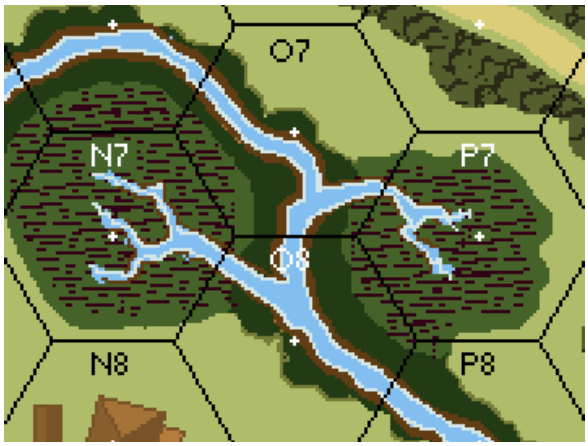
Lumberyards	R	G	B
Lumberyard1	166	115	55
Lumberyard2	135	94	46
Lumberyard3	105	73	36
Lumberyard4	75	52	26
Lumberyard5	45	31	16

Lumberyard s are similar to Wooden Buildings, but the unique colors enable them to be immune to color changes that affect Wooden Buildings. Lumberyards also stand out better if you give them Building Shadows.

Marsh	R	G	B
MarshGreenL_2	68	98	39
MarshGreenL_1	68	98	40
MarshGreenL0	68	98	41
MarshGreenL1	68	98	42
MarshBrownL_2	49	0	22
MarshBrownL_1	50	0	22
MarshBrownL0	51	0	22
MarshBrownL1	52	0	22
SeparatedMarshGreenL_2	67	98	39
SeparatedMarshGreenL_1	67	98	40
SeparatedMarshGreenL0	67	98	41
SeparatedMarshGreenL1	67	98	42
SeparatedMarshBrownL_2	49	0	21
SeparatedMarshBrownL_1	50	0	21
SeparatedMarshBrownL0	51	0	21
SeparatedMarshBrownL1	52	0	21
SwampGreenL_2	69	98	39
SwampGreenL_1	69	98	40
SwampGreenL0	69	98	41
SwampGreenL1	69	98	42
SwampBrownL_2	49	1	22
SwampBrownL_1	50	1	22
SwampBrownL0	51	1	22
SwampBrownL1	52	1	22

Marsh and SeparatedMarsh are different because they behave differently under Flooded/Dry conditions as well as in the PTO. SeparatedMarsh is a Marsh hex that is not adjacent to a Woods hex. Marsh turns into a water hex when a water obstacle is Flooded, but Separated Marsh does not.

Marsh hexes in the PTO that are adjacent to a Jungle (Woods) hex are Swamps. Swamp needs to be different from both types of Marsh because Swamp does not turn into Open Ground when there's Ground Snow while both types of Marsh do. Swamp is also not affected by the water depth of a stream or river.



It's often a nice effect to have StreamWater and StreamWhite "tendrils" penetrating into Marsh from adjacent Streams.

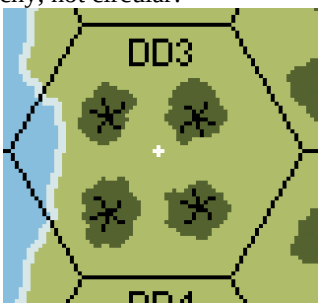
Miscellaneous	R	G	B
Black	0	0	0
WhiteHexNumbers	251	252	253
ManHoles	0	1	0
BoardNum 36pt Basic Sans SF Bold in hex B8	255	255	11
HillNum 12 pt Verdana Bold	255	255	10
BarbedWire	59	50	43
White	255	255	255
Slope	121	89	57
VineyardGreen	84	100	45
Transparent (light pink)	238	176	237
DuttyWhiteHexDots	255	254	253

White hexdots (255, 255, 255) stay white even in Snow, while DuttyWhite hexdots (255, 254, 253) become grey in Snow in order to remain visible against a white background. Thus, use DuttyWhite for hexdots in Level 0 terrain that can go to White in Snow: Open Ground, Grain, Brush, and Marsh.

Open Ground	R	G	B
Level_2	88	110	50
Level_1	119	146	74
Level0	175	188	106
Level1	176	147	70
Level2	147	111	31
Level3	118	80	0
Level4	94	57	0
Level5	74	45	0
Level6	56	34	0

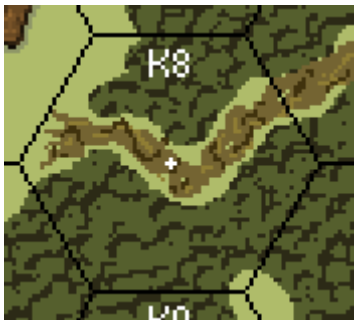
Orchards	R	G	B
OrchL_2	85	100	46
OrchL_1	83	98	45
OrchL0	83	98	46
OrchL1	83	98	47
OrchL2	83	98	48
OrchL3	83	98	49
OrchL4	83	98	50
OrchL5	83	98	51
OrchL6	83	98	52
OrchTrees	82	98	46

An OrchTree is the squiggly asterisk-type thing that stays onboard when orchards are out of season. The level-specific orchard colors revert to their level colors when the orchard is out of season. And please, for the love of Pete, make the Orchard shapes blotchy, not circular!



OrchTrees are shown in black here.

Paths	R	G	B
PathBrown1	143	125	52
PathBrown2	116	100	50
PathBrown3	64	50	13



Dark PathBrown1 and light PathBrown3 speckles on the medium PathBrown2 background

The idea with paths is to create a "braided strand" look with the light and dark brown on top of the medium brown. Path colors are also used for Fords.

Some paths have a thin border of Open Ground between the PathBrown2 and the woods on that side. Other paths have little sprays of PathBrown2 dots leading out into open ground from where the path begins at the tree line. Shrug. To each his own.

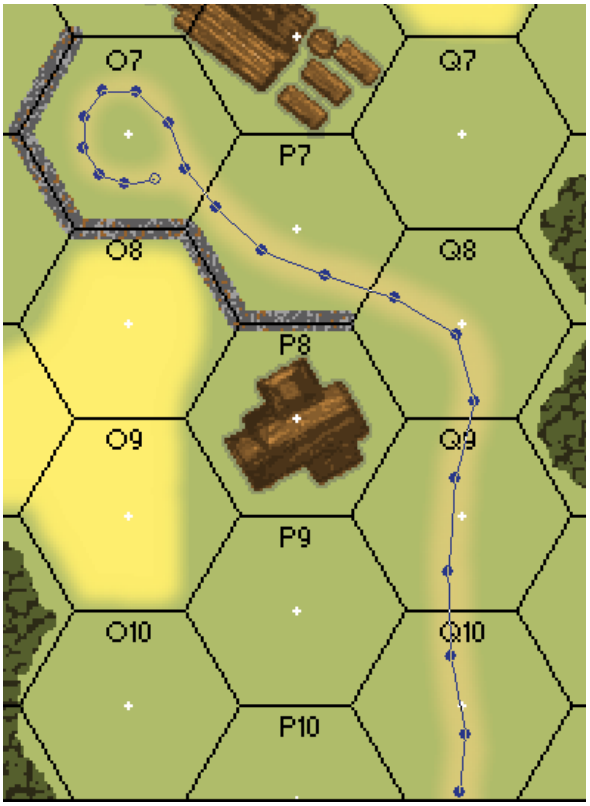
Remember, though, that Paths still need to point to hexsides even under Dense Jungle, so make them long enough to reach the hexside and don't obscure the Path artwork with the Dense Jungle artwork.

Paved Roads	R	G	B
PavedRdL_2	192	192	205
PavedRdL_1	192	192	206
PavedRdL0	192	192	207
PavedRdL1	192	192	208
PavedRdL2	192	192	209
PavedRdL3	192	192	210
PavedRdL4	192	192	211
PavedRdL5	192	192	212
PavedRdL6	192	192	213
PavedRdBridge	192	193	207

Paved roads on SK-style hardcopy boards have a thin dark grey border to improve visibility. VASL v5 boards don't follow this convention, as the thin grey line looks too pixellated at 72dpi.

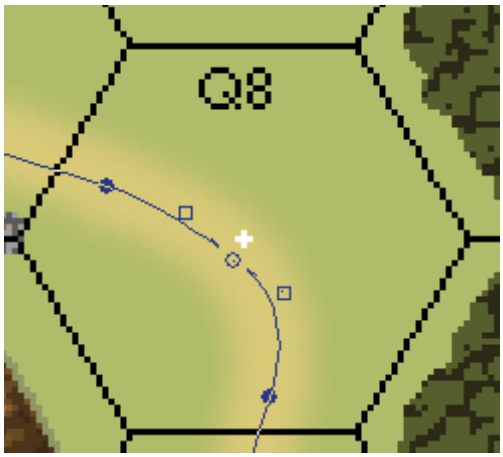
For Roads of all kinds, we highly recommend getting familiar with drawing paths – not ASL Paths, but the vector-based paths you can create in most modern drawing apps. These make it easy to construct roads with nice smooth edges – something almost impossible to do by hand with a mouse.

In GIMP, select the "Create and edit paths" tool and go into Design mode. On the map's Road layer, click on the path the road will take – here it helps to be working from a scan, as you can simply click on the center of the road as it travels across the board.



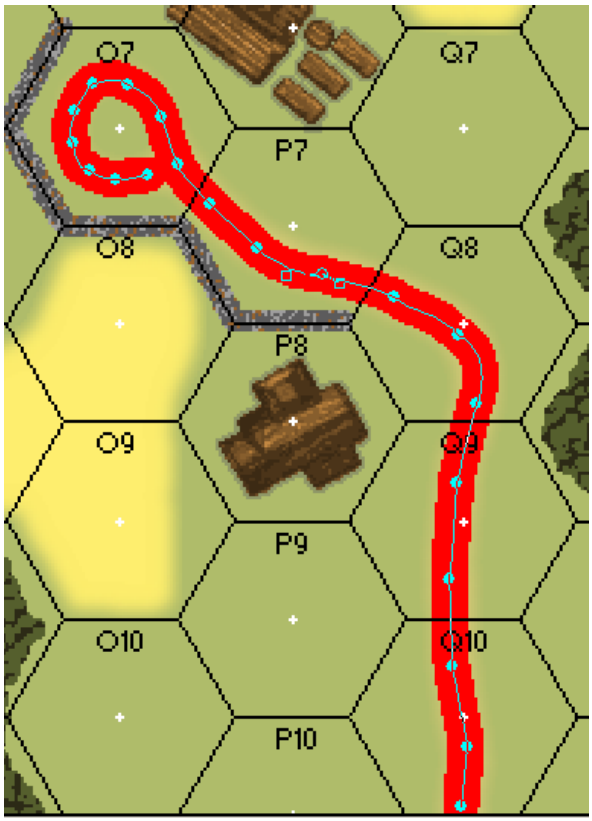
Constructing a road skeleton in Design Mode using GIMP. The blurry brown line represents the scanned bitmap of the road (in a layer underneath the Road layer).

Once the road skeleton is constructed, switch to Edit Mode and tweak the control handles at each skeleton point, pulling them away from the point so that the road line curves to follow the scanned road underneath.



Control Handles (square boxes) pulled away from their skeleton point (open circle) to make the road line follow the curve of the scanned road underneath.

After the entire road skeleton is tweaked to follow the road scan, switch to hit the Stroke Path button and set the parameters of the line you're going to stroke (draw) along the now-curvy road skeleton. The Line Width is typically 12 pixels, with Round Cap and Join styles, ***NO*** Antialiasing, and stroking with a Solid color.

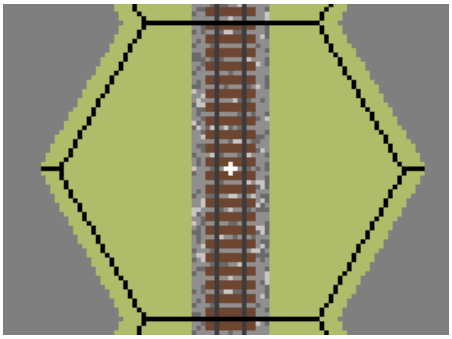


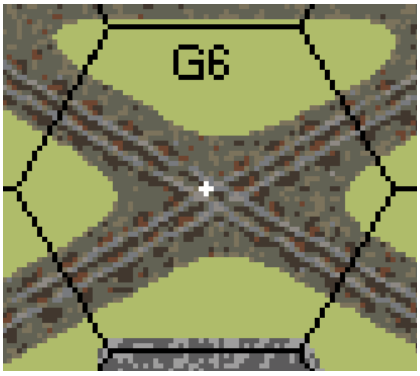
Curved road stroked with red.

Exit from the Create Paths tool and fill in the Road with the correct level-specific color, tweaking by hand as necessary (usually around intersections). Voila, a nice smoothed-edge road net.

Rails	R	G	B
Rails1	120	122	126
Rails2	60	60	60
RailTies	110	69	48
RailBed1	201	197	197
RailBed2	144	142	142
RailBed3	107	105	105

Rail lines generally take 5-6 colors to depict well. The first image below is from the hand-drawn RR1 overlay; only one Rail color was necessary, along with the RailTies and RailBed colors. The second image is from Bounding Fire Productions board BFPA; here, the Cut & Paste artwork looked better with two separate colors for the rail lines, so Rails1 and Rails2 were both used. Note the color substitutions for all the Rail colors, done to be more true to the hardcopy colors on that map.





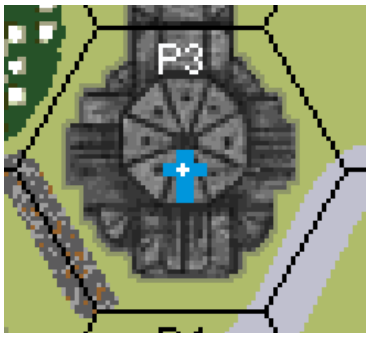
Rice Paddies	R	G	B
RicePaddy	103	75	39
PaddyInterior	174	168	78

Shellholes	R	G	B
Shellhole1	177	136	57
Shellhole2	97	68	27
Shellhole3	48	23	8

Shellholes use Cut & Paste artwork. When putting Shellholes on hills (perhaps as part of an Orchard/ Crag-to-Shellhole overlay), you may need to change Shellhole1 to Shellhole2 in order to avoid blending in with Open Ground Level 1. This is known as a chocolate shellhole.

Stone Building	R	G	B
StoneBldg1	25	25	25
StoneBldg2	40	40	40
StoneBldg3	55	55	55
StoneBldg4	70	70	70
StoneBldg5	85	85	85
StoneBldg6	100	100	100
StoneBldg7	115	115	115
StoneBldg8	130	130	130
StoneBldg9	145	145	145
StoneBldg10	160	160	160
StoneBldg11	175	175	175
StoneBldg12	190	190	190
StoneBldg13	205	205	205
StoneBldg14	220	220	220
StoneBldg15	235	235	235
StoneStairwell	251	254	255
StoneRowhouseBar	1	1	0

Be sure to keep a small hex center dot, square, or cross of pure White inside the larger Stairwell square so that the hex center dot is still there if NoStairwells is invoked. Use the stone stairwell color for steeple crosses in stone buildings as well.

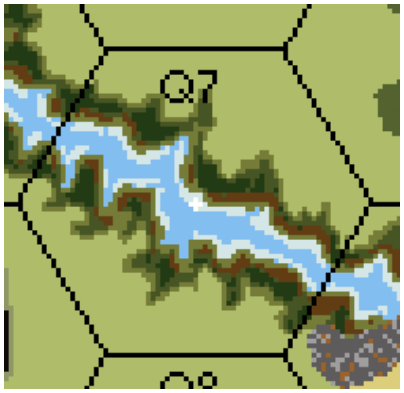


Steeple cross, showing the pure white hex center dot in the middle, surrounded by the StoneStairwell white cross (shown in blue here for clarity)

Coloring the hex center dot of a steeple in black conforms with the ASLRB, but has not been strictly adhered to by steeple-makers on VASL maps because on real maps, you need the black dot to provide a reference to string a LOS to, but this is not necessary with VASL - it strings LOS to the center of the hex automatically.

Streams	**R**	**G**	**B**
StreamBrown	98	61	19
StreamGreen1	117	124	55
StreamGreen2	70	85	35
StreamGreen3	35	59	20
StreamWhite	213	230	230
StreamWater	131	190	240

Streams greatly benefit from Cut & Paste artwork.



Under Winter or Mud conditions, StreamGreen1 goes to Level 0 Open Ground, so for streams that exist at other levels besides level 0 (like on board 47), you can define a different color in that board's **colors** file (e.g., something closer to Level -1 for board 47) and use that board's **colorSSR** file to have that color go to the correct level-specific color under Winter or Mud conditions.

Sunken Road	**R**	**G**	**B**
SunkRoad1	146	110	30
SunkRoad2	94	57	23

The same two colors get used for both Sunken and Elevated Roads. In both cases, the darker color is used for the higher terrain (outside part of the Sunken Road, inside part of the Elevated Road)

Walls	**R**	**G**	**B**
WallBrown	152	99	44
WallGrey1	158	157	157
WallGrey2	92	91	92

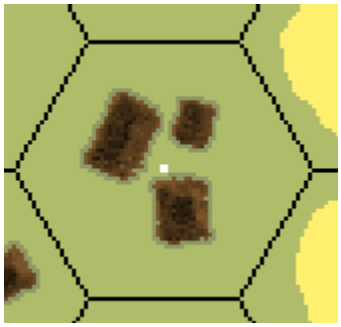
Walls and Bridges on SK-style hardcopy boards have a definite 3-D effect, with a thin dark shadow "behind" the wall and a thin white highlight "in front of" it. We find this distracting and have not replicated this in VASL.

Water	**R**	**G**	**B**
Water	135	190	220
WaterWhite	214	230	230
PondWater	135	190	221
PondWhite	215	230	230

Wooden Building	R	G	B
WoodBldg1	15	10	5
WoodBldg2	30	21	10
WoodBldg3	45	31	15
WoodBldg4	60	42	20
WoodBldg5	75	52	25
WoodBldg6	90	62	30
WoodBldg7	105	73	35
WoodBldg8	120	83	40
WoodBldg9	135	94	45
WoodBldg10	150	104	50
WoodBldg11	166	115	54
WoodBldg12	181	125	59
WoodBldg13	194	135	66
WoodBldg14	199	145	81
WoodBldg15	204	154	96
WoodStairwell	250	254	255
WoodRowhouseBar	2	1	0

Woods	R	G	B
WoodsGreen	85	95	45
WoodsBlack	44	42	22

Just as for Stone Buildings, be sure to keep a small hex center dot of pure White inside the larger Stairwell square so that the hex center dot is still there if NoStairwells is invoked. Use the WoodStairwell color for steeple crosses in wooden buildings as well.



Good Huts have a ragtag unkempt look about them.

Desert Colors – unless specified here, a color is the same as it is in the ETO [EX: BrushBlack, Crags, Walls].

Broken Ground	R	G	B
BrokenGround	69	69	69

Brush	R	G	B
BrushDesertL_1	180	166	70
BrushDesertL0	204	194	91
BrushDesertL1	213	203	95
BrushDesertL2	213	203	96
BrushDesertL3	213	203	97
BrushDesertL4	213	203	98
BrushDesertL5	213	203	99
BrushDesertL6	213	203	100
BrushCrestDesertL_1			
BrushCrestDesertL0	214	201	117
BrushCrestDesertL1	204	178	72
BrushCrestDesertL2	198	181	68
BrushCrestDesertL3	191	175	63
BrushCrestDesertL4	183	168	69
BrushCrestDesertL5	175	161	68

Building Shadows	R	G	B
BldgShadowDesertL_1	166	155	114
BldgShadowDesertL0	188	177	130
BldgShadowDesertL1	167	143	79
BldgShadowDesertL2	142	115	58

Cactus	R	G	B
CactusBrown1	222	189	99
CactusBrown2	90	66	33
CactusGreen	123	132	8

Cliff	R	G	B
DesertCliff	67	47	34
WadiCliff	66	47	34

Deir	R	G	B
DeirBase	233	212	155

Dirt Road	R	G	B
DirtRdDesertL_1	200	173	102
DirtRdDesertL0	215	188	113
DirtRdDesertL1	229	201	127
DirtRdDesertL2	229	201	128
DirtRdDesertL3	229	201	129
DirtRdDesertL4	229	201	130
DirtRdDesertL5	229	201	131
DirtRdDesertL6	229	201	132
DirtRdDesertBridge	215	188	114

Hammada	R	G	B
Hammada1	59	55	18
Hammada2	210	210	210

Hedge	R	G	B
DesertHedgeGreen	131	132	46

Hillock	R	G	B
Hillock	145	107	28
HillockCrest	116	84	41

Miscellaneous	R	G	B
DesertBoardNum	200	160	20

Open Ground	R	G	B
DesertL_2	181	166	104
DesertL_1	202	185	116
DesertL0	227	211	149
DesertL1	206	176	96
DesertL2	176	139	58
DesertL3	145	107	27
DesertL4	116	84	42
DesertL5	94	68	34
DesertL6	71	52	26

Orchard	R	G	B
OrchDesertL_2	130	138	66
OrchDesertL_1	130	138	67
OrchDesertL0	130	138	68
OrchDesertL1	130	138	69
OrchDesertL2	130	138	70
OrchDesertL3	130	138	71
OrchDesertL4	130	138	72
OrchDesertL5	130	138	73
OrchDesertL6	130	138	74
OrchTreesDesert	130	139	68

Sand	R	G	B
Sand1	217	184	53
Sand2	197	155	92
DuneCrest	120	79	48

Scrub	R	G	B
Scrub1 (interior)	153	96	60
Scrub2 (exterior)	119	65	47

Wadi	R	G	B
WadiBrown1	206	176	118
WadiBrown2	186	153	88
WadiBrown3	119	86	57
WadiBlack	83	51	37
WadiCliff	66	47	34

Ye Old Booke of VASL Secrets

Appendix C - Big Table of v5-Winter-Mud-Desert Color Swaps

This table presents VASL's color changes in a way that makes sense for certain tasks. The left-hand columns show the terrain type with its v5 name, followed by its v5 RGB triplet. The remaining columns show the RGB triplets for that color under Winter, Mud, and Desert conditions.

Notes:

	When a color does not change under Winter, Mud, or Desert, its entries for those columns are dark grey.
L1Winter	If a color change exists but is named somewhere else, the color name is given in italics.
	If a color change has not been decided on yet, the entry is blank. Hey, it's a work in progress.

	v5			Winter			Mud			Desert		
	R	**G**	**B**	**R**	**G**	**B**	**R**	**G**	**B**	**R**	**G**	**B**
Bocage												
BocageGreen	62	90	47	*HedgeGreen Winter*						131	132	46
BocageBrown	176	145	70	*L2Winter*			*MudLevel1*					
HillBocageBrown	149	113	33									
Bridges												
BridgeBrown	153	99	43	*WallBrown Winter*						174	130	83
BridgeGrey1	159	157	157	*StoneBldg10 Winter*						177	176	176
BridgeGrey2	93	91	91	*StoneBldg6 Winter*						124	123	124
Brush												
BrushL_2	123	165	83	196	207	187	138	149	89			
BrushL_1	123	165	84	196	207	187	138	149	89	180	166	70
BrushL0	123	165	85	196	207	187	138	149	89	204	194	91
BrushL1	123	165	86	196	207	187	138	149	89	213	203	95
BrushL2	123	165	87	196	207	187	138	149	89	213	203	96
BrushL3	123	165	88	196	207	187	138	149	89	213	203	97
BrushL4	123	165	89	196	207	187	138	149	89	213	203	98
BrushL5	123	165	90	196	207	187	138	149	89	213	203	99
BrushL6	123	165	91	196	207	187	138	149	89	213	203	100

	v5			Winter			Mud			Desert		
	R	G	B	R	G	B	R	G	B	R	G	B
BrushBlackL_2	68	62	27	*WoodsBlack Winter*								
BrushBlackL_1	68	63	27	*WoodsBlack Winter*								
BrushBlackL0	68	64	27	*WoodsBlack Winter*								
BrushBlackL1	68	65	27	*WoodsBlack Winter*								
BrushBlackL2	68	66	27	*WoodsBlack Winter*								
BrushBlackL3	68	67	27	*WoodsBlack Winter*								
BrushBlackL4	68	68	27	*WoodsBlack Winter*								
BrushBlackL5	68	69	27	*WoodsBlack Winter*								
BrushBlackL6	68	70	27	*WoodsBlack Winter*								
BrushCrestL_1	112	148	78	179	191	168	125	130	78			
BrushCrestL0	142	173	92	179	191	168	125	130	78	214	201	117
BrushCrestL1	129	151	71	179	191	168	125	130	78	204	178	72
BrushCrestL2	132	144	65	179	191	168	125	130	78	198	181	68
BrushCrestL3	121	139	61	179	191	168	125	130	78	191	175	63
BrushCrestL4	114	131	61	179	191	168	125	130	78	183	168	69
BrushCrestL5	106	125	60	179	191	168	125	130	78	175	161	68
BrushCrestL6	109	138	72	179	191	168	125	130	78			
BuildingShadows												
BldgShadowL_1	101	123	63	183	195	161	98	91	48	166	155	114
BldgShadowL0	143	151	103	206	206	206	148	141	92	188	177	130
BldgShadowL1	149	125	59	186	179	158	138	119	66	167	143	79
BldgShadowL2	127	95	27	177	160	127	118	95	46	142	115	58
BldgShadowL3	96	71	16	141	126	93	100	76	24			
Cliffs												
Cliff	35	20	10	101	83	71				67	47	34
GullyCliff	36	20	10	101	83	71				66	47	34

	v5				Winter				Mud				Desert		
	R	G	B		R	G	B		R	G	B		R	G	B
Crags															
CragGrey	184	184	176		*StoneBldg12 Winter*										
CragBlack	68	68	67		*StoneBldg4 Winter*										
DirtRoads															
DirtRdL_2	220	205	121		235	231	205		164	147	82				
DirtRdL_1	220	205	122		235	231	205		164	147	82		200	173	102
DirtRdL0	220	205	123		235	231	205		164	147	82		215	188	113
DirtRdL1	220	205	124		235	231	205		164	147	82		229	201	127
DirtRdL2	220	205	125		235	231	205		164	147	82		229	201	128
DirtRdL3	220	205	126		235	231	205		164	147	82		229	201	129
DirtRdL4	220	205	127		235	231	205		164	147	82		229	201	130
DirtRdL5	220	205	128		235	231	205		164	147	82		229	201	131
DirtRdL6	220	205	129		235	231	205		164	147	82		229	201	132
DirtRdBridge	220	206	123		235	231	205		164	147	82		215	188	114
Grain															
GrainL_2	255	240	109		*L_2Winter*				230	230	127				
GrainL_1	255	240	110		*L_1Winter*				230	230	127		239	217	95
GrainL0	255	240	111		*White*				230	230	127		244	223	106
GrainL1	255	240	112		*L1Winter*				230	230	127		244	223	107
GrainL2	255	240	113		*L2Winter*				230	230	127		244	223	108
GrainL3	255	240	114		*L3Winter*				230	230	127		244	223	109
GrainL4	255	240	115		*L4Winter*				230	230	127		244	223	110
GrainL5	255	240	116		*L5Winter*				230	230	127		244	223	111
GrainL6	255	240	117		*L6Winter*				230	230	127		244	223	112
FurrowsL_2	255	239	109		*L_2Winter*				100	69	14				
FurrowsL_1	255	239	110		*L_1Winter*				100	69	14		239	216	95
FurrowsL0	255	239	111		*White*				100	69	14		244	222	106
FurrowsL1	255	239	112		*L1Winter*				100	69	14		244	222	107
FurrowsL2	255	239	113		*L2Winter*				100	69	14		244	222	108
FurrowsL3	255	239	114		*L3Winter*				100	69	14		244	222	109
FurrowsL4	255	239	115		*L4Winter*				100	69	14		244	222	110
FurrowsL5	255	239	116		*L5Winter*				100	69	14		244	222	111
FurrowsL6	255	239	117		*L6Winter*				100	69	14		244	222	112

Ye Old Booke

	v5			Winter			Mud			Desert		
	R	G	B	R	G	B	R	G	B	R	G	B
GrainCrestL_1	235	216	98	*L_1Winter*								
GrainCrestL0	235	216	99	*White*								
GrainCrestL1	235	216	100	*L1Winter*								
GrainCrestL2	235	216	101	*L2Winter*								
GrainCrestL3	235	216	102	*L3Winter*								
GrainCrestL4	235	216	103	*L4Winter*								
GrainCrestL5	235	216	104	*L5Winter*								
GrainCrestL6	235	216	105	*L6Winter*								
Furrows	135	89	9									
PlowedFields	208	177	104	*White*								
Graveyard												
GraveyardWhite1	204	215	171	*StoneBldg15 Winter*						222	223	217
GraveyardWhite2	164	164	132	*StoneBldg11 Winter*						198	198	194
GraveyardGreen	38	82	40	190	202	190				189	173	107
GraveyardShadow	53	62	15	*StoneBldg3 Winter*						128	135	99
Gullies												
GullyGreen1	117	124	56	*White*			*MudStream Green1*			206	176	118
GullyGreen2	70	86	35	233	233	225	*MudStream Green2*			206	176	118
GullyGreen3	35	60	20	233	233	225	*MudStream Green3*			186	153	88
GullyBrown1 (forhillgullies)	156	120	51	*Level1Winter*			*MudGully Brown1*					
GullyBrown2	136	95	33	*WoodsBldg11 Winter*			*MudGully Brown2*			119	86	57
GullyBrown3	93	56	22	*WoodsBldg6 Winter*			*MudStream Brown*			119	86	57
GullyBlack	40	19	10	*WoodsBlack Winter*						83	51	37
GullyCliff	36	20	10	101	83	71				66	47	34
Hedge												
HedgeGreen	62	91	47	160	174	154				131	132	46

	v5			Winter			Mud			Desert		
	R	**G**	**B**	**R**	**G**	**B**	**R**	**G**	**B**	**R**	**G**	**B**
Lumberyards												
Lumberyard1	166	115	55	*WoodBldg11Winter*								
Lumberyard2	135	94	46	*WoodBldg9Winter*								
Lumberyard3	105	73	36	*WoodBldg7Winter*								
Lumberyard4	75	52	26	*WoodBldg5Winter*								
Lumberyard5	45	31	16	*WoodBldg3Winter*								
Marsh												
MarshGreenL_2	68	98	39	*L_2Winter*			74	84	42	139	124	56
MarshGreenL_1	68	98	40	*L_1Winter*			74	84	42	139	124	57
MarshGreenL0	68	98	41	*White*			74	84	42	139	124	58
MarshGreenL1	68	98	42	*L1Winter*			74	84	42	139	124	59
MarshBrownL_2	49	0	22	*L_2Winter*								
MarshBrownL_1	50	0	22	*L_1Winter*								
MarshBrownL0	51	0	22	*White*								
MarshBrownL1	52	0	22	*L1Winter*								
SeparatedMarshGreenL_2	67	98	39	*L_2Winter*			74	84	42	138	124	56
SeparatedMarshGreenL_1	67	98	40	*L_1Winter*			74	84	42	138	124	57
SeparatedMarshGreenL0	67	98	41	*White*			74	84	42	138	124	58
SeparatedMarshGreenL1	67	98	42	*L1Winter*			74	84	42	138	124	59
SeparatedMarshBrownL_2	49	0	21	*L_2Winter*								
SeparatedMarshBrownL_1	50	0	21	*L_1Winter*								
SeparatedMarshBrownL0	51	0	21	*White*								
SeparatedMarshBrownL1	52	0	21	*L1Winter*								
SwampGreenL_2	69	98	39	*L_2Winter*			74	84	42	140	124	56
SwampGreenL_1	69	98	40	*L_1Winter*			74	84	42	140	124	57
SwampGreenL0	69	98	41	*White*			74	84	42	140	124	58
SwampGreenL1	69	98	42	*L1Winter*			74	84	42	140	124	59
SwampBrownL_2	49	1	22	*L_2Winter*								
SwampBrownL_1	50	1	22	*L_1Winter*								
SwampBrownL0	51	1	22	*White*								
SwampBrownL1	52	1	22	*L1Winter*								

	v5			Winter			Mud			Desert		
	R	G	B	R	G	B	R	G	B	R	G	B
Miscellaneous												
Black	0	0	0									
WhiteHexNumbers	251	252	253									
ManHoles	0	1	0	*StoneRowhouseBar Winter*						34	35	35
BoardNum 36ptBasicSansSFBold	255	255	11	*Black*			90	77	42	200	160	20
HillNum 12ptVerdanaBold	255	255	10	*Black*			195	196	38			
BarbedWire	59	50	43									
White	255	255	255									
Slope	121	89	57									
VineyardGreen	84	100	45	*White*								
Transparent	238	176	237									
DuttyWhiteHexDots	255	254	253	168	182	203						
OpenGround												
Level_2	88	110	50	225	239	187	107	100	53	181	166	104
Level_1	119	146	74	213	224	194	127	119	63	202	185	116
Level0	175	188	106	255	255	255	170	160	95	227	211	149
Level1	176	147	70	222	213	188	154	131	70	206	176	96
Level2	147	111	31	195	178	145	135	106	45	176	139	58
Level3	118	80	0	164	144	102	118	87	23	145	107	27
Level4	94	57	0	149	125	87	91	64	10	116	84	42
Level5	74	45	0	130	105	70	74	50	10	94	68	34
Level6	56	34	0	110	85	60	56	38	5	71	52	26
Orchards												
OrchL_2	85	100	46	*L_2Winter*			65	90	48	130	138	66
OrchL_1	83	98	45	*L_1Winter*			65	90	48	130	138	67
OrchL0	83	98	46	*White*			65	90	48	130	138	68
OrchL1	83	98	47	*L1Winter*			65	90	48	130	138	69
OrchL2	83	98	48	*L2Winter*			65	90	48	130	138	70
OrchL3	83	98	49	*L3Winter*			65	90	48	130	138	71
OrchL4	83	98	50	*L4Winter*			65	90	48	130	138	72
OrchL5	83	98	51	*L5Winter*			65	90	48	130	138	73
OrchL6	83	98	52	*L6Winter*			65	90	48	130	138	74
OrchTrees	82	98	46	*Black*			65	90	48	130	139	68

	v5				Winter				Mud				Desert		
	R	**G**	**B**		**R**	**G**	**B**		**R**	**G**	**B**		**R**	**G**	**B**
Paths															
PathBrown1	143	125	52		*WoodBldg15 Winter*								192	154	74
PathBrown2	116	100	50		*WoodBldg12 Winter*								170	134	72
PathBrown3	64	50	13		*WoodBldg7 Winter*								147	100	27
PavedRoads															
PavedRdL_2	192	192	205		222	222	222		177	177	177		225	225	228
PavedRdL_1	192	192	206		222	222	222		177	177	177		225	225	229
PavedRdL0	192	192	207		222	222	222		177	177	177		225	225	230
PavedRdL1	192	192	208		222	222	222		177	177	177		225	225	231
PavedRdL2	192	192	209		222	222	222		177	177	177		225	225	232
PavedRdL3	192	192	210		222	222	222		177	177	177		225	225	233
PavedRdL4	192	192	211		222	222	222		177	177	177		225	225	234
PavedRdL5	192	192	212		222	222	222		177	177	177		225	225	235
PavedRdL6	192	192	213		222	222	222		177	177	177		225	225	236
PavedRdBridge	192	193	207		222	222	222		177	177	177		225	224	230
Rails															
Rails1	120	122	126												
Rails2	60	60	60												
RailTies	110	69	48												
RailBed1	201	197	197												
RailBed2	144	142	142												
RailBed3	107	105	105												
RicePaddies															
RicePaddy	103	75	39						153	126	21				
PaddyInterior	174	168	78		*White*										
Shellholes															
Shellhole1	177	136	57		*WoodBldg13 Winter*				146	117	62				
Shellhole2	97	68	27		*WoodBldg8 Winter*										
Shellhole3	48	23	8		*WinterCliff*										

Ye Old Booke

	v5			Winter			Mud			Desert		
	R	G	B	R	G	B	R	G	B	R	G	B
StoneBuilding												
StoneBldg1	25	25	25	90	90	90				*StoneBldg1Winter*		
StoneBldg2	40	40	40	107	107	107				*StoneBldg2Winter*		
StoneBldg3	55	55	55	115	115	116				*StoneBldg3Winter*		
StoneBldg4	70	70	70	123	123	123				*StoneBldg4Winter*		
StoneBldg5	85	85	85	132	132	132				*StoneBldg5Winter*		
StoneBldg6	100	100	100	148	148	148				*StoneBldg6Winter*		
StoneBldg7	115	115	115	156	156	156				*StoneBldg7Winter*		
StoneBldg8	130	130	130	165	165	165				*StoneBldg8Winter*		
StoneBldg9	145	145	145	181	181	181				*StoneBldg9Winter*		
StoneBldg10	160	160	160	189	189	189				*StoneBldg10Winter*		
StoneBldg11	175	175	175	198	198	198				*StoneBldg11Winter*		
StoneBldg12	190	190	190	214	214	214				*StoneBldg12Winter*		
StoneBldg13	205	205	205	222	222	221				*StoneBldg13Winter*		
StoneBldg14	220	220	220	231	231	231				*StoneBldg14Winter*		
StoneBldg15	235	235	235	247	247	247				*StoneBldg15Winter*		
StoneStairwell	251	254	255									
StoneRowhouseBar	1	1	0	71	70	70				34	34	34
Streams												
StreamBrown	98	61	19	*WoodBldg10 Winter*			80	61	38	*WadiBrown3*		
StreamGreen1	117	124	55	*White*			130	123	78	*WadiBrown1*		
StreamGreen2	70	85	35	233	233	225	101	94	45	*WadiBrown1*		
StreamGreen3	35	59	20	233	233	225	81	74	31	*WadiBrown2*		
StreamWhite	213	230	230				*MudWaterWhite*					
StreamWater	131	190	240	200	230	230	*MudWater*			181	206	227
SunkenRoad												
SunkRoad1	146	110	30	*L2Winter*			*MudLevel2*			176	139	59
SunkRoad2	94	57	23	*L4Winter*			*MudLevel4*			116	85	42
Walls												
WallBrown	152	99	44	177	150	121				173	130	83
WallGrey1	158	157	157	*StoneBldg10 Winter*						176	176	176
WallGrey2	92	91	92	*StoneBldg6 Winter*						123	123	124

	v5				Winter				Mud				Desert		
	R	**G**	**B**		**R**	**G**	**B**		**R**	**G**	**B**		**R**	**G**	**B**
Water															
Water	135	190	220		200	230	230		152	186	169		181	205	227
WaterWhite	214	230	230						170	175	147				
PondWater	135	190	221		200	230	230		*MudWater*				181	207	227
PondWhite	215	230	230						*MudWaterWhite*						
WoodenBuilding															
WoodBldg1	15	10	5		123	90	66						85	62	39
WoodBldg2	30	21	10		132	107	74						96	72	44
WoodBldg3	45	31	15		140	107	74						107	80	49
WoodBldg4	60	42	20		148	115	82						118	89	54
WoodBldg5	75	52	25		156	123	90						129	97	59
WoodBldg6	90	62	30		165	132	90						140	105	64
WoodBldg7	105	73	35		165	140	99						149	112	69
WoodBldg8	120	83	40		173	148	107						160	121	74
WoodBldg9	135	94	45		181	148	115						171	129	79
WoodBldg10	150	104	50		181	156	123						178	137	88
WoodBldg11	166	115	54		189	165	132						184	145	98
WoodBldg12	181	125	59		189	173	140						189	152	109
WoodBldg13	194	135	66		198	173	148						193	159	119
WoodBldg14	199	145	81		206	181	165						198	167	130
WoodBldg15	204	154	96		206	189	165						204	175	140
WoodStairwell	250	254	255												
WoodRowhouseBar	2	1	0		72	70	70						34	34	35
Woods															
WoodsGreen	85	95	45		166	171	145						169	157	101
WoodsBlack	44	42	22		94	90	75						83	79	41

Appendix D – Hill Number List

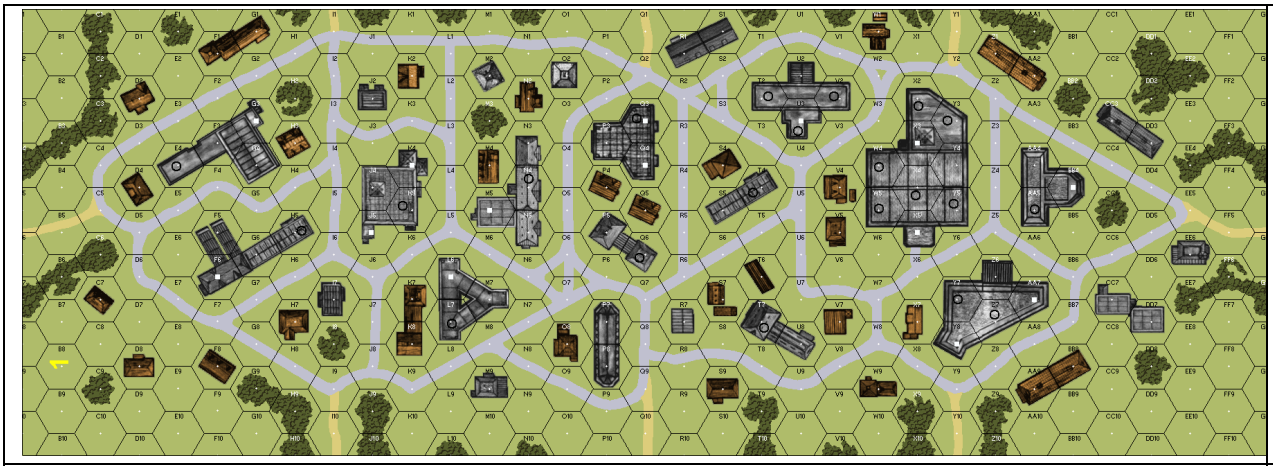
To help board designers avoid the crushing embarassment of assigning a hill number that's already been used, here is a list of what's out there. Yes, Hill 520 is on both boards 40 and 41.

Hill #	Board
490	36
497	41
498	3
502	36
507	36
513	41
520	41
520	40
522	3
526	36
534	3
538	2
547	3
621	2
654	9
714	15
733	9
740	9
758	25
766	25
779	39
783	15
792	25
801	36
804	36

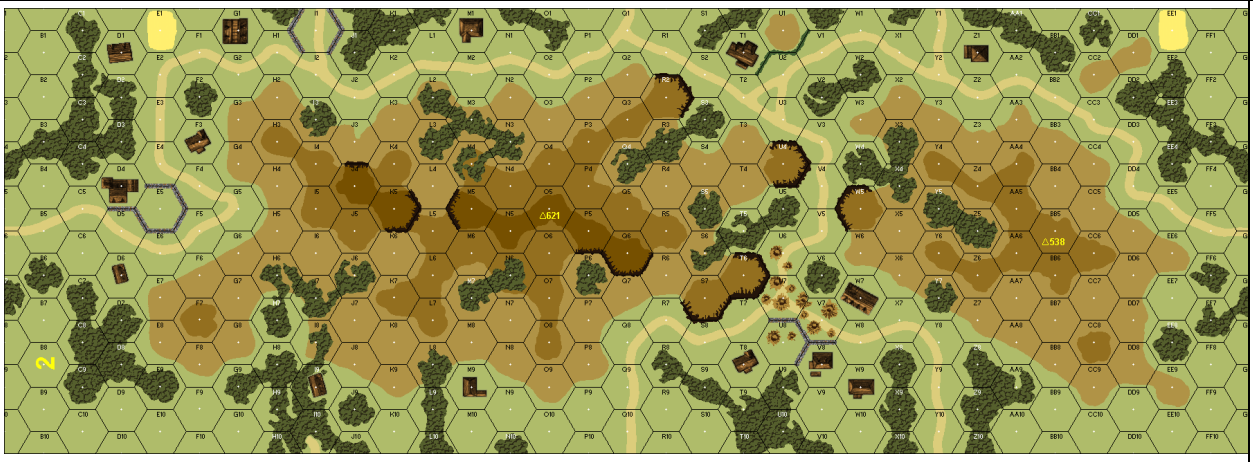
Appendix E - Thumbnails of Boards

Flagrantly stealing a good idea from Michael Dorosh's Scenario Design Handbook, here we present thumbnails of the existing official (and some unofficial) geo boards. Scenario designers may find this useful when searching for that special piece of terrain.

E.1 Official Geo Boards



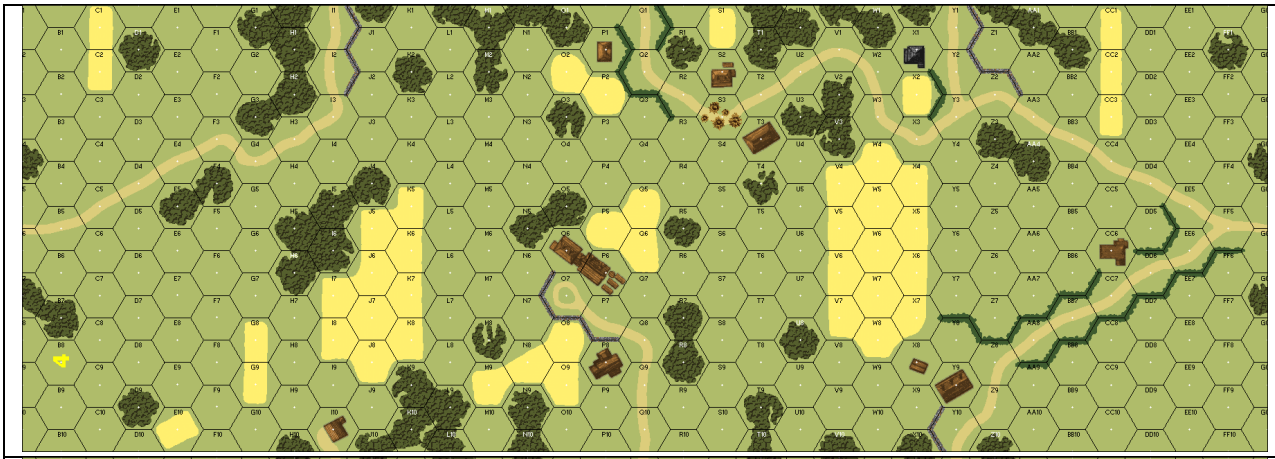
1



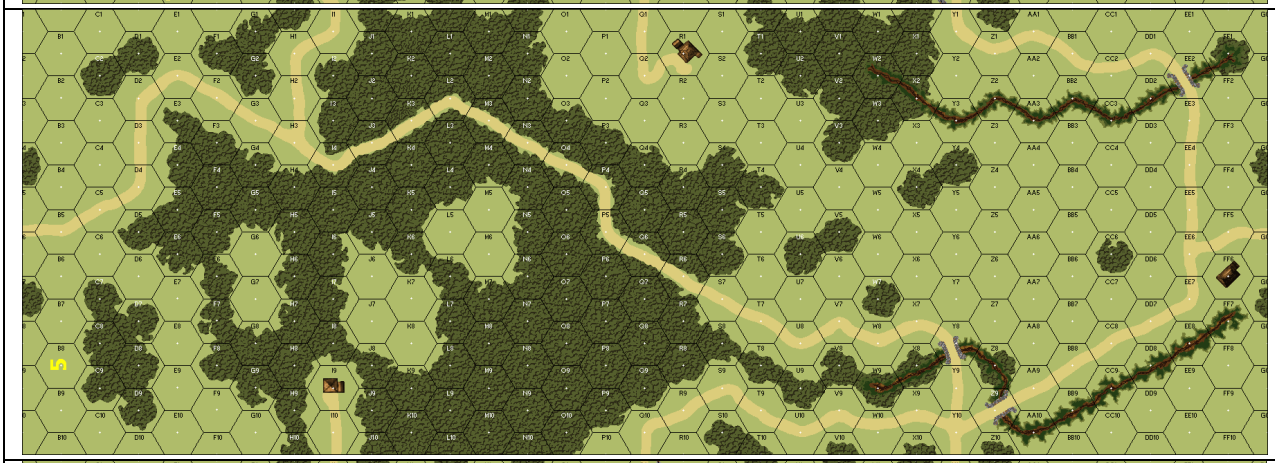
2

3

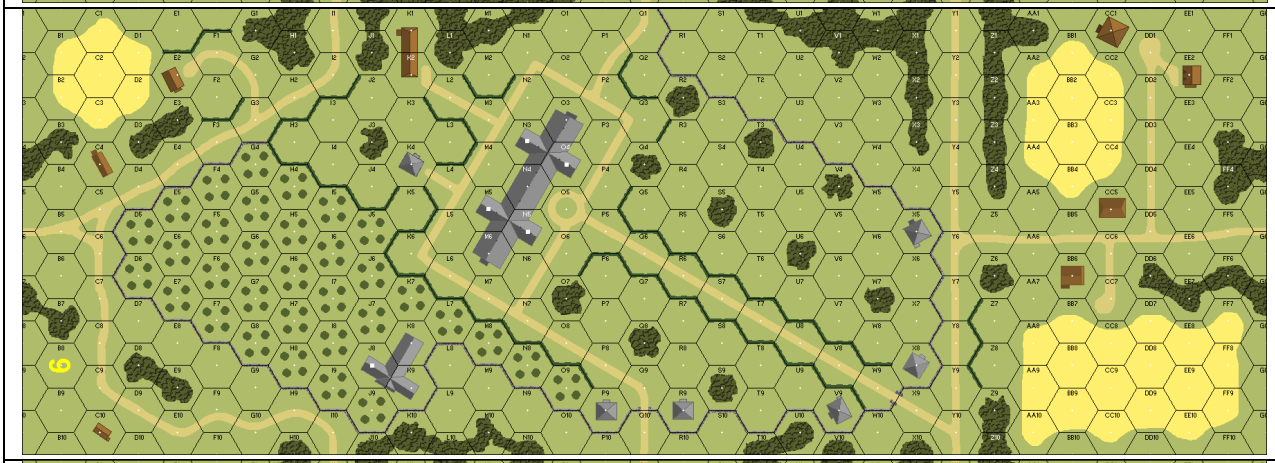
Ye Old Booke



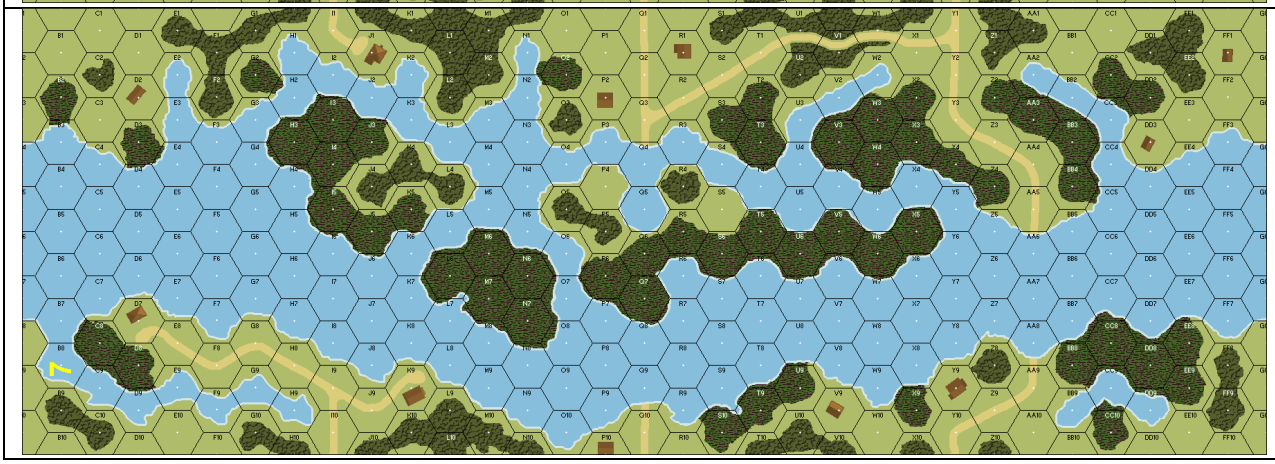
4



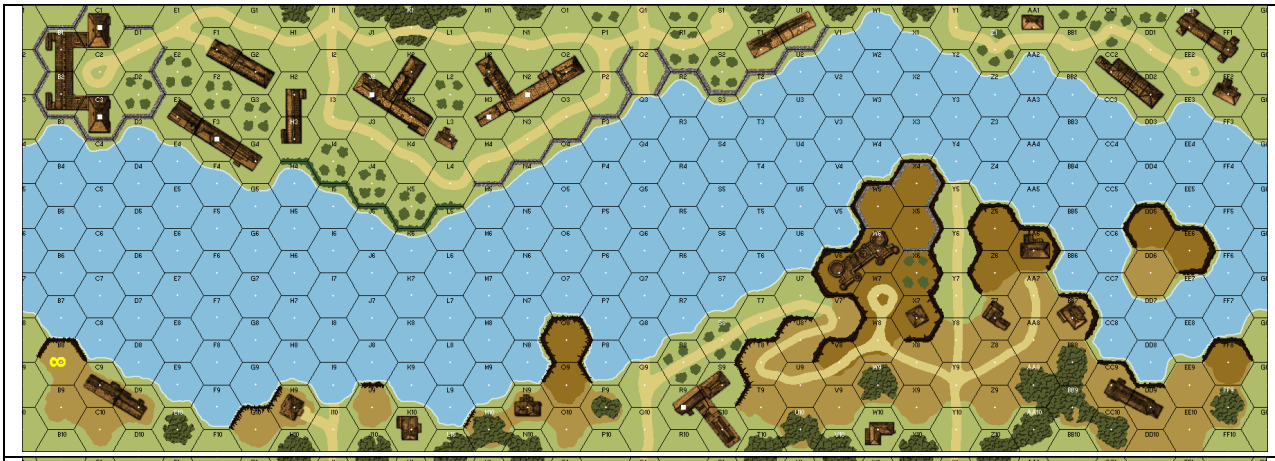
5



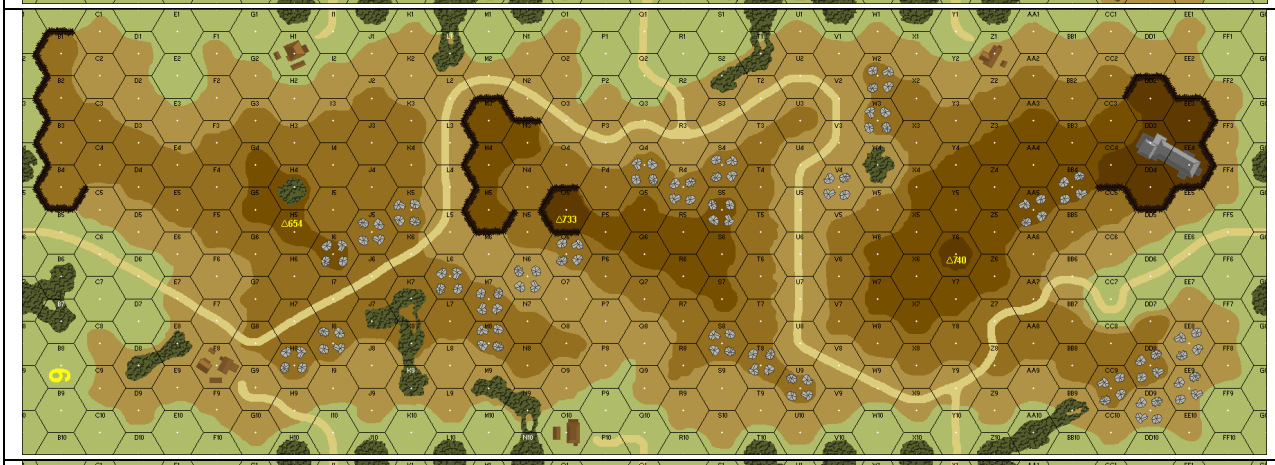
6



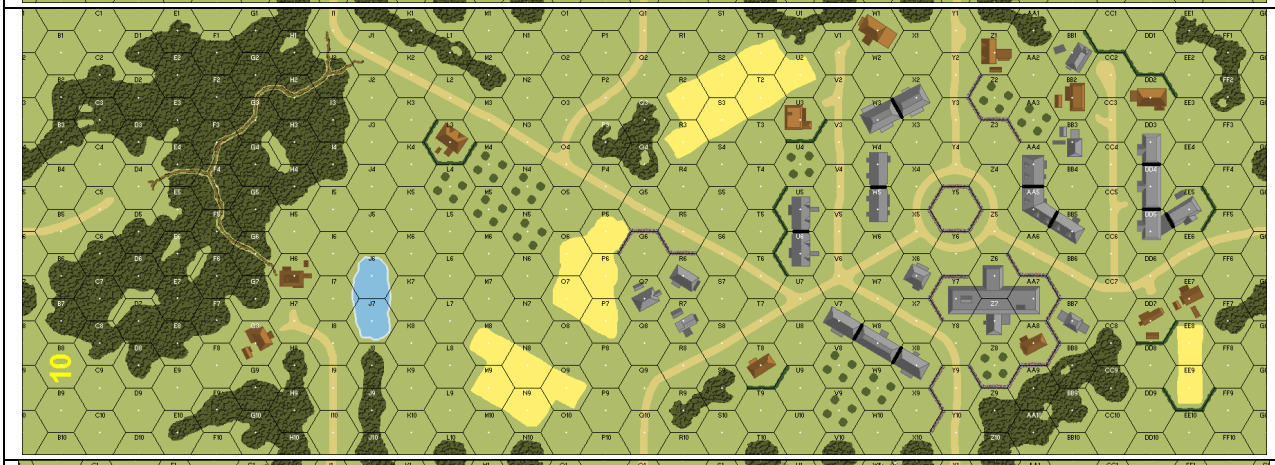
7



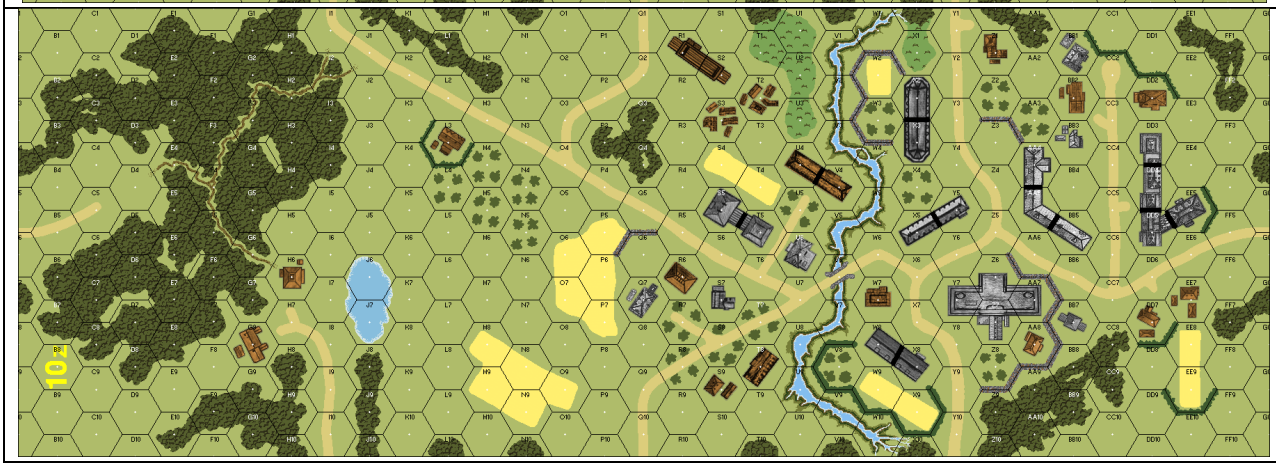
8



9

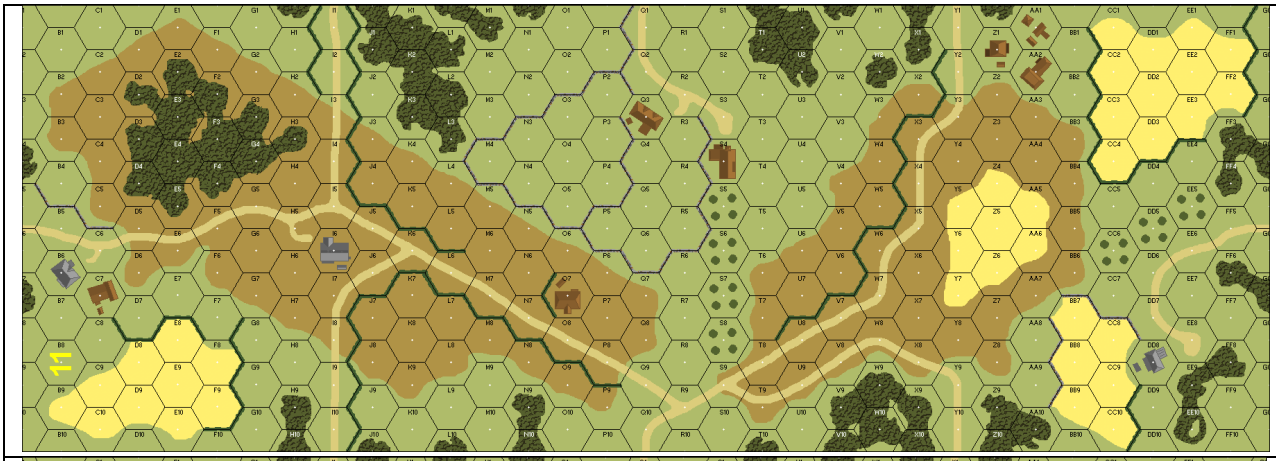


10

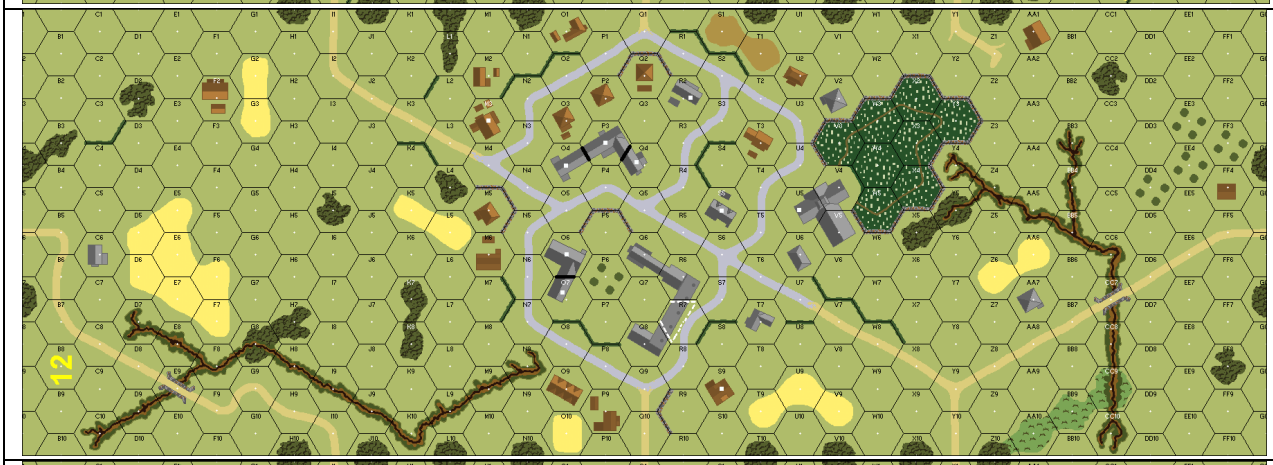


10z

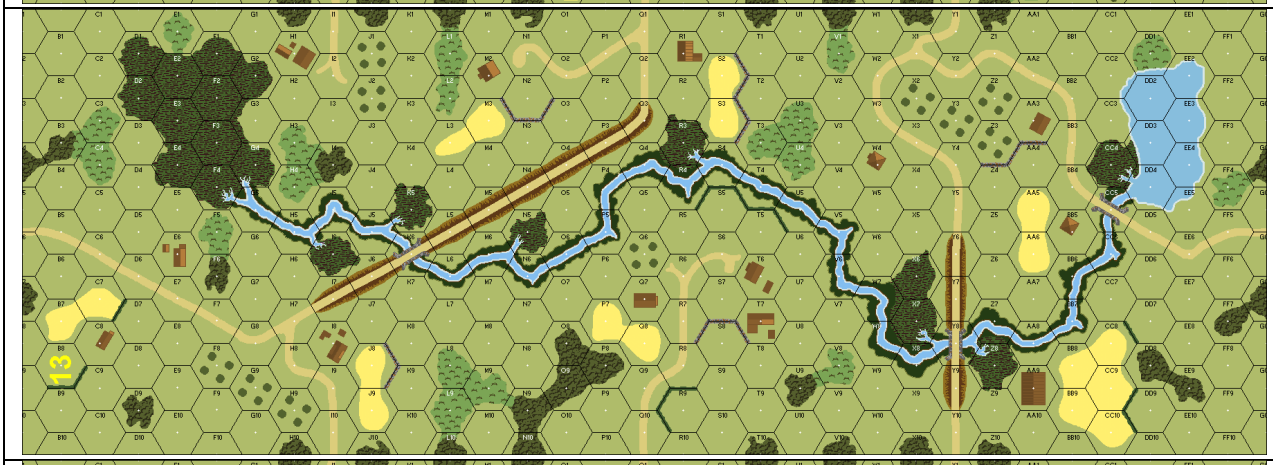
Ye Old Booke



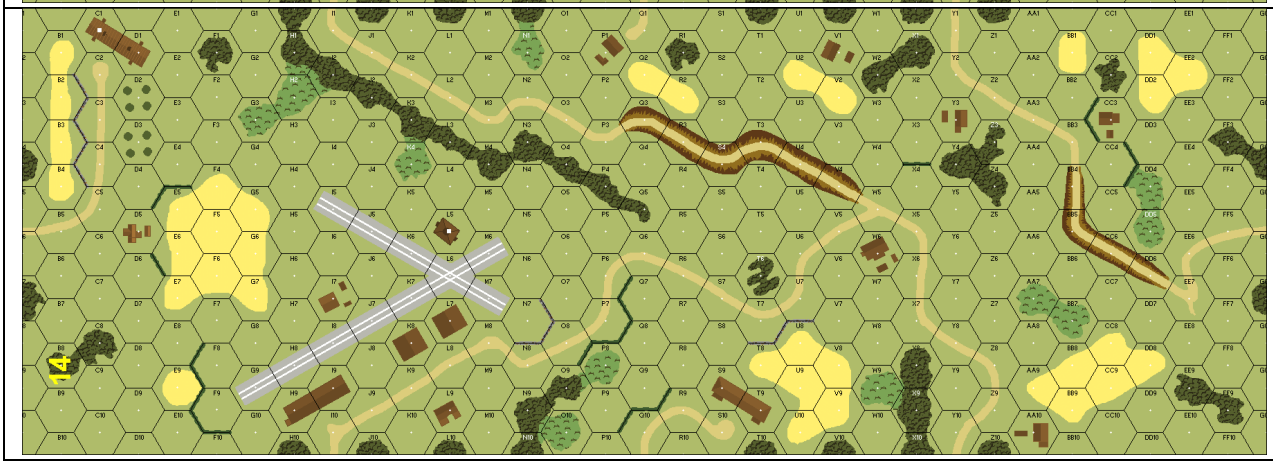
11



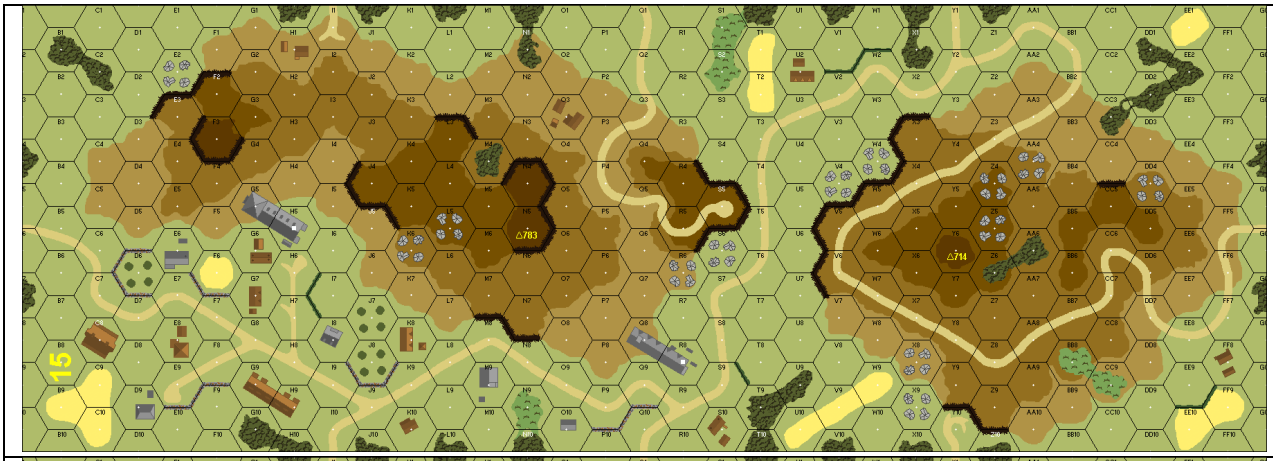
12



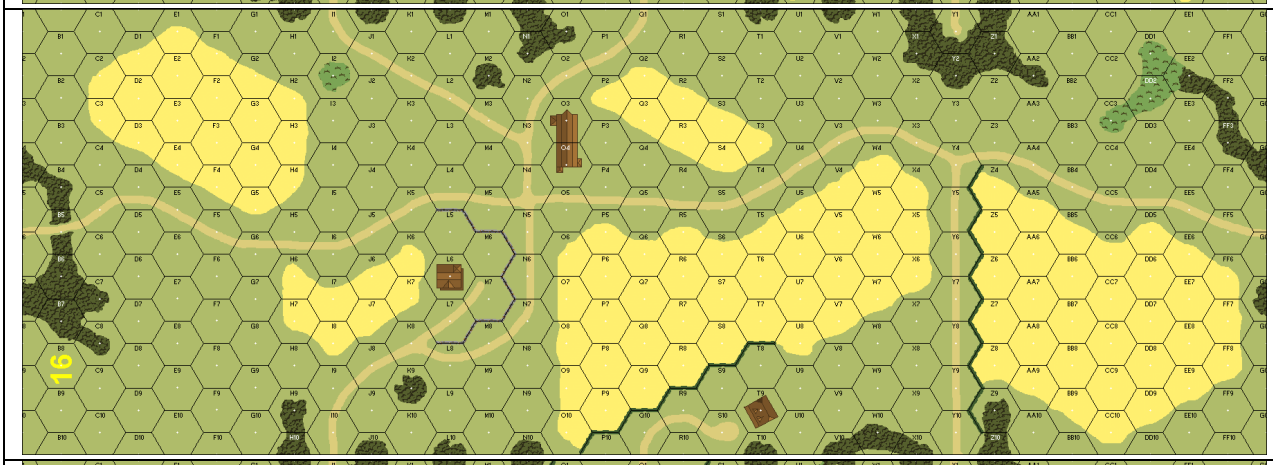
13



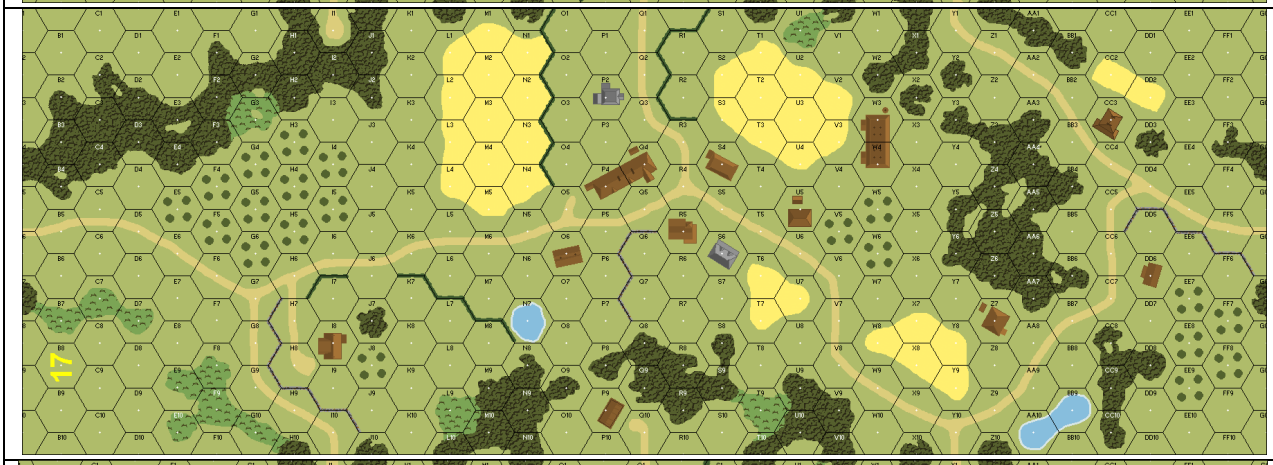
14



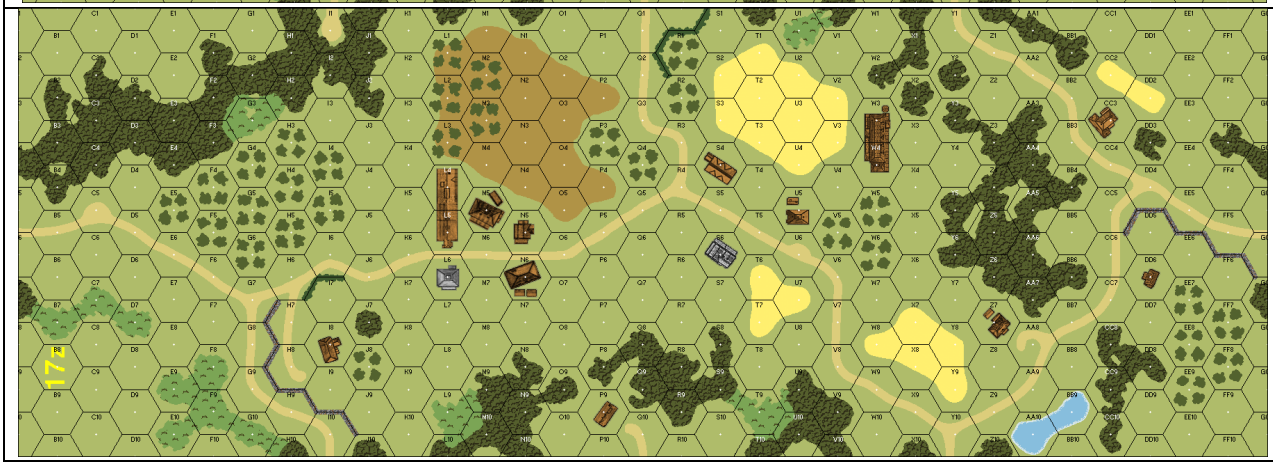
15



16

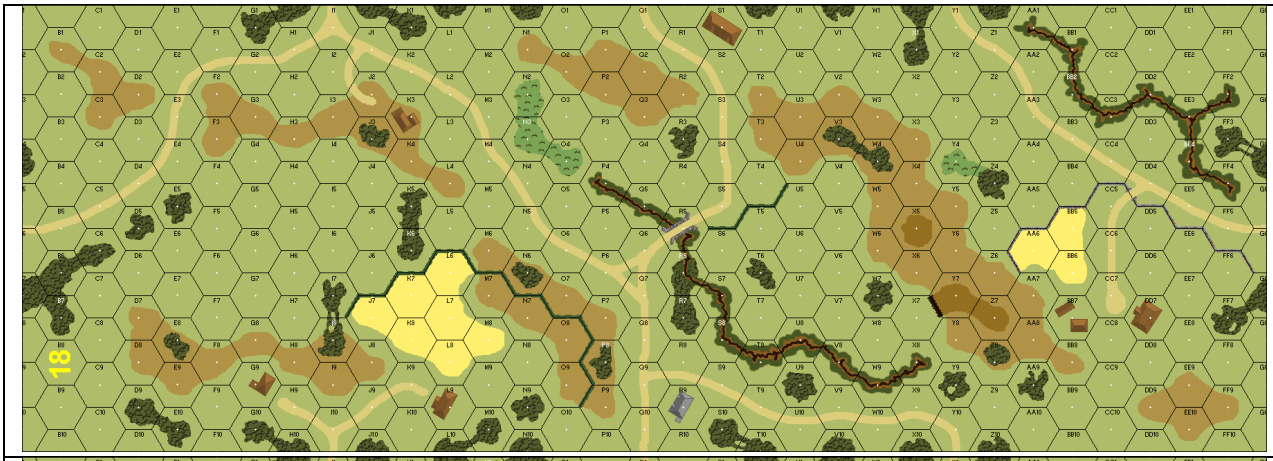


17

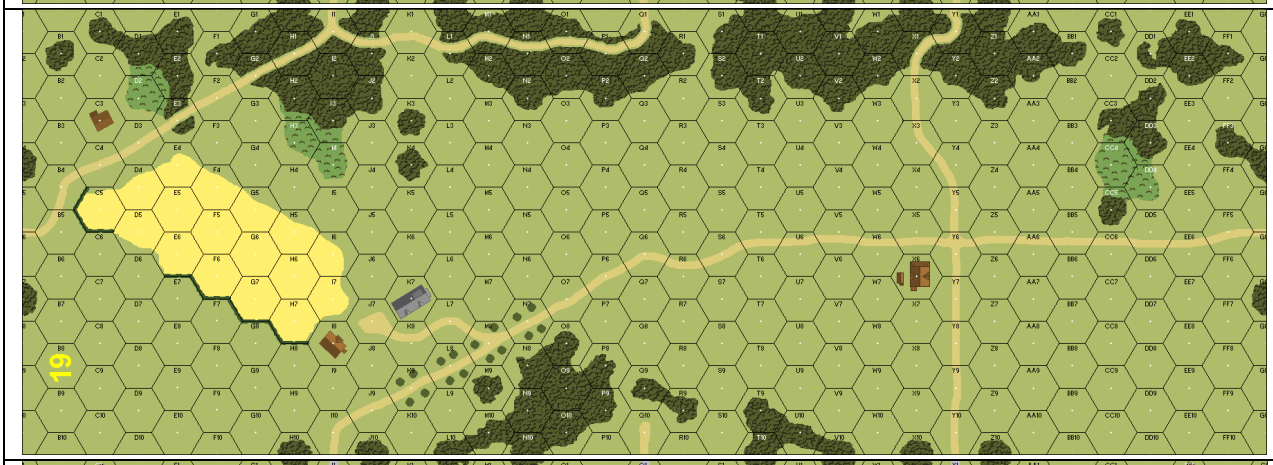


17z

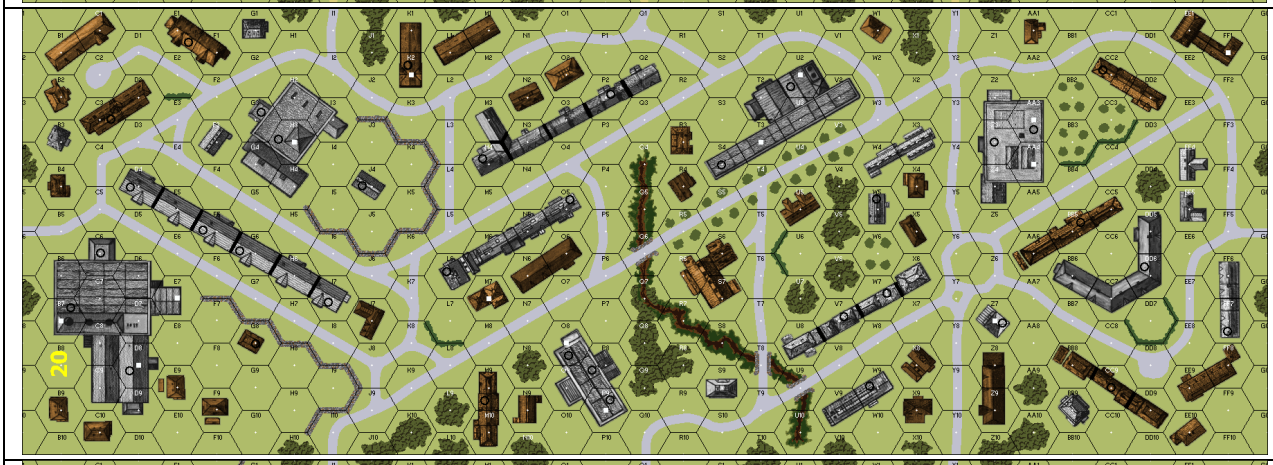
Ye Old Booke



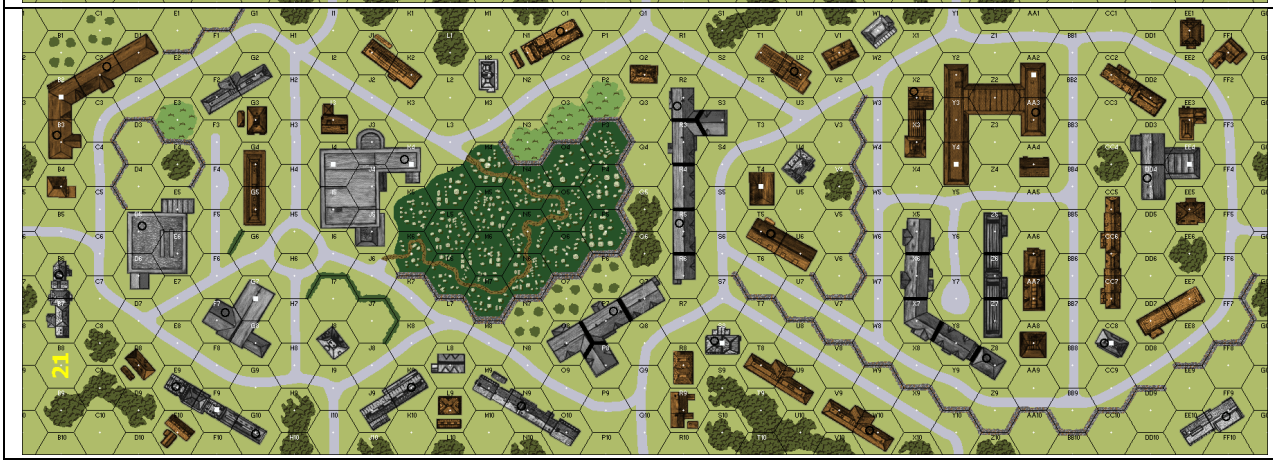
18



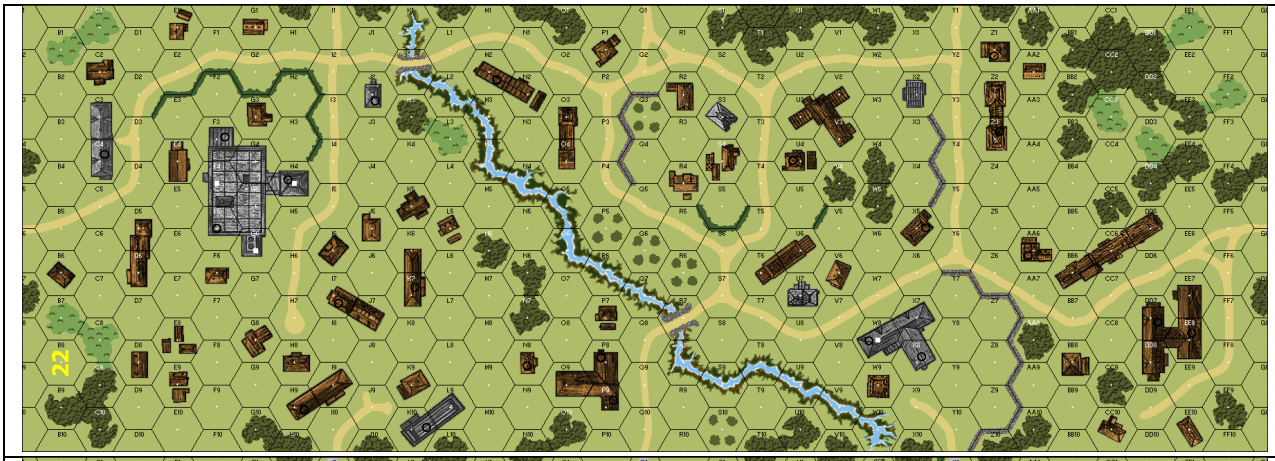
19



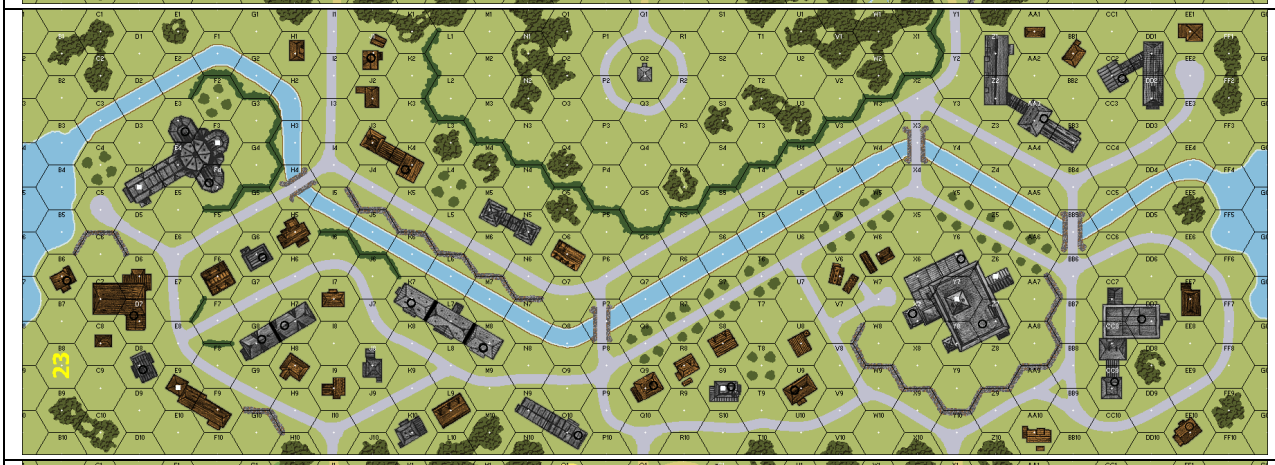
20



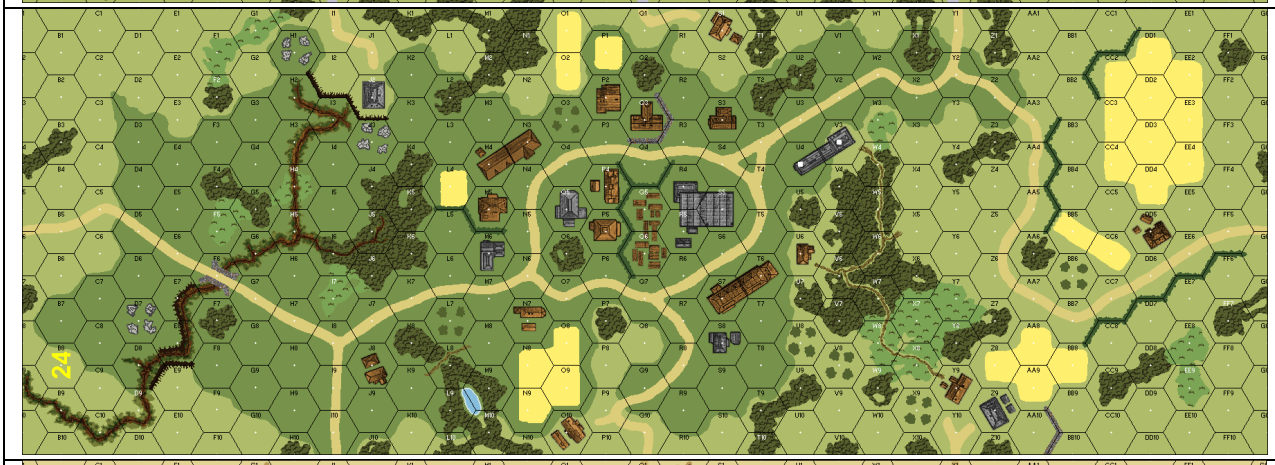
21



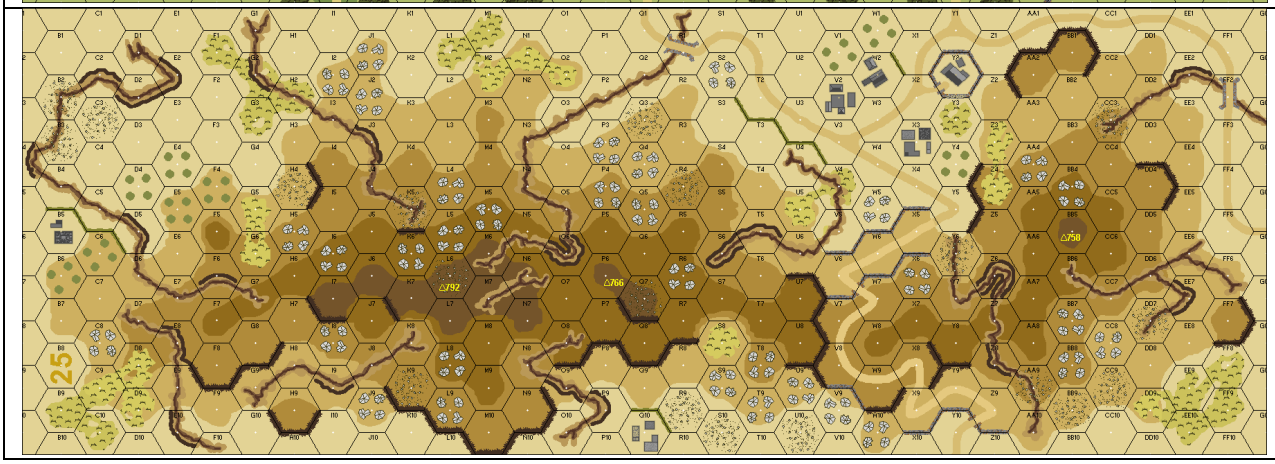
22



23

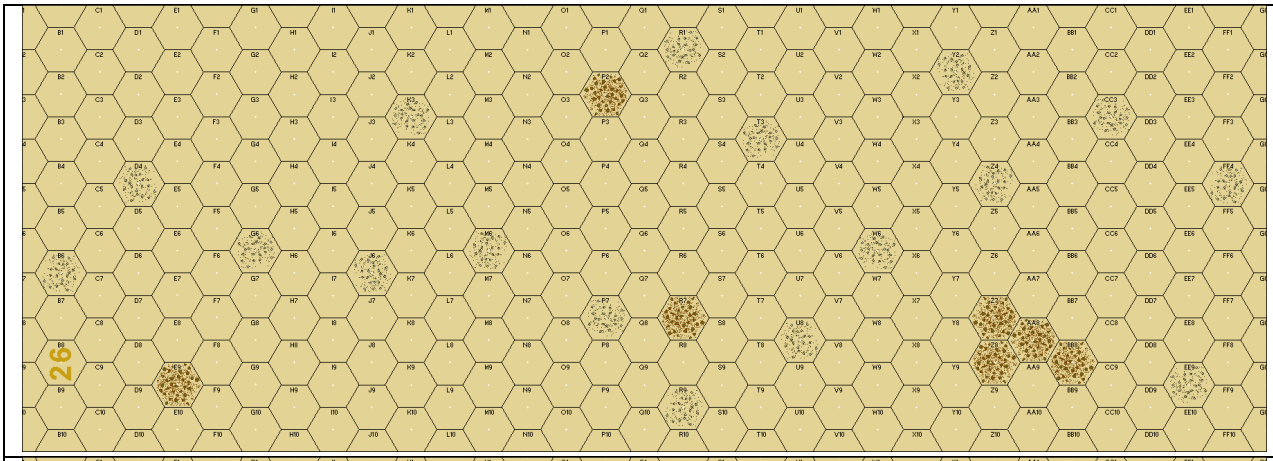


24

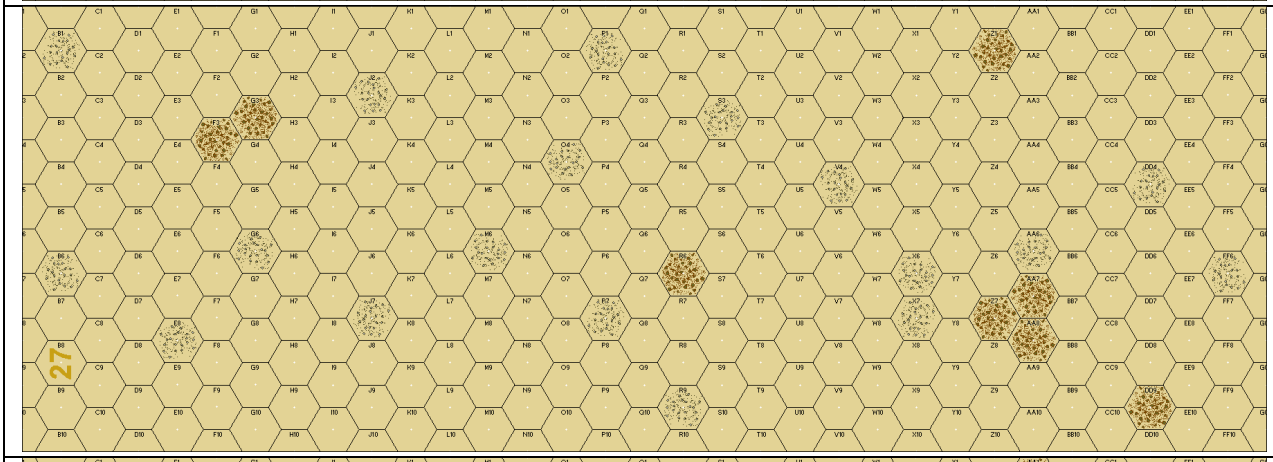


25

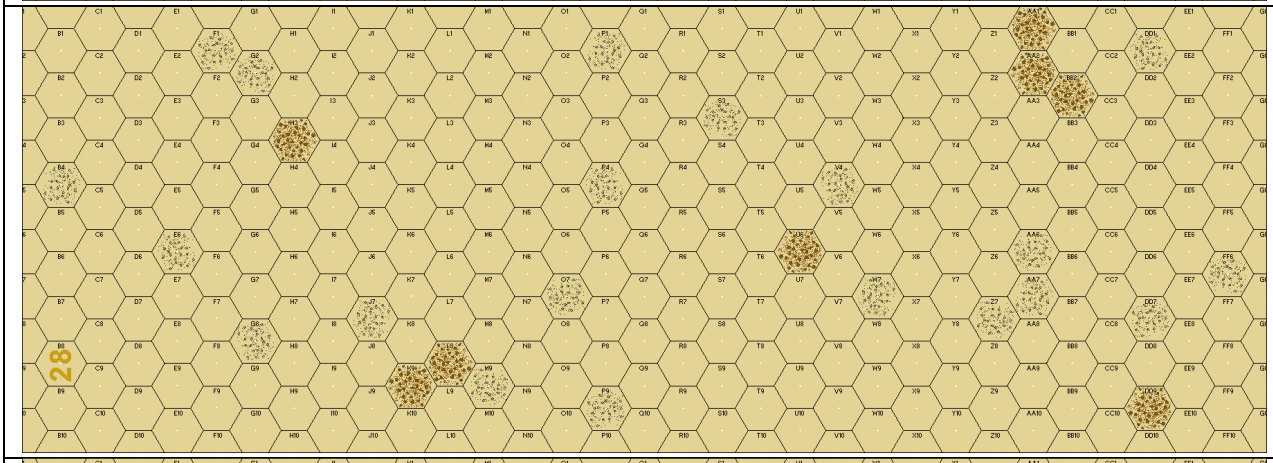
Ye Old Booke



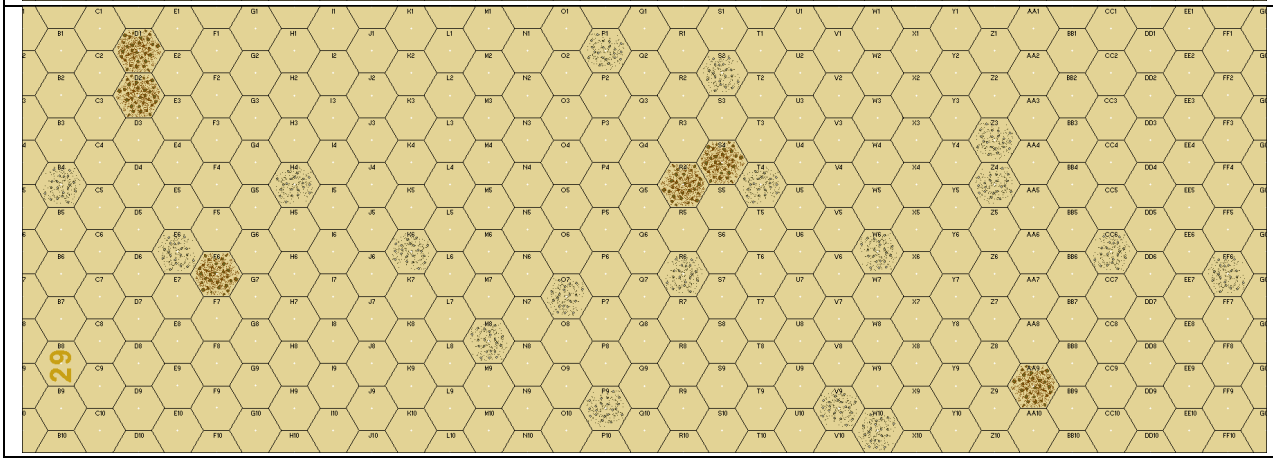
26



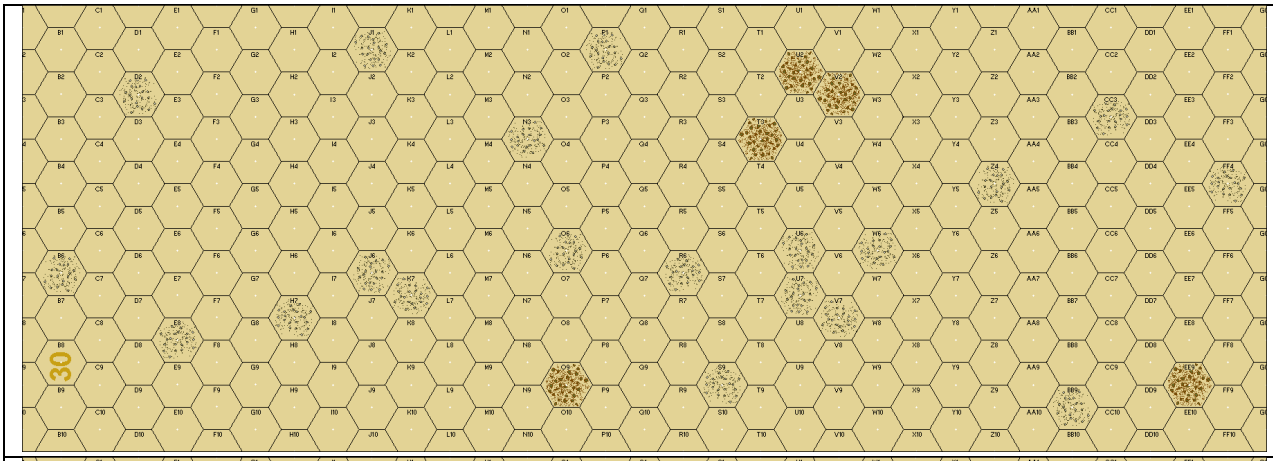
27



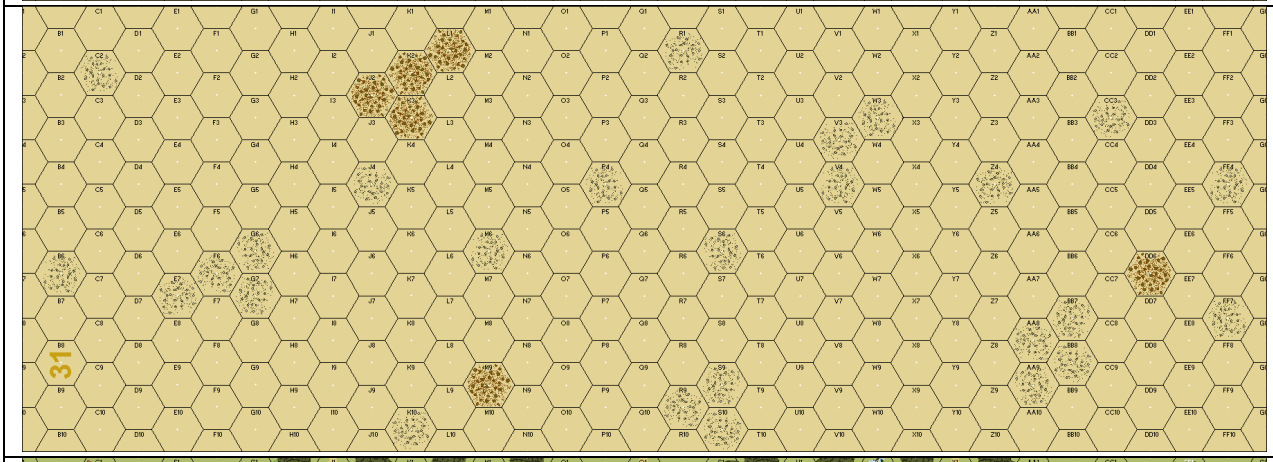
28



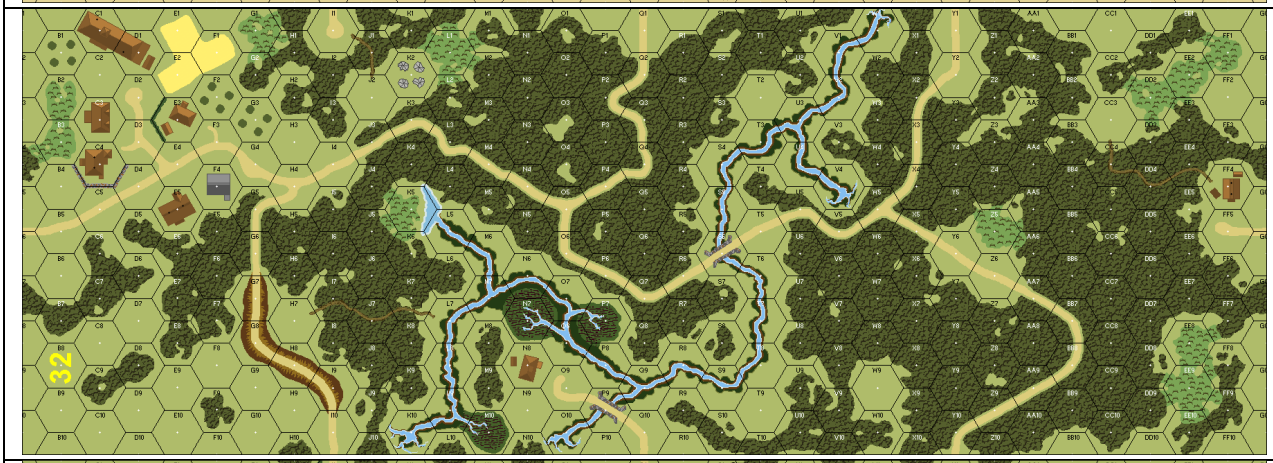
29



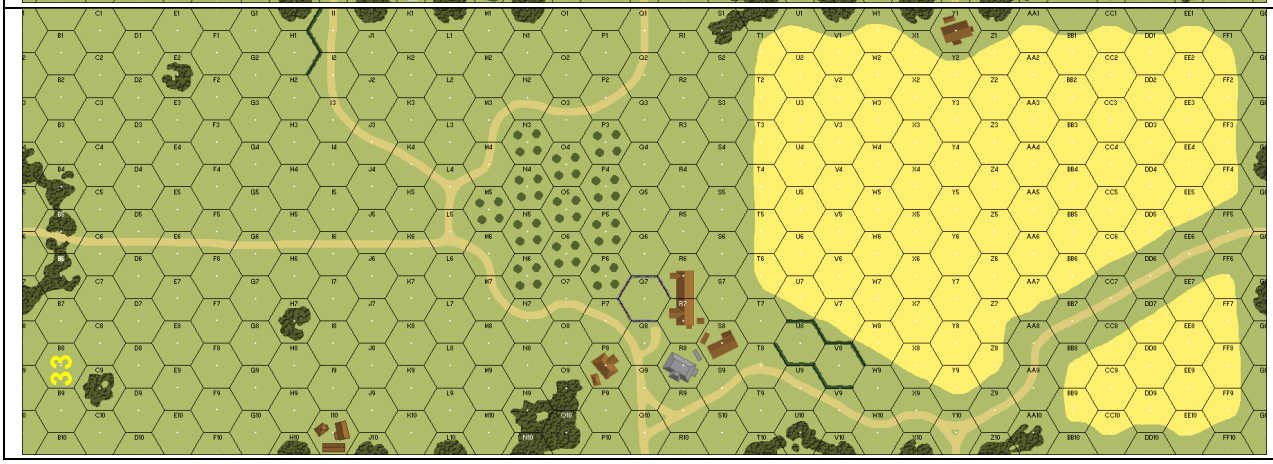
30



31

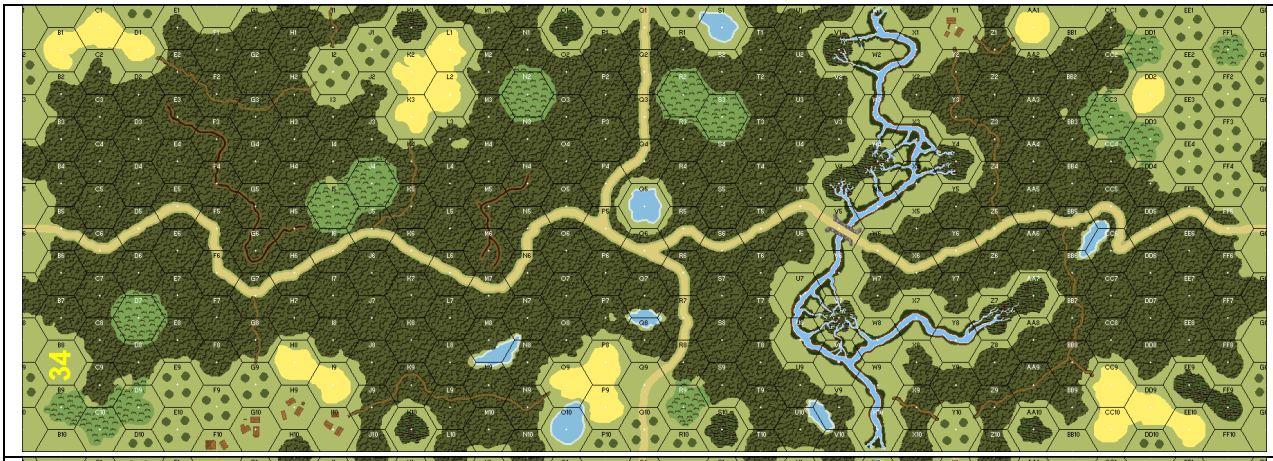


32

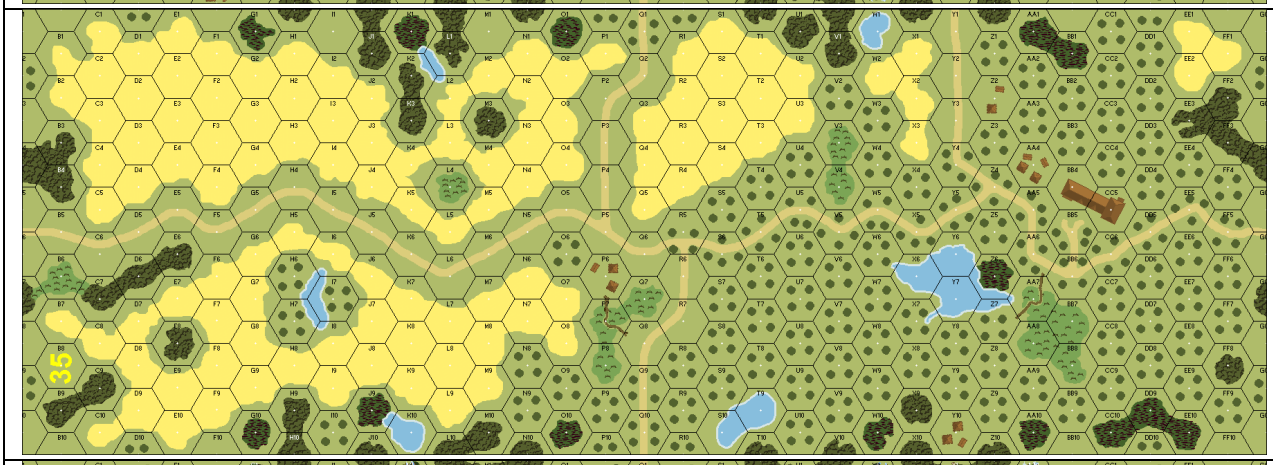


33

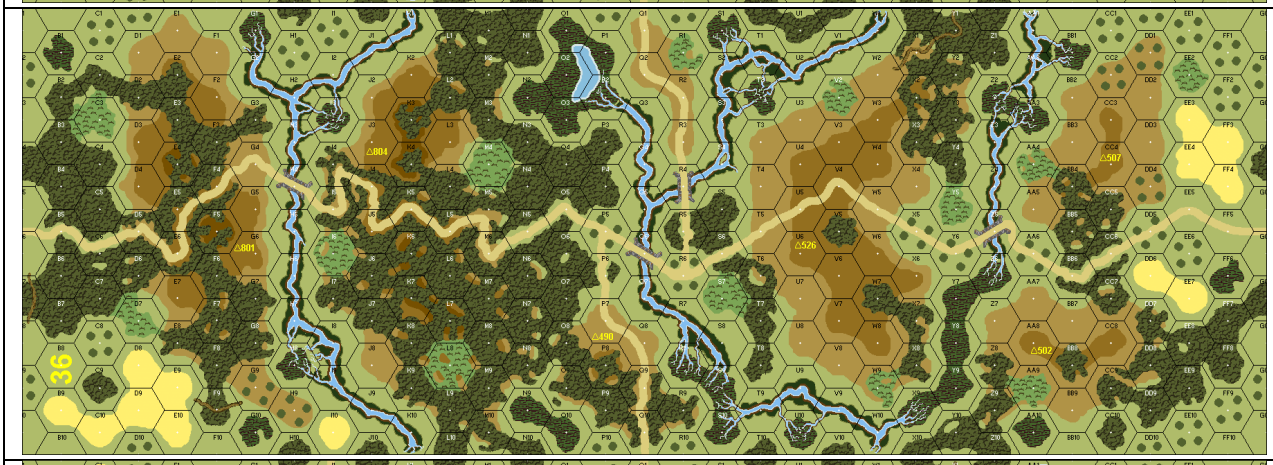
Ye Old Booke



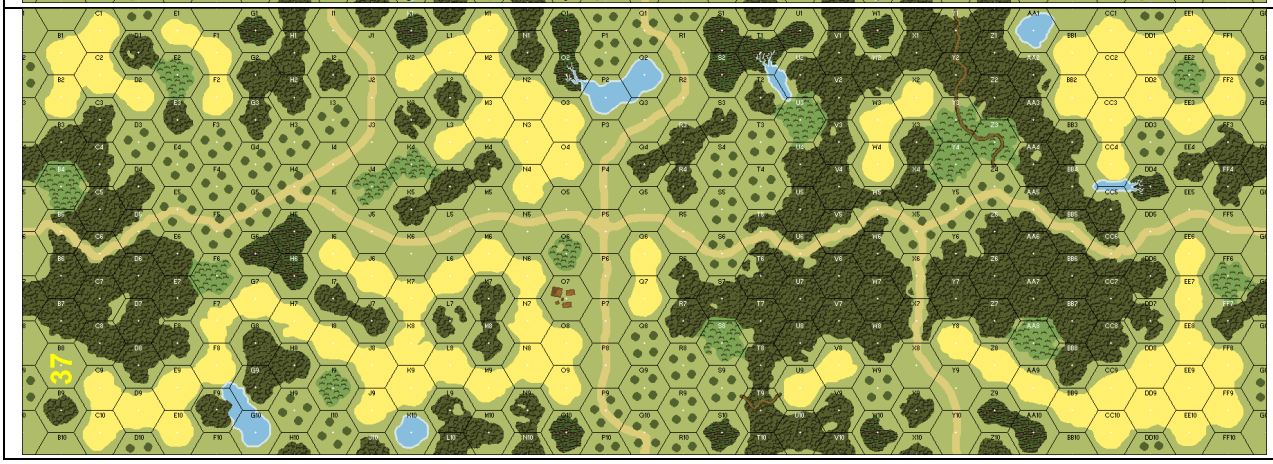
34



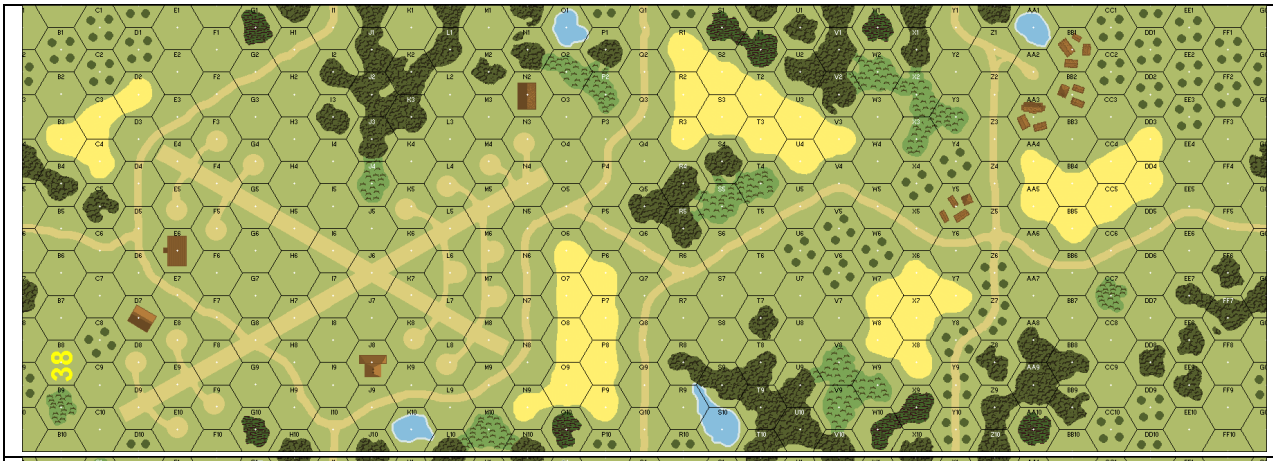
35



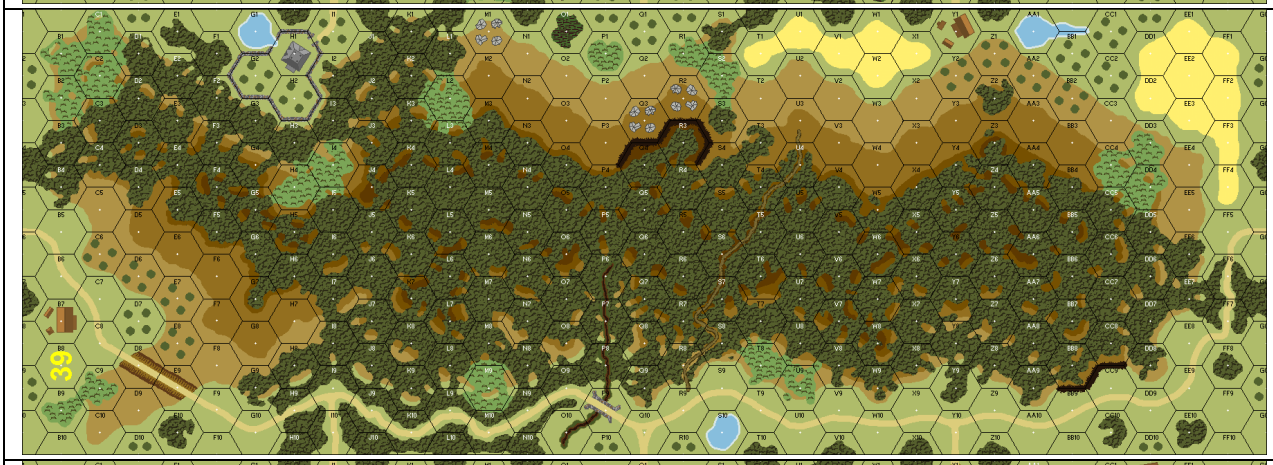
36



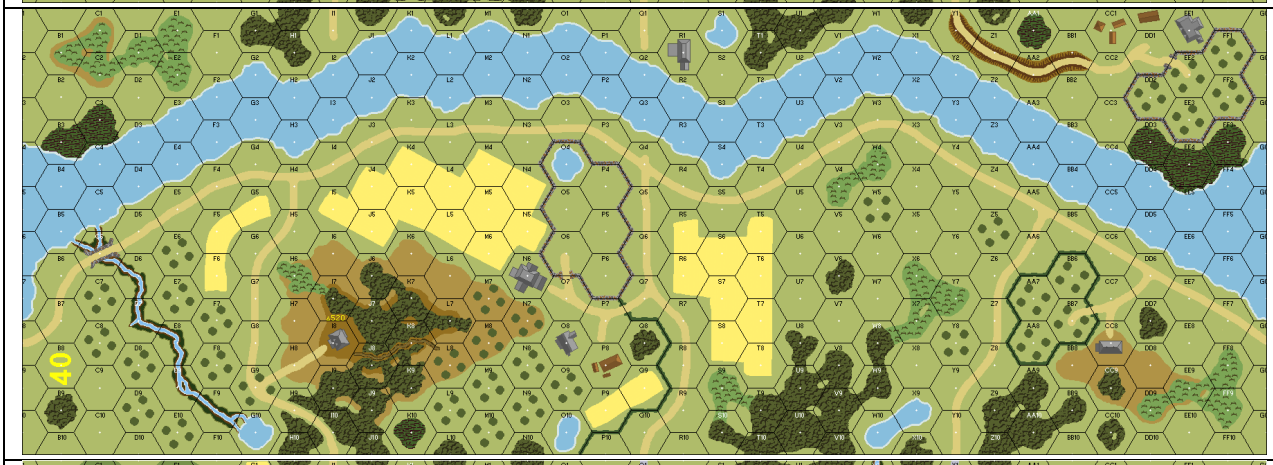
37



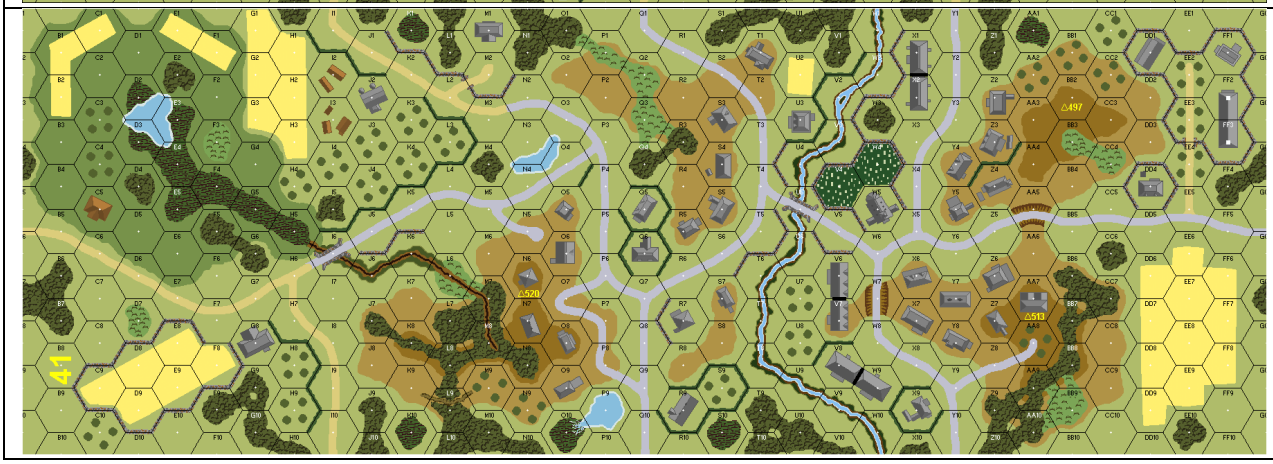
38



39

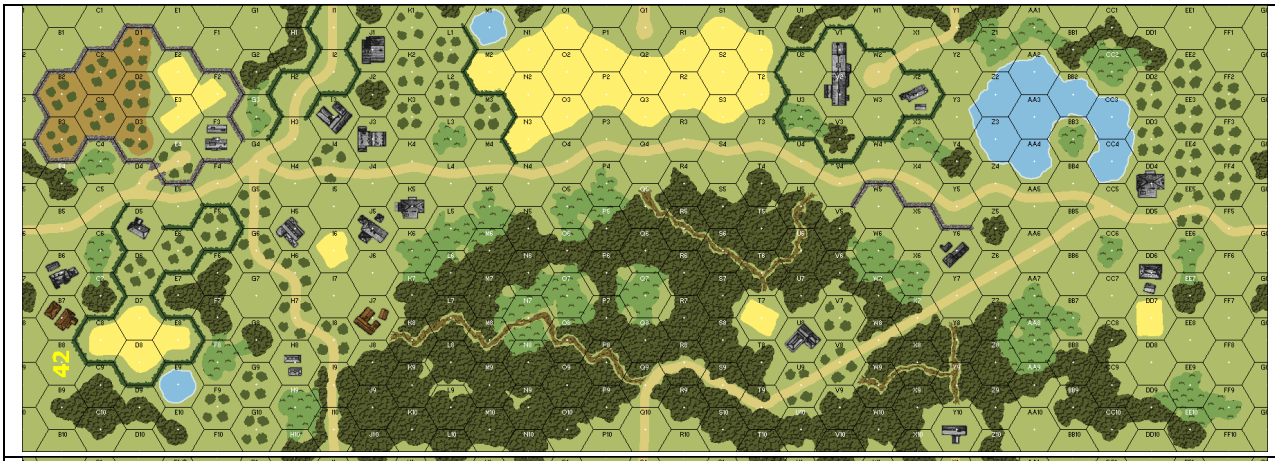


40

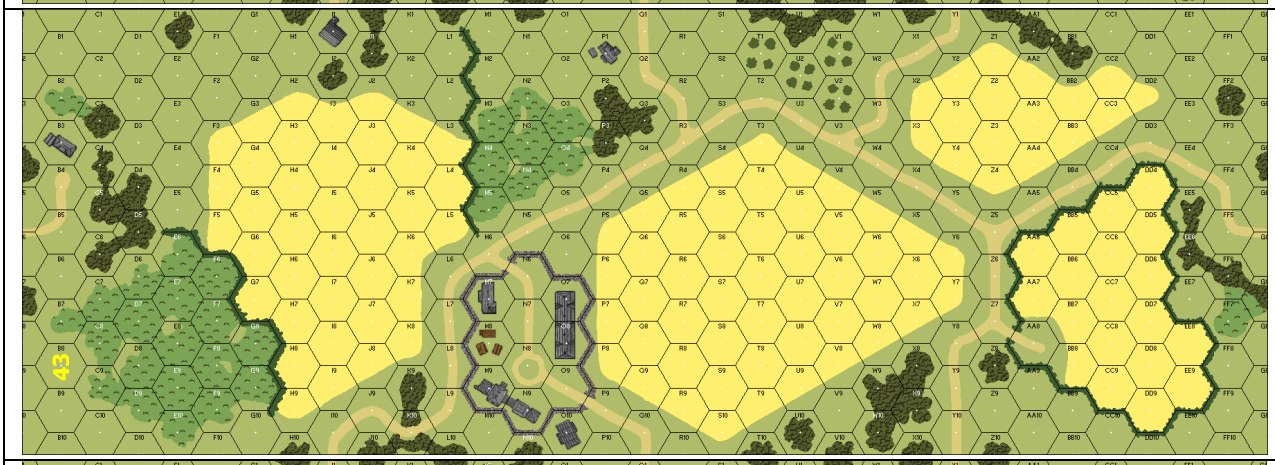


41

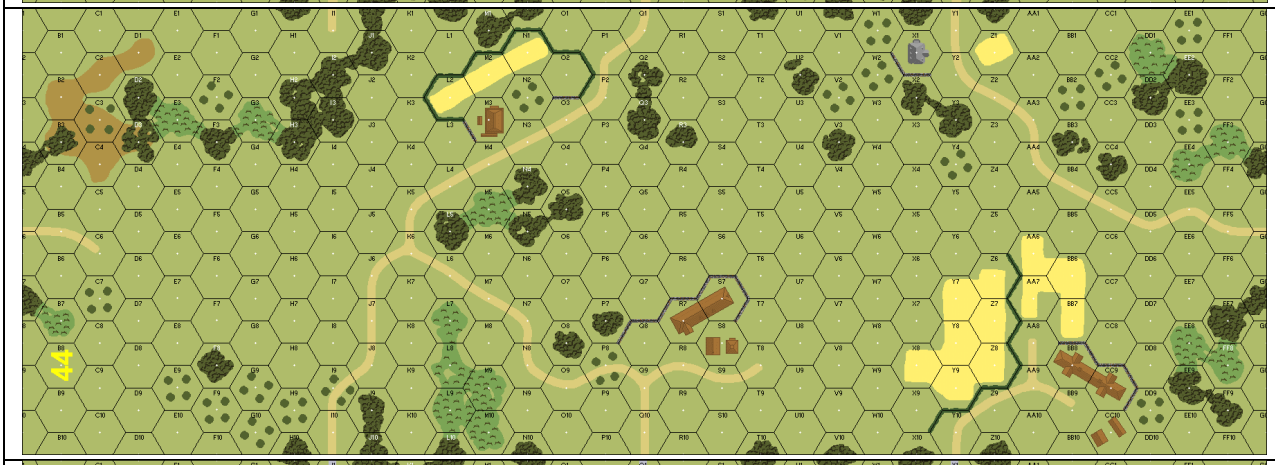
Ye Old Booke



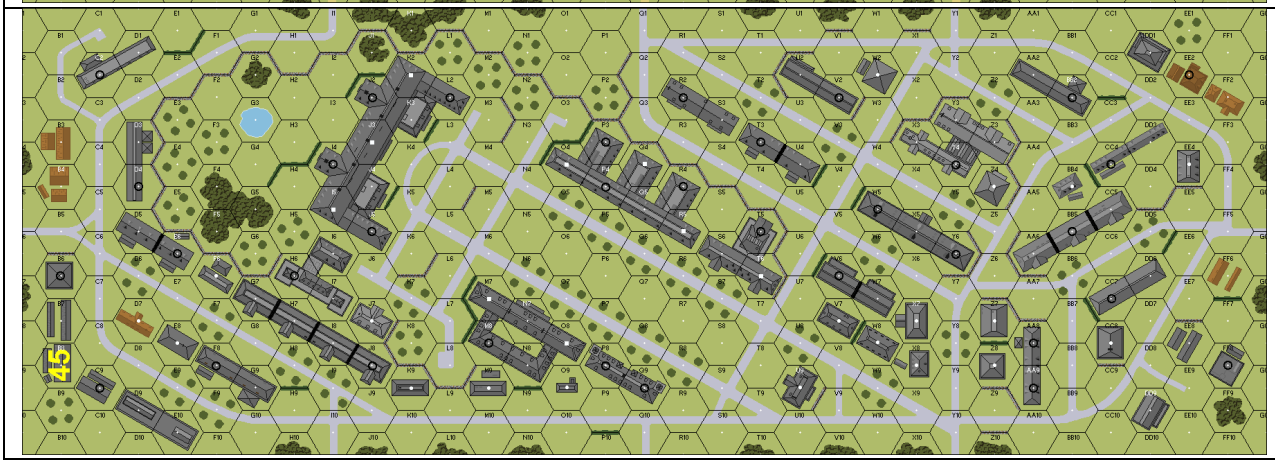
42



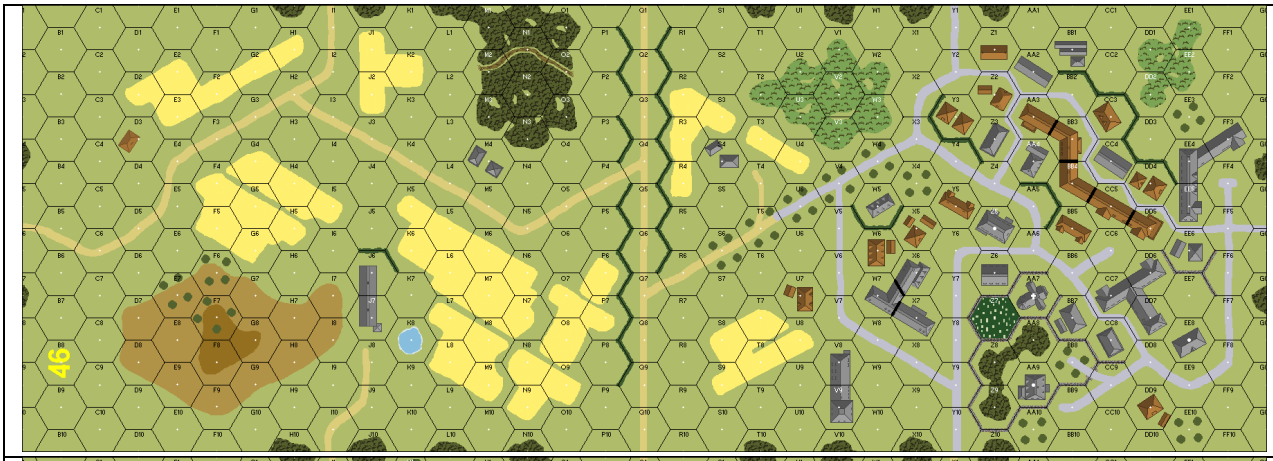
43



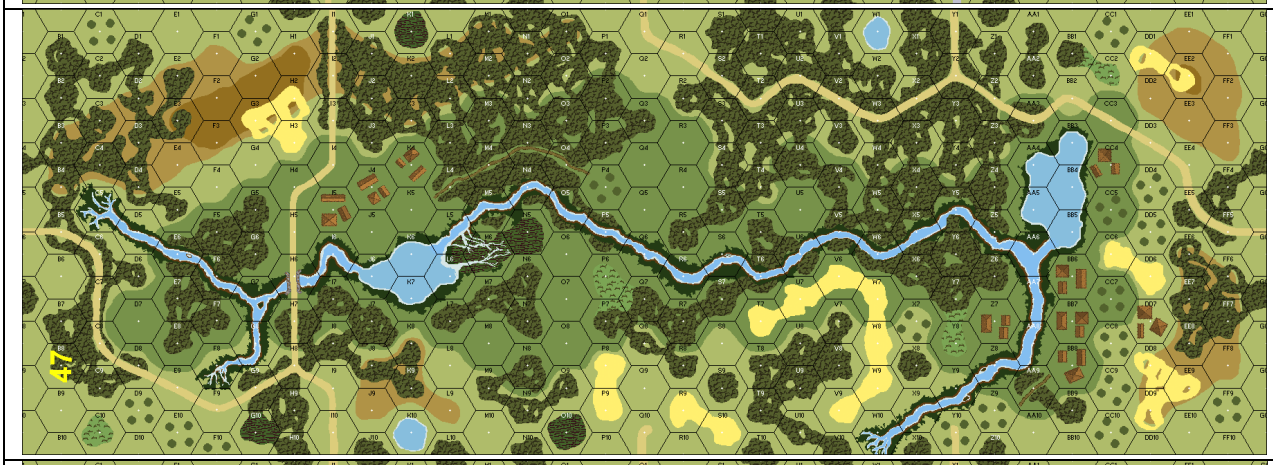
44



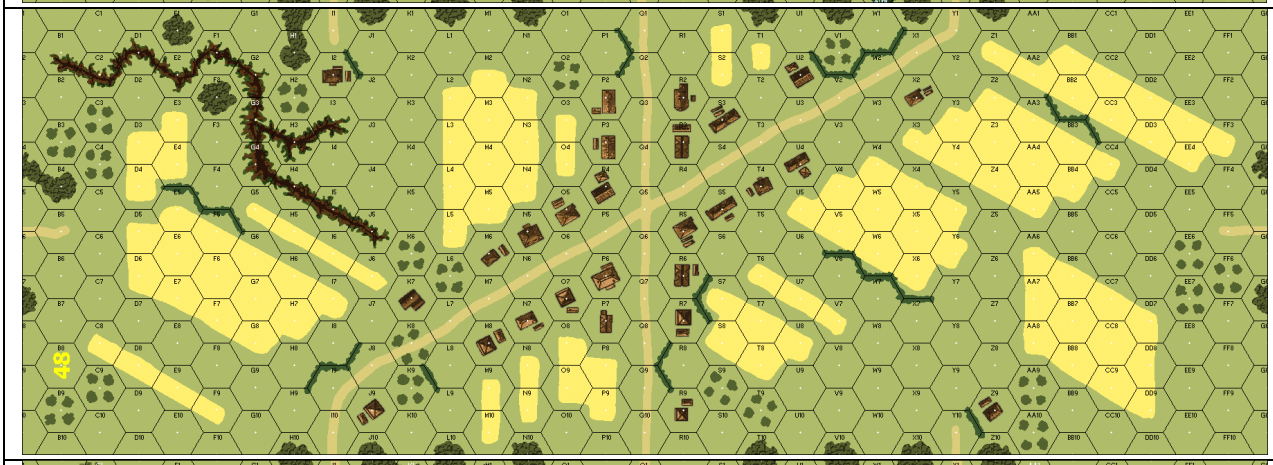
45



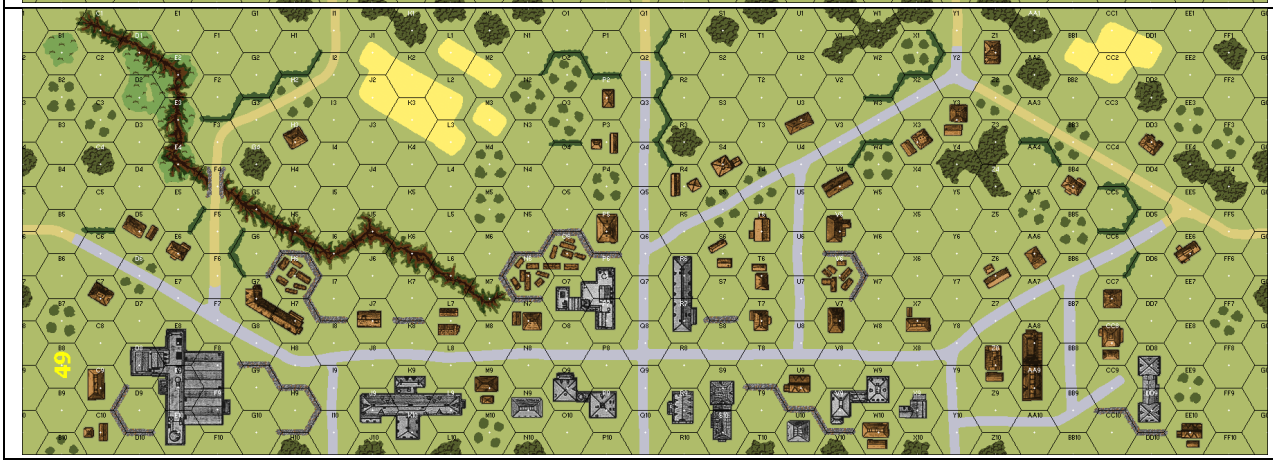
46



47

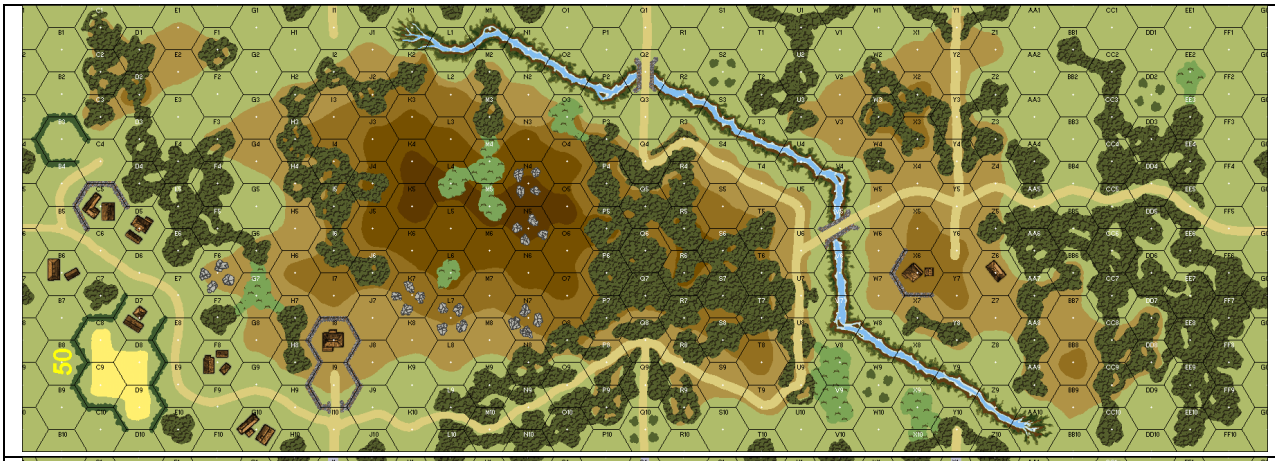


48



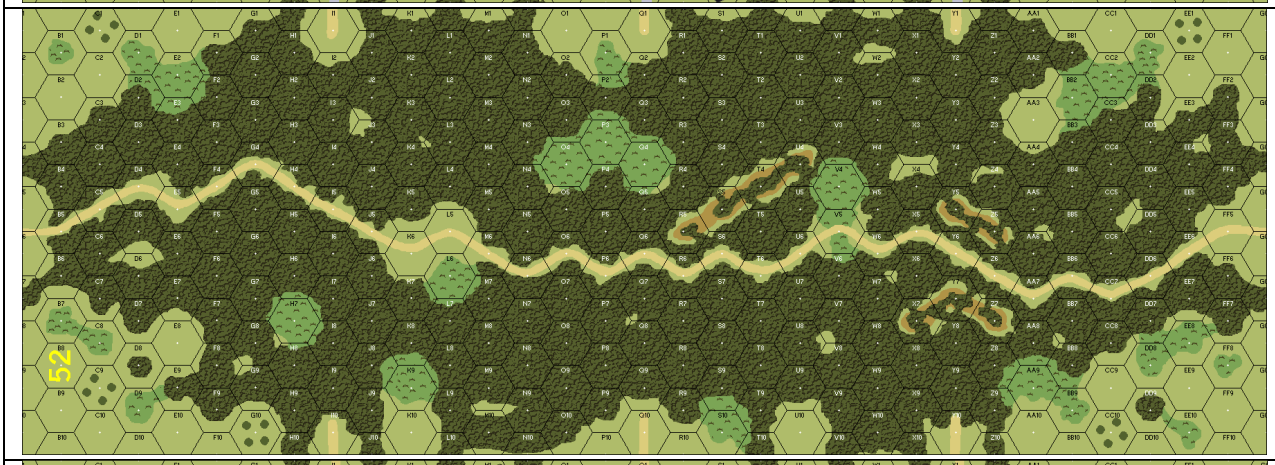
49

Ye Old Booke

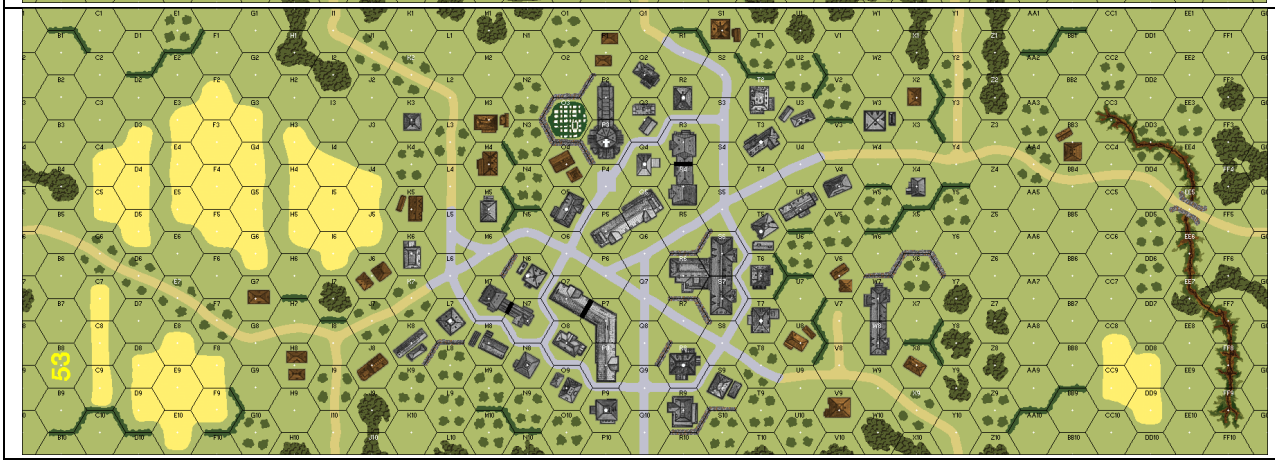


50

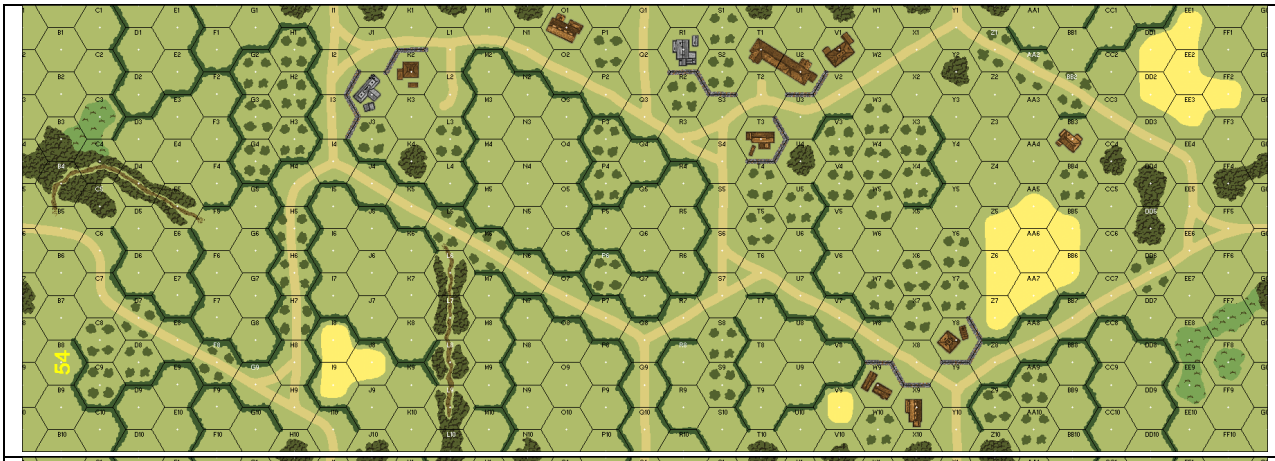
51



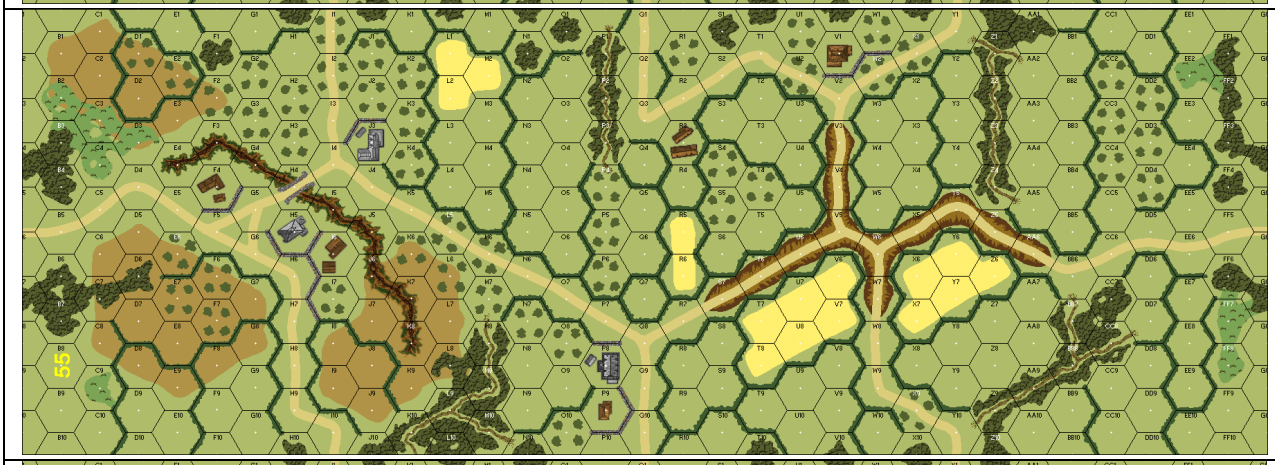
52



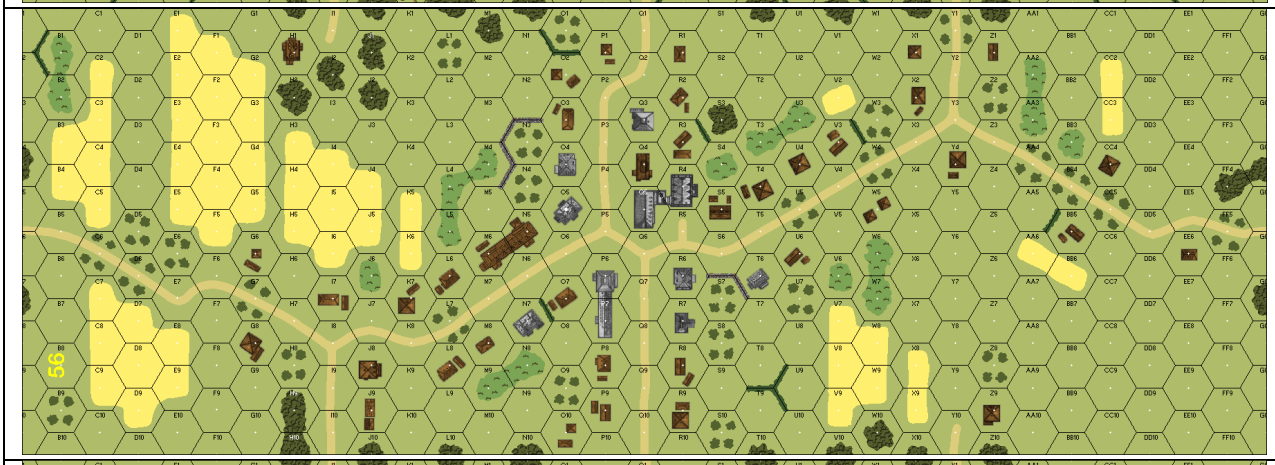
53



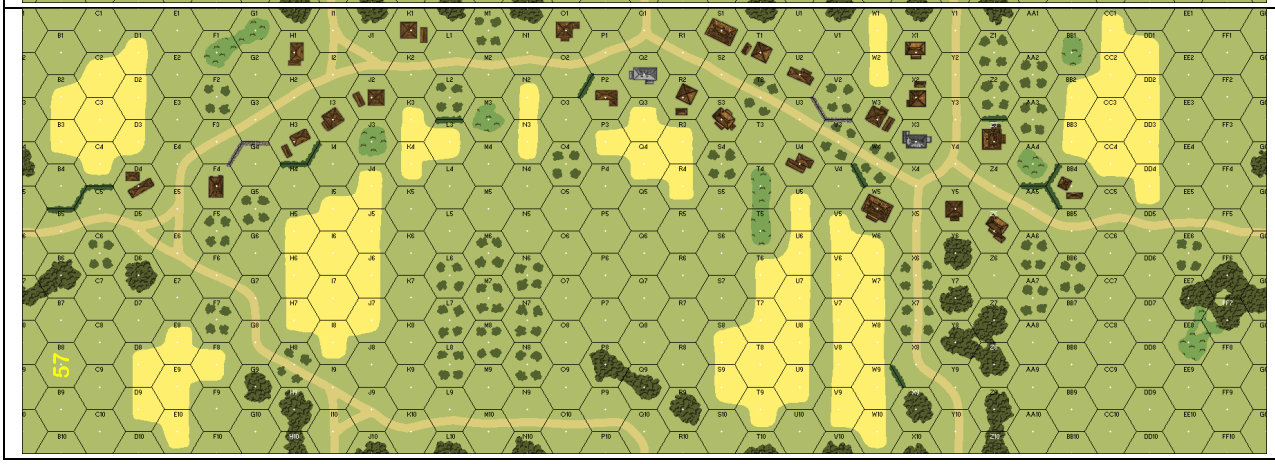
54



55

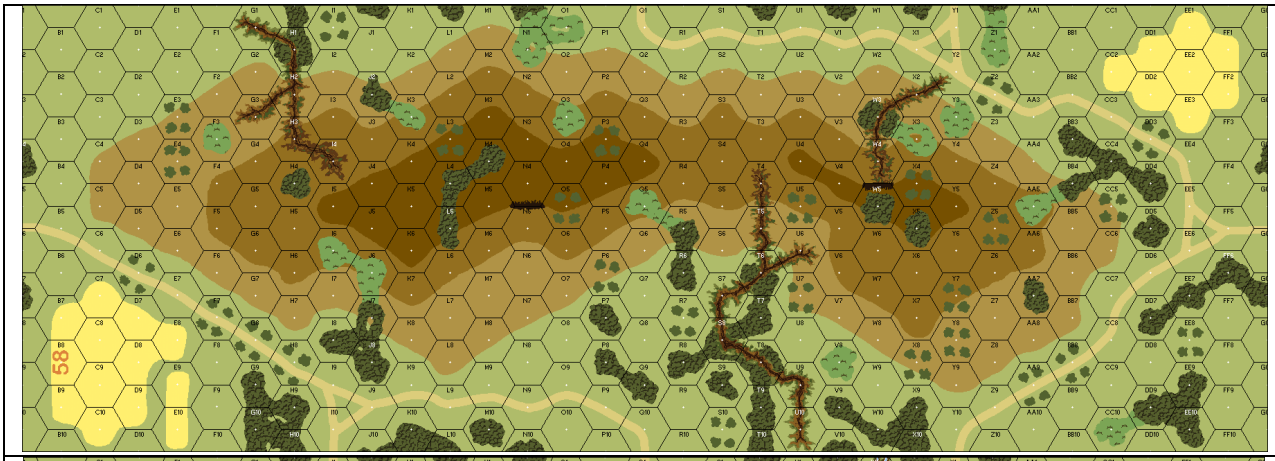


56

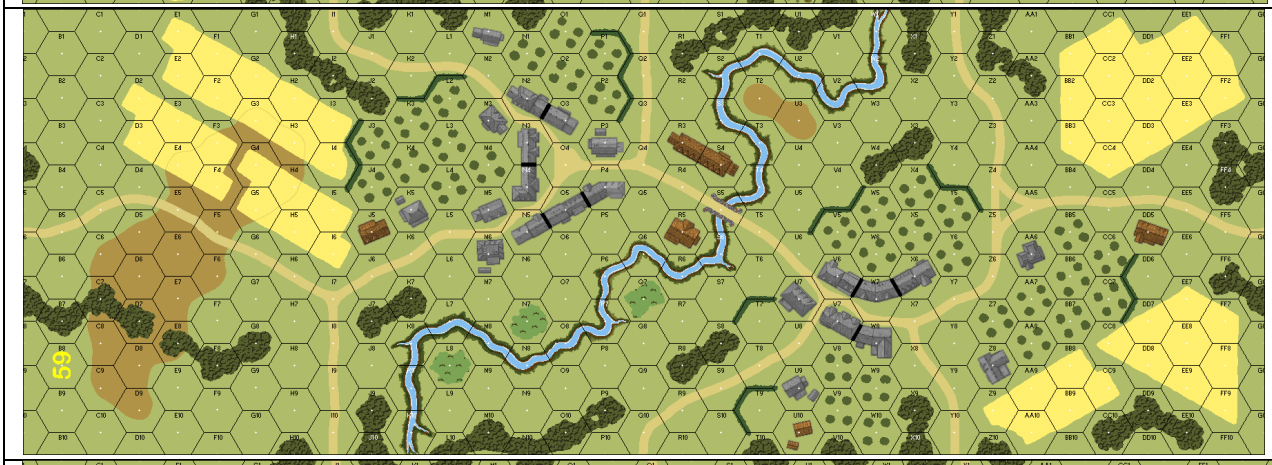


57

Ye Old Booke

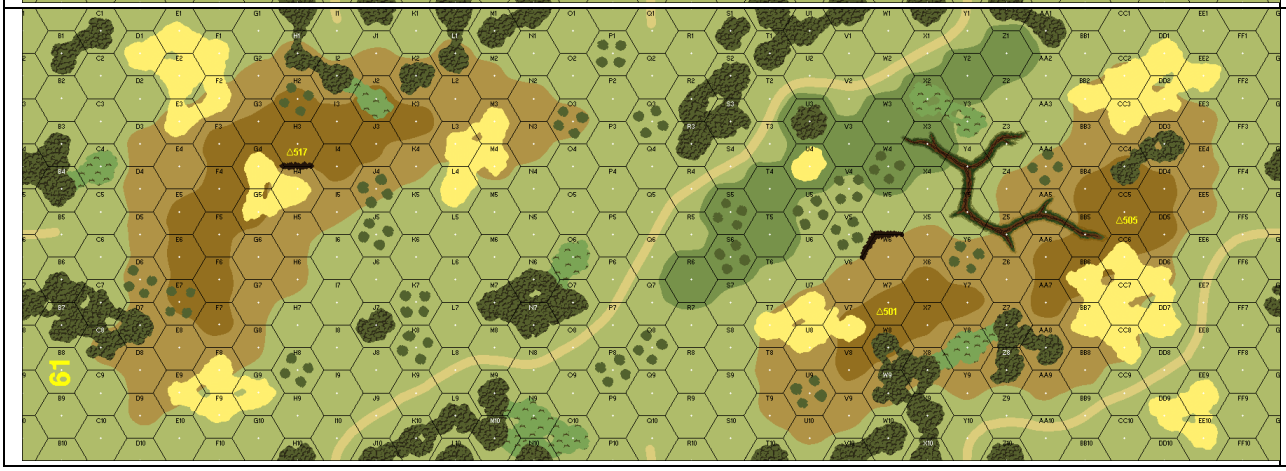


58

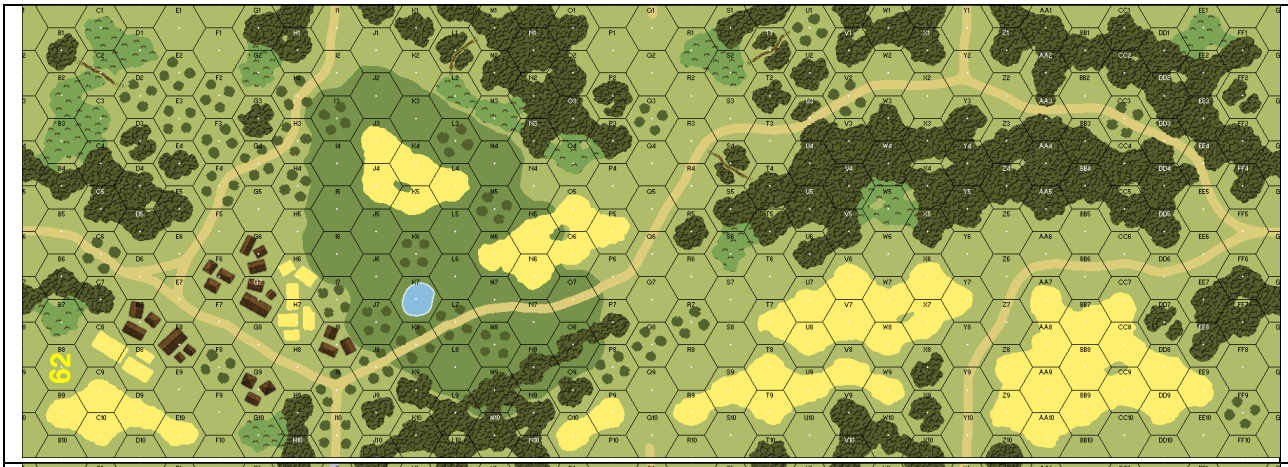


59

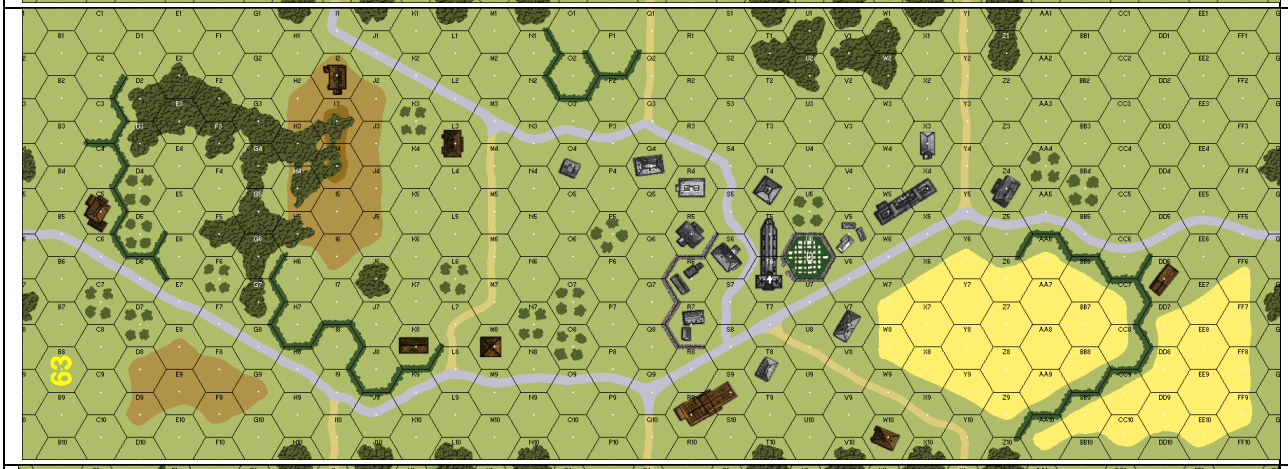
60



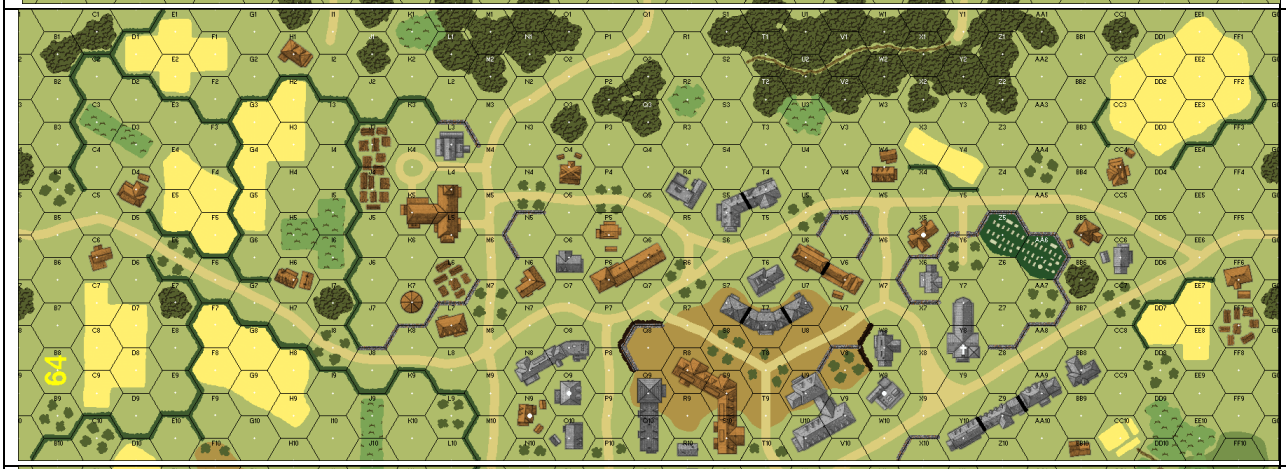
61



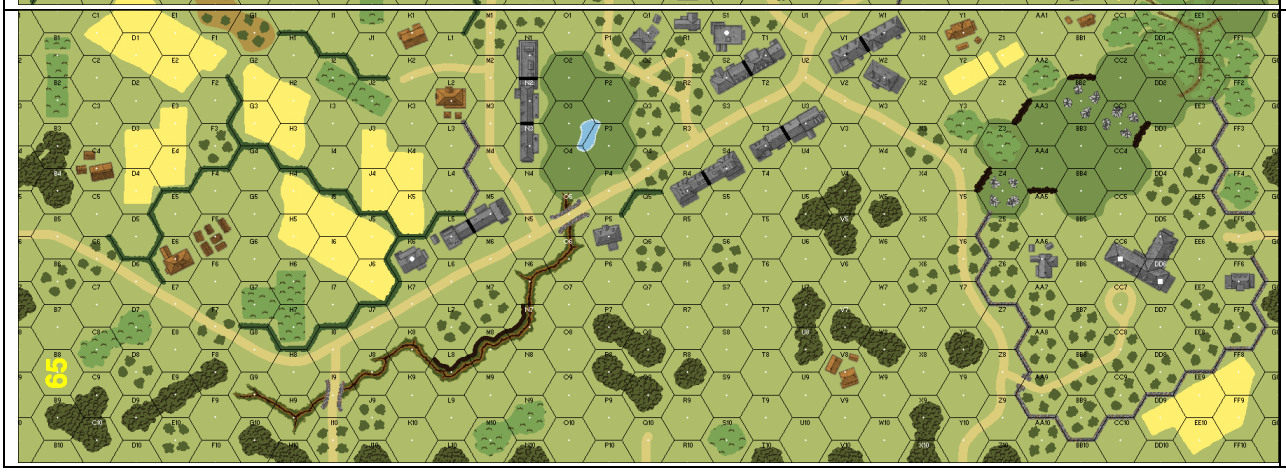
62



63

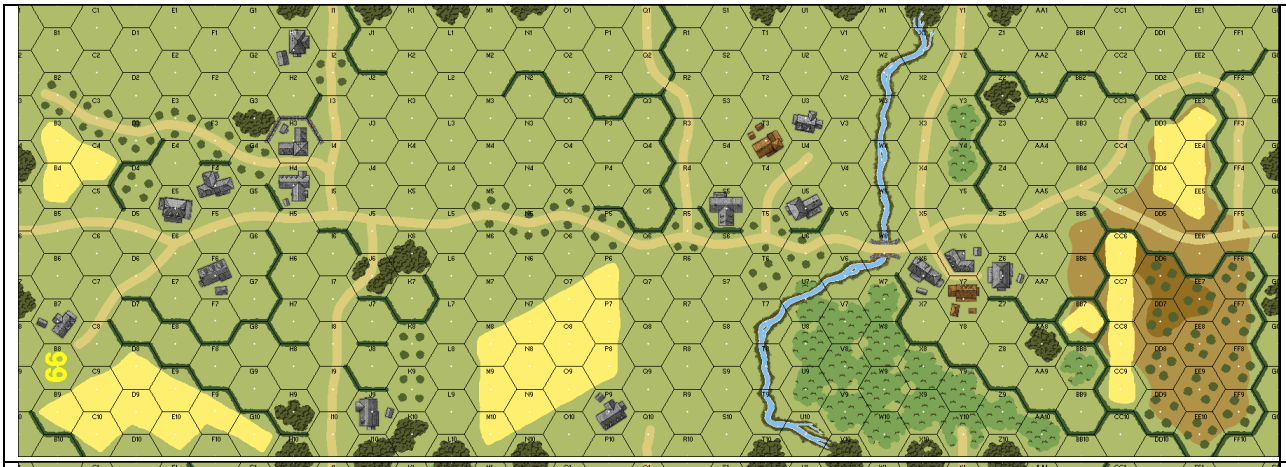


64

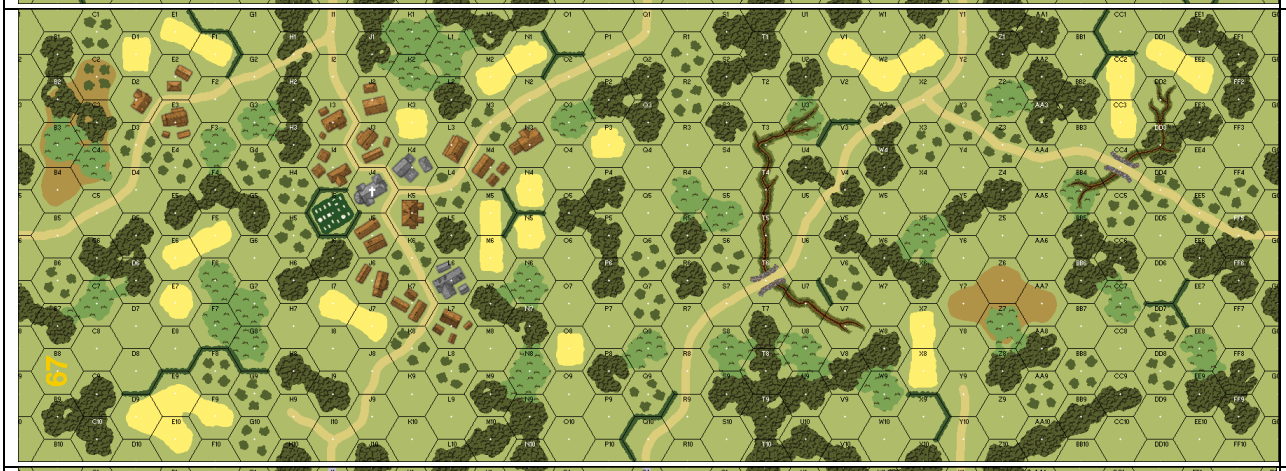


65

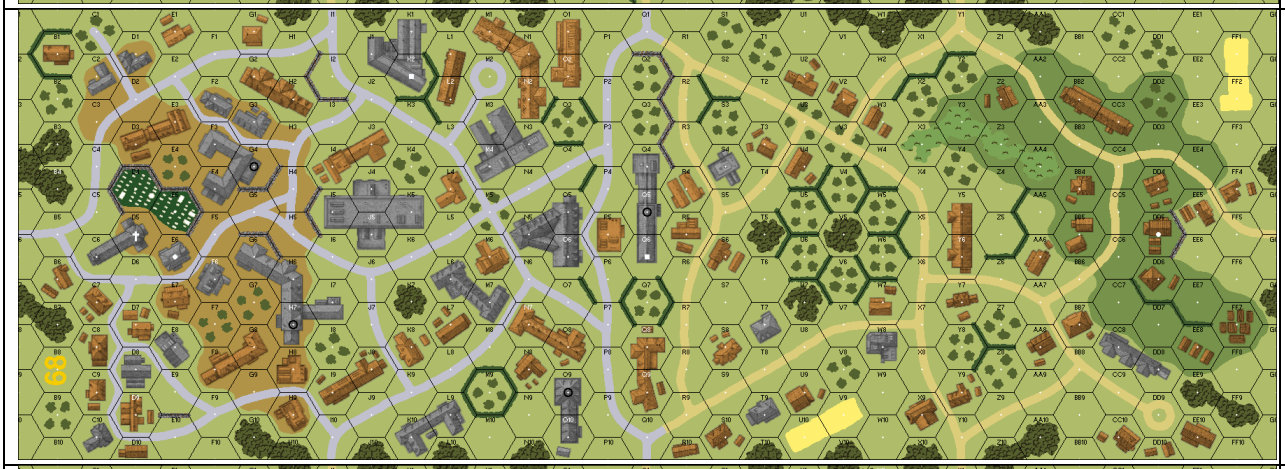
Ye Old Booke



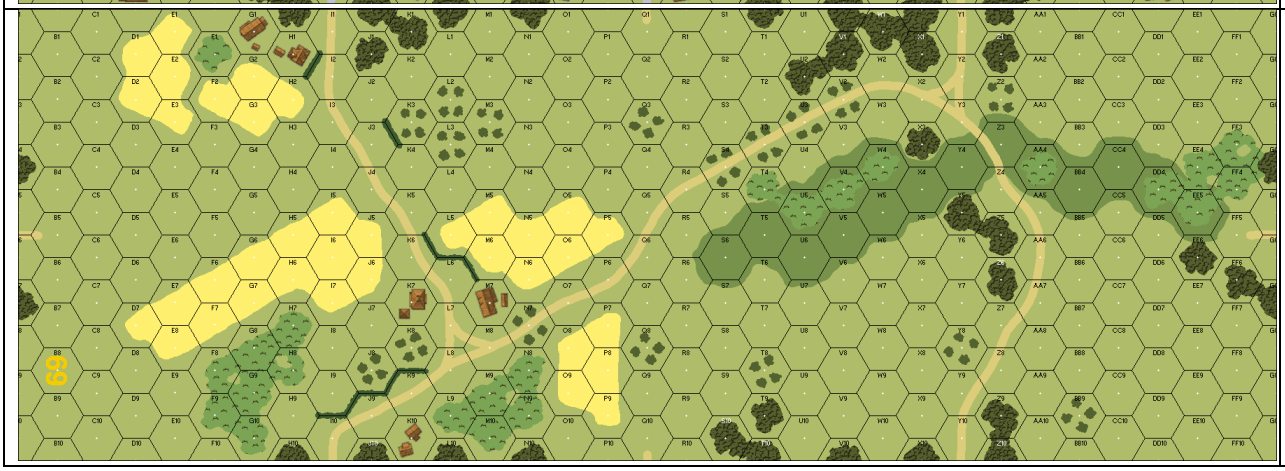
66



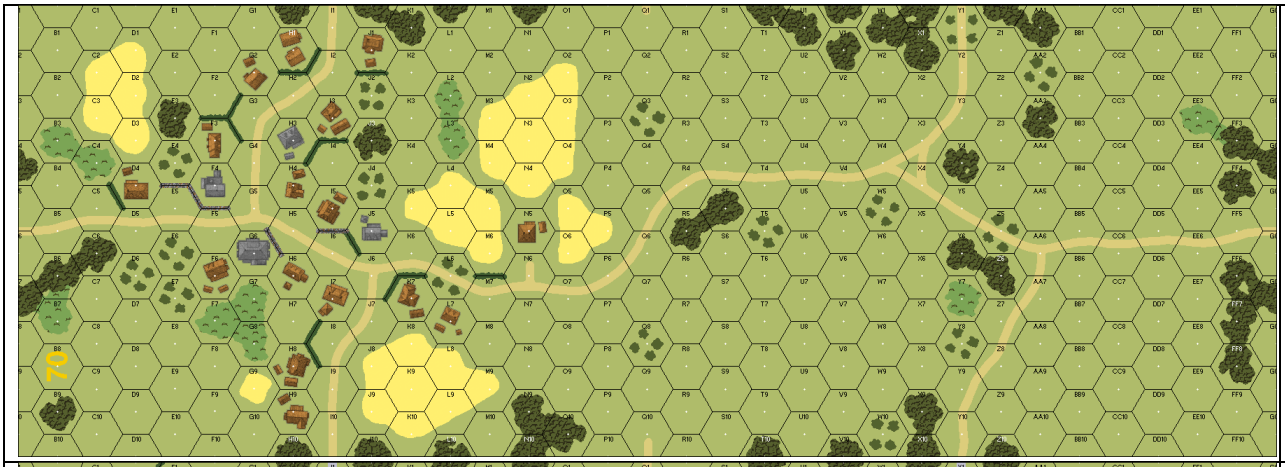
67



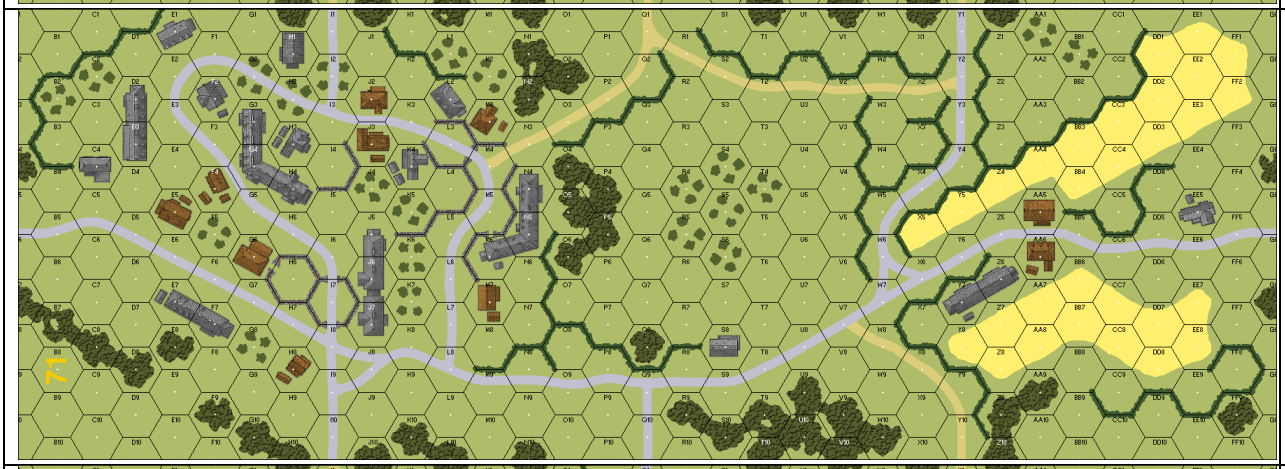
68



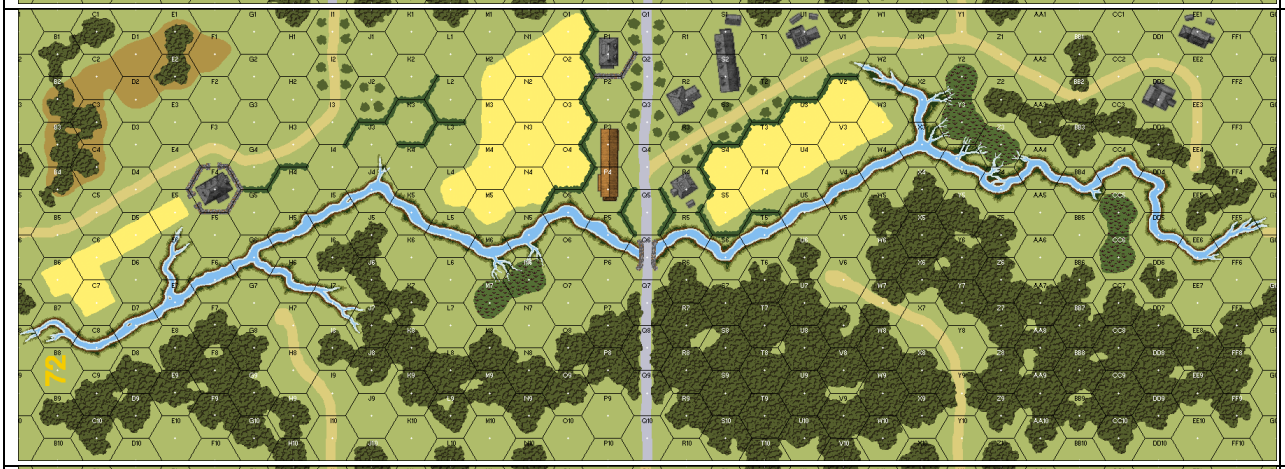
69



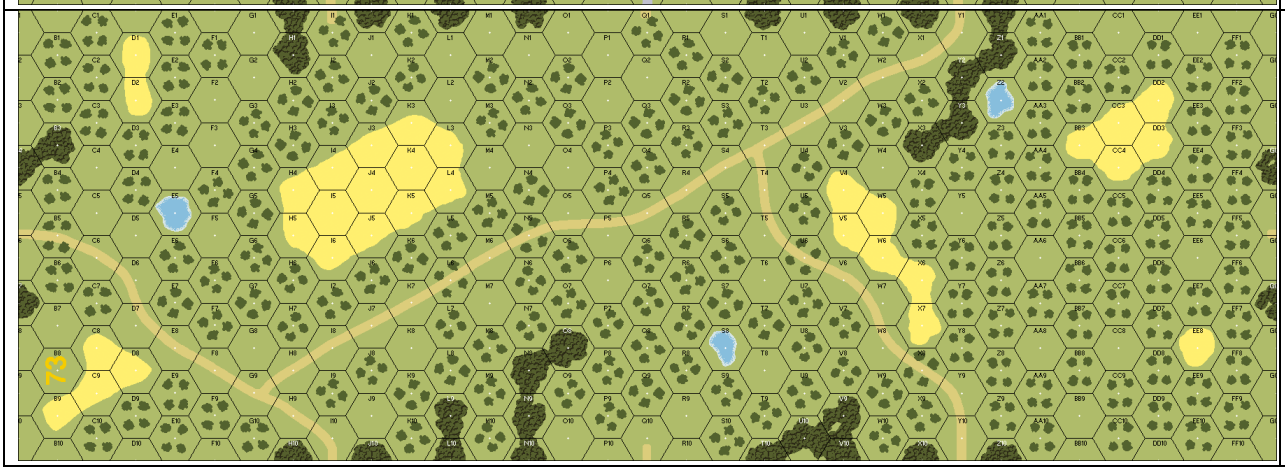
70



71

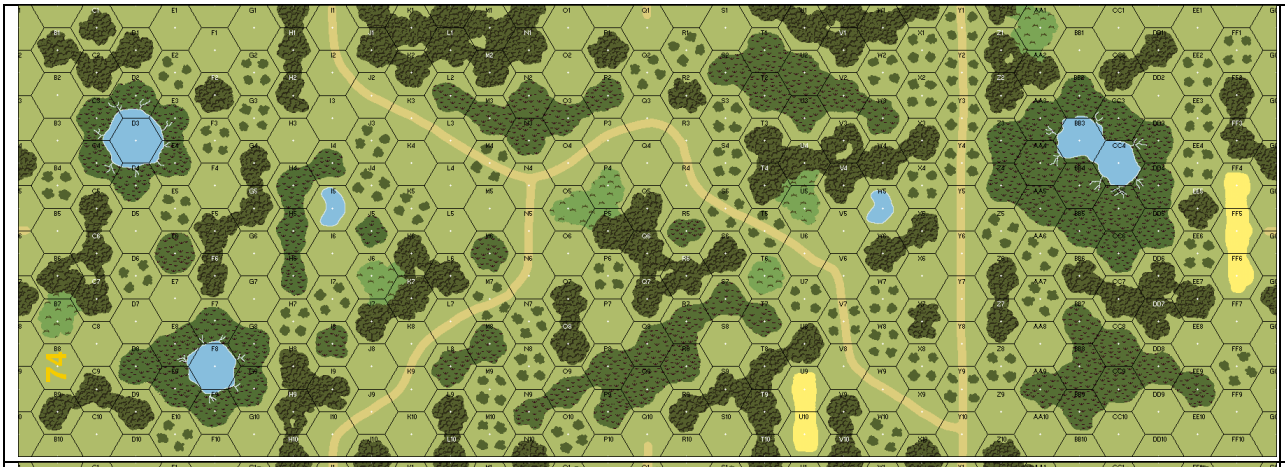


72

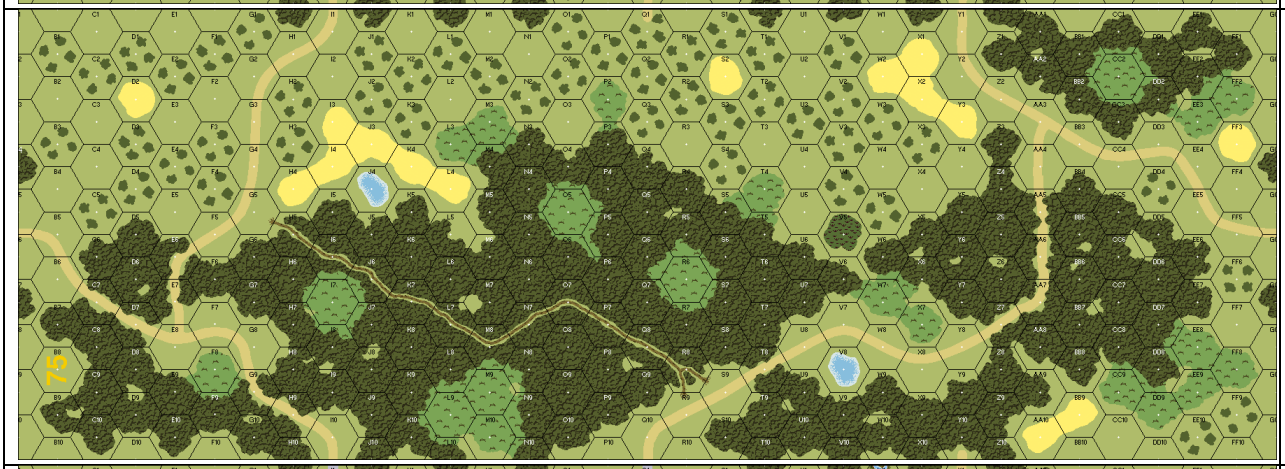


73

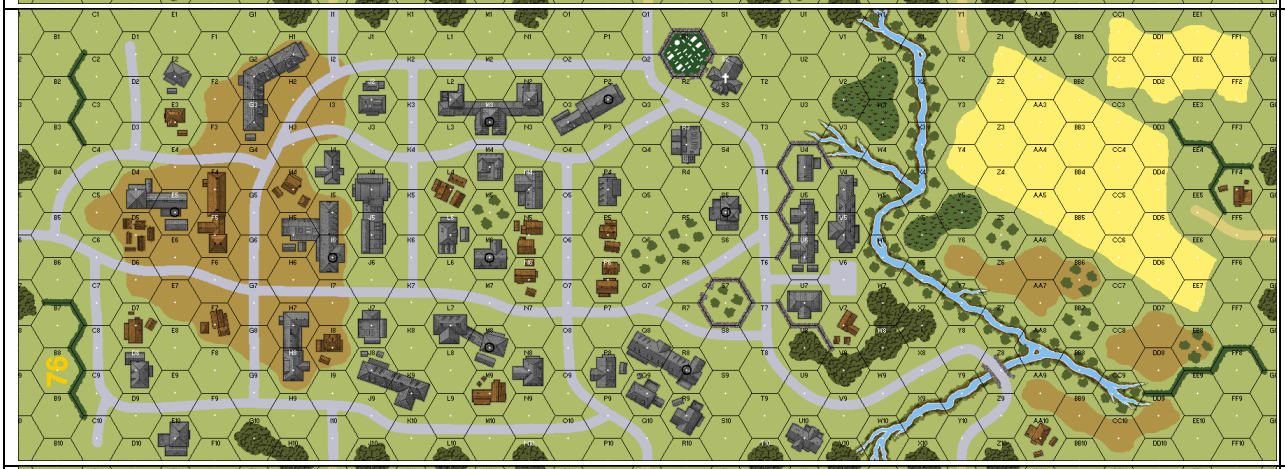
Ye Old Booke



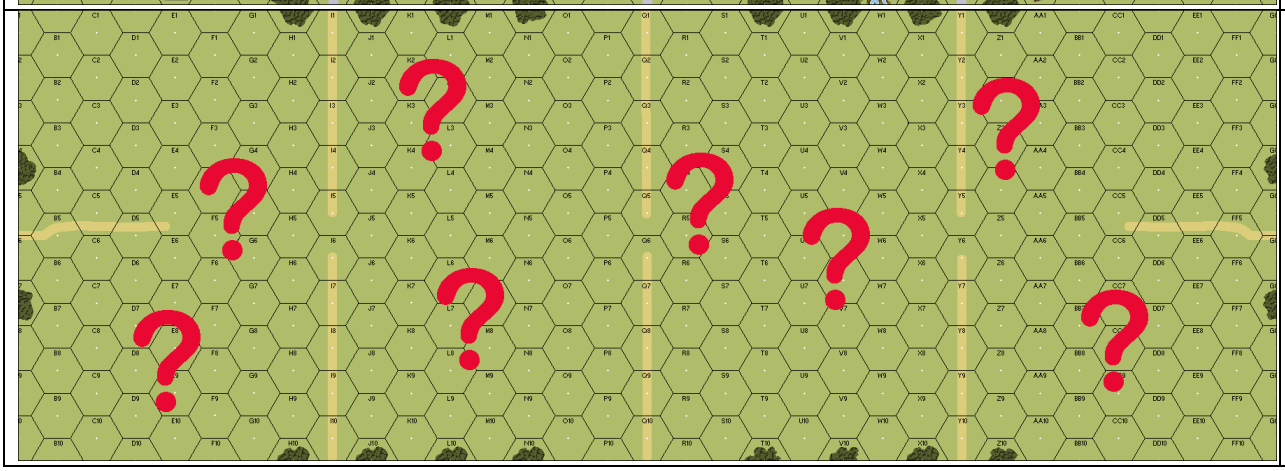
74



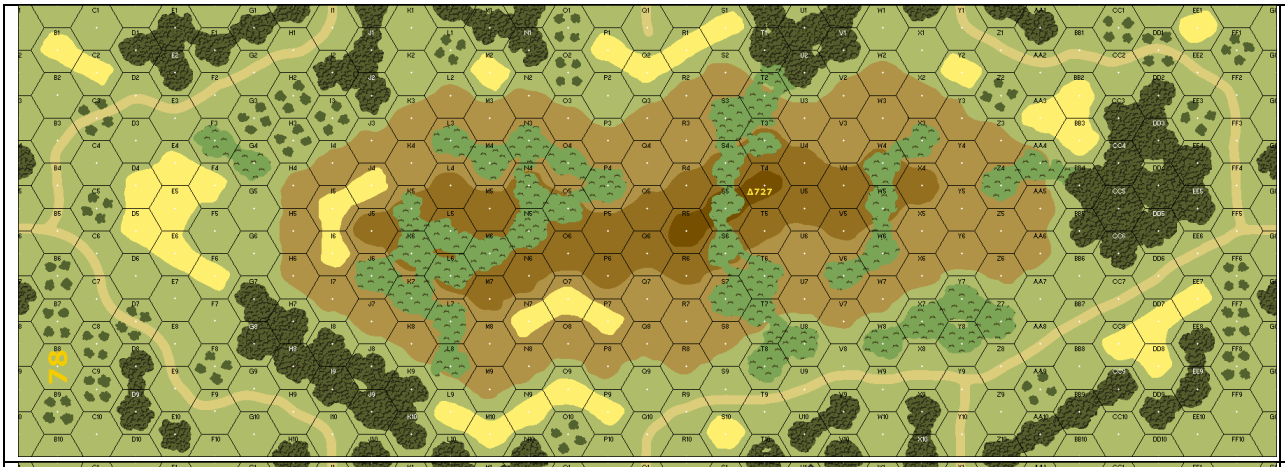
75



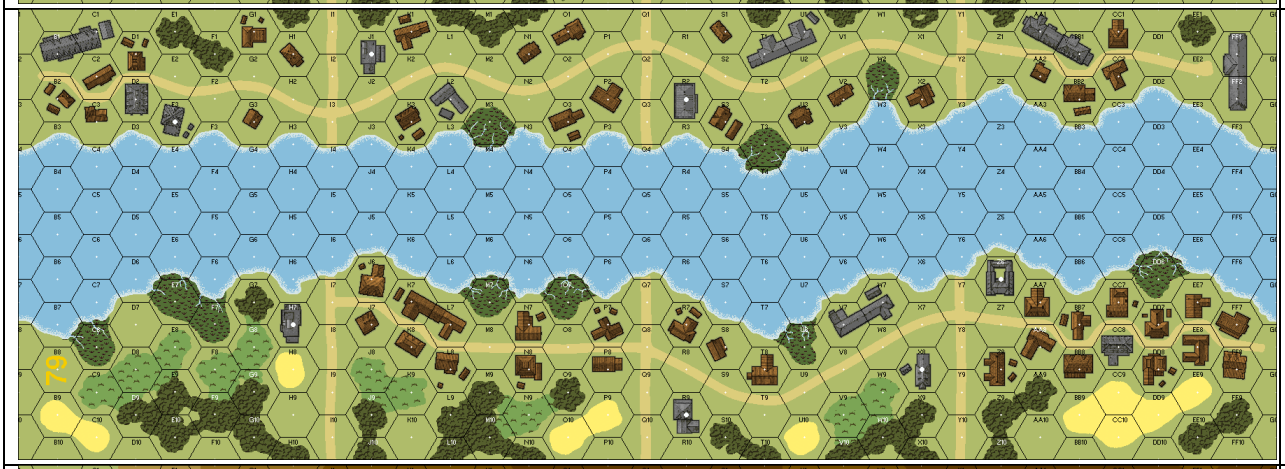
76



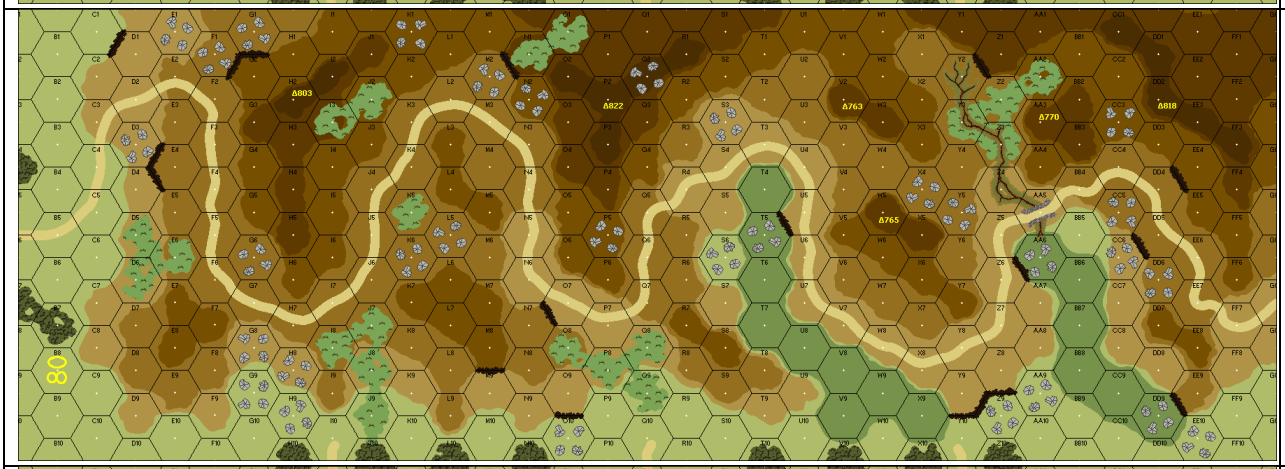
77



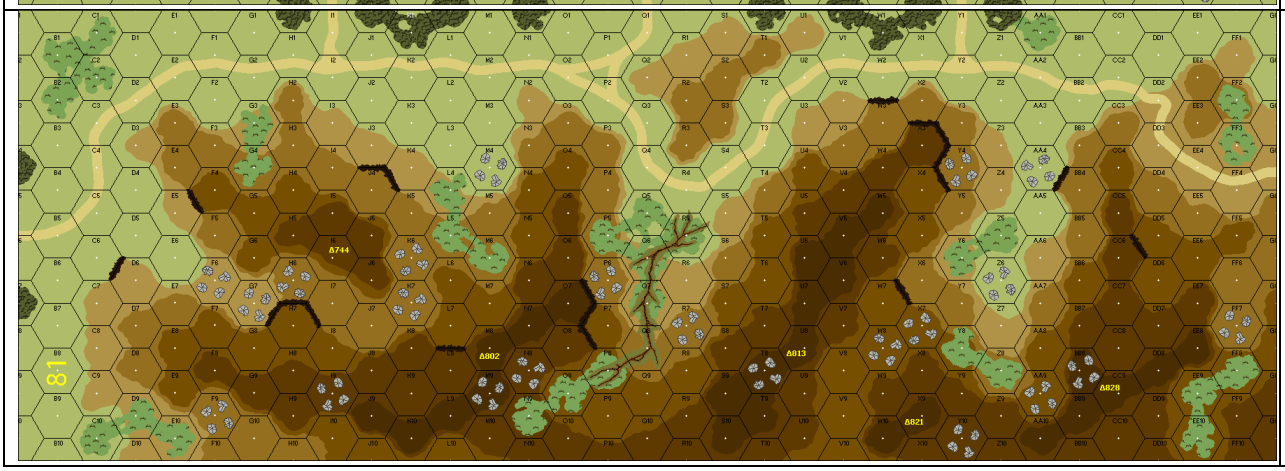
78



79

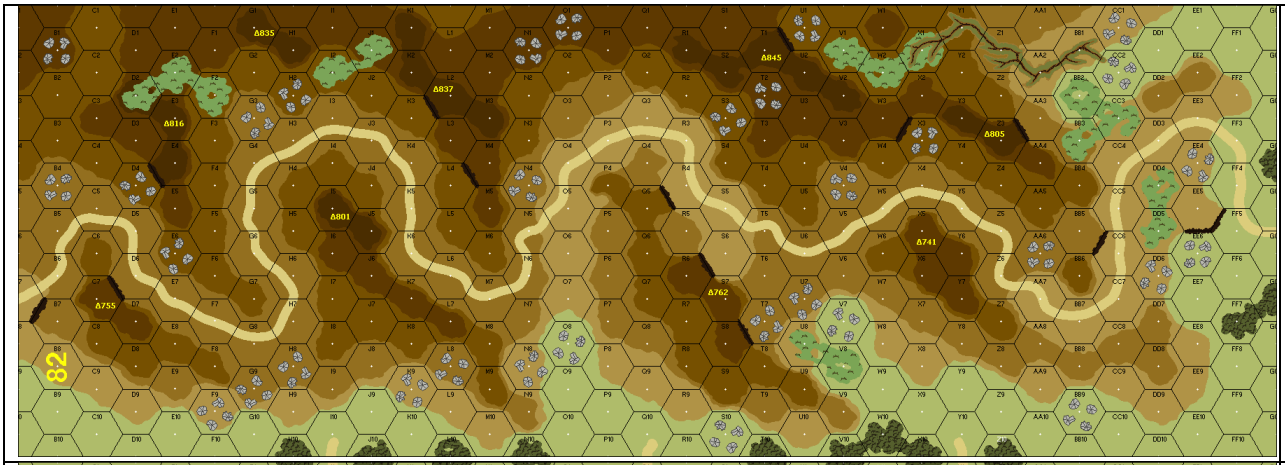


80

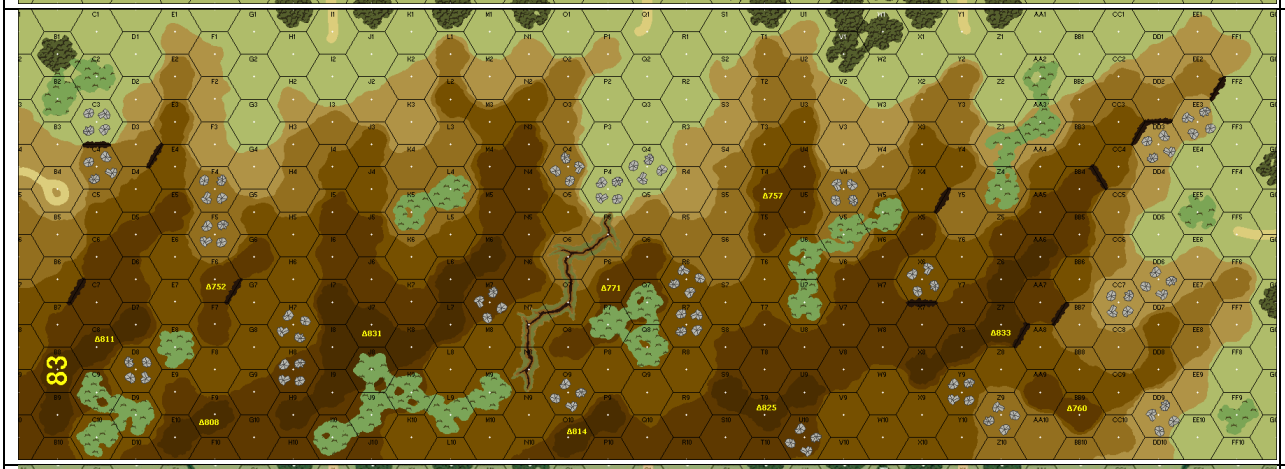


81

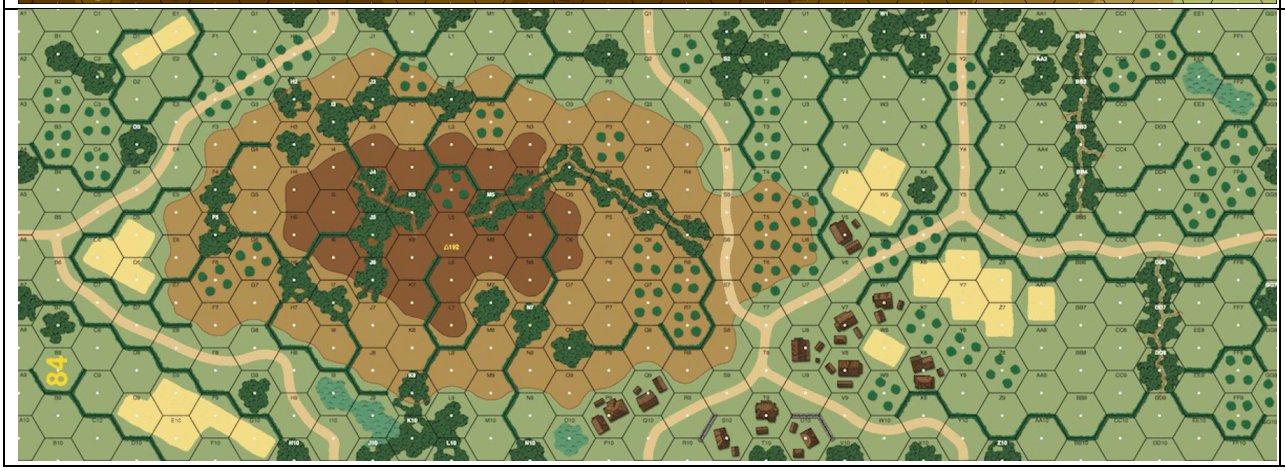
Ye Old Booke



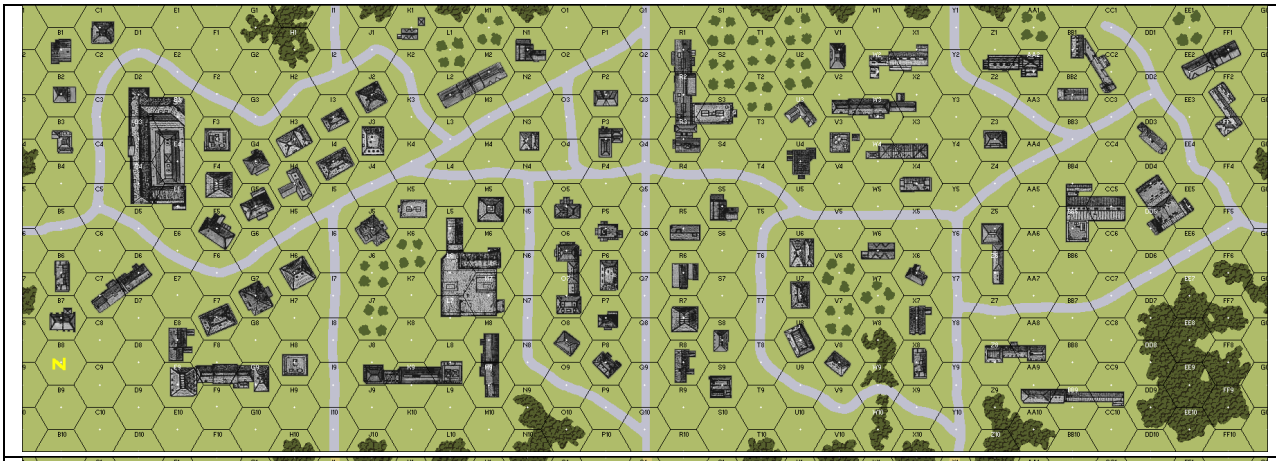
82



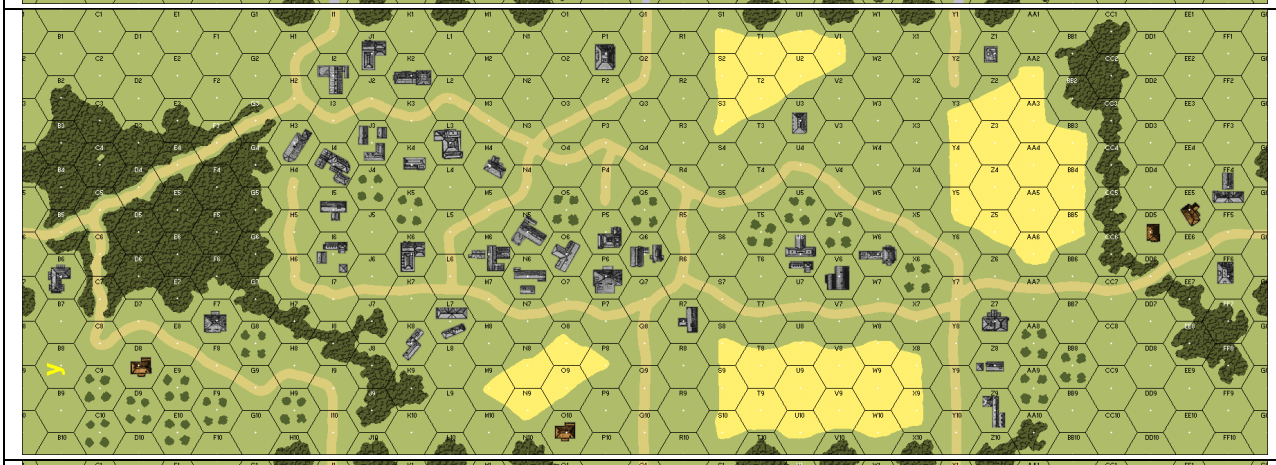
83



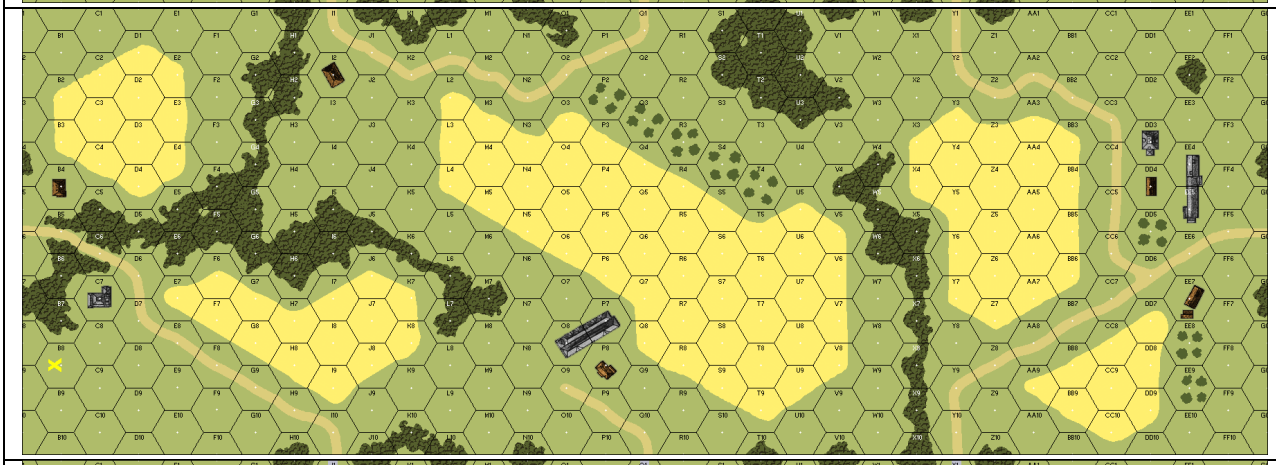
84



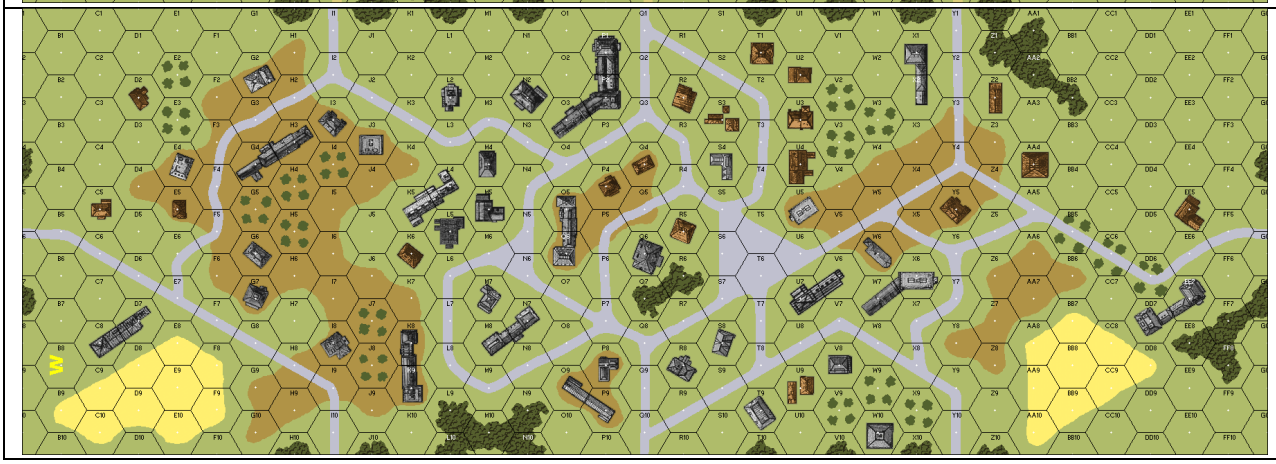
z



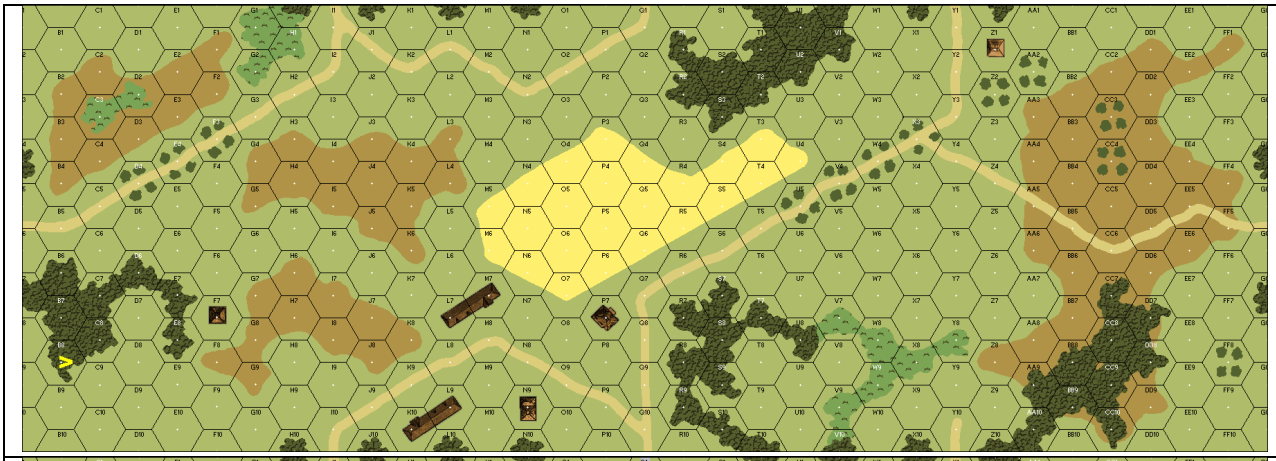
y

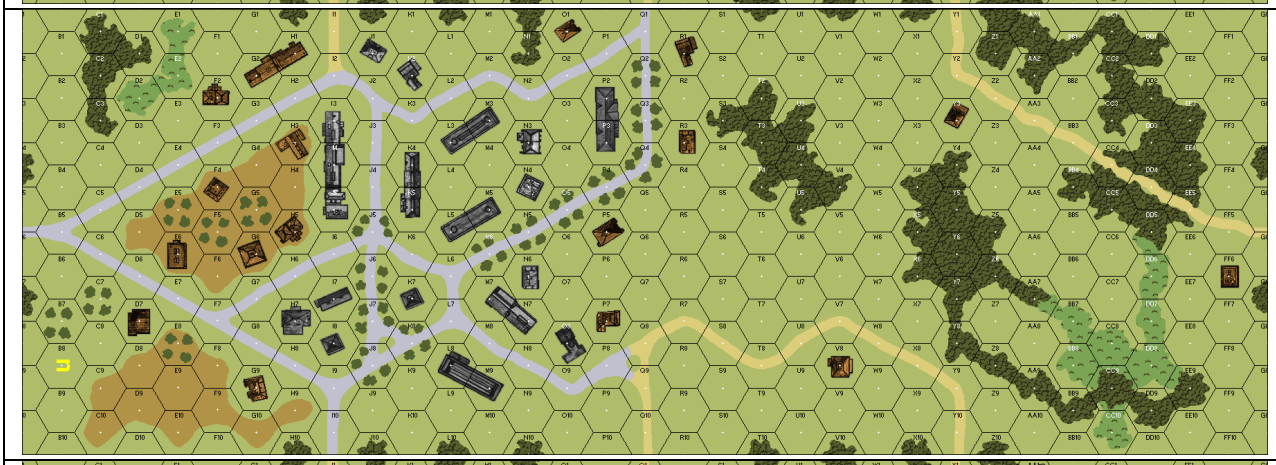


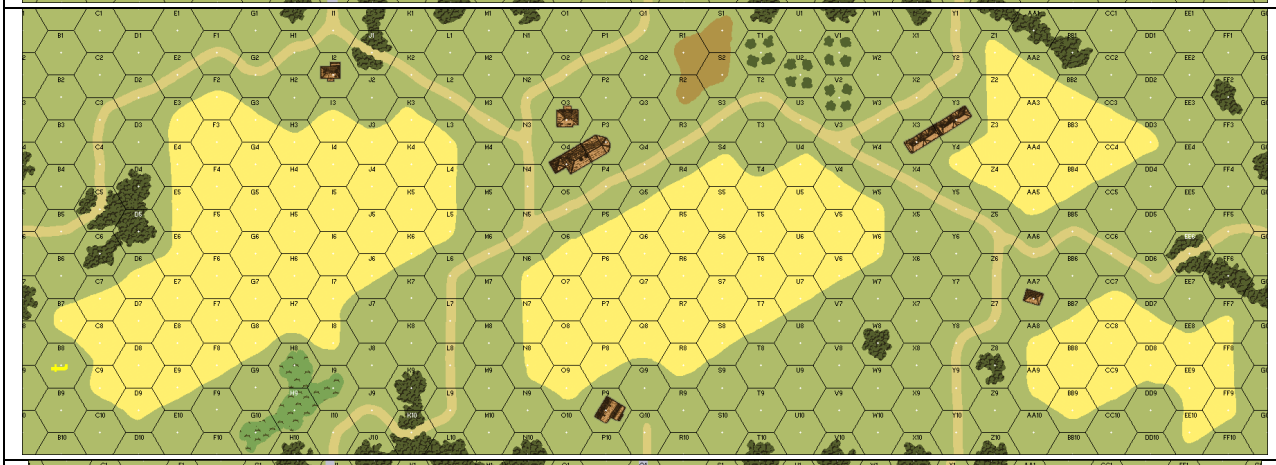
x

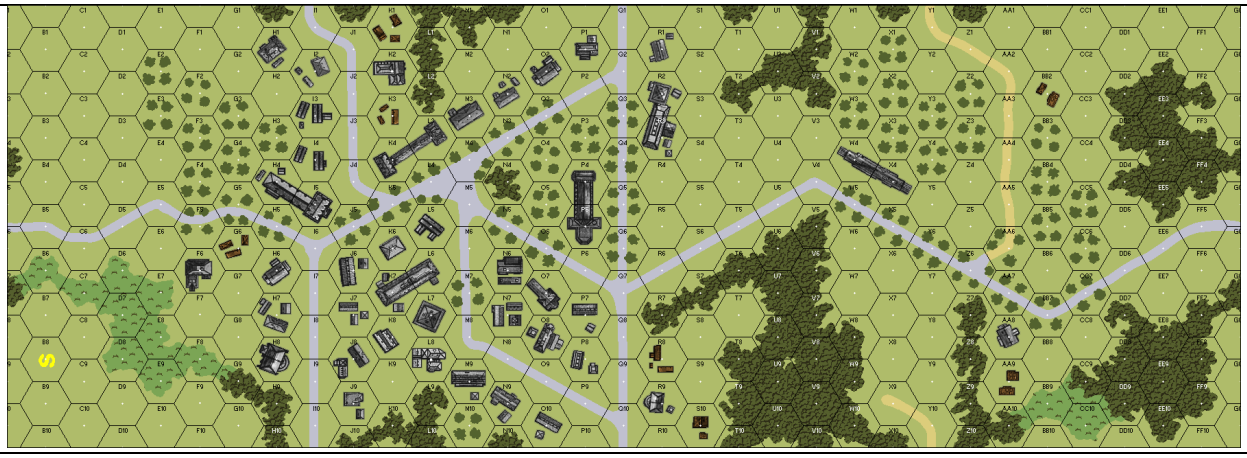


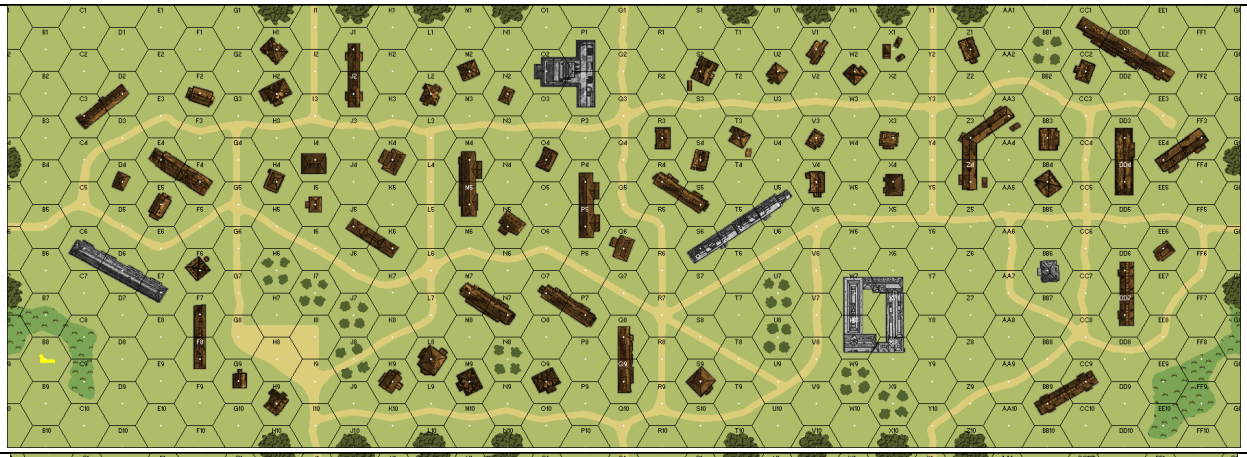
w


v

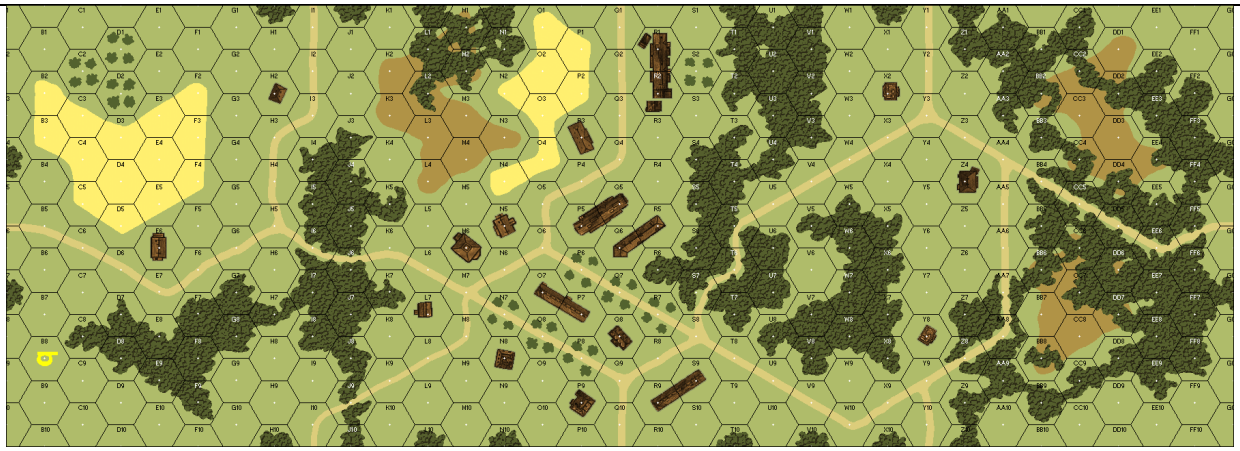

u


t

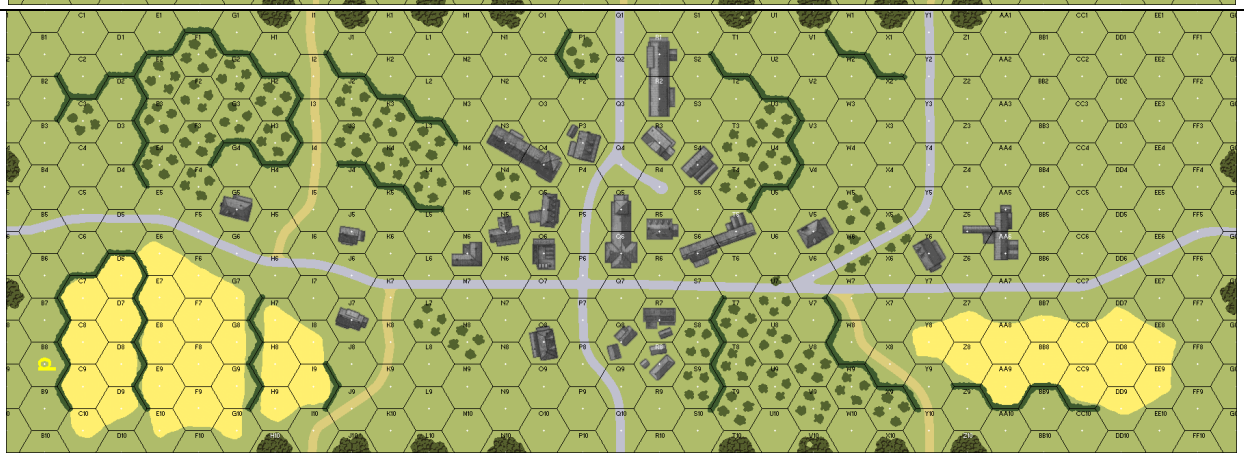

s



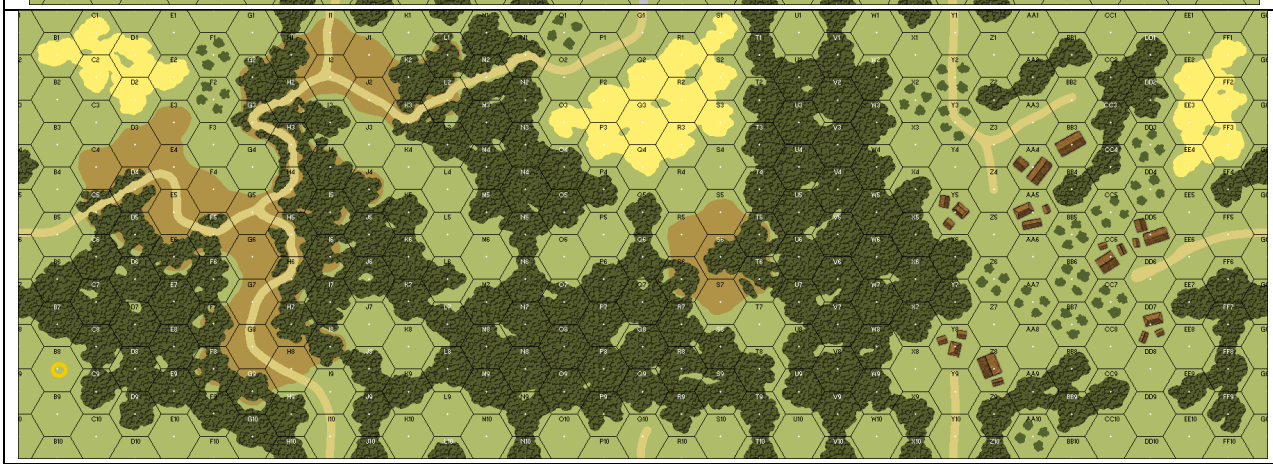
r



q

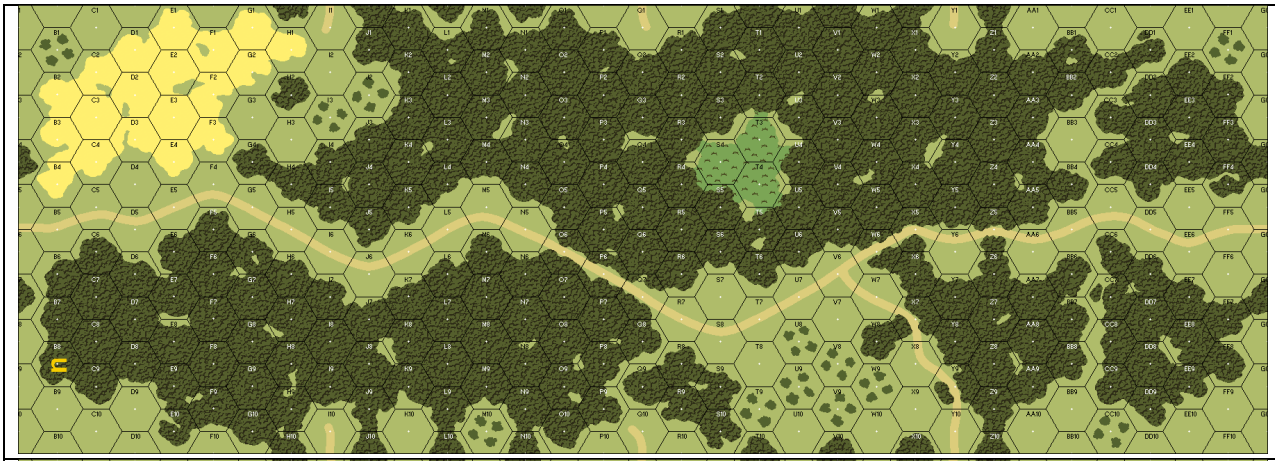


p

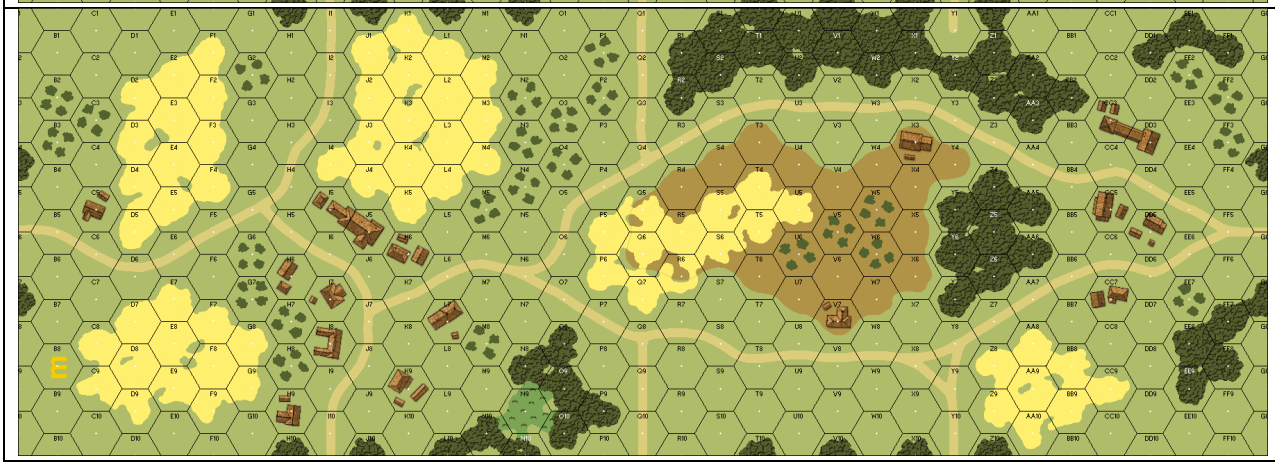


o

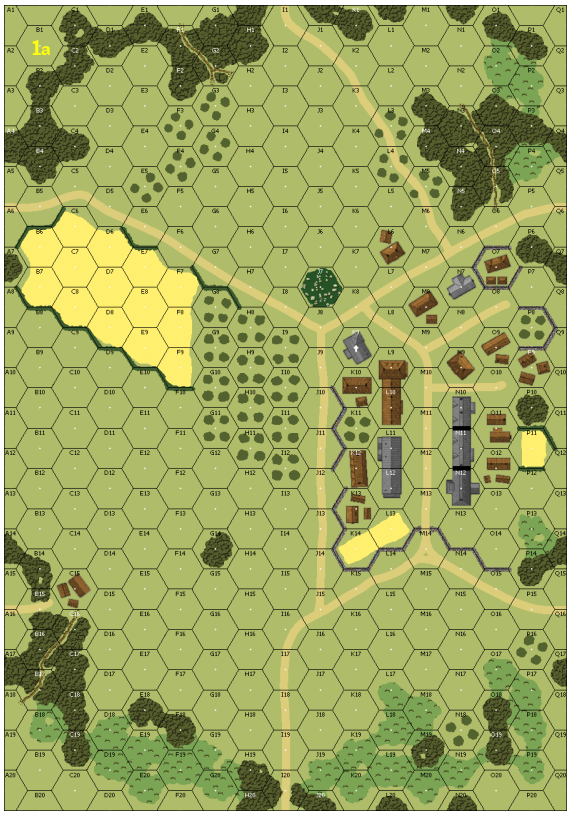
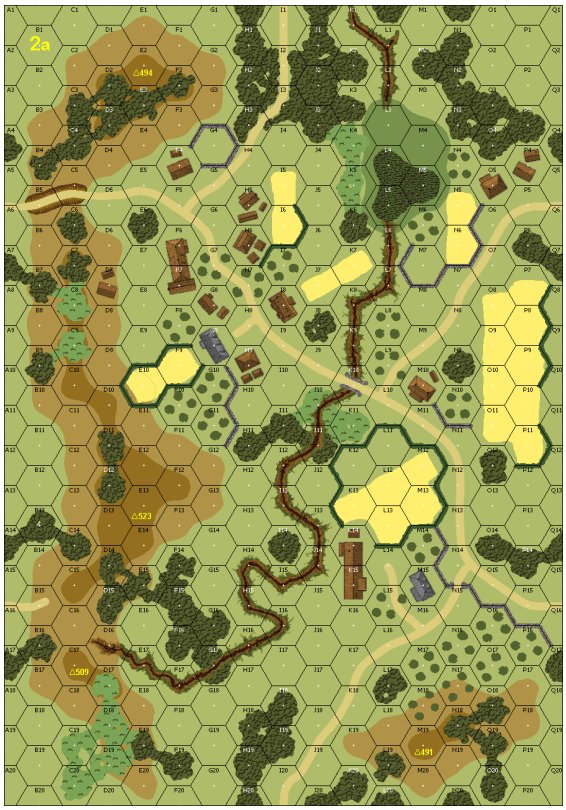
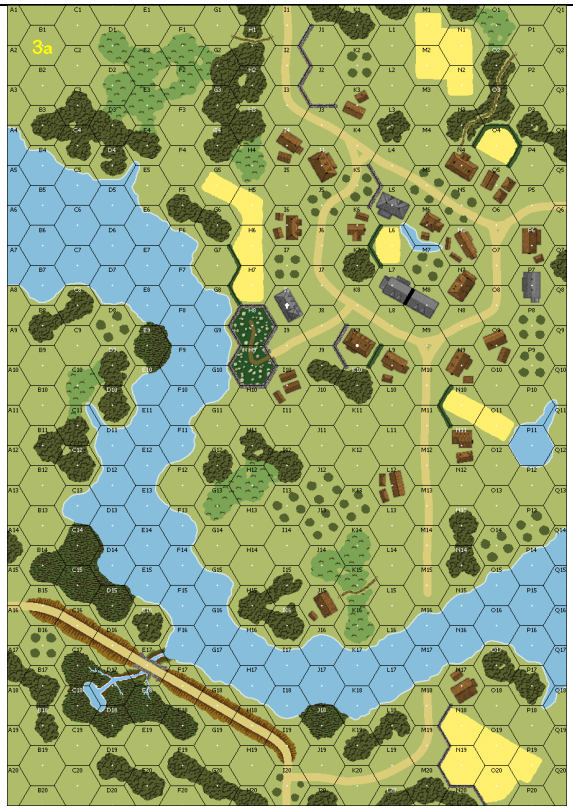
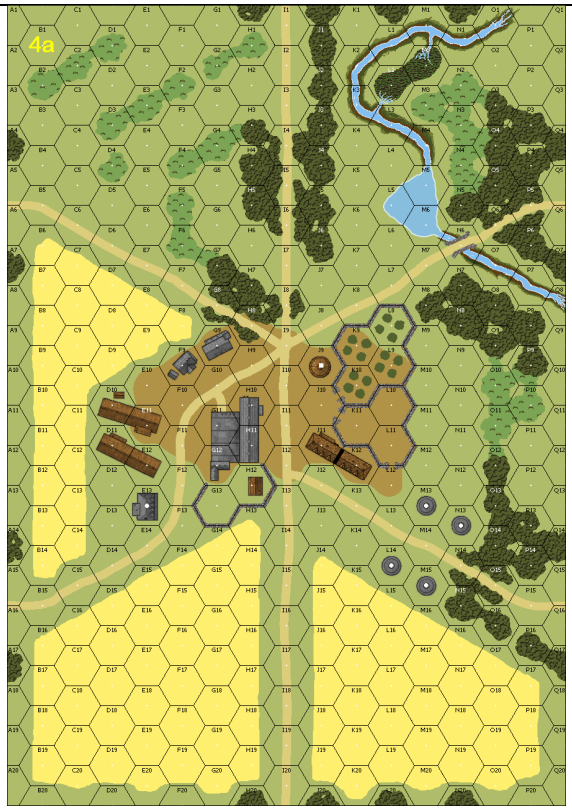
Ye Old Booke



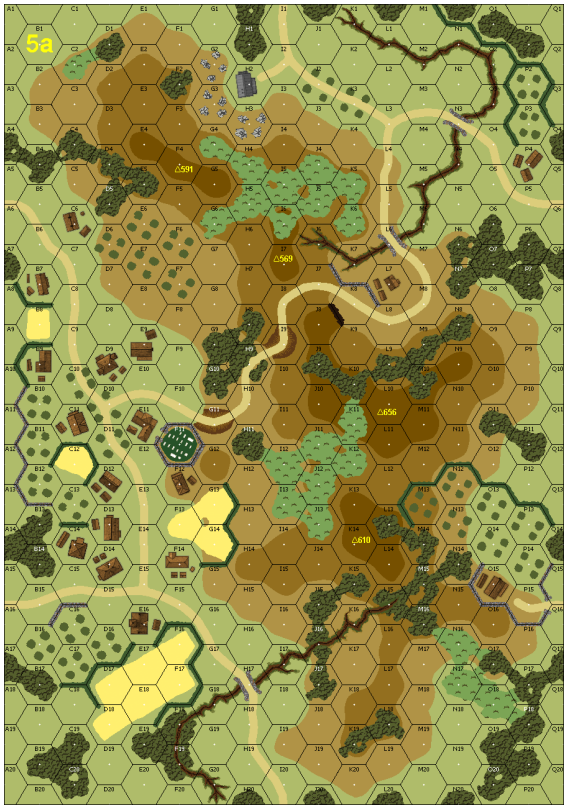
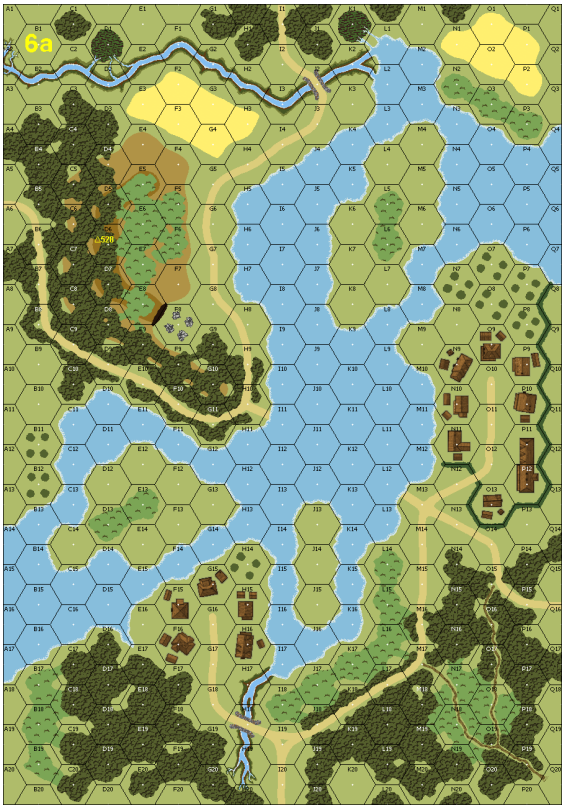
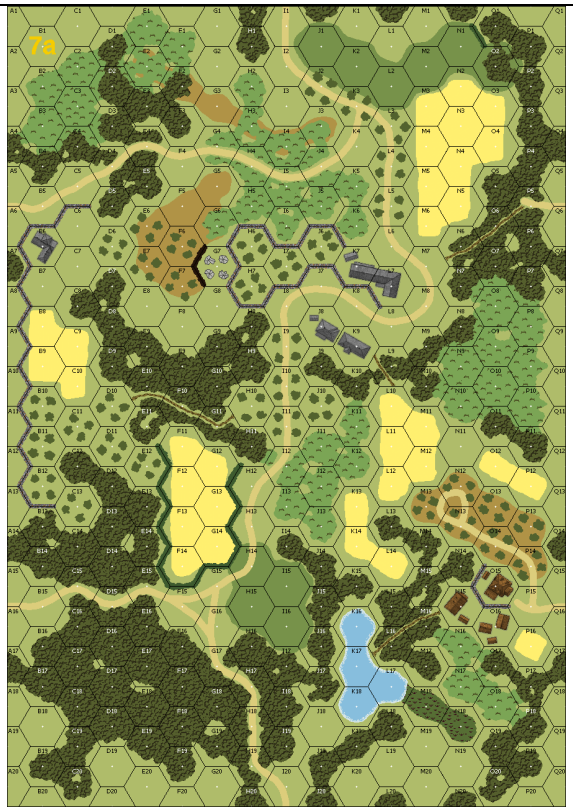
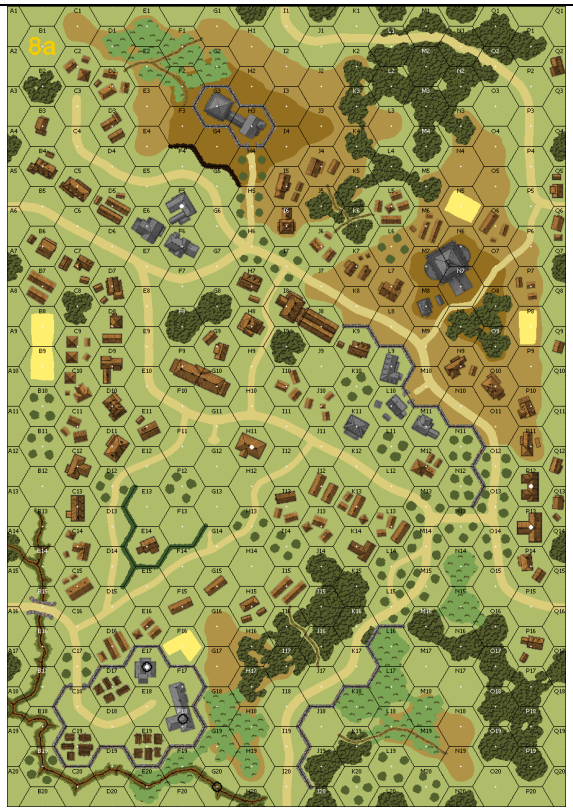
n

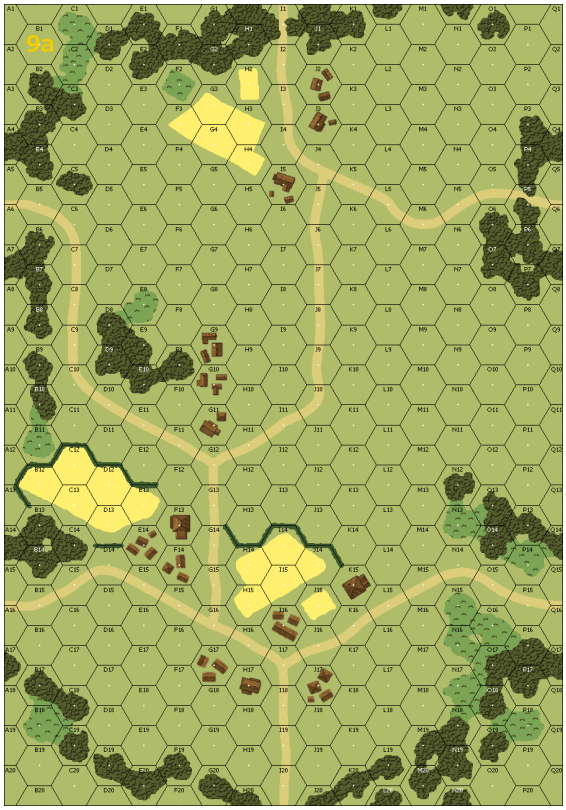


m

1a			**2a**
3a			**4a**

Ye Old Booke

5a			6a
7a			8a

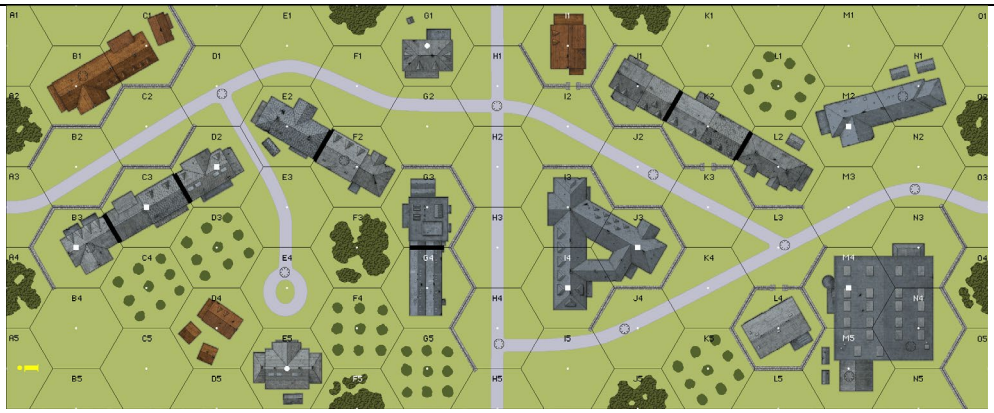
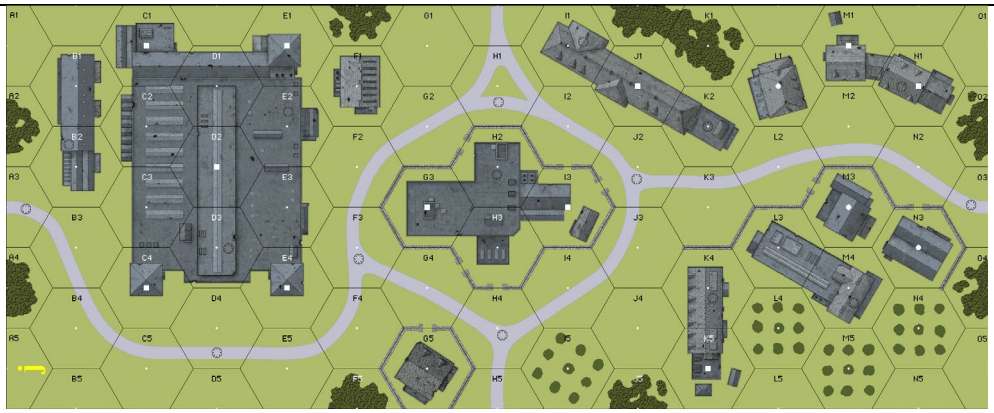
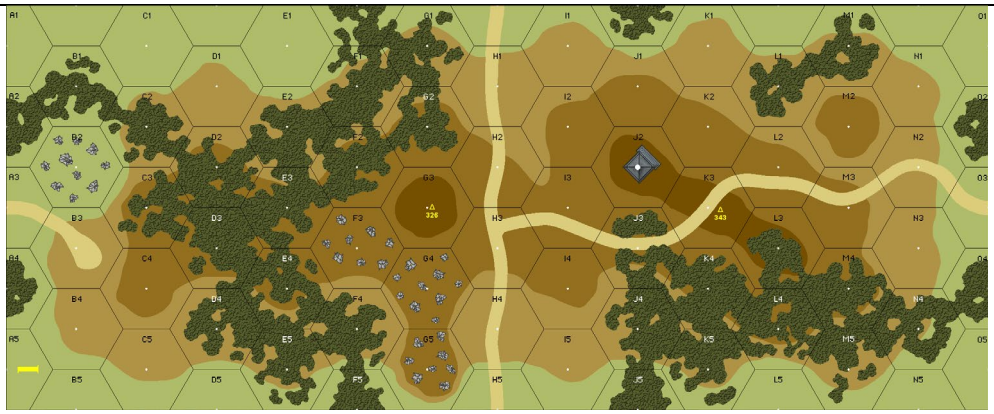
9a			10a
11a			12a

E.2 – Deluxe ASL Boards

	**a**
	**b**
	**c**
	**d**

	**e**
	**f**
	**g**
	**h**

Ye Old Booke

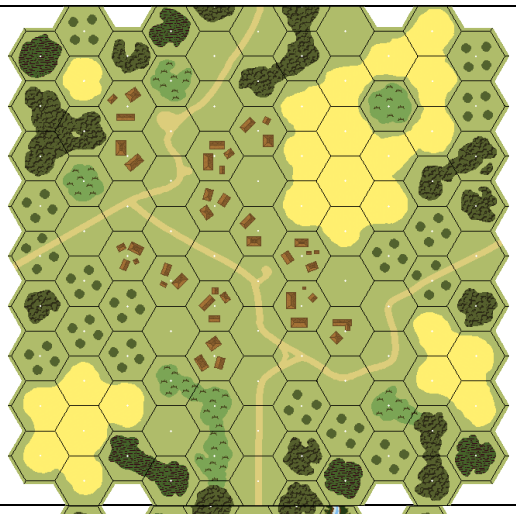
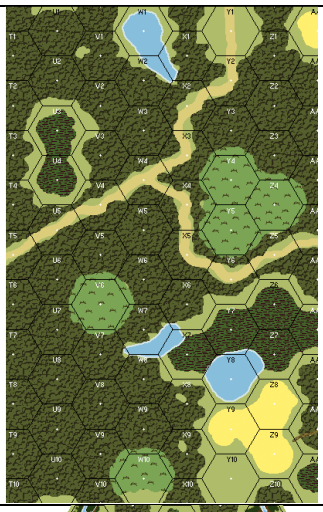
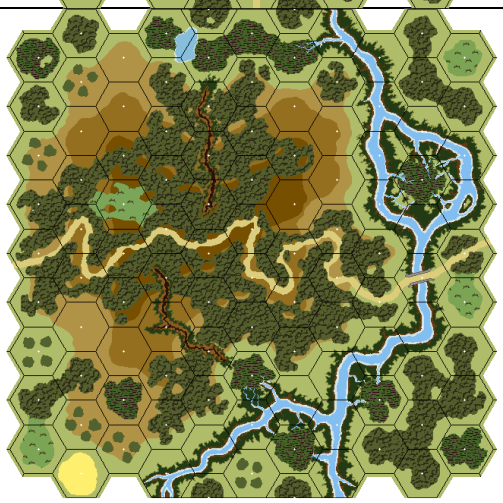
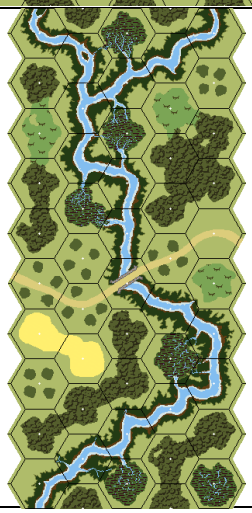
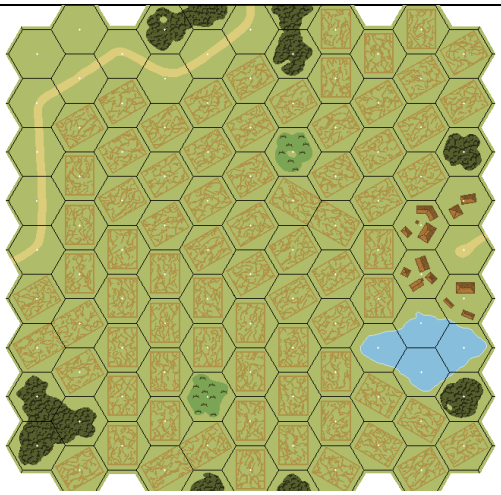
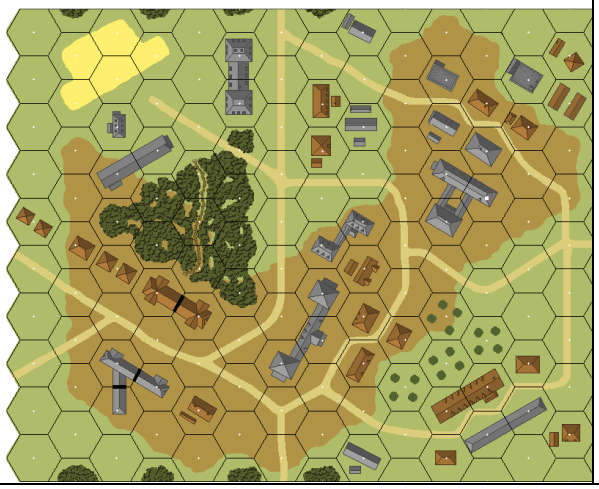
	**i**
	**j**
	**k**
	**l**

Ye Old Booke of VASL Secrets

Appendix F - Thumbnails of Overlays

Unless otherwise specified, the “a” orientation is shown.

F.1 Numbered Overlays

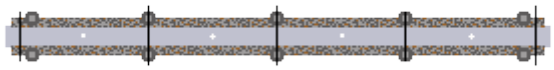
1			4
2			5
3			6

F.2 Generic 1-5 Hex Overlays

Several terrain types have 1-5 hex overlays that simply contain a 1-5 hex patch of that terrain. These types are: Brush (B), Grain (G), Hedge (HD), Hill (HI), Marsh (M), Orchard (O), Open Ground (OG), Ponds (P), Rice Paddies (RP), Shellholes (SH), Vineyards (V), Woods (WD), and Water (WT).

F.3 Bridge (BR) Overlays

Bridge overlays (br1-7) are 1-7 hexes long.



br4 (rotated 90 degrees clockwise)

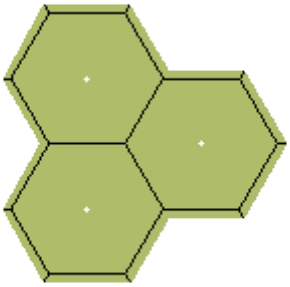
F.4 Deluxe (DX) Overlays

dx1

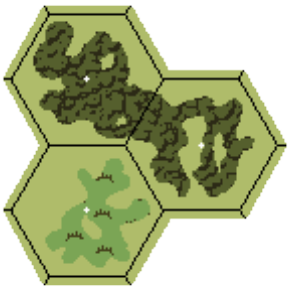
dx2

dx3

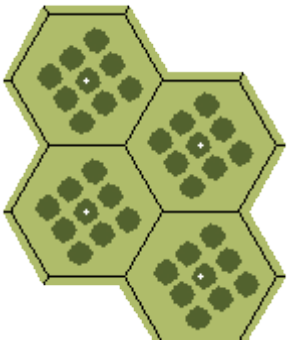
dx4



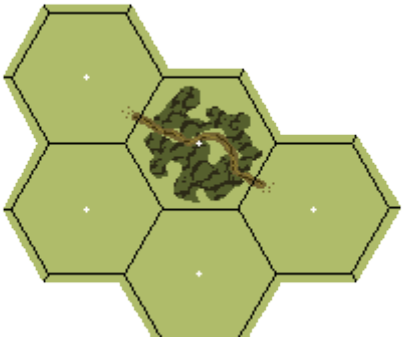
dx5



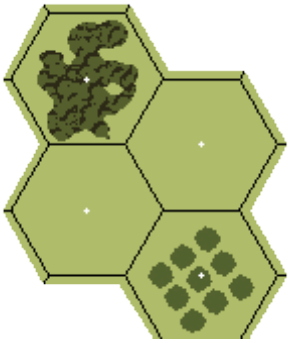
dx6



dx7



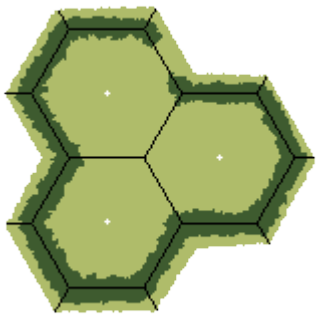
dx8



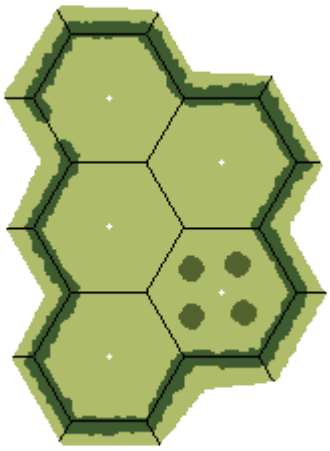
dx9

F.5 Hedge (HD) Overlays

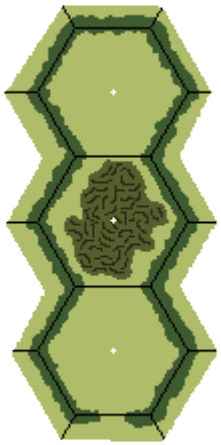
HD1-5 are generic overlays with 1-5 hexes.



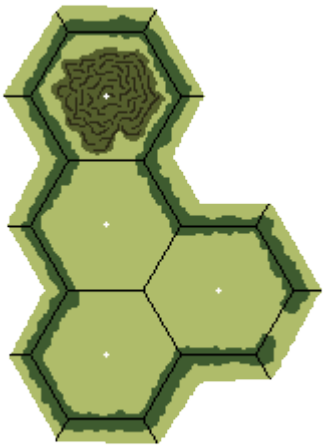
hd3



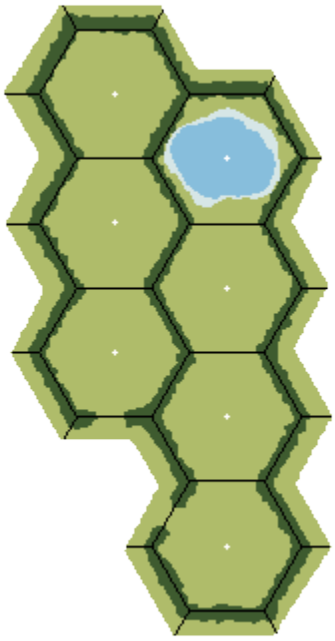
hd5



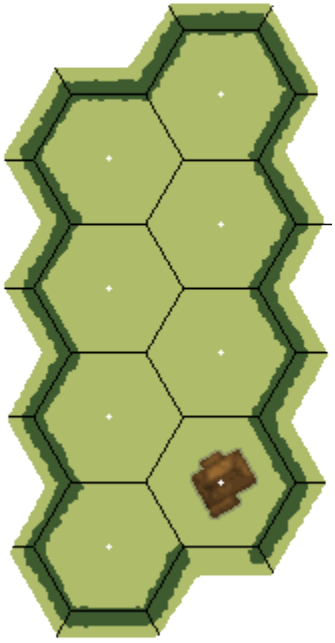
hd6



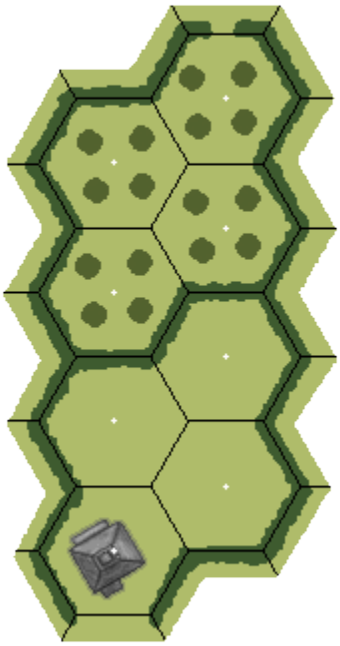
hd7



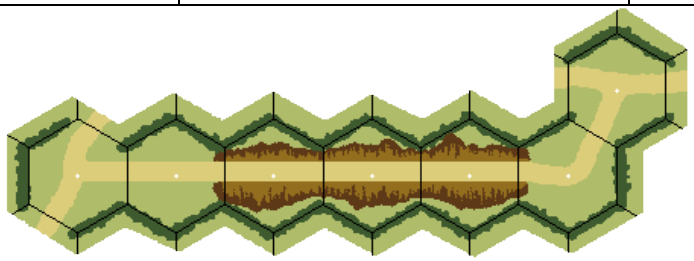
hd8



hd9



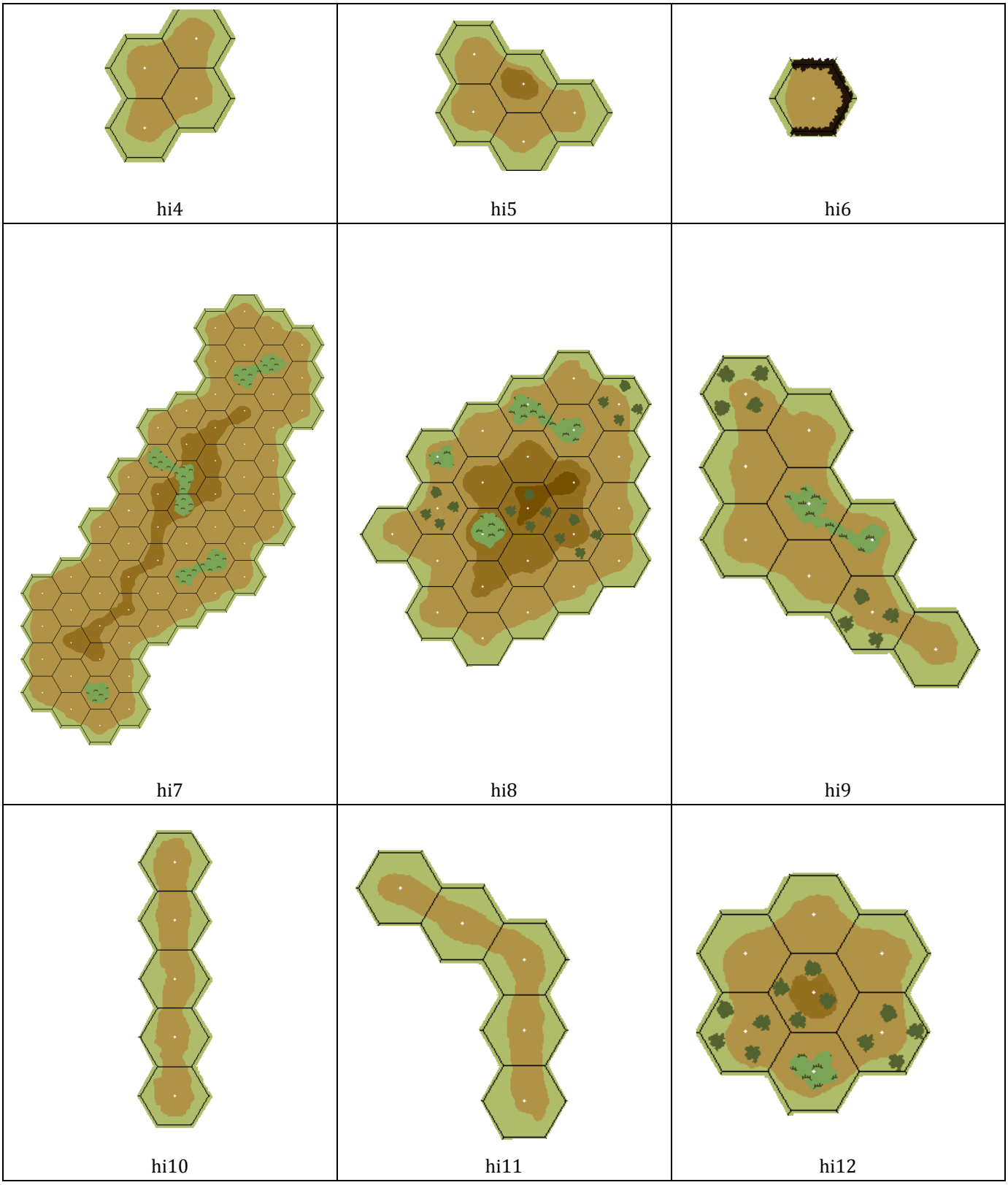
hd10



hd11 (rotated 90 degrees clockwise)

F.6 Hill (HI) Overlays

hi1-3 are simple 1-3 hex Level 1 hill overlays, much like hi4 is a four-hex Level 1 overlay, shown below.

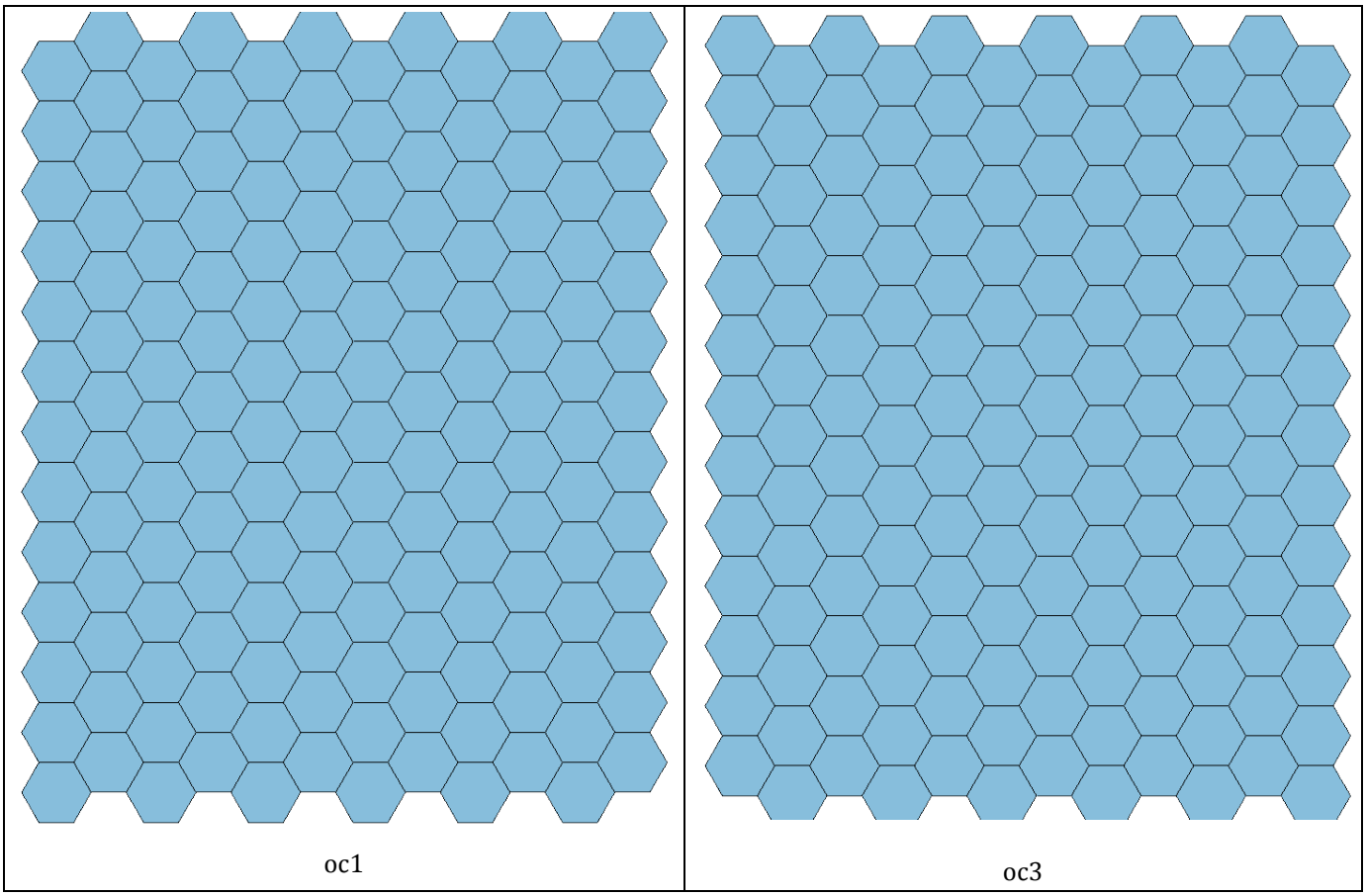


hi4 | hi5 | hi6

hi7 | hi8 | hi9

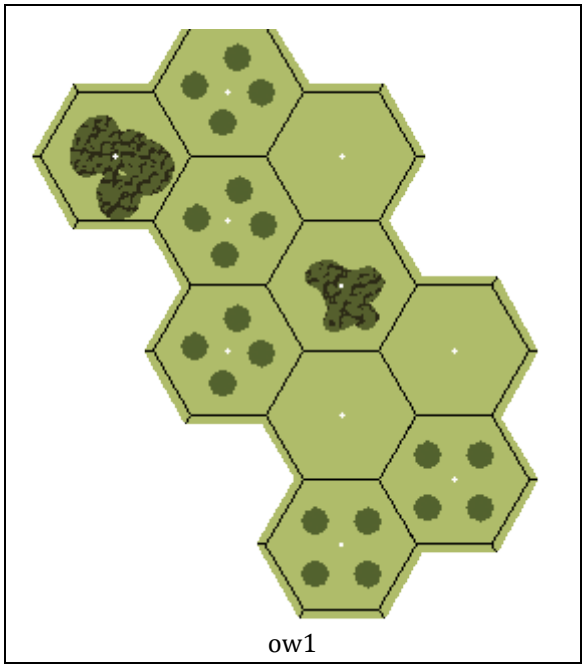
hi10 | hi11 | hi12

F.7 Ocean (OC) Overlays



oc1 oc3

F.8 Orchard-Woods (OW) Overlays

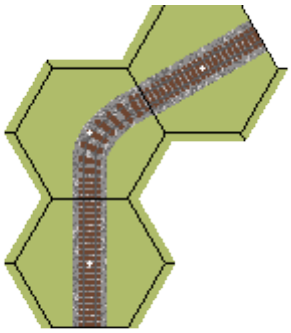


ow1

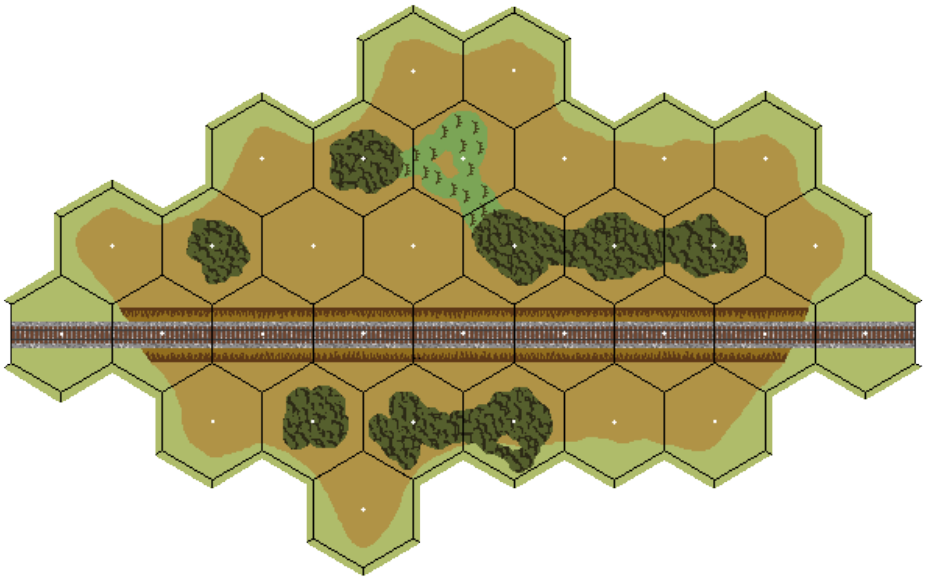
F.9 Railroad (RR) Overlays

Several of the Railroad overlays are completely identical to each other [EX: RR1 and RR2] while others have the same artwork but different positions of hexes 1 and 2 [EX: RR3 and RR4].

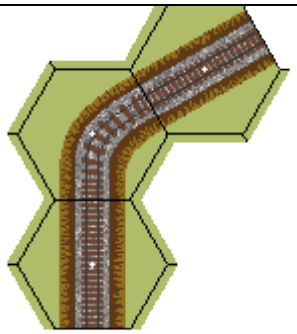
RR1 & RR2: 11-hex straight GLRR
RR3 & RR4: 6-hex straight GLRR
RR7 & RR8: 11-hex straight ELRR
RR11 & RR12: 6-hex straight ELRR



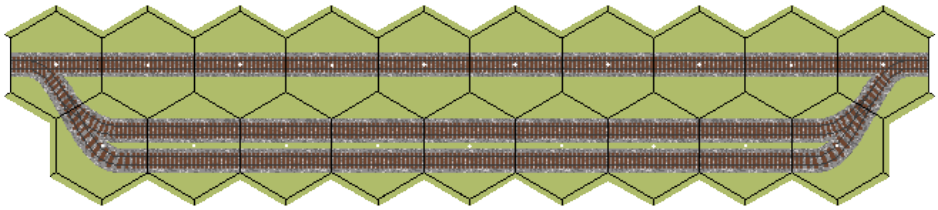
rr5 and rr6



rr13 (rotated 90 degrees clockwise)

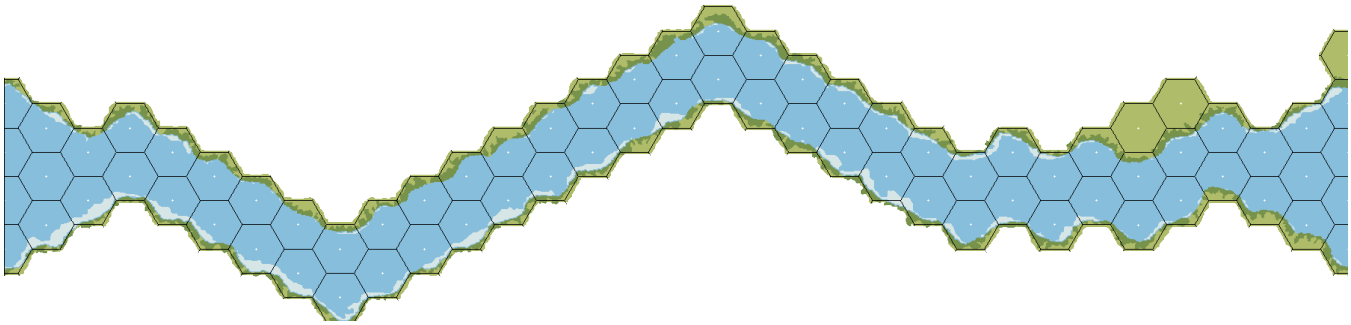


rr9 and rr10



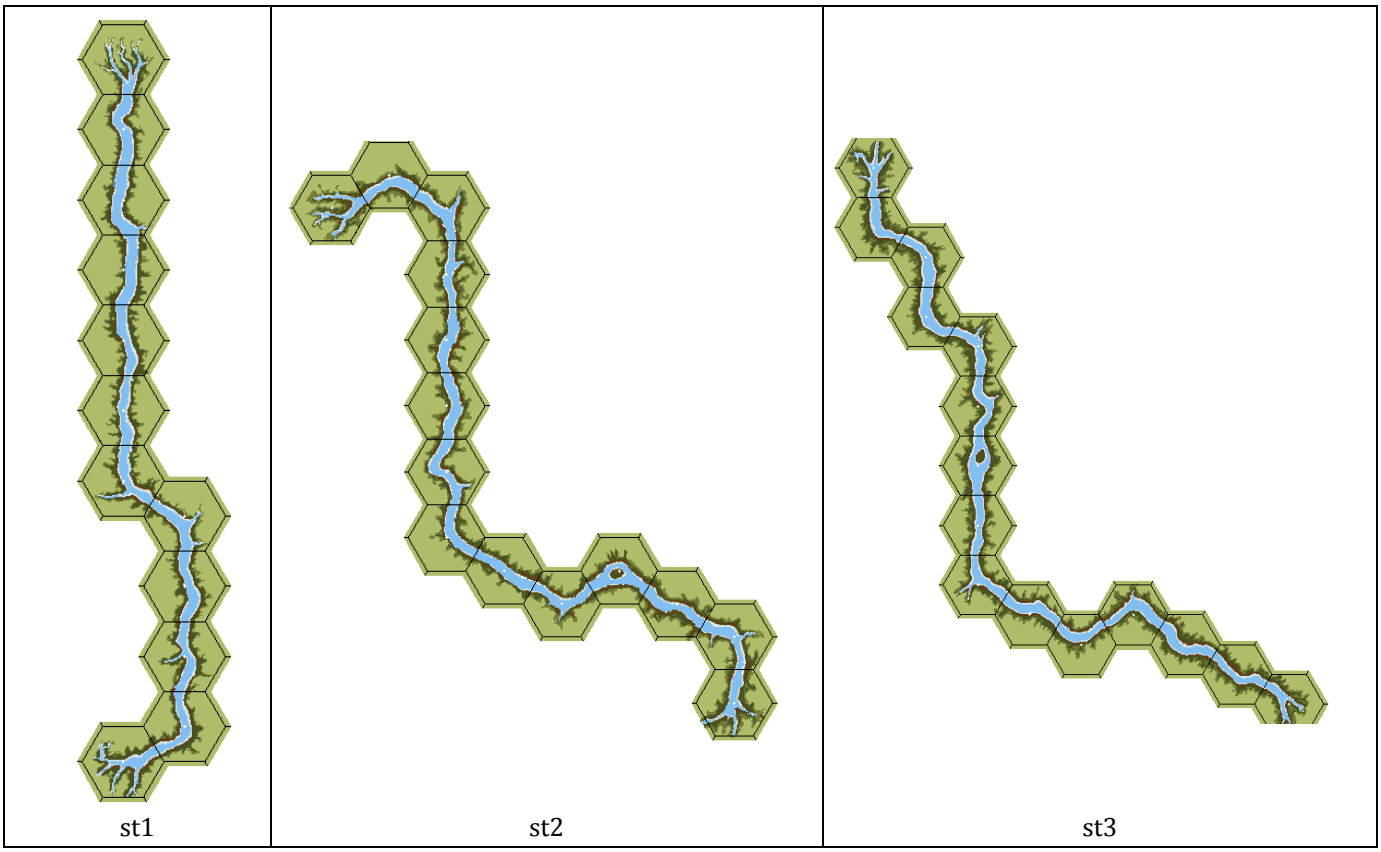
rr14 (rotated 90 degrees clockwise)

F.10 River (RV) Overlay



rv1b

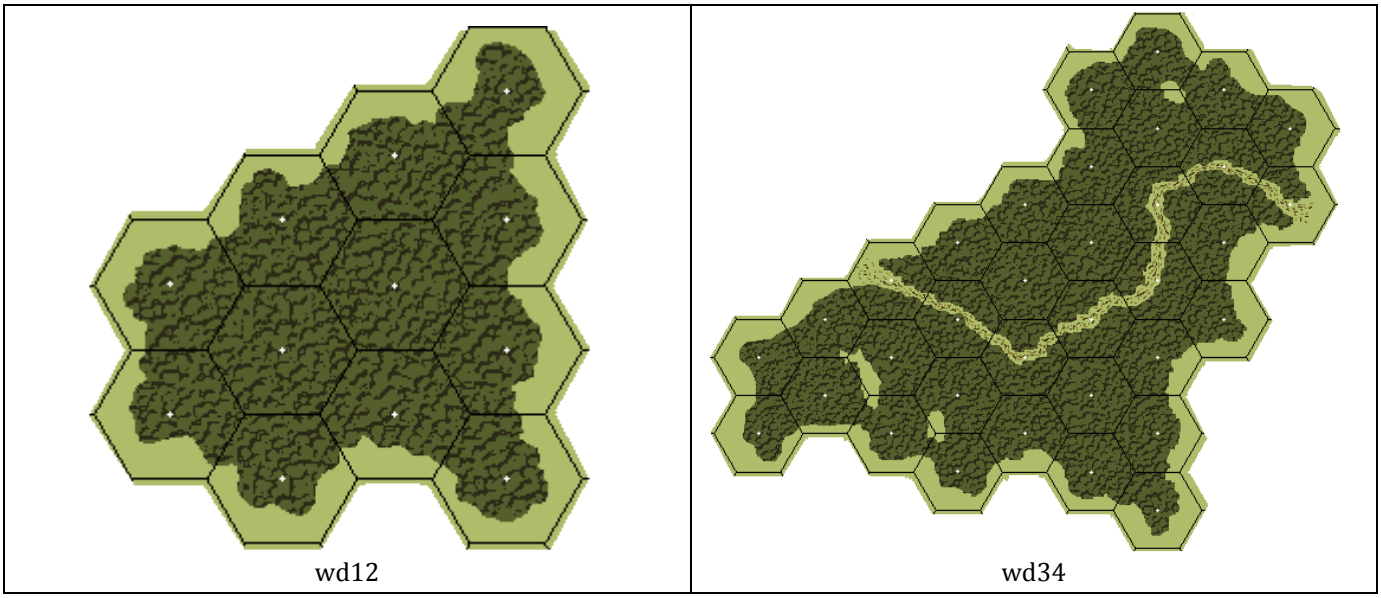
F.11 Stream (ST) Overlays



st1 st2 st3

F.12 Woods (WD) Overlays

WD1-5 are simple 1-5 hex generic overlays.



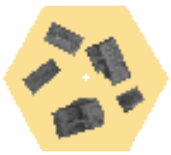
wd12 wd34

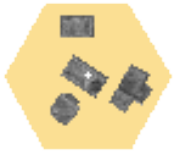
F.13 Building (X) Overlays


x1


x2


x3


x4


x5


x6


x7

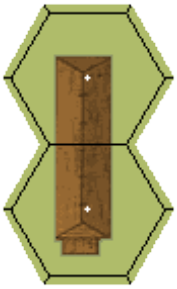

x8


x9

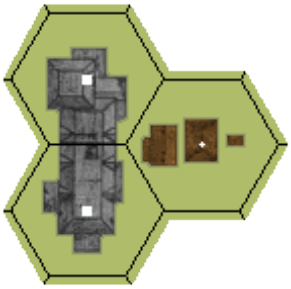

x10

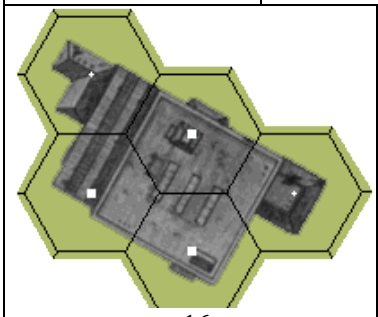

x11

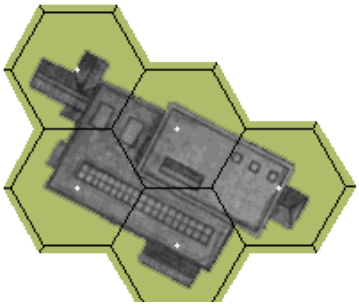

x12

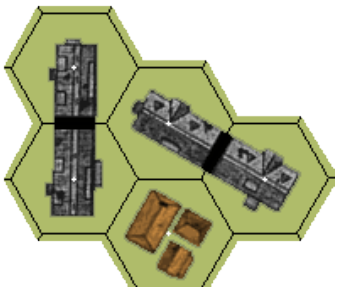

x13

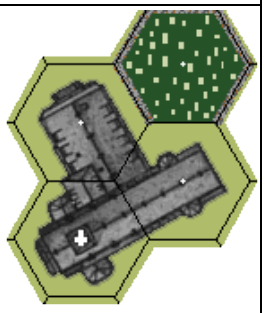

x14

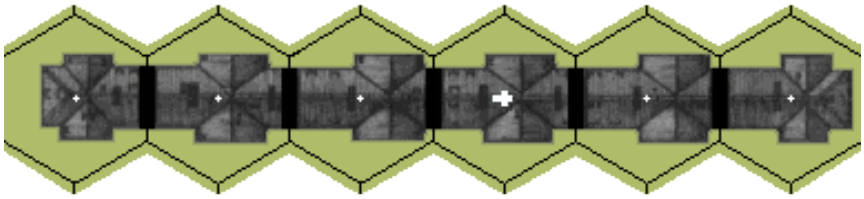

x15


x16

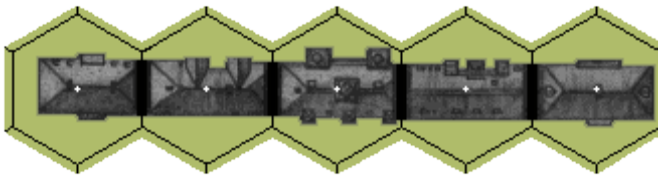

x17

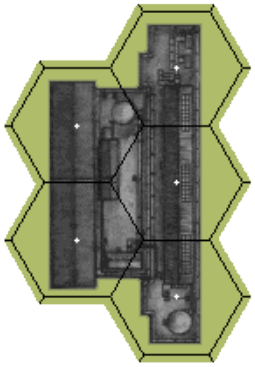

x18


x19

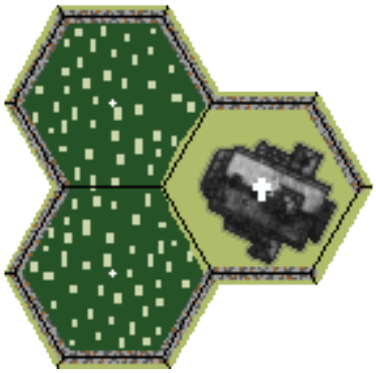

x20 (rotated 90 degrees clockwise)


x22

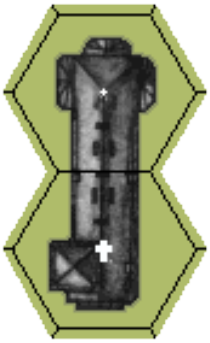

x21 (rotated 90 degrees clockwise)


x23


x24


x25


x26

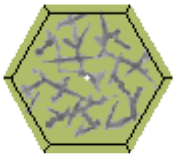

x27

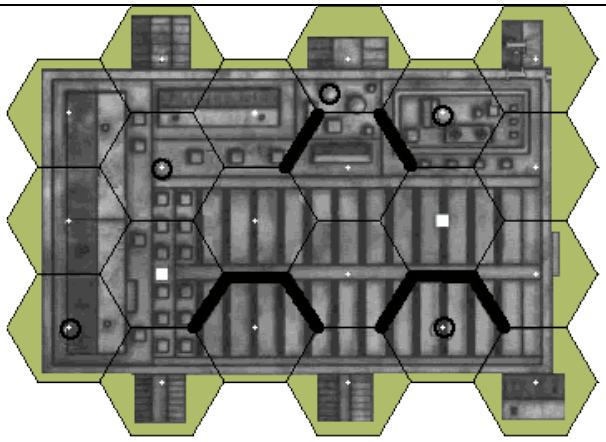

x28

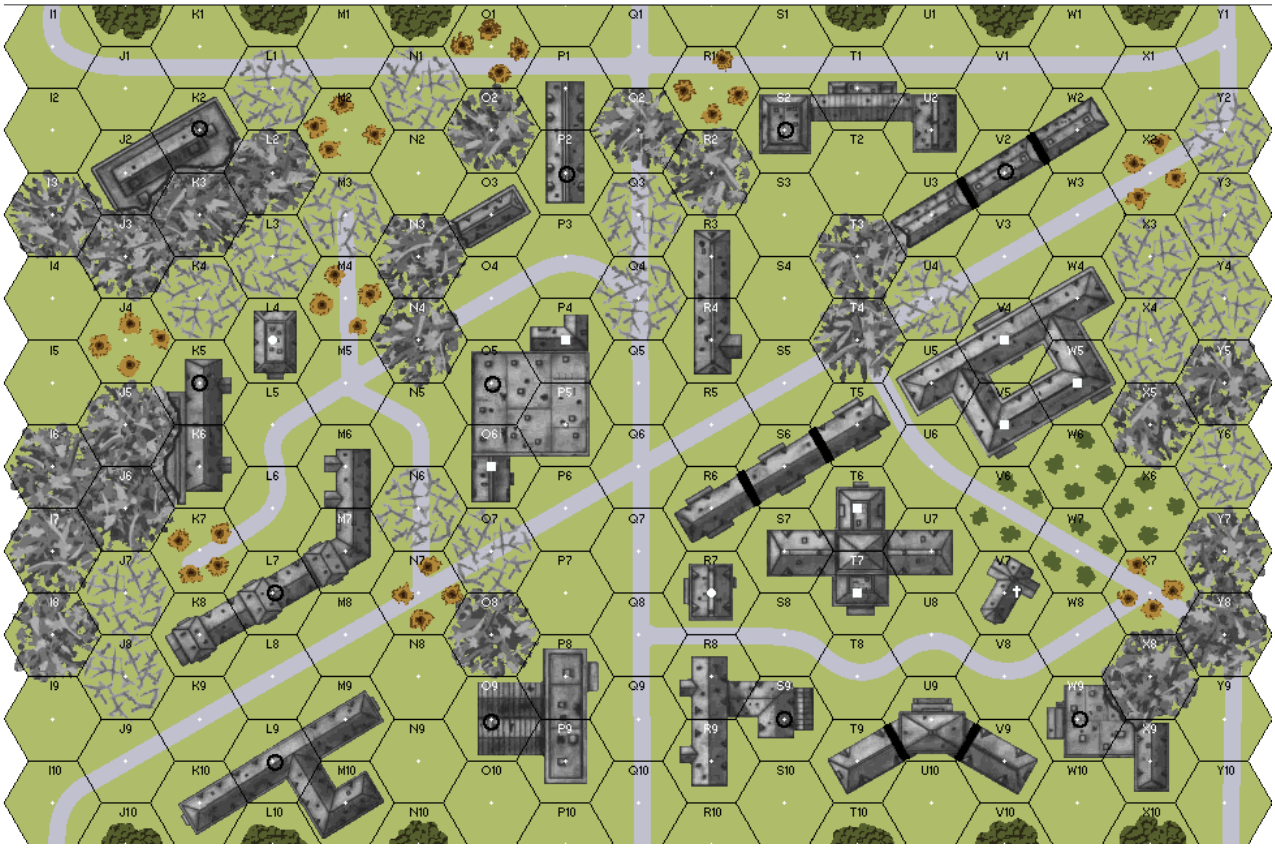

x29

F.14 Third-Party Overlays

F.14.1 Bounding Fire Productions


bfpd1


bfpf1



bfprc1

Ye Old Booke of VASL Secrets

Appendix F - Historical Note: The Perl/Postscript Map Generator

You can't know where you're going unless you know where you've been. Or something like that.

Has VASL always had beautiful hand-drawn maps, Unca Bill? No, Junior, there once was a time when the only maps VASL used were ones like you see at right. Computer-generated maps. They were good for their time, but now they're Old Technology - the hand-drawn maps are far and away much nicer to play VASL with, and now that VASL can do terrain changes, there's really no reason to use the computer-drawn maps.

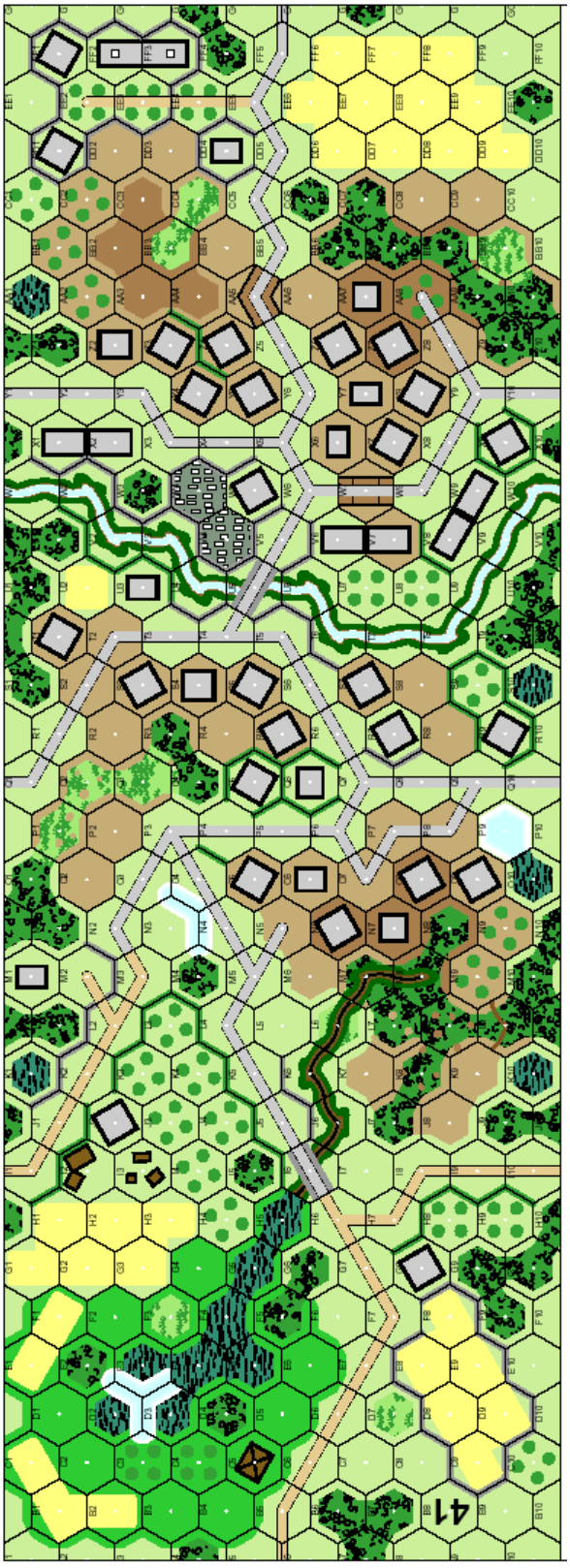
The map at right is a bitmap of a postscript file which was written by a perl script. The input to the perl script was a short file that looked like this:

```
board 41.map
print all
```

Various other commands could be inserted to do nifty things like remove grain, change green orchards to grey for out-of-season, and do global transformations like "all Woods are Brush". Similar stuff to what VASL can do now, but inherently limited by the graphics abilities of postscript (and the postscript abilities of the Cabal). And ultimately far more difficult to teach others than the hand-drawn maps are; Cabaleers are a rare enough breed without further winnowing down the pool by requiring them to know perl and postscript.

What the perl code would do is read the commands in the input file and then call up the appropriate map description files from its library. Those files contained hex-by-hex descriptions of the board, including the types of terrain in the hex and whether any terrain crossed over into adjacent hexes. This told the perl script *what* to draw in the hex; it then referred to its own subroutines to see *how* to draw those types of terrain. The perl script wrote its output in postscript form, and as you can see from the board at right, heavy use was made of the postscript graphics primitives. All of the terrain is simple shapes, lines, and fills, including things like the woods, brush, and marsh patterns, which put down the correct background color and then randomly put down the terrain indicators in the right colors. Primitive compared to today's standards, but Pretty Darn Tricky in its own right.

Giving credit where credit is due, the perl/postscript map generator was originally written by Eric Young and was further refined by Patrik Manlig and Klas Malmstrom. Salute!



Postscript board 41