TRACK EDITOR TUTORIAL FOR GENERALLY

Written by **TuomoH**

Version 1.00 (June 16th, 2020)

Table of Contents

INTRODUCTION

In this tutorial, I will go over techniques of making a GeneRally (GR) track using only the Track Editor (TE) that comes with the game. However, lots of these ideas also apply to tracks that also use external editors since many things including object placement and AI lines are made the very same way.

Everything is based on my experience and opinions on what makes a track good or fun. Some of the tips are common sense but others are influenced by my personal bias. There's not just one way to make a good track! I think that the best results come if you **make tracks for yourself**. Of course, it's nice when others like them too but you shouldn't follow trends just to get more downloads. If the track is fun for you, the chances are that it's fun for someone else as well.

About this tutorial

I'll walk you through all of the TE functionalities and explain how they can be used to make a nice track. In the text, all of the menu choices and tools are *italicized* for your convenience. Because an example is the best teacher, I'm making a track at the same time, and refer to this through the text. I won't explain every single decision I make but there will be lots of screenshots to help you understand what I'm talking about.

The tutorial follows the steps I usually take when I create a track. However, as some decisions you make earlier may make your life more difficult later, at some places I suggest you to read a later section before continuing. For the most part, I'm talking about TE but sometimes you need to check things in GR. At that point you should save the current version of the track and switch to the game to test the track.

I've tried to write this tutorial so that it can be used by a total rookie but there might be some tips even for veterans! I've highlighted several things:

- **Bolded words and sentences** are the key things for making a good track, in my opinion.
- Underlined red words and sentences are warnings and things I don't think you should do.
- Green words and sentences underlined with dotted line are tips, tricks and pieces of information that you might not know even if you've made lots of track. Some of these tricks I've discovered myself, many others I've found out thanks to the community.

My example track (I'll call it our project here and there) is a tarmac circuit. However, it doesn't really matter if you want to make something else, such as a street course, or perhaps a rally track – the basics are the same. You can find the example track in the same zip as this tutorial.

General(ly) advice

The biggest advice I can give is based on two p's: **practice and patience**. Practice is self-evident – you can't become good at anything without practicing a lot. The patience is needed in making the best possible track. It may happen that you get tired of your track so you want to rush it. However, if you want to make a good track, rushing doesn't help. Maybe start a new track and come back to this track next week.

Furthermore, you should **not be afraid to try different things**. Some things will work while some others won't. Don't worry, just take the ones that work! It's the same with this tutorial: take what you think is useful and improvise around the rest! You know what works and what doesn't by playing the game. **Test your track a lot!** I personally do it often; I add some things and take a lap or two. When I'm finished with the track, I've driven much already. Naturally one needs to be careful – you may become too used to the track for other to enjoy it. Also remember this: **if you see any problems at any time, fix them immediately**!

Finally, I think that for the track to be fun, it's more important that it drives well than looks good. When I play tracks from the early days of GR, I'm more drawn into tracks that are a blast to drive even if they are

ugly by today's terms. Therefore you should pay attention to how it feels. However, if you can make the track also visually appealing, why not do that as well? Alright, let's go!

LET'S MAKE A TRACK!

Go on, start TE if you already haven't.

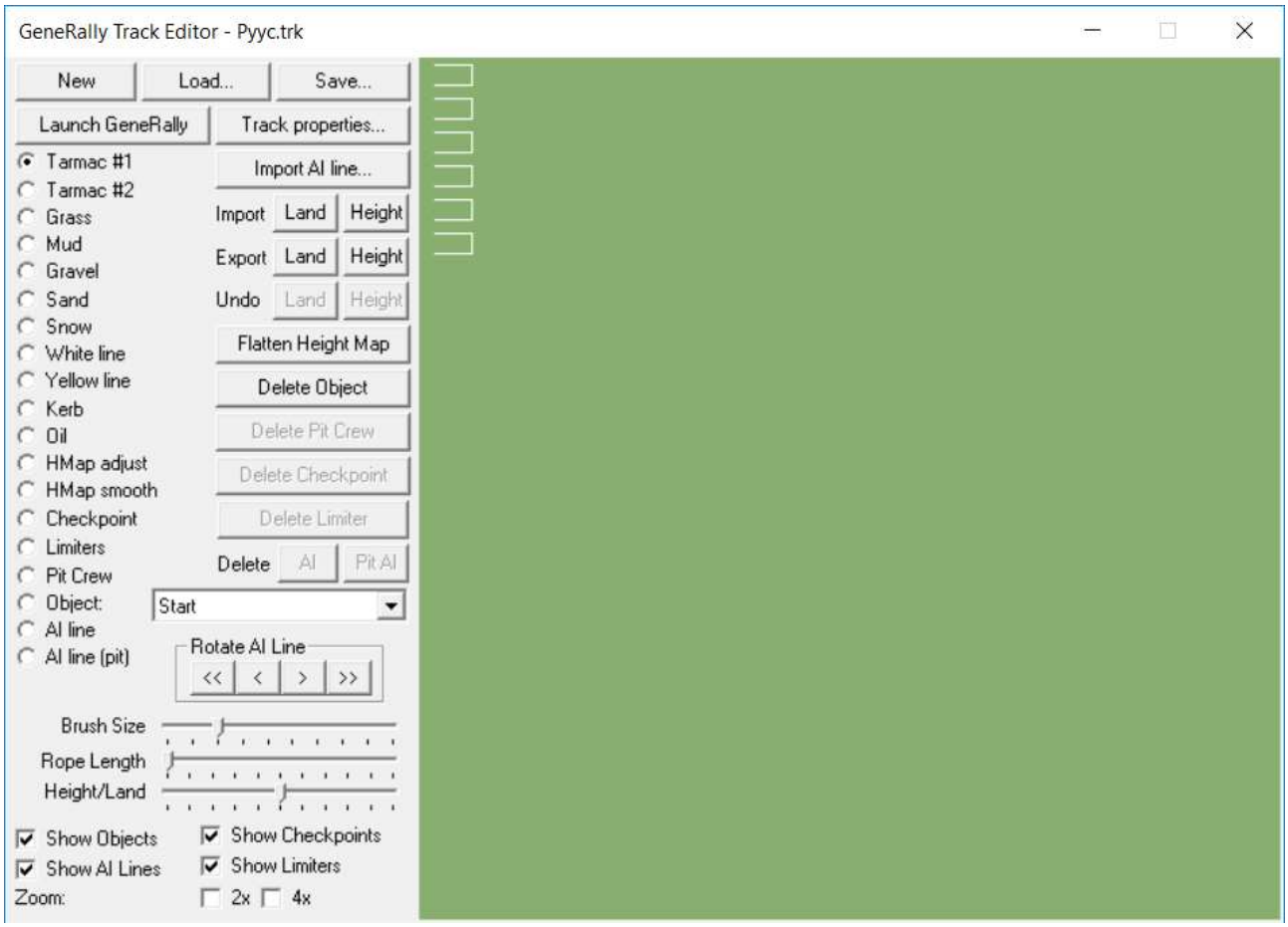

Figure 1. Track Editor.

TE opens with an empty map filled with *Grass* on the right (Figure 1). The map area is where you do all the work: paint roads, make height changes, add objects etc. There are already some objects, more precisely the 6 *Start* positions. On the left side you have all the tools. I'll go over the tools as they become important in our project.

The first four buttons are pretty self-evident. With *New* you start a new project. We don't need to do this now as we already have a new project open but if you click it, TE replaces any progress with an empty map and opens the *Track properties* window. More about that window later. Note that this is the most dangerous button in TE as it doesn't ask you if you want to start a new track. It means that you may accidentally delete everything if you haven't saved the track!

Next, we have the *Load* and *Save* buttons. TE and GR use .trk format for the tracks and with these buttons you can load/save these track files. In fact, save the track at this point! The track already has a name that's been randomly selected at startup. As you can see from my example, my track is called Pyyc. You don't have to stick with the random name but if you have no idea for a name, you can think of the final name

later. As usual with pretty much any project, **save often**. However, be aware that with undo buttons there's only one undo you can do. Therefore you may want to **save an extra copy when you're in a critical place**! I'll remind you of this at the most critical moments.

The fourth button is *Launch GeneRally*. As long as you have GR in the same folder as TE, it should start the game. You should actually click this button right now since we'll be going back and forth a lot to check if everything works. I like to have GR windowed using a relatively small resolution so that I can have both **TE and GR open next to each other**. In GR, you can change the resolution and turn full screen off under Options > Advanced.

World size

If you have played GR (of course you have) you know that there are tracks of different world sizes (WS) – some are small and others are large. In TE, you can select a WS between 25 and 255. This is actually meters, and means the side of the map: with WS100 you have a 100*100 meters area to fill.

You should think about the WS already now, at least in general terms. Is it around 100, around 150, or maybe around 200? If your idea is to have a long track with many corners, then you need to have a large WS. You can change the WS later but remember that while changing WS doesn't touch land map (surfaces, called lmap) or height map (terrain forms, called hmap), the objects you have planted earlier will not be nicely placed any longer. Furthermore, even though changing WS doesn't touch hmap, it affects the way hmap works: with a smaller WS the hmap is more pronounced. If you want to have a wild ride with sudden hmap changes, you have to use a small WS.

In my opinion, GeneRally works best around WS 100-150. I'm now thinking of a circuit at WS100. You can change the *World size* under *Track properties* (Figure 2). If you can't get the right value with your mouse, you can use arrow keys when the slider is activated. Also write your name or username to the *Author* field so everybody can see who made this track when it's finished! We see that there are some other options too but let's not change anything else yet.

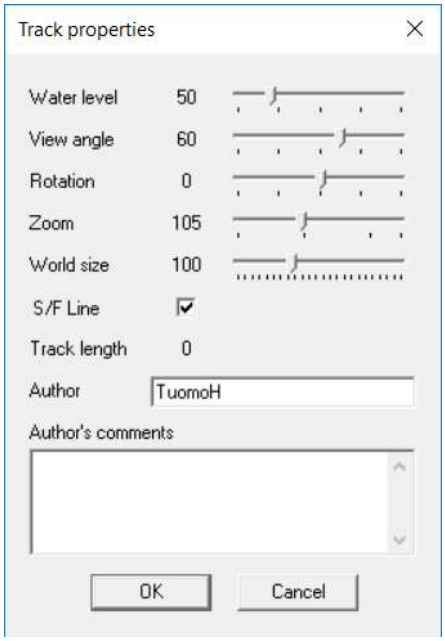

Figure 2. Track properties *window.*

LANDMAP

Most of the time track-making starts with the layout. Of course, there are some exceptions, such as when you want your track to be based on the shapes of the terrain, or when you want to go experimental. **Think beforehand what kind of a layout you want.** Also think what makes the layout fun. I think this track should have both a fast part and a technical part. It's not so much fun to drive fast to a hairpin, then fast to the next hairpin. I also don't like tracks that are basically turn after turn after turn without time to breathe and enjoy speed. Tracks with only rectangular shapes are often boring for me as well. But maybe you think otherwise! That's fine too. No matter what kind of layout you're thinking about, don't make the track go too close to the world borders. It's nicer to work – and race – if you have some space.

I'm now thinking of a kind of a heart-shaped track. It's got a bit curved main straight, a hairpin, a right, a left, and back to the main straight through a long left-hander. Nothing too fancy.

While you are thinking of the layout, you might also **think about the height changes** at the same time. More precisely, make sure that there are not too drastic changes in hmap in relation to your layout since those are difficult to do properly in TE. I know my track is not going to be flat but I'll do the decisions a bit later since my track idea is not based on height changes.

Surfaces

Before drawing the layout, I'll go over the surfaces than can be chosen. They are the top 11 radio buttons under *Launch GeneRally*. First you see there are two types of tarmac. *Tarmac #1* is a bit darker and gives you lines at the track edge if you use the left mouse button (LMB). Notice that many tools have different functionalities under the left and right mouse button (RMB)! If you draw *Tarmac #1* with RMB, you get yellow lines. *Tarmac #2* is lighter and you don't get any lines at the track edge. *Tarmac #1* is also a special surface in the sense that if you draw with any other "earth" surface (*Tarmac #2*, *Grass*, *Mud*, *Gravel*, and *Sand*) using RMB, you can't draw on top of *Tarmac #1*. I think that's a neat feature since it makes it easy to change the track-side surface without the fear of ruining the layout.

Below the tarmacs there are the other "earth" surfaces; *Grass*, *Mud*, *Gravel*, and *Sand*. Each surface affects the cars in different way. This naturally depends on the car – Formula is fast on tarmac but lousy on sand. Next we've got *Snow*. Notice that with RMB you get ice! Then there are two types of painted lines, *White* and *Yellow*. If you use RMB with them, you get a dotted line. This feature can be used to draw road markings. Next surface is *Kerb*, which gives you alternating white and red patches with LMB. With RMB *Kerbs* are yellow and red. Notice that the lines and the *Kerbs* are special in the same way as *Tarmac #1* – the "earth" surfaces painted with RMB can't destroy them. Finally, there's *Oil*. *Oil* is also a bit of a special surface – you can't draw a continuous line with it, and you can only draw *Oil* on either tarmac. Then there's this curiosity that if you draw *Oil* on an area with hmap changes, the *Oil* follows the shapes! Of course, *Oil* is a slippery surface, at least with the basic cars.

When you paint with different surfaces on other surfaces, you notice that some combinations give an edge that is a type of dust. For instance, drawing *Tarmac #1* on *Grass* gives you that edge. Once you become used to these combinations, you know how one surface affects the other. Remember to be careful at the edges of one surface! You don't want to spoil a nice patch with the other surface creeping in. Another thing you need to know about the edges of the surfaces is that some surfaces such as gravel spill to the surrounding areas. In other words, we get speckles of *Gravel* on *Tarmac* at the edge of these two surfaces.

Brush size

Before I draw the layout, I **test which *Brush Size* is the best**. That's the first tool with a slider, under the radio buttons and the arrows. Go on, test the slider at different positions and draw some lines on the grass. You see how the line width changes. Also notice that the *Brush Size* depends on the surface. For example,

with *White line*, the 1st tick gives a one pixel wide line but the *Mud* line is wider. My method for deciding the correct *Brush Size* is to **draw a short straight with *Tarmac #1* and then position *Start* objects on top of it to see how wide the track is** (see Figure 3). You can place a *Start* object by clicking on *Object,* making sure the dropdown menu is showing *Start,* and clicking on top of the *Tarmac* you just drew with LMB. If you need to rotate the *Start*, use RMB. I like that the track is not that wide but there should always be room for two cars to fight side by side safely. With WS100, I'll choose the 6th tick in *Brush Size*. With this I can occasionally get 3-wide fights. You may want to have a wider track but I advise you not to go narrower. 5th tick may still be possible but below that I don't think it's fun anymore.

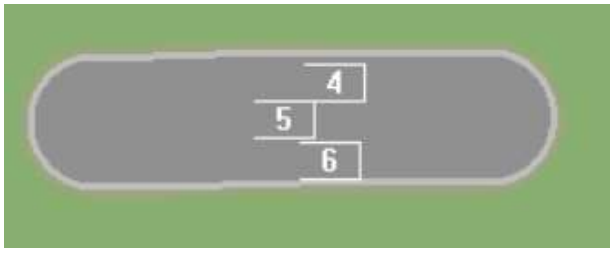

Figure 3. Testing the correct Brush Size *with the help of* Start *objects.*

Rope length

Next, it's wise to test **which *Rope Length* you want to use**. This is a tool to get smoother lines. You can find it below the *Brush Size*. The default value is the 1st tick, and this is the least smooth line you get – it draws even the tiniest movements your hands make. That's not good for a beautiful track layout. The further to the right you move the *Rope length*, the longer you have to drag for the line to follow you. The line gets smoother – try to do small zig-zag movements while drawing – but at some point the *Rope length* is too big to get an interesting layout. I personally usually use the 3rd or 4th tick. In Figure 4, the corner on the left is drawn with the 1st tick, the corner on the right with the 5th tick. For both, I made similar hand movements.

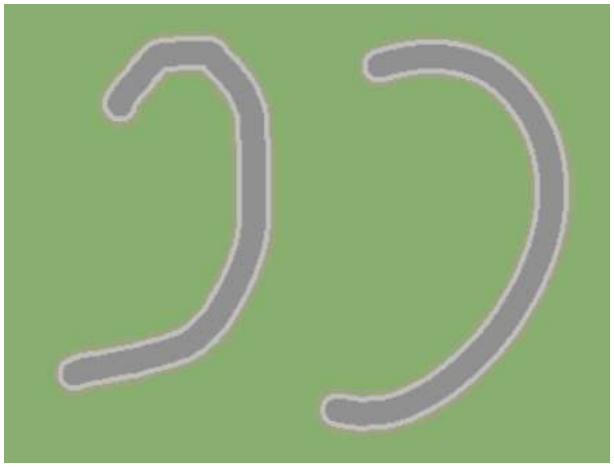

Figure 4. A corner drawn using Rope Length *1 (left) and* Rope Length *5 (right).*

Drawing the layout

Now we can draw the layout. First, **get rid of the test lines you've done**. You can start a new project if you want to, but then you have change the WS again. I often take *Grass*, set *Brush Size* to maximum and *Rope Length* to minimum and just erase everything there was. Then I choose *Tarmac #1* again and set *Brush Size* and *Rope Length* to the desired values. Then we can **draw the layout**.

Remember that you don't need to get the layout right the first time since there is the *Undo Land* button. Too bad it's only got one undo! For my TE tracks, I have drawn each line many times: after each badly drawn line, I've clicked *Undo Land* and repeated the line until I've gotten it the way I want.

When I'm making a circuit, I try to draw the main layout with one continuous movement since, in my opinion, it's difficult to continue from exactly the correct place – if my cursor is not 100% in the middle of the existing road, I get "bubbles" on one side (Figure 5). Furthermore, it's difficult to continue exactly to the correct direction so the side of the road is not smooth. Said that, you can try to **fix those errors by using a smaller *Brush size* and carefully adding *Tarmac*** so that the side line becomes smooth again (Figure 5). However, it's not as nice as one continuous line. In fact, this kind of fixing often leads to adding more and more tarmac to the problem area. **Remember *Undo Land*!**

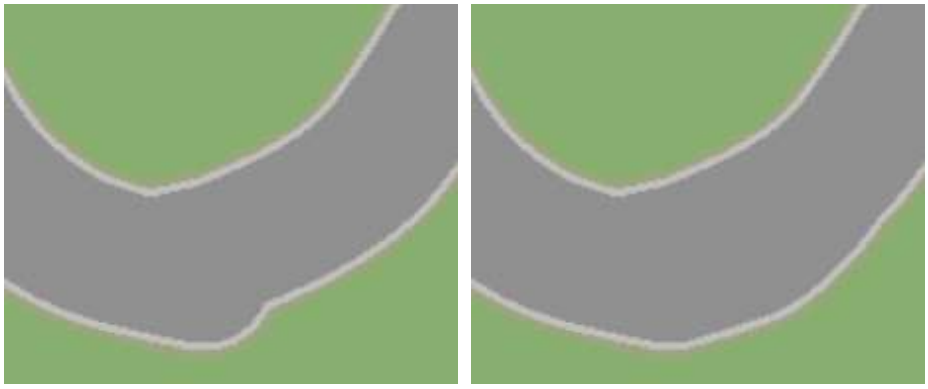

Figure 5. Left panel: A corner made with two strokes – notice the "bubble" at bottom. Right panel: The same corner fixed with by adding more Tarmac*.*

Once you've drawn the layout, **place the six *Start* positions along the main straight**. Notice that the last one you've clicked is always number 6. Don't worry, just place the rest 5 of them and the numbers will rotate so that the one you placed first becomes number 1. Then save the track and try driving it a bit. **If you see any problems with the layout, now it's easy to fix them.** In Figure 6, you can see what I've done – it took me maybe 40 attempts to get that continuous line good enough for my liking.

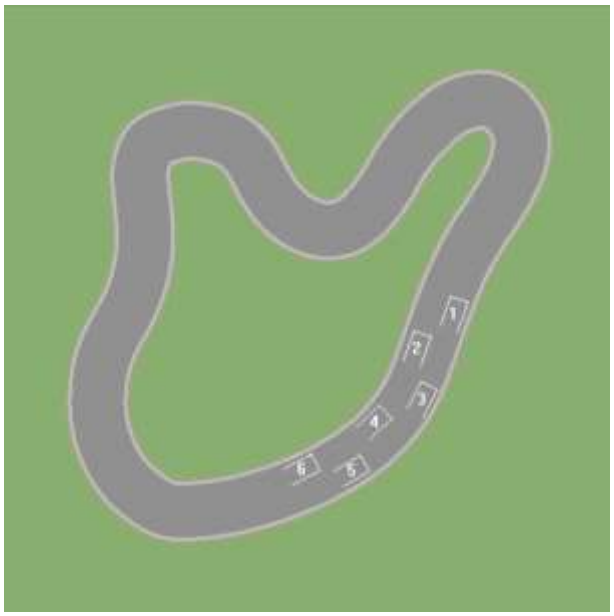

Figure 6. The example layout drawn with one continuous stroke, with Start *positions on the main straight.*

THE PITS

At this point, it's wise to **think about the pit placement!** Too many times I've done a nice track and only at some later point realize that there's no proper place for the pits. If there's no *Tarmac* to place the pits, I either draw a pit lane next to the main straight with *Tarmac #2* (**using RMB**), or a separate road. Think of the road placement! It's not cool to have a pit lane that can be used to shortcut! Another tip is to draw the pit lane so that the drivers enter the pits from the left side. This is because the *Pit crews* have one person less on the left side, giving more space for smoother pit action. I actually thought about this already, and that's why I left some room on the bottom right of my map.

I want to have a separate pit road with a narrower entry and exit. Because of this, I first draw a driving line using *Brush Size* 3. Then I draw a wider part for the *Pit crews* using *Brush Size* 5. Both steps can be seen in Figure 7.

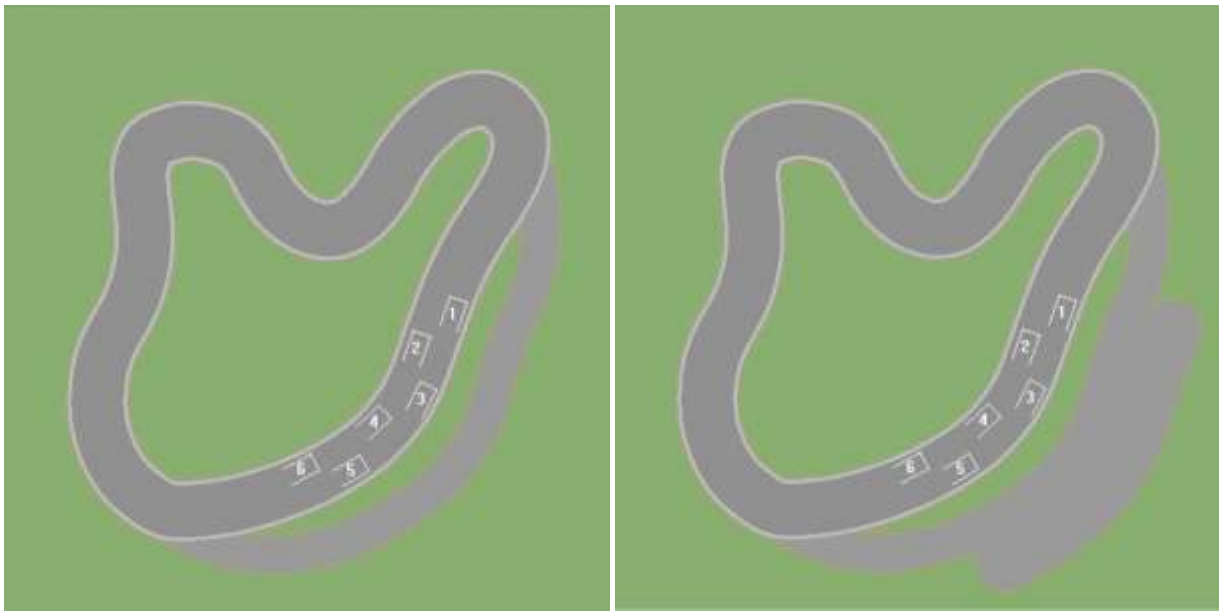

Figure 7. Left panel: Pit road added. Right panel: Area for the Pit Crews *added.*

I finalize the transition from narrow to wide with *Brush Size* 2 using **the *Zoom* tool**. You can find this tool at the bottom where you can select either a *2x* or *4x Zoom*. I use *Zoom* a lot. The transition and the *Zoom* feature can be seen in Figure 8.

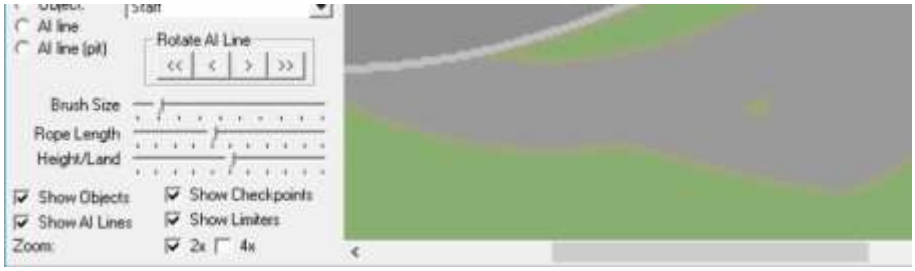

Figure 8. The transition from pit lane to Pit Crew *area drawn with the help of* Zoom 2x.

For **placing the pits**, you have the *Pit Crew* tool. You can rotate the *Pit crew* with RMB, then place the crew with LMB. There are 6 *Pit crews* that can be added. A bit annoying thing is that the *Delete Pit Crew* button deletes them in the reverse order you've put them. In other words, if you later notice that the first *Pit crew* you placed is in a wrong place, you have to delete all of them to fix the first one.

Even though there are 6 *Pit crews*, you don't need to use all of them. Usually, player's game settings are such that all AI guys don't pit at once. Therefore, 4 is quite often enough. Of course, if you can fit all the *Pit*

crews nicely, do it. There are several principles for placing the *Pit crews* optimally. First, make sure they are easily accessible. This means that **the *Pit crew* opening should face the opposite of the driving direction** (see Figure 9). Second, think of the placement in relation to the driving line so that **the AI doesn't need to do sudden movements** to get there. Third, make sure that you **don't hit the pit guys on your regular racing line**. Fourth, the ***Pit crews* should be a bit separated** (unless you use clustered pits) so that there's room to get to the pit and from the pit safely (again, see Figure 9). If there are only 3 pits or less on the track, I use double pits. For this I overlap two (or even three) *Pit crews* perfectly. This way the pitting is faster and you can get away with 3 pits.

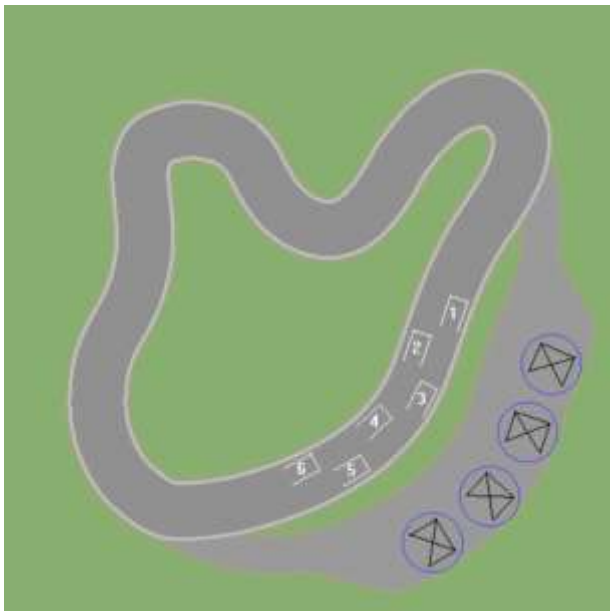

Figure 9. Pit Crews *added.*

KERBS

If you want to have *Kerbs* (some people write that Curbs) inside the corners (or outside the corner exits), now it's a good time to draw them in. Many people don't like the TE *Kerbs* but I think they are alright, as long as you draw them carefully. For this, I **save a copy of the track** so I can go back to the current one if I don't get the *Kerbs* correct right away. I'll call it PyycB. Then I **draw a small piece of *Kerb* next to the track and test it in GR** to see if it's wide enough (but not too wide). *Brush Size* 5 seems to work very well here. In Figure 10, you can see what I think is approximately the correct width in relation to the car.

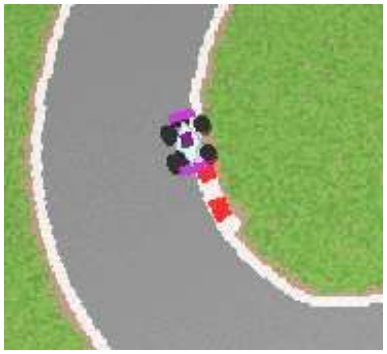

Figure 10. Testing the correct Kerb *width in GR by driving on top of it.*

Then I undo the practice *Kerb* I made and **draw *Kerbs* one by one**. For this, I use *Zoom x4* and *Rope Length* 1. I make very slow movements and am ready to *Undo Land*. When I'm really concentrating, sometimes my RMB finger accidentally clicks and then even the *Undo Land* can't save me. That's when the extra save comes in handy. I personally want my *Kerbs* to be as symmetric as possible, so they start and end with white. I also don't want the *Kerbs* to show white line between the *Tarmac* and *Kerb*, or overlap the *Tarmac* (see Figure 11). Every time I get one piece of *Kerb* finished, I save the track.

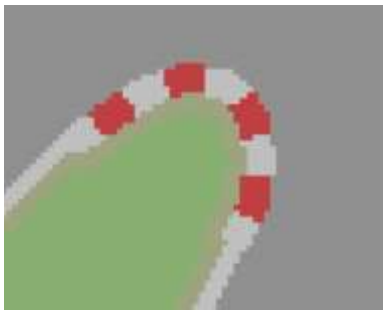

Figure 11. A perfectly drawn Kerb*.*

For the *Kerb* at the outside of the main straight entry I decide to use *Brush Size* 6. Yes, now the layout is ready (Figure 12)!

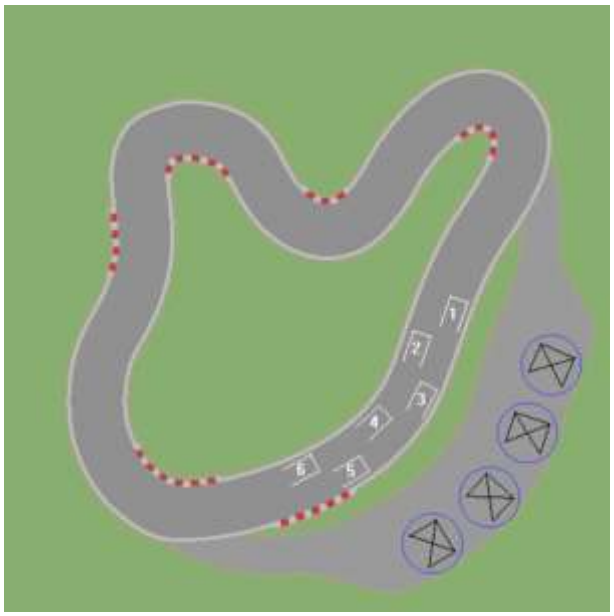

Figure 12. Finished layout with pits and kerbs.

HEIGHT MAP

Once the main layout is ready, *Pit crews* in their place, and the possible *Kerbs* drawn, it's time to **think about the height changes**. This is actually very important since a totally flat track is usually not so fun but hmap changes in carefully thought places can turn even a basic layout into much fun. As with the layout, you should think about hmap before drawing it. Notice that if your track includes roads at different levels next to each other, it's quite difficult to get the hmap right in TE. For our project, the area in the upper part is going to be higher, then there's a downhill towards the main straight.

When thinking about the hmap, you should also consider how the AI will succeed with the hmap. Therefore, I suggest you **read the section AI and hmap now**. Also, **if you're making a track that has a figure**

of 8 with a *Bridge* going over the road, you should now check the section Bridges and hmap now. Even though I don't talk about objects just yet, if a *Bridge* is needed for you to complete a lap, you should place it as soon as possible.

In my opinion, TE is actually a very good tool to make most types of hmap. The only things it can't make at all are very precise sharp edges and angles you see in some stunt tracks, for example. Also, it is next to impossible to do smooth gradients. Those I leave to external editors. Apart from that, I use TE for my hmap work 95% of the time even with non-TE tracks.

Hmap technicalities

For making the hmap, your tool is *HMap adjust*. With this tool, **LMB raises the land and RMB lowers**. Technically, the terrain height is represented in grey scale. Black (RGB 0,0,0) is the lowest point there can be and white (RGB 255, 255, 255) is the highest point. When you start a new track, the terrain is flat and at the midpoint.

The numbers 0-255 actually correspond to real world as well: 1 point means 10 cm. However, the way GR handles the hmap, this doesn't make much sense so you probably don't need to think about it. This is partly due to the hmap being less accurate than lmap: lmap uses a 512x512 pixel bitmap image whereas hmap is only 64x64 pixels. This means that you can't make very detailed hmap work in relation to lmap even outside TE.

If you slide the *Height/Land* slider to the very left, you only see the hmap with the shades of grey. Now it's pure grey. The ***Height/Land* slider is actually a handy tool** since sometimes you want to see the lmap details you're working on better. Then you just move the slider to the right. At other times you want to see hmap well. For now, move the slider back to the midpoint (6th tick).

Drawing hmap

First, **save a copy of your track** – you already guessed that, right? Then it's time to start painting. With hmap, *Rope Length* doesn't work but the *Brush Size* is the same as for surfaces. **First draw with a big brush to make the overall terrain.** Then move to **smaller brushes for different parts of the track.** Finally add the details such as bumps on the road. Notice that you need to move the mouse in order to make any bigger changes. A single click does change the hmap but if you don't move the mouse, it doesn't matter if you keep the button pressed. When making steep changes, you need to circle or saw with the mouse. Whatever you do, remember to take it easy. Check your results in GR often and save a copy of your track when trying to do something critical. There's also an undo button for hmap, *Undo Height*. However, as with lmap, it's only got one undo. Because it's so easy to overdo the hmap, I always try to make sure that when finetuning, I only click once before checking the result in GR.

Problems with hmap

If you want to make things smoother, **there is a tool called *HMap smooth***. What it does, it makes the area under the brush flat. It's alright when trying to get rid of small bumps but when you're doing a hill, for instance, it just ruins the hill. That's the reason I don't like *HMap smooth* in TE much.

Try to avoid having sudden hmap changes on the road. There are two reasons for this. First, unless you're making an offroad track, it's not fun for the player when the hmap suddenly throws you in a random direction. Player should be able to predict where the track goes. Second, sudden hmap changes are not pretty to look at.

Be careful not to go too high or too low. When you reach 0 or 255, you get a plateau at your track (Figure 13). If it's a small area, it can be fixed relatively easily, but the bigger the area is, the more difficult it becomes.

Figure 13. Hmap raised to the maximum, creating a plateau.

If you feel that the hmap doesn't work at all, there's always the button *Flatten Height Map*. This button brings you back to the default hmap. However, this is the second dangerous button in TE! If you click it by accident, you have the *Height Undo* button but as I mentioned, there's only one undo. You can you're your whole hmap if you're not careful enough!

Water level

When you lower the hmap enough, something blue may spill into your track. That's water, and you can see it in Figure 14.

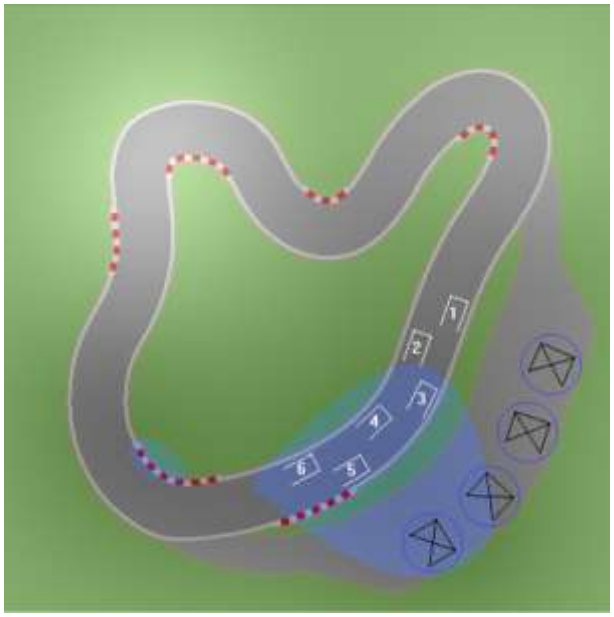

Figure 14. Lowering hmap enough brings water to the map.

TE has a *Water level* setting – everything below that level is under water. In other words, you can't have a mountain lake high up the mountain and dry land lower. The *Water level* can be adjusted under the *Track properties*. The default is 50, and it refers to the hmap points I already mentioned. Now that we're working with hmap, it's wise to **decide what the *Water level* should be.** Of course, you can change it at any point later too. I don't like tracks that have deep water on the track but a slight splash can be fun. Actually, if you set the water level just at the level of the racing surface, you may get nice sparkles even if you don't see the water. Since I got water on my track, I'll lower the *Water level*. I'll go with 30. Now I'm happy with my track hmap, and you can see it in Figure 15.

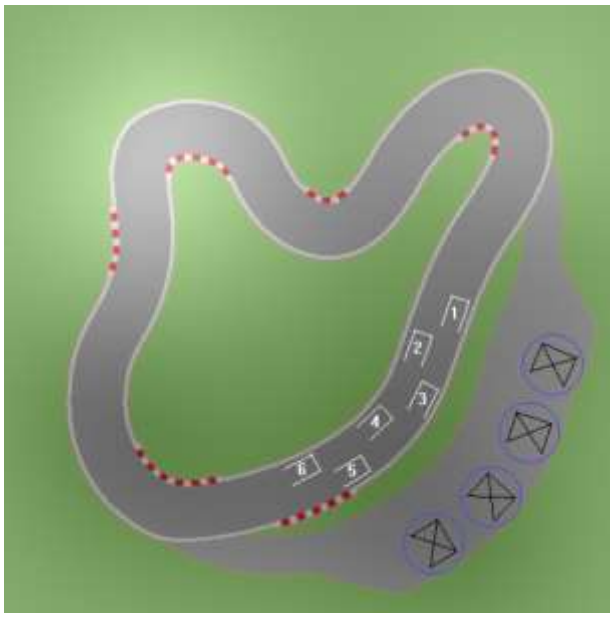

Figure 15. After the Water level *is set at 30, the track hmap looks like this. Brighter areas are higher, darker areas lower.*

Now that I'm talking about water, you may notice that the surface under water looks different than above water. You can use this to your advantage when making the track look prettier. Try drawing different surfaces under water! You might like some combination, and this can make your track look more alive.

Surrounding hmap

After you've made the hmap for the track area, think whether you want to add terrain outside the racing area. For sudden changes (e.g. cliff or mountains), use the smallest possible brush. I always use the *Brush Size* 1 to make the edges because then I avoid touching the areas which I want to be at a different level. **First make the changes further away from the road to see how the hmap works.** Make changes to one place at a time and check the result in GR, then do more changes. For our project, I want to have water in the left side. For that, I first draw a "river" with the smallest brush, far enough from the track (Figure 16).

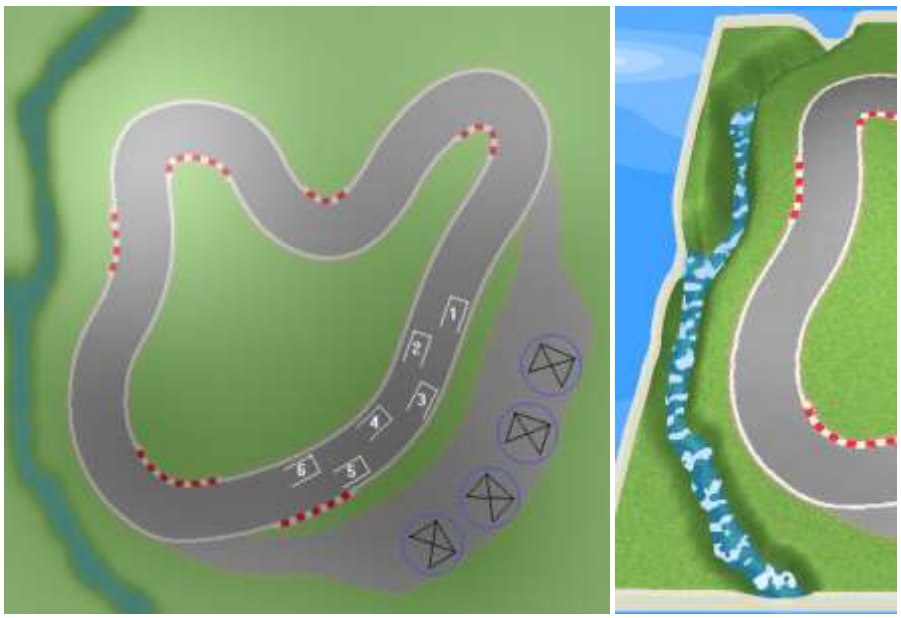

Figure 16. Left panel: Added a "river" to the hmap with the smallest brush. Right panel: checked the river in GR to ensure it doesn't interfere with the track.

It seems to work well so I edit a bit more: erase more land, leave some small islands, add just a bit of mountains above the top-left corner (Figure 17).

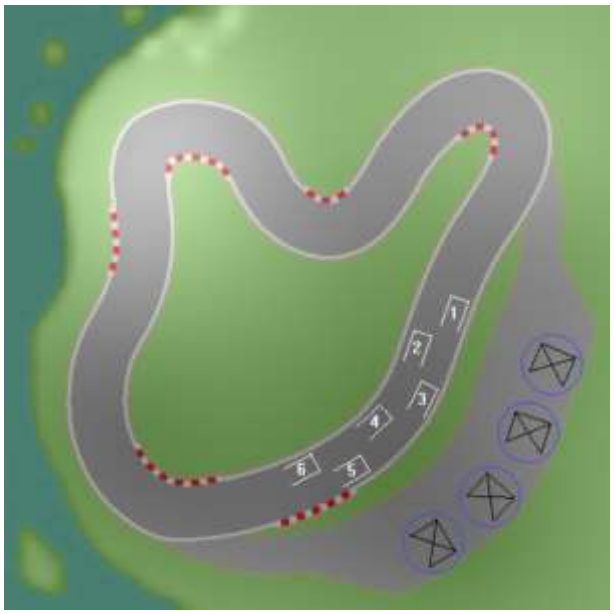

Figure 17. Did more hmap work for the water area to make it look more natural.

Using hmap to guide player

While I'm not a fan of sudden hmap changes on racing line, I think the hmap can be nicely used to guide the player. What I mean is that while the road should be relatively smooth, it's cool to have hills right next to the track. The inside of the apex can be easily marked with a small hill inside the corner. Just make sure the hill doesn't spill on the road! Player quickly learns to stay away from the terrain but it adds a lot to the visuals. I'll continue with those mountain spikes, carefully of course. The result can be seen in Figure 18.

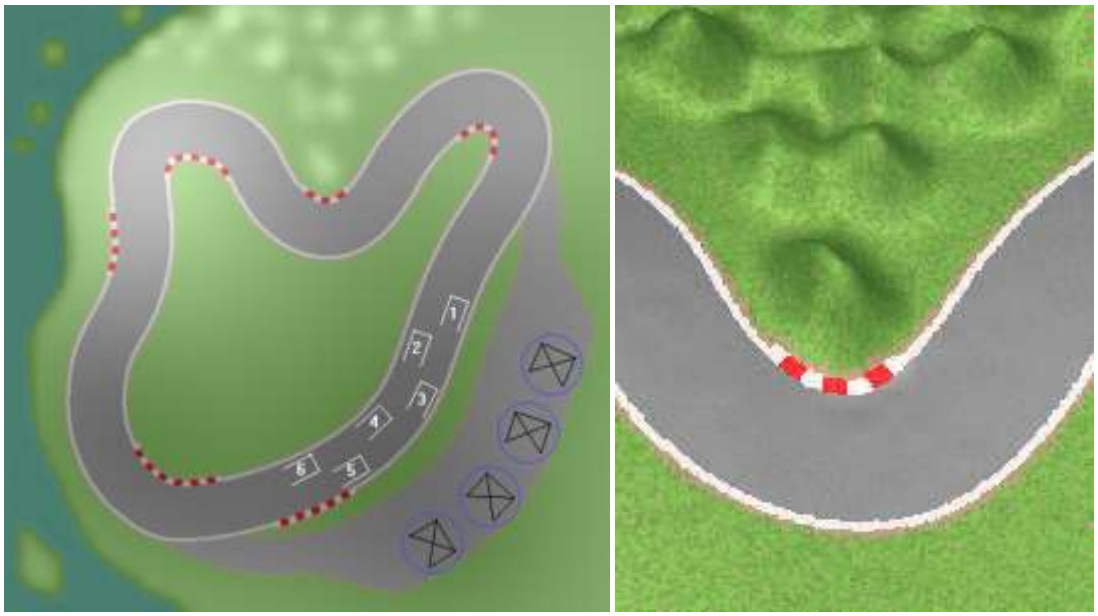

Figure 18. Left panel: Added hmap close to second corner. Right panel: Checked that the hill doesn't interfere with the track in GR.

Banking

One cool thing about GR is that the banking works very well. If you tilt a corner towards the apex, the cars sweep quickly around that corner, taking advantage of the hmap. It's a way to make a difficult part easier

plus it's fun to drive through. Banking is relatively easy to make in TE. In fact, if I don't want very precise banking or maybe multiple banked corners, I make them in TE. I choose *Brush Size* that is a tick smaller than the one I drew the track with, in this case the 5th tick. Then I position the brush near the apex and lower the hmap in the corner with just one small and quick circular motion. Again, *Undo Height* is your friend if you don't get it right the first time. I bank the hairpin.

In Figure 19, I've set the *Height/Land* slider to 1st and 2nd tick so you can see only the hmap and then some of the lmap (left and right panel, respectively). The circle that's darker than the surrounding area is the banking I made. As you can see, the center of the "hole" is quite close to the center of the apex.

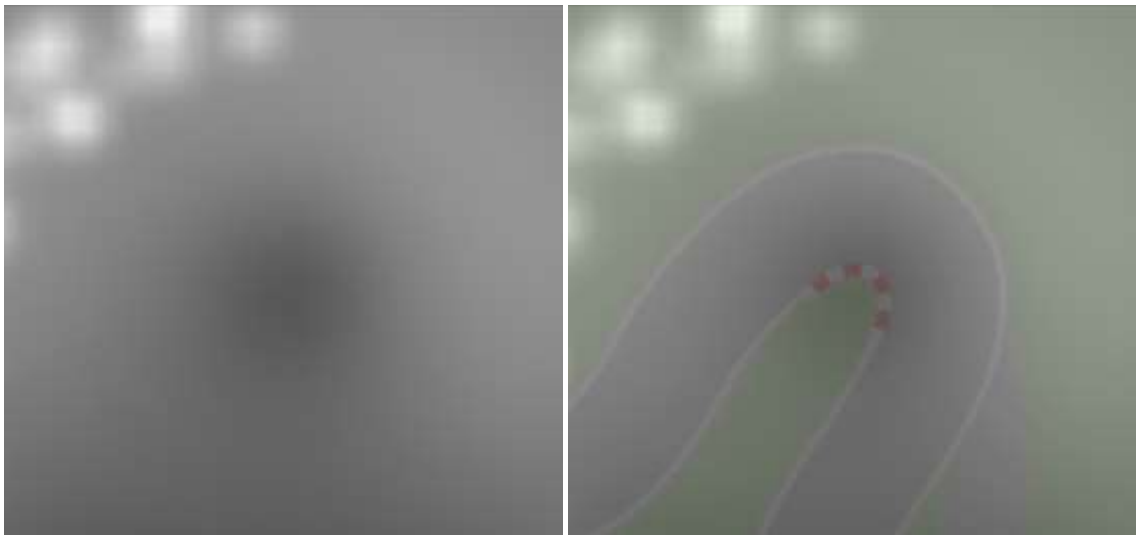

Figure 19. Made a banking to the hairpin. Left panel shows just the hmap, right panel adds a bit of lmap to the picture.

You shouldn't make too steep banking since then cars getting out of the corner may jump to unexpected places. Because of this, I often try to make the corner entry and exit smoother by carefully adjusting the hmap transition. Again, I don't think *HMap smooth* works for this. With our project, I don't think the banking is too steep so I'm not going to adjust it.

Hmap and lmap

After you've made the hmap, it's a good idea to go back to the lmap. Do you have something like hills or edges? You don't have to do anything if you don't want to but using another surface than *Grass* where the hmap changes can be a big deal to the visuals of your track. For instance, using *Gravel* at the edges makes the edge stand out. And if you've made a terrain that raises a lot above the ground, could it look more like rock? When I make small "mountains", I often raise the hmap quite lot and make several peaks in one area – that's what I just did! I then carefully color the raised areas with *Tarmac#2* (so that I don't get white lines). As the hmap and lmap don't line up perfectly in TE, I usually color a bit further than where the steep hmap changes seem to end. I then finalize the coloring by adding dots of gravel. **You can try different surface combinations yourself**, maybe you'll find something cool.

Here we have both a cliff and some mountains so I'm going to paint those areas. Near the track I'll use RMB since I can't erase *Kerb* or *Tarmac #1* with that. I also exaggerate the hmap changes for my mountains a bit more. I haven't explained the objects yet but I'll tell in advance that I'm going to place some *Stones* in the mountain area later to make it look more natural. I decide that the small island in the bottom left corner is nice with *Grass* but the little islands at the top are part of the mountains. Finally, I notice that *Gravel* looks better than *Grass* underwater so I paint the underwater area with *Gravel*. In Figure 20, I have set *Height/Land* slider at the 9th tick so you can see mostly lmap.

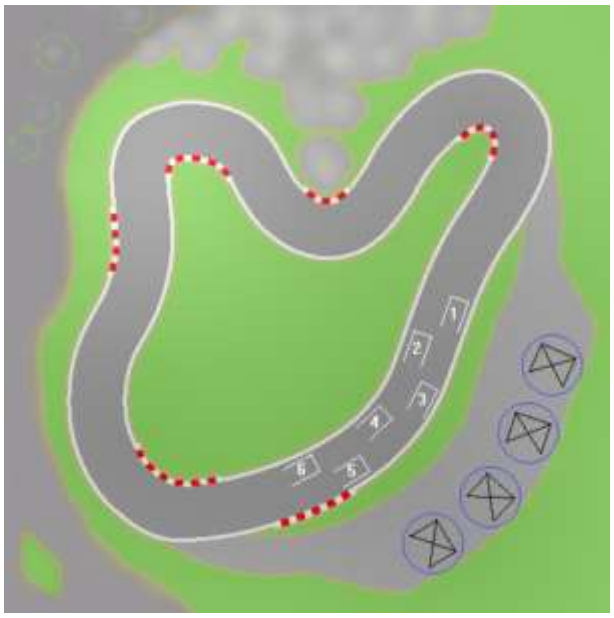

Figure 20. Added Tarmac#2 *to the mountain area and* Gravel *to the cliff edges and underwater.*

A PEEK AT TRACK PROPERTIES

Now we're pretty much done with the basic lmap and hmap so it might be a good time to **check the viewing options**. For this, click *Track properties*. Now we're going to edit values that take effect only when you go to GR. Luckily you can always change them back to what they were. First, let's take a look at the *View angle*. This means how high above the track you, as a player, will be. The *View angle* can be anything between 10 and 90. 10 means that you're almost at the level of the track, 90 means that you're looking at it directly from above. **GR handles the default view angle 60 the best but there's still room to try other angles comfortably**.

I personally like that the hmap can be seen. Therefore, I usually set the *View angle* lower than default. When I'm making the hmap, I often put the angle quite low, maybe even around 35-40 so I can really see the changes well but after I'm finished with the hmap, I'll find a better angle for racing. I'll set it now at 50.

You may also notice that you've made the track in a way that it'd be nicer to look at from a different *Rotation*. Try moving *Rotation* from 0. This means the direction from which you will look at the track. If we think of the compass, *Rotation* 0 means looking from South and *Rotation* 90 looking from East. The value can be anything from -180 to 180, and at those points you're looking at the track from North. You might get carried away with trying rotations that give cool views from interesting directions but the further you get from 0, +-90, or +-180, the further away from the track you go. As result, cars also get smaller. I'll come back to this point later. For our project, I'm happy with the default *Rotation*.

We also have the *Zoom* setting here, but for the time being, you can either leave it be or set to 100. I'll come back to this as well. For our project, I set the *Zoom* to 100. In Figure 21, you can see how the track looks so far.

Figure 21. Track after the basic lmap and hmap work has been done.

OBJECTS

Once I'm finished with the basic lmap and hmap, I move to the objects. As the track is drivable, I can already feel whether it's fun or not. The next step is making it pretty. However, if your track doesn't feel fun to drive at this point, adding objects isn't probably going to change that.

For me, placing the objects is the way to make the track alive. Whereas creating the lmap and especially hmap are primarily about getting the technical issues right, placing the objects is closer to art. In my opinion, object placement is even more important than making a fancy lmap, and this is true not only for TE tracks but most tracks! Naturally, tracks with excessive lmap art are an exception to this rule.

When you place the objects, you need to ask yourself a series of questions. Which object should go where? Do I need street *Lights* somewhere? Are those buildings too much or too little? Try to break the monotony and give your track a room to breathe. A common rookie mistake is to fill huge parts of the map with one type of objects, such as trees. This just makes the track look boring. At any rate, no matter what you do, **make it logical, at least in your track's universe. It doesn't need to look like real world if it works in the game.**

In TE, we have the *Object* dropdown menu. Choose the object you want, rotate it with RMB and place it with LMB. If you want to get rid of an object, select *Delete Object* and click on the object you don't want anymore. Notice that when you have many objects next to each other, you may accidentally delete the wrong ones. This happens particularly when the objects are on top of each other. When this happens, you need to place the wanted objects again. I also remind you of **the *Zoom* tool since it makes it much easier to place objects precisely**.

Object inventory

Let's go over the objects. For the most part, the order in this tutorial follows the order of the dropdown menu. However, I wanted to group all the wall-type object together so that's what I'll do here. I will also mention whether the objects are hard or soft. Hard means that you get damage when you hit the object while the soft objects don't have such effect.

First, there are several types of walls (Figure 22 shows them in GR). *Concrete wall*, *Invisible wall* and *Armco barrier* are hard walls. *Armco wall* is two sided in the sense that the side that should face the track has no vertical poles. **In TE, place it so that the orange side is facing the track**. *Invisible wall* extends to the sky and can be used to make sure the drivers don't drive outside of the track. I personally may use it in a tightly banked corner lining the outside corner to make sure that even a bad driver doesn't find themselves out in the woods. If you use the *Invisible wall*, **make sure that the driver can't get on the wrong side of the wall**! If you have hmap changes where the *Invisible wall* is, it may be possible to drive under the wall! Also, *Hedges* (normal and *short*) should be group into hard objects (maybe call it semi-hard), even though you don't get as much damage as from those others. TE seems to group *Hedges* as vegetation but I've always considered them as another type of wall.

Then there are soft "walls", namely *Soft wall*, *Hay bale*, and *Hedge* (normal and *short*). Many people don't like the *Soft wall* since it's bright red so you may want to use them sparingly. I think that when wisely used, it's a good object. For me, the problem with the *Soft wall* is that it sucks in the driver that hits it, killing all the speed. Because of this, think twice whether you want to use it right next to the racing line.

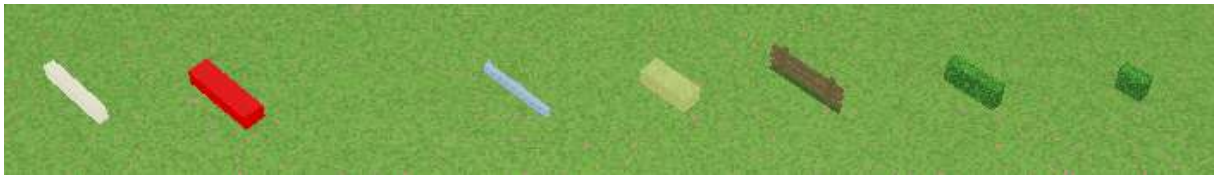

Figure 22. L-R: Concrete wall, Soft wall, Invisible wall, Armco barrier, Hay bale, Fence, Hedge, Hedge (short).

Vegetation (Figure 23) consists of four types of trees, *Fir tree*, *Leaf tree*, *Pine tree*, and *Palm tree*, as well as *Cactus* and *Bush*. The trees are (semi-)hard objects while *Cactus* and *Bush* are soft. Notice that you don't need to rotate the trees to get variation to their looks as GR plants them in a random order every time a track is run. Thanks to this feature, a neat trick for a fifth type of tree is to combine *Palm trees* by putting maybe 16 or 32 of them exactly at the same spot. For *Cactus*, you need to set the rotation yourself. In the right side of Figure 23 I have overlapped 24 palm trees.

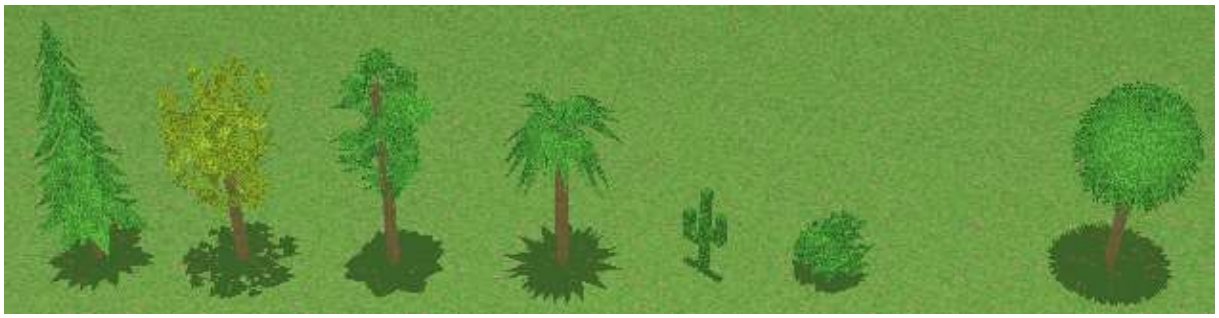

Figure 23. L-R: Fir tree, Leaf tree, Pine tree, Palm tree, Cactus, Bush, *24* Palm trees *on the same spot.*

Then there are four hard objects that are typical in cities (Figure 24, left side): *Traffic sign*, *Light*, *Telephone post*, and *Concrete post*. Notice that **the sign side of the *Traffic sign* faces the small line in the TE object**. This small sign also shows the rotation of the other objects in this and the next category.

Next objects are often used to denote track boundaries (Figure 24, right side): *Road cone* (normal and *movable*), *Stone*, and *Sunk tyre*. All of them are soft objects. The normal *Road cone* (red and white) stops you at where you are, the *movable Road cone* (blue and white) gets thrown around when you hit it. Notice that the *movable Road cones* don't move that naturally in the game and might act strangely with hmap changes. *Stone* and *Sunk tyre* slow speed and throw the car in the air. *Stones* can naturally be used as part of nature.

Figure 24. L-R: Traffic sign, Light, Telephone pole, Concrete post, Road cone, Road cone (movable), Stone, Sunk tyre.

Then we have different types of buildings (Figure 25), all of them hard objects. *Booth* is a small shack with a window on one side and a type of door on the opposite side. **In TE, the red side shows the window.** *House* is your regular house, and **the yellow side in TE is the front**. *Office blocks* are variations of a theme: The original *Office block* is the large one, *Office block (short)* is half the height of the original, and *Office block (small)* is ¼ of the original *Office block* in area and ¾ in height.

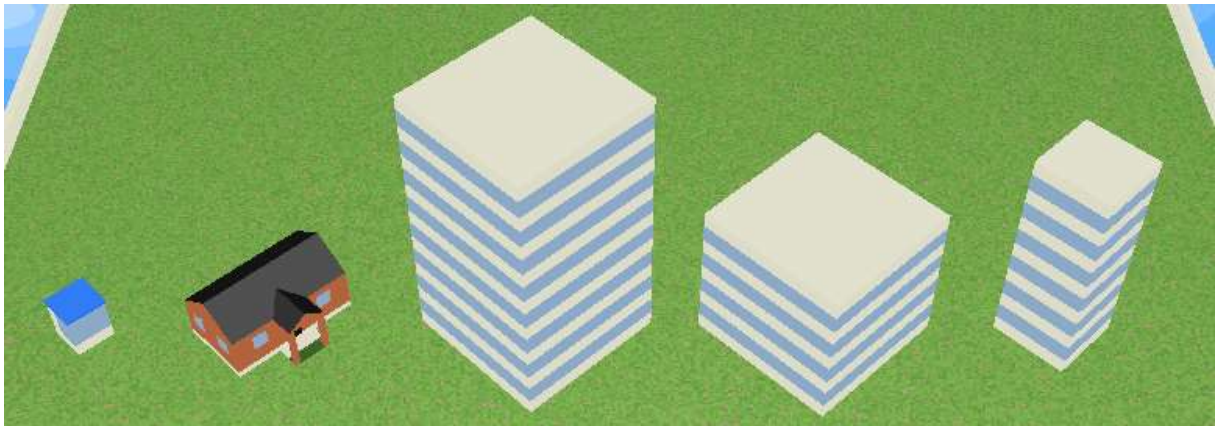

Figure 25. L-R: Booth, House, Office block, Office block (short), Office block (small).

Next there is a *Stand* for the spectators – have you noticed that the fans' shirt colors reflect the cars that are driving in the race? *Gate* is a… well, a gate. *Boat* is an object that responds to water level. If it's placed on land, it's just like any other object but when you place it on water, it always floats. All of the three objects I just mentioned are hard. ***Pier* is another object that responds to water. You can only place it on water.** In other words, it floats on water but disappears if placed on land. All these objects can be seen in Figure 26.

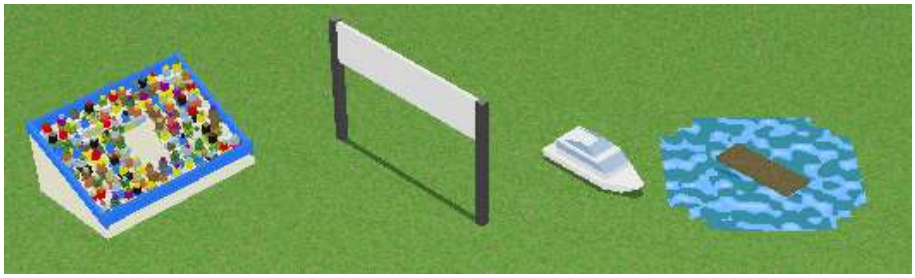

Figure 26. L-R: Stand, Gate, Boat, Pier.

Then we have three types of bridges (Figure 27). *Bridge* and *Flat bridge* are meant for racing, *Pedestrian bridge* is an overpass. The regular *Bridge* can be put on flat ground but the ***Flat bridge* requires that you raise/lower the hmap to make it work**. I'll explain this in the next section. The bridges are hard objects. **When you are using them, try to match the road width with the bridge width so it looks better.** At WS100, *Brush Size* 6 is perfect for the road. When placing a bridge, also make sure that the drivers don't hit the supporting poles under the bridge.

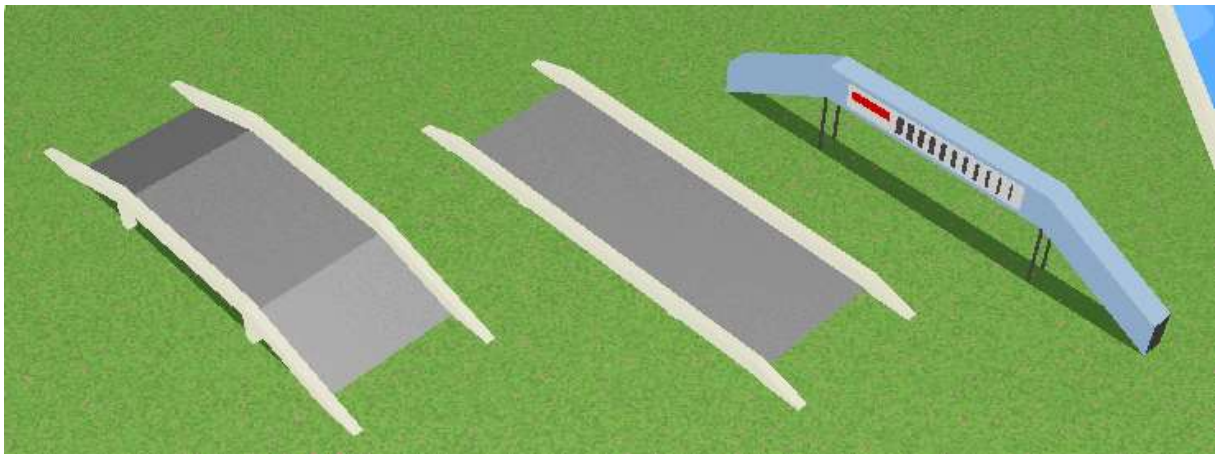

Figure 27. L-R: Bridge*,* Flat bridge*,* Pedestrian bridge*.*

Finally, there's *Person*. You can put these little guys on land but notice that GR randomizes what you get in terms of the *Person* size, shirt color, skin tone and hair. You may want to place track marshals somewhere but every now and then they are kids! In Figure 28, I have placed five guys in TE but in GR they appeared like this under two different runs. People are soft objects but they don't appreciate being hit – would you?

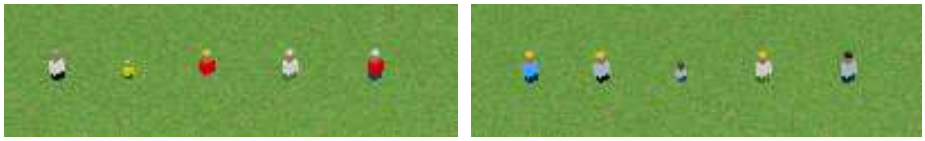

Figure 28. The same five Persons *under different runs.*

Bridges and hmap

As I mentioned, for the *Flat bridge* to work you need to do some hmap work. More precisely, **both ends of the *Flat bridge* should be on one level and the area under the bridge should be lower**. There's actually a so called bridge bug that affects the game play: if the transition under the bridge is too steep, the car jumps when driving to one direction over the bridge. The bug affects only one driving direction. Luckily the bug is easy to fix. If you make the hmap transition smooth enough (_/ instead of |_|), the bug goes away. Or then you reverse the driving direction. Figure 29 shows a correctly made hmap for the *Flat bridge*.

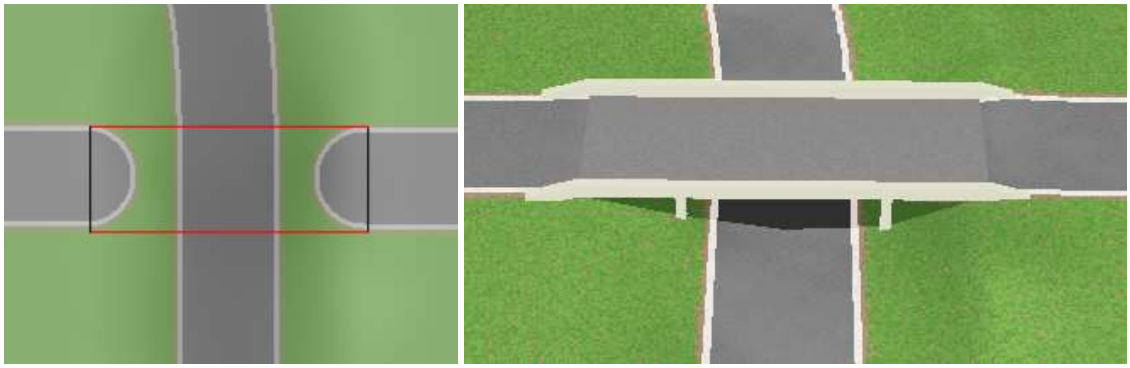

Figure 29. A Flat bridge *with smooth transition and no bridge bug in TE (left panel) and GR (right panel).*

Object placement

With object placement I usually **first place the objects that are critical for driving** (insides of the corners, walls near the track etc.) Then I fill in the surroundings moving from one place to another. I think about it a lot – I watch the track in GR repeatedly, add some objects and go back to GR to see whether it works or not. For me, people come in last! It's as if I first make the world ready and then let the people come.

Objects and gameplay

As I mentioned, I start with objects that are critical for driving and affect gameplay. **I suggest that you take a quick look at the section Checkpoints in corners** since that might affect what you want to put inside the corners. I usually design my tracks so that there's nothing too solid inside most of the corners. Quite usual way at circuits is to use *Sunk tyres*. They slow you down and make you jump so you get punished for trying to shortcut but don't break the car. When you place *Sunk tyres*, don't hide them by placing them on white surface! That advice should be followed with any other object – don't hide your objects! Sometimes people use two *Sunk tyres* on top of each other but turn the second one 90 degrees. This results in a nice ball-shaped marker (see Figure 30). If you zoom to one in GR, you can see it's not really a ball but with the usual WS's it doesn't matter. I put one inside the hairpin and another one inside the top corner since I think there's not enough space to use separate *Sunk tyres* at either place. Another object I often use inside the corners is *Bush*. The *Bush* makes it impossible to shortcut, slows down racing but it doesn't damage the car. I use one in the last corner. As you remember, the second corner doesn't require any objects as I put a small hmap mountain there, so I have now placed the objects that mark how far inside the corner drivers can go.

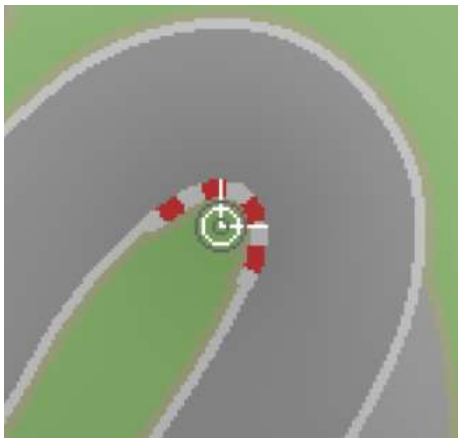

Figure 30. Two Sunk tyres *placed on top of each other to create a ball-shaped corner marker.*

Next, I add other objects that are related to the racing line. My advice is to **be kind to the racer**. For example, see where it's easy to go wide and give some room for error. A very good general rule is don't put sharp objects near the racing line. This game should be fun and that means the track should be fair. You may want to make a wildly difficult track but even the difficult track should be fair. Don't get me wrong – it's alright to have a place where you simply need to get it right (or slow down). In places like that, part of the fun is learning to drive through it. For instance, it's ok to have a *Concrete wall* in the corner in a street course since the idea is to get around dangerous walls. However, even then I'd consider moving the walls a bit further. Anyway, now I need to add a wall next to the track since I don't want cars to fall into water.

Making walls

When you make strings of objects, a longer wall for instance, make it look as smooth as possible. With a small WS it's more difficult than with a large WS but try to make it smooth nevertheless. The key to the smooth walls is to rotate them the minimum amount needed and to overlap the edges so that you don't see any "extra" pixels where the two walls meet. You can actually overlap them to a large extent as long as there are no extra pixels. This way the wall is smooth. The smoothness can be tested in GR by driving so that the side of your car scratches along the wall. If there are any jagged edges, your car bounces off the wall. I do this often when there's a wall next to a banked corner, for instance. Don't leave gaps between the objects that are meant to be continuous. Even if you put two pieces of wall right next to each other, you may see the surface behind them. It's a small thing but may have a large impact on the overall look.

I want my wall to be precise and the *Hedge* is my best bet since it's shorter than a *Concrete wall*. Moreover, there's even a *Short Hedge*. I place them next to the road and continue from top to bottom. I don't manage to do it without some "extra pixels" (see Figure 31) but as it looks alright in GR, I'm happy.

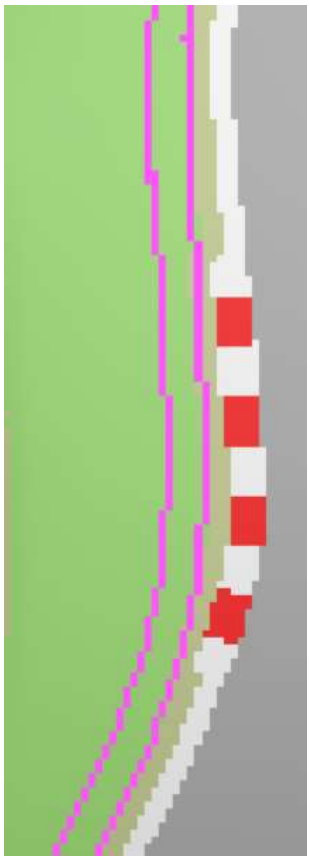

Figure 31. Hedges *joined to make a smooth continuous wall. Notice two "extra pixels" close to the top of the wall.*

Lining up objects

I don't want the driver to go wide at the exit to the main straight but don't want any sharp objects where the pit lane separates from the track. Therefore, I add some *Sunk tyres*. Figure 32 shows how I line them up so that each of them is approximately as far from the track as the others, uniformly following each other. I pay attention to **these kinds of things because it gives the track a more professional feel.** The same goes to edges of the walls I just talked about. If the walls look like they are simply thrown together, it makes a worse impression than a smoothly placed wall. When placing objects like these, the object shape helps you. For instance, *Sunk tyre* object consists of several circles, and I've made the white circles touch each other. I've also turned the rotation a bit from tyre to tyre since the straight curves a bit.

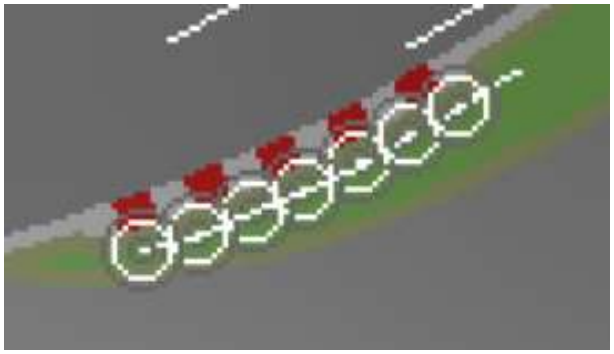

Figure 32. Sunk tyres *lined up.*

When I place objects that should be spread at even distances, for instance *Lights*, I usually use my eyes to guess the correct distance, then check it in GR. However, if I want to be very precise, I use some other objects for measure in between. Let me show you! I place the first *Light* along the main straight, then select *Office block* since that's the distance I want between my *Lights*. I rotate the *Office block* to the correct direction and make sure that one edge touches the center of the first *Light*, then place the building. Now I know where to place the next *Light*. I continue like this until I'm at the end of the main straight. In Figure 33, you can see what I've done. Now I just delete the *Office blocks*. This way I can be 100% sure that the *Lights* are uniformly spread.

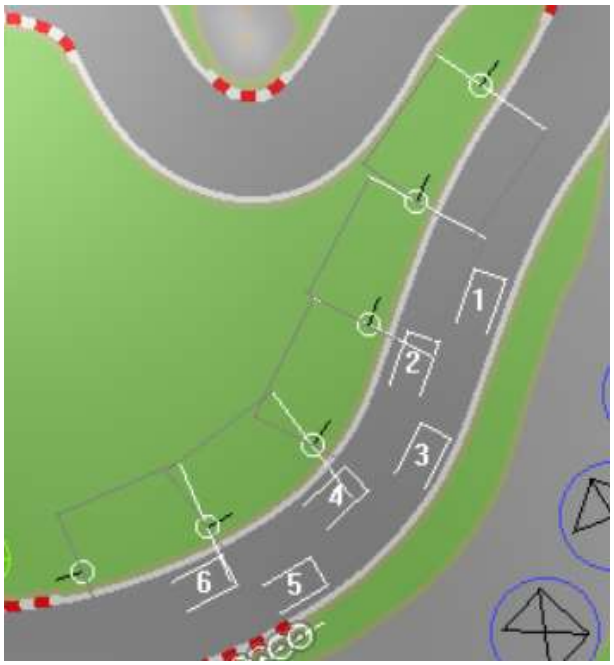

Figure 33. Using Office blocks *as measures to place* Lights *at uniform distances from each other.*

Objects and aesthetics

What I didn't tell you in the previous section is that my first attempt at placing *Lights* with the help of the *Office blocks* didn't work. I started at a wrong place and thought that the end result was lopsided so I had to delete them all and start again. In fact, through this tutorial, I have already tried several things that didn't work the first time (or second, or third, or…) but I haven't mentioned them all. The more I've made tracks, the more important for me is that they please my eye. That's why I think a lot about the objects. **I pay attention to how the objects work in relation to each other. I also give the track some "space to breathe" so that everything is not cramped.**

I add more objects next to the main straight. Since the *Sunk tyres* may throw the car in the air, I want something soft next. That's a *Bush*. Then I add few pieces of *Concrete wall* and end the wall with a set of three *Booths* next to the (imagined) finish line. I finalize the pit exit with a *Road cone*. I use *Bushes* at either end of a *Concrete wall* quite often. That's my way of making the wall safer. In the old times, I often put a *Hay bale* or a *Soft wall* there. However, these days I don't think that looks good. Sometimes I even struggle in making the end look nice. I don't worry about that because **objects can also be used to hide ugly transitions**!

Part of the aesthetics is adding objects that hide parts of the track. I have never liked tracks where objects block the view of a corner apex. I think that the driver should be able to see how to take the corner. On the other hand, placing some trees, for example, so that they block a bit of a straight makes the track look cool without adding too much to the difficulty. However, they shouldn't block the view completely so that when

another driver bumps into the player, player doesn't know where they are anymore. The difficulty should come from the track itself being tricky, not dodgy object placement. I'll add a *Leaf tree* that casts just a bit of its leaves on the main straight.

Starting positions

Now that my main straight is close to be finished, I should think whether the *Start* positions are in a good place. I asked you to place them on the main straight quite early but now that the track layout is final, it's good to think again. Do you think they are in the correct place? Are they spread uniform? **You should leave some space between them so that the start of a race is safe**. Are they in the correct order (1-2-3-4-5-6)? Some people like to play around with the *Start* position order but many people want to have them ordered similarly so that their championships don't get ruined (order points, reversed, for instance). **Notice that even if you haven't meant your track to be raced with all 6 cars, you need to place all 6 positions somewhere.** Also **make sure that they are on a straight**, at least more or less, since all the computer-controlled drivers start driving to the same direction. I'll tell more about this later.

I think my *Start* positions are in a good place but I want them to be even tidier. Therefore, I use the trick I used with the *Lights*: add one *Start*, measure the distance with a *Booth*, add second *Start* etc. Finally, I delete the *Booths*. The starting positions didn't move much but now they are more precise (see Figure 34).

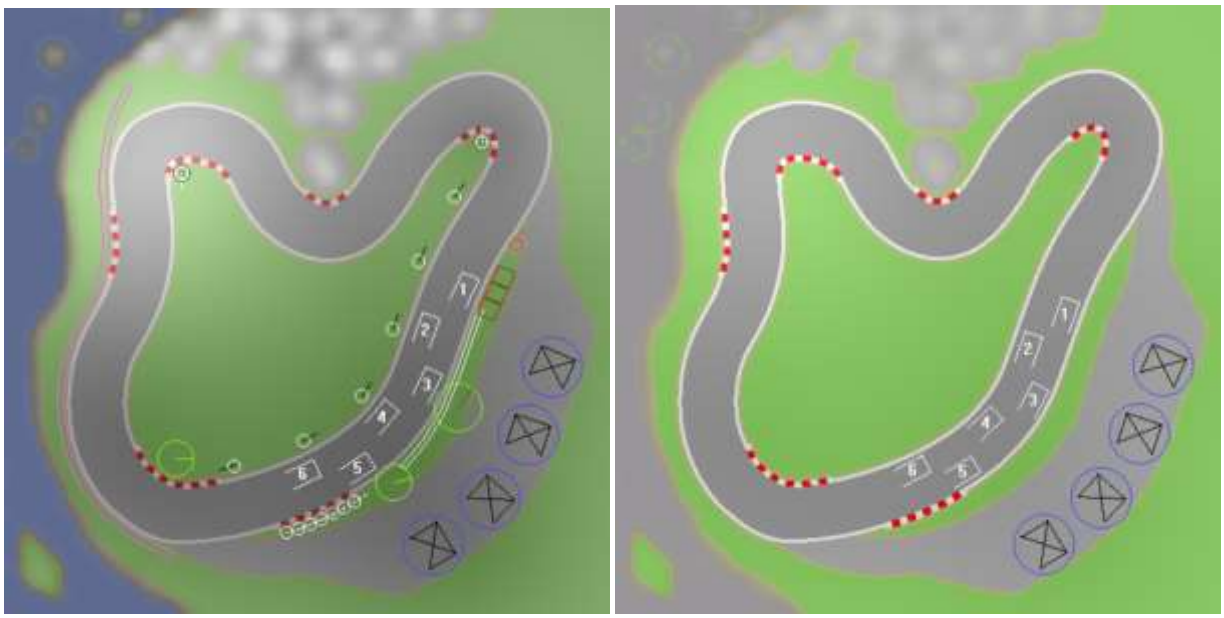

Figure 34. Left panel: The final places for the Start *positions. Right panel:* Start *positions from Figure 20 for comparison.*

Combining objects

The object inventory in TE is quite limited. Therefore, people combine them into different objects. I have already talked about making a ball-shaped marker from two *Sunk tyres* but it definitely doesn't stop there! I'm going to give you just some examples but **don't be afraid to experiment with other objects as well**! In the history of GR, there have been some very wild combinations. My first example is a round building. I select the *Office block* and place several of them on top of each other so that they are at different angles. **The key to a beautiful round building is to center all of them on the same pixel.** For this, you can draw one *White* pixel that you use as the center. However, unless you know the dimensions of the *Office block* well, you can't be 100% sure what's the correct place and you may even end up with a building bleeding on the road.

I use the following method for round buildings: I find the approximate place, push RMB and rotate the building while paying attention to the area the building corners cover. If the place is correct, I rotate the building back to a straight angle, click LMB and release RMB. Then I make sure the building tool overlaps the first building perfectly, rotate the object to 45 degrees, click LMB and release RMB. I continue systematically and plant two more buildings rotated so that their corners are at the middle between the two previous corners, rotated approximately at 22.5 and 67.5 degrees. Finally, I add four extra buildings so that their corners are at the middle between the previous corners. You can see the steps in Figure 35.

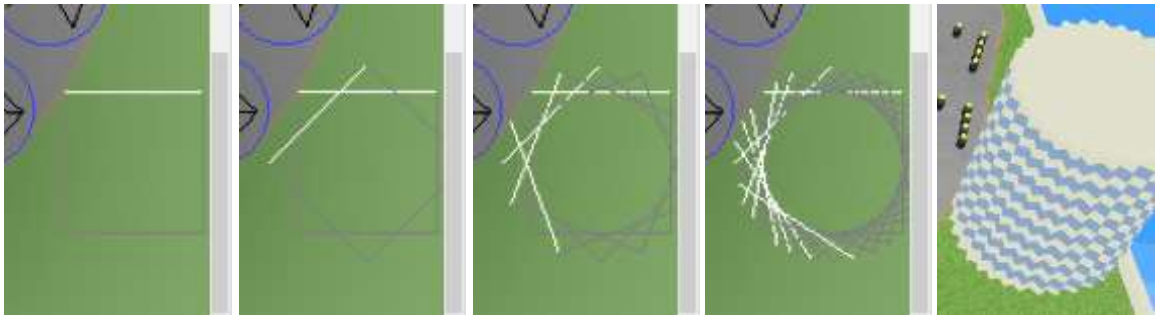

Figure 35. Panels A-D: Placing Office blocks *systematically to get a round building. Panel E: The finished building in GR.*

You can also combine different objects. **If you place a *House* under an *Office block* so that only its front end (yellow side) is visible, you get a nice door to your *Office block*.** Just make sure the orientation is the same for the *Office block* and the *House*. See Figure 36 for an example.

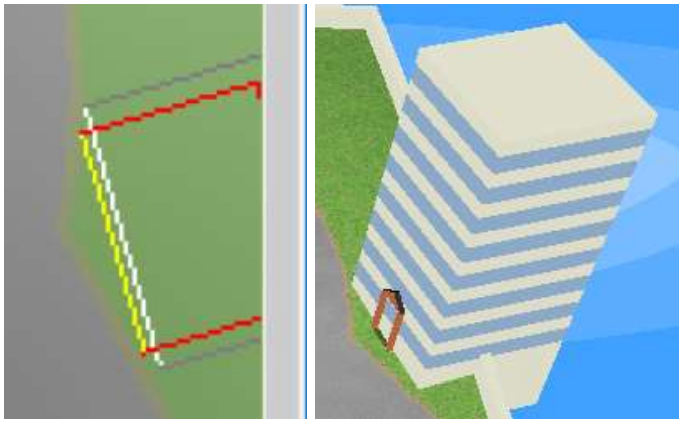

Figure 36. Left panel: Office block (small) *with a* House *placed so that the front is just a few pixels out of the* Office block*. Right panel: The finished building in GR.*

In the early years of GR, it was common to combine a *Boat* and several *Booths* into a truck. You may even add patches of *Oil* as tyres – remember that you need *Tarmac* below – and pieces of *Soft wall* so that only their ends stick just a pixel out of the *Booths*: red lights! See an example in Figure 37.

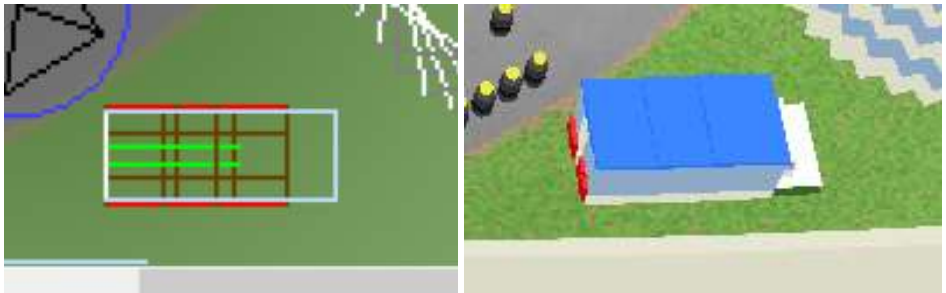

Figure 37. Left panel: A Boat*, 6* Booths *and two* Soft walls *to make a truck. Right panel: The truck in GR. Notice that it's missing the wheels made out of Oil.*

Object height

When placing objects so that they touch each other (making bigger buildings for instance, or that truck), I think it's important they look uniform. If you have hmap changes in the area you place two buildings next to each other, their windows or roofs are not aligned. **The trick to get them at the same level is to raise or lower hmap directly at the center of object.** Use *Brush Size* 1 and be careful. For the truck, I click three times on the center of both *Booths* on the left and once for both *Booths* at the middle. I check the result in GR and click once more in between the two *Booths* at the right. Now the roofs are as aligned as I can get (see Figure 38). I do the same for the three *Booths* at the main straight.

Figure 38. The truck after hmap adjustments.

This trick can also be used to get buildings of different heights. I add a round building with 8 *short Office blocks* and lower its center (Figure 39).

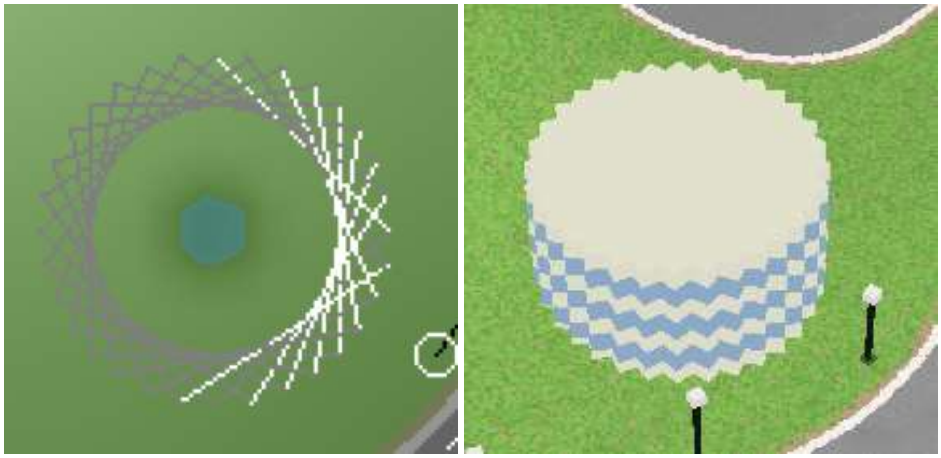

Figure 39. Left panel: A round building made of 8 short Office blocks *with hmap lowered at the center. Right panel: The finished building in GR.*

FINETUNING THE VISUALS

When I've reached the stage where I've added a bunch of objects, I usually notice that there are places that could look even better.

Retouching the lmap

When I place certain objects, I often need to add color to the lmap. For instance, when I added the *Office blocks*, I realized that for the people living in the game, there's no road for them to get to the buildings. Therefore, I add a pathway around the building and a small pathway from the track using *Tarmac#2* so that I don't get white lines, and use RMB for this so I don't erase my track (Figure 40). For the round pathway around the building I use the largest *Brush Size* and click just once on the exact center of the building.

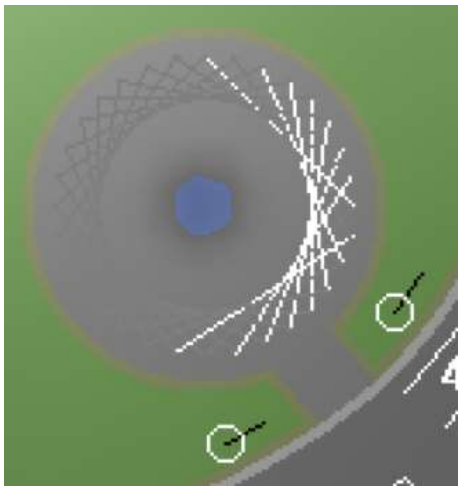

Figure 40. Added a pathway to and around the round building.

I also need *Tarmac* under the truck to draw the *Oil* wheels. Because that means adding new *Tarmac* next to the pitlane, I want to separate the pit area with some lines. While I'm at it, I decorate the pit lane. I choose *Yellow line* and draw both pointed (RMB) and solid (LMB) line. I use *Brush Size* 2 and *Rope Length* 3. I could've chosen *Brush Size* 1 as well for more finetuned lines but for this track, somewhat wider lines fit the atmosphere better. **I try to follow the same shape as the original pit lane and stay a fixed distance from the edges** while doing it. Boxes I draw with *White line* using the *Pit crew* objects as visual help. I also add a bit of *Tarmac#2* next to the *Office block* at the pit exit. Finally, I decide to add *Yellow* stripes. I need to *Undo Land* quite many times since I want the lines to look good. (I also saved an extra copy before I started drawing, just in case). **Now I can add a *Tarmac#2* path using RMB so I don't have to worry about the lines I just drew.** Then I can add wheels to my truck using *Oil*. You can see all these additions in Figure 41.

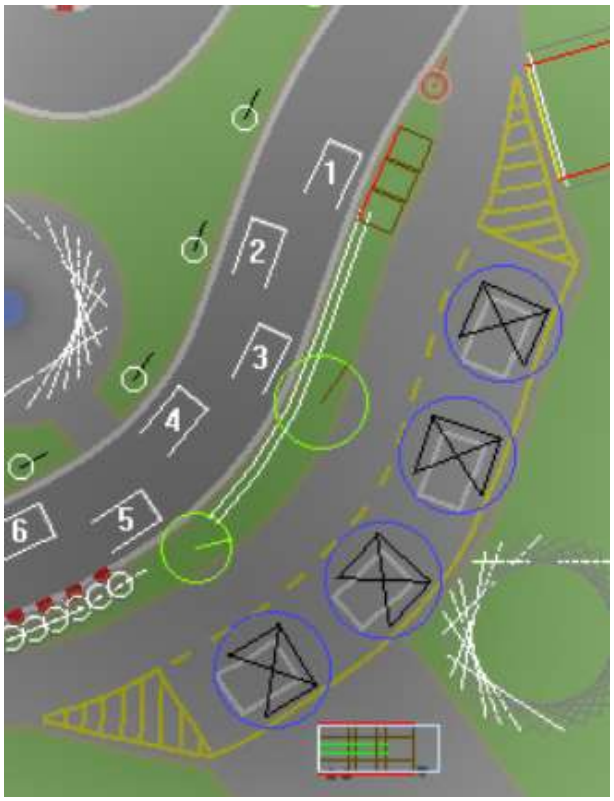

Figure 41. Added line paintings to the pit area and wheels made out of Oil *to the truck.*

When I'm doing this kind of (pixel) work, I sometimes find it handy to turn off the view of objects etc. By default, TE shows everything you've placed on the map, but there are options you can tick off: *Show Objects*, *Show Checkpoints*, *Show AI Lines*, and *Show Limiters*. I haven't talked about *Checkpoints*, *AI lines* or *Limiters* yet but I'll come to them in a minute.

Now that I'm doing lmap work again, I also want to draw lines for the starting grid. If I'm really in the mood of fine details, **I'll *Zoom* in *4x* and draw them pixel by pixel following the *Start* objects.** Now I'm in a more generous mood so I'll just slash them in. I know for sure that I want to draw the finish line as well but I'll do that a bit later after I've placed the checkpoints.

Let me also mention *Grass* textures. I usually do such things only after I've placed all the objects apart from the people but this seems the right spot in this tutorial. Think whether there's a place where the *Grass* should be worn off. Then choose *Sand* and the smallest *Brush size* and click once, maybe twice or three times to that area. You get ugly circles but next you choose *Grass*. Now click somewhere over the *Sand* and over the edge to break the monotony of the circles. Check the result in GR and finetune the patch if needed. See, you don't need an external program for textures! This also works with some other combinations. Since the dust on the edge of some surfaces is quite faint color, Figure 42 has *Height/Land* slider set all the way to the right.

Figure 42. Left panel: Added three Sand *circles to* Grass*. Middle panel: Broke the form by clicking several times with the* Grass *tool. Right panel: Final texture in GR.*

At this point of the tutorial, I'll stop reporting what I've done for a while. In fact, I also delete that *Grass* texture since I want something else there – I just wanted to show you the technique! Now I'll add more objects using the same principles I've already explained. See you in a bit!

Adding the final objects

Ok, I'm back, and Figure 43 shows what I've done. Notice that I've added *Invisible wall* next to the cliff since I realized it was relatively easy to get over the *Hedge* if you took the upper corner wrong. I don't want any drivers in the water! For this, I drove over the *Sunk tyres* on purpose. **I often drive badly on purpose in spots where I suspect hides a hazard to see whether it's true.** To make sure the *Invisible wall* is foolproof, I've made it continuous from one end of the map to another. At the top, I had to turn the wall along the track a bit since if I made it straight towards the top, I could drive under the *wall* because of the bumpy hmap! I also added those straight *Concrete walls* outside the hairpin and in the infield. **If you have problems making smooth curved walls, why not try if you can do them like this!** It looks both tidy and functional.

The nature is made entirely of *Leaf trees*, *Bushes*, and *Stones*. I thought that using any other types of trees than *Leaf trees* wouldn't work here. I mentioned the mountains earlier and, as promised, I've added some *Stones* there to make it look more real. I've kept the infield quite empty but I've strategically added one *Stone* so it doesn't seem too empty. **Sometimes adding a single object to an area is enough!** I think that in

GR visuals, less is often more. Everything that I added, I placed on quite individual basis and checked how it feels to me in GR. I also decided against *Grass* textures as I love that green color.

I added the people last. With them, I often try different combinations to see whether it looks right. For our project, I didn't feel that many of them are needed. I placed track marshals next to the wall openings, a couple of guys near the *Office blocks*, and some more around the truck. **If you want to make a crowded area, then you should place *Persons* even closer to each other than what I've done here.**

Figure 43. The track with all the objects added in TE (Left panel) and GR (Right panel).

Adjusting the zoom and rotation

You should always check how the track looks like under different aspect ratios. For this, use GR in window and test ratios 4:3, 16:9 and 16:10 (e.g., 1024*768, 1600*900, and 1680*1050). This is because the *Zoom* in *Track properties* doesn't work properly the more you move away from *Rotation* 0 (or +-90, +-180) and the lower your *View angle* is. Is some part of the track not visible? This is the most crucial. I've re-tested lots of old tracks recently and there are often problems with this since in the old days (GR 1.05 or earlier) the only option was 4:3. **If there are any problems, change the *Zoom/Rotation/View angle.* The safest bet is to use the default values.** Well, you can *Zoom* in to 100 or even 95. That way the track is a bit closer.

It may also be that at a different aspect ratio you see places that you shouldn't see, such as an empty place behind an *Office block*! I usually make my tracks pretty like a movie set. I don't care if things are not finalized behind the buildings for instance. However, sometimes there are surprises at different aspect ratios and I need to add something in that space.

At one point in the history of GR, it was fashionable to make full screen tracks –tracks where you can't see the sky or white edges of the map. However, with the current screen ratio and approach to *Zoom* values in GR, lots of those tracks are completely broken. **If you want to make a full screen track, pay extra attention to this, and maybe prepare two version of your track, one for 4:3 and one for 16:9.** 4:3, you ask. Why would anybody use that? At least I do, in the windowed mode. GR was originally made for 4:3 and that's how lots of old tracks should be driven.

CHECKPOINTS

GR counts completed laps with *Checkpoints*. They are invisible lines around the track, and the cars need to go through all of them in a numbered order for the lap to count. I usually add the *Checkpoints* after all the objects. This way I know for sure how wide they need to be. Of course, you can set them early and come back to them later. Notice that if you have *Checkpoints* but no *AI line* (I'll come to that in a second) and race length is set to kilometers, you can only get one lap. **To get around that just set your race length to laps.**

Placing the *Checkpoints* is very straightforward. You draw them to the track with the *Checkpoint* tool using LMB. You can delete them with the *Delete Checkpoint* button. Clicking this once deletes the *Checkpoint* with the highest number. You need at least one *Checkpoint* for the lap counter to work. However, just one *Checkpoint* means that you can make 0.01s laps so **I suggest you always place at least two.**

Checkpoints in corners

Nicely placed *Checkpoints* go unnoticed but bad ones ruin the experience. One thing I don't like is tracks where you don't know how much you can cut the corner. I think that the driver should be able to use every inch of the racing area to their advantage! Because of this, **you should always make sure that the driver can't accidentally miss a *Checkpoint*.** This is done by either designing the track so that there simply are no wrong routes you can choose, or by marking the ends of the *Checkpoints* very clearly by objects (or hmap).

Don't put the end of the *Checkpoint* too close to the track! Another driver might push the player off the track and the lap doesn't count. **With *Sunk tyres* inside the corners, you can safely move the *Checkpoint* end further away** from the corner because *Sunk tyres* punish the driver for their bad driving. No need to take a lap away! This is especially important in chicanes. If a driver drives straight because of an error, you shouldn't punish by taking the lap away. **Instead think what you can put inside the chicane** – maybe a *Hay bale* does the job. **Also think whether you need to use something solid** or whether can you get away with those *Sunk tyres*, for instance. At any rate, don't put those objects too close to the proper racing line.

Other checkpoint issues

You don't need *Checkpoints* on straights! In fact, **the best practice is to have as few *Checkpoints* as possible**. If your track has four major corners, like our project, five *Checkpoints* should be enough – one for each corner and one for finish line.

The last *Checkpoint* always counts as the finish line. **Make sure that you position the last *Checkpoint* in front of the starting positions** (unless you want to do something different on purpose). Too often I have seen tracks where the last *Checkpoint* is in the last corner. Because of this, it's basically impossible to get the fastest lap on any other lap than the first lap. Speaking of finish line, in *Track properties* there's an option *S/F Line*. If this option is checked, GR automatically draws the finish line on top of the last *Checkpoint*. This works for *Tarmacs*. I already mentioned that I want to draw my finish line, so I'll take that checkmark away and draw it now.

Finally, also make sure that **if you have multiple routes (including a separate pit lane!), the driver can drive through all *Checkpoints***. Sometimes people make a separate pit line and then put a *Checkpoint* that doesn't cross the pit line. This means that if one pits, their lap doesn't count. Not cool.

Figure 44 shows my *Checkpoints*. Each one in a corner starts from the objects that I've put inside (or hmap in the second corner) and stretch way past the track. If my corner is near the edge of the map, I usually stretch the *Checkpoint* all the way to the edge. Now there's no way someone will accidentally miss one of them – it'd require deliberate driving outside of the track to find new routes. I also made sure that the last *Checkpoint* is where I want the finish line to be (notice how I already drew the finish line within the last *Checkpoint* with *Yellow*) and it stretches over the pit lane.

Figure 44. Added five Checkpoints *and drew a finish line with* Yellow.

AI LINES

AI lines are GR's way to tell the computer (artificial intelligence, AI) where it should go. *AI lines* are always the last thing I add unless I notice a design flaw at the last minute. I think racing is in the heart of GR so **you should try to make a good *AI line***, even if you only hotlap. Well, what is a good *AI line*? **It should be relatively fast and it should not crash.** Some crashes are unavoidable depending on the racing situations but they should be an exception. On the other hand, **if the AI is too good, it's not fun for the player either**.

You can't always make a good *AI line*. If that's the case, you can try some tricks or simply live with the fact that the AI is not perfect. Ask yourself whether the racing is otherwise tight and fun. For example, at small WS the racing gets chaotic so it doesn't matter that much whether the AI is very fast.

The *AI line* also defines the length of the track. You may remember seeing it set at 0 under *Track properties*. When you add the *AI line*, that value tells the length of the line in meters. Sometimes people want their track to have several racing lines and draw the *AI line* twice around the track. However, this messes up the length and speed calculations, and it affects the final average speed. It may even affect pitting.

Making the AI line

You can draw the *AI line* if you want to but it's easier to make a base and import it. First, make a nice ghost lap in GR using Mini. When you use this ghost as the base for your *AI line*, you can drive as slow as you want. **For the best result take precise, smooth lines and drive quite close to the apex but not too close!** The AI tends to cut a little, and with practice you learn to see how near the apex you can go.

After you've set the ghost lap, click *Import AI line* and choose the track file. You should now see a yellow line going around the track. Figure 45 shows my imported *AI line*. Technically, the *AI line* consists of points that are connected by the line you see. The AI always looks for a certain number of upcoming points and calculates the correct path on basis of them. Bad AI only looks for the very next point but a good AI takes maybe 3 or 4 points into account. This is why the good AI tends to cut a little.

Figure 45. Drove a ghost lap with Mini in GR and imported it as the AI line*.*

If you're familiar with the *AI line* editing, you can edit the parts that you know could be better already. If not, it's better to check out how the AI drives in GR. Before that, **make sure that the small red part on the otherwise yellow *AI line* is in front of the first *Start* position, and there's a clear path from every start position to that red part.** In Figure 45, you can see that red part between first *Start* and *Checkpoint* #5. I usually move the red part close to the first corner, if possible. That's the point the AI will aim at in the start of the race. Only after they have reached it, they start to follow the proper *AI line*. If the red part isn't at the right spot, you can rotate its position by using the *Rotate AI line arrows*. The single arrows (< and >) move it point by point, the double arrows (<< and >>) a bit longer.

Tuning the AI line

You should always make the AI line for AI level 100. This way the tracks don't vary too much in terms of the AI levels. AI above level 100 actually cheats with better traction than you! **Try to make the *AI line* either for the original cars (e.g. Formula or General) or say to other community members which car is preferred.**

Test the *AI line* with 6 AI drivers with the car you think is the most suitable and see where AI gets in any trouble. I test our track with Formula, and see that the ghost lap is alright. There are some problems, though. After the start AI gets close to the wall so I move the red part towards the hairpin. The cars also hit the *Sunk tyres* in the hairpin regularly (Figure 46). Therefore, I pull the *AI line* a bit away from the apex.

Figure 46. The AI keeps hitting the Sunk tyre *ball.*

In TE, you get to the correct spot of the *AI line* by rotating the red part. **The correct spot to start editing is always after the point that you need to edit**, since the *Delete AI* button erases *AI line* backwards. Delete

the *AI line* from the area where the AI got into trouble, both before and after the spot. The part connecting the yellow is still red but now the red part is longer. In Figure 47, you see what I've done at the hairpin.

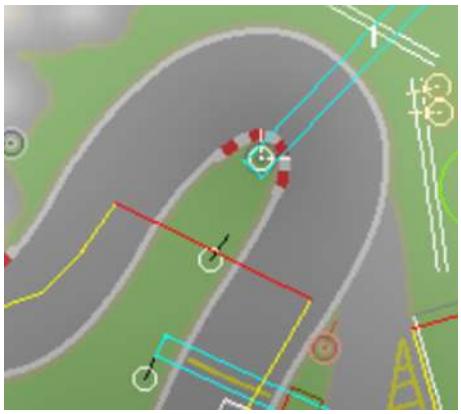

Figure 47. Deleted the AI line *around the hairpin area.*

Now you can fix it with the *AI line* tool. I prefer drawing *AI line* point by point (Figure 48). However, you can also draw it like it was a surface line. *Rope length* works here as well. The reason why I prefer clicking point by point is that I can get the line smoother and go exactly where I want. **After I'm done, I move the red part to where I think it should go** (last panel of Figure 48). Then I go test the changes in GR.

Figure 48. Panels A-F: Drew new AI line *point by point. Panel G: Moved the red part close to the corner for the AI to aim at during the start.*

Oh no! The red part causes the last AI car on the grid to crash on a *Light*. In the end, I have to move the red part backwards by four clicks. Now the hairpin works well.

After you've fixed the places where the AI gets into trouble, you should **revise other places to make the AI better**. I mentioned that you will learn how near the apex the *AI line* should go. This is one thing. I also notice that the AI could be better in the second corner – I change that part too. **The transitions to curves should be smooth: the smoother the *AI line* is, the better the AI drives.** Because of this, **you may need to start the transition to a corner earlier and end it later than you think.** This way the cars don't have to brake too hard. **The less AI needs to brake, the faster it drives.** Of course, you can't make a tight corner without braking. Because of the smooth lines, it's not enough that I change the second corner *AI line*, I have to change the third corner *AI line* too! Compare the original and redrawn *AI lines* in Figure 49.

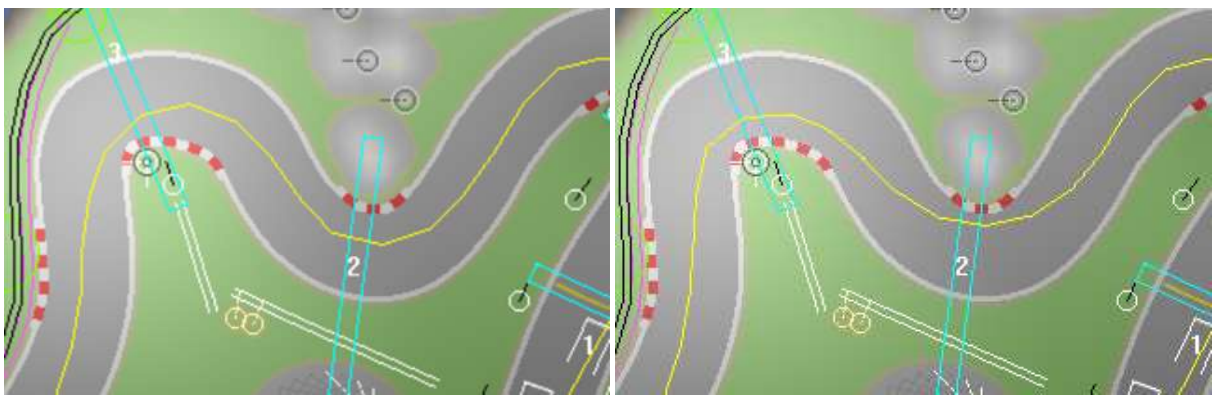

Figure 49. Left panel: Imported AI line *around second and third corners. Right panel: Redrawn* AI line *around the same area.*

I mentioned that the fast AI should go near the apex since that is the fastest way around the corner. However, **if you notice that the AI tries to overtake in a particular corner often, give them a bit more room for overtaking by moving the *AI line* a bit away from the apex**. This makes racing safer. Of course, you shouldn't do this at every corner but only if one corner is clearly a good place for the AI to overtake each other.

Also make sure that the AI drives straight when it should drive straight. If there are even small alternating turns on the straight, the AI starts to squiggle. I edit my main straight *AI line* just a bit. You can actually make the AI faster by deleting points on a straight from the *AI line*. I don't want to do it here but I do delete that tiny part that was colored red in Figure 45. As I mentioned, the AI aims at few next points and if there are less *AI line* points on a straight, the faster the car gets. However, if you delete points on a straight and the *pit AI line* joins this specific part (more about the *pit AI line* soon), the AI doesn't know what to do.

Remember to pay attention to the start. As race start is usually the most crowded phase of the race, you may get much chaos in the first corner. If this happens, you might want to stretch the starting positions a bit further from each other.

I'm happy with our *AI line* but let me tell you about some other things you need to pay attention to.

AI problems

AI might have problems when a corner is immediately followed by another corner to another direction. **In cases like these, it's better to find the line that is best overall instead of first trying to get one corner good and then the other.**

Another potential problem is when the AI gets away from the *AI line* since then it tries to get back to the *AI line* as directly as possible. This becomes an issue if the AI gets to the wrong side of a wall, for instance. This can happen when an AI wants to pit but another AI hits it on the main straight. Now the AI is stuck on the wrong side of the wall and tries to get through the wall no matter what. It may even start doing kamikaze moves and torpedo your race. **Test the AI enough so you see this can't happen or perhaps make small openings to the pit wall.**

Finally, AI might get into trouble after it has been damaged badly. This is because when it's crawling at a very slow speed, it takes the *AI line* as tight as possible, cutting more than it would at normal racing speed. Therefore, it may drive into places it otherwise doesn't. You need to take this into account as well. My suggestion is to **test the track with the AI enough so that you know that the AI doesn't crash much** – if this happens only rarely, it's just a matter of unforeseen circumstances.

AI and hmap

AI doesn't like big hmap changes. In fact, even a small bump in a wrong place may throw it off the *AI line*. Therefore, you should test even more extensively when you have hmap changes. The good thing with our project is that the AI works nicely despite the hmap changes – I managed to make smooth hmap transitions. If you see AI trouble because of the hmap, **you can make the hmap a bit less extreme.** The smoother the hmap is, the better the AI succeeds. Of course, with small WS and hmap changes you have even harder time making a good *AI line*.

I personally hate touching the hmap at this stage to make it less extreme. Therefore, I always try to make the AI work with redrawing it. Luckily, there is a trick you can try to make the cars slow down appropriately. If you draw a part of the *AI line* as zig-zag (\/\/), it forces the AI to slow down (see Figure 50). Too bad it also makes the AI squiggle so that may cause other problems!

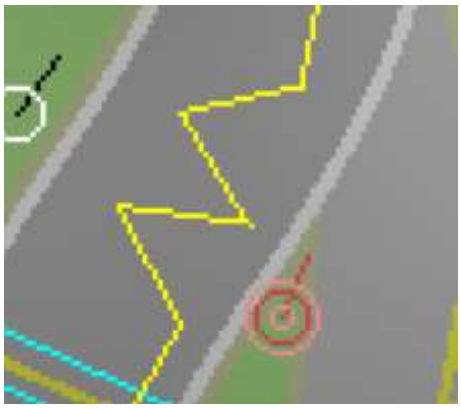

Figure 50. A zig-zag in the AI line *slows down the AI drivers.*

Another trick is to draw one point in the *AI line* to the exact opposite direction so that there are points on top of each other (see Figure 51). With this technique, the AI doesn't squiggle but remember that it works properly only if the cars are driving on a perfect straight.

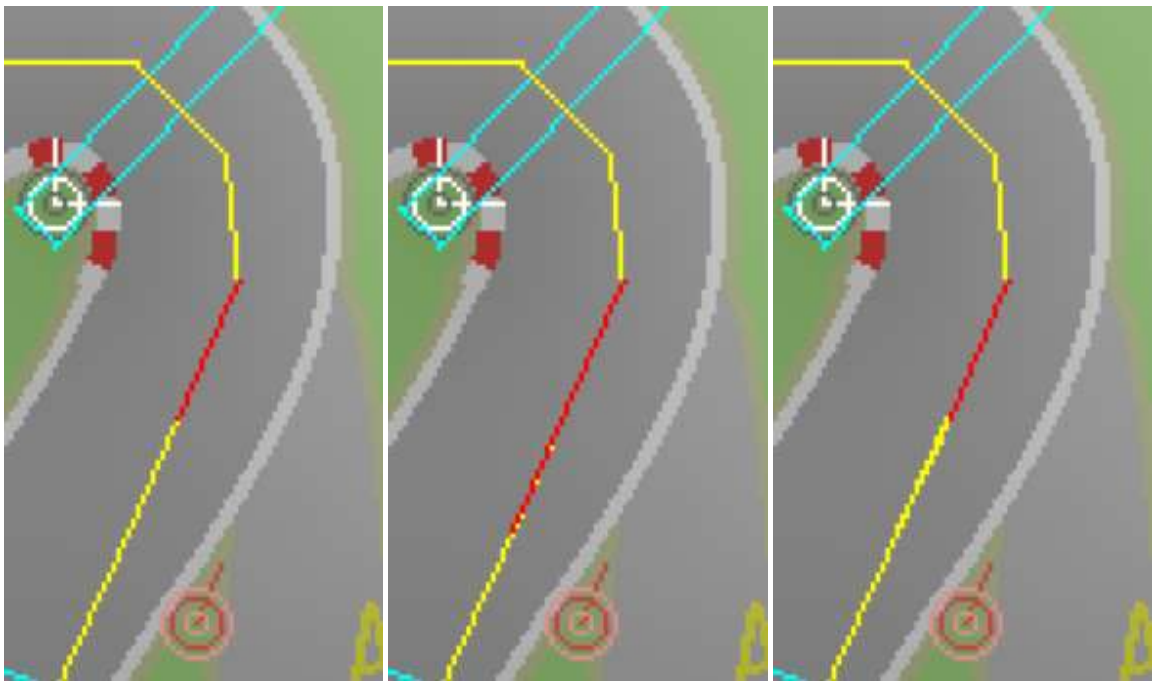

Figure 51. Left panel: The place where we force the AI to brake. Middle panel: First make the AI line *to go backwards by clicking as closely on the existing line as you can. Right panel: Then click the second point to the racing direction, again as closely on the existing line as you can.*

Yet another trick is to use the *Pit limiter*. I'll come back to this in a minute.

Sometimes you just have to be very creative with the *AI line* to make it work. I had one track with "rocky" surface (actually tarmac) next to water and the best racing line was to use the "rocky" part as banking. I couldn't get the AI to work at all until I drew the *AI line* to go in water. It didn't make any logical sense to me but somehow it worked.

Pit AI line

After you've made the *AI line*, you need to make the *pit AI line* as well. For this, the tool is *AI line (pit)*. Unfortunately, the *pit AI line* can't be rotated and you can't import a ghost either. In other words, you need to draw the *pit AI line* yourself. First, click the spot where the *pit AI line* should divert from the actual *AI line* and then draw it. **I suggest doing it point by point.** If you make mistakes, you can use the *Delete Pit AI line*

button. This button always deletes the line from the end towards the beginning. Be careful that you don't click *Delete AI line* instead – I've done it way too many times!

The same rules apply as for the regular *AI line*. In other words, the *pit AI line* should be smooth**. It should start from the regular racing line and end at the regular racing line** in a smooth fashion. **It should go near the pits but not too near so that the cars leaving the pit don't hit the other *Pit crews*.** Look at Figure 52 to see what I mean. There shouldn't be big corners and big hmap changes at the pit entries. You should also make sure that the cars going into the pit and out of the pit don't hit objects or other cars. Make sure there's enough space to enter the pits safely. The biggest mistake you can make is to place the *Pit crews* so that an AI hits some object on its way out. Rookies often draw the *pit AI line* directly over the *Pit crews*. That's a big no-no! This is a way to make the AI hit other cars and that's not cool.

Figure 52. Added pit AI line.

For testing the *pit AI line*, I usually have AI races with higher fuel consumption and damage. This way I get cars to pit earlier. Furthermore, I can observe how the cars drive damaged. The *pit AI line* in Figure 52 is my second attempt. The first *pit AI line* I drew went too close to the *Pit crews*: the AI drivers "sniffed" the pits too early and entered them in a non-optimal way.

Pit limiters

As you know, many professional car racing series have pit speed limits. You can do that in GR with a tool called *Limiters*. With this tool you can draw two (and only two) *Checkpoint*-like lines. When a car passes one *Limiter*, it slows down to a crawling speed until it passes the other *Limiter*. It doesn't matter which direction the car is going. The *Limiters* can be deleted with the *Delete Limiter* button, which erases them in the order of first last, then first. I'm not a big fan of *Limiters*, so I try to design my tracks so that I don't need them.

Pit *Limiters* should be placed with care. You need to be absolutely sure that a car doesn't hit the pit *Limiter* accidentally. If a car hits only one *Limiter*, it keeps crawling around the track while the others are zooming with fast speed. This will only end in mass chaos.

Limiters actually have another use. If you make a track with sudden hmap changes, you may not get the AI to behave nicely. Strategically placed two *Limiters* just before a jump can slow down the cars just enough! The *Limiters* don't need to be far from each other for this to work. In fact, the closer they are, the better it is for smooth racing. Cleverly placed *Limiters* may even go unnoticed by the player! When using this trick, you have to make sure that every car always passes both *Limiters*.

IS IT READY?

After you've put everything in place, test the track more. Drive some races with the AI. **Even if you think you have made everything correct, test it.** Maybe you forgot to rotate the red point of the *AI line* to a wrong place! Or maybe your track has multiple layouts and you need to finalize the other versions. If you're not in a rush, you could even sleep one night and test the track once more before the release.

Ask yourself whether everything works. If not, go back to that part. If yes, open the *Track properties* once more. I asked you to write your name or username to the *Author* field in the beginning. **Make sure your name is there**. It's a bummer to see so many old tracks with no information about who made them! You may also want to add something to the *Comments* field. Notice that it can't be seen in GR but in TE. Some people put here version information of the track. I put here my website address.

Does your track have a cool name? If you didn't think of anything in the beginning, the track is still saved with the random name. Sometimes that random name is actually good but, in my opinion, very often it's not. Is there a feature in your track that could be used as a name? If you want to rename the track, just save it under a different name. I don't want to release a track called Pyyc so… ha, here's a word play: I'll call this track Tutoring.

Once the track is done, **take a screenshot of it in GR with F12**, edit it to smaller size in a graphics program (or use F11 for an automatically resized screenshot), maybe write a readme with information about the track. **Zip the track and readme and post it to GRIF alongside the screenshot.**

Yes, it is…

Phew, that's it! When I started writing this, I had an idea of a short tutorial but look what monster grew out of it! I have shown you many tricks that I know and I've made another cool little track (Figure 53). I really hope this tutorial is useful to you. Spread the joy of GeneRally!

Figure 53. Finished track in GR.