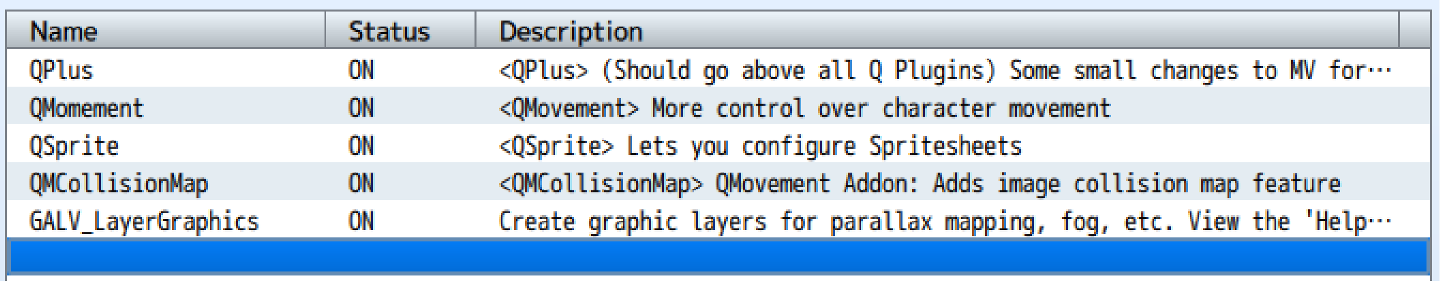
Layers and Movement with QMovement

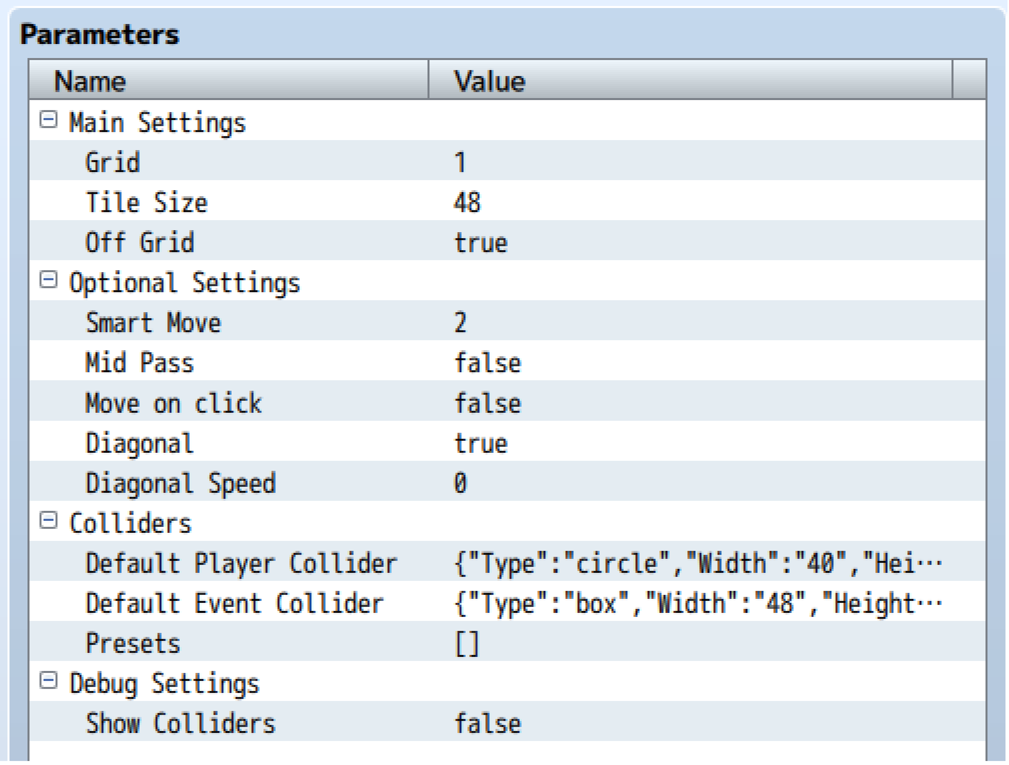
Required Plugins



Name	Status	Description
QPlus	ON	<QPlus> (Should go above all Q Plugins) Some small changes to MV for…
QMomement	ON	<QMovement> More control over character movement
QSprite	ON	<QSprite> Lets you configure Spritesheets
QMCollisionMap	ON	<QMCollisionMap> QMovement Addon: Adds image collision map feature
GALV_LayerGraphics	ON	Create graphic layers for parallax mapping, fog, etc. View the 'Help…

I have other tutorials available for the setup of QSprite, so I will not be including its extensive setup as a part of this document.

QMovement Parameters

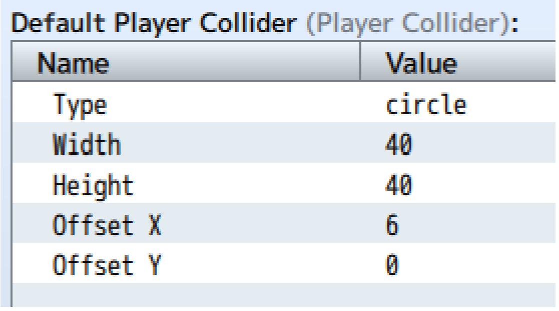


Parameters

Name	Value
⊟ Main Settings	
Grid	1
Tile Size	48
Off Grid	true
⊟ Optional Settings	
Smart Move	2
Mid Pass	false
Move on click	false
Diagonal	true
Diagonal Speed	0
⊟ Colliders	
Default Player Collider	{"Type":"circle","Width":"40","Hei…
Default Event Collider	{"Type":"box","Width":"48","Height…
Presets	[]
⊟ Debug Settings	
Show Colliders	false

Set "Move on Click" to False. If you want to work this click to move later, there is an additional QPathfinding plugin available, but even with it, I find that in my maps, the character still doesn't path well.

Change the default "Show Colliders" to False. You can toggle colliders on when needed by pressing F10



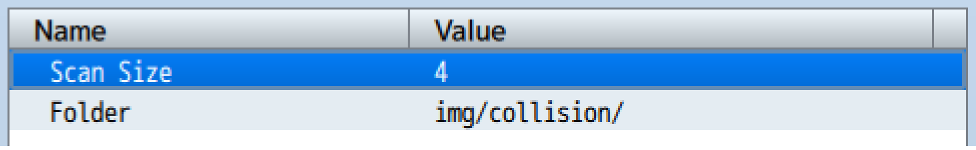
Default Player Collider (Player Collider):

Name	Value
Type	circle
Width	40
Height	40
Offset X	6
Offset Y	0

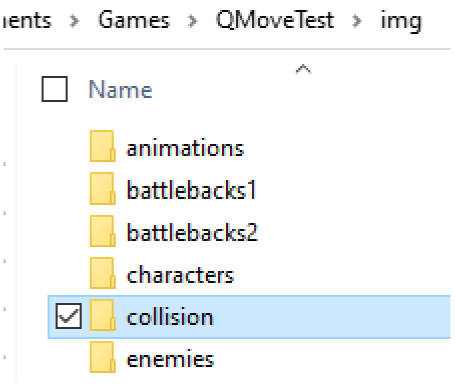
Set up your "Default Player Collider".
When playtesting, you can press F10 to view the collider, and adjust this to meet your needs.

Mine is probably smaller than normal so the character can fit through the doorways in my maps.

QCollisionMap Parameters



Name	Value
Scan Size	4
Folder	img/collision/



ıents › Games › QMoveTest › img

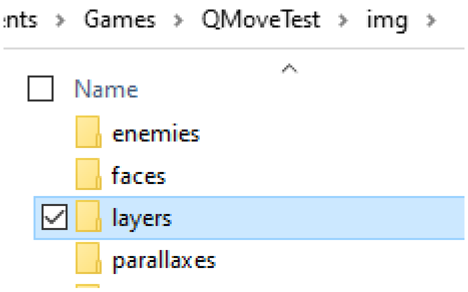
□ Name

- animations
- battlebacks1
- battlebacks2
- characters
- ☑ collision
- enemies

By default, this plugin uses the Parallax folder for collision maps.

However, since I will be using that folder for Parallax maps, I prefer to create a separate folder dedicated to Collision Maps. Then I adjust the parameter to point to it instead.

GALV_LayerGraphics



:nts › Games › QMoveTest › img ›

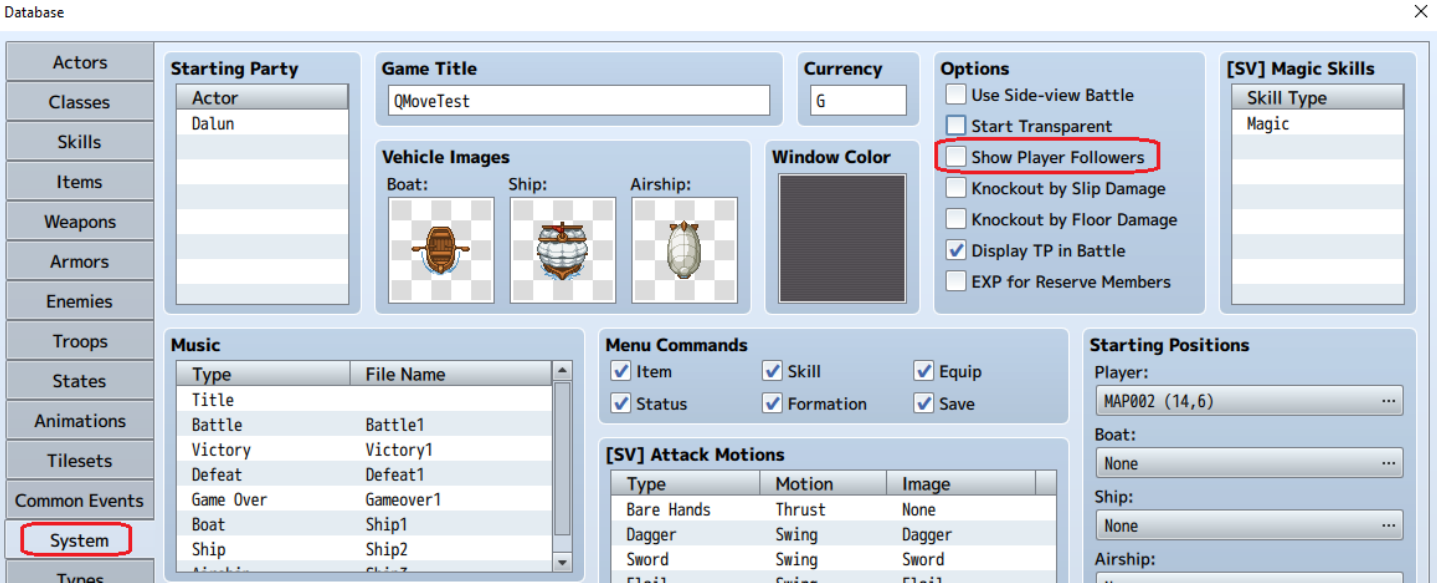
□ Name

- enemies
- faces
- ☑ layers
- parallaxes

No changes need to be made to the plugin parameters, but the documentation of the script does specify that all layer images must be stored in a newly created folder called Layers. Create that folder now.

Additional Database Setup

In the RPG Maker Database, go to the System Tab and turn off “Show Player Followers” since followers are not supported by QMovement.



Starting Party

Actor
Dalun

Game Title: QMoveTest

Currency: G

Options

- □ Use Side-view Battle
- □ Start Transparent
- □ Show Player Followers
- □ Knockout by Slip Damage
- □ Knockout by Floor Damage
- ☑ Display TP in Battle
- □ EXP for Reserve Members

[SV] Magic Skills

Skill Type
Magic

Music

Type	File Name
Title	
Battle	Battle1
Victory	Victory1
Defeat	Defeat1
Game Over	Gameover1
Boat	Ship1
Ship	Ship2

Menu Commands

☑ Item ☑ Skill ☑ Equip
☑ Status ☑ Formation ☑ Save

[SV] Attack Motions

Type	Motion	Image
Bare Hands	Thrust	None
Dagger	Swing	Dagger
Sword	Swing	Sword

Starting Positions

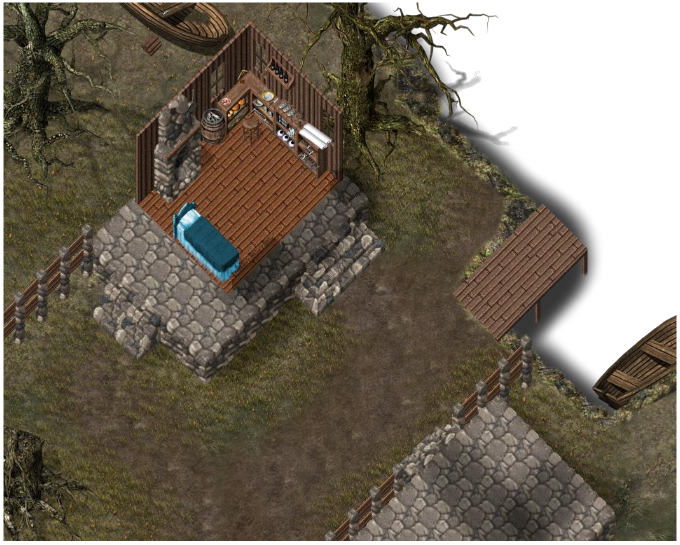
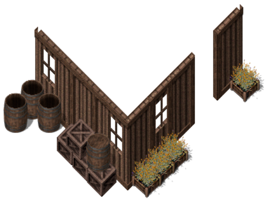
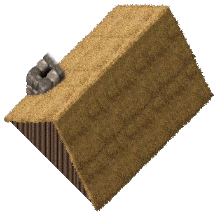
Player: MAP002 (14,6)
Boat: None
Ship: None
Airship:

If you want to have followers, will have to make use of additional plugins.

Map Layers

I will be using 5 images for this tutorial map. Each layer image must be stored I the img/layers folder to be used by the Layers plugin.

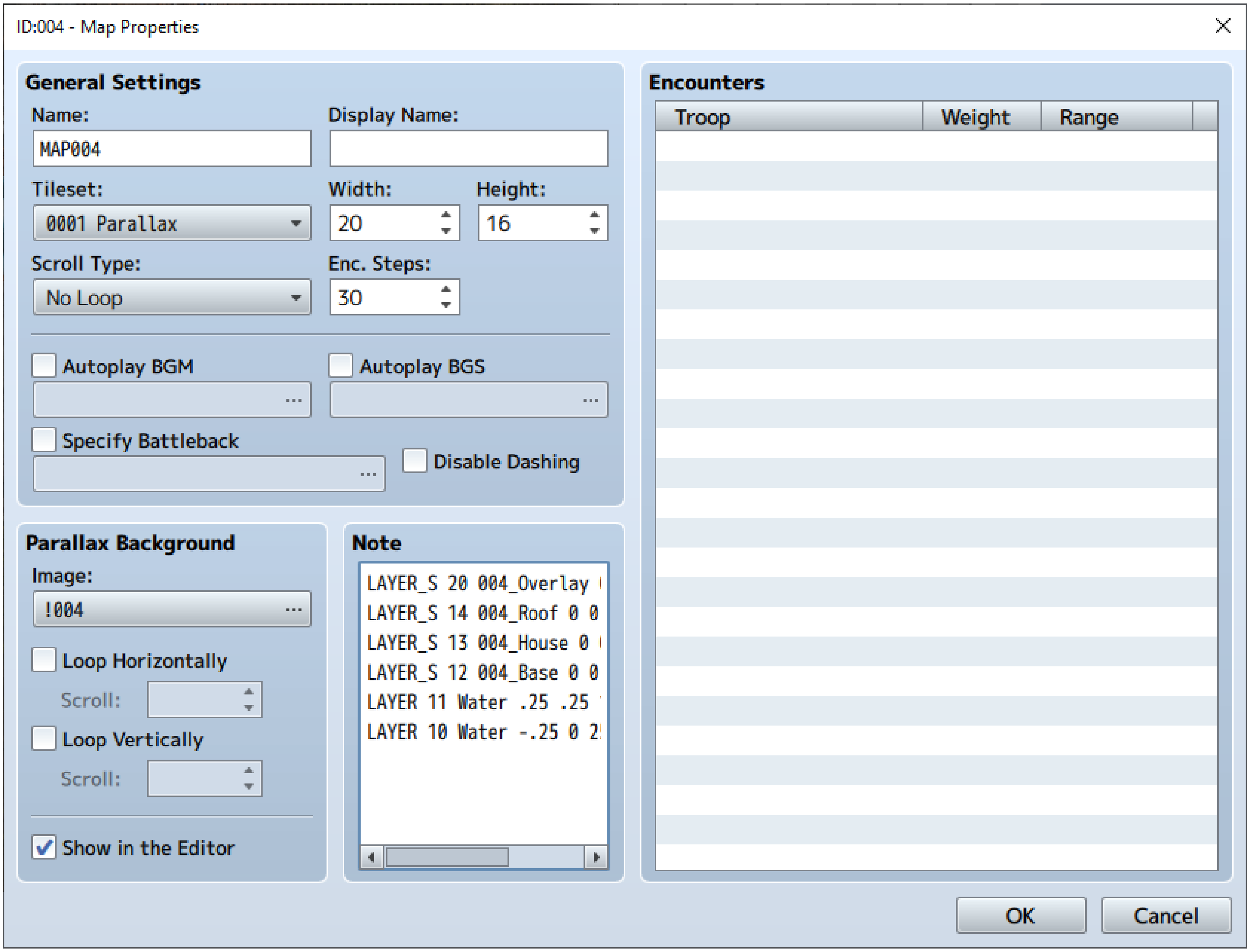
For management purposes, I prefer to prefix my map related images with the Map Number where they will be used. Since I am using map 004 in my project for this tutorial, all layer images for this map will be prefixed with 004_.

004_Base.PNG  The first Base image will always sit at Layer 1 below the character to be walked over.	**004_House.PNG**  The House Wall image will start off at Layer 1 when the player is outside. It will be moved up to Layer 5 to be above the player when going inside the house.
004_Roof.PNG  The Roof layer will always be at Layer 5 above the player. It will be changed to Transparent when the player goes inside the house.	**004_Overlay.PNG** The final Overlay layer is additional objects that will also always be at Layer 5 above the player, but do not have to be changed.

I will also be using a Water image (not shown) to animate the water in the lake below the Base layer.

To make it easier to place events in the RPG Maker editor, a copy of the Base map should also be stored in the Parallax folder with an exclamation point (!) at the beginning of the name.   Ex: !004.PNG

Again, for ease of management, I am using the Map Number as the image name.   The addition of the ! to the filename forces RPG Maker to lock the image to the map so it doesn't slide as the player moves around the map.



ID:004 - Map Properties

General Settings

Name: MAP004

Display Name:

Tileset: 0001 Parallax

Width: 20

Height: 16

Scroll Type: No Loop

Enc. Steps: 30

☐ Autoplay BGM

☐ Autoplay BGS

☐ Specify Battleback

☐ Disable Dashing

Parallax Background

Image: !004

☐ Loop Horizontally

Scroll:

☐ Loop Vertically

Scroll:

☑ Show in the Editor

Note

```
LAYER_S 20 004_Overlay
LAYER_S 14 004_Roof 0 0
LAYER_S 13 004_House 0
LAYER_S 12 004_Base 0 0
LAYER 11 Water .25 .25
LAYER 10 Water -.25 0 2
```

Encounters

Troop	Weight	Range

Now that all of the images are saved, create or edit your map in RPG Maker.

Set the Width and Height of your map (measured in 48x48 grid tiles) to match the size of the map images you created. Attach the Parallax copy of the Base Map, and make sure it is set up to show in the editor.

In the Notes section, you will include the settings for the Layer plugin so that the layers will appear on the map.

```
LAYER_S 20 004_Overlay 0 0 255 5 0
LAYER_S 14 004_Roof 0 0 255 5 0
LAYER_S 13 004_House 0 0 255 1 0
LAYER_S 12 004_Base 0 0 255 1 0
LAYER 11 Water .25 .25 100 0 0 0 0
LAYER 10 Water -.25 0 255 0 0 0 0
```

GALV_LayerGraphics

The LAYER_S command creates a static layer attached to the map.

```
   LAYER_S PIC# LAYERFILE X Y OPACITY Z BLEND
```

The LAYER command creates a constantly moving image, typically for fogs and water

```
   LAYER PIC# LAYERFILE XSPEED YSPEED OPACITY Z XSHIFT YSHIFT BLEND
```

PIC#: Each layer uses a picture image from the "Show Picture" Event command.
This specifies which picture ID will be used for the layer. For consistency, and to ensure certain picture ID's are available for other purposes without damaging the layers, I set aside a specific range of ID's for layers (10 – 30 for me).

LAYERFILE: The filename of the layer graphic, without the PNG Extension.

X / Y: Moves the layer image on the map. Leave this set to 0 if the layer is the same size as the map. If you use smaller images for some layers with a lot of transparent space to reduce file size, this would be needed to position the smaller image correctly in the map.

OPACITY: Sets the transparency of the layer. 0 = Fully Transparent, 255 = Fully Solid

Z: Sets the level the layer will be drawn on in relation to the character and other game objects.

- RPG Maker Parallax
  - Layer: 0
  - Layer: 1
- Event "Below Player"
  - Layer: 2
  - Layer: 3
- Player
  - Layer: 4
  - Layer: 5
- Event "Above Player"
  - Layer: 6
- Weather Effects

For multiple layers within the same Z Layer, pictures with a higher Pic# will be drawn above lower Pic#s

BLEND: Graphical Blend Mode (0 = normal, 1 = add, 2 = multiply, 3 = screen)
0 will be used for normal layering, the other options seem to useful for lighting effects.

XSPEED / YSPEED: Vertical / Horizontal movement speed of the image

XSHIFT / YSHIFT: Similar to X/Y, moves the base point of the picture's movement.

In addition to the map notes, the layers can also be adjusted via Plugin Commands, with the addition of one parameter, to specify the map the layer command will be applied to.

```
   LAYER_S MAPID PIC# LAYERFILE X Y OPACITY Z BLEND
   LAYER MAPID PIC# LAYERFILE XSPEED YSPEED OPACITY Z XSHIFT YSHIFT BLEND
```

This is a good point to test your game to make sure everything done so far is working.

When testing, you should see the animated water along with all of the map layers.

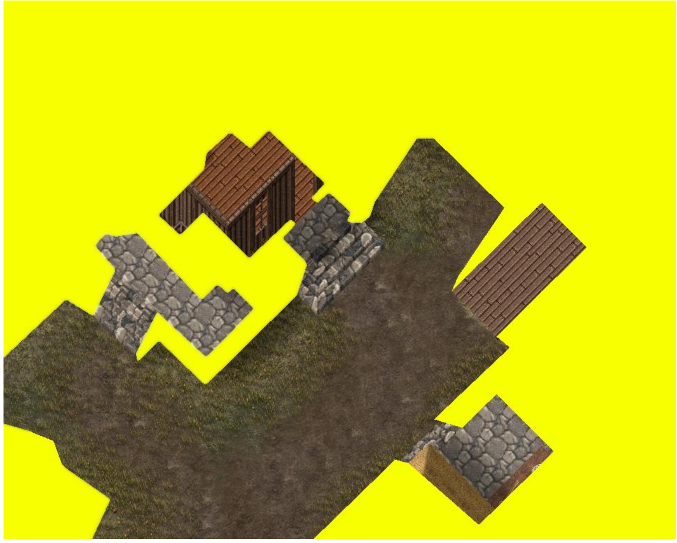
Since we have not built collisions yet, you will be able to walk anywhere, including on top of walls, but you should be able to go under the roof.

Adding A Collision Map

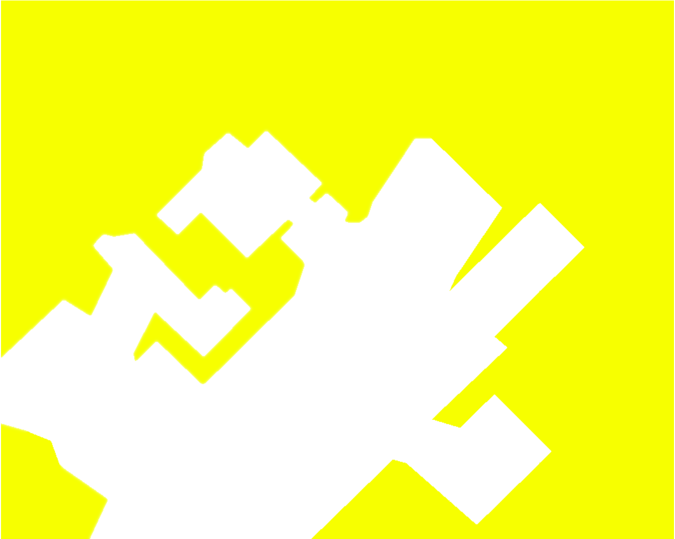
A Collision map is an image used to define places that the character is not able to move. This image can be created in GIMP or your graphic editor of choice.
With the QCollisionMap plugin, any transparent areas of the image will be passable, any non-transparent areas will be blocked.

Using my map layers created previously, I use the Free Select (lasso) tool to define the areas that will be restricted from movement. Once set, this overlay image can be exported into the Collision Map folder defined in the Plugin Parameter. Like all other map related images, I am going to name this the same as my map number: 004.PNG.

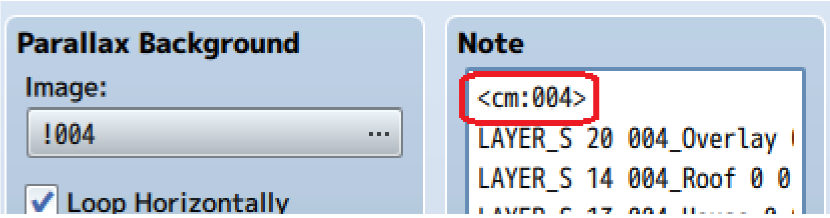
Collision area drawn on a layer over the map in GIMP.



Exported Collision Image: 004.PNG



Once the Collider Image is created, it will need to be connected to the Map by adding an additional Note along with the Layer commands in the map properties.



Parallax Background

Image: !004

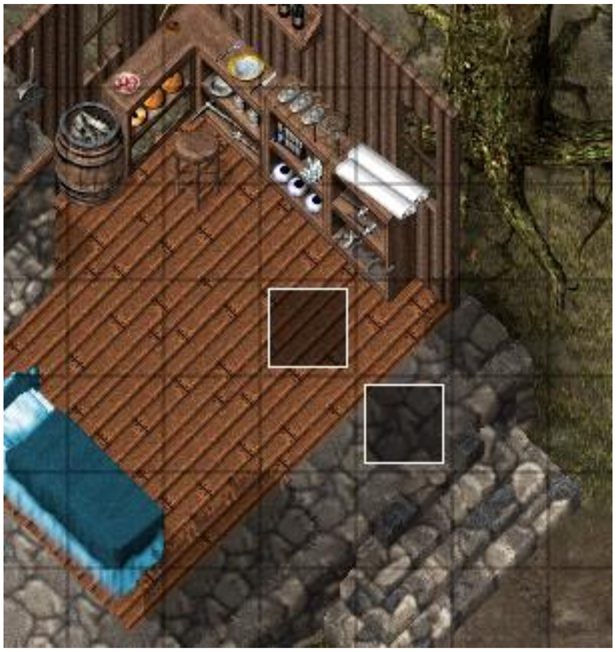
☑ Loop Horizontally

Note

```
<cm:004>
LAYER_S 20 004_Overlay
LAYER_S 14 004_Roof 0 0
```

The format for this note is
<cm:filename>
Where filename will be the name of the collision map image without the .PNG extension.

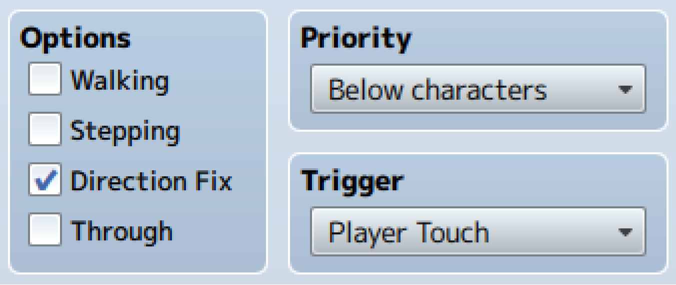
Test play your game again with this added, and your movement should now be restricted, though you will still be able to walk on walls if you move inside the house. That will be fixed in the next step.



The first step is to create a new event inside the doorway of the house that will be used to change the roof layer to transparent and move the walls above the player.

A second event will be placed outside the door to put the roof back in place and move the walls back below the player when going back outside.

Both of these will need to be set up as non-animated "Player Touch" events.



Options

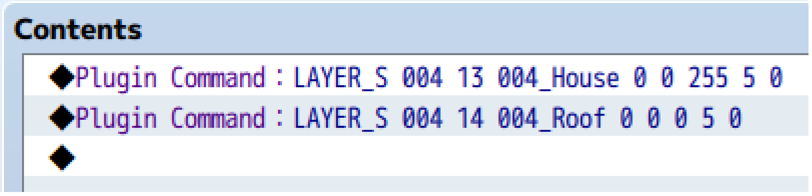
☐ Walking
☐ Stepping
☑ Direction Fix
☐ Through

Priority

Below characters

Trigger

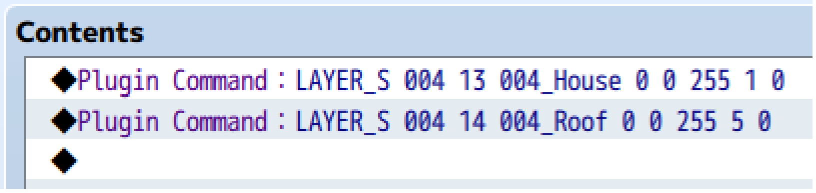
Player Touch



Contents

```
◆Plugin Command：LAYER_S 004 13 004_House 0 0 255 5 0
◆Plugin Command：LAYER_S 004 14 004_Roof 0 0 0 5 0
◆
```

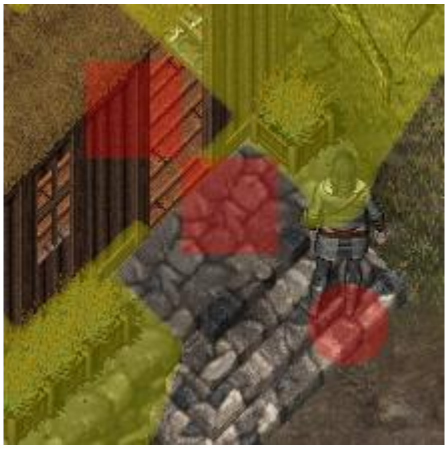
Plugin Commands for the Transfer Inside Event.
- The first changes the layer of the House walls from 1 to 5.
- The second changes the opacity of the Roof from 255 to 0.



Contents

```
◆Plugin Command：LAYER_S 004 13 004_House 0 0 255 1 0
◆Plugin Command：LAYER_S 004 14 004_Roof 0 0 255 5 0
◆
```

Plugin Commands for the Transfer Outside Event.
- The first moves the layer of the house walls back down from 5 to 1.
- The second changes the opacity of the Roof from 0 back to 255.



Run the game to test the events. Depending on your map design, it may be difficult to trigger the events.
While the game is running, press the [F10] key. This will show the Collision Map and all events.

If you were still using grid movement, this would be the best it gets, but with QMovement, the collision boxes for the doorway events can be re-shaped more appropriately.

QMovement Event Colliders

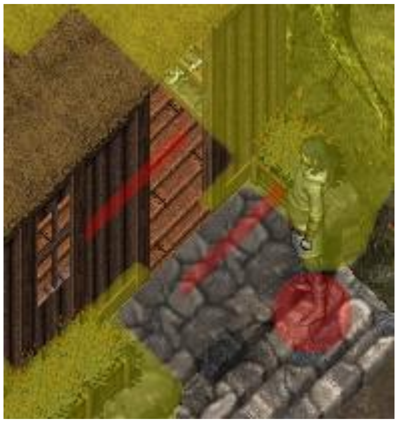
Box	`<collider,box,width,height,x,y>`	Creates a box collider of specified width/height at offset x,y
Circle	`<collider:circle,width,height,x,y>`	Creates a circle/oval collider of the specified size at offset x,y
Polygon	`<collider:poly,(x,y),(x,y),(x,y),(x,y)>`	Creates an irregular shaped polygon defined by the set of provided coordinate points.

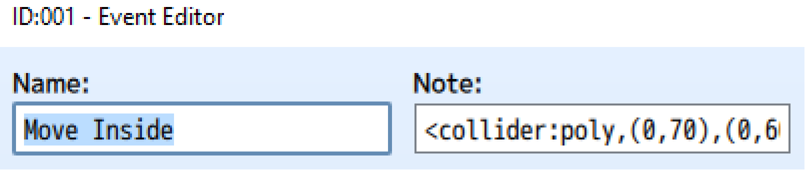
Adjusting The Doorway Events

For this doorway, I will be using the Polygon Collider on each event to create a 10-pixel high line collider to go across the doorway. The collider settings will be added to the Event Note.

```
    <collider:poly,(0,40),(0,30),(50,-20),(50,-10)>
```

You will have to adjust your coordinates to fit your map. They should also be adjusted just far enough apart that the character sprite can fit between the two colliders so the player isn't triggering both touch events at once.
They will also need adjusting so they are triggered in a good position for making the layers adjust at just the right time (not too soon or not too late). This usually takes repeated adjustments of the coordinates and playtesting.

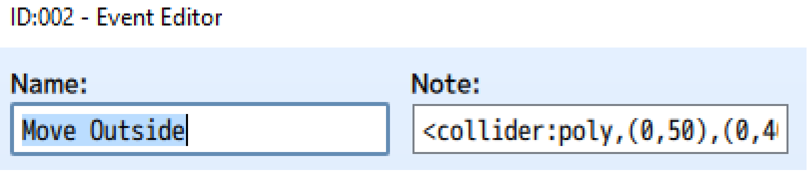




ID:001 - Event Editor

Name: Move Inside

Note: <collider:poly,(0,70),(0,6



ID:002 - Event Editor

Name: Move Outside

Note: <collider:poly,(0,50),(0,4

Once your event colliders are set up, your map will be fully playable, with your character able to move cleanly into and out of the house.