

**Monster Rapist v. 2.0**  
**INSTRUCTION MANUAL**

**Written by**  
**GUS STEVENSON**

**Edited by**  
**JOMB**

## TABLE OF CONTENTS

INTRODUCTION.....	3
Construction of background graphics.....	3
Color palettes for background graphics.....	3
Construction of sprite graphics.....	3
Color palettes for sprite graphics.....	3
GETTING STARTED.....	4
EDITING MODES.....	4
Change Levels.....	4
Controls.....	5
Using the Change Levels Editor.....	5
Level Palette Editing.....	5
Other Editing Features.....	5
Change Column Arrangements.....	6
Change Block Arrangements.....	7
Editing Block Properties.....	8
Change Palette Arrangements.....	9
Editing palette arrangements.....	9
Changing palette color content.....	9
Changing non-round specific palettes .....	9
Editing static screen palette arrangements.....	9
Alter Doorways.....	10
Sprite Workshop.....	11
Arrangement of the Sprite Workshop.....	11
The Tile Menu.....	11
Sprite editing with the Sprite Workshop.....	12
Jomb's Lab.....	13
Enemies.....	14
Normal Enemy Attributes.....	14
Boss Enemy Attributes.....	16
Enemy Hacks.....	18
Change Static Screen.....	19
Known limitations and issues of Monster Rapist.....	19

## INTRODUCTION

Monster Rapist is a modification program for the 1989 NES game *Monster Party*. It allows for editing of level layout, sprite and background graphics, and the positions and destinations of the game's doors. It can also change the color content of the game's color palettes and which palette a particular block of tiles is using. Editing of some game mechanics and boss enemy data is also possible.

### Construction of background graphics

To understand how level editing and editing of game graphics with this program works, it is helpful to understand how graphics in *Monster Party* are constructed. Levels in *Monster Party* are called "*rounds*". Each round consists of a series of "*columns*", arranged horizontally across the screen. Each column consists of 24 "*blocks*", arranged 2 blocks wide by 12 blocks tall. Each block contains 4 tiles, arranged 2 tiles wide by 2 tiles tall. As with all NES games, each tile is 8 pixels wide by 8 pixels tall, with 3 colors and 1 transparent color per tile.

Each round uses separate column and block arrangements for the room and non-room (main level) sections.

### Color palettes for background graphics

Each round contains 4 color palettes for background graphics outside of rooms, with another 4 color palettes for inside the rooms. Some rounds have individual rooms with a different palette still. Each palette consists of 4 colors. For background graphics, palettes are arranged in columns, and may be applied to individual blocks, but not individual tiles.

### Construction of sprite graphics

In contrast to the background graphics, the construction of sprite graphics is not bound by column and block arrangements. Individual tiles can be arranged in any order desired, as dictated by the size and shape of the sprite.

### Color palettes for sprite graphics

Each round contains 4 color palettes for sprite graphics outside of rooms, with another 4 color palettes for inside the rooms. Each palette consists of 3 colors and 1 transparency. For sprite graphics, separate palettes may be applied to individual tiles.

## GETTING STARTED

Start the program. You should see the same charming image featured on this manual's cover. Click on "File", then on "Load ROM". Click through the file menu and open a ROM of *Monster Party*. To avoid saving over the original file, it's probably a good idea to save your altered ROM under a new filename right off the bat (pun intended).

## EDITING MODES

### Change Levels

This editing mode shows the full layout of background graphics in each round and room. Each area of the game is made up of the arrangement of columns. Each column of blocks is labeled A-P. This labeling is used by the "Enemies" editing feature (see below.) All of the columns which compose the current level are displayed in a separate window at the bottom of the screen. This mode also features a series of controls that are outlined below.

## CONTROLS

**Beg / End** – Shifts the level viewer to the start or end of the level, respectively.

**<< / >>** - Shifts the level viewer by one screen.

**< / >** - Shifts the level viewer by one column.

**Toggle Palettes** – shifts between the four background color palettes, and chooses which is used within the column viewer.

**Toggle Palette View** – Shows/hides which color palette is in use by each block within the level's columns, as per each column's palette arrangement.

**Toggle Palette Editing** – Toggles between column and palette editing. Column

editing allows selection and placement of the current level's column arrangements. Palette editing allows selection and placement of the current level's palette arrangements. (Neither the column nor palette arrangements themselves can be edited in this mode. They can be edited in the column and palette arrangement modes, respectively.)

**Reveal Doorways** – Reveals or hides the doorways in the game's programming. When revealed, the doorways are highlighted by a white border and the door's corresponding number. (See below for instructions on moving doors.)

**Toggle Enemy View** – Reveals/hides enemy positions and items they drop. (See below for instructions on how to edit enemies using this feature.)

## Using the Change Levels Editor

Columns within the level can be replaced by others within that set of columns. Select a column in the bottom display, and click on a column in the level in the top display. The currently selected column is shown in the bottom-right display.

## Level Palette Editing

After changing a column, you'll probably need to change that column's palette arrangement as well. For that, click the "Toggle Palette Editing" button, select the appropriate palette arrangement in the bottom display, and click on the column in the level in the top display. The "Toggle Palette View" button can be used to show which palette each block is using.

## Other Editing Features

### To move a door:

1. Click on "Reveal Doorways".
2. Select a door from the "Move This Door" drop-down menu.
3. Use the directional buttons to move the door to the desired location.  
If there are no door blocks in said location, replace the current column with one that has a door there, or use the column editor to edit the column to place a door there.

**NOTE: Doors do not work without door blocks.** Also, if an extra door graphic is added without a door being moved to that location, it will be purely

cosmetic. (Which blocks serve the door function is editable in the "Block Arrangement" editor.)

NOTE: The exit doors at the end of each level do **not** need to be moved with this feature. The blocks of those doors are programmed to end the level when used after the key has been obtained. Only the blocks need be moved. (Which blocks serve as the exit door can also be edited in the "Block Arrangement" editor.)

### **To edit enemies and the items they drop:**

1. Click on "Toggle Enemy View".
2. In the "Enemies" menu section, select the block position (A-P) that you want to edit.
3. Select the desired preset number. Enemy, Y Coordinate, and Item for each preset are shown in the green boxes below the menus.

NOTE: Selecting an enemy in the presets that is not on the current round will likely result in it being composed of a jumbled mass of tiles. This is because the game is currently using the wrong tilesheet for that enemy.

NOTE: Only three enemies will appear on the screen at a time, regardless of how many are put in an area.

## **Change Column Arrangements**

This editing mode allows editing of each individual column arrangements that compose each level. Columns are displayed in sets of three, and can be navigated using the tall buttons on either side. Each column consists of 24 blocks, arranged 2 blocks wide by 12 blocks tall.

The window on the bottom shows the blocks that compose every column on the current level. Blocks within a column can be changed by selecting a block in the bottom window, and clicking on a block in the column viewer. The currently selected block is shown in the bottom-right display.

The "Toggle Palettes" button shifts between the four background palettes, to choose which is used within the two displays.

**NOTE:** Some columns may be used more than once in the level. Editing a column will change every instance of that column in the level. Replacing the other instances of the column with similar ones that also match up with neighboring columns may be one solution.

## **Change Block Arrangements**

This mode edits the blocks of tiles that compose the columns. Each block contains 4 tiles, arranged 2 tiles wide by 2 tiles tall.

The top display shows each block arrangement used on the current level. The tileset used by the current level is displayed on the bottom. Each tile can be selected and placed in the desired block.

The "Toggle Palettes" button shifts between the four background palettes, to choose which is used within the two displays. The "Toggle Block Type View" button reveals/hides the controls for editing the properties of each block (see below for instructions.)

**NOTE:** Editing blocks will change every instance of that block in the level.

## Editing Block Properties

The way each block behaves when the player touches it can be changed.

### To change block properties:

1. Click on "Toggle Block Type View" to show the controls.
2. Click on the desired control button.
3. Click on the block you want to change.

The controls and their functions are as follows:

**No block property** – nothing happens when the player touches the block.

**Water** – the player wades through the block, as if it's made of water.

**Platform** – the player can stand on the block, and jump through it from below.

**Ladder** – the player can climb up the block.

**Solid** – the player can stand on the block, but can't jump through it from below.

**Spikes** – the player takes damage by touching the block.

**Regular doorway** – used to enter a room. (Requires a doorway. See the "Change Levels" and "Alter Doorways" editing modes.)

**Round end door** – ends the round if used after the key has been obtained.



## **Change Palette Arrangements**

This editing mode can change the color content of each color palette in the game, and can change which palette a particular tile or block of tiles is using.

Four color palettes are used at any one time for the background graphics, and four more for the sprite graphics. Each palette contains four spaces – the first for the transparent color, followed by the three colors. Each level of the game has a separate set of color palettes for room and non-room areas.

### **Editing palette arrangements**

Color palette data for background graphics is arranged in columns, similar to the columns used to arrange the blocks of background tiles. These palette arrangements are shown in the top-left display, and, as with the column arrangements, are displayed in sets of three. The individual arrangements can be navigated using the long buttons on either side. The four background palettes can be selected using the buttons marked 1-4 to the lower right of the palette arrangements display. To edit the palette arrangements, select the desired palette and then click on the blocks you want to change.

### **Changing palette color content**

To the right of the palette arrangement display are the four background palettes and sprite palettes used in the current section of the game. In the bottom display are all 64 colors used by the game. To change a color in the color palette, click on the color that you want to change. That color will appear in the bottom-right display. Then click the desired color in the bottom window.

### **Changing non-round specific palettes**

At the bottom of the display is a selectable list from which you can select any palette in the game. Doing so will change the current palette to the selected palette, allowing you to easily edit it's color content. Hit the Revert Palette button to go back to the palette being used by this area.

### **Editing static screen palette arrangements**

If you are editing a static screen (the title screen for example) there will be a shift up and shift down button present to allow you to view up and down the palette arrangement of the screen. Static screens do not use blocks, so you are editing

individual tile properties here.

## **Alter Doorways**

This mode allows for changes to the programming of the game's doors. While the "Change Levels" editing mode can be used to edit the *position* of doors on each level, this mode controls what happens when the doors are used. It contains a series of drop-down menus that affect various attributes.

### **Select Round/Select Door**

Used to select the door you wish to edit.

### **Destination Type**

Controls how the area you go to behaves and which sprite sheet it uses. Be careful changing this value, as it can cause bizarre behavior and crashes if set to the wrong things when combined with other options.

### **Set Screen at**

Controls where the camera is centered if the door leads back into a main level, by setting the X coordinate of the level by block number. For example, if this is set to 15, the screen starts with block 15 on the left. This is used often for the doors that teleport you around the level (such as Round 6). For entering a room, this is always set to 0.

### **Door Contents**

Controls what is in the room. To teleport around on a stage, this must be set to "Not a room". If you select a boss that does not exist on the current round, the room will either be empty or behave strangely. If you have more than one doorway lead to the same boss, only the first that is entered will actually lead there, the others will be empty. If you don't include doors to all bosses, it is possible to become stuck on the round.

### **Door Leads to**

Controls what screen on the map the door will lead to. This is often used to designate rooms. For example, screen 0 is almost always the empty room, screen 1 is usually the first boss room, and so on. In non-room areas, if this value is set to anything other than 0, it may prevent the player from walking to screens earlier than this and allow the player to walk off the end of the map.

### **X and Y**

Controls the X and Y coordinates that the player starts at when entering the screen that the door leads to. As with "Set Screen at", these variable represent block numbers. For example, setting X to 4 and Y to 3 sets the player 3 blocks down and 4 blocks right from the upper left corner of the screen. Be careful not to make the player appear inside a wall or below the floor in the status bar, or the game may become stuck or lock up.

**NOTE: A spreadsheet that outlines the stats for each door in the game is**

included with this program's documentation.

## Sprite Workshop

The Sprite Workshop allows for editing of the game's sprites. This is probably the most complex area of the program.

**NOTE:** Tile arrangements for sprite graphics are *not* limited to columns and blocks, in contrast to background graphics. Also, individual color palettes can be applied to individual tiles.

### Arrangement of the Sprite Workshop

At the top are drop-down menus for selecting a sprite sheet and sprite number. Each round and the rooms of each round have at least one sprite sheet. The sprites for the player characters and the items the player can collect have a separate sprite sheet of their own.

The top-left display shows the assembled sprite currently selected in the "Sprite Number" menu. To the right of this display is the Tile Menu, which lists and allows changes to each tile used in the sprite. The bottom-left display shows every tile within the current sprite sheet. The bottom-right display shows the currently selected tile.

There are also controls for removing, adding, and shifting sprite tiles (see the "sprite editing" section below). Just above the Selection window are the "Toggle" buttons. The "Toggle Tilesheet" button does not actually change which tiles the sprites use. It just shows what the sprite would look like with a different set.

### The Tile Menu

The tile menu is a radio button menu that lists each tile that makes up the sprite, arranged by tile number. This menu shows and allows editing of:

- Which tile from the tileset is drawn (represented by a picture of the tile)
- How the tile draws (normal, background, mirrored, etc.)
- Which sprite palette the tile uses

**NOTE:** Background priority mode means that the tile is drawn behind every color except color 1 (usually black), giving the effect of walking behind things.

## Sprite editing with the Sprite Workshop

Using the Sprite Workshop, sprite graphics can be edited in various ways – by moving tiles, changing the tiles that the sprite is composed of, by changing the palettes used by the sprite, and even adding or removing tiles from the sprite.

### Changing a Tile:

1. Select the tile you want to change in the radio button menu.
2. Select a tile from the tileset in the bottom-left display.
3. Click on the picture of the tile in the radio button menu.

### Moving a Tile:

1. In the radio button menu, select the tile you want to move.
2. Use the "Shift Selected Tile" navigation controls to move the tile to the desired location.

### Removing a Tile:

1. Select a tile in the radio button menu.
2. Click the "Remove a tile" button.

### Adding a Tile:

**NOTE:** This can only be done if tiles have been removed from the same sprite.

1. Click "Add a tile" button.
2. New tile should appear in upper left corner. (It appears as the first tile in the tileset.)
3. Select the tile from the tileset that you want the new tile to use.
4. Click on the picture the new tile in the radio button menu to change it.
5. Move new tile to the desired location.

## **Jomb's Lab**

Jomb's Lab contains several drop-down menus that allow for changes to various game mechanics. This allows for changes with the way certain aspects of the game behave. However, several of the options can cause the game to crash, especially when combined with other options.

Some miscellaneous game play hacks are also included in this section, such as being able to change how Mark/Bert animate, how much life a heart is worth, how much life you start the game with, how much life you get for winning a round, how many bosses must be defeated on any given round to obtain the key, control of how Bert's twin-shot behaves, etc.

## **Enemies**

This editor contains drop-down menus that allow for changes to enemy data. As of this revision, some parts of this section are still experimental and not well understood.

### **Normal Enemy Attributes:**

#### **Life**

Number of times the enemy needs to be hit before it is defeated. (Setting this to 0 is identical to setting it to 1.)

#### **Collision X/Y**

Collision detection of the enemies in this game is independent of their sprite. A rectangle is defined, and that's where you have to aim your attacks and that's where you get hurt if you bump into. It's done in a very peculiar manner. From the point of origin there is an attribute that defines how many pixels the collision field extends on the X axis, IN BOTH DIRECTIONS, then one that does the same for the Y axis.

#### **Animation Frame/Sprite Used**

Animation data for enemies is on a frame by frame basis. This gives you complete control of which sprite is called for each frame of animation. The teleporting skeleton on Round 3 and the pod on Round 2 do not seem to use this animation attribute. They have entries but changing them does nothing. The monster from Round 7 with the changing heads has 2 separate sets of animation.

#### **Points**

Of course, everyone knows this is the most important trait, because points are so vital in this game. Sometimes, selecting 9999990 will cause a gibberish score.

#### **Enemy with Projectile**

Besides the base enemy stats, there is a second set of stats related to enemies that fire projectiles. Select an enemy below to alter its projectile stats.

#### **Style**

You can change the style in which an enemy fires projectiles. Some enemies behave strangely with certain styles, or the style fails. For example, when an enemy is set to alternating down and down at angle, some enemies interpret this as just shooting straight down.

#### **Animation Frame/Sprite Used**

Here is the animation data for the projectiles on a frame by frame basis. Note that the three varieties of the Constellations on Round 8 all use the same animation data for their projectiles. Also, as with any animation, if you call a

sprite number which is too high you will either get gibberish or a game crash.

### **Arc Projectile Time, etc.**

Some projectile types also have information stored for how they move. The arc and straight down style projectiles work like this: At each "clock tick", there is a value for how far downward the projectile moves in pixels. This value wraps around the screen so a high value is actually making the projectile move upwards. The projectile moves on the x-axis independent of the values used here. The projectile will continue using these values until it runs into something which stops it, or goes far enough off the screen. If you keep the projectile on screen until it runs out of this data, it will start using whatever data is next in the ROM.

### **Angle 1-4**

Additionally, the alternating projectile (used by the constellations on Round 8), also has four values which determine which angle each shot travels at. I'm not really sure how the value used here maps to actual angles. 0 is straight down, 8 is about 45 degrees to the right, 249 is about 45 degree left. The change is gradual within this range. Outside of this range the angle can be unpredictable.

### **Enemy Presets**

This edits the enemy presets used in the "Enemies" section of the "Change Levels" editor. Each preset consists of a type of enemy, the Y coordinate the enemy starts at, and what item the enemy drops after being defeated. The Y coordinate is measured in blocks, with 0 being the top of the screen. If an enemy starts below the round in status bar, the game is likely to crash or glitch up.

**NOTE: Changing a preset will also change any other enemies that use the same preset.**

## **Boss Enemy Attributes:**

### **Life**

Number of times the boss needs to be hit before it is defeated.

### **Boss X/Y Coordinate (in pixels)**

Determine where the boss appears in the room.

### **Number of Animation Frames/Animation Timing**

Determine the number of frames the boss uses, and the animation speed.

If you set the number of frames to more than it currently uses, the extra frames might be sprites from other bosses, or sprites comprised of gibberish. If you make this value too high the game may crash, depending on how other attributes have been set up and which boss you are working on. Be sure to test what you've done to make sure it plays.

The animation timing variable for the bosses is somewhat unpredictable. The properties were named based on what was observed.

Certain bosses get attacks based on when a certain sprite is shown, so if you remove that sprite the boss will get no attacks, and if you make the sprite appear at a different rate the bosses rate of attacks will correspond (observed in the plant and cat bosses)

These two boss properties seem to be unused on certain bosses. They were included anyway, since they seem to have an entry in the data field, but you may find a couple of bosses which these values have no effect on or have an unexpected effect.

### **Speed/Other Movement Attribute**

The movement rate of bosses is controlled by two variables. How this works is not yet fully understood. The first trait seems to be a pretty straightforward speed rating. Making it 0 usually means the boss does not move, 1 it moves slowly, 5 it moves fast, etc. But, it can become unpredictable if you use a very high number.

The second movement attribute is a bit of a mystery. It also seemed to control the speed to some degree, and caused movement in the opposite direction a lot in test cases. How these 2 attributes play on each other is something you will have to experiment with. If you figure it, please email Jomb (email address is on the last page of this manual.)

A final note... Changing these attributes for bosses which are normally stationary seems to do nothing. (Email Jomb if you find out differently.) Also, changing these values for the dragon boss resulted in very odd behavior. Instead of moving faster it started broken up in fragments all over the screen



then reformed itself and behaved normally.

### **Base Sprite**

The following attribute is the base sprite for the given boss. Changing the base sprite will also change the animation, because unless otherwise specified, animation is taken from consecutive sprites after the base one.

When changing this value, be advised that if you change this to a sprite not on the current sprite sheet it will show up as gibberish (though with some work you could use this to remap which sprite sheet a boss uses). If you change this to a non-existent sprite you will either get gibberish or a game crash.

There are some exceptions to this attribute. Certain bosses seem to always retain their base sprite no matter what you change this to, though the rest of their animation will proceed with this new value. This was observed in the fast food boss, and in background bosses once made visible.

### **Animation Frame/Sprite Used**

For the bosses that animate, there is also animation data on a frame by frame basis. This gives you complete control of which sprite is called for each frame of animation. However, this may not work right if mixed with changing of the base sprite above, or changing of the number of animation frames. Try it and see what happens if you desire. When changing animation sprites this way, you can't select a sprite off the current sprite sheet, and if you select a sprite number which is too high and doesn't exist, the game will crash when you enter this room.

### **Damage Palette**

When a boss is damaged, it responds by flashing colors. This is actually done by briefly changing the palette for the whole room. The reason only the boss seems to flash is because a palette is chosen where only the boss colors are different. Using this attribute, you can select which palette to change to when the boss is damaged. If you want no flashing, try to select the regular palette for the room.

It is possible to select a palette which does not exist in the game, if you chose a number that is too high. This won't crash the game; the game will just try to make sense of whatever data it ended up with. Usually this leads to lots of black being used.

### **Defeat Palette**

When a boss is defeated, the palette of the room changes. This seems to be done mostly to hide the bosses that are drawn in the background, creating the illusion that they vanished after you beat them. Use this attribute to change which palette is used on boss defeat.

### **X/Y Collision**

Collision detection of the bosses in this game is independent of their sprite.

Basically, a rectangle is defined, and that's where you have to aim your attacks and that's where you get hurt if you bump into it. It's done in a very peculiar manner. From the point of origin, there is an attribute that defines how many pixels the collision field extends on the X axis, IN BOTH DIRECTIONS, then one that does the same for the Y axis.

These attributes sometimes behave strangely at very low numbers. For certain bosses, they will always be at least size 1 even if you make them 0. For other bosses you can make them invincible by making one or both of these values 0. A few bosses become invisible and invincible if one or both of these are set to 0.

### **Points**

Finally, what you've all been waiting for, how many points you get for defeating the boss. Everyone knows this is the most important trait because points are so vital in this game. How this hex value maps to actual point value seems wildly unpredictable.

### **Boss projectile characteristics**

As of version 2.0, it is now possible to further alter boss projectiles. You can change what type each boss has, how it will behave, the collision data for the projectile, which sprites it uses and how it animates.

### **Enemy Hacks**

Also as of version 2.0 almost all bosses and enemies have hacks that can be applied to them similar to the hacks in the Jomb's Lab section of Monster Rapist. All manner of strange and unusual behavior can be applied, and in some cases movement patterns controlled. Many of these hacks are experimental however, and care should be taken when applying them, as they can act unpredictable in combination with each other, and can lead to game crashes. Make sure you thoroughly play test your game after applying any of these to make sure there were no unintended consequences.

## Change Static Screen

This section of the editor allows you to rearrange the tiles of any of the non-scrolling screens found throughout the game (the title screen, password screen, etc.). Simply select a tile then click where you want to put it. Use the Up/Down buttons to scroll up and down in the larger screens which don't all fit on the screen at once.

## Known limitations and issues of Monster Rapist

So far, this program does not allow for direct editing of the game's tiles or in-game text. For that, you'll need to use a separate tile or hex editor, respectively.

The game will crash if a level is arranged so that a black pixel exists on the very bottom line of pixels across the screen. When the screen scrolls, it will lock up. (This may have to do with the status bar.)

Editing doorways can be tricky. If you overwrite doorway tiles, the door becomes inaccessible, possibly causing a round to be unbeatable. Also, if you add an extra doorway without moving a door to that location, it will be purely cosmetic.

When editing sprites you can rarely encounter a tile which will crash the game if altered. Even if altered in a separate tile-editor. This can be worked around by layering another tile over it. What is most likely happening here is the game is using this tile as the last tile check before the status bar, and it's alteration causes the level to roll over into the status bar, crashing the game.

There are a couple special palettes which cannot be edited using Monster Rapist, because they are nonstandard. For example the palette used in the ending cut scenes is not fully editable using this editor. The missing palette fragment can be found fairly easily using a hex editor however.

The palette arrangements of the "window" part of the intro and ending cut scenes can not be edited in Monster Rapist. This is simply because they were handled differently than all the other palette arrangements in the game, and I have been unable to locate them in the ROM.

Certain features of this editor can cause a lag. This is due to the program loading up all the graphic data. Be patient, it will do as you requested.

## **Questions, comments, want to bitch Jomb out?**

You can contact Jomb at [nnnmmm1134@yahoo.com](mailto:nnnmmm1134@yahoo.com).

To make sure you have the latest version of this program, visit [badderhacks.net](http://badderhacks.net), the official home of Monster Rapist.

**Monster Rapist is Copyright © 2011 by Jomb.**

**Contents of this manual are Copyright © 2011 by Gus Stevenson and Jomb.**

**Monster Party is Copyright © 1989 by Bandai. All rights reserved.**

**Special thanks to Dr. Floppy for miscellaneous help and guidance on this project.**