

# World Machine 2 User's Manual

## ***Table of Contents***

1. Introduction & Concepts
2. Devices and the Device Workspace
3. Project Setup & Render Extents
4. Terrain Views
5. Layout Generator
6. File Input and Output
7. RGB Textures
8. Macros
9. Program Config
10. Professional Edition Addendum

## **1. An Introduction to World Machine 2**

World Machine 2 is the successor to the popular World Machine terrain editing tool. It radically expands the available features and abilities while keeping all of the power that made World Machine unique.

### ***Proceduralism: The fundamental difference***

Forget what you know about editing terrains. Unlike many other terrain editors that function like paint programs, World Machine operates on a procedural level. A World Machine world file is not the terrain, but the **steps** that creates the terrain. This is the source of both its power and its complexity.

When working in World Machine, your role is to determine the overall appearance and characteristics of the terrain, and then allow the generators and effects to create and detail the appearance of your world. This way of working and thinking at a higher level takes some getting used to, but the rewards are huge: You can create impressive terrains with relatively little work, and once you have created a terrain that you like, you can use the same world file with different random "seeds" to produce a different terrain with the same feel.

The tradeoff in computer graphics for this is usually a lack of control over terrain feature placement; but WM2's Layout Mode allows you to work in a more artistic fashion, plotting out the terrain as roughly or precisely as you like. The ability to determine for yourself the level of control you wish is one of the great new abilities available to you as an artist.

### ***Concepts you should know***

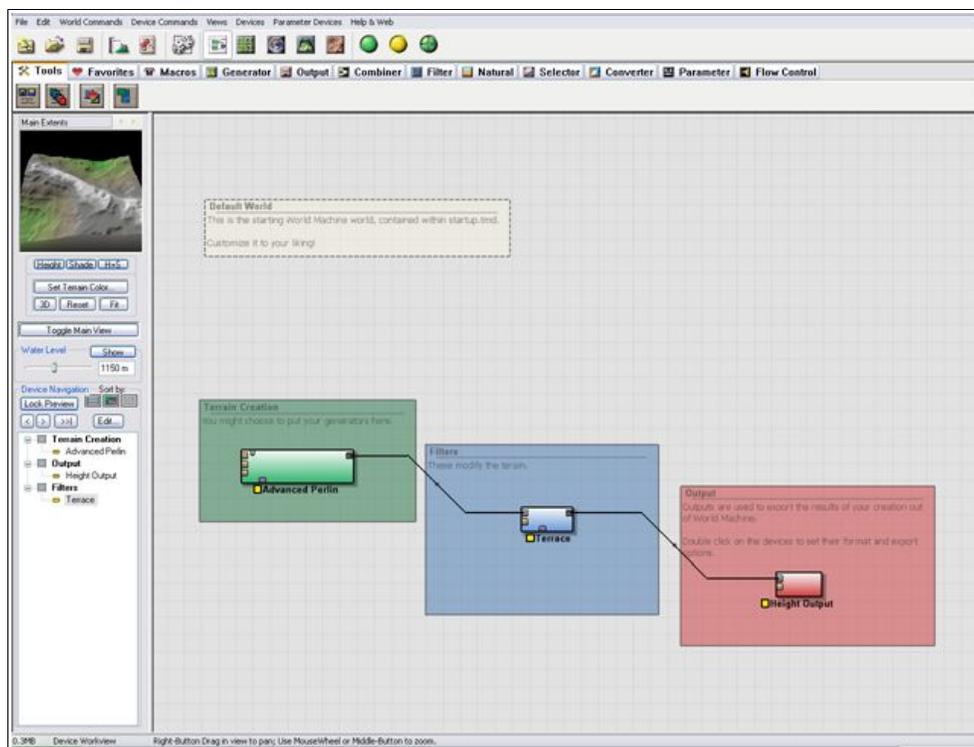
World Machine operates on *worlds*. A world is simply the collection of steps used to create the terrain, as well as information about areas to view, detail level, and so on. These steps are referred to as

*devices*, and each device will perform some operation on the world. The *workview* is where you can see and change the devices (steps taken). World Machine retains a preview of each device at all times that is displayed when that device is selected. To create a final product, you must *build* the world, which creates a high resolution version of the world according to the instructions that the devices provide.

In the next chapter, you'll learn how to use the device workview and interact with devices to create your world.

## The Starting World

When you first start World Machine, you'll see a screen like the below:



Each aspect of the screen above will be visited in great detail in later chapters. The most important point is to understand how each of the basic concepts relates to what you can see:

- The *world* is the contents of the device workview (the screen visible above)
- The *devices* are the small colored blocks; there are three of them in the photo above.
- The black lines between blocks are wires, showing the order that actions are taken in and the way they're combined.
- World Machine is displaying a preview of the currently selected device in the upper-left corner
- To *build* the world, you must choose the Build option from the menu or toolbar, which will

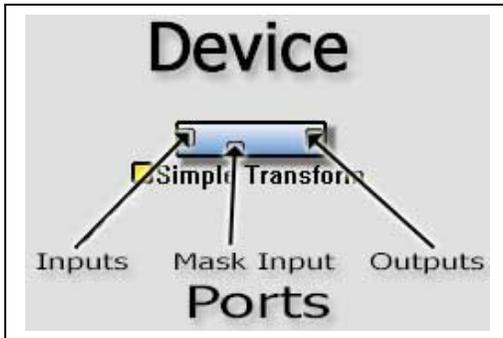
bring up a progress report dialog while calculations are being performed. Once the world is built, you can then export the terrain as mentioned in [Chapter 6](#).

You can select any device and see what the terrain looks like at that point by clicking on it. If you like learning by exploration, there is a large collection of various example files sorted by difficulty in concepts in the "/Examples" folder under your World Machine installation. Otherwise, continue reading below about how to use the Device Workview and what it all means.

## 2. Devices and the Device Workspace

### 2.1: Devices

A **Device** represents an *action*. In World Machine, devices are visualized as small, colored boxes:



A device has **Ports**. A port is simply a place where terrain data can move in or out of a device. These can be divided into two main categories: **inputs** and **outputs**. Inputs are on the left side of the device, and bring data into a device to be acted on. After the action of the device takes place, the data is then taken out of the device via an output, on the right side. Lastly, some devices have a *mask input port*, which allows you to control the device's area of influence.

There is one other type of port in World Machine, called a parameter port. These are located along the top edge of the device, and allow advanced users to adjust the settings of a device from a different one. Use of Parameter Ports is covered later in the manual.

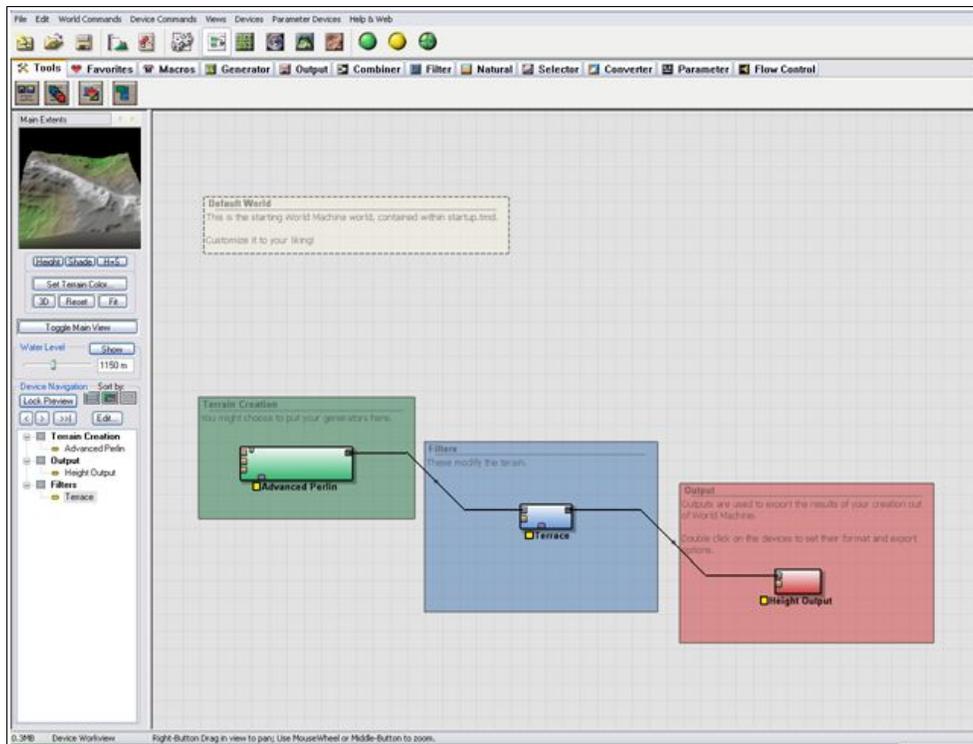
You can separate devices into three broad categories: In the World Machine system, terrains are:

- Created** by **Generator** devices;
- Modified** by **Filter & Combiner** devices;
- Saved** by **Output** devices.

Each type of device has a different responsibility. Generators are devices that produce the basic terrain. The output of a generator can be either used directly, OR routed into the input of a Filter device to shape, change, and otherwise influence the terrain. Finally, Outputs are used to save the created terrain to disk.

### 2.2 Working with Devices

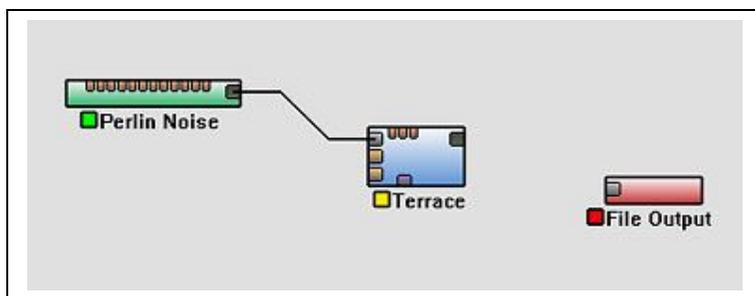
The Device Workview is the primary means of interacting with World Machine. It contains all of the devices in the current world, and the connections between them.



The Device Workview can be selected from the [Command Bar](#), or by pressing F5.

### Device Build Status:

Devices in World Machine always have a build status; this tells you if the device is connected properly and if it has been built yet.



Pictured above are three devices in various build states. The build state is shown by the colored rectangle just to the left of the device name.

- **Green Status:** The device is properly connected and Build results exist for the device.
- **Yellow Status:** The device is properly connected, however it has not yet been built.
- **Red Status:** The device is unconnected, or its inputs are otherwise not sending data.

You can make a device go from *red* to *yellow* status by connecting any required inputs.

You can make a device go from *yellow* to *green* by building the world with the [Build World Command](#).

## Adding Devices:

To add a device to the world, you can select one from either the menu bar, or the parts toolbar. To select a device from the menu, simply go to *Heightfield Operators*, select the category of device you wish to add, and then a device from that Category.



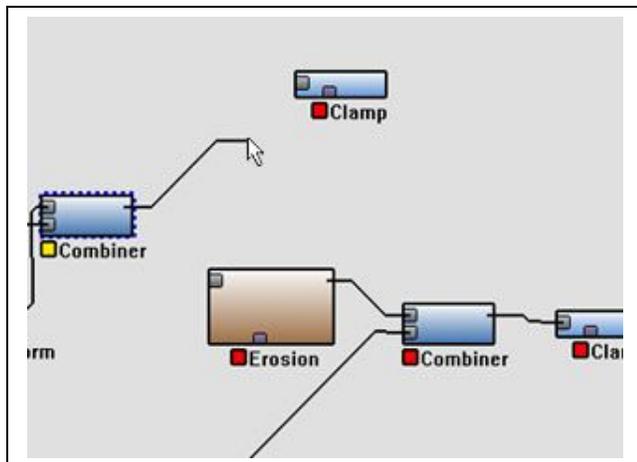
Alternatively, and more conveniently, you can select any device from the parts toolbar.



Once a device is ready to be placed into the world, the mouse cursor will have a + sign next to it in the workview. Clicking the left mouse button anywhere in the workview will place that device there. You may place multiple devices of the same type simply by clicking multiple times in an empty region of the workview. You may stop placing devices by either right clicking, or clicking on any device present in the world.

## Connecting Devices:

To connect two devices together, you must wire an output port on one device to an input port on another, or vice versa. To do this, click on one of the ports. A wire will become attached to that port on one end, and the mouse cursor on the other. All ports will disappear except the ports that you are allowed to wire that device to.



Once you have clicked on both ports there should now be a wire connecting the two devices, if there is not then either you attempted to wire an incorrect combination or you missed the port you were trying to connect to.

### **Multiply-Connecting an Output (Splitterless Networks)**

You can connect a single output port to many different devices, eliminating the need for the Splitter device of World Machine 1.x. The default behavior is the same as before, however, if you connect the input of the next device to the output port in question (rather than vice versa), it will not disconnect the

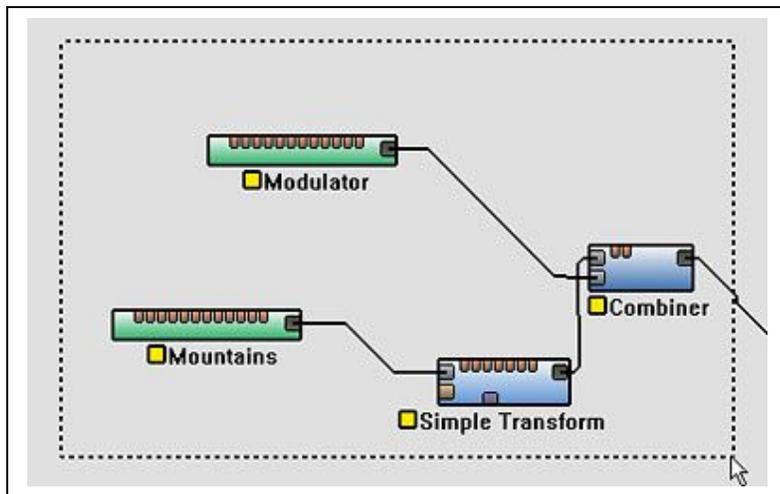
### **Editing a Device's Properties:**

To change the properties of a device, simply double click it in the workview. A dialog will appear that allows you to change the parameters and settings of the device. The Leftside preview will show you the results of any adjustments you make.

### **Deleting Devices:**

To delete a device, simply select it and then choose Delete Device from the Edit menu, or simply hit the Delete key. By default, you will be prompted to confirm the decision to delete the device – if you wish to avoid this prompt, hold the shift key while you hit Delete.

### **Selecting & Moving Devices:**

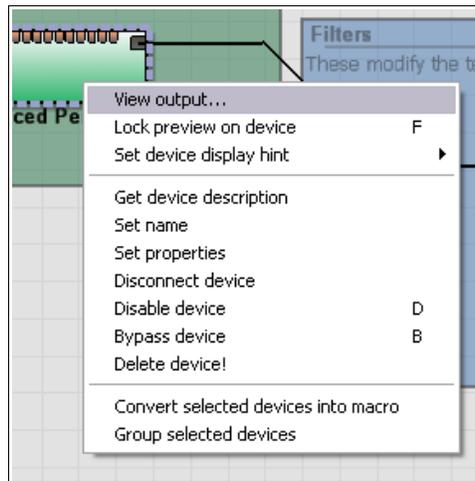


- You may select a single device by simply clicking it.
- You may add another device to the selection by holding **SHIFT** and clicking on that device. Clicking a selected device with shift held will toggle it out of the selection.
- You may select multiple devices at once by drag-selecting a group of devices, by holding down the mouse button and drawing a box around the devices you wish to select.
- Holding down shift while drag-selecting will add the drag-selected group to the currently selected devices.

You may move any selected device(s) by simply clicking and dragging on the device.

## Device Context Menu:

Right clicking on any device will bring up the device menu:



- **View output:** Swaps the main view to the last-used view (2D or 3D view) to view the build results at this device.
- **Lock Preview on Device:** [Freezes the preview](#) on the currently selected device.
- **Set device display hint:** Control how WM displays the terrain. See the section below for more detail.
- **Set Name:** Renames the device.
- **Set Properties:** Sets the properties of the device (same as double-clicking).
- **Disconnect Device:** Remove all wires from this device so that it is isolated from the rest of the network.
- **Disable Device:** Disables the device. A disabled device is grayed out, and will not be activated when the world is build. Any devices depending on its output will also fail to build. Selecting this again will re-enable the device.
- **Bypass Device:** Bypassing a device temporarily makes the device perform no action. This can be very useful for judging the effects of a device on a network by toggling bypass mode on and off for that device.
- **Delete Device:** Deletes the device.
- **Convert Devices into Macro:** Converts the selected device(s) to a macro. Any links leading into or out of the collection of selected devices are converted to macro ports. See Chapter 8: Macros for more information.
- **Group Devices:** Create a new group around the currently selected device(s). See Chapter 2.3 below for more information on groups.

## Copy & Paste:

You may cut, copy, and paste any set of devices you wish with the standard Windows keyboard shortcuts (Ctrl-X, Ctrl-C, and Ctrl-V respectively).

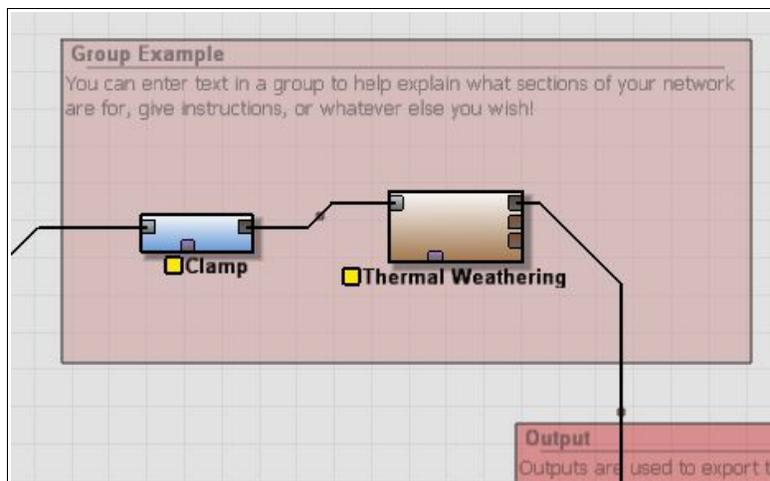
## Terrain Hints:

World Machine allows you to provide a hint as to how the heightfield should be displayed. The two options are to display the output as a terrain or a mask; you should choose whichever option makes the most sense for the device in question. Many devices will automatically flag their output appropriately.



## 2.3 Groups

Groups are a way to logically organize your device network. Groups are named, and you can add a description to annotate a particular set of operations.



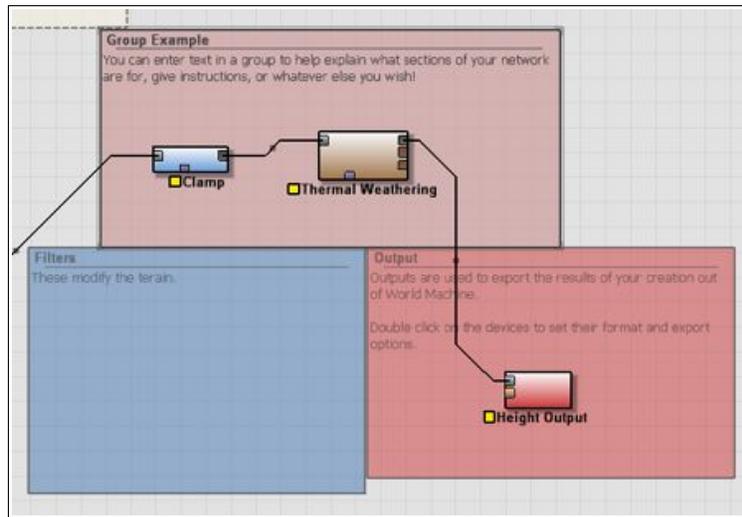
### Group Creation

To create a group, you have two options:

- Select a number of devices, then right click on them and choose "Create Group" from the context menu.
- Use the Grouping Tool to draw a group in the device worldview (described below)

## Group Manipulation

When you move a group, all the devices inside of it are moved as well. Moving a solid group will push all other solid groups encountered out of the way.

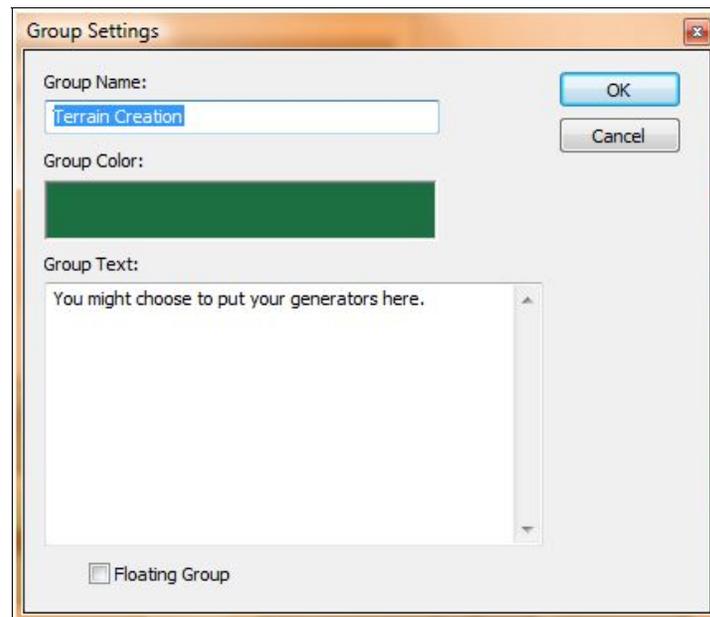


## Solid vs Floating Groups

Floating groups, on the other hand, will not interact with other groups in any way, and so are very useful for text annotations. A solid group has a solid border. A floating group has a dashed border.

## Editing a Group

To edit a group's properties, double click on the group to open the Group Settings dialog:



This dialog allows you to set a group's name, text, color, and floating status.

## 2.4 The Device Toolbar

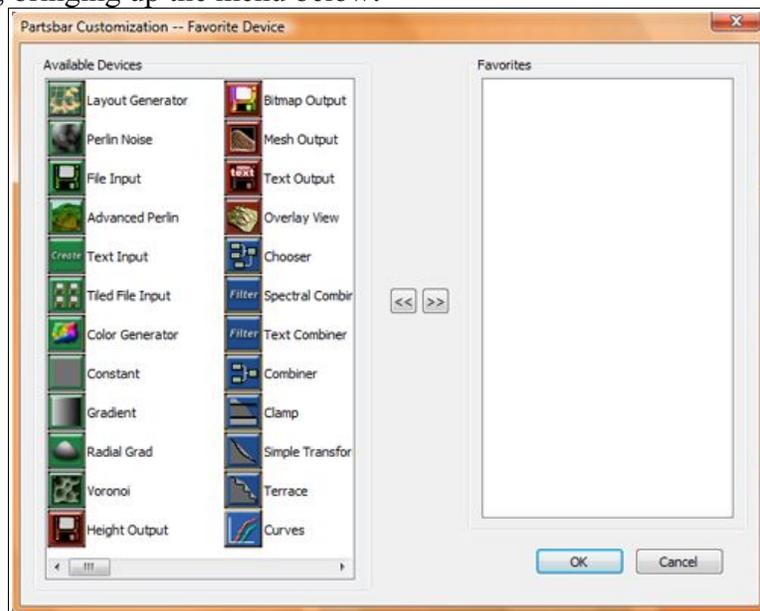
The device toolbar consists of the *Tools* section, the *Macro* and *Favorites* section, and an additional set of tabs organizing all of the available devices by a number of different schemes.



- **Sort by Data Type:** Creates several tabs based upon the *data* type; For example, Heightfield versus Bitmap.
- **Sort by Device Type:** Creates tabs based upon the *device* type; For example, Generators, and Filters.
- **Set Favorites:** Enter the Define Favorites dialog, described below.
- **Grouping Tool:** The grouping tool allows you to create groups by visually drawing their extents into the workspace. To use, simply select it then drag a selection around the area you wish to group in the workspace.

### 2.4.1 Favorites List

You can define a "Favorites" section of the toolbar to contain a custom toolbar of your most frequently used devices. To define the favorite devices, click on "Define Favorites List" button from the Tools section of the toolbar, bringing up the menu below:



To add a device to the toolbar, find it in the list of all devices on the left side, then click the right-transfer button (>>) to add it to the favorites.

Similarly, to remove a device from the favorites, select it and push the left-transfer button.

The selected devices will appear under the "Favorites" section of the toolbar.

## 3. Render Extents and Project Setup

### About Render Extents

Although the devices you connect determine the look and feel of your world, it is the render extents that define the area of the world that you are working in. In this way, the render extents control what you're looking at. You can zoom down to a very small area of a few meters, or cover a vast thousand-kilometer region of terrain.

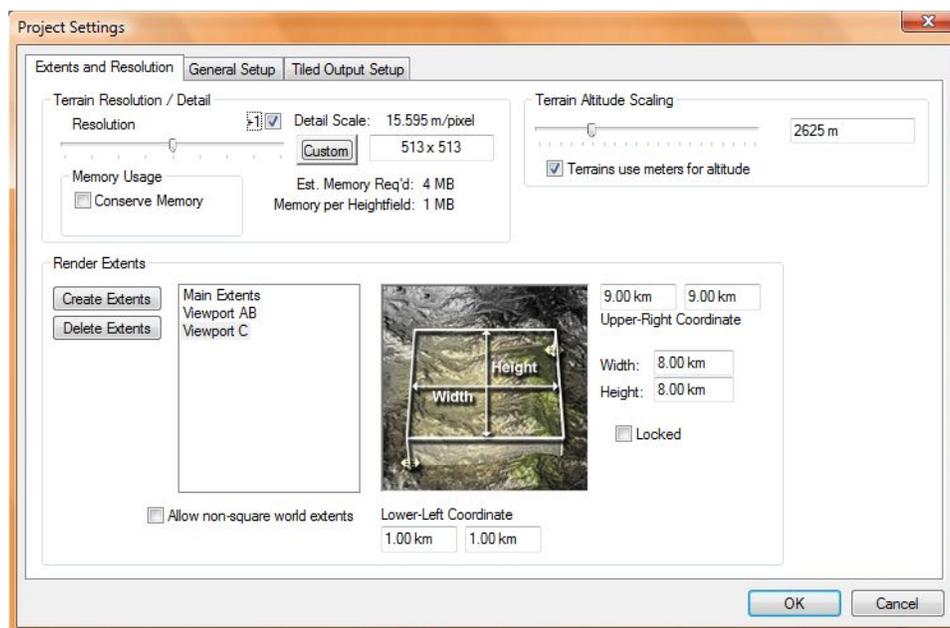
**Definition:**

*Render Extent:* A rectangular area defined in worldspace, used to set the area of terrain being exported or viewed.

World Machine allows you to have multiple Render Extents defined. This ability is useful to be able to see, work on, or export multiple different areas of the world. Render Extents can be created and manipulated in the Project Setup dialog, described below. In addition, Render Extents can be graphically manipulated in the Layout View.

### 3.1 Project Setup

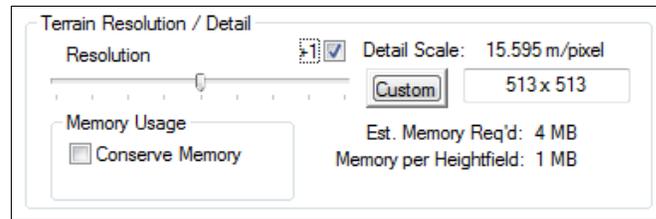
To setup your project options, select **Project Settings** from the *World Commands* menu, or click the appropriate icon in the toolbar.



#### 3.1.1 Resolution

Please note that the Basic Edition of World Machine is limited to only exporting 513x513 or lower

resolution.

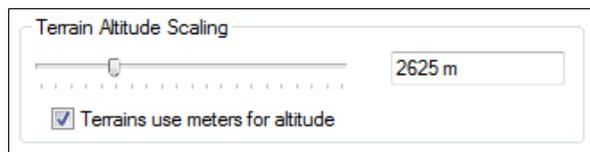


The Resolution slider controls the size in pixels of the world to be built. Higher settings are more detailed but consume more memory and are slower to build. The exact maximum resolution at which you can build a given world depends upon the devices present in the world. The 32bit edition of World Machine is allowed by Windows to only access 2-3GB of memory at most; the Professional Edition is available in a 64bit edition that removes this fundamental limit.

- **+1 Checkbox:** This option allows you to have the slider select resolutions of the form  $2^{n+1}$  rather than  $2^n$ . Some rendering applications and display engines require power of two plus one rather than power of two sizes, and this allows you to easily use this form.
- **Custom Resolution:** You can override the slider and set any size terrain output you wish by clicking the Custom button, and then typing the size into the input field.
- **Memory Conservation:** The "Conserve Memory" checkbox changes World Machine's behavior in terms of keeping final-build heightfield files around. With memory conservation **off**, all of the devices will preserve their full resolution build, allowing you to inspect the final output at all stages of the world. However, this consumes a great amount of memory.

For most higher-resolution builds (2048+) memory conservation must be turned **on**. In this mode, full-resolution output is preserved only at the output nodes in the network!

### 3.1.2 Terrain Altitude Scaling

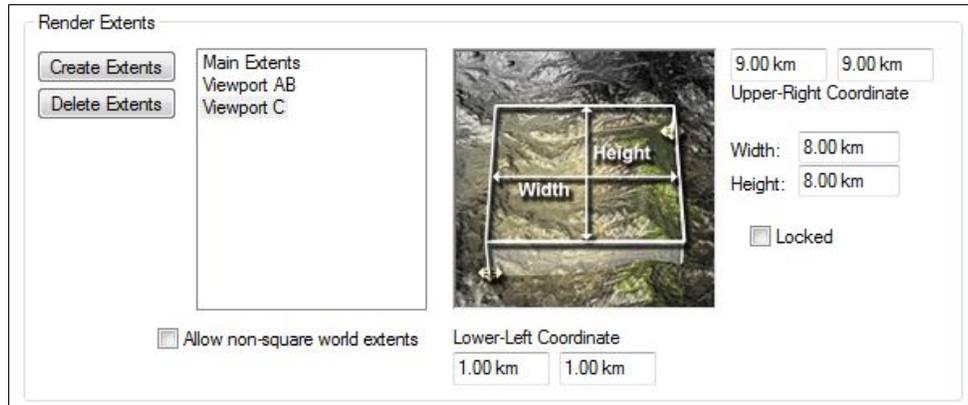


Internally, World Machine keeps all height information stored as a number between 0 and 1. The scaling value allows you to set the elevation in meters that a maximum height value refers to. The greater this number, the taller the maximum mountain height.

Note that this scaling value is applied worldwide; increasing the value will increase the apparent height of every terrain in the world and vice versa. If the terrain you're creating is sensitive to having exact elevations, it's advisable to set this once at the start of your project to encompass the greatest range of elevations you foresee and then not change the value.

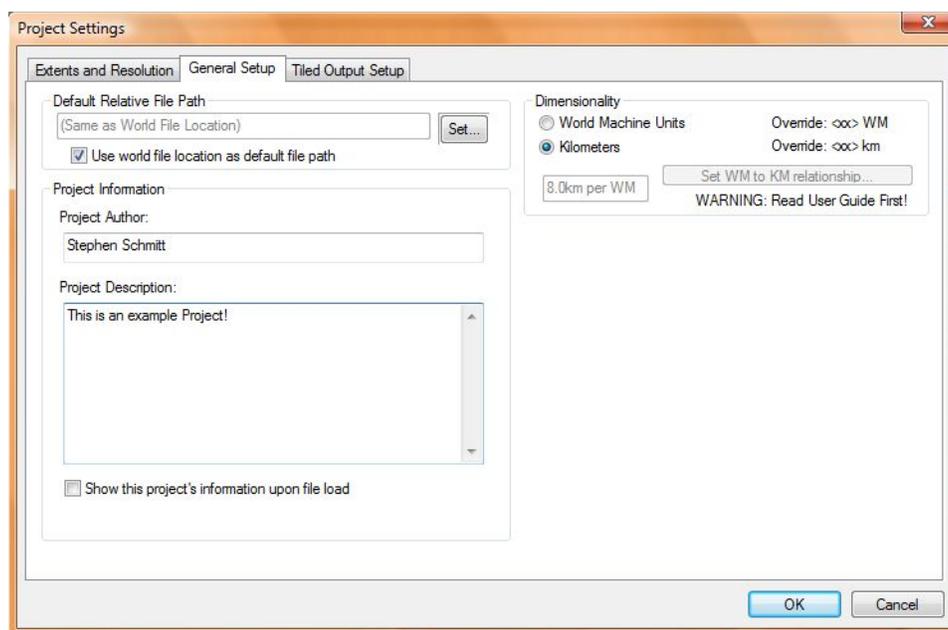
### 3.1.3 Render Extent Management

This very important section allows you to create, delete, select, and modify your render extents.



- **Create Extents:** Create a new Render Extent, prompting you for a name. By default the new render extent will cover the same region as the currently selected extents.
- **Delete Extents:** Permanently delete the currently selected extents! Note that it is impossible to delete the first render extent, as you must always have at least one extent in the world.
- **Extents List:** You may select a render extent by clicking it in the list. By click-hovering over an entry, you can rename it.
- **Coordinate Boxes:** Enter the X and Y coordinates in worldspace of the corners of the render extent. You may specify a coordinate in either kilometers or World Machine Units.
- **Width & Height:** Set the width and height of the render extents, adjusting the Upper-Right coordinates of the extent as needed.
- **Lock:** Locking the extent will disallow further changes until unlocked. Locking an extent is useful to prevent accident changes to an important render extent
- **Non-square extents:** Checking this box will allow the render extent to be a rectangle of any aspect ratio, rather than just a square. The resolution of the rectangle will be the chosen resolution on the longest side.

## 3.2 General Setup



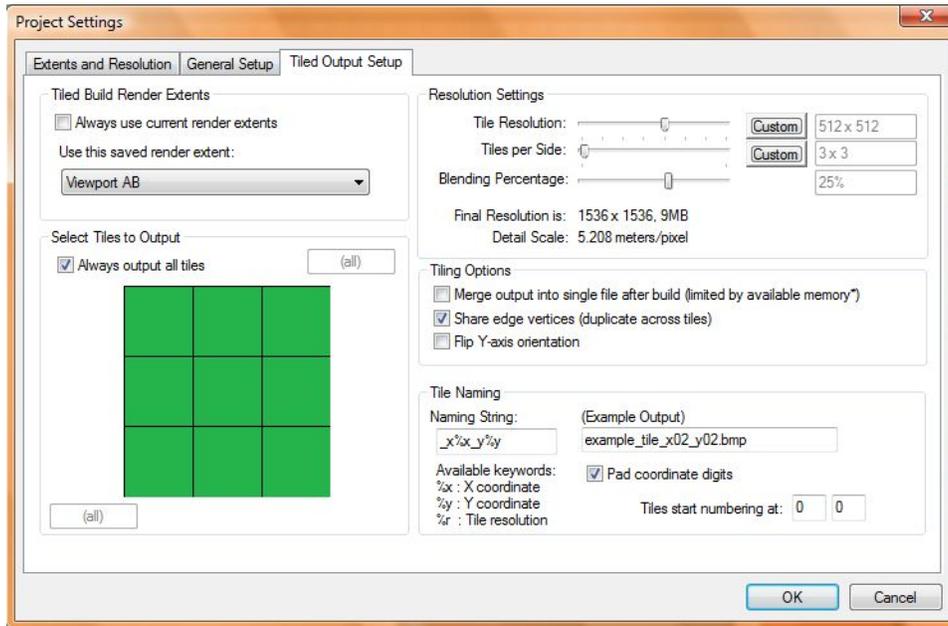
This tab contains some general options that you can set for your project.

- **Relative File Path:** This allows you to set the base folder from which World Machine will try to resolve all relative paths specified to input or output devices. Using relative paths in your project allows you to easily archive multiple copies of a world or transfer it to a computer with a different folder structure.

The checkbox allows you to set the default file path to the location of the Project TMD file.

- **Project Information:** Specify the author of the world file, and add any description to the project that you wish that will help yourself or others work with your project. You can optionally have World Machine show this information every time the project is loaded.
- **Dimensionality:** Controls whether all worldspace coordinates are shown in kilometers or the older World Machine Unit. Unless you have a specific reason, you will usually want to leave this set to Kilometers.

### 3.3 Tiled Export Options (Professional Edition Only)



Please see the Professional Edition Appendix for information on this section!

## 4. Terrain Views

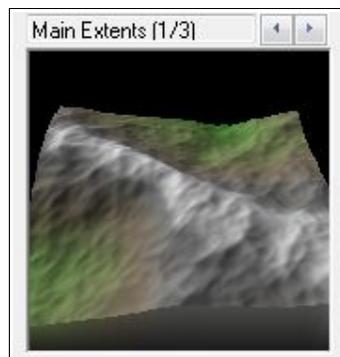
World Machine has a variety of views which provide a view of the current terrain. Some display the current render extents, while others view the entire infinite procedural world in which the render extents are defined.

### 4.1 Leftside View

The “Leftside View”, as it is called, is the toolbar running vertically along the left side of the screen. The leftview can be shown and hidden by pressing the **F4** key. For clarity, we will examine each section of the toolbar by itself.

#### Terrain Preview Window

This window allows you to view a small, preview version of the current terrain area before it is built. This window is updated in real-time as you adjust device settings, allowing you to quickly see the effect your modifications have on the terrain.



Above the preview window, the name of the current render extent is displayed. Clicking the scroll arrows next to the name allows you to cycle through all of the currently defined render extents.

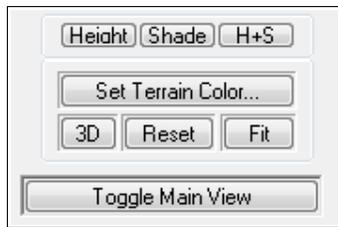
The preview window can be set to either 2D or 3D mode. Depending on the mode, you may be able to perform different actions, shown below. To toggle the preview mode, click the “**2D/3D**” button below the preview window.

The exact operation performed when you click and drag in the preview window depends upon your [view navigation options](#). Click dragging will perform the following actions:

**2D Mode:** You may change the direction of the current terrain lighting in the preview area.

**3D Mode:** You may rotate and zoom the view to find a new viewing angle of the terrain. In addition, you may change the direction of the current lighting of the terrain.

#### View Option Buttons:



The three buttons right below the preview window allow you to change how terrains are shaded in World Machine:

- **Height:** Do not shade maps; Assign color by height only.
- **Shade:** Shade by terrain features (diffuse lighting)
- **H+S:** Assign color by height and also shade the terrain features
  
- **Set Terrain Color:** Choose a color scheme for the terrain. This is a cosmetic choice only and will not affect the terrain in any way.
- **2D/3D:** This button toggles the preview window between 2D and 3D display modes.
- **Reset Preview:** Resets the lighting and camera angle of the preview window to default. Useful if you ever “get lost”.
- **Fit:** Automatically sets the zoom level of the preview to fit the terrain into the view.
  
- **Toggle Main View:** This button will toggle the display in the main window between the Device Workview and the last active Viewing mode. *Right-clicking on any empty region in one of the other views will also accomplish this.*

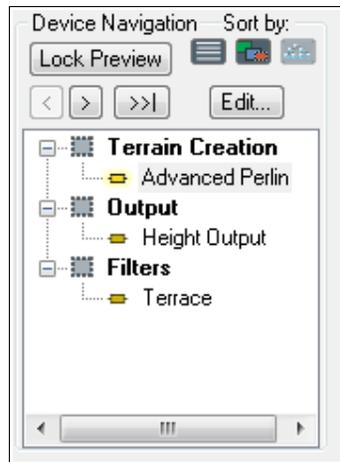
### Water Level

This section controls the optional water plane that WM can display to help you visualize a global water level that your application might use.



You can show or hide the waterlevel display by clicking the appropriate button. Either entering a value directly or moving the slider will allow you to change the waterlevel.

### Device Navigation Area

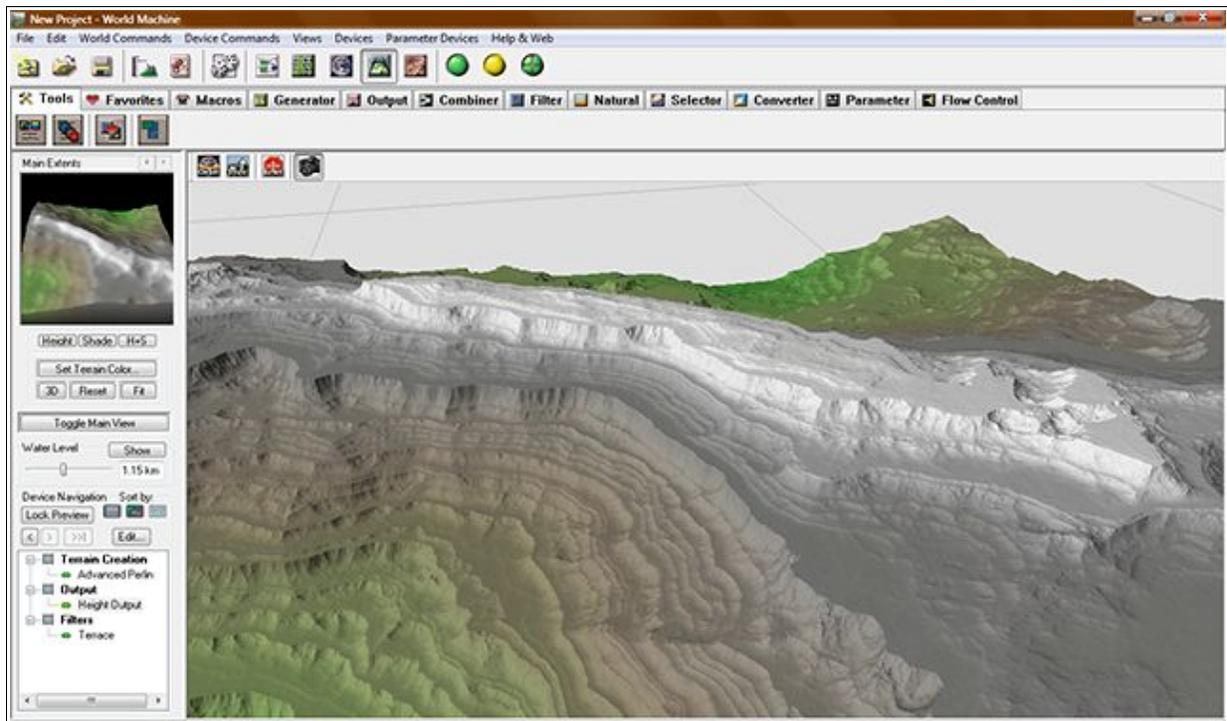


The device navigation list shows every device that exists in the current world. Devices are listed by either the creation order of devices or by groups. You can change the currently selected device by clicking on a new one; double-clicking will change that device's properties.

- **Lock Preview on Device (*f* hotkey):** This very useful function allows you to 'freeze' the preview window onto a device other than the currently selected one. For example, you may wish to freeze the preview onto a terrain filter device, and then go back up the chain and modify the settings of the generator device and see the *filtered* changes in real-time. Toggling the lock button again will disable the preview freeze.
- **Move Backward:** Changes the selected device to the prior device in the chain from the currently selected device.
- **Move Forward:** Selects the next device in the chain from the current one.
- **Move to End:** Selects the LAST device in the chain from the current one.
- **Sort-By:** Allows you to sort the device listing by either creation order, or grouping.
- **Edit...:** Brings up the properties dialog for the currently selected device.

## 4.2 3D View

The 3D View can be selected from the [Command Bar](#), or by pressing **F8**.



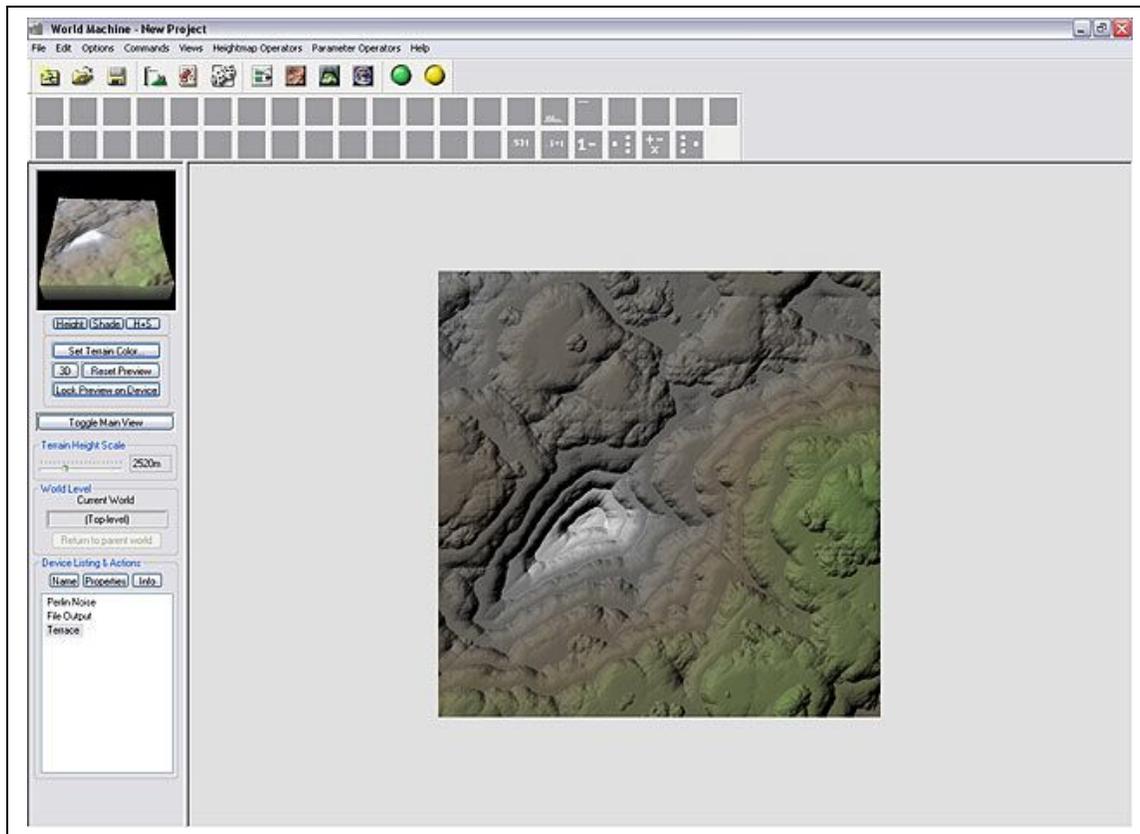
The 3D View will always show the current contents of the active terrain. If the currently selected device has not been built yet (displays a yellow rather than green light in the Workview), the 3D View will show the lower-resolution preview results rather than the full-detail build results.

The 3D View also has its own toolbar:



- **Orbit:** This button puts the 3D View into Orbit mode. In this mode, moving the mouse will orbit the view around the terrain. This mode makes it easy to inspect the terrain from a variety of angles.
- **Free Look:** The Free Look mode is similar to the traditional camera controls in WM 0.99. You can rotate the camera in any direction you want by left-clicking and dragging with the mouse; right-clicking and dragging up and down will move the camera vertically; and the arrow keys on the keyboard move the view in the direction you are facing. This allows you to freely navigate and inspect the terrain from wherever you wish.
- **Reset View:** This resets the current mode to its default rotation – useful if you’ve navigated somewhere strange and can’t find the terrain anymore!
- **Snapshot Mode:** The Snapshot mode performs a "render" using OpenGL at full terrain resolution. Typically terrains higher than 1024x1024 are rescaled down for display due to the millions of triangles produced. Snapshot mode lets you view the terrain in its fully detailed glory. Note that as soon as you move or adjust the view, it will revert back to the realtime display.
- **Detail Toggle:** Turns detail texturing on or off. Using a detail texture can help give the terrain a better sense of scale, but you may wish to disable it in some instances to see the "true" terrain.

## 4.3 2D View



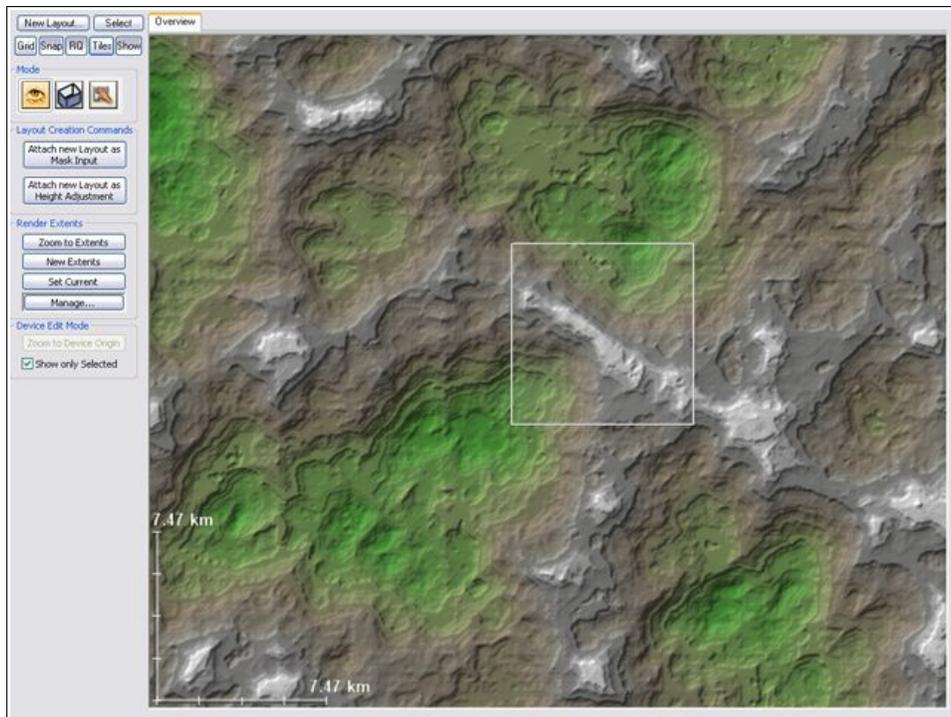
The 2D View can be selected from the [Command Bar](#), or by pressing **F9**.

The 2D View offers you an overhead view of the terrain with optional shading applied to show the relief of the terrain. The 2D View does not use OpenGL or other 3D systems and so is a good viewing choice for very large maps, systems with slow 3D cards, or other conditions that favor 2D use.

The 2D View offers very little interactivity. The display will update to reflect new sunlight conditions when you use the [Leftside view](#) to adjust the lighting. When the map is large enough to extend past the view boundaries, you can use the scrollbars to view different areas of the terrain.

## 4.4 Layout View

Layout View can be selected from the Command Bar, or by pressing **F6**.



Layout View is extremely useful for planning out your world. Much like Explorer View, it displays terrain outside of the currently defined render extents. Layout View allows you to rapidly pan and zoom across the entire world, zooming in to view details at a range of a few meters, then pulling out to view many thousands of kilometers of terrain.

#### **A Note about Terrain Artifacts:**

Layout and Explorer View can display any part of the entire virtual world, not just the render extents you designate. However, this ability means that some devices, in particular simulation-based devices such as Erosion, may not perform identically in these views as they do within the render extents. World Machine marks these devices with an [optional](#) warning symbol in the Device Workview. The warning symbol is a red exclamation point: "!" The severity of the problem depends upon the device and zoom level.

In addition, Layout View provides a visual way to set and modify render extents, and manipulate certain devices that expose parameters to visual adjustment.

The most important use of Layout View, however, is in conjunction with a Layout Generator to edit shapes in the Shape View. **Chapter 5** of this manual is devoted to using this incredibly powerful feature. We'll only slightly address this functionality here.



### Layout Generator Commands

- **New Layout:** Creates a new Layout Generator and selects it into the view.
- **Select:** Selects as the current device, the Layout Tab being displayed in the view.
- **Attach Layout as Mask:** Creates a new Layout Generator and attaches it to the mask input of the currently selected device. This action is not available if the selected device has no mask input.
- **Attach Layout as Height Adjustment:** Attaches a Layout Generator to the output of the currently selected device. This action is not available if you are viewing a device with no output.

### Common State Toggles

Clicking any of the buttons in the common function area will toggle their status on/off. The button state is remembered separately for the Overview tab and the Layout tabs.



- **Grid:** Display of a grid atop the terrain
- **Snap:** Snap the mouse to the grid
- **RQ:** Displaying the render extents
- **Tiles:** Show the render extent being used for tiled output, and show which areas of the terrain will be in which tile
- **Show:** Show or hide the display of terrain in the Layout View

### Mode Buttons



The three mode buttons control the operating mode while in the Overview.

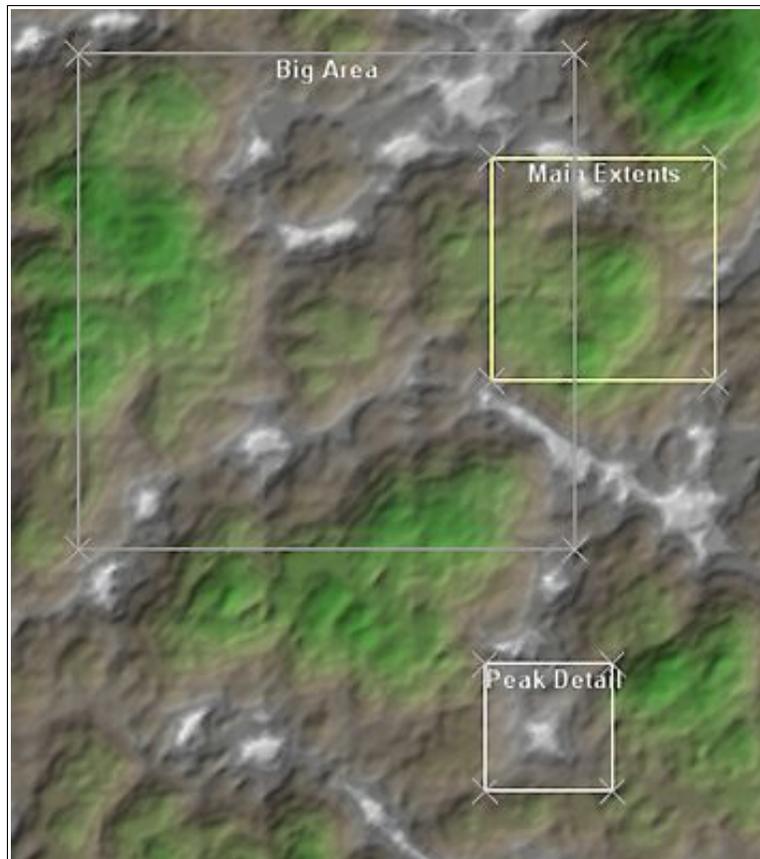
### 4.4.1 Overview Mode

Overview Mode is the default mode. You can perform no actions, merely pan and zoom around the terrain.

### 4.4.2 Render Extent Mode

Render Extents Mode allows you to graphically add and manipulate your render extents.

#### Render Extent Manipulation



While in Render Extents mode, all of the currently defined extents are visible at their appropriate locations in worldspace. You can click and drag on either the corners of the extent quadrant, or on the extent itself to change its size or location.

#### Render Extent Commands

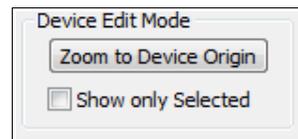
The following commands are available in Render Extent Mode:



- **Zoom to Extents:** Brings into view the currently selected Render Extents
- **New Extents:** Adds a new Render Extent to the world, that defaults to the same location as the currently selected one.
- **Set Current:** Set the currently selected Render Extent by dragging a rectangle in the view. Once you finish the drag, the rectangle you defined is the new borders of the current Render Extent.
- **Manage...:** Bring up the Project Setup dialog to precisely define the Render Extents.

### 4.4.3 Device Mode

Device Mode lets you manipulate certain device parameters visually that Devices in your world may expose.

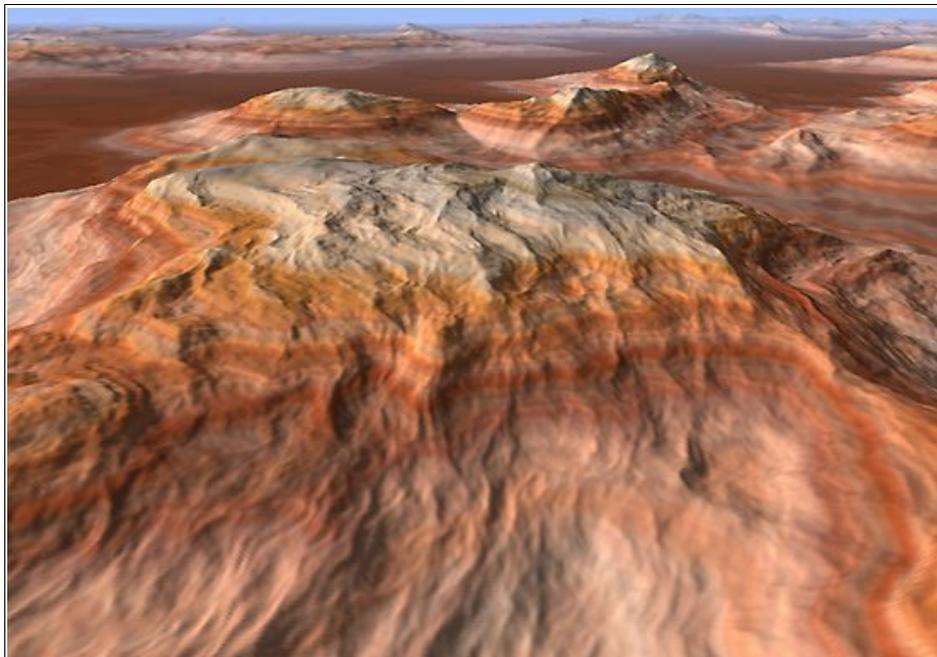


There are two options on the Layout Overview Bar that are applicable to Device Mode:

- **Zoom to Device Origin:** If the currently selected device supports a device origin, this button will zoom to its location. Only certain devices, principally Generators, have an origin.
- **Show only Selected:** When checked, you will only see controls (if any) from the currently selected device.

## 4.5 Explorer View

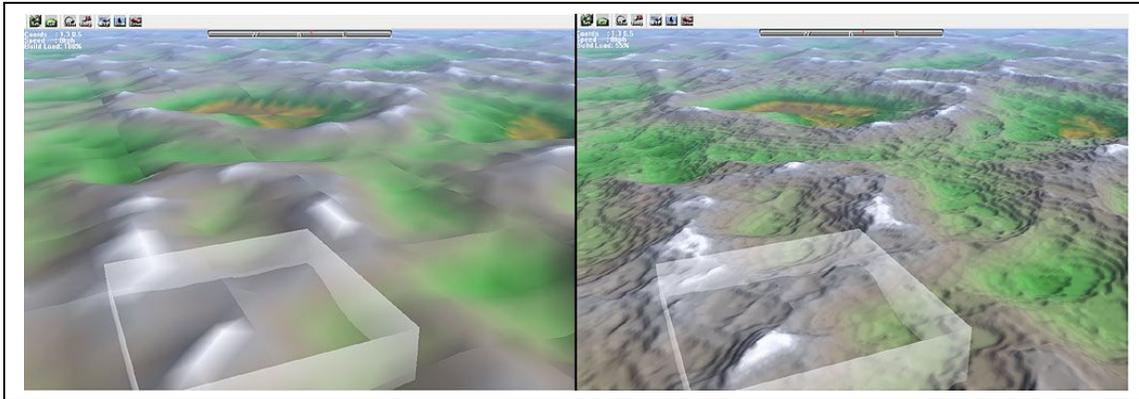
Explorer View can be selected from the [Command Bar](#), or by pressing **F7**.



Explorer View allows you to easily explore the infinite world defined by the devices you placed into your World Machine world. Unlike the 2D and 3D views, you can see and travel over a representation of what the world looks like everywhere, not just within the render extent square. This is *extremely* useful for locating interesting areas of terrain in the current world defined by your device networks.

#### 4.5.2: Explorer Details

What exactly Explorer mode looks like depends greatly upon the power of your computer system and the settings you selected in the [Graphics Options](#) dialog.



Explorer mode uses a *refining progressive level of detail* system. In non-technical terms, this means that the world starts out in very low detail, as seen on the left, and then is continuously refined so that nearby areas are shown with greater resolution, as on the right. In addition, beware that an old or underpowered system might be only able to run Explorer mode at a detail level equal to the left view above. A high-end computer with a powerful 3D card can look like the right side or better.

You can see that in the upper middle of the screen, there is a compass; the red marker indicates the current view direction. North is in the middle, indicated by N; the extreme sides of the compass both represent south.

On the left, there are three lines of text. They indicate the **current coordinates** of the view along the x and y axis, the **speed** with which you are moving, and the **build load**. Although the first two are self-explanatory, the build load is less obvious.

**Build load:** The percentage of time during the last second that Explorer mode has been actively building tiles. Thus, if this value is greater than 0%, tiles are currently being created that will increase the level of detail of the current scene.

#### 4.5.3: Exploring Explorer!

To use Explorer mode, you can simply jump in and go! Explorer starts in "God's Eye" navigation mode. You can use the arrow keys and/or the drag the mouse on the landscape to move.

If you ever get lost, simply push the "View" button on the Explorer Toolbar. This will reset your current position so that you are looking at the current terrain render quad.

#### 4.5.4: Explorer Toolbar



The Explorer Toolbar is divided into two segments that deal with different aspects of Explorer. They are, from left to right across the toolbar:

*View Locations:*

- **View:** Move the camera to bring the current render quad into view.
- **Jump:** Jump to a particular location by specifying X/Y coordinates.

*Navigation Mode:*

Each navigation mode uses the keys in a slightly different way:

- **God's Eye:** Pan around the terrain by dragging it.
  - Dragging the terrain with the **left mouse button** moves the camera
  - Dragging up/down with the **right button** changes your elevation
  - **Left + Right mouse buttons** together change the current viewing angle
- **Fly:** Fly above the terrain.
  - **Up/down arrow** controls speed
  - **Left/right arrow AND/OR mouse** controls heading and elevation
  - **Right mouse button + mouse up/down** adjusts elevation
- **Walk:** Walk over the terrain.
  - Up/down arrow steps forward/back
  - Left/right arrow sidesteps
  - Left mouse button + mouse left/right adjusts heading
  - **Hold down the right mouse button** to step forward
- **Drive:** Drive around the terrain in a buggy.
  - Up/down arrow accelerates/brakes
  - Left/right arrow turns
  - Holding the ctrl key puts the buggy into “hover” mode
  - Holding the shift key acts as a “turbo boost”

#### 4.5.5: Explorer Caveats

Explorer Mode is a very powerful and useful feature. However, because of its demanding nature, it requires a lot from your computer system. Specifically: a fast CPU to generate terrain tiles, and a fast video card to display them. You will likely need to play with the quality presets in the [Layout/Explorer Performance Options](#) dialog to find a performance level acceptable for your system.

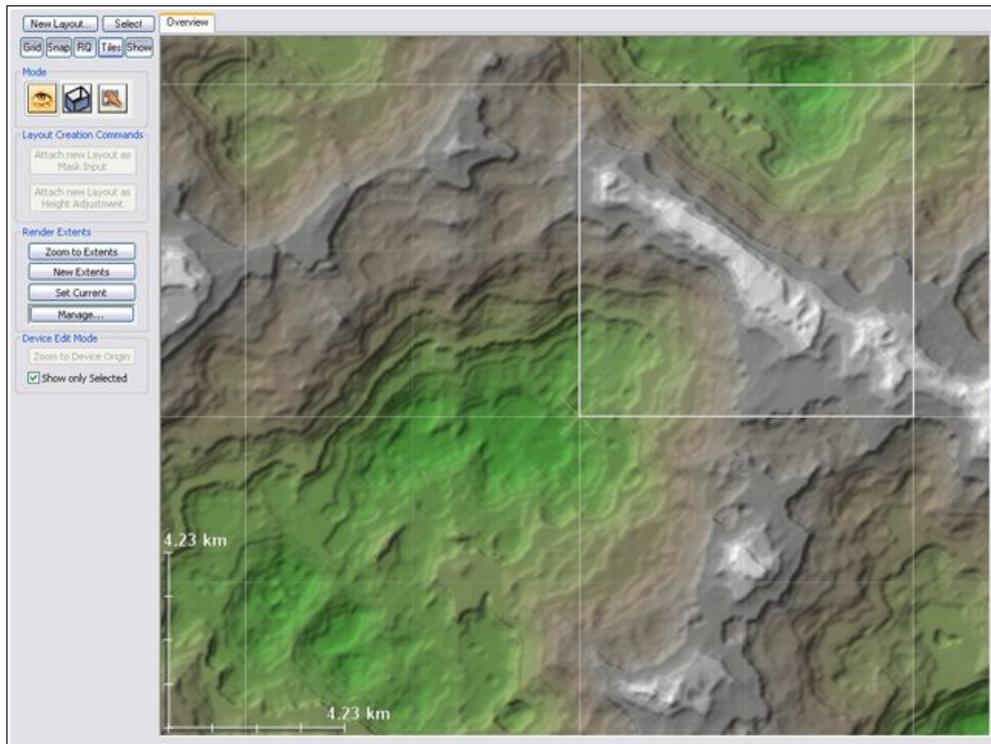
In addition, much like Layout View, some devices will cause visual artifacting within Explorer Mode! See the warning in section 4.4 for more information.

## 5. Layout Generator

The Layout Generator is a graphical design tool that allows you to bridge the gap between fully procedural and fully artist-created terrains. With a Layout Generator, you have a suite of vector drawing tools that let you craft terrain areas, insert roads and rivers, draw regions of influence as effect masks, and much more.

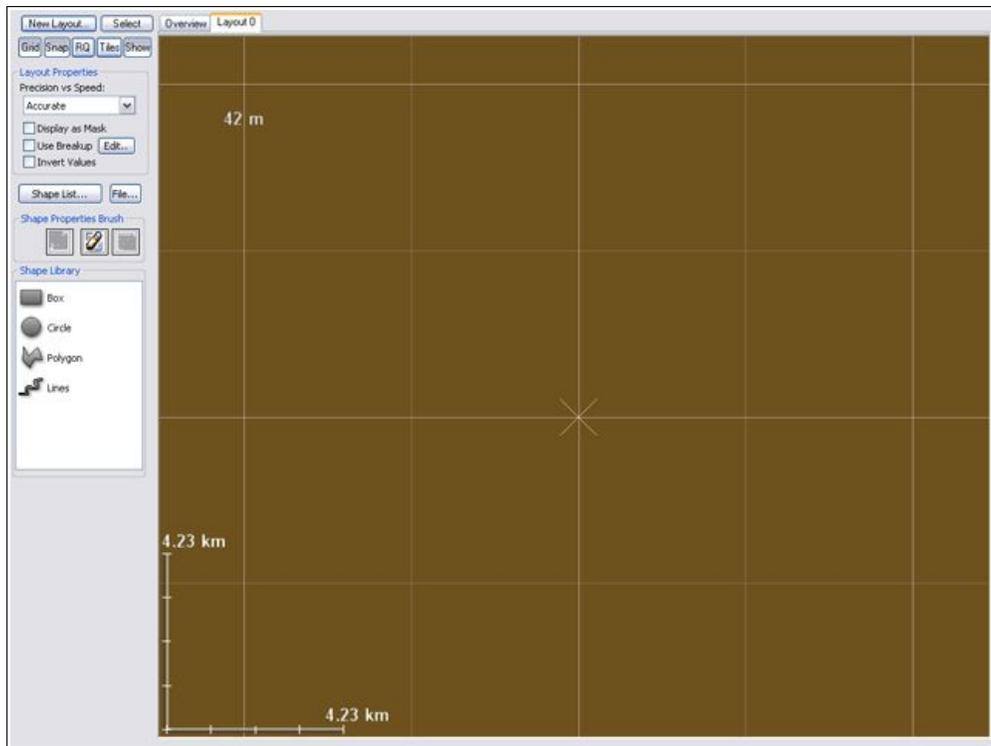
### 5.1 Layout Generator Basics

To create a Layout Generator, go to the Layout View:



Click the **New Layout** button in the upper-left corner of the above photo. As you create layout generators, they will appear as tabs across the top of the view.

When a layout is selected in the view, the screen will change to the Layout Editing mode, as you can see below:



A Layout Generator acts as a container for shapes. You can have many different layout generators. each one you create will show up as an associated device in the Device Workview. The output of the layout generator can then be used just like any other component in the workflow.

Also, note that you can change the currently viewed output while editing a layout generator and its shapes. This ability lets you view the final output from farther down the device flow while working on a layout generator.



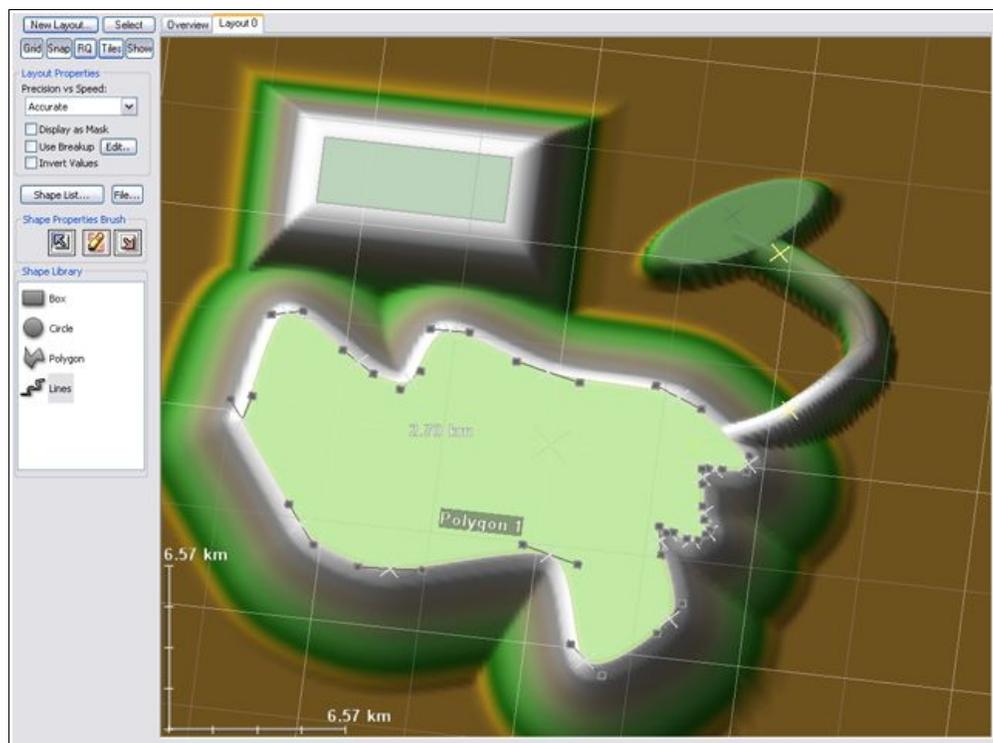
## Layout Generator Options

- **Rendering Mode:** This allows you to choose whether to create full detail or reduced-detail shapes. In most cases you will want to leave this set to "Accurate", unless you are working with a layout containing a great number of shapes that is building extremely slowly.
- **Display as Mask:** Causes the output of the layout generator to be flagged as a mask rather than a terrain, and display as such. Useful when creating a layout that is intended to be used as a mask.
- **Use Breakup:** Enables fractal breakup on the shapes in the layout. This allows you to add more natural-looking profiles to your basic geometric shapes. See section 5.4 for details.
- **Invert Values:** inverts all height values of the layout. Useful when creating masks, as well as things like canyons and other terrain where you'll be removing rather than adding things.

- **Shape List Button:** brings up a dialog that lists all shapes contained within the current layout generator. Clicking on a shape in the list will select it, allowing you to locate and manage many shapes.
- **File Button:** Imports or exports shapes. See the Shape I/O section below for more information.

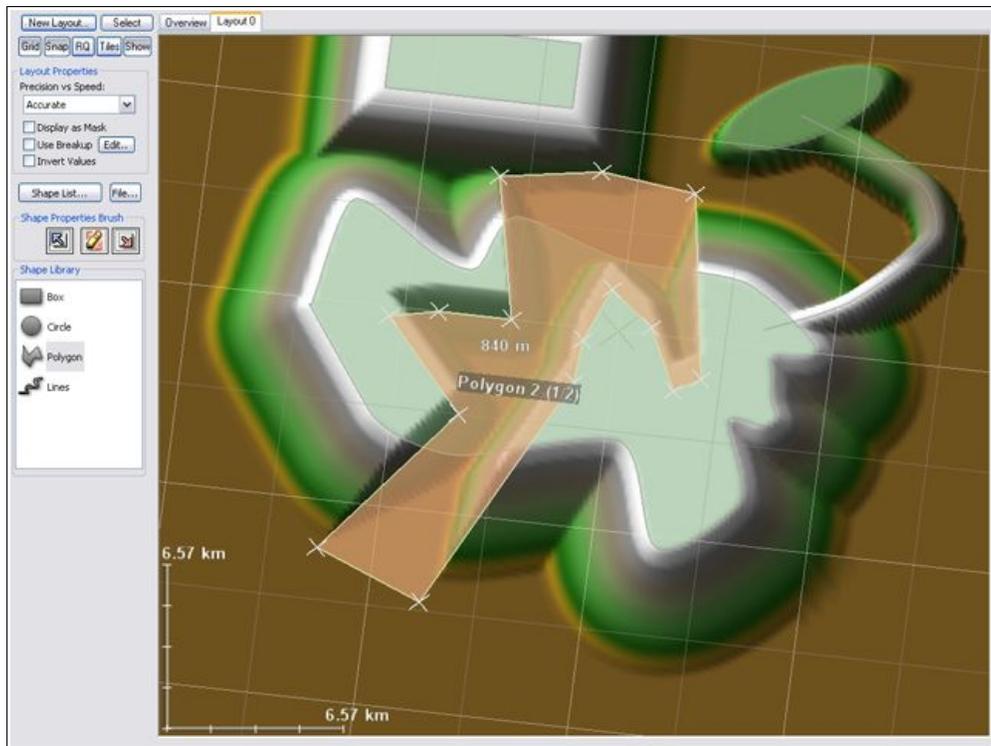
## 5.2 Creating and using Layout Shapes

On the left is the Shape Library, where you can access the various shapes available for you to draw. To create a new shape, simply select the variety you desire from the library, and click-drag in the terrain area. All shapes are created in a similar way, although some allow you more flexibility during shape creation. For example, polygon and path shapes can be either sketched or plotted more precisely by clicking individual vertices rather than dragging.



Note that around each created shape there is a "transition zone", where the shape's influence fades away to the background. When multiple shapes overlap, the one with the greatest height governs.

You can also choose to set a shape to Subtractive mode, in which case it will be subtracted from all of the positive shapes at that same point.



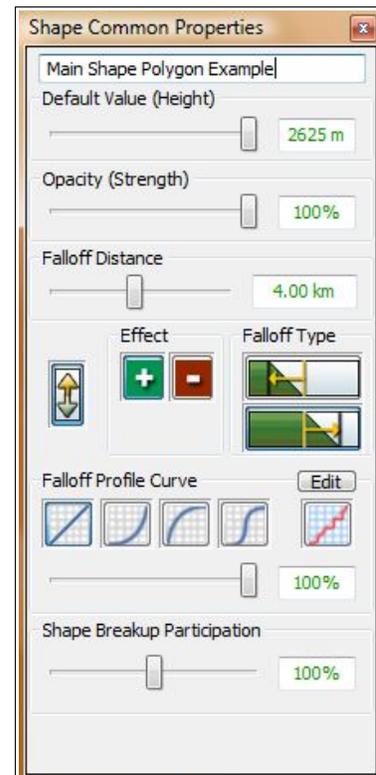
Shapes can be selected by left-clicking on them. Multiple shapes can be either drag-selected using the selection marquee, or added to the selection individually by holding down shift while clicking on new shapes.

## Shape Properties

Once a shape has been created, you can bring up a shape's properties by double-clicking on it:

All shapes have several properties in common that can be edited.

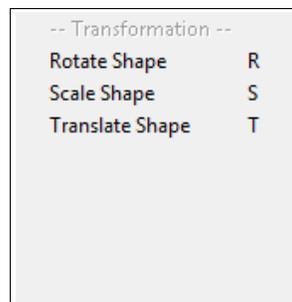
- **Default Value:** Controls the height value of the shape
- **Opacity:** The power of this shape versus the background. Low opacity values allow background heights to show through.
- **Falloff distance:** the distance between the border of the shape and the region in which the shape has no influence on the terrain.
- **Falloff type:** Many shapes allow you to how you wish the falloff zone to be defined:
  - **Exterior falloff** means that the shape is full strength to the border, and then falls away on the outside of the shape.
  - **Interior falloff** means the shape's falloff occurs inside of the shape, reaching no influence at the border of the shape.



- **Operation:** Shapes may be positive or negative. A negative shape is subtracted away from other existing shapes.
- **Shape Falloff function:** you can select from a variety of preset falloff functions.
  - **Linear falloff** provides an even gradient to the falloff.
  - **Squared falloff** provides gentle lower regions and a sharp ridge on top.
  - **Square root falloff** provides a gentle top transition and steep lower areas
  - **S-falloff** provides smooth falloff on both the top and bottom of the shape.
  - **Custom falloff** allows you to define a custom falloff function, which can be very useful for among other things, defining road and river profiles.
- **Falloff Fader:** Lets you control how much influence the falloff function chosen above has on the falloff curve. A value of 0% will always give you a linear falloff gradient, while 100% means that you are using the chosen function.
- **Shape Breakup Participation:** Controls the extent to which this shape is effected by the Layout Generator's fractal breakup setting.

### 5.2.1 Transforming Shapes

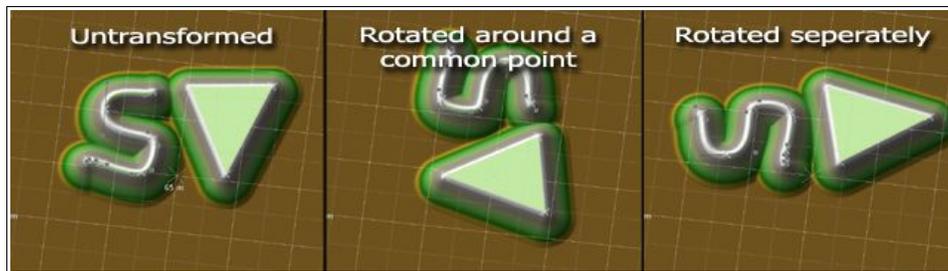
Dragging a selected shape will move the shape in the direction you drag. Right-clicking brings up the a context menu for the selected shapes:



All shapes support at least some transformation options. These allow you to translate, rotate, or scale the selected shapes. Simply select the transformation option you want (or use the Hotkey for it from the the Layout View) and drag in the viewport to perform the action. Clicking without dragging will end the transformation mode.

#### Transformation Center

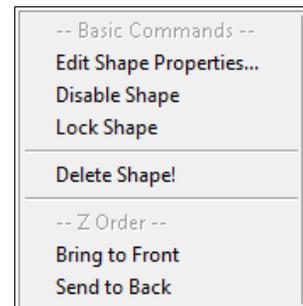
Holding down shift or ctrl while rotating or scaling will perform each action around the shape's own origin rather than the center-point of the group. The graphic below illustrates the difference:



Shapes can also be copied or pasted by selecting the appropriate option from the edit menu, or using the standard CTRL-X, CTRL-C, CTRL-V shortcut keys.

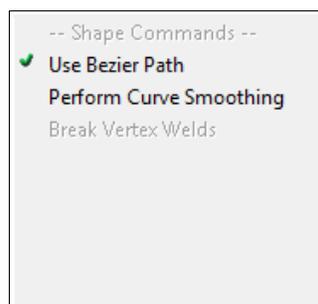
### 5.2.2 Other shape options

- **Enable/Disable Shape:** if a shape is disabled, it no longer effects the world in any way.
- **Lock Shape:** make the shape unchangable until this is selected again. Use this to prevent accident modification of important shapes.
- **Delete Shape:** As you might expect, this deletes the currently selected shape(s)!
- **Bring to Front/Send to Back:** Moves the selected shapes forward or backward in the Z order. This has no effect on output, but allows you to change which shape is selected first between overlapping shapes.



### 5.3 Vertex-based shape options

The polygon and path tools are defined by a sequence of vertices. There are some additional editing options available for vertex based shapes.

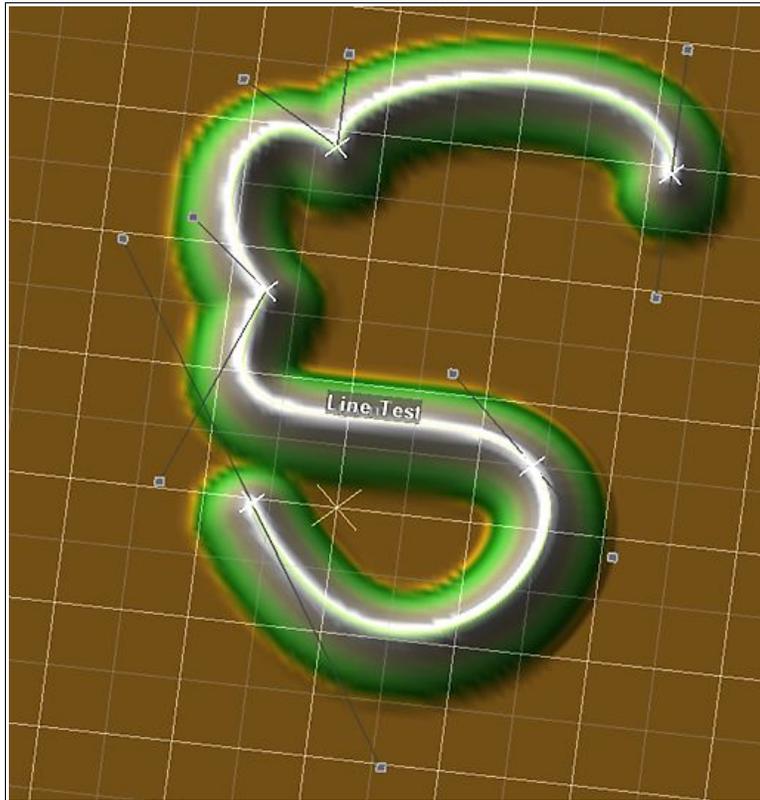


#### 5.3.1 Bezier Shapes

Vertex-based shapes support bezier splines for smoothly interpolating between each vertex. This is slower to process but is often highly desirable to create smooth and complex geometry.

You can convert between linear and bezier curve segments by checking or unchecking the "Use Bezier Path" option in the right-click context menu for that shape.

If a shape has bezier curves enabled, the shape will have additional handles available representing the bezier control points for the curve.

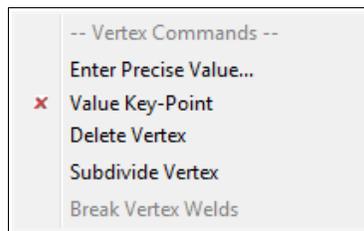


By default, the handles on either side of a vertex are linked together to provide a smooth transition through that vertex, but you can override this behavior by holding down **shift** while dragging a bezier handle.

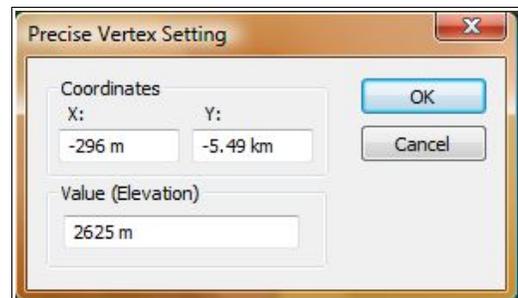
In addition, you can set all of the bezier handles in a shape at once to provide a smooth shape by selecting the "Perform Curve Smoothing" option from the context menu.

### 5.3.2 Vertex Editing

Vertices have their own context menu that you access by right-clicking on a specific vertex. This allows you to set some of their properties directly.



- **Enter Precise Value:** This option allows you to enter specific numbers for the (X,Y) coordinate of the vertex and its elevation value.



- **Value Key-Point:** Toggle whether the selected vertex is interpolated or whether it is a "keyframe" that has its value set manually.
- **Delete Vertex:** Removes the vertex from the shape, connecting the two vertices on either side together.
- **Subdivide Vertex:** Inserts two new vertices adjacent to the selected vertex.
- **Break Vertex Welds:** Break any welds that involve the selected vertex.

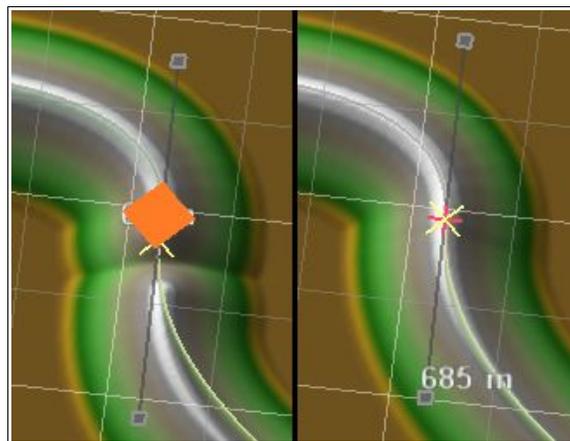
In addition, there is a quick shortcut for setting shape heights on a per-vertex basis. Right-click on a vertex and without releasing the mouse button, drag up or down to quickly set the height value of that point.



### 5.3.3 Vertex Welding

You can weld together vertices that occur on different shapes. Welded vertices will always occupy the same position in space, allowing you to link several shapes together in, for example, a road or river.

To weld together vertices, simply drag one vertex over another. The cursor will highlight to show the potential welding action.

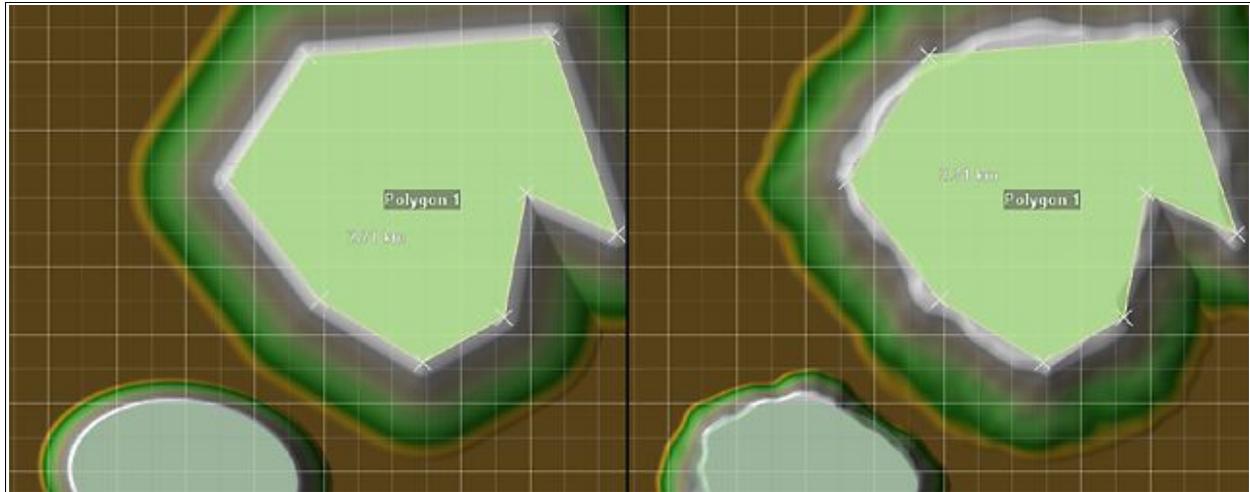


After welding, any transformations performed on one shape that effect that vertex will also move the linked vertex in the other shape(s).

If you decide later that you wish to break the weld, simply select the vertex or shape in question and right-click to bring up the context menu, and select "Break Weld"

## 5.4 Fractal Breakup

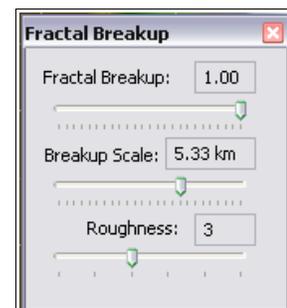
Fractal Breakup allows you to distort the unnaturally straight edges often produced by the geometric primitives. This is often desirable when creating natural shapes.



Fractal Breakup must be enabled on a per-layout basis when you want to use it. However, you can decide on a per-shape basis which shapes are distorted by changing that Shape's "Fractal Breakup Participation" value.

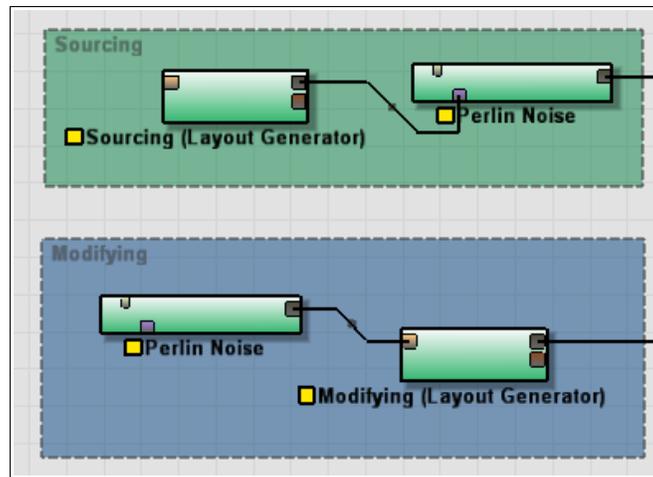
You can change the scale and intensity of the breakup effect by editing the Fractal Breakup properties:

- Fractal Breakup: The intensity of the effect. Lower values provide a more subtle distortion.
- Breakup Scale: The distance that the distortion effect is applied for. Large values will produce slowly varying but large amplitude variations, while low values produce more of a ripple effect.
- Roughness: The complexity of the edge. This number corresponds to the number of octaves of fractal noise to be applied.



## 5.5 Sourcing vs. Modifying terrain with a Layout Generator

You can use the Layout Generator in your world in two different ways.



1. Use the Layout Generator as a Generator, creating terrain or mask information to be wired to a different device in the world.
2. Use the Layout Generator to modify an existing terrain by connecting that terrain into the Primary input port on the LG.

### Deciding on the type of connection to use

You should use the Layout Generator as a source of terrain when you want to generate guide shapes for a fractal generator or a mask for a particular effect. The image below shows the result of connecting the output of a Layout Generator to a Perlin Noise device:



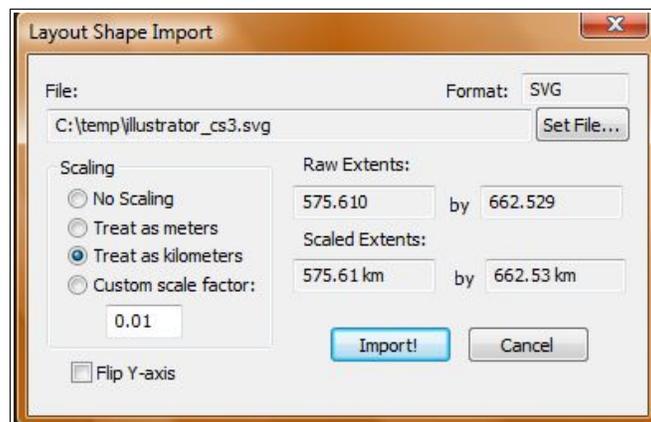
On the other hand, you should use the Layout Generator to modify the terrain when you want to **embed** shapes into the terrain.



For example, leveling areas to a certain height, creating a road that cuts/fills through an existing terrain, or inserting waterways.

## 5.6 Importing and Exporting Shapes

You can import or export shapes with one of several vector-graphics art formats. In particular, the SVG and AI formats are supported.



### Import Scaling

How to scale the raw coordinates is the most important decision when moving shapes to and from World Machine. You can have WM treat the file coordinates as meters, kilometers, a custom scale factor. A display shows you the current region to be exported to a file.

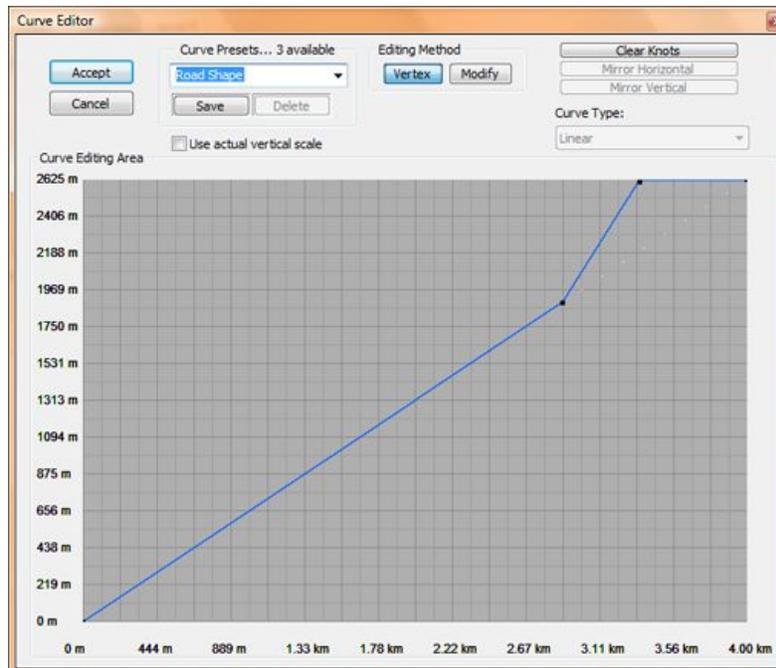
You can also choose to flip the Y-axis of all of the imported shapes; this is useful when interacting with a program that measures from an origin in the top-left corner instead of the bottom-left system that World Machine uses.

Note that there are some serious limitations in what information can be imported or exported; any per-

vertex data you have set will be lost, and World-Machine specific shape information also will not transfer. Nonetheless it can be very useful to be able to re-use vector data from other sources or export the exact paths of splines to the next stage in your production pipeline.

## 5.7 Curve Editor

Layout View contains a curve editor to allow you to define custom falloff profiles for shapes as well as view the elevation profile of a vertex based shape.



The horizontal axis shows the distance along the curve, while the vertical axis shows elevation. The curve editor works in one of several modes depending on what you're editing; when modifying a falloff curve, you can freely add new knots to the curve. When editing a lofting curve (heights along a path), the knots are fixed in position and can only be changed in height. These two modes of operating are reflected in the **Editing Method**:

- Vertex mode allows you to add or delete vertices, and move them horizontally
- Modify mode allows you to change the height of existing points only

Lastly, you can load and save preset falloff curves so that you can re-use particular profiles among layouts or projects.

## 6. File Input and Output

World Machine worlds only describe a set of operations to create the terrain rather than the terrain itself. The action of importing or exporting a terrain is described by adding device nodes to the

world.

## 6.1 File Output

Saving a file to disk is a simple but important part of building your world. Exporting a file is done by using an Output Node to save terrain (or other data) to disk. There are several types of output possible, including terrain heightfields, meshes, and bitmaps. You can output many files from an individual world. For example, from a single project file you could save a terrain heightfield, a texture map to apply to the heightfield, and several additional masks for further use (perhaps in placing vegetation or other purposes).

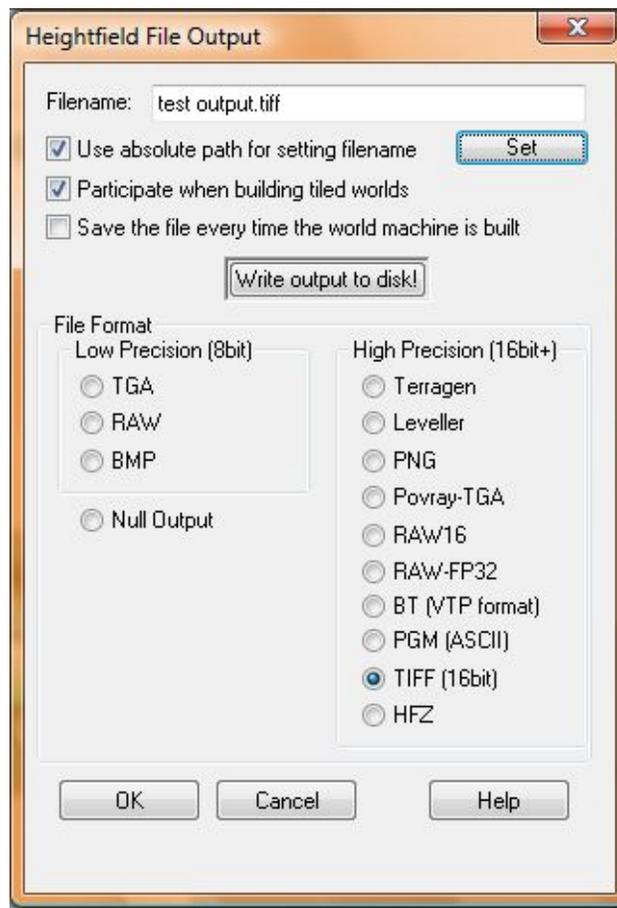
To export data from World Machine, you must place an appropriate Output node from the toolbar into the world and wire your final terrain data into that node.



Note: the area exported will always be the currently selected Render Extents!

### 6.1.1 Heightfields

To export a terrain heightfield, use the default File Output node that the starting world includes, or place a new File Output node into the world from the toolbar.



You can export in many different formats, as seen above. Your choice of what format to use will be governed by the needs of your rendering software, terrain engine, or other further terrain application.

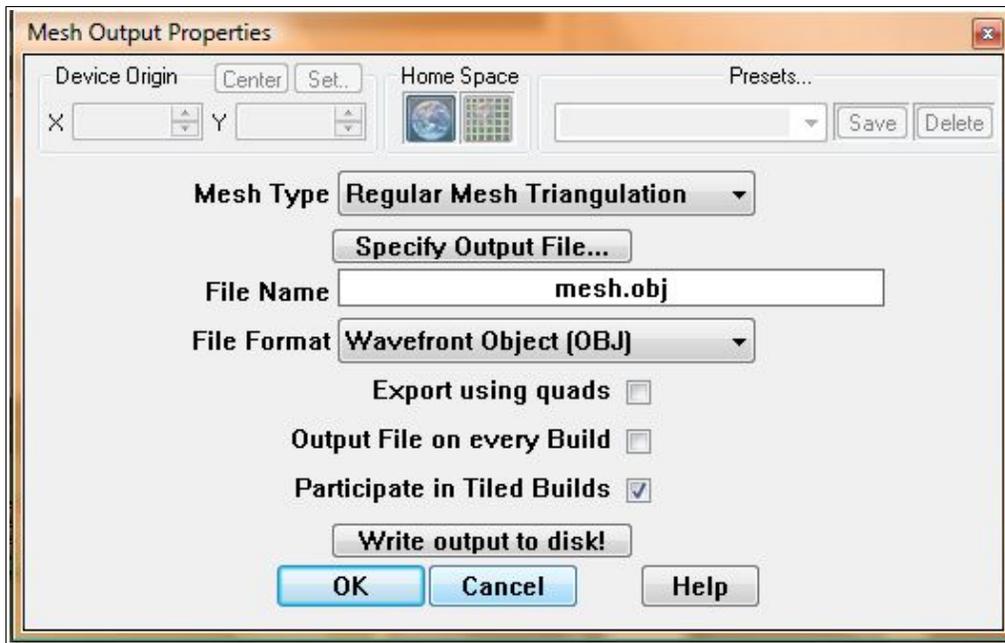
The basic sequence of steps are:

1. Select the file format you wish to export
2. Set the filename by pushing the "Set" button and providing a filename
3. Push the "Write output to disk!" button to export the file.

For more details on the various other options available in the File Output dialog, see the Device Reference.

### 6.1.2 Terrain Meshes

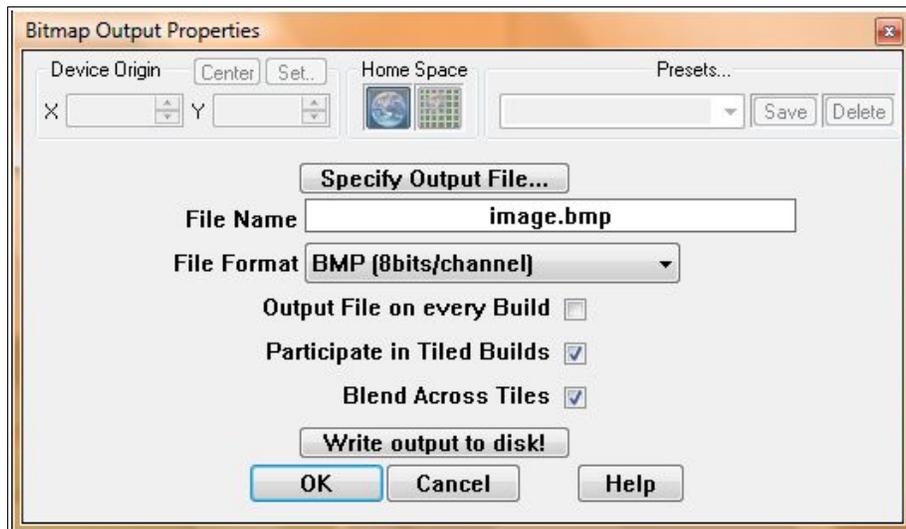
The Mesh Output creates a triangulated mesh from the input heightfield. This is intended for general purpose rendering software or other applications that cannot use a Heightfield; The heightfield should be the preferred output method as it is a much more compact and efficient way of storing terrain data.



The same three steps described in 6.1.1 are used to export a mesh, and indeed any of the file output devices.

### 6.1.3 Bitmaps

To export a bitmap, use the Bitmap Output device. This device accepts a bitmap datatype and an optional heightfield you can use to specify a 4th channel for alpha info.



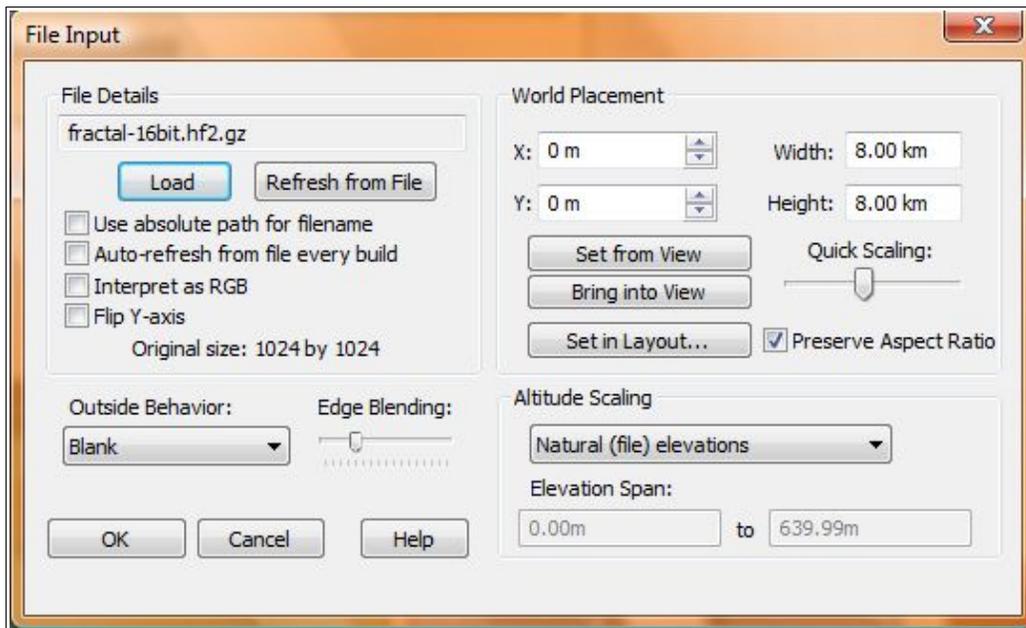
Export Formats:

- **8bit Windows BMP**
- **16bit TIFF, PNG**

The same three steps apply: Set a file format, set a filename, and click "Write output to disk".

## 6.2 File Input

To import an existing file, use the File Input device. It can import a variety of file formats and brings them into the device world as either terrain heightfields or color bitmaps.



For more information on the various options available in the File Input dialog, see the Device Reference. The most important ones will be described below.

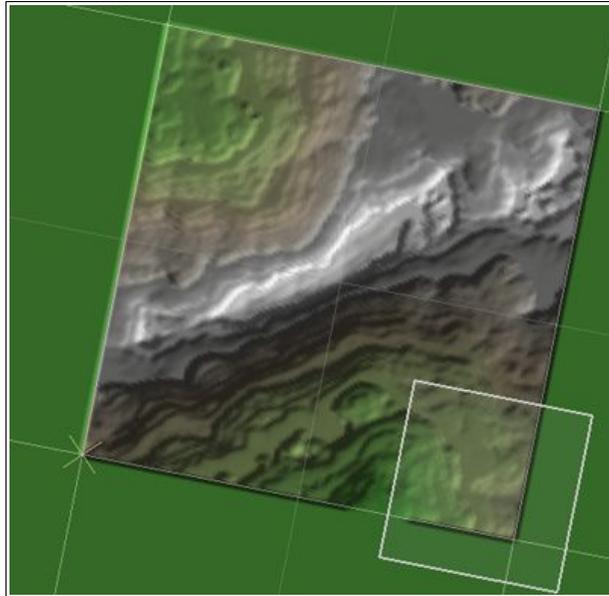
### Loading your file from disk

Click on the "Load" button to browse to and select your input file. Once you've done so, WM will load it from disk; check to ensure that the "Original Size" readout matches what you expect. If you want to import a file as a Bitmap rather than a Heightfield, make sure that "Interpret as RGB" is checked.

### Locating your file in the world

It's very important to understand the relationship between your imported file, the world, and the area being viewed or exported. You can use file inputs for a great many things, from importing very large, rough guide maps that will be refined further in World Machine, to importing specific small features.

When you import a file, you also specify its location in world space via the options in the "World Placement" section.



The above view shows how this file input imports to a particular area of the world, which only partially overlaps with the current render extents (the white box).

You can either set the worldspace location numerically, or use one of the provided quick placement buttons to place the location of the file into the current render extents. You may also set the extents graphically by clicking the "Set in Layout..." button.

## Setting your file's elevations

There are several ways you can map the file's elevations into the world:

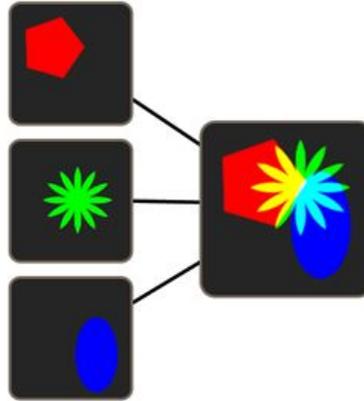
- **Natural Elevations:** Ensure that the height values in the input file are translated correctly into the same elevation in World Machine
- **Full Range:** Map the elevations present in the file so that the highest and lowest elevation completely span the possible elevation range inside of World Machine
- **Specify:** Specify an altitude range in meters that the input data should fall between.

After the mapping of height values, any heights out of range will be clipped to the minimum and maximum altitudes allowed.

You should now have all of the knowledge you need to be able to bring existing terrains into World Machine, operate on them, and export to a file. There is one more file input device, called the Tiled Input Device. It is designed to work on sets of individual tiles that collectively define a terrain region. As a professional-edition only feature, it is described in the Pro Addendum of Chapter 10.

## 7. Bitmaps and Textures

New in World Machine 2 is the ability to work with bitmaps and textures. Conceptually, a bitmap consists of 3 color channels (Red, Green, Blue). These color channels can be manipulated together, or split apart and processed on a per-channel basis as if they were heightfields.



Most devices in World Machine only operate on heightfields. However, some will work on both heightfields and bitmaps, and a few are designed for bitmaps alone.

You can see Bitmap-only devices by using the Filter by Type option on the Tools tab of the toolbar.

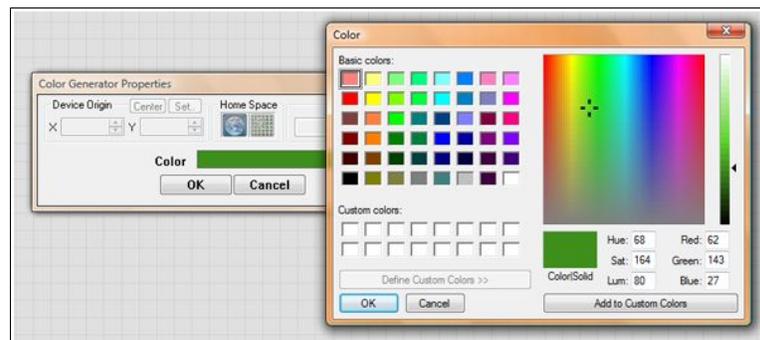


The RGB tab on the toolbar then displays all compatible devices:



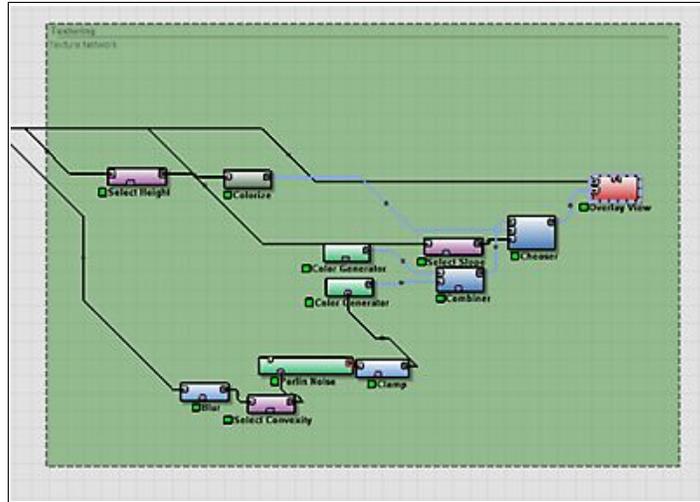
### 7.1 Creating Color

The simplest way to produce color is using the Color Generator device:



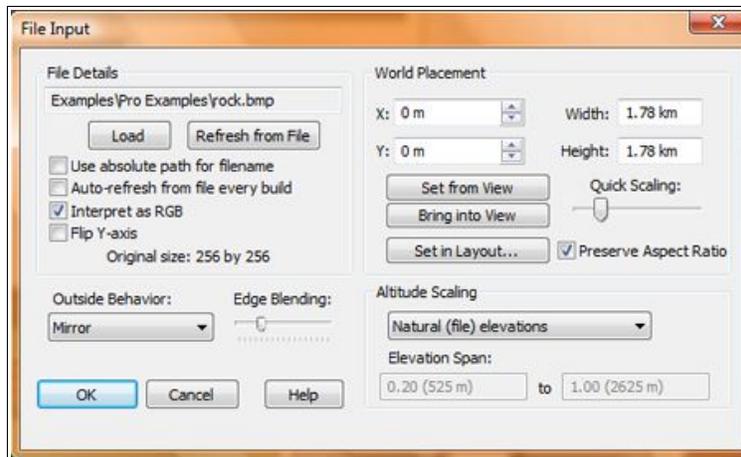
It produces a single color of your choice. By combining several colors via a Chooser or Combiner

device and Chooser masks, you can achieve many different blending possibilities for texturing.

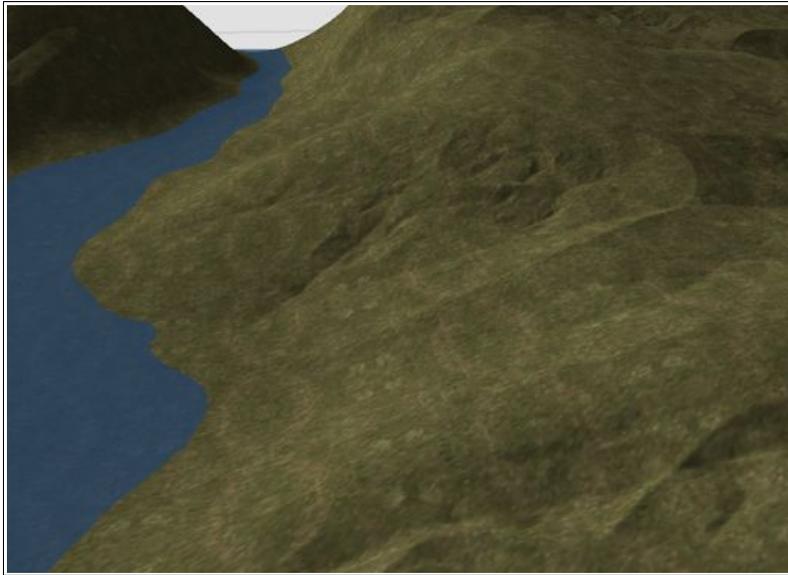


In the example network above, several Selector devices (purple) are being used to pick out various regions of the terrain (such as high slopes, low elevation, and so on), and then merge together various colors based upon that.

There are additional methods you can use to create colors. For example, you can import a tiled texture by using a File Input device and setting an appropriate scale and tiling behavior, as shown below:



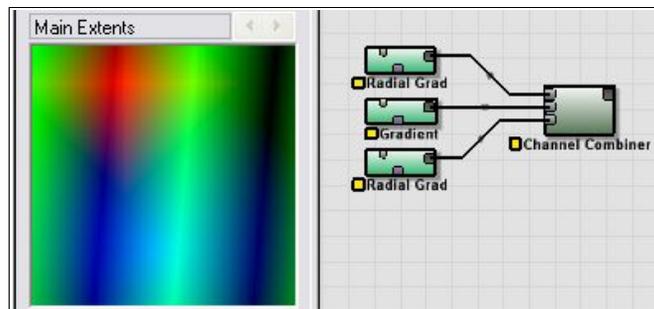
Which overlaid atop the terrain would look like the below:



Of course, once the texture is imported into the world, it can be combined with other textures and colors as much as you want, to produce very rich and complex texture maps.

## 7.2 Working with Color Channels

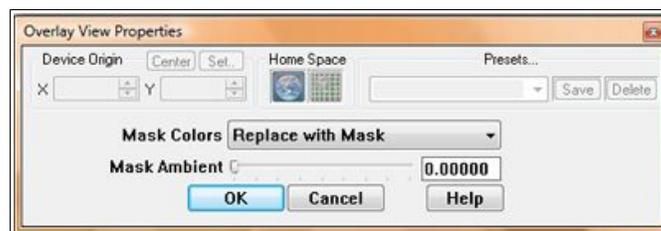
You can combine several heightfields together with a channel combiner to create a single bitmap data packet.



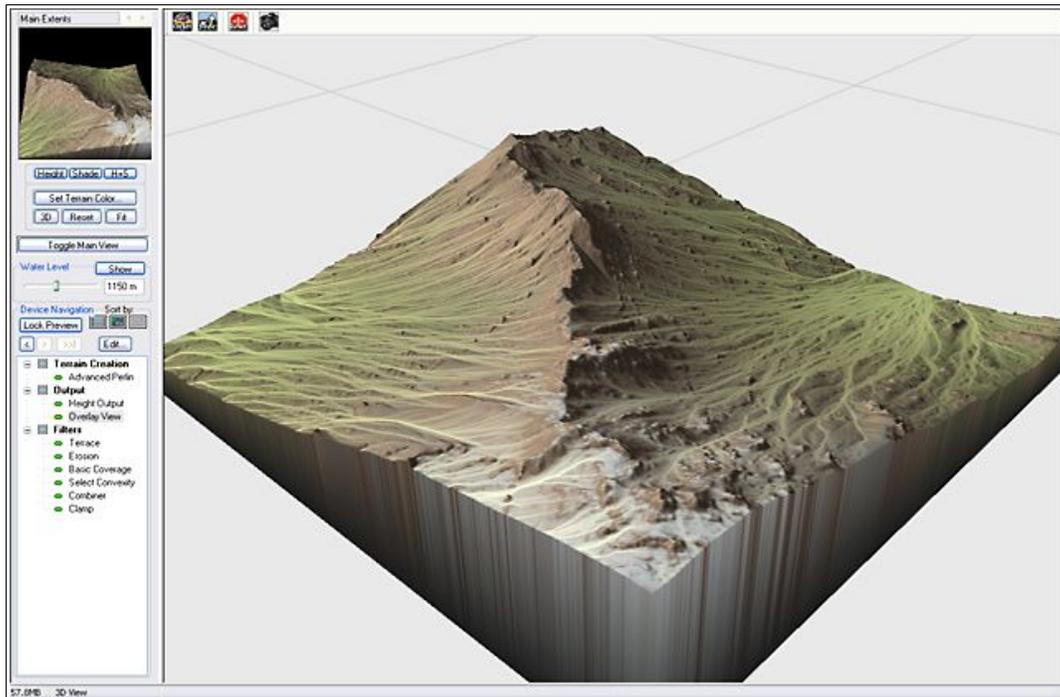
If you want to use a heightfield operator that doesn't support bitmaps (such as the Expander device), you can break apart the bitmap into either RGB or HSL channels, operate on one or all of the channels, and then recombine them using a Channel Combiner.

## 7.3 Overlaying Bitmaps atop a Terrain

If you want to view a bitmap or texture atop a terrain, you can use the Overlay View device to combine the two for display.



By creating your texture from mask data derived from the terrain, you can create realistic (or completely stylized) texturing for further export.

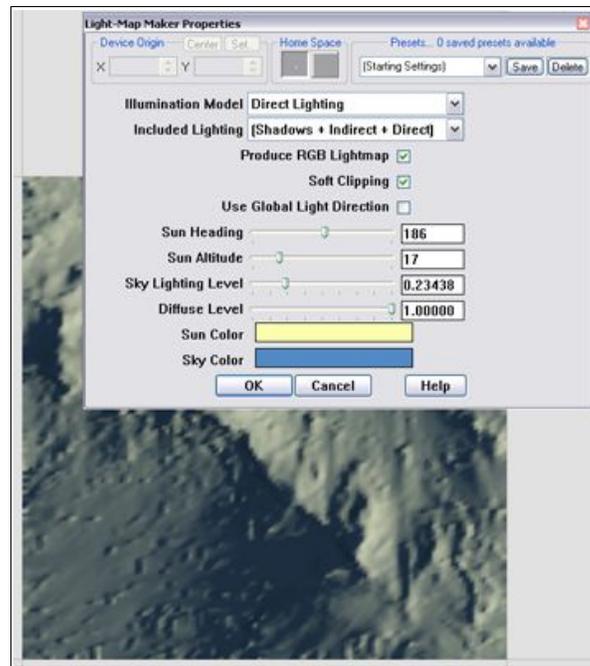


The example above uses some simple elevation and slope-based texturing, along with the flow map output from the Erosion device to create the "snail-trail" watercourse effect.

## 7.4 Specialty Devices

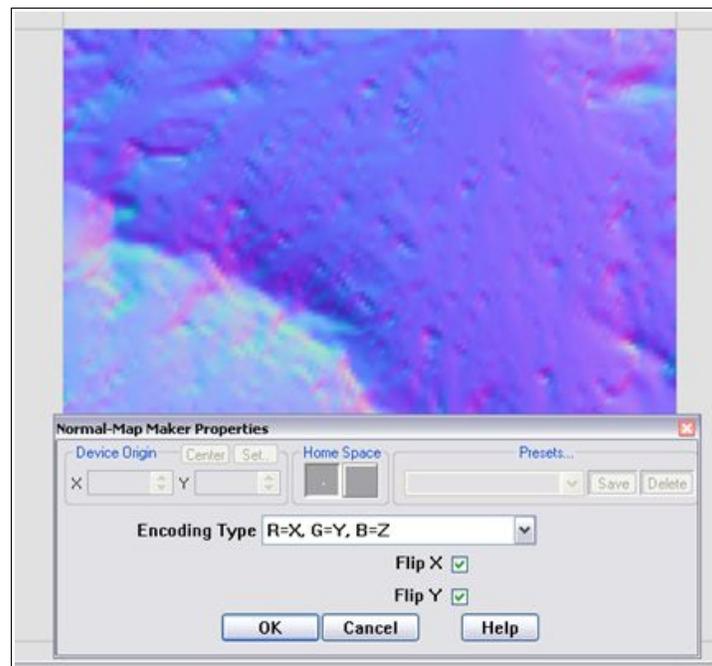
There are two special purpose devices available that produce RGB output as well: A Lightmap Generator and a Normalmap Generator.

### 7.4.1 Lightmap Generator



The Lightmap Generator creates a greyscale or colored lighting map from a source terrain.

## 7.4.2 Normalmap Generator



The normalmap generator encodes the terrain normal into an RGB texture; each direction component of the normal is mapped to a color; (R=X, G=Y, B=Z) is the default encoding. This information can be then used over a low-resolution mesh, giving much of the feel of a high resolution terrain with a fraction of the geometry data.

## 8. Macros

What is a macro?

Essentially, a macro is a single device that captures the functionality of an entire network of World Machine devices. Macros consist of an inside macroworld that is connected via ports to the outside world.



Macros are user-created, and can be saved and traded between users, essentially offering an easy way to encapsulate and exchange useful effects. The power of macros come especially into play when they are parameterized, as they then can contain options, controls, and parameters just like any other device you can place into the network. Extremely powerful macros rivaling custom-programmed device plugins can be created.

### 8.1 Using Macros

#### 8.1.1 Installing a Macro

To install a macro you've received from another user into World Machine, copy the macro file to the World Machine Macro folder (`/macros`). You may create a subfolder under this directory to organize the macro if you wish (see section 8.1.2 below).

The macro will then appear the next time you start World Machine.

#### 8.1.2 Organizing your Macros

The macro system has been designed to be easy for you to organize into whatever hierarchies you prefer. To do so, simply open up the World Machine Macro folder (`/macros`) from the folder you installed WM into.

You can now create subfolders and drag and drop macros to and from them using Windows Explorer! Your changes will be reflected in World Machine upon startup.

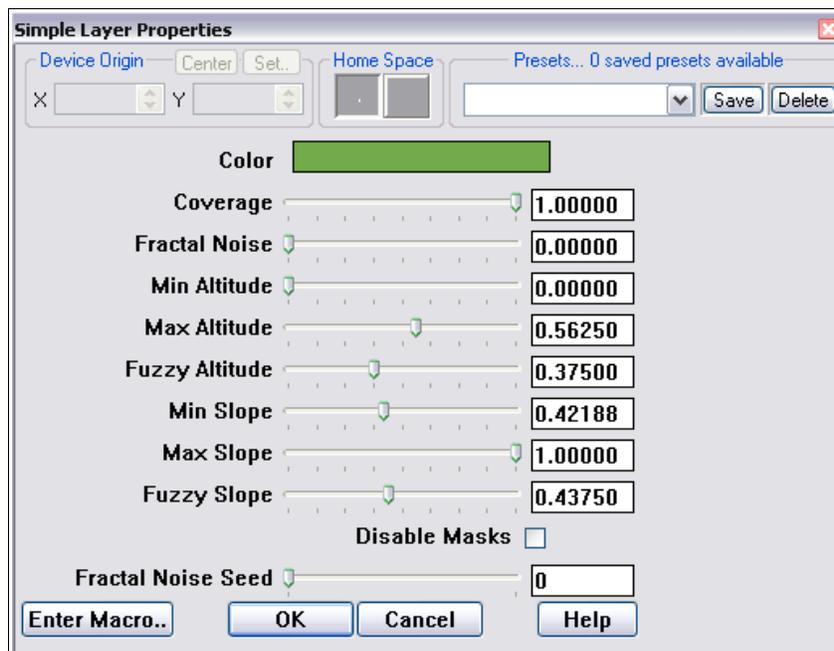
#### 8.1.3 Inserting a Macro into your world

Adding a macro to your world is simple: from the menu bar, select Heightfield Operators->Load Device from Library. You will be presented with a list of the macros installed on your machine.



Clicking on a macro reveals details about it on the right, such as the author, what version of World Machine the macro was created for, and a description of the macro. After you've selected the macro you want, simply click OK to choose that macro. It is now the active drop-device just as if you had clicked on a device in the toolbar. Click anywhere in the workview to place the macro device there.

In your deviceworld, macros behave exactly like any other device. They have required and optional input ports, parameters that can be adjusted, and so on.



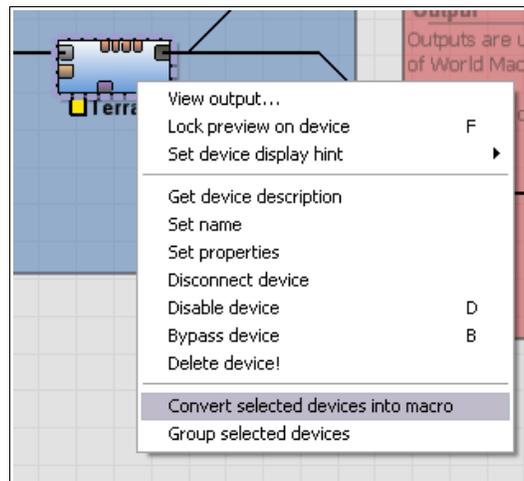
## 8.2 Authoring Macros

### 8.2.1 Creating the macro

There are two methods you can use to create a macro.

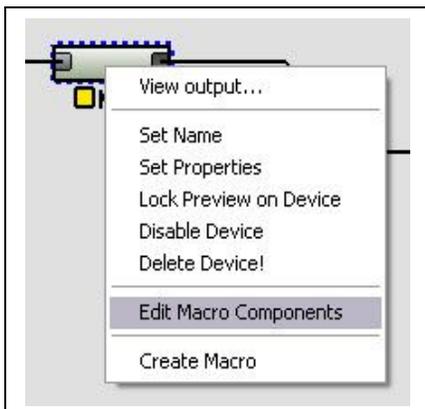
1) You can create a blank macro. Do this by executing the command Heightfield Operators->Utilities->Blank Macro, and click in the work view. The macro will default to having no ports and no internal devices - it is a blank slate ready for your customization.

2) You can convert an existing section of your network into a macro. This option is particularly useful if you've created a specific effect that you want to save and possibly use again in the future.



To do this, select one or more devices and then right-click on one of them and choose "Create Macro" from the pop-up menu. The selected devices will be placed inside of a new macro, and ports will be created on the macro device so that all links into and out of the internal devices will be re-routed with no change in overall network functionality.

### 8.2.2 Inside the Macro World



Once your macro is created by any method, you can enter the Macro World to edit it. There are two ways to do this:

- you can edit the macro by right clicking on it and selecting "Edit Macro Components".
- You can double click on the macro then choose "Enter Macro".

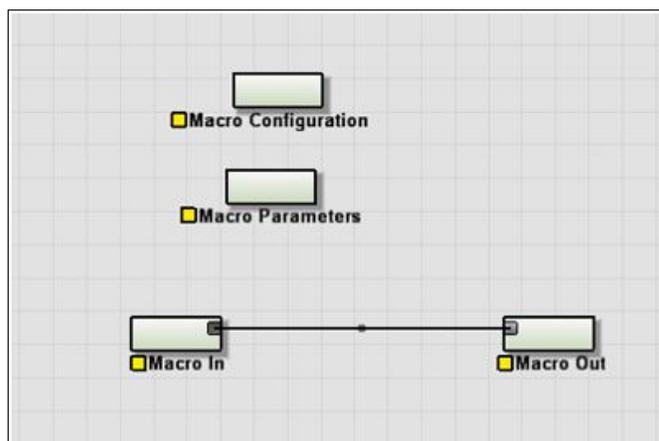
Doing so will change the workview dramatically. The outside device network is nowhere to be seen! Instead, you see several new unfamiliar devices, plus any devices you might have converted to a macro earlier in 8.2.1.

Don't be worried; the outside world is not actually gone. Instead, WM is now showing you the inside of the macro device. A line of tabs will have appeared across the top of the workview, showing you which world you are editing.



You can change back and forth from the macro to the outside by click on a tab; right-clicking on a tab will close the view of that macro.

Macros can be more than one layer deep - your macros may also use macros that consist of still more macros, and so on. You can "tunnel" in as far as you want.



There are four permanent devices inside of all macros. They cannot be deleted -- they are integrated parts of the macro that are exposed for your use. These devices are:

- **Macro In**: Represents the place where outside information flows into the macro world
- **Macro Out**: The place where the results of the macroworld's calculations return to the parent world
- **Macro Config**: A place to provide a description of the macro, as well as information such as the author's name
- **Macro Parameters**: The most complicated of the four macro devices, The Parameter device allows the outside user to interact and change device network settings inside the macro without having to open the macro up to modify it. This is an extremely powerful ability, so much so that several entire chapters will be devoted to describing how best to use this feature.

These four devices are the portals through which information can flow between the outside world and the inner world of the macro. You can configure them in the same way as any other device – by double-clicking them.

### 8.2.3 Ports

The **Macro In** and **Macro Out** devices control the ports on the macro. They are identical except that for the fact that one is devoted to input; the other to output.



To add another port to the macro, simply click "Add New Port". You can then give it any name you want to describe the purpose of it.

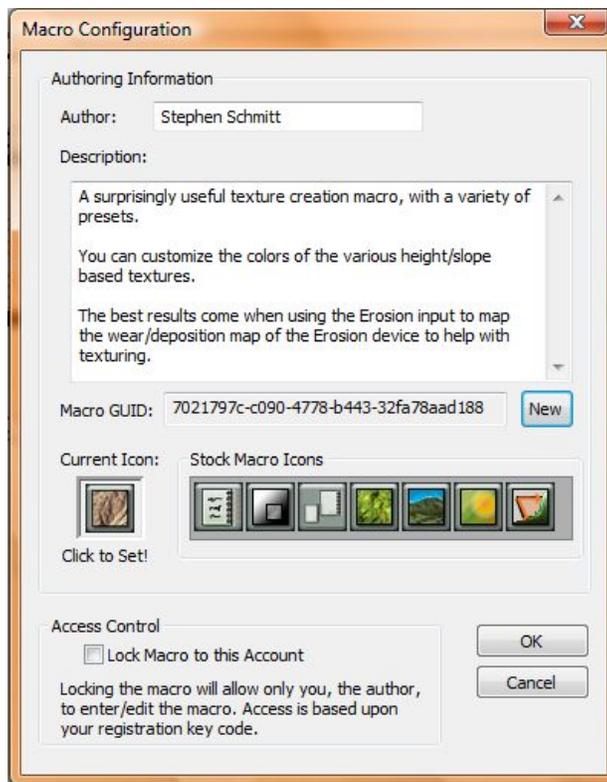
Similarly, to delete a port from the macro, click the delete button next to that entry to flag it for deletion. The port will be deleted when you click "OK" on the dialog.

The "Settings" button brings up a dialog that allows you to configure that port.



The primary decision to make is what kind of data the port will accept. The default is heightfield data, but you can set the data type to bitmap, text, or to accept any type of data. You can flag a port as "optional" as well. A normal port is one that is required for operation of the macro. If a normal input is not connected, the macro will not function. An *optional* input signifies a port that is not absolutely required for proper operation. A typical use of this might be, for example, for an input to allow the user to override a default terrain used inside of the macro, or to specify an optional mask whereby the macro is only applied in certain areas. For outputs, the optional flag is usually used to denote an auxiliary output; perhaps an erosion mask for texturing or something else that is a useful but unnecessary result of the macro.

### 8.2.4: Macro Configuration



The Macro Configuration page lets you set various items that help instruct and inform users of your macro as to its purpose. You can enter your name as well as a description of the purpose of the Macro.

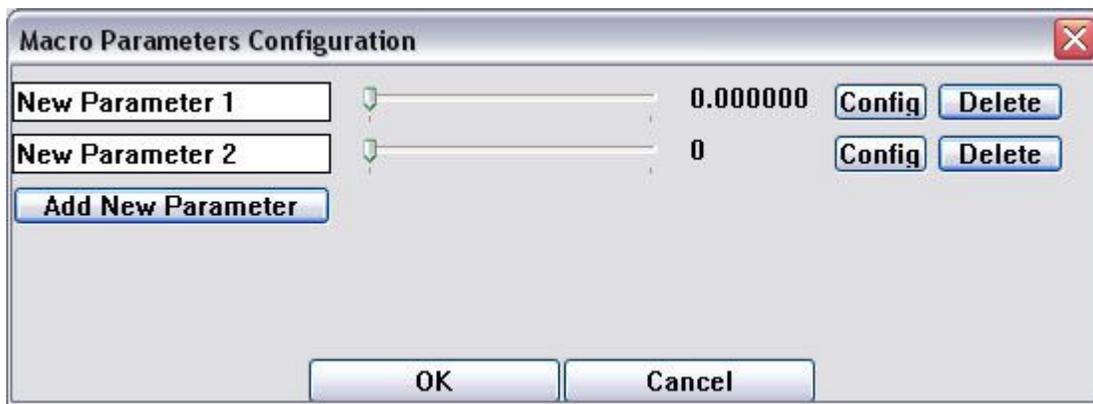
In addition, you can select an icon for the macro that will be displayed in the toolbar. The icon should be a 32x32 24 or 32bit BMP file. If you don't wish to use a custom icon, you can choose a stock icon to use for the macro from the stock icon library.

Lastly, in the Professional edition only, you may specify that your macro should be locked. This means that other users can only use your macro - they cannot open the macro or edit it. In the spirit of information sharing, this option should be used rarely. World Machine uses a one-way hash of your registration key to determine macro access; thus, only you, using your registration code can access the internals of the macro.

## 8.2.5 Parameters

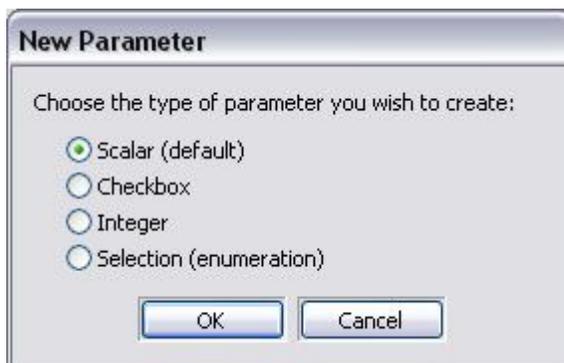
Parameters are the most powerful aspect of macros. [Section 8.3](#) will delve into them in great detail; here we only concern ourselves with the actual options shown in the dialogs.

The parameter device looks like this:



As can be seen above, from the main Macro Parameter screen you can do the following things:

**1) Add new parameters.** Click on the "Add Parameter" button to add a new parameter to the macro. You will then be presented with a choice of the type of parameter you wish to create.



The default parameter type is a scalar; this can be directly used to adjust any device parameter in World Machine. However, several other types are possible and will be useful depending on the purpose of the macro. For a full discussion of parameter types, see [section 8.3](#).

**2) Change the current value of a parameter.** Adjusting the slider or entering a new value is exactly equivalent to doing the same thing to the outside of the macro device.

**3) Rename a parameter.** This is pretty self-explanatory!

**4) Flag a parameter for deletion.** Like the Port device, parameters are only deleted once you click OK in the parameter dialog.

**5) Configure any advanced options** and help tool tip for each parameter.



For every parameter, you can assign help text that will appear if the user hovers the mouse over the parameter. The tool tip text should be one or two sentences that give the user a better idea about what adjusting the parameter will do.

The other sections in this dialog are only valid for particular types of parameters:

### **Selections**

- You can add, edit or delete selection entries. Every entry you add here will show up as an option to select from in a list box in the macro.

### **Integers**

- You can specify the minimum and maximum values of an integer to control the range of the parameter.

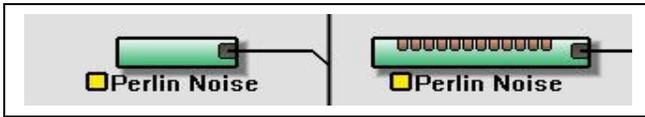
See [section 8.3.3](#) for more details about using these types of parameters.

## **8.3 Macro Parameter System**

Understanding the way that macro parameters, scalar networks, and device parameters interact is critical to being able to create useful and effective macros.

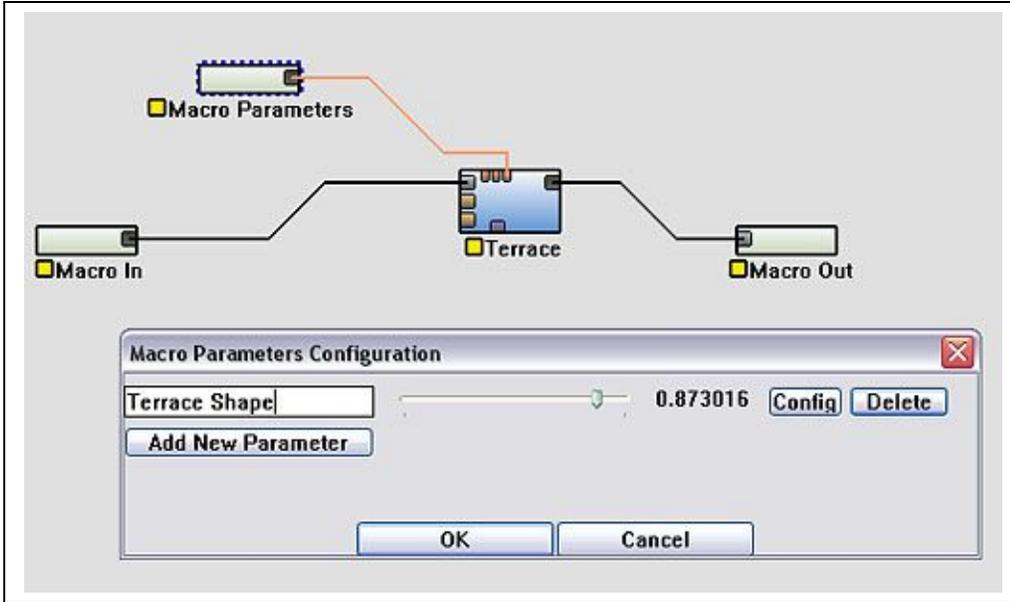
### **Device Parameters Inputs:**

Way back in [Chapter 1](#) we mentioned Parameter Input ports along the top of the devices. They are essential for controlling parameters inside of macros. These appear when a device is selected; you can see the difference below:



### 8.3.1 Simple Parameters

In this case the parameter that you manipulate in the macro is a direct link to the internal device's parameter. It's as if the user could reach into the macro and adjust that device directly. This is the simplest and easiest to understand use of macro parameters.



To create a simple parameter, create a scalar parameter as in [4.2.5](#), naming it as you desire.

After you click OK, the Macro Parameter device will have a new output port. Simply wire that port into the parameter input port of the parameter you want to change.

Note that wires that carry scalar or parameter data are **orange**, not black. Each type of data that is carried in the network has a different wiring color code. Any future data types added to World Machine would similarly have different link colors.

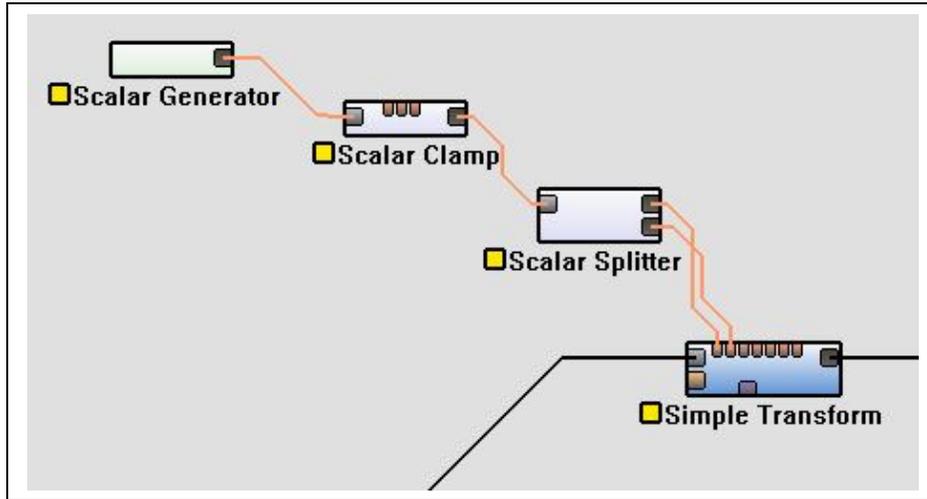
### 8.3.2 Scalar Device Networks

Sometimes you want to do more than is possible with simple parameters. For example, you may want to adjust two related parameters at the same time when the user adjusts a control. The solution to this is **scalar device networks**.



The scalar devices are accessible from the Parameter Operators menu bar. They are also the devices in the parts toolbar with a light blue background shown above.

From left to right, they are the **Scalar Generator**, **Scalar Clamp**, **Scalar Inverter**, **Scalar Arithmetic**, **Scalar Combiner**, and **Parameter Splitter**.



Above is a typical scalar device network. What is it doing?

- 1) The Scalar Generator creates a number, such as **.35**.
- 2) The Scalar Clamp rescales the range of values. Thus, a **0.0** input into the clamp might become a **.25**, and a **1.0** input might become a **.75**. Our value of **.35** would be rescaled according to those new ranges – so it would become **0.425**.
- 3) The Scalar Splitter has multiple output ports so that you can use the value for several parameters
- 4) The scalar value is wired into the Simple Transform device, controlling both the *Canyonize* and *Glaciate* parameters.

So in effect, adjusting one control now adjusts two parameters instead of one – and the adjustments are constrained to be in a smaller range. With the addition of more devices to the scalar network, we could control more parameters, or have them change differently in regards to each other.

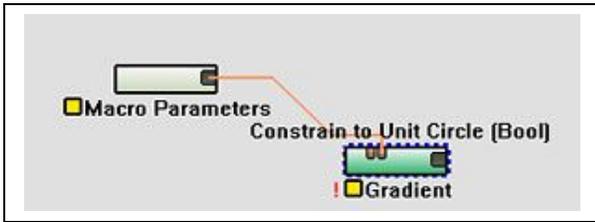
As you can already tell, Scalar networks are strongly mathematical in nature, and can be quite complicated. Most of the challenge lies in figuring out how you want the different driven parameters to change as the Scalar Generator output changes.

Lastly, notice that you can replace the Scalar Generator in the network with the output from a Macro Parameter device. They are interchangeable.

### 8.3.3 Non-scalar Parameters

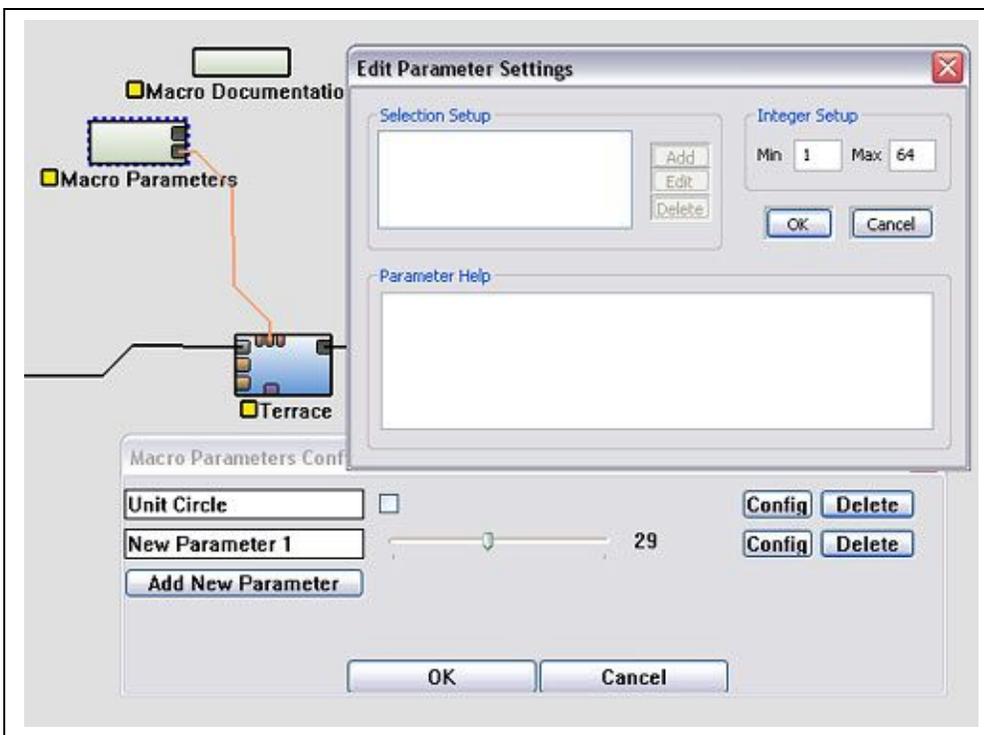
Scalars are not the only type of parameter that you can drive from a macro:

#### Booleans



A Boolean (checkbox) parameter should be used when you want to drive a Boolean device parameter. Above, a Boolean is being used to drive the “Constrain to Unit Circle” parameter on the Gradient device, which as can be noted by the (Bool) on the right of the parameter name, takes a Boolean parameter.

## Integers



Integer Macro Parameters are useful for driving the integer parameter input ports of some devices. Above, an Integer parameter is being used to drive the “Number of Terraces” parameter on a Terrace device.

Note that as shown above, in order to use an Integer parameter, you must push the “Config” button in the Macro Parameters dialog for that parameter, and set the range of values that the integer can take on.

## Selections

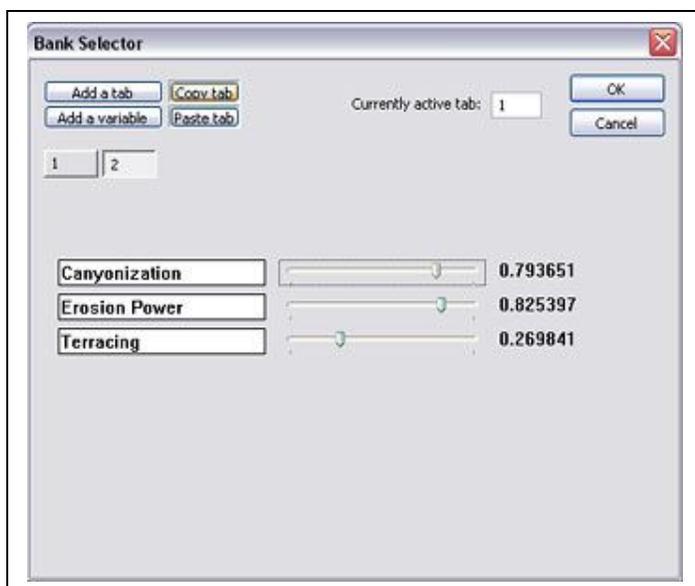
A Selection parameter is, at its core, essentially an integer. However, each value (0,1,2,etc) is labeled. World Machine then presents the user with a list of the labels and allows the user to choose. When used in conjunction with a Bank Selector (see below), Selection parameters become extremely useful.

### 8.3.4 Bank Selectors

The Bank Selector is a powerful utility device designed for parameters. Its purpose is to allow you to drive many scalar values with the input of a single parameter. Doing this allows you to setup “Styles” inside of your macro, so that with a selection from a simple list box, the user can change many (hidden) parameters at once.

The Bank Selector consists of “tabs” and “variables”. Each variable has an output port on the Bank Selector, and will output a scalar value reflecting its current value. A tab contains a set of values for all of the variables. Thus, by choosing a different tab, you can radically change a whole host of parameters that the user need not be exposed to.

The dialog for it is shown below:



There are 6 actions that can be performed in this dialog:

- **Add a variable:** Adds a new variable to the Bank Selector. Each variable can be named, and drives a single scalar output value.
- **Add a tab:** This option creates a new bank of values for all variables
- **Copy tab:** Copies the variable configuration from the current tab
- **Paste tab:** Pastes the stored variable configuration to the current tab
- **Name a variable:** Type a name into the blank area in the variable configuration
- **Adjust a variable’s value:** Adjust the slider to change what the variable’s value should be in the current tab
- **Change tabs:** Adjust the tab to be modified by clicking the numbered buttons above the sliders. The number of buttons will depend upon how many tabs you have created.

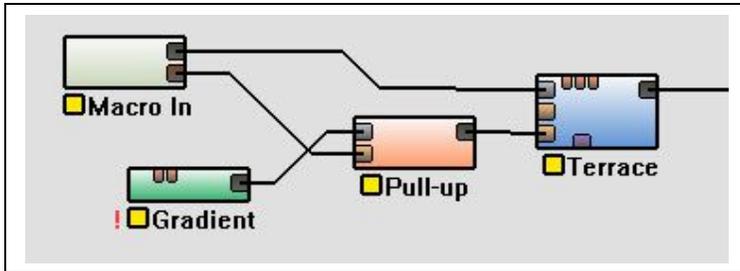
The parameter input to a Bank Selector chooses the current tab, and can be either an integer or a selection – the latter is preferred for user friendliness reasons.

## 8.4 Using Heightfield Utility Devices with Macros

Utility devices are those that don't directly affect the contents of a map; instead, they typically affect the flow of heightfields in the network.

The following utility devices are helpful inside of macros to help direct the flow of data:

### 8.4.1 Pull-up device

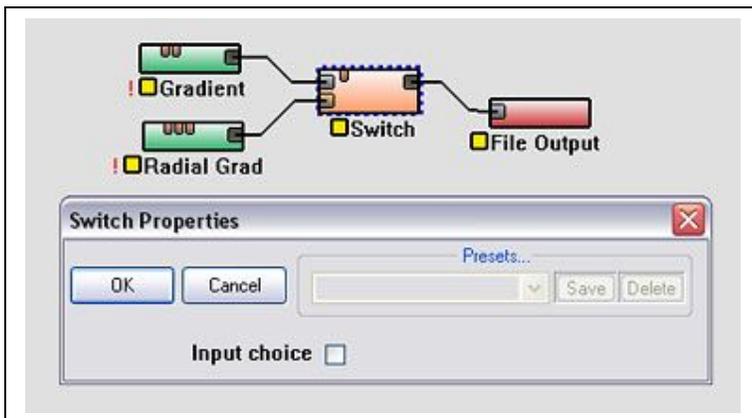


The function of the Pull-up device is to provide a default heightfield if one is not specified. To do this, hook up your reference (default) heightfield to the Reference Input, as shown above. Then the optional heightfield is plugged into the Override Input.

If the Override Input is wired, it will always pass that heightfield along. Otherwise, it will pass the Reference Input along.

### 8.4.2 Switch device

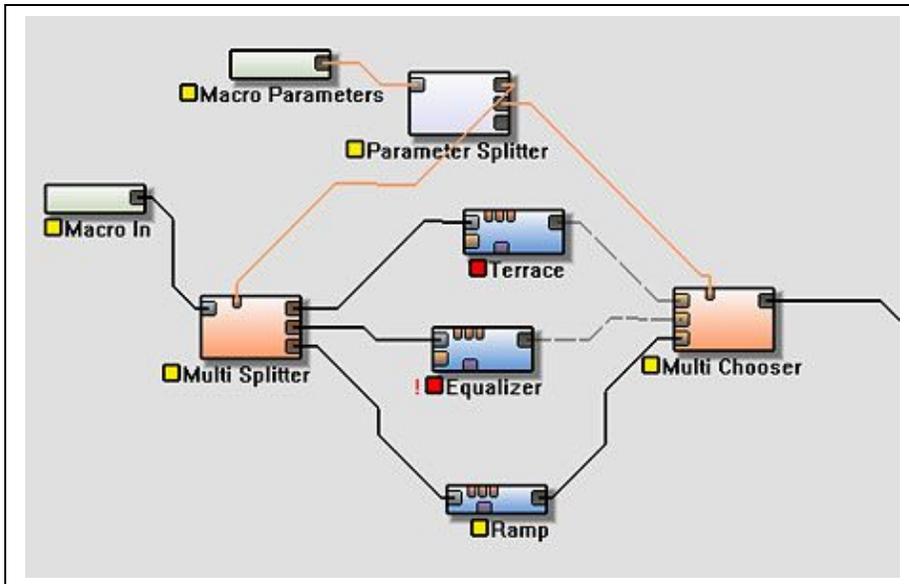
The Switch device allows you to control which heightfield is passed along to the output with a parameter.



Depending on the value of the "Input choice" checkbox parameter, in the example above either the Gradient or the Radial Gradient will be passed along to the File Output device.

### 8.4.3 Multi-Splitter & Multi-Chooser

The Multi Splitter and Multi Chooser are designed to allow you to send the terrain data through only one of many paths through the macro at a time. You can set the maximum number of inputs or outputs, as well as control the chosen input or output through a parameter.



This ability is extremely useful for macros that might have several different effects paths, and where you want to allow the user to choose only one in particular, as in the above illustration. Notice that the paths not currently selected are not built (they have red status icons). This saves the build time and memory for those paths!

## 8.5 Guidelines for Professional Quality Macros

Once you understand the concepts, macros are relatively easy to create. However, to create a professional-quality macro is a different story. There are many design elements, some subtle and others obvious, that go into designing a quality macro. The following guidelines are a good starting point.

### 8.5.1 Limited Focus

The first guideline is extremely simple - and the most important. The most successful macros are those that are focused and targeted on achieving a certain effect. Try not to do too much in your macro - not only will it become overly complicated, but it loses the reusability that makes macros special. When in doubt, create several macros that can be strung together rather than one monolithic macro that attempts to do many tasks.

### 8.5.2 Use the right number of parameters

This is perhaps the most often ignored guideline for macros, and the results can be disastrous. There are two sides to err on here:

Too few a number of parameters is the less common problem. It happens when very important or essential user controls have been left out of the interface. The desire for a professional macro is to have

something that will be useful to many people and is focused on the task at hand; this means that having a macro where you constantly need to open it up and change things in the macro world is very counterproductive.

On the other hand, a far more common mistake is to expose too many parameters to the user. The first impulse of most macro creators is to include as many parameters as possible, under the assumption that more control is always better. The problem is that presenting a smorgasbord of options to the user is extremely confusing - especially when they have non-descriptive names or only change the output very subtly! Every parameter you add dilutes the attention that a user can give to any particular parameter. It's all too easy to drown the extremely important parameters under an ocean of irrelevant options.

The best macros will strike a balance between tune-ability and usability. For each parameter, ask yourself if this control is an important addition to the overall operation of the macro. If you cannot answer yes, then don't provide that parameter! The latest research shows that humans can keep between 4 and 7 'chunks' of information in their mind at the same time. Keeping this in mind, it would seem to be best to limit the number of parameters to the lower side of that range if possible; and if not, then to group together parameters that do similar things spatially in the dialog, a technique called 'chunking'.

Another method to reduce the number of parameters is to use scalar device networks or bank selectors to adjust multiple parameters at a time by just controlling a single parameter in the macro itself. See [section 4.3.4](#) for more details on them.

### **8.5.3 Use the right parameter for the job**

In version 0.99 of World Machine, scalars were the only parameters a macro could have. This resulted in people driving integers and even booleans with a scalar parameter. This is both inefficient and very confusing for the user who expects to be able to smoothly change a setting when they adjust a scalar, and instead sees the output suddenly jump from setting to setting!

The rule is simple: if a device uses an integer parameter, drive it with an integer, not a scalar. Likewise for booleans.

### **8.5.4 Allow only valid parameter ranges**

This may be the second most common sin involving macros. A scalar has a valid range of 0.0 through 1.0. If a certain effect is only really useful in the range of 0.2 to 0.5, then why let the user go outside of that range? It's far more likely to confuse than empower. Use a scalar clamp device to rescale the 0 to 1 range of a macro to the useful range of the effect you're trying to create.

If you have parameters that are valid only at certain times in certain ranges, a far better option is to use the Bank Selector device to setup banks of settings that change according to a list box parameter in your macro. This allows you to change the parameters that need changing without exposing the user to all of them and potentially leading to confusion.

### **8.5.5 Use bank selectors! I'm not kidding!**

This new capability of World Machine has been mentioned several times already in these guidelines,

simply because they are such a simplifying influence on the user experience for otherwise complex macros. List boxes are an easy and labeled way for the user to interact with the macro. [Go here](#) to see details on how to use these devices.

### 8.5.6 Short-circuit untaken options when possible

If you have several different paths through the inside of the macro world, evaluating all of the devices can be expensive. This is particularly true if, for example, you have a parameter that allows the user to choose between two different erosion devices. A naive implementation of this would calculate both of the devices and then choose between them only at the end.

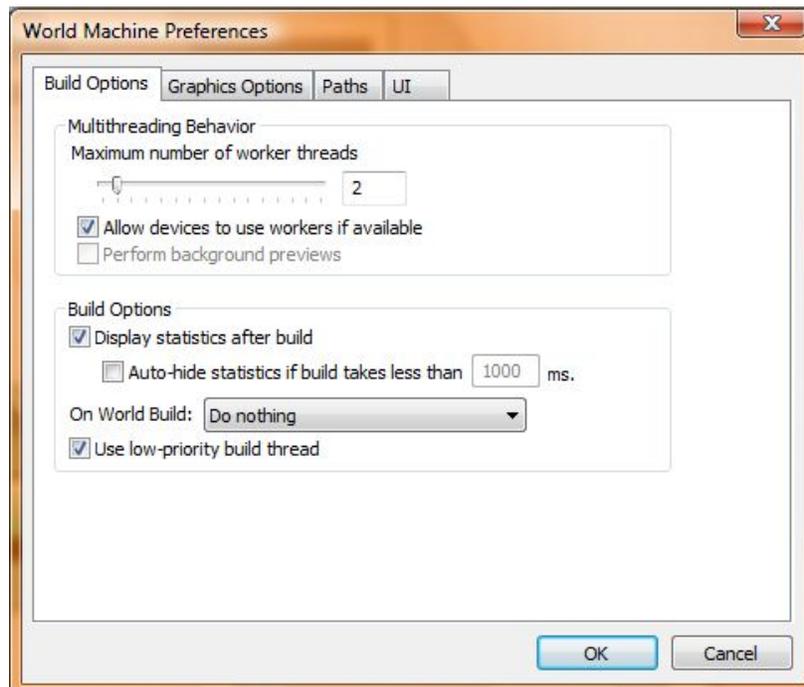
Instead, you can deactivate those paths that will not be taken! This saves the processing time and memory of those paths that no longer will be built. This is particularly important with macros that are going to be used at high resolutions, as the memory used by a set of devices might spell the difference between what can be created on a machine with a certain amount of RAM and what cannot.

See [chapter 4.4.3](#) for more details on using a Multi-Splitter & Multi-Combiner combination to accomplish this.

## 9 Program Configuration

The Preferences dialog allows you to configure various options in World Machine to match either your tastes or your machine's abilities.

### 9.1 Build Options



#### **Multithreading Behavior (Pro Only):**

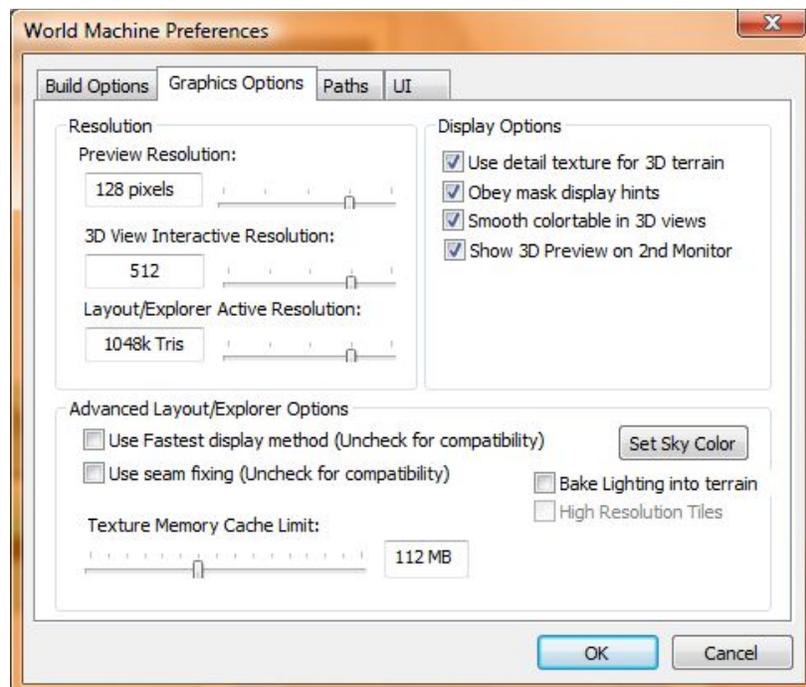
These options are only available in the Professional Edition of World Machine 2!

- **Maximum worker threads:** Set the maximum number of threads to spawn. This should optimally be set to roughly the number of cores that your system possesses. However, you can set it less if you want to reserve some computing power for other applications.
- **Allow devices to use workers:** If set, individual devices can launch worker threads if any are available. Only certain devices are multithreaded, but this option can noticeably speed up normal builds.

### Build Options:

- **Display statistics after build** toggles whether the [progress dialog](#) that appears when the world is built disappears after the build process has completed, or if it remains for you to examine.
- **Auto-hide statistics** can intelligently hide the [progress dialog](#) if the build takes less than a certain amount of time in milliseconds. This is handy, as when you're building just a few devices it saves you a few mouse clicks, while still showing the build stats for longer builds.
- **On World Build** allows you to control what World Machine does by default after the world is built. You may *Do Nothing*, *Show the Last Active View*, *Show the 2D View*, or *Show the 3D View*.
- **Use low-priority build thread** means that World Machine will not monopolize the processor, allowing other applications to pre-empt the build process easier and enabling you to have a build proceeding in the background with less noticeable impact to your other computing activities. However, the build will take longer to complete.

## 9.2 Graphics Options



### Resolution:

- **Preview Resolution:** Controls the detail level of the preview terrains that World Machine creates (used for the Leftside preview window and the 3D View before a device is built). The higher the resolution, the greater the amount of detail in all of the previews, however, they will take longer to create. A value of 64 to 128 is appropriate for most computer systems.
- **3D View Resolution Limit:** Limits the maximum amount of detail that World Machine will allow in realtime in a 3D view. This is intended to allow users with less-powerful video cards to throttle down the 3D views to a level that works smoothly on their computers. Modern systems can leave this at 1024; older computers may need to limit to less.
- **Layout Explorer Resolution:** Control the amount of detail to allow in Layout and Explorer views. Adjust this setting to achieve a fluid framerate while still viewing as much detail as possible.

### Display Options:

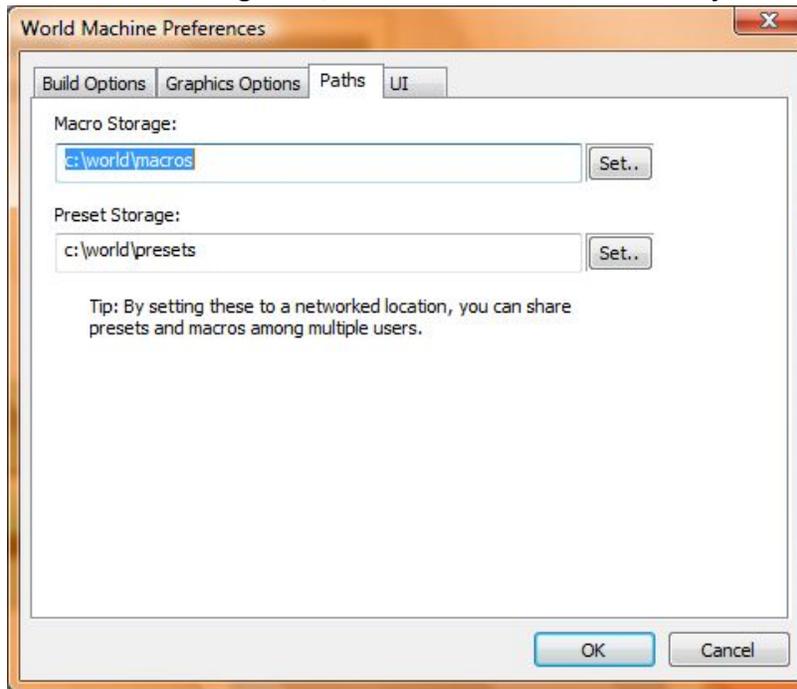
- **Use detail texture:** Enabling this will use a detail texture in the 3D view to help provide a sense of scale for the terrain.
- **Obey mask display hints:** When this is checked, heightfield hints are obeyed. See Section 2.x for more information on display hints.
- **Smooth colorable:** Smooths the color transitions between the different heights. There is no performance hit for this option, therefore it is recommended that it remain enabled.
- **Show 3D on 2nd monitor:** When checked, World Machine will create an extra window that displays the current preview output to a secondary display.

### Advanced Display Options:

- **Use fastest display option:** When this is enabled, World Machine will use OpenGL Vertex Buffer Objects to manage Layout/Explorer meshes. This results in the highest frame rates possible, however, some older and less compatible video cards have issues with VBOs. If you are experiencing random crashes in Layout view, try unchecking this option.
- **Use seamfixing:** When checked, Layout and Explorer mode will try to fix any seams that exist across tiles by performing blending. This option is recommended to be enabled, but it does cause a slight performance hit so you may disable it if absolutely necessary.
- **Bake Lighting into terrain:** When enabled, this setting turns off the real-time lighting of the terrain, and uses light maps to create shadows. This tends to be faster for most video cards. Note that with this enabled, moving the sun will cause all tiles to be regenerated.
- **Set Sky Color:** This option sets the color of the sky in Explorer View. The default is a light blue. You didn't really expect it to do something else, did you?

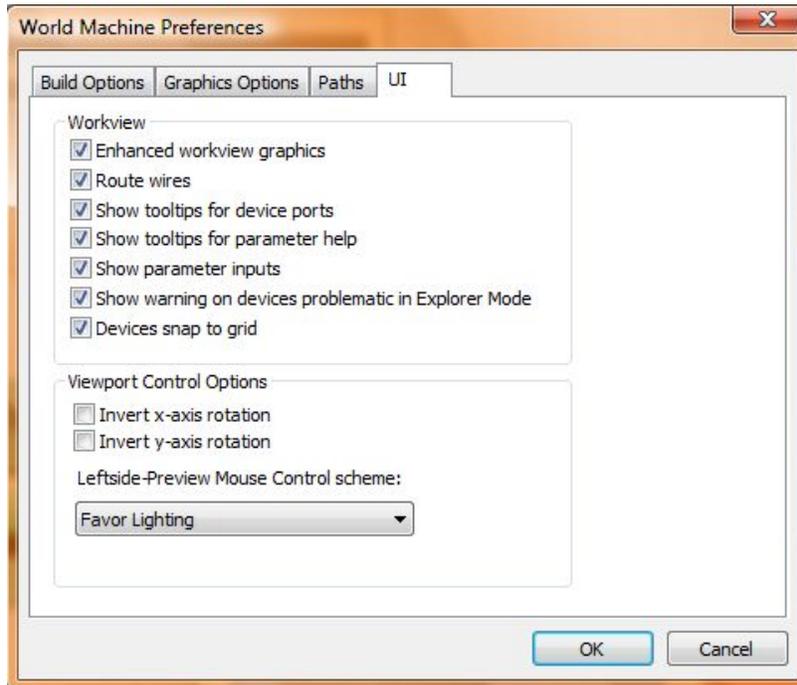
### 9.3 Paths

The Paths dialog simply allows World Machine to use a specific folder to store its presets and macros in. Typically you will not need to change this; it would be useful to do so if you wanted to share macros



and presets across a local network.

### 9.4 UI Options



**Workview:**

- **Enhanced workview graphics:** Shades the device network using gradients and a more aesthetically pleasing method than the older, traditional shading.
- **Route wires:** Makes all wires run at only 0, 45, and 90 degrees, providing more clarity to the device network.
- **Show tooltips for device ports:** Displays the type of data that the port accepts, as well as the current range/value for parameter-type ports.
- **Show tooltips for parameter help:** Controls the help displays for parameters in the device configuration dialogs. To display help text, simply hover the mouse over the parameter name.
- **Show parameter inputs:** Toggles the display of the parameter input ports along the top edge of the devices. The ~ (or `) key toggles this option from the workview.
- **Show warnings on devices:** Displays a red ! mark next to the name of any devices in the network that are likely to not properly display when using Explorer mode. Any devices that display this symbol will not accurately represent the real shape of the terrain when viewed in Explorer mode, as explained [here](#).
- **Devices snap to grid:** When moving devices, when this is checked they will snap their coordinates to grid points.

### Viewport Control Options:

These settings change the navigation behavior of ALL 3D views in World Machine (Leftside, Main 3D View, and Explorer Mode).

- **Invert x-axis rotation:** When this is checked, moving the mouse left and right will cause the view to rotate in the opposite direction from the usual.
- **Invert y-axis rotation:** Turning on “Invert-y” is a popular option that makes the 3D views behave more like an airplane; pulling back on the mouse moves the view up instead of down.
- **Leftside Preview Mouse Controls:** By choosing between Favor Rotation and Favor Lighting, you can control which functions get assigned to the Left Mouse Button (LMB), the Right Mouse Button (RMB), or the Middle Mouse Button (MMB). Note that if you don’t have a middle mouse button, you can press the left and right together to simulate it.

## 10. World Machine Professional Edition Addendum

The professional edition contains additional features targeted at the visualization and game development industries. Several of these new features are transparent; you use World Machine the same as before except with additional power.

### 10.1 Transparent Improvements

#### 64bit Support

Workstations running 64bit Windows XP or Vista can use the 64bit World Machine executable. This removes the often-encountered 2GB 32bit memory addressing limit, and allows World Machine to take advantage of as much memory as your workstation contains.

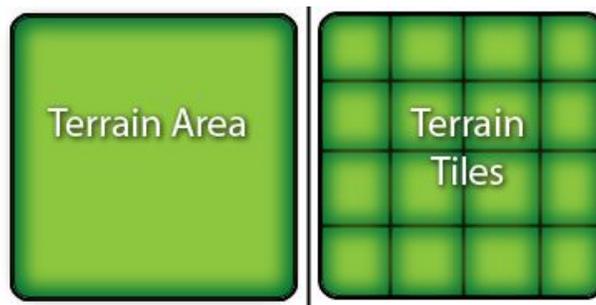
#### Massive Multithreading

The dramatic improvements in performance possible with multi-core processors allows for greatly enhanced speed and usability. World Machine Professional edition supports up to 16 simultaneous processors for use during both normal and tiled builds. In addition, they are used to accelerate the view during Layout and Explorer Views, dramatically reducing the time to see changes and improving the utility of these views.

(See Section 9.1 to configure the allowed number of threads)

### 10.2 Tiling

A tiled terrain is simply a terrain stored across many individual files instead of a single large file. The files are named similar to each other in some logical fashion. This can be advantageous for several reasons -- some display engines may be able to page individual files in and out of memory to improve performance, and in a shared environment individual artists can work on separate files without issue.



Most importantly however, the practical resolution limit for a single heightfield in World Machine is around 8192x8192; even with 64bit systems working with a single file this size or larger can become quite unwieldy. Tiling offers a way around this problem. By saving the terrain area as a rectangular set of individual files, you can export vast areas of terrain at effectively unlimited resolution.

World Machine 2 Professional supports seamless use of tiled terrains; you won't have to worry about individual files, instead you will be working with the tileset as a whole. Tiled terrain support is not

limited to export; you can complete the circle by importing tiled terrain datasets, allowing you to work with datasets that may span many hundreds of files and multiple-gigabytes easily and intuitively.

### 10.2.1 Tiled Output

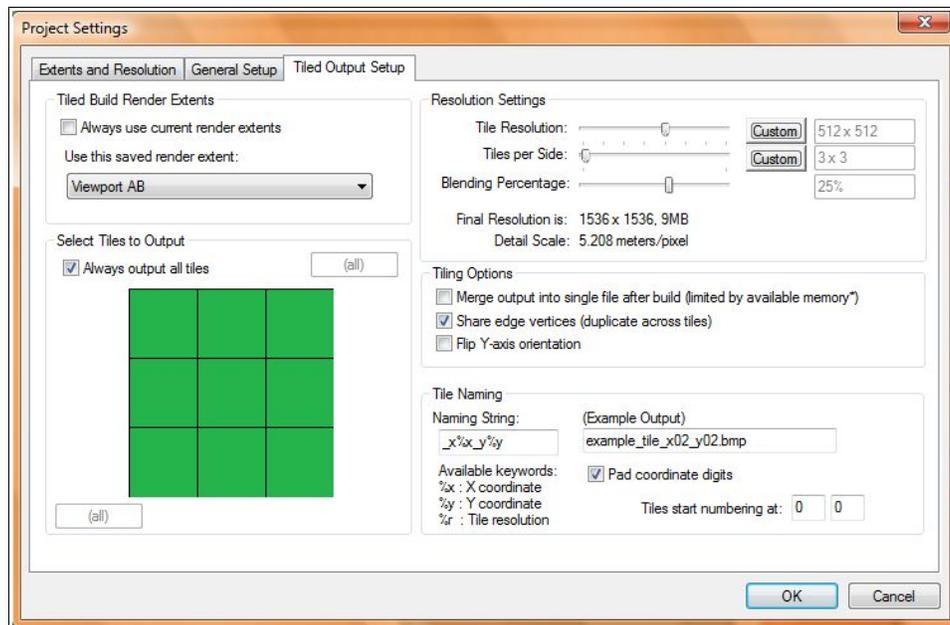
There are three steps required to export a tiled terrain:

1. Set the tiled export parameters described below.
2. Enable tiled export for each of the outputs you wish to export.
3. Select the Tiled Build action from the toolbar or menu to commence building.

We'll follow each step required for export in order.

#### Step 1: Tiled Export Parameters

Set the Tiled Output Setup by visiting the 3rd tab of the Project Settings Dialog:



#### Tiled Build Render Extents

Set which render extent you wish for a tiled build to use; by default it will always use the currently set render extent, but you can override this behavior to have the tiled build always be performed on a particular render extent.

#### Tile Output Subset

You can optionally export less than the entire tileset; this is useful if you have made modifications to only certain regions of your world and want to overwrite only those specific tiles.

#### Resolution Settings

- **Tile Resolution:** The resolution of each individual tile. You can use the slider to select a power-of-two size, or click the Custom button to enter a custom tile size.
- **Tiles per Side:** The number of tiles per side to export. The final resolution of the tileset will be

the tile resolution times the number of tiles per side.

- **Blending Percentage:** The blending percentage controls the amount of additional data created for each tile to ensure that terrain behaves correctly across tile edges; see "Potential Tile Issues" below for more information.

### Tiling Options

- **Share Edge Vertices:** This option controls how edges are handled across tiles. When Share Edge Vertices is checked, the edges of each tile will have the same value as the edges of each adjacent tile. This allows any meshes created from them to sit seamlessly against each other.
- **Merge Output into single file:** When this option is checked, World Machine will attempt to merge all of the tiles into a single file. The ability to take this action is largely limited by available memory, as it needs to be able to allocate the entire container file at once. If it fails, it will export tiles like normal.
- **Flip Y-axis orientation:** Invert each tile on the Y axis; useful when converting between a coordinate system where Y increases downward versus upward.

### Tile Naming String

A tile is located in the tileset by a coordinate appended to the filename. During a Tiled Build, the naming string appends the coordinate of the tile to the output filename. You can control the string with keywords:

Keyword	Action
%x, %y	Insert the X or Y coordinate respectively into the name
%res	Insert the resolution of the tile into the name

The default naming string is : `_x%x_y%y`

*An Example using this String:*

You set in File Output : `test_output.png`

**Normal Build** exports: `test_output.png`

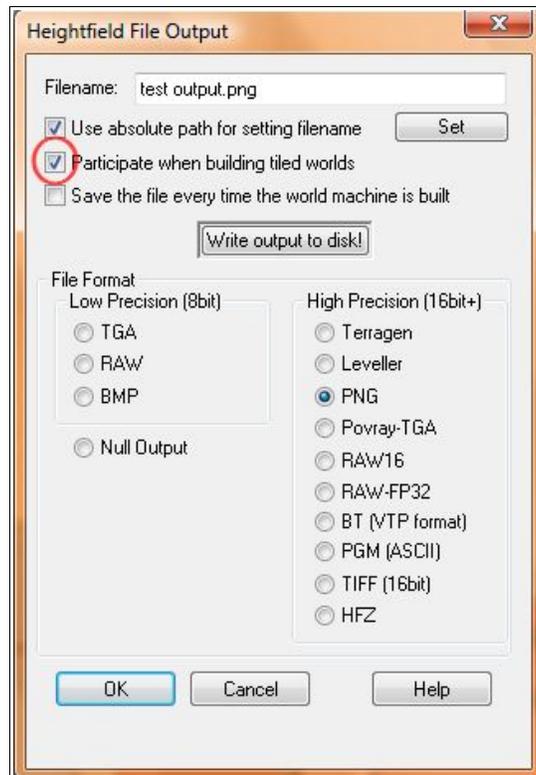
**Tiled Builds** will export: `test_output_x5_y9.png` for the tile that is at location (5,9) in the tileset

There are several additional options when naming tiles:

- **Pad coordinate digits:** This determines whether leading zeros are applied to the coordinate numbers. for example, in a 30x30 tileset:
  - Padding *checked*: `test_output_x08_y15.png`
  - Padding *unchecked*: `test_output_x8_y15.png`
- **Tile numbering start:** The number to start counting the tileset from. Usually either 0 or 1.

## Step 2: Enabling Tiled Export

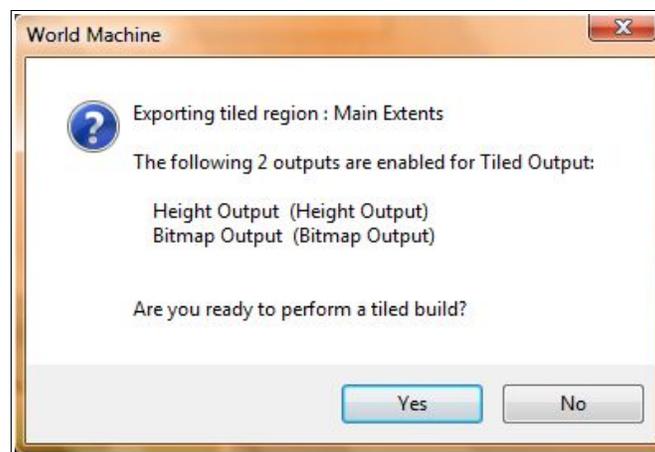
Each output that you wish to save during a tiled export must be enabled for that purpose. To do this, visit the properties page of the output in question. Below is the example for Heightfield Output.



Make sure that the "Participate when building tiled worlds" checkbox is enabled, and that you've set a filename and type.

### Step 3: Build the World

Select the Tiled Build option from the toolbar or menu to commence building. A confirmation dialog will appear, showing which outputs you enabled in the step above for export:



Clicking Yes will begin the build. The tileset will be exported to the same folder as the single file would be created in during a normal build.

### Potential Issues with Tiling

There is one major potential issue to be aware of when dealing with tiled output. Some devices,

including the simulation-based devices such as Erosion and Snow, can produce different results in tiled mode versus normal builds. The reason for this has to do with the context-sensitive nature of those algorithms; when tiling, they do not have access to information from outside of the tile region.

World Machine compensates for this by building an extra area around the tile and then blending between the tiles during the "Merging" Phase of the build. You can control this with the "blending percentage" slider in the Tiled Output setting. In general, terrains with light erosion can use very little blending, while terrains that have huge amounts of variance across tiles will need more blending.

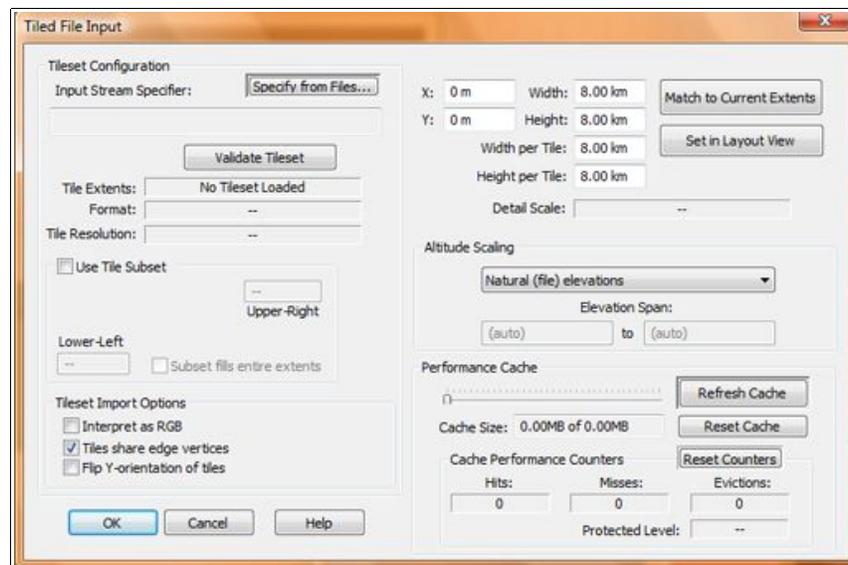
## 10.2.2 Tiled Input

The ability to import a tiled terrain set completes the circle, allowing for data to flow round-trip in and out of World Machine and other software. The power of the Tiled File Input device lies in the fact that it maps the entire tileset into a user-defined area in worldspace. From there, you can work with as large or small a portion of the tiled area as you want. In addition, since the Tiled File Input is simply another device in your world machine workflow, you can have as many input streams as you wish, allowing you to use many possibly overlapping tiled datasets, or to import texturing as well.

To place a Tiled File Input device into the world, select it from the toolbar:

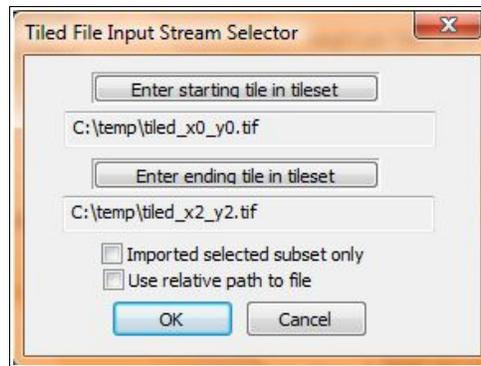


Editing the properties brings up the following dialog:



### Establishing the tileset input stream

Once you click "Specify Input Stream", it will bring up the following dialog:



By specifying two tiles within the set of tiles you want to import, World Machine will attempt to determine the naming scheme in use for tile coordinates. In general, as long as each tile is identified with a discrete coordinate within the tileset, it will be able to parse out the naming system.

By default, World Machine will determine the total extents of the tileset even if they lie outside of the two files you specify; for example, even if you selected "tile\_3\_1" and "tile\_5\_2", World Machine would search for all other matching tiles and determine the full set by itself.

There are two extra options available while specifying the input dataset:

- **Import Selected subset only:** If you select this option, only the tiles that lie between the selected ones will be imported. In the example mentioned above, with this checked, only the rectangular area of tiles between tile (3,1) and tile (5,2) will be imported.
- **Use relative path to file:** allows you to locate a dataset by relative path rather than absolute. If the dataset is located in a folder that is underneath the location specified in Project Settings, the path will be stored as a relative path; otherwise it will remain absolute.

Once you have selected the tilestream, World Machine will attempt to validate the tileset by determining the tileset extents and making sure that each tile exists.

## Establishing world space location

The upper-right corner of the dialog allows you to specify where in worldspace the tileset should be located. You can set the location and size of the overall tile area, or set the size of each tile and let WM calculate the total size for you. If you prefer, you can set the location graphically by clicking on the "Set in Layout View" button.

## Setting your file's elevations

There are several ways you can map the tileset's elevations into the world:

- **Natural Elevations:** Ensure that the height values in the input files are translated correctly into the same elevation in World Machine
- **Full Range:** Map the elevations present so that the highest and lowest elevation completely span the possible elevation range inside of World Machine
- **Specify:** Specify an altitude range in meters that the input data should fall between.

After the mapping of height values, any heights out of range will be clipped to the minimum and

maximum altitudes allowed.

## Tileset Options

You can set options for tiled input, including:

- **Interpret as RGB:** When toggled, the Tiled Input device will produce a Bitmap output instead of Heightfield output.
- **Share Edge Vertices:** If World Machine should assume that the edges of each tile will match. When this option is not set correctly, there will be a one pixel offset in your tileset; Check the export options from World Machine or your other source of tiled data to see what this should be set to.
- **Flip Y-axis:** Enable this if the source of your tiled data exported each tile using a top-left origin rather than bottom-left; You will be able to tell because each tile will be in the right order but upside down.

## Performance tuning

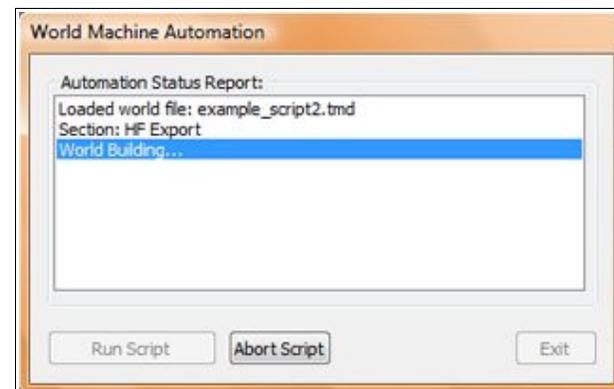
The Tiled File Input device uses a multi-resolution cache to making working with your tileset easy. The default settings will usually achieve optimal performance; however, you can adjust the total amount of space allocated to the cache if you want to accomodate an extremely large tileset.

## 10.3 Automation

World Machine Professional supports a simple but useful form of automation using XML-based scripting. This allows you to load, build, export, and modify basic world settings all from the commandline.

You can launch automation in two ways:

1. From within the World Machine GUI, by choosing "Run Automation Script..." from the File menu:
2. From the commandline, by dragging a WM script onto the executable, or launching World Machine with the name of the script file as the parameter.



### 10.3.1 A Brief Introduction to XML

For those not familiar with the XML markup language, it is quite similar to HTML in overall approach. You create an XML document by using tags denoted with the `<tag>` and `</tag>` markers. Unlike HTML however, the application defines its own tags; the tags you define determine what is in the datafile. Tags are hierarchical, and may contain children by defining new tags between the begin `<>` and end `</>` tags. Tags may have any number of parameters associated with them. For example

`<mytag> ... </mytag>` is a valid XML tag

`<mytag1> <mytag2> </mytag2> </mytag1>` is a valid XML relationship; mytag1 is mytag2's parent.

`<mytag name="My.. tag..." action="none" />` is a valid single-tag XML tag with two parameters, name and action.

The tag `<!-- ... -->` is a comment tag. You may put any text into the ... area and it will be ignored by the XML parser.

### 10.3.2 World Machine XML Scripts

Always start the script file with the XML header declaration.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
```

After the header, there must be one and only one root tag, in this case using the `<automation>` container tag.

The tags inside are processed and executed in line order.

There are three main types of commands that you can execute from a script file:

1. **Worldfile Actions:** Load or save a world file (TMD). Each script must include a load command to load a worldfile from disk.
2. **Modification:** Change something within the world. The most common modifications would be to change the output resolution, or enable/disable devices or groups of devices.
3. **Building/Export:** Actually perform the build process and export to files.

### 10.3.3 An Example Script

An example script is shown below. Most of the tags are self-explanatory; a full listing of the available tags is in the following section.

```
<!-- -->
<!-- Example World Machine Pro script file. These
files are in XML format -->
<!-- The example_script2.tmd file is setup to allow us
to quickly change what outputs we're using to export a
low-res mesh and a high-res normalmap.
<!-- -->
<!-- XML header declaration -->

<?xml version="1.0" encoding="UTF-8"
standalone="no"?>
<automation version="WMP2">

<!-- Load a TMD file into WM-->
<load file="example_script2.tmd"/>
```

```

<section name="HF Export">
<!-- Set the resolution to 512, enable the output group
named A, then build and export the world (this builds
the normalmap) -->
  <modify>
    <world res="512" />
    <enable group="A" />
    <disable group="B" />
  </modify>
  <build mode="normal"/>
  <output/>
</section>

<echo text="Group changover, and re-build for mesh
export"/>

<section name="Mesh Export">
<!-- Set the resolution to 64, enable the output group
named B, then build and export the world (this builds
the mesh) -->
  <modify>
    <world res="64" />
    <enable group="B" />
    <disable group="A" />
  </modify>
  <build mode="normal"/>
  <output/>
</section>

<echo text="Example Script Completed"/>
</automation>

```

### 10.3.4 World Machine Scripting Tags : General

**Tag:** <automation> ... </automation>

Parameters:

- version ("WMP2")

This tag is the parent container for the WM automation scripting data. The version parameter must be always equal to "WMP2", this is the currently defined version number for World Machine v2.0 Professional.

**Tag:** <section name="" enable=""> ... </section>

Parameters:

- name ( = string )

- `enable (= "true"|"false"|"yes"|"no" )`

The section tag allows you to group together sets of commands into logical sections. You can name each section, and enable/disable it by changing the enable parameter. A disabled section is skipped during script parsing.

The other important ability of a section tag is to localize failure. If any major command (e.g. `<load/>`) fails, that command propagates an abort message until it hits a section tag or the root container. Thus without sections, any one failure aborts the script. Within a section, only that particular section will abort; other sections will continue to process, which is usually the intended behavior.

**Tag:** `<echo text=""/>`

Parameters:

- `text (= string )`

This tag echoes the text string input to the scripting output display. It has no other effect.

**Tag:** `<load file=""/>`

Parameters:

- `file (= file path )`

Loads a world file (TMD) from disk. You may specify either an absolute or relative file path.

**Tag:** `<save file=""/>`

Parameters:

- `file (= file path )`

Saves a world file (TMD) to disk. You may specify either an absolute or relative file path.

**Tag:** `<build mode=""/>`

Parameters:

- `mode (= "normal"|"tiled" )`

This very important tag builds the currently loaded world. Note that output files will only be created at this point if they normally would be during a build within WM -- this means that your File Output devices must have "Save file every time the world machine is built" checked to save during a normal build.

The mode parameter controls whether a single-file or tiled build is performed. Either build type uses the current appropriate extents & resolution settings of the world file.

**Tag:** <output/>

Parameters: none

Can be called after a world build is performed, this command will export files from all currently enabled outputs.

Unless your File Outputs have "Save file every time" checked, this is the **only** way to export files from your normal build mode script!

### 10.3.5 World Machine Scripting Tags : Modification

World Machine supports a limited ability to modify the state of the device world from a script file. World modifications are all contained within a special <modify> container tag. Within, the various modification tags allow you to do things like enable or disable sets of devices, change the resolution of the world or the render extents used, and so on. By designing your world file with script control in mind, between the modify section and variables (described in section 10.3.6) you should be able to control virtually anything you wish from the script file.

#### Example use of the Modify Container

```
<modify>
  <world res="512" use_extents="1" />
  <enable group="A" />
  <disable group="B" />
</modify>
```

**Tag:** <modify></modify>

Parameters: none

The <modify> tag is the container for all of the other commands in this section intended to modify the settings of the world. It must enclose every other modification tag.

**Tag:** <world />

Parameters:

- res ( = value denoting the resolution of the world to build )
- use\_extents ( = value specifying which render extent to use: zero-based, these correspond to the order they are listed within the Project Setup dialog )

The <world> tag modifies the project file's world settings. You can change the build resolution and render extents used for any following normal build command.

**Tag:** <tile />

Parameters:

- res ( = integer value denoting the resolution of each tile for a tiled build )
- number ( = value giving the number of tiles to export )

- use\_extents ( = value specifying which render extent to use: zero-based, these correspond to the order they are listed within the Project Setup dialog)

The <tile> tag modifies the project file's tiled build settings. You can change the tile resolution, number of tiles, and which render extent is used for any following tiled build command.

**Tag: <enable />**

Parameters:

- device ( = string )
- group ( = string )

This tag allows you to **enable** any disabled single device, or group of devices. You must specify the exact name (case sensitive) of the device or group in question. Specify only a device or a group but not both, one per enable command.

**Tag: <disable />**

Parameters:

- device ( = string )
- group ( = string )

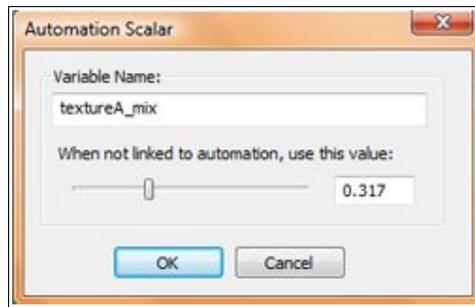
This tag allows you to **disable** any enabled single device, or group of devices. You must specify the exact name (case sensitive) of the device or group in question. Specify only a device or a group but not both, one per enable command.

### 10.3.6 World Machine Scripting Tags : Variables

A variable allows you to pass a value in from the script to influence device settings inside the deviceworld. For example, you could control the amount of erosion, the number of terrace steps, color mixing amounts, or limitless other settings.

You link a variable to the device parameters you want to control by following these steps:

1. Create a <var> tag within the <variables> section. Provide a name and floating-point value parameter in the variable definition.
2. Creating an "Automation Scalar" parameter device in the device world and set the "Variable Name" field to the name of the variable above.
3. Connect the output of the Automation Scalar device to the parameter input(s) on the devices that you wish to control.



### Example use of Variables in the Script file

```
<variables>
  <var name="erosion" value="0.6"/>
  <var name="textureA_mix"
value="0.1"/>
  <var name="textureB_mix"
value="0.55"/>
</variables>
```

**Tag:** `<variables> ... </variables>`

Parameters: none

The `<variables>` tag is the container for variables. All `<var>` tags must be defined within a `<variables>` section block.

**Tag:** `<var name="var1" value="0.1"/>`

Parameters:

- name ( = string )
- value ( = floating point value )

The `<var>` command tag lets you define a variable. Each variable must have two parameters defined: the variable name, and its value. Variables must be defined within a `<variables>` section of the script file.

## 11. Further Help

There are still more resources for additional help with World Machine.

By far the best resource is the [Community Forums](#). It is a place that World Machine users can gather to ask questions, trade tips, and get help with using WM.

The Help section of the [World Machine website](#) will also list any new tutorials and other resources available to you to help learn how to use this software!

