Arctographer User Manual - Version 0.3

Brian Schott (SirAlaran)
briancschott@gmail.com
http://www.hackerpilot.org/

December 2009

Contents

1 Overview

Arctographer is an editor for tile-based game levels. It is primarily designed to work with the ArcLib game programming library for the D programming language, but there is nothing that specifically ties the editor to Arclib. Any program that understands the file format that Arctographer uses will be able to use maps created with it. For details on the file format, see section ??.

2 Download and Installation

2.1 Where to Download

Packaged versions of Arctographer are available for download from http://www.hackerpilot.org/map_editor.php. This includes a Windows installer and a TAR archive for Linux. Development of Arctographer takes place on the ArcLib project site, which is hosted on dsource.org. To check out a copy of the current source code, use the following command:

```
svn co http://svn.dsource.org/projects/arclib/tools/arctographer
```

2.2 System Requirements

1. GTK+ runtime libraries. (GTK+, GLIB, Cairo, etc...) These are included by default on almost any Linux distribution. For Windows, use the installers located at http://gtk-win.sourceforge.net/home/index.php/en/Downloads.

2. Python 2.6. This is included in almost any Linux distribution or Mac OS-X install. For Windows, use the installers located at http://python.org/download/

2.3 Running Arctographer

- Windows: Rename the script “arctographer” to “arctographer.pyw” and double-click on it.
- Linux: Make the script executable through your favorite file manager, or use the command

  ```
  chmod +x arctographer
  ```

Arctographer accepts a file name as a command-line argument. To see other command-line options, run

```
./arctographer --help
```

3 Tile maps

3.1 Overview

3.2 The Tile Palette

3.3 The Layer View

3.4 The Map View

4 Parallax backgrounds

4.1 Overview

4.2 Adding Layers

5 Physics

5.1 Overview

5.2 Adding Shapes

5.3 Modifying Shapes

5.4 Deleting Shapes

6 File Format

6.1 Specification

Arctographer writes files in the JSON file format. JSON was chosen because it is simple, very fast to parse, and human-readable. The root object of the file is the level.

Level

A level consists of:

- one “tileMap” object
- one or more “background” objects
- one “blazeworld” object

tileMap

A tileMap consists of:

- one “layers” array, which consists of layer objects
- one “images” array, which consists of image objects
- one “height” value – an integer specifying the height of the map measured in tiles
- one “tileSize” value – an integer specifying the width and height of a single tile measured in pixels
- one “width” value – an integer specifying the width of the map measured in tiles

layer

A layer consists of:

- one “index” value – non-negative integer specifying the layer’s place in the drawing stack. Layers with a lower index are drawn underneath layers with a higher index
- one “tiles” array, which consists of tile objects
- zero or one “name” values – a string describing the layer
- one “visible” value – a boolean indicating whether or not this layer is visible by default

tile

A tile consists of:

- one “y” value – a non-negative integer specifying the vertical position of the tile
- one “x” value – a non-negative integer specifying the horizontal position of the tile
- one “iy” value – a non-negative integer specifying the vertical position of the image section to use for this tile
- one “ix” value – a non-negative integer specifying the horizontal position of the image section to use for this tile
- one “ii” value – a non-negative integer that corresponds to the “index” value of an image

image

An image consists of:

- one “index” value – a non-negative integer that uniquely identifies this image.
- one “fileName” value – a string specifying the path to the image file. Currently only the Portable Network Graphics (.png) format is valid.

background

A background consists of:

- one “bgColor” vaule – a string in the format “#rrggbbaa” where r, g, b, and a are hexadecimal digits specifying the intensity of the red, green, blue, and alpha components of the color, respectively. The “#” sign is for consistency with HTML markup. The alpha component should almost always be “ff”
- one “parallaxes” array, which consists of zero or more parallax objects.

parallax

A parallax consists of:

- one “index” value – a non-negative integer specifying the position of this parallax layer in the drawing stack. Layers with lower index values will be drawn below layers with higher index values.
- one “hTile” value – a boolean value indicating whether or not the background should be tiled (repeated) horizontally.
- one “vTile” value – a boolean value indicating whether or not the background should be tiled (repeated) vertically.
- one “hScroll” value – a boolean value indicating whether or not the background should scroll horizontally (true), or be fixed on the x-axi (false)
- one “vScroll” value – a boolean value indicating whether or not the background should scroll vertically (true), or be fixed on the y-axis (false)
- one “fileName” value – a string specifying the path to the image file. Currently only the Portable Network Graphics (.png) format is valid.
- one “hScrollSpeed” value – a floating-point number specifying the ratio between camera movement and parallax movement along the x-axis.
- one “vScrollSpeed” value – a floating-point number specifying the ratio between camera movement and parallax movement along the y-axis.
- one “visible” value – a boolean indicating whether or not this layer is visible by default

blazeworld

A blazeworld consists of:

- one “shapes” array, which consists of zero or more shapes
- one “gravityX” value – a floating-point number specifying the strength of gravity in the x direction in $\frac{meters}{second^2}$. This is usually zero.
- one “gravityY” value – a floating-point number specifying the strength of gravity in the y direction in $\frac{meters}{second^2}$. This is usually -9.8.

shape

A shape consists of:

- one “damage” value – a floating-point number specifying the amount of damage taken by an object that comes into contact with this shape per second. Usually 0.0 for terrain.
- one “friction” value – a floating-point number specifying the coefficient of friction for this shape.
- one “restitution” vaule – a floating-point number specifying the coefficient of restitution for this shape.
- one “type” value – a string specifying the type of shape. Valid values are “circle” and “polygon”.
- one “center” value if the shape is a circle, otherwise zero. The center is a “point” object.
- one “points” array if the shape is a polygon, otherwise zero. Each element of the array is a “point”

point

A point consists of:

- one “x” value – an integer specifying the x-coordinate of this point on the map. Units are pixels.
- one “y” value – an integer specifying the y-coordinate of this point on the map. Units are pixels.

6.2 Example

```
{
    "tileMap": {
        "layers": [
            {
                "index": 0,
                "tiles": [
                    {
                        "y": 0,
                        "x": 0,
                        "iy": 19,
                        "ii": 0,
                        "ix": 5
                    },
                    {
                        "y": 1,
                        "x": 0,
                        "iy": 21,
                        "ii": 0,
                        "ix": 1
                    }
                ],
                "name": "Base",
                "visible": true
            },
        ],
        "images": [
            {
                "index": 0,
                "fileName": "tiles/free_tileset_version_10.png"
            }
        ],
        "height": 15,
        "tileSize": 32,
        "width": 20
    },
    "blazeWorld": {
        "shapes": [
            {
                "points": [
                    {
                        "y": 32,
                    "x": 208
                },
                {
                    "y": 64,
                    "x": 256
                },
                {
                    "y": 160,
                    "x": 224
                },
                {
                    "y": 96,
                    "x": 128
                }
            ],
            "type": "polygon",
            "damage": 0.0,
            "friction": 1.0,
            "restitution": 0.0
        },
        {
            "center": {
                "y": 208,
                "x": 176
            },
            "damage": 0.0,
            "friction": 1.0,
            "radius": 80,
            "type": "circle",
            "restitution": 0.0
        }
    ],
    "gravityX": 0.0,
    "gravityY": -9.8000000000000007
},
"background": {
    "bgColor": "#227db8ff",
    "parallaxes": [
        {
            "hTile": true,
            "index": 0,
            "vTile": false,
                "vScroll": true,
                "fileName": "parallax/forest.png",
                "hScrollSpeed": 0.5,
                "visible": true,
                "hScroll": true,
                "vScrollSpeed": 1.0
            }
        ]
    }
}
```