Idris Documentation



Table of Contents

[**Contributors**](#_2c97sveoprb7) **3**

[**Attributes**](#_7mm1ncp4nrck) **4**

[**Game Designs**](#_pc1qvz7rx7u) **5**

[**Functionality**](#_vktex5ervnt4) **6**

# Contributors

This project in its entirety has been worked on and developed by Joshua Zebrowski and Thomas Elliott. This project was started on the 25th of March 2025.

Game Description

Strategy game like advanced wars with a heavy focus on weather effecting the game board such as earthquakes / tusnamis hurricanes ect taking place in current day due to climate change the climate has started to collapse last remaning countires are fighting over the last resources avaible / places you can live european coilition russain federation usa union of nations (mainly brick countries minus russia and china)

# Attributes

# 

**Sea Tile Names & Their Meanings**

\*NOTE: When ‘lake’ is mentioned, this refers to a thick body of water that usually contains a sea tile or at least many water tiles clumped together. For example, the difference between a SeaSemiCircularCorner and a SeaLakeCorner would be that a semicircular corner would be used when the strip of water is one tile wide, whereas lake corners would be used when the strips of water are thicker.

S -> Sea (a tile of water surrounded by all sides with more water tiles)

SS -> Single Sea (an isolated tile of water; surrounded by no other tiles of water)

**Semi-Circular Corners**

SSCTR (semi-circular top right)

SSCTL (semi-circular top left)  
SSCBR (semi-circular bottom right)  
SSCBL (semi-circular bottom left)

**Semi-Circular Straights**

SSCR (semi-circular right)

SSCL (semi-circular left)

SSCB (semi-circular bottom)

SSCT (semi-circular top)

**Straights**

SHS (sea horizontal straight)

SVS (sea vertical straight)

**T-Functions (Junctions)**

STFT (sea t-function top)

STFB (sea t-function bottom)

STFL (sea t-function left)

STFR (sea t-function right)

STFFS (sea t-function fully surrounded)

**Lake Inside Corners**

SLICBR (sea lake inside corner bottom right)

SLICBL (sea lake inside corner bottom left)

SLICTL (sea lake inside corner top left)

SLICTR (sea lake inside corner top right)

**Lake Middle Corners**

SLMCP ( sea lake middle corner positive)

SLMCPCR( sea lake middle corner positive cropped-right)

SLMCNCR (sea lake middle corner negative cropped-right)

SLMCPCL (sea lake middle corner positive cropped-left)

SLMCNCL (sea lake middle corner negative cropped-left)

SLMCN (sea lake middle corner negative)

**Lake Extending Corners**

SLECTRV (sea lake extending corner top right vertical)

SLECTRR (sea lake extending corner top right right)

SLECBLV (sea lake extending corner bottom left vertical)

SLECTLV (sea lake extending corner top left vertical)

SLECBRV (sea lake extending corner bottom right vertical)

SLECTLH (sea lake extending corner top left horizontal)

SLECBRH (sea lake extending corner bottom right horizontal)

SLECBLH (sea lake extending corner bottom left horizontal)

**Lake Corners:**

SLCBL (sea lake corner bottom left)

SLCTL (sea lake corner top left)

SLCTR (sea lake corner top right)

SLCBR (sea lake corner bottom right)

**Lake Straights:**SLHT (sea lake horizontal top)

SLVR (sea lake vertical right)

SLHB (sea lake horizontal bottom)

SLVL (sea lake vertical left)

**Lake T functions**

SLTFB (sea lake t function bottom)

SLTFL (sea lake t function left)

SLTFT (sea lake t function top)

SLTFR (sea lake t function right)

### Terrain json layout

{

"id": "SLECBLH",

"excludeTilePicker": true,

"terrainName": "SLECBLH",

"texturePath": [

"sea/plains/slecblh.png",

"sea/desert/slecblh.png"

],

"defense": 0.0,

"speed": 0.2,

"canBeDestroyed": false,

"damagedTextures": [],

"joinTexturePath": []

}

Id -> contains the map reference of the terrain piece

excludeTilePicker -> used to excluded tiles that shouldnt be able to be visible in the placement bar

terrainName -> The unabreivated name or just type of tile i.e forest or mountain instead of FP OR MD

texturePath -> The path to the texture or textures if its multiple the layout goes as follows

* Index 0 : plains
* Index 1 : desert
* Index 2 : winter

This is how the render grabs the correct texture using the base type of the tile its assigned to

Defense -> Amount of defense the terrain offers when calculating damage on a unit take into account the terrain they are in

Speed -> Amount of movement it costs a unit to move through

UnitRestrictions -> Restricted units that can’t move through the terrain and either need to find a way around or if applicable can use a troop carrying vehicle such as on a sea

canBeDestroyed -> Whether the terrain takes damage or not

damagedTextures -> the different damage states of a terrain if it can take damage can be as many textures as wanted and currently works out texture switch by taking the number of images and dividing the overall health by it to get the increments and when it crosses that threshold it switches the texture for example if you have two then once it gets past 50% health .5 it switches to index 0 texture and when it gets to 0 or bellow switches to the one at index 2

joinTexturePath -> Used for blending terrain thats between two different types for instance a straight vertical sea stuck between desert and plain needs to have the desert side desert themed and the other side plain themed , layout goes as follows

* Index 0 : plain desert
* Index 1 : plain winter
* Index 2 : desert plain
* Index 3 : desert winter
* Index 4 : winter plain
* Index 5 : winter desert

If no texture for a certain one leave “ ” currently alpha 0.0.1 no custom class structure to handle it so leaving it blank will break it and crash the game

# Game Designs

# Functionality

# 