SOROLIF RTS MAP GENERATOR

Parameter Reference

Offline manual - version 1.8.0

All 214 public map-definition fields, asset configuration and 273 material properties, followed by detailed usage conditions and visual parameter examples.

The online documentation contains the complete guide collection and parameter comparison gallery. Authoring sources and standalone images are maintained outside the Unity package.

Source defaults are distinct from saved scene values and tuned biome materials. Use the installed Inspector tooltips and explicit presets when configuring your project.

Contents

Contents 2

Current parameter catalogue 5

MapDefinition . . . 5
General . . . 5
Map Altitude . . . 6
Image Import . . . 6
Strategic Layout . . . 11
Terrain Features . . . 13
Grass Details . . . 13
Shared Scene Settings . . . 18
Optional Outputs . . . 19
Project Integration . . . 19
Terrain Materials . . . 20
Water Settings . . . 21
Cliff Settings . . . 25
Resource Settings . . . 27
Gameplay Constraints . . . 28
Generation . . . 29
Screenshot Settings . . . 29
BiomeSetting . . . 30
Configuration . . . 30
Terrain Materials . . . 31
Biome Prefabs . . . 32
Decoration . . . 32
MapGeneratorImageLayer . . . 34
Configuration . . . 34
GrassPrefabLibrary . . . 34
Configuration . . . 34
Shader material properties . . . 35
Sorolif/RTS Terrain/Autumn Water . . . 35
Sorolif/RTS Terrain/Forest Water . . . 38
Sorolif/RTS Props/Atlas Lit . . . 42
Sorolif/RTS Terrain/Snow Water . . . 43
Sorolif/RTS Terrain/Stylized Grass . . . 46
Sorolif/RTS Terrain/Stylized Water . . . 48
Sorolif/RTS Terrain/Triplanar . . . 51

Sorolif/RTS Terrain/Triplanar Stylized . . . 52

Parameter reference **54**

Grass details (18 October 2026) . . . 54
How to compare settings . . . 54
Corrected behavior . . . 55
Optional outputs and project integration . . . 55
Application by parameter . . . 57
General . . . 57
Map Altitude . . . 59
Image Import . . . 60
Strategic Layout . . . 66
Terrain Features . . . 67
Shared Scene Settings . . . 73
Terrain Materials . . . 75
Water Settings . . . 76
Cliff Settings . . . 81
Resource Settings . . . 83
Gameplay Constraints . . . 84
Generation . . . 85
Screenshot Settings . . . 85
Biome, decoration and image-palette fields . . . 87
Visual examples . . . 91
ForestDensityMultiplier . . . 91
EnablePathSmoothing . . . 92
ImportedMountainHeight . . . 92
AltitudeMaxHeight . . . 92
ForestZoneCountMultiplier . . . 93
MainBaseRampCount . . . 93
HasMountainZones . . . 94
MountainDensityMultiplier . . . 94
MountainCliffMaterial . . . 94
EnableWaterFloorNoise . . . 95
WaterFloorNoiseStrength . . . 95
WaterFloorNoiseScale . . . 96
WaterFloorNoiseSmoothingIterations . . . 96
LakeRadiusMin . . . 96
LakeRadiusMax . . . 97
ChokePoints . . . 97
PlateauRampWidth . . . 98

RampDirtPathApproachLength 98
RampDirtPathApproachWidth 98
RampCliffDepthScale 99
RampCliffInset 99
GenerateNaturalExpansionPlateaus 100
PlateauReliefScale 100
NaturalExpansionSecondAccess 101

Current parameter catalogue

Generated from package 1.8.0 source declarations. Inspector labels and tooltips are authoritative for the installed version. Values below are source defaults; existing scenes, custom presets and authored materials keep their saved values.

The catalogue covers 214 public MapDefinition fields, biome and prop configuration, image-palette settings, the shared grass library, and all eight included shader property blocks. Hidden compatibility fields are marked explicitly.

MapDefinition

General

MapName

Type: `string`. **Source default:** `"New Map"`.

Name used for the generated map hierarchy, saved recipe and screenshot filename.

MapSize

Type: `int`. **Source default:** `70`.

Inspector bounds: `Range(30, 200)`.

Number of logical cells along each side of the square map. Imported images are sampled to this grid resolution.

PlayerCount

Type: `int`. **Source default:** `2`.

Inspector bounds: `Range(2, 12)`.

Requested number of procedural player bases before additional main-base locations are applied.

FullMapSymetry

Type: `bool`. **Source default:** `false`.

Opt into full-map symmetry for procedural terrain, water, paths, forests, resources and props. Disabled by default: keep the original natural generator and player-start placement. Imported images remain authored.

Symmetry

Type: `SymmetryType`. **Source default:** `SymmetryType.Quadrant`.

Player-start layout in the original generator. When Full Map Symmetry is enabled, this also mirrors the whole map and adds missing start locations to complete symmetry groups. None keeps free generation.

GenerationMode

Type: `MapGenerationMode`. **Source default:** `MapGenerationMode.Procedural`.

Choose procedural generation or import terrain classifications from Source Map Image.

Biome

Type: `BiomeType`. **Source default:** `BiomeType.Forest`.

Biome identifier used to select compatible presets. Assign Biome Override for explicit runtime asset references.

BiomeOverride

Type: `BiomeSetting`. **Source default:** `Unassigned / type default`.

Biome asset used by this map. Assign a compatible preset for runtime generation; the Editor selector filters available project assets by Biome Type.

WaterLayout

Type: `WaterLayoutType`. **Source default:** `WaterLayoutType.River`.

Procedural water layout. Random resolves one layout using the map seed before layout-dependent generation.

Map Altitude

EnableMapAltitude

Type: `bool`. **Source default:** `false`.

Adds height-map relief to the map. Lakes and seas stay level; through-flow rivers share a gentle slope toward their lowest outlet. Seabed and nearby banks follow these water constraints. Existing mountain, plateau and path relief is preserved; placed objects follow the adjusted terrain.

AltitudeMinHeight

Type: `float`. **Source default:** `0f`.

Height-map offset in map units at black pixels. Local terrain features are added on top of this altitude.

AltitudeMaxHeight

Type: `float`. **Source default:** `8f`.

Height-map offset in map units at white pixels. Local terrain features are added on top of this altitude.

AltitudeHeightMap

Type: `Texture2D`. **Source default:** `Unassigned / type default`.

Optional readable grayscale height map, independent of Source Map Image. Leave empty for seeded hills. Import with Read/Write enabled, sRGB disabled and no compression.

AltitudeHeightMapOriginTopLeft

Type: `bool`. **Source default:** `false`.

Flip the custom height map vertically. By default its bottom-left pixel maps to the terrain's minimum X/Z corner.

AltitudeNoiseScale

Type: `float`. **Source default:** `0.035f`.

Inspector bounds: `Range(0.005f, 0.15f)`.

Frequency of automatically generated hills in cell units. Lower values give broader hills. Ignored when a custom height map is assigned.

Image Import

SourceMapImage

Type: `Texture2D`. **Source default:** `Unassigned / type default`.

Readable color image used in Image generation mode. Import with Read/Write enabled; the image palette classifies its terrain colors.

ImageLayerSet

Type: `MapGeneratorImageLayer`. **Source default:** `Unassigned / type default`.

Color palette and material-channel mappings used to classify Source Map Image in Image generation mode.

SimplifyImageLayers

Type: `bool`. **Source default:** `true`.

For painted/scanned maps: use only the selected palette, sample a wider area to suppress grain, and merge shades of the same terrain/Unity layer into the most-used reference. Disable to preserve separate painted material variants and outline segmentation.

EnableImageContourSmoothing

Type: `bool`. **Source default:** `true`.

Rounds imported terrain contours with shared mesh vertices fitted to the source image. Preserves tile types, water, resources and narrow passages; independent of cell-color cleanup.

ImageContourSmoothingStrength

Type: `float`. **Source default:** `0.85f`.

Inspector bounds: `Range(0f, 1f)`.

Amount of shared contour rounding for imported surfaces when Image Contour Smoothing is enabled. Zero retains the unsmoothed outline.

ImageOriginTopLeft

Type: `bool`. **Source default:** `false`.

Flip the source image vertically so its top row is used as the map's image origin.

ImageCellSampleRadius

Type: `float`. **Source default:** `0.089f`.

Inspector bounds: `Range(0.05f, 0.45f)`.

Sampling area around a cell. The simplified palette also enforces a minimum wider sample.

ImageColorTolerance

Type: `float`. **Source default:** `0.043f`.

Inspector bounds: `Range(0f, 1f)`.

Tolerance used to match image colors against palette or fallback reference colors.

EnableImageRegionSegmentation

Type: `bool`. **Source default:** `true`.

Ink/boundary region voting. Simplify Image Layers with a palette bypasses this pipeline.

ImageRegionInkValueThreshold

Type: `float`. **Source default:** `0.335f`.

Inspector bounds: `Range(0f, 1f)`.

Maximum pixel value classified as dark region ink.

ImageRegionInkSaturationThreshold

Type: `float`. **Source default:** `0.434f`.

Inspector bounds: `Range(0f, 1f)`.

Saturation threshold for region ink recognition.

ImageRegionBoundaryColor

Type: `Color`. **Source default:** `new(0.475f, 0.478f, 0.467f)`.

Reference boundary ink color.

ImageRegionBoundaryColorTolerance

Type: `float`. **Source default:** `0.22f`.

Inspector bounds: `Range(0f, 1f)`.

Tolerance for the boundary color mask.

ImageRegionBoundaryExpansion

Type: `int`. **Source default:** `3`.

Inspector bounds: `Range(0, 4)`.

Pixel dilation of the detected region boundary mask.

ImageMinimumRegionPixelCount

Type: `int`. **Source default:** `51`.

Inspector bounds: `Range(1, 4096)`.

Smallest connected pixel region considered for a cell-region override.

ImageMinimumRegionCellVoteRatio

Type: `float`. **Source default:** `0.811f`.

Inspector bounds: `Range(0.2f, 1f)`.

Minimum dominant-region votes required to override local sampling.

ImageLargeWaterRegionOverrideMaxRatio

Type: `float`. **Source default:** `0.836f`.

Inspector bounds: `Range(0f, 1f)`.

Limit for large-water region overrides relative to the full image.

ImageLargeWaterRegionLocalKeepRatio

Type: `float`. **Source default:** `0.19f`.

Inspector bounds: `Range(0f, 1f)`.

Local water evidence needed to keep a large-water region decision.

EnableImageSurfaceSmoothing

Type: `bool`. **Source default:** `true`.

Cell-type cleanup after classification, independent of mesh contour smoothing.

ImageSurfaceSmoothingIterations

Type: `int`. **Source default:** `1`.

Inspector bounds: `Range(0, 3)`.

Number of classification cleanup passes.

EnableImageGroundRelief

Type: `bool`. **Source default:** `true`.

Adds a subtle, deterministic low-frequency elevation relief to imported ground cells.

ImageGroundReliefStrength

Type: `float`. **Source default:** `0.14f`.

Inspector bounds: `Range(0f, 0.25f)`.

Height variation of the subtle seeded relief added to imported ground when Image Ground Relief is enabled.

ImageGroundReliefScale

Type: `float`. **Source default:** `0.055f`.

Inspector bounds: `Range(0.015f, 0.2f)`.

Spatial frequency of seeded imported-ground relief. Lower values produce broader elevation variations.

ImageWaterCleanupNeighborCount

Type: `int`. **Source default:** `6`.

Inspector bounds: `Range(4, 8)`.

Water neighbors required by cell cleanup.

ImageWaterIslandCleanupMaxArea

Type: `int`. **Source default:** `7`.

Inspector bounds: `Range(0, 512)`.

Largest small land patch considered for water-island cleanup.

ImageWaterIslandCleanupMinWaterBorderRatio

Type: `float`. **Source default:** `0.58f`.

Inspector bounds: `Range(0f, 1f)`.

Required surrounding-water ratio before removing a small patch.

ImageMinimumSurfaceVoteRatio

Type: `float`. **Source default:** `0.16f`.

Inspector bounds: `Range(0.04f, 0.5f)`.

Votes needed to recognize a sampled non-mountain surface.

ImageMinimumMountainVoteRatio

Type: `float`. **Source default:** `0.4f`.

Inspector bounds: `Range(0.04f, 0.7f)`.

Votes needed to recognize a mountain patch.

ImportImageResources

Type: `bool`. **Source default:** `false`.

Imports painted gold/oil markers from the source image. Disabled by default so strategic resources can be placed manually.

ImageOilFieldVoteRatio

Type: `float`. **Source default:** `0.45f`.

Inspector bounds: `Range(0.04f, 0.8f)`.

Votes needed to import an oil marker.

ImageGoldMineVoteRatio

Type: `float`. **Source default:** `0.18f`.

Inspector bounds: `Range(0.04f, 0.8f)`.

Votes needed to import a gold marker.

ImageInkValueThreshold

Type: `float`. **Source default:** `0.24f`.

Inspector bounds: `Range(0f, 1f)`.

Threshold excluding dark local ink from terrain-color votes.

ImagePaperSaturationThreshold

Type: `float`. **Source default:** `0.10f`.

Inspector bounds: `Range(0f, 1f)`.

Low saturation threshold for detecting empty paper.

ImagePaperValueThreshold

Type: `float`. **Source default:** `0.611f`.

Inspector bounds: `Range(0f, 1f)`.

High value threshold for detecting empty paper.

ImportedMountainHeight

Type: `float`. **Source default:** `5f`.

Inspector bounds: `Range(TerrainHeightSettings.MaxGroundHeight, TerrainHeightSettings.MaxMountainHeight)`.

Peak height for mountains painted in an imported image. Procedural mountain settings do not change their authored footprint.

ImageGrassColor

Type: `Color`. **Source default:** `new(0.78f, 0.92f, 0.86f)`.

Fallback reference for grass recognition without a palette.

ImageDirtForestColor

Type: `Color`. **Source default:** `new(0.55f, 0.75f, 0.72f)`.

Fallback reference for forest recognition without a palette.

ImageBeachColor

Type: `Color`. **Source default:** `new(1f, 0.94f, 0.72f)`.

Fallback beach reference; imports as grass unless Beach As Dirt Path is enabled.

ImageWaterColor

Type: `Color`. **Source default:** `new(0.72f, 0.90f, 0.93f)`.

Fallback reference for water recognition without a palette.

ImageDirtPathColor

Type: `Color`. **Source default:** `new(0.90f, 0.86f, 0.84f)`.

Fallback reference for painted dirt paths without a palette.

ImportImageBeachAsDirtPath

Type: `bool`. **Source default:** `false`.

Classifies fallback beach paint as DirtPath instead of grass.

ImageMountainColor

Type: `Color`. **Source default:** `new(0.86f, 0.86f, 0.86f)`.

Fallback reference for mountain recognition without a palette.

ImageOilFieldColor

Type: `Color`. **Source default:** `new(0.48f, 0.48f, 0.48f)`.

Fallback oil-marker reference without a palette.

ImageGoldMineColor

Type: `Color`. **Source default:** `new(0.90f, 0.85f, 0.27f)`.

Fallback gold-marker reference without a palette.

Strategic Layout

AdditionalMainBases

Type: `int`. **Source default:** `0`.

Inspector bounds: `Range(0, 5)`.

Extra main-base anchors in addition to Player Count.

ExpansionsPerPlayer

Type: `int`. **Source default:** `1`.

Inspector bounds: `Range(0, 12)`.

Requested owned expansions per main base.

ContestedExpansionCount

Type: `int`. **Source default:** `2`.

Inspector bounds: `Range(0, 10)`.

Neutral objectives scattered by seed across free terrain. The complete set balances estimated access from player bases and favors underused sectors; individual objectives need not lie on player medians. Crowded maps may fit fewer anchors.

GoldMinesPerPlayer

Type: `int`. **Source default:** `1`.

Inspector bounds: `Range(1, 10)`.

Starting gold mines per main base. Expansions also request their own mine.

OilFieldsPerPlayer

Type: `int`. **Source default:** `1`.

Inspector bounds: `Range(0, 10)`.

Requested starting offshore oil fields; not generated on resolved River layouts.

NeutralOilFieldCount

Type: `int`. **Source default:** `2`.

Inspector bounds: `Range(0, 10)`.

Requested additional neutral offshore oil fields, outside River layouts.

ChokePoints

Type: `int`. **Source default:** `0`.

Inspector bounds: `Range(0, 6)`.

Strategic bottlenecks between player territories (0 disables them). Multiple passages are spaced across broad approaches and share land roads. Small pockets and crowded optional openings are skipped; existing settlement access can require extra passages. Continuous forest/rock dividers funnel land routes; Islands creates variable-width naval straits between irregular low or wooded islands, reusing nearby neutral expansion islands; mountain cores remain optional.

NavalZones

Type: `int`. **Source default:** `0`.

Inspector bounds: `Range(0, 4)`.

Additional water features created after the primary Water Layout. Zero still generates the selected layout's main water feature.

Terrain Features

HasWater

Type: `bool`. **Source default:** `true`.

Procedural water switch. Image import automatically enables it when water is classified.

HasBridges

Type: `bool`. **Source default:** `true`.

Safe bridge planning perpendicular to the local course on resolved River layouts, including oblique path crossings. Dry approaches reconnect paths to shifted landings; the river remains underneath.

HasCliffs

Type: `bool`. **Source default:** `false`.

Generates cliff details for steep terrain breaks. Plateau walls remain decorated automatically, while low buildable shore accesses stay sloped for ports.

HasPlateaus

Type: `bool`. **Source default:** `false`.

Builds strategic plateaus and their access ramps. With HasWater and WaterLayout = Lake, contested expansions retain ordinary ground relief to form low islands.

GenerateNaturalExpansionPlateaus

Type: `bool`. **Source default:** `true`.

When enabled, natural expansions receive their own lower plateau and a shared transition from the main base. Disable it to keep naturals on regular ground.

HasForestZones

Type: `bool`. **Source default:** `true`.

Procedural forest surface patches. Does not erase forests painted in an image.

HasMountainZones

Type: `bool`. **Source default:** `true`.

Procedural mountain placement, or recognition/shaping of painted image mountains.

EnablePropsGeneration

Type: `bool`. **Source default:** `true`.

Instantiate the resolved decoration and grass prefab rules after terrain generation.

Grass Details

EnableGrassDetails

Type: `bool`. **Source default:** `true`.

Enables grass prefab rules in MapPropsGenerator. Uses the included prefabs when no Is Grass Detail rule is configured. Requires Enable Props Generation; no grass mesh is generated.

GrassDetailDensity

Type: `float`. **Source default:** `1f`.

Inspector bounds: `Range(0f, 4f)`.

Multiplies each grass rule's placement chance, capped at 100%. One patch per free planting area; 1 preserves the biome percentages.

GrassDetailHeight

Type: `Vector2`. **Source default:** `new(.5f, .8f)`.

Minimum and maximum target blade height in cell units (0.05-2). Changes vertical scale only; authored ground coverage and shared meshes are preserved. Medium covers one cell and Large four cells per side.

GrassDetailMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Optional grass-card material override. Otherwise uses the biome material. Included Editor defaults are serialized for scene-based runtime generation.

GrassPrefabs

Type: `GrassPrefabLibrary`. **Source default:** `Unassigned / type default`.

Shared grass prefabs used when no grass rule is configured. The Editor can fill the included library; runtime-created definitions should assign this reference or provide explicit grass rules.

PropConfigs

Type: `InstConfig[]`. **Source default:** `Unassigned / type default`.

Optional per-map prop placement override. When empty, uses the active BiomeSetting. If grass is enabled and no Is Grass Detail rule exists, included grass rules are added in memory.

PlateauReliefScale

Type: `float`. **Source default:** `1f`.

Inspector bounds: `Range(0.4f, 1.4f)`.

Vertical relief multiplier for strategic plateau tiers.

PlateauFootprintScale

Type: `float`. **Source default:** `1f`.

Inspector bounds: `Range(0.75f, 2f)`.

Scales the buildable footprint around main bases and expansions. Values above 1 create broader plateau tops; available map space and nearby strategic zones can still limit the final size.

PlayerPlateauBorderSafetyDistance

Type: `int`. **Source default:** `0`.

Inspector bounds: `Range(0, 20)`.

Minimum distance in grid cells between a player plateau's outer cliff border and the map boundary. Zero allows the plateau to touch the boundary.

PlateauContourIrregularity

Type: `float`. **Source default:** `0.13f`.

Inspector bounds: `Range(0.04f, 0.22f)`.

Low-frequency contour variation. Higher values create stronger lobes and recesses without cell-by-cell noise.

MainBaseRampCount

Type: `int`. **Source default:** `1`.

Inspector bounds: `Range(1, 4)`.

Number of distinct accesses distributed around each main plateau (1-4). A shared main-to-natural ramp counts as one access. Unsafe directions are replaced with nearby valid directions.

PlateauRampWidth

Type: `int`. **Source default:** `4`.

Inspector bounds: `Range(3, 5)`.

Width of plateau access ramps in cell units. Straight accesses render as a single inclined surface, including diagonal ramps.

PlateauRampLength

Type: `float`. **Source default:** `5f`.

Inspector bounds: `Range(2f,18f)`.

Horizontal length of an added plateau ramp in cell units. Longer ramps descend more gently; shorter ramps are steeper. The plateau top is retained.

PlateauEdgeThickness

Type: `int`. **Source default:** `1`.

Inspector bounds: `Range(1, 2)`.

Thickness of the non-walkable cliff border around plateau tops.

NaturalExpansionSecondAccess

Type: `bool`. **Source default:** `true`.

Creates two distinct exterior exits from each natural plateau instead of one, while retaining the access toward its main base. Requires enough safe space for separate ramps.

RampDirtPathApproachLength

Type: `int`. **Source default:** `3`.

Inspector bounds: `Range(2, 8)`.

Length in cells of the worn DirtPath area generated before and after every ramp.

RampDirtPathApproachWidth

Type: `float`. **Source default:** `1f`.

Inspector bounds: `Range(0.75f, 2f)`.

Width multiplier for the worn DirtPath approaches at both ends of a ramp, relative to its actual opening. Use a value near 1 to keep paths close to the ramp width.

ForestDensityMultiplier

Type: `float`. **Source default:** `50f`.

Inspector bounds: `Range(0f, 100f)`.

Target forest coverage as a percentage, including values between zero and five. Strategic clearings can reduce the actual coverage.

ForestZoneCountMultiplier

Type: `float`. **Source default:** `7f`.

Inspector bounds: `Range(1f, 10f)`.

Controls forest fragmentation independently of target density: 1 creates a few broad woods; 10 creates many smaller groves. Distribution is seeded and scales with map size; strategic clearings remain protected.

PlateauForestDensityMultiplier

Type: `float`. **Source default:** `0.45f`.

Inspector bounds: `Range(0f, 1f)`.

Controls forest pockets on safe plateau tops as a proportion of the global forest coverage. Zero disables plateau forests; build pads, ramps and strategic paths remain clear.

PlayerBaseForestClearRadius

Type: `int`. **Source default:** `11`.

Inspector bounds: `Range(0, 15)`.

Radius of open ground around each main-base center, kept clear of forests and procedural rivers, even when forests are disabled. Existing base and plateau footprints also remain dry.

ExpansionForestClearRadius

Type: `int`. **Source default:** `4`.

Inspector bounds: `Range(0, 15)`.

Radius of open ground around each owned or contested expansion center, kept clear of forests and procedural rivers. Existing expansion and plateau footprints also remain dry.

MountainZoneCount

Type: `int`. **Source default:** `3`.

Inspector bounds: `Range(1, 8)`.

Base number of independent massifs for an 80x80 map at density 1. The requested total scales with land area and Mountain Density Multiplier.

MountainDensityMultiplier

Type: `float`. **Source default:** `3f`.

Inspector bounds: `Range(0.25f, 4f)`.

Amount of procedural mountain formations. The default 3 distributes massifs across the map; lower values leave broader plains. Routes, resources and plateau accesses remain protected.

MountainFalloff

Type: `MountainFalloffType`. **Source default:** `MountainFalloffType.Random`.

Controls how quickly each ridge rises from its foothills.

MountainMaxHeight

Type: `float`. **Source default:** `5.5f`.

Inspector bounds: `Range(2.5f, 6f)`.

Maximum procedural mountain elevation before continuous Map Altitude. Painted mountains use Imported Mountain Height instead.

MountainChainWidth

Type: `float`. **Source default:** `1.25f`.

Inspector bounds: `Range(0.5f, 2.5f)`.

Width multiplier for massifs and ridges. Their two sides vary independently for an asymmetric silhouette.

MountainMaxArea

Type: `int`. **Source default:** `180`.

Inspector bounds: `Range(24, 600)`.

Maximum footprint, including low foothills, for each mountain formation.

MountainRiseSharpness

Type: `float`. **Source default:** `1.15f`.

Inspector bounds: `Range(0.4f, 3f)`.

Higher values sharpen the rocky crest while keeping a low foothill transition.

MountainPathClearance

Type: `float`. **Source default:** `3f`.

Inspector bounds: `Range(1f, 6f)`.

Minimum open buffer reserved around strategic routes, choke corridors, ramps and imported DirtPaths.

EnableMountainCliffDetails

Type: `bool`. **Source default:** `true`.

Embeds seeded cliff faces along mountain ridges and slopes, with slight lean and joined groups. Independent of Has Cliffs. Full tilted renderer/collider bounds remain inside safe mountain footprints.

MountainCliffDetailScale

Type: `float`. **Source default:** `1f`.

Inspector bounds: `Range(0.5f, 2f)`.

Uniform size multiplier for mountain rocks. Authored proportions are preserved. Large rocks are tried first, with smaller pieces filling safe gaps; neighboring terrain and passages stay clear.

MountainCliffMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Optional material for embedded mountain cliff faces. Empty uses the mountain terrain material, then keeps prefab materials if no terrain material is available.

Shared Scene Settings

CellSize

Type: `float`. **Source default:** `1f`.

Inspector bounds: `Min(0.1f)`.

World-space width and depth of one logical grid cell. Terrain and placed prefab footprints follow this value.

EnableMeshOptimization

Type: `bool`. **Source default:** `true`.

Shares compatible vertices and edges, groups surfaces with the same material and Unity layer, and uses 16-bit indices when possible. Custom UV/normal seams are preserved.

EnableVertexPositionNoise

Type: `bool`. **Source default:** `true`.

Small vertex displacement; planar jitter yields to shared contour smoothing.

EnableProceduralContourSmoothing

Type: `bool`. **Source default:** `true`.

Rounds procedural forest, plateau, mountain and shoreline contours, including material junctions, with shared mesh vertices. Preserves gameplay cells, resources, ramps and reserved passages. Uses contour displacement instead of random planar vertex jitter.

ProceduralContourSmoothingStrength

Type: `float`. **Source default:** `0.85f`.

Inspector bounds: `Range(0f, 1f)`.

Amount of shared contour rounding for procedural terrain when Procedural Contour Smoothing is enabled.

EnablePathSmoothing

Type: `bool`. **Source default:** `true`.

Rounds road contours with finer geometry. Curved routes, town connections and local dead-end branches are generated in both modes.

PathConnectionCount

Type: `int`. **Source default:** `1`.

Inspector bounds: `Range(1, 6)`.

Overall road connectivity. One gives settlements essential access to the road network; higher values add alternative links between settlements.

RandomLongPathCount

Type: `int`. **Source default:** `0`.

Inspector bounds: `Range(0, 12)`.

Optional long roads crossing the map. Keep at zero for a sparse natural network.

PathBranchCount

Type: `int`. **Source default:** `0`.

Inspector bounds: `Range(0, 6)`.

Maximum number of extra short decorative branches, beyond local town streets. The generated count is randomized.

WanderingPathCount

Type: `int`. **Source default:** `4`.

Inspector bounds: `Range(0, 12)`.

Number of additional meandering trails, independent of choke points and strategic roads. They can open narrow passages through forest. Zero disables them.

Optional Outputs

EnableMinimapGeneration

Type: `bool`. **Source default:** `false`.

Generates a separate water rendering copy for minimap cameras. Disabled by default; normal water rendering and collisions are unaffected.

MinimapLayerName

Type: `string`. **Source default:** `"BackgroundMinimap"`.

Unity layer for the optional minimap water copy. Missing or empty names fall back to the configured water layer. Does not create a Unity layer.

EnableAerialGeneration

Type: `bool`. **Source default:** `false`.

Generates a hidden flight-surface mesh and collider above ground/water. Disabled by default; mountains may protrude through it.

AerialLayerName

Type: `string`. **Source default:** `"Aerial"`.

Unity layer for the optional flight collider. Missing or empty names fall back to Ignore Raycast. Does not create a Unity layer.

AerialClearance

Type: `float`. **Source default:** `3f`.

Inspector bounds: `Min(0.1f)`.

Minimum flight-surface clearance above ordinary ground and water, in world units. Mountains intersect this surface instead of raising it.

Project Integration

GroundLayerName

Type: `string`. **Source default:** `"Ground"`.

Unity layer for generated ground surfaces. Missing or empty names fall back to Default. Explicit image-palette layers keep priority.

EnvironmentLayerName

Type: `string`. **Source default:** `"Environment"`.

Unity layer for mountain rock, seabed, cliffs and altitude side walls. Missing or empty names fall back to Default.

WaterLayerName

Type: `string`. **Source default:** `"Water"`.

Unity layer for the normal water surface and its collider. Missing or empty names fall back to Default.

BridgeLayerName

Type: `string`. **Source default:** `"Walkable"`.

Unity layer applied to bridges and their children. Missing or empty names fall back to the configured ground layer, then Default.

AdditionalPropPlacementLayers

Type: `LayerMask`. **Source default:** `Unassigned / type default`.

Extra collider layers considered during prop placement, in addition to Default, TransparentFX and the configured ground, environment and bridge layers. Prop instance layers remain controlled by biome rules.

EnableNavigationGraphUpdates

Type: `bool`. **Source default:** `false`.

Opt-in compatibility with an installed Pathfinding.GraphUpdateScene component. No navigation package is required. Disabled by default; uses the configurable water and aerial tags below.

WaterNavigationTag

Type: `int`. **Source default:** `1`.

Inspector bounds: `Range(0, 31)`.

A* graph node tag for water (not a Unity GameObject tag or layer). Used only with Enable Navigation Graph Updates.

AerialNavigationTag

Type: `int`. **Source default:** `2`.

Inspector bounds: `Range(0, 31)`.

A* graph node tag for the optional flight surface (not a Unity GameObject tag or layer).

Terrain Materials

GrassMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Optional terrain material override for grass surfaces. Leave empty to use the selected biome's default.

PlateauMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Optional top material for strategic plateaus. Falls back to Grass Material.

DirtPathMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Optional material override for dirt paths and ramp approaches. Leave empty to use the selected biome's default.

DirtForestMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Optional material override for forest-floor terrain. Leave empty to use the selected biome's default.

UnderwaterMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Optional material override for seabed terrain. Leave empty to use the selected biome's default.

EnvMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Optional environment terrain material override. Leave empty to use the selected biome's default.

MountainMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Optional mountain terrain material override. Also supplies cliff materials when no dedicated override is assigned.

ImageLayerMaterials

Type: `Material[]`. **Source default:** `Unassigned / type default`.

Optional material channels for image-import palettes. Explicit palette-layer materials take priority; empty channels can fall back to the biome.

Water Settings

RiverBridgePrefab

Type: `GameObject`. **Source default:** `Unassigned / type default`.

Optional bridge prefab override for this map. Leave empty to use BiomeSetting.RiverBridgePrefab. Requires a non-trigger deck collider. Crossings follow the local river normal.

EditorBridgePrefabSearchPath

Type: `string`. **Source default:** `"Assets/SorolifMapGenerationStarter/Prefabs/RiverBridge.prefab"`.

Optional project-relative prefab path searched in the Editor when both the map override and biome bridge are empty. Empty disables lookup. Assign the resulting prefab reference before building a player.

RiverBridgeWidth

Type: `int`. **Source default:** `3`.

Inspector bounds: `Range(2, 6)`.

Target width of generated river-bridge decks in grid cells. The source prefab is fitted to the planned crossing.

RiverBridgeMaxSpan

Type: `int`. **Source default:** `64`.

Inspector bounds: `Range(8, 100)`.

Maximum distance between bridge landings, in grid cells. Bridges follow the local river normal, with a small tolerance for grid-aligned landing cells. Clear banks may have a gentle height difference.

WaterMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Optional animated water-surface material override. Leave empty to use the selected biome's default.

WaterMinimapMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Material for the optional water rendering copy; inactive when minimap generation is disabled.

WaterHeight

Type: `float`. **Source default:** `-0.5f`.

Fixed water surface at -0.5 before Map Altitude. Normalized during water generation.

WaterFloorHeight

Type: `float`. **Source default:** `TerrainHeightSettings.WaterFloorHeight`.

Reference deep seabed height; the deepest configured depth can lower it further.

EnableWaterFloorNoise

Type: `bool`. **Source default:** `true`.

Add seeded height variation to the seabed without changing the water-surface level.

WaterFloorNoiseStrength

Type: `float`. **Source default:** `0.85f`.

Inspector bounds: `Range(0f, 1f)`.

Magnitude of seabed height variation when Water Floor Noise is enabled.

WaterFloorNoiseScale

Type: `float`. **Source default:** `0.065f`.

Inspector bounds: `Range(0.02f, 0.18f)`.

Spatial frequency of seeded seabed noise. Lower values produce broader underwater features.

WaterFloorNoiseSmoothingIterations

Type: `int`. **Source default:** `1`.

Inspector bounds: `Range(0, 3)`.

Number of smoothing passes applied to the noisy seabed height field.

WaterVisualBorder

Type: `int`. **Source default:** `1`.

Inspector bounds: `Range(0, 2)`.

Extra visual water coverage around the map boundary, measured in grid cells.

RiverWidth

Type: `int`. **Source default:** `4`.

Inspector bounds: `Range(4, 20)`.

Target width of procedural river courses in grid cells.

CurveStrength

Type: `int`. **Source default:** `4`.

Inspector bounds: `Range(2, 8)`.

Amount of lateral curvature added to procedural river courses.

ControlPointCount

Type: `int`. **Source default:** `6`.

Inspector bounds: `Range(4, 20)`.

Number of control points used to shape procedural river courses.

IslandRadiusMin

Type: `int`. **Source default:** `13`.

Inspector bounds: `Range(4, 30)`.

Minimum target radius of a player island in grid cells. The organic contour can vary around this target.

IslandRadiusMax

Type: `int`. **Source default:** `15`.

Inspector bounds: `Range(4, 30)`.

Maximum target radius of a player island in grid cells. Placement also adapts to map size and player spacing.

IslandContourIrregularity

Type: `float`. **Source default:** `0.72f`.

Inspector bounds: `Range(0f, 1f)`.

Controls broad coastline deformation for procedural islands: coves, asymmetric lobes and tapered peninsulas. Strategic footprints remain protected.

IslandMinimumWaterGap

Type: `int`. **Source default:** `6`.

Inspector bounds: `Range(2, 16)`.

Minimum water gap, in grid cells, between procedural islands, including contested islands in Lake mode. Beach shelves preserve this gap. Six gives ships room between separate islands.

NaturalIslandRadiusMin

Type: `int`. **Source default:** `10`.

Inspector bounds: `Range(3, 30)`.

Minimum target radius of a natural-expansion island in grid cells.

NaturalIslandRadiusMax

Type: `int`. **Source default:** `13`.

Inspector bounds: `Range(3, 30)`.

Maximum target radius of a natural-expansion island in grid cells.

IsolatedExpansionIslandRadiusMin

Type: `int`. **Source default:** `7`.

Inspector bounds: `Range(3, 30)`.

Minimum target radius of an isolated player-expansion island in grid cells.

IsolatedExpansionIslandRadiusMax

Type: `int`. **Source default:** `11`.

Inspector bounds: `Range(3, 30)`.

Maximum target radius of an isolated player-expansion island in grid cells.

ContestedIslandRadiusMin

Type: `int`. **Source default:** `4`.

Inspector bounds: `Range(3, 30)`.

Minimum target radius of a contested-objective island in grid cells.

ContestedIslandRadiusMax

Type: `int`. **Source default:** `6`.

Inspector bounds: `Range(3, 30)`.

Maximum target radius of a contested-objective island in grid cells.

BonusIslandRadiusMin

Type: `int`. **Source default:** `5`.

Inspector bounds: `Range(3, 30)`.

Minimum target radius of an additional unassigned island in grid cells.

BonusIslandRadiusMax

Type: `int`. **Source default:** `9`.

Inspector bounds: `Range(3, 30)`.

Maximum target radius of an additional unassigned island in grid cells.

LakeRadiusMin

Type: `int`. **Source default:** `18`.

Inspector bounds: `Min(1)`.

Minimum radius of the central lake in cells. The requested min/max range is honored within the map footprint; bases and ramp landings are protected locally. Additional lakes use half this range when enough room remains.

LakeRadiusMax

Type: `int`. **Source default:** `50`.

Inspector bounds: `Min(1)`.

Maximum radius of the central lake in cells. Set equal to Lake Radius Min for a fixed radius. Very large values are limited by the map footprint, without a hidden visual minimum.

MinWaterDepth

Type: `float`. **Source default:** `-1f`.

Shallow seabed target; safety keeps it below the fixed water surface.

MaxWaterDepth

Type: `float`. **Source default:** `-3f`.

Deep seabed target, combined with Water Floor Height by choosing the deepest value.

BeachHeightInner

Type: `float`. **Source default:** `0.02f`.

Height of the inner beach transition, subject to dry landing safety.

BeachHeightOuter

Type: `float`. **Source default:** `0.16f`.

Height of the outer beach transition and buildable shore-access checks.

Cliff Settings

CliffPrefabs

Type: `GameObject[]`. **Source default:** `Unassigned / type default`.

Optional wall/plateau cliff overrides. Empty uses the biome's Cliff Prefabs.

MountainCliffPrefabs

Type: `GameObject[]`. **Source default:** `Unassigned / type default`.

Optional mountain crest overrides. Empty uses the biome's Mountain Cliff Prefabs; older biomes without this list retain their legacy Cliff Prefabs fallback.

CliffParent

Type: `Transform`. **Source default:** `Unassigned / type default`.

Generated automatically under Env/Cliffs by MapGenerator. Used as an explicit output parent only when calling CliffGenerator directly.

CliffHeightThreshold

Type: `float`. **Source default:** `1.5f`.

Minimum height break covered by cliff details. Values above the terrain split threshold still cover every generated vertical face.

CliffEdgeOffset

Type: `float`. **Source default:** `0.1f`.

Inspector bounds: `Min(0f)`.

Offsets the center of cliff details from the steep terrain face toward the low ground.

CliffVerticalOffset

Type: `float`. **Source default:** `0.12f`.

Inspector bounds: `Min(0f)`.

Raises cliff details relative to the center of the abrupt face, increasing their visible overhang above the high ground.

CliffHeightScale

Type: `float`. **Source default:** `MinimumCliffHeightScale`.

Inspector bounds: `Range(MinimumCliffHeightScale, 1.5f)`.

Extra vertical coverage. Legacy values below 1.15 are upgraded automatically so cliff details extend beneath the complete abrupt face.

CliffSegmentOverlap

Type: `float`. **Source default:** `0.12f`.

Inspector bounds: `Min(0f)`.

Extra overlap between cliff details in world units. When positive, an allowance of 1.2 cells or 20% of the piece length also closes irregular rock caps and centered diagonal joins, including free ends. Inclined pieces add coverage for their pitched end caps. Zero disables overlap. Passage clipping still keeps openings clear.

CliffTilesPerPrefab

Type: `int`. **Source default:** `12`.

Inspector bounds: `Range(1, 16)`.

Preferred cliff length in terrain cell units. Complete compatible contours are divided into balanced pieces, with up to 15% length tolerance to avoid small leftovers. Diagonal grid steps can share one oblique prefab; curvature, height and passage safeguards still split unsafe spans.

CliffDepthScale

Type: `float`. **Source default:** `0.72f`.

Inspector bounds: `Range(0.35f, 1.5f)`.

Scales prefab depth across the terrain break without changing its wall length. Lower values reduce visual overlap. Oblique contours retain enough depth to bury the complete stepped face inside the rock.

RampCliffDepthScale

Type: `float`. **Source default:** `1.15f`.

Inspector bounds: `Range(0.5f, 2f)`.

Width multiplier for low ramp borders. The default width is about one third of a terrain cell. Borders are embedded in the ramp with only a small lip above its surface.

RampCliffInset

Type: `float`. **Source default:** `DefaultRampCliffInset`.

Inspector bounds: `Range(0f, 1.5f)`.

World-space overlap of each ramp rail beneath the ramp edge. The 0.32 default keeps the prefab tight against the complete descent without materially narrowing its traversable center.

CliffVerticalOverlap

Type: `float`. **Source default:** `0.15f`.

Inspector bounds: `Min(0f)`.

Extra vertical overlap used to hide the terrain above and below a cliff prefab.

CliffRunHeightTolerance

Type: `float`. **Source default:** `0.18f`.

Inspector bounds: `Range(0.01f, 1f)`.

Maximum high/low height deviation from a continuous slope merged into one cliff run. A steady rise can share an inclined prefab; changes of slope or wall thickness still split the run.

CliffPrefabLengthAxis

Type: `CliffPrefabLengthAxis`. **Source default:** `CliffPrefabLengthAxis.Auto`.

Local prefab axis treated as its horizontal wall length. Auto selects the largest horizontal extent.

MarkCliffCellsUnbuildable

Type: `bool`. **Source default:** `true`.

Marks cells touching generated cliff walls as non-buildable.

CliffRotationOffsetY

Type: `float`. **Source default:** `0f`.

Additional yaw in degrees used to align the authored cliff prefab with the fitted terrain boundary.

Resource Settings

GoldMinePrefab

Type: `GameObject`. **Source default:** `Unassigned / type default`.

Optional gold-resource prefab override. Leave empty to use the selected biome's resource prefab.

OilFieldPrefab

Type: `GameObject`. **Source default:** `Unassigned / type default`.

Optional oil-resource prefab override. Leave empty to use the selected biome's resource prefab.

ResourceParent

Type: `Transform`. **Source default:** `Unassigned / type default`.

Generated automatically under Players/Nature by MapGenerator. Used as an explicit output parent only when calling ResourcePlacementSystem directly.

GoldMineMinDistanceFromAnchor

Type: `int`. **Source default:** `3`.

Inspector bounds: `Range(2, 12)`.

Minimum candidate distance in grid cells between a gold mine and its associated settlement anchor.

GoldMineMaxDistanceFromAnchor

Type: `int`. **Source default:** `6`.

Inspector bounds: `Range(3, 16)`.

Maximum candidate distance in grid cells between a gold mine and its associated settlement anchor.

OilFieldMinDistanceFromAnchor

Type: `int`. **Source default:** `5`.

Inspector bounds: `Range(2, 20)`.

Minimum candidate distance in grid cells between an offshore oil field and its associated anchor.

OilFieldMaxDistanceFromAnchor

Type: `int`. **Source default:** `10`.

Inspector bounds: `Range(3, 24)`.

Maximum candidate distance in grid cells between an offshore oil field and its associated anchor.

Gameplay Constraints

MinimumRushDistance

Type: `float`. **Source default:** `20f`.

Inspector bounds: `Range(10f, 120f)`.

Requested minimum base spacing and the rush-distance score floor; dense maps may need relaxed placement.

ExpansionFairnessTolerance

Type: `float`. **Source default:** `0.15f`.

Inspector bounds: `Range(0f, 1f)`.

Relative expansion-access distance spread accepted by the gameplay score.

ResourceFairnessTolerance

Type: `float`. **Source default:** `0.10f`.

Inspector bounds: `Range(0f, 1f)`.

Allowed relative spread in starting-resource access distances between players. Affects the resource component of the gameplay score; does not move resources or reject the map.

Generation

Seed

Type: `int`. **Source default:** `0`.

Seed used by deterministic generation when Use Random Seed is disabled.

UseRandomSeed

Type: `bool`. **Source default:** `true`.

Choose a new seed for each generation. Disable to reproduce a map while comparing settings.

Screenshot Settings

GenerateScreenshotAfterGeneration

Type: `bool`. **Source default:** `false`.

Captures a top-down PNG after a successful map generation.

ScreenshotOutputFolder

Type: `string`. **Source default:** `"MapScreenshots"`.

Absolute folder or relative folder. Relative paths use the Unity project root in the Editor and Application.persistentDataPath in a build.

ScreenshotWidth

Type: `int`. **Source default:** `1024`.

Inspector bounds: `Range(256, 4096)`.

Exported PNG width, independently of map resolution.

ScreenshotHeight

Type: `int`. **Source default:** `1024`.

Inspector bounds: `Range(256, 4096)`.

Exported PNG height, independently of map resolution.

ScreenshotPadding

Type: `float`. **Source default:** `0.05f`.

Inspector bounds: `Range(0f, 0.5f)`.

Extra space around the rendered map as a ratio of its visible bounds.

ScreenshotViewMode

Type: `MapScreenshotViewMode`. **Source default:** `MapScreenshotViewMode.TopDown`.

Top Down preserves a map-like orthographic view. Isometric reveals terrain height while keeping an orthographic projection.

ScreenshotIsometricPitch

Type: `float`. **Source default:** `62f`.

Inspector bounds: `Range(35f, 80f)`.

Downward camera pitch used by Isometric screenshot mode.

ScreenshotIsometricYaw

Type: `float`. **Source default:** `45f`.

Inspector bounds: `Range(-180f, 180f)`.

Horizontal rotation used by Isometric screenshot mode.

ScreenshotBackgroundColor

Type: `Color`. **Source default:** `new Color(0.08f, 0.1f, 0.12f, 1f)`.

Camera clear color used for the exported screenshot background.

ScreenshotTransparentBackground

Type: `bool`. **Source default:** `false`.

Export screenshot background alpha instead of an opaque background where the selected capture settings support it.

ScreenshotLayerMask

Type: `LayerMask`. **Source default:** `~0`.

Unity layers included by the capture camera; affects the export, not generation.

ScreenshotOverwriteExisting

Type: `bool`. **Source default:** `true`.

When disabled, a timestamp is appended if the seed image already exists.

BiomeSetting

Configuration

BiomeType

Type: `BiomeType`. **Source default:** `Unassigned / type default`.

Biome identifier supported by this preset. The generator's preset selector filters assets by this value.

ShowInBiomeSelector

Type: `bool`. **Source default:** `true`.

Offer this preset and its biome in the generator's Editor selectors. Disable unfinished or optional example presets. Existing explicit references and runtime generation remain usable.

BorderPrefab

Type: `GameObject`. **Source default:** `Unassigned / type default`.

Optional prefab used to represent the generated map border.

BorderBoxMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Default material used for the map's enclosing border geometry.

Terrain Materials

GrassMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Default terrain material for grass surfaces. A material assigned directly on MapDefinition takes priority.

GrassDetailMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Material for decorative grass cards. Independent of the ground Grass Material; optional per-map override is available.

PlateauMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Default terrain material for plateau tops. A material assigned directly on MapDefinition takes priority.

DirtPathMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Default terrain material for dirt paths and ramp approaches. A map-level override takes priority.

DirtForestMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Default terrain material beneath forest decoration. A map-level override takes priority.

UnderwaterMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Default seabed terrain material beneath water surfaces. A map-level override takes priority.

WaterMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Default animated water-surface material for this biome. A map-level water material takes priority.

WaterMinimapMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Default material for optional minimap water output. It does not replace the main water material.

EnvMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Default environment terrain material used when the map does not assign its own override.

MountainMaterial

Type: `Material`. **Source default:** `Unassigned / type default`.

Default mountain terrain material, also used by fitted cliff details when no dedicated override is assigned.

ImageLayerMaterials

Type: `Material[]`. **Source default:** `Unassigned / type default`.

Materials addressed by Biome Material Id in an image palette. Explicit per-layer materials take priority.

RampLayerName

Type: `string`. **Source default:** `"Walkable"`.

Legacy serialized layer name retained for compatibility. Generated ramps now use the map's shared Walkable layer settings.

Compatibility/internal Inspector field: retained for serialized data; normally hidden from the Inspector.

Biome Prefabs

RiverBridgePrefab

Type: `GameObject`. **Source default:** `Unassigned / type default`.

Default bridge deck for this biome. Requires a non-trigger deck collider. MapDefinition.RiverBridgePrefab overrides this choice when assigned.

CliffPrefabs

Type: `GameObject[]`. **Source default:** `Unassigned / type default`.

Rectangular details fitted continuously over global cliff faces and the steep outer walls of plateaus.

MountainCliffPrefabs

Type: `GameObject[]`. **Source default:** `Unassigned / type default`.

Rock crests embedded in mountain ridges. Independent of the wall/plateau Cliff Prefabs. Empty keeps the legacy Cliff Prefabs fallback.

GoldMinePrefab

Type: `GameObject`. **Source default:** `Unassigned / type default`.

Default gold-resource prefab for this biome. MapDefinition.GoldMinePrefab overrides it when assigned.

OilFieldPrefab

Type: `GameObject`. **Source default:** `Unassigned / type default`.

Default oil-resource prefab for this biome. MapDefinition.OilFieldPrefab overrides it when assigned.

Decoration

configs

Type: `InstConfig[]`. **Source default:** `Unassigned / type default`.

Prop rules used by generated surface prefixes. Direct MapDefinition.PropConfigs take priority when non-empty.

prefab

Type: `GameObject`. **Source default:** `Unassigned / type default`.

Prefab instantiated on eligible generated surface patches.

color

Type: `Color`. **Source default:** `Unassigned / type default`.

Reference color for the standalone legacy Scan By Color action. Normal map generation selects surfaces by Prefix.

Prefix

Type: `string`. **Source default:** `Unassigned / type default`.

Generated surface prefix to populate, for example DirtForest_. Matching is case-sensitive.

tilePercent

Type: `float`. **Source default:** `Unassigned / type default`.

Placement probability from 0 to 100. For matching World Object rules, the highest value sets group coverage; all positive values also weight the model choice. Footprint checks can reject a placement.

Scale

Type: `float`. **Source default:** `Unassigned / type default`.

Patch size in tiles, not an object transform multiplier. Grass is placed largest first and reserves this area; zero/-1 infer the authored grass size. Other props retain their existing scan passes.

RandomRotation

Type: `bool`. **Source default:** `Unassigned / type default`.

Randomizes yaw using the generator seed.

IsWorldObject

Type: `bool`. **Source default:** `Unassigned / type default`.

Uses the world-object placement convention and the wider world-object footprint altitude tolerance.

Layer

Type: `int`. **Source default:** `Unassigned / type default`.

Unity layer of the instantiated root. Missing or invalid layer indices fall back to the supported 0-31 range.

HasMaxAltitude

Type: `bool`. **Source default:** `Unassigned / type default`.

Rejects patches above Max Altitude. During map generation this checks terrain relief before the continuous Map Altitude offset.

MaxAltitude

Type: `float`. **Source default:** `Unassigned / type default`.

Maximum allowed terrain relief for this rule when Has Max Altitude is enabled. Placement evaluates relief before continuous Map Altitude is added.

IsGrassDetail

Type: `bool`. **Source default:** `Unassigned / type default`.

Grass prefab rule: MapPropsGenerator places the largest patch sizes first, then fills free tiles with smaller rules. Tile Percent is the per-rule chance; Random Rotation is respected. Uses the map grass controls and shared prefab meshes.

MapGeneratorImageLayer

Configuration

SourceImageNamePatterns

Type: `List<string>`. **Source default:** `new()`.

Source image name fragments handled by this shared palette. Matching ignores spaces, punctuation, case and file extensions.

Layers

Type: `List<ImageImporterLayer>`. **Source default:** `new()`.

Ordered color-to-surface mappings used when classifying an imported image.

Color

Type: `Color`. **Source default:** `Color.white`.

Reference image color matched by this palette layer.

UnityLayer

Type: `int`. **Source default:** `Unassigned / type default`.

Unity layer assigned to terrain generated from this palette layer.

Prefix

Type: `string`. **Source default:** `"ImportedLayer_"`.

Terrain prefix, for example Grass_, DirtPath_, DirtForest_, Water_ or Mountain_. A shade suffix such as Grass_Green_ is optional; reserved names are protected automatically without changing biome/prop prefix rules.

BiomeMaterialId

Type: `int`. **Source default:** `Unassigned / type default`.

Inspector bounds: `Min(0)`.

Index into the map or biome image-material channels. An explicit layer material takes priority.

GrassPrefabLibrary

Configuration

Medium

Type: `GameObject`. **Source default:** `Unassigned / type default`.

Authored small grass-patch prefab used by the fallback grass placement rules.

Large

Type: `GameObject`. **Source default:** `Unassigned / type default`.

Authored large grass-patch prefab used by the fallback grass placement rules.

BladeTexture

Type: `Texture2D`. **Source default:** `Unassigned / type default`.

Shared blade texture supplied when an owned grass material needs an explicit texture reference.

Shader material properties

These defaults come from shader source, not from the tuned biome materials. A material can override them. Shader labels describe their purpose; the technical manual explains the depth, camera, lighting, mask and pipeline requirements.

Sorolif/RTS Terrain/Autumn Water

_Color - Deep Forest Water

Type / range: `Color`. **Shader default:** `(0.035,0.17,0.14,0.91)`.

_ShallowColor - Clear Amber Shallows

Type / range: `Color`. **Shader default:** `(0.28,0.43,0.26,0.28)`.

_FoamColor - Foam Color

Type / range: `Color`. **Shader default:** `(0.92,0.94,0.85,1)`.

_ReflectionColor - Soft Reflection Tint

Type / range: `Color`. **Shader default:** `(0.58,0.62,0.44,1)`.

_UseDepth - Use Camera Depth

Type / range: `Float`. **Shader default:** `1`.

_DepthRange - Shallow Depth Range

Type / range: `Range(.1,12)`. **Shader default:** `2.6`.

_EdgeFade - Edge Transparency Depth

Type / range: `Range(0,1)`. **Shader default:** `0.12`.

_FoamWidth - Shore Foam Depth

Type / range: `Range(0,1.5)`. **Shader default:** `0.3`.

_FoamStrength - Shore Foam Strength

Type / range: `Range(0,1)`. **Shader default:** `0.3`.

_FoamScale - Foam Breakup Scale

Type / range: `Range(.1,8)`. **Shader default:** `2.5`.

_ShoreWaveStrength - Rolling Shore Foam

Type / range: `Range(0,1)`. **Shader default:** `0.08`.

_ShoreWaveSpeed - Shore Wave Speed

Type / range: `Range(0,1)`. **Shader default:** `0.09`.

_FlowFoamStrength - Broken Current Foam

Type / range: `Range(0,1)`. **Shader default:** `0.06`.

_FlowDirection - Flow Direction X Z

Type / range: `Vector`. **Shader default:** `(0.8,0.6,0,0)`.

_WaveScale - Ripple World Scale

Type / range: `Range(.01,4)`. **Shader default:** `0.65`.

_WaveSpeed - Ripple Speed

Type / range: `Range(0,2)`. **Shader default:** `0.2`.

_WaveStrength - Ripple Normal Strength

Type / range: `Range(0,.5)`. **Shader default:** `0.12`.

_WaveColorStrength - Broad Color Variation

Type / range: `Range(0,.5)`. **Shader default:** `0.1`.

_WaveHeight - Visual Wave Height

Type / range: `Range(0,.25)`. **Shader default:** `0.01`.

_GlintStrength - Sun Glint Strength

Type / range: `Range(0,2)`. **Shader default:** `0.08`.

_SpecularPower - Sun Glint Sharpness

Type / range: `Range(8,256)`. **Shader default:** `72`.

_ReflectionStrength - Fresnel Reflection

Type / range: `Range(0,1)`. **Shader default:** `0.16`.

_FresnelPower - Fresnel Power

Type / range: `Range(.5,8)`. **Shader default:** `3`.

_LightingStrength - Directional Lighting

Type / range: `Range(0,1)`. **Shader default:** `0.35`.

_ShadowStrength - Received Shadow Strength

Type / range: `Range(0,1)`. **Shader default:** `0.45`.

_SunDirection - Fallback Light Direction

Type / range: `Vector`. **Shader default:** `(0.35,0.8,0.45,0)`.

_LeafGold - Golden Leaves

Type / range: `Color`. **Shader default:** `(0.96,0.63,0.08,1)`.

_LeafCopper - Copper Leaves

Type / range: `Color`. **Shader default:** `(0.78,0.31,0.08,1)`.

_LeafRusset - Russet Leaves

Type / range: `Color`. **Shader default:** `(0.55,0.1,0.055,1)`.

_LeafDensity - Shore Leaf Density

Type / range: `Range(0,1)`. **Shader default:** `0.78`.

_LeafOpenWaterDensity - Open Water Leaf Density

Type / range: `Range(0,1)`. **Shader default:** `0.015`.

_LeafShoreDepth - Leaf Shore Depth Band

Type / range: `Range(.1,4)`. **Shader default:** `1.2`.

_LeafCellScale - Leaf Pattern World Scale

Type / range: `Range(.3,5)`. **Shader default:** `2.4`.

_LeafSize - Leaf Size in Cell

Type / range: `Range(.08,.42)`. **Shader default:** `0.34`.

_LeafDriftSpeed - Leaf Drift World Units per Second

Type / range: `Range(0,.3)`. **Shader default:** `0.025`.

_LeafTurnSpeed - Leaf Turn Speed

Type / range: `Range(0,1)`. **Shader default:** `0.16`.

_LeafClusterStrength - Leaf Cluster Strength

Type / range: `Range(0,1)`. **Shader default:** `0.55`.

_LeafColorVariation - Leaf Color Variation

Type / range: `Range(0,.6)`. **Shader default:** `0.2`.

_LeafOpacityFade - Leaf Opacity Fade

Type / range: `Range(.01,.5)`. **Shader default:** `0.2`.

_CausticsStrength - Soft Underwater Light

Type / range: `Range(0,1)`. **Shader default:** `0.045`.

_CausticsScale - Underwater Light Scale

Type / range: `Range(.1,8)`. **Shader default:** `0.8`.

_CausticsDepth - Underwater Light Visible Depth

Type / range: `Range(.1,16)`. **Shader default:** `3.5`.

_RefractionBlend - Refracted Bed Blend

Type / range: `Range(0,1)`. **Shader default:** `1`.

_RefractionStrength - Bed Distortion

Type / range: `Range(0,.08)`. **Shader default:** `0.007`.

_AnimationTime - Animation Time -1 Uses Clock

Type / range: `Float`. **Shader default:** `-1`.

Sorolif/RTS Terrain/Forest Water

_Color - Deep Blue Water

Type / range: `Color`. **Shader default:** `(0.015,0.37,0.62,0.92)`.

_ShallowColor - Clear Turquoise Shallows

Type / range: `Color`. **Shader default:** `(0.025,0.78,0.84,0.38)`.

_FoamColor - Foam Color

Type / range: `Color`. **Shader default:** `(0.92,0.98,0.95,1)`.

_ReflectionColor - Soft Reflection Tint

Type / range: `Color`. **Shader default:** `(0.48,0.71,0.79,1)`.

_UseDepth - Use Camera Depth

Type / range: `Float`. **Shader default:** `1`.

_DepthRange - Shallow Depth Range

Type / range: `Range(.1,12)`. **Shader default:** `3.8`.

_EdgeFade - Edge Transparency Depth

Type / range: `Range(0,1)`. **Shader default:** `0.12`.

_FoamWidth - Shore Foam Depth

Type / range: `Range(0,1.5)`. **Shader default:** `0.24`.

_FoamStrength - Shore Foam Strength

Type / range: `Range(0,1)`. **Shader default:** `0.5`.

_FoamScale - Foam Breakup Scale

Type / range: `Range(.1,8)`. **Shader default:** `2.6`.

_ShoreWaveStrength - Incoming Shore Waves

Type / range: `Range(0,1)`. **Shader default:** `0.12`.

_ShoreWaveSpeed - Shore Wave Speed

Type / range: `Range(0,1)`. **Shader default:** `0.13`.

_FlowDirection - Flow Direction X Z

Type / range: `Vector`. **Shader default:** `(0.85,0.52,0,0)`.

_WaveScale - Wave World Scale

Type / range: `Range(.01,4)`. **Shader default:** `0.48`.

_WaveSpeed - Wave Speed

Type / range: `Range(0,2)`. **Shader default:** `0.55`.

_WaveStrength - Wave Normal Strength

Type / range: `Range(0,.5)`. **Shader default:** `0.085`.

_WaveColorStrength - Broad Color Variation

Type / range: `Range(0,.5)`. **Shader default:** `0.32`.

_WaveHeight - Visual Wave Height

Type / range: `Range(0,.25)`. **Shader default:** `0.025`.

_GlintStrength - Small Sun Glints

Type / range: `Range(0,2)`. **Shader default:** `0.7`.

_SpecularPower - Sun Glint Sharpness

Type / range: `Range(8,256)`. **Shader default:** `64`.

_ReflectionStrength - Fresnel Reflection

Type / range: `Range(0,1)`. **Shader default:** `0.14`.

_FresnelPower - Fresnel Power

Type / range: `Range(.5,8)`. **Shader default:** `3`.

_LightingStrength - Directional Lighting

Type / range: `Range(0,1)`. **Shader default:** `0.38`.

_ShadowStrength - Received Shadow Strength

Type / range: `Range(0,1)`. **Shader default:** `0.42`.

_SunDirection - Fallback Light Direction

Type / range: `Vector`. **Shader default:** `(0.35,0.8,0.45,0)`.

_ShoalDepth - Shallow Detail Depth Band

Type / range: `Range(.5,8)`. **Shader default:** `1.8`.

_ShoalWaveStrength - Shallow Ripple Boost

Type / range: `Range(0,2)`. **Shader default:** `1.2`.

_RippleScale - Fine Ripple World Scale

Type / range: `Range(.1,4)`. **Shader default:** `1.35`.

_RippleStrength - Fine Ripple Normal Strength

Type / range: `Range(0,.4)`. **Shader default:** `0.3`.

_RippleSpeed - Fine Ripple Speed

Type / range: `Range(0,3)`. **Shader default:** `0.85`.

_ShoalSpecularStrength - Shoal Specular Boost

Type / range: `Range(0,2)`. **Shader default:** `1.1`.

_OceanDepthStart - Open Sea Start Depth (0 = Any Depth)

Type / range: `Range(0,20)`. **Shader default:** `0`.

_OceanDepthTransition - Open Sea Depth Transition

Type / range: `Range(.01,10)`. **Shader default:** `1`.

_OceanWaveHeight - Open Sea Wave Height

Type / range: `Range(0,.5)`. **Shader default:** `0.18`.

_OceanWaveScale - Open Sea Wave Scale

Type / range: `Range(.1,3)`. **Shader default:** `0.95`.

_OceanWaveSpeed - Open Sea Wave Speed

Type / range: `Range(0,3)`. **Shader default:** `0.9`.

_OceanFoamThreshold - Crest Foam Height Threshold

Type / range: `Range(0,.9)`. **Shader default:** `0.3`.

_OceanVoronoiStrength - Wave Voronoi Variation

Type / range: `Range(0,1)`. **Shader default:** `0.35`.

_OceanVoronoiScale - Wave Voronoi Scale

Type / range: `Range(.02,1)`. **Shader default:** `0.18`.

_ShoalFoamStrength - Open Sea Crest Foam Strength

Type / range: `Range(0,1)`. **Shader default:** `0.85`.

_ShoalShoreClearance - Open Sea Distance From Shore

Type / range: `Range(0,12)`. **Shader default:** `3`.

_SurfaceFoamTex - Surface Foam Texture

Type / range: `2D`. **Shader default:** `"black" {}`.

_SurfaceFoamTextureStrength - Crest Foam Texture Mask

Type / range: `Range(0,1)`. **Shader default:** `0`.

_SurfaceFoamTextureScale - Foam Texture World Scale

Type / range: `Range(.005,.5)`. **Shader default:** `0.06`.

_SurfaceFoamTextureSpeed - Foam Drift Speed

Type / range: `Range(0,1)`. **Shader default:** `0.18`.

_SurfaceFoamTextureContrast - Foam Texture Contrast

Type / range: `Range(.25,3)`. **Shader default:** `1.4`.

_ShoalWaveHeight - Raised Seabed Wave Height

Type / range: `Range(0,.3)`. **Shader default:** `0.12`.

_FishColor - Fish Silhouette Tint and Opacity

Type / range: `Color`. **Shader default:** `(0.025,0.13,0.15,0.52)`.

_FishDensity - Fish Spawn Probability

Type / range: `Range(0,1)`. **Shader default:** `0.13`.

_FishSize - Fish Size World Units

Type / range: `Range(.05,.5)`. **Shader default:** `0.24`.

_FishScale - Fish Pattern World Scale

Type / range: `Range(.2,2)`. **Shader default:** `0.65`.

_FishSpeed - Fish Swim Speed

Type / range: `Range(0,2)`. **Shader default:** `0.65`.

_FishDepth - Fish Depth Below Surface

Type / range: `Range(.1,2)`. **Shader default:** `0.45`.

_FishVisibilityDepth - Fish Visible Water Depth

Type / range: `Range(1,8)`. **Shader default:** `4`.

_CausticsTex - Seabed Caustics Mask

Type / range: `2D`. **Shader default:** `"black" {}`.

_CausticsStrength - Soft Seabed Light

Type / range: `Range(0,1)`. **Shader default:** `0.1`.

_CausticsScale - Seabed Light Scale

Type / range: `Range(.1,8)`. **Shader default:** `1.1`.

_CausticsDepth - Seabed Light Visible Depth

Type / range: `Range(.1,16)`. **Shader default:** `4`.

_RefractionBlend - Refracted Bed Blend

Type / range: `Range(0,1)`. **Shader default:** `1`.

_RefractionStrength - Bed Distortion

Type / range: `Range(0,.08)`. **Shader default:** `0.009`.

_AnimationTime - Animation Time -1 Uses Clock

Type / range: `Float`. **Shader default:** `-1`.

Sorolif/RTS Props/Atlas Lit

_MainTex - Albedo Atlas

Type / range: `2D`. **Shader default:** `"white" {}`.

_Color - Tint

Type / range: `Color`. **Shader default:** `(1,1,1,1)`.

_Cutoff - Alpha Cutoff

Type / range: `Range(0,1)`. **Shader default:** `.3`.

_GroundBlendStrength - Grass Soil Blend

Type / range: `Range(0,1)`. **Shader default:** `0`.

_ReferenceHeight - Grass Blade Height

Type / range: `Float`. **Shader default:** `.65`.

_FoliagePaletteStrength - Foliage Palette Strength

Type / range: `Range(0,1)`. **Shader default:** `0`.

_FoliageShade - Foliage Deep Green

Type / range: `Color`. **Shader default:** `(.08,.24,.16,1)`.

_FoliageSun - Foliage Sunlit Olive

Type / range: `Color`. **Shader default:** `(.5,.6,.25,1)`.

_WindStrength - Wind Strength (World Units)

Type / range: `Range(0,.4)`. **Shader default:** `0`.

_WindSpeed - Wind Speed

Type / range: `Range(0,4)`. **Shader default:** `1.1`.

_WindScale - Wind Spatial Variation

Type / range: `Range(.05,4)`. **Shader default:** `.6`.

_WindDirection - Wind Direction X Z

Type / range: `Vector`. **Shader default:** `(1,0,.4,0)`.

_WindRootHeight - Anchored Root Height

Type / range: `Range(0,1)`. **Shader default:** `.08`.

_WindBendHeight - Full Bend Height

Type / range: `Range(.1,8)`. **Shader default:** `2`.

_WindFlutter - Foliage Flutter

Type / range: `Range(0,1)`. **Shader default:** `.35`.

_WindTime - Wind Time -1 Uses Clock

Type / range: `Float`. **Shader default:** `-1`.

_GroundColor - Sampled Soil

Type / range: `Color`. **Shader default:** `(0,0,0,0)`.

Sorolif/RTS Terrain/Snow Water

_Color - Deep Cobalt Water

Type / range: `Color`. **Shader default:** `(.012,.13,.64,.96)`.

_ShallowColor - Clear Cyan Water

Type / range: `Color`. **Shader default:** `(.02,.75,.96,.46)`.

_FoamColor - Shore Foam

Type / range: `Color`. **Shader default:** `(.86,.96,1,1)`.

_ReflectionColor - Sky Reflection Tint

Type / range: `Color`. **Shader default:** `(.40,.72,.96,1)`.

_UseDepth - Use Camera Depth

Type / range: `Float`. **Shader default:** `1`.

_DepthRange - Shallow Depth Range

Type / range: `Range(.1,12)`. **Shader default:** `2.2`.

_EdgeFade - Edge Transparency Depth

Type / range: `Range(0,1)`. **Shader default:** `.12`.

_FoamWidth - Shore Foam Depth

Type / range: `Range(0,1.5)`. **Shader default:** `.28`.

_FoamStrength - Shore Foam Strength

Type / range: `Range(0,1)`. **Shader default:** `.48`.

_FoamScale - Foam Breakup Scale

Type / range: `Range(.1,8)`. **Shader default:** `2.8`.

_ShoreWaveStrength - Rolling Shore Foam

Type / range: `Range(0,1)`. **Shader default:** `.12`.

_ShoreWaveSpeed - Shore Wave Speed

Type / range: `Range(0,1)`. **Shader default:** `.10`.

_FlowDirection - Surface and Ice Direction X Z

Type / range: `Vector`. **Shader default:** `(.85,.52,0,0)`.

_WaveScale - Ripple Scale

Type / range: `Range(.01,4)`. **Shader default:** `.35`.

_WaveSpeed - Ripple Speed

Type / range: `Range(0,2)`. **Shader default:** `.25`.

_WaveStrength - Ripple Normal Strength

Type / range: `Range(0,.5)`. **Shader default:** `.055`.

_WaveColorStrength - Broad Color Variation

Type / range: `Range(0,.5)`. **Shader default:** `.12`.

_WaveHeight - Visual Wave Height

Type / range: `Range(0,.25)`. **Shader default:** `.015`.

_GlintStrength - Sun Glint Strength

Type / range: `Range(0,2)`. **Shader default:** `.14`.

_SpecularPower - Sun Glint Sharpness

Type / range: `Range(8,256)`. **Shader default:** `100`.

_ReflectionStrength - Fresnel Reflection

Type / range: `Range(0,1)`. **Shader default:** `.24`.

_FresnelPower - Fresnel Power

Type / range: `Range(.5,8)`. **Shader default:** `3`.

_LightingStrength - Directional Lighting

Type / range: `Range(0,1)`. **Shader default:** `.30`.

_ShadowStrength - Received Shadow Strength

Type / range: `Range(0,1)`. **Shader default:** `.40`.

_SunDirection - Fallback Light Direction

Type / range: `Vector`. **Shader default:** `(.35,.8,.45,0)`.

_CurrentColor - Cold Current Tint

Type / range: `Color`. **Shader default:** `(.04,.70,.90,1)`.

_CurrentStrength - Current Volume Strength

Type / range: `Range(0,2)`. **Shader default:** `.30`.

_CurrentScale - Current World Scale

Type / range: `Range(.02,2)`. **Shader default:** `.42`.

_CurrentSpeed - Current Travel Speed

Type / range: `Range(0,1)`. **Shader default:** `.10`.

_CurrentDepth - Current Visible Depth

Type / range: `Range(.1,16)`. **Shader default:** `7`.

_CurrentDirection - Current Direction X Z

Type / range: `Vector`. **Shader default:** `(.75,.35,0,0)`.

_IceColor - Ice Color and Opacity

Type / range: `Color`. **Shader default:** `(.76,.93,1,.96)`.

_IceDensity - Ice Floe Density

Type / range: `Range(0,1)`. **Shader default:** `.06`.

_IceScale - Ice Cell World Scale

Type / range: `Range(.05,3)`. **Shader default:** `.45`.

_IceSize - Ice Floe Size

Type / range: `Range(.05,.42)`. **Shader default:** `.32`.

_IceFreeboard - Ice Height Above Water

Type / range: `Range(.01,.35)`. **Shader default:** `.16`.

_IceDriftSpeed - Ice Drift World Units per Second

Type / range: `Range(0,.5)`. **Shader default:** `.045`.

_IceRotationSpeed - Ice Rotation Speed

Type / range: `Range(0,.5)`. **Shader default:** `.035`.

_ShoreIceStrength - Shore Ice Coverage

Type / range: `Range(0,1)`. **Shader default:** `.85`.

_ShoreIceWidth - Shore Ice Depth Band

Type / range: `Range(.1,2)`. **Shader default:** `.75`.

_ShoreIceBreakSpeed - Shore Ice Break Cycle Speed

Type / range: `Range(0,.3)`. **Shader default:** `.06`.

_ShoreIceDrift - Breakaway Travel in Ice Cells

Type / range: `Range(0,.6)`. **Shader default:** `.35`.

_CausticsStrength - Soft Seabed Light

Type / range: `Range(0,1)`. **Shader default:** `.12`.

_CausticsScale - Seabed Light Scale

Type / range: `Range(.1,8)`. **Shader default:** `.60`.

_CausticsDepth - Seabed Light Visible Depth

Type / range: `Range(.1,16)`. **Shader default:** `5.6`.

_RefractionBlend - Refracted Bed Blend

Type / range: `Range(0,1)`. **Shader default:** `1`.

_RefractionStrength - Bed Distortion

Type / range: `Range(0,.08)`. **Shader default:** `.008`.

_AnimationTime - Animation Time -1 Uses Clock

Type / range: `Float`. **Shader default:** `-1`.

Sorolif/RTS Terrain/Stylized Grass

_BaseColor - Root Color

Type / range: `Color`. **Shader default:** `(.16,.28,.055,1)`.

_TipColor - Tip Color

Type / range: `Color`. **Shader default:** `(.51,.65,.18,1)`.

_ColorVariation - Tuft Color Variation

Type / range: `Range(0,1)`. **Shader default:** `.3`.

_RootShade - Root Occlusion

Type / range: `Range(0,1)`. **Shader default:** `.22`.

_GroundBlendStrength - Root Tint Strength

Type / range: `Range(0,1)`. **Shader default:** `1`.

_GroundBlendHeight - Root Tint Gradient Height

Type / range: `Range(.05,.8)`. **Shader default:** `.38`.

_GroundTintStrength - Blade Soil Tint Strength

Type / range: `Range(0,1)`. **Shader default:** `0`.

_GroundColor - Sampled Soil Color

Type / range: `Color`. **Shader default:** `(0,0,0,0)`.

_Cutoff - Alpha Cutoff

Type / range: `Range(.05,.95)`. **Shader default:** `.38`.

_MainTex - Grass Texture (RGBA)

Type / range: `2D`. **Shader default:** `"white" {}`.

_ReferenceHeight - Fallback Grass Height

Type / range: `Float`. **Shader default:** `.65`.

_WindDirection - Wind Direction X Z

Type / range: `Vector`. **Shader default:** `(1,0,.4,0)`.

_WindStrength - Wind Bending

Type / range: `Range(0,1)`. **Shader default:** `.3`.

_WindSpeed - Wind Speed

Type / range: `Range(0,5)`. **Shader default:** `1.3`.

_WindScale - Gust World Scale

Type / range: `Range(.01,2)`. **Shader default:** `.22`.

_WindTint - Wind Highlight Color

Type / range: `Color`. **Shader default:** `(.62,.69,.35,1)`.

_WindHighlightStrength - Wind Highlight Strength

Type / range: `Range(0,1)`. **Shader default:** `.4`.

_WindHighlightStart - Wind Highlight Start

Type / range: `Range(0,.95)`. **Shader default:** `.45`.

_WindHighlightContrast - Wind Highlight Contrast

Type / range: `Range(.5,6)`. **Shader default:** `2`.

_ShadowStrength - Received Shadow Strength

Type / range: `Range(0,1)`. **Shader default:** `.65`.

_FadeStart - Fade Start (World Units)

Type / range: `Float`. **Shader default:** `100`.

_FadeEnd - Fade End (World Units)

Type / range: `Float`. **Shader default:** `150`.

_AnimationTime - Animation Time -1 Uses Clock

Type / range: `Float`. **Shader default:** `-1`.

Sorolif/RTS Terrain/Stylized Water

_Color - Deep Water

Type / range: `Color`. **Shader default:** `(0.035, 0.25, 0.34, 0.96)`.

_ShallowColor - Shallow Water

Type / range: `Color`. **Shader default:** `(0.16, 0.54, 0.49, 0.64)`.

_FoamColor - Foam

Type / range: `Color`. **Shader default:** `(0.80, 0.93, 0.87, 1)`.

_ReflectionColor - Sky Reflection Tint

Type / range: `Color`. **Shader default:** `(0.43, 0.66, 0.73, 1)`.

_UseDepth - Use Camera Depth

Type / range: `Float`. **Shader default:** `1`.

_DepthRange - Shallow Depth Range

Type / range: `Range(0.1, 12)`. **Shader default:** `2.2`.

_EdgeFade - Edge Transparency Depth

Type / range: `Range(0, 1)`. **Shader default:** `0.12`.

_FoamWidth - Shore Foam Depth

Type / range: `Range(0, 1.5)`. **Shader default:** `0.36`.

_FoamStrength - Shore Foam Strength

Type / range: `Range(0, 1)`. **Shader default:** `0.48`.

_FoamScale - Foam Breakup Scale

Type / range: `Range(0.1, 8)`. **Shader default:** `1.8`.

_FoamBreakup - Foam Lace Contrast

Type / range: `Range(0, 1)`. **Shader default:** `0`.

_ShoreWaveStrength - Rolling Shore Foam

Type / range: `Range(0, 1)`. **Shader default:** `0.18`.

_ShoreWaveSpeed - Shore Wave Speed

Type / range: `Range(0, 1)`. **Shader default:** `0.12`.

_FlowDirection - Flow Direction X Z

Type / range: `Vector`. **Shader default:** `(0.85, 0.52, 0, 0)`.

_WaveScale - Ripple Scale

Type / range: `Range(0.01, 4)`. **Shader default:** `0.32`.

_WaveSpeed - Ripple Speed

Type / range: `Range(0, 2)`. **Shader default:** `0.32`.

_WaveStrength - Ripple Normal Strength

Type / range: `Range(0, 0.5)`. **Shader default:** `0.16`.

_WaveHeight - Visual Wave Height

Type / range: `Range(0, 0.25)`. **Shader default:** `0`.

_WaveColorStrength - Surface Color Variation

Type / range: `Range(0, 0.5)`. **Shader default:** `0.08`.

_CrestStrength - Sparse Surface Foam

Type / range: `Range(0, 1)`. **Shader default:** `0.12`.

_GlintStrength - Sun Glint Strength

Type / range: `Range(0, 2)`. **Shader default:** `0.2`.

_SpecularPower - Sun Glint Sharpness

Type / range: `Range(8, 256)`. **Shader default:** `64`.

_ReflectionStrength - Fresnel Reflection

Type / range: `Range(0, 1)`. **Shader default:** `0.2`.

_FresnelPower - Fresnel Power

Type / range: `Range(0.5, 8)`. **Shader default:** `3.2`.

_LightingStrength - Directional Lighting

Type / range: `Range(0, 1)`. **Shader default:** `0.55`.

_ShadowStrength - Received Shadow Strength

Type / range: `Range(0, 1)`. **Shader default:** `0.45`.

_SunDirection - Fallback Light Direction

Type / range: `Vector`. **Shader default:** `(0.35, 0.8, 0.45, 0)`.

_CausticsStrength - Shallow Caustics

Type / range: `Range(0, 1)`. **Shader default:** `0.08`.

_CausticsScale - Caustics Scale

Type / range: `Range(0.1, 8)`. **Shader default:** `1.1`.

_CausticsDepth - Caustics Visible Depth

Type / range: `Range(0.1, 16)`. **Shader default:** `3.2`.

_CausticsDistortion - Caustics Flow Distortion

Type / range: `Range(0, 1)`. **Shader default:** `0.28`.

_CausticsSoftness - Caustics Diffusion

Type / range: `Range(0, 1)`. **Shader default:** `0`.

_CausticsBlur - Seabed Voronoi Blur

Type / range: `Range(0, 0.5)`. **Shader default:** `0`.

_RefractionBlend - Refracted Bed Blend

Type / range: `Range(0, 1)`. **Shader default:** `0`.

_RefractionStrength - Bed Distortion

Type / range: `Range(0, 0.08)`. **Shader default:** `0`.

_SurfaceFilmColor - Surface Film Tint

Type / range: `Color`. **Shader default:** `(0.38, 0.42, 0.20, 1)`.

_SurfaceFilmStrength - Surface Film Strength

Type / range: `Range(0, 1)`. **Shader default:** `0`.

_SurfaceFilmScale - Surface Film Scale

Type / range: `Range(0.05, 3)`. **Shader default:** `0.42`.

_EmissionStrength - Fluid Emission

Type / range: `Range(0, 2)`. **Shader default:** `0`.

_AnimationTime - Animation Time -1 Uses Clock

Type / range: `Float`. **Shader default:** `-1`.

Sorolif/RTS Terrain/Triplanar

_TopMap - Top Texture

Type / range: `2D`. **Shader default:** `"white" {}`.

_SideMap - Cliff / Side Texture

Type / range: `2D`. **Shader default:** `"white" {}`.

_TopTint - Top Tint

Type / range: `Color`. **Shader default:** `(0.5, 0.7, 0.45, 1)`.

_SideTint - Side Tint

Type / range: `Color`. **Shader default:** `(0.34, 0.3, 0.25, 1)`.

_TextureScale - World Texture Scale

Type / range: `Range(0.01, 2)`. **Shader default:** `0.10`.

_TextureStrength - Texture Detail Strength

Type / range: `Range(0, 1)`. **Shader default:** `0.65`.

_TextureBrightness - Texture Brightness

Type / range: `Range(0.5, 2)`. **Shader default:** `1.15`.

_ShadowStrength - Cast Shadow Strength

Type / range: `Range(0, 1)`. **Shader default:** `0.7`.

_ShadowTint - Cast Shadow Tint

Type / range: `Color`. **Shader default:** `(0.45, 0.57, 0.68, 1)`.

_SeamBlend - Repeat Seam Blend

Type / range: `Range(0, 0.2)`. **Shader default:** `0.05`.

_MacroVariation - Broad Color Variation

Type / range: `Range(0, 0.2)`. **Shader default:** `0.025`.

_BlendSharpness - Projection Blend

Type / range: `Range(1, 16)`. **Shader default:** `5`.

_SlopeDarkening - Slope Darkening

Type / range: `Range(0, 0.7)`. **Shader default:** `0.10`.

_HeightTint - High Ground Tint

Type / range: `Color`. **Shader default:** `(0.7, 0.78, 0.68, 1)`.

_HeightStart - High Ground Start

Type / range: `Float`. **Shader default:** `2`.

_HeightBlend - High Ground Blend

Type / range: `Range(0.01, 10)`. **Shader default:** `3`.

_SunDirection - Light Direction

Type / range: `Vector`. **Shader default:** `(0.35, 0.8, 0.45, 0)`.

_Ambient - Ambient Light

Type / range: `Range(0, 1)`. **Shader default:** `0.58`.

_LightingStrength - Lighting Strength

Type / range: `Range(0, 1)`. **Shader default:** `0.62`.

Sorolif/RTS Terrain/Triplanar Stylized

_TopMap - Top Pattern

Type / range: `2D`. **Shader default:** `"white" {}`.

_SideMap - Side Pattern

Type / range: `2D`. **Shader default:** `"white" {}`.

_TopTint - Top Tint

Type / range: `Color`. **Shader default:** `(0.42, 0.68, 0.35, 1)`.

_SideTint - Side Tint

Type / range: `Color`. **Shader default:** `(0.28, 0.24, 0.2, 1)`.

_TextureScale - World Texture Scale

Type / range: `Range(0.01, 2)`. **Shader default:** `0.10`.

_TextureStrength - Texture Detail Strength

Type / range: `Range(0, 1)`. **Shader default:** `0.65`.

_TextureBrightness - Texture Brightness

Type / range: `Range(0.5, 2)`. **Shader default:** `1.15`.

_ShadowStrength - Cast Shadow Strength

Type / range: `Range(0, 1)`. **Shader default:** `0.7`.

_ShadowTint - Cast Shadow Tint

Type / range: `Color`. **Shader default:** `(0.45, 0.57, 0.68, 1)`.

_SeamBlend - Repeat Seam Blend

Type / range: `Range(0, 0.2)`. **Shader default:** `0.05`.

_MacroVariation - Broad Color Variation

Type / range: `Range(0, 0.2)`. **Shader default:** `0.025`.

_BlendSharpness - Projection Blend

Type / range: `Range(1, 16)`. **Shader default:** `7`.

_SlopeAccentColor - Slope Accent

Type / range: `Color`. **Shader default:** `(0.16, 0.12, 0.1, 1)`.

_SlopeAccentStrength - Slope Accent Strength

Type / range: `Range(0, 1)`. **Shader default:** `0.18`.

_LightBands - Light Bands

Type / range: `Range(2, 8)`. **Shader default:** `4`.

_BandSoftness - Light Band Softness

Type / range: `Range(0, 0.5)`. **Shader default:** `0.18`.

_SunDirection - Light Direction

Type / range: `Vector`. **Shader default:** `(0.35, 0.8, 0.45, 0)`.

_Ambient - Ambient Light

Type / range: `Range(0, 1)`. **Shader default:** `0.62`.

_RimColor - Rim Color

Type / range: `Color`. **Shader default:** `(0.65, 0.82, 0.7, 1)`.

_RimStrength - Rim Strength

Type / range: `Range(0, 1)`. **Shader default:** `0.04`.

Parameter reference

Audited on 16 October 2026 against the working module, Unity 6000.2.13f1.

The current MapDefinition exposes **214 public recipe fields**. See the current catalogue for every field, source default, tooltip and shader property. The following usage notes retain their original audit dates.

Grass details (18 October 2026)

Enable Grass Details

Use and condition: Enables grass prefab rules in MapPropsGenerator in both modes. Requires Enable Props Generation.

Grass Detail Density

Use and condition: 0-4 multiplier on each grass rule placement chance, capped at 100%. Use 1 for biome percentages. Zero disables placements; higher values never stack patches.

Grass Detail Height

Use and condition: Minimum/maximum blade height in cell units (0.05-2), applied through vertical scale. Medium fills a 1 × 1 cell tile and Large 4 × 4, with overlapping border blades to avoid empty seams between adjoining patches. Mesh assets stay shared.

Grass Detail Material

Use and condition: Per-map override; otherwise BiomeSetting.GrassDetailMaterial, then the included material for that biome.

BiomeSetting.GrassDetailMaterial

Use and condition: Separate from the ground GrassMaterial; shader controls color, blade shape and wind.

InstConfig.IsGrassDetail

Use and condition: Marks a grass prefab and opts into map height/material styling. MapPropsGenerator places larger Scale areas first, then fills free tiles with smaller rules; Tile Percent and yaw are respected. GrassPrefab and legacy TransparentFX prefabs named Grass are also recognized. If no grass rule is authored, included rules are added in memory.

See GrassDetails.md for placement, wind, palettes and performance.

How to compare settings

Disable Use Random Seed, keep the seed and all other inputs fixed, change one setting and regenerate. A parameter is active only when its feature, generation mode and required assets apply. Requested counts can be reduced by space, resource, landing and path safety checks.

A screenshot illustrates a visible result; it does not validate gameplay connectivity. Mesh optimization changes vertex/index storage while preserving appearance. Cell Size changes world scale even when the export camera fits the new bounds. Fairness tolerances change the score rather than repositioning terrain.

Corrected behavior

- Resource Fairness Tolerance contributes to the starting-resource score. Weights: bases 25, expansions 25, rush safety 20, connectivity 20 and resources 10. Missing requested resources still reduce the score with a loose tolerance. Resources are assigned to the nearest base and starting access compares the closest requested resources; neutral/expansion mines can therefore contribute if they are closer.

- Forest Density Multiplier means a percentage throughout 0-100. Legacy serialized values 1-5 migrate once to their old 18-62% coverage. New low percentages remain low after saving/reloading.

- Forest Zone Count Multiplier controls spatial fragmentation: 1 produces a few broad woods and 10 produces many smaller groves at comparable density. The seed fixes the layout and the pattern scale follows map size. Density is still controlled separately. Path Connection Count uses its full 1-6 range when enough eligible expansion links exist.

- Main Base Ramp Count distributes 1-4 distinct accesses around the whole main plateau, including attached main/natural pairs. Their shared transition counts as one access. Natural Expansion Second Access requests two exterior exits rather than one, in addition to the main-facing access. Unsafe directions use alternatives without merging existing ramp bands. The gallery compares both settings at fixed camera positions.

- Procedural plateau roads use the final terrain and its real ramps. Choke routes reserved before height generation are rerouted on dry ground when a cliff blocks them. Curve smoothing respects plateau walls; river crossing reservations remain unchanged. An unreachable link is omitted instead of painting a road against a cliff.

- Mountain cliff formations follow local ridges and slopes, with buried bases and joined faces. Their renderer and collider bounds include tilt. Mountain Cliff Material overrides the mountain terrain fallback on generated instances only.

- Procedural mountain density follows land area. Mountain Zone Count is the reference for 80x80 land cells at density 1; Mountain Density Multiplier defaults to 3. Formations spread across map sectors. A candidate cannot disconnect previously connected open terrain, including walkable foothills.

- Imported Mountain Height controls painted mountain peaks independently of Mountain Max Height. The supported imported range is 2.5-6 units. Mountain count, width, falloff, area, sharpness and path clearance apply to procedural placement.

- Oil fields use the resolved water layout, including Random; River skips automatic oil placement.

- Water Height is fixed at -0.5 before the optional altitude offset. Cliff Parent and Resource Parent display automatic output locations; they remain API output-parent inputs when calling the individual subsystems directly.

Optional outputs and project integration

Both helper outputs and navigation graph updates are disabled by default. These settings apply to both generation modes and are retained in saved recipes. See Optional outputs for setup and fallbacks.

`EnableMinimapGeneration`

Applies when: Water exists; both modes

Effect: Opt-in water rendering copy without colliders; false by default.

Consumer: WaterGenerator.cs

`MinimapLayerName`

Applies when: Minimap enabled

Effect: Existing Unity layer; falls back to the configured water layer.

Consumer: WaterGenerator.cs

`EnableAerialGeneration`

Applies when: Both modes

Effect: Opt-in hidden flight mesh/collider; false by default.

Consumer: MapGeneratorTester.cs, MapAerialMesh.cs

`AerialLayerName`

Applies when: Aerial enabled

Effect: Existing flight layer; falls back to Ignore Raycast.

Consumer: MapAerialMesh.cs

`AerialClearance`

Applies when: Aerial enabled

Effect: Minimum world-space clearance over ordinary ground/water; 3 by default.

Consumer: MapAerialMesh.cs

`GroundLayerName`

Applies when: Generated ground surfaces

Effect: Ground, paths and plateaus; Default fallback. Explicit image-palette layers keep priority.

Consumer: TerrainMeshGenerator.cs

`EnvironmentLayerName`

Applies when: Rock, seabed, cliffs and side walls

Effect: Existing environment layer; Default fallback.

Consumer: TerrainMeshGenerator.cs, CliffGenerator.cs, MapAltitudeBorderMesh.cs

`WaterLayerName`

Applies when: Water exists

Effect: Normal water surface and collider layer; Default fallback.

Consumer: WaterGenerator.cs

`BridgeLayerName`

Applies when: Generated ramp surfaces and bridge decks

Effect: Shared Walkable layer, with configured ground fallback. Separate barrier colliders and ramp sides use Environment Layer Name.

Consumer: MapDefinition.cs

`AdditionalPropPlacementLayers`

Applies when: Prop generation

Effect: Extra collider layers added to the automatic placement mask; does not change per-prop layers.

Consumer: MapGeneratorTester.cs, MapDefinition.cs

`EnableNavigationGraphUpdates`

Applies when: Compatible optional API installed

Effect: Opts into legacy A* graph updates; false by default.

Consumer: MapGeneratorTester.cs

`WaterNavigationTag`

Applies when: Graph updates enabled; water exists

Effect: Graph node tag index, 0-31, initially 1.

Consumer: MapGeneratorTester.cs

`AerialNavigationTag`

Applies when: Graph updates and Aerial enabled

Effect: Graph node tag index, 0-31, initially 2.

Consumer: MapGeneratorTester.cs

`EditorBridgePrefabSearchPath`

Applies when: Editor; map and biome bridge prefabs empty

Effect: Optional project-relative prefab lookup; clear to disable. No game-specific asset location.

Consumer: MapDefinition.cs

Application by parameter

Direct material and prefab overrides take priority over their biome defaults. Empty overrides use the resolved biome. The consumer links point to implementation, not merely serialization or Inspector display.

General

`MapName`

Applies when: Both modes

Effect: Generated hierarchy name, saved recipe name and screenshot filename.

Consumer: MapGeneratorTester.cs, MapScreenshotCapture.cs

`MapSize`

Applies when: Both modes

Effect: Number of cells along each side; imported images are sampled to this resolution.

Consumer: TerrainMeshGenerator.cs, MapGenerationPreview.cs

`PlayerCount`

Applies when: Procedural

Effect: Requested player bases, before Additional Main Bases.

Consumer: BasePlacementSystem.cs

`FullMapSymetry`

Applies when: Procedural

Effect: Opt into full-map symmetry; false by default. Disabled retains the original natural generator and player-start layout without extra symmetry-group starts.

Consumer: Full-map symmetry

`Symmetry`

Applies when: Procedural

Effect: Player-start placement in the original generator. With Full Map Symmetry enabled, also mirrors water, relief, paths, forests, resources and props; extra starts complete the selected groups. None keeps free generation.

Consumer: Full-map symmetry

`GenerationMode`

Applies when: Both modes

Effect: Selects procedural planning or image classification.

Consumer: MapGeneratorTester.cs, TerrainMeshGenerator.cs

`Biome`

Applies when: Both modes; no explicit Biome Override

Effect: Biome type used to filter compatible Editor presets. Runtime uses the explicit Biome Override reference.

Consumer: MapDefinition.cs, MapGeneratorTester.cs

`BiomeOverride`

Applies when: Both modes

Effect: Explicit biome asset used at runtime; direct per-map fields retain priority.

Consumer: MapDefinition.cs, MapGeneratorTester.cs

`WaterLayout`

Applies when: Procedural

Effect: Primary water layout. River skirts settlement clearings and uses opposite borders through the center when the full channel fits; occupied central zones take priority. Additional rivers also respect clearings. Lake keeps chokes and paths on dry land. Random uses the resolved layout rules.

Consumer: MapGeneratorTester.cs, ResourcePlacementSystem.cs

Map Altitude

`EnableMapAltitude`

Applies when: Both modes

Effect: Adds height-map relief while keeping lakes/seas horizontal and through-flow rivers descending toward a common outlet. Seabed and nearby banks follow those levels; objects follow the adjusted terrain.

Consumer: MapGeneratorTester.cs, MapAltitudeHeightMap.cs

`AltitudeMinHeight`

Applies when: Both modes; Map Altitude enabled

Effect: Height-map offset in map units at black pixels. Local terrain features are added on top of this altitude.

Consumer: MapAltitudeHeightMap.cs

`AltitudeMaxHeight`

Applies when: Both modes; Map Altitude enabled

Effect: Height-map offset in map units at white pixels. Local terrain features are added on top of this altitude.

Consumer: MapAltitudeHeightMap.cs

`AltitudeHeightMap`

Applies when: Both modes; Map Altitude enabled

Effect: Optional readable grayscale height map, independent of Source Map Image. Leave empty for seeded hills. Import with Read/Write enabled, sRGB disabled and no compression.

Consumer: MapAltitudeApplier.cs, MapAltitudeHeightMap.cs

`AltitudeHeightMapOriginTopLeft`

Applies when: Both modes; Map Altitude enabled; custom height map

Effect: Flip the custom height map vertically. By default its bottom-left pixel maps to the terrain's minimum X/Z corner.

Consumer: MapAltitudeHeightMap.cs

`AltitudeNoiseScale`

Applies when: Both modes; Map Altitude enabled; no custom height map

Effect: Frequency of automatically generated hills in cell units. Lower values give broader hills. Ignored when a custom height map is assigned.

Consumer: MapAltitudeHeightMap.cs

Image Import

`SourceMapImage`

Applies when: Image

Effect: Source Map Image setting; applied by the consumer below.

Consumer: ImageMapImportSystem.cs

`ImageLayerSet`

Applies when: Image

Effect: Image Layer Set setting; applied by the consumer below.

Consumer: TerrainMeshGenerator.cs, ImageMapImportSystem.cs

`SimplifyImageLayers`

Applies when: Image

Effect: For painted/scanned maps: use only the selected palette, sample a wider area to suppress grain, and merge shades of the same terrain/Unity layer into the most-used reference. Disable to preserve separate painted material variants and outline segmentation.

Consumer: ImageMapImportSystem.cs

`EnableImageContourSmoothing`

Applies when: Image

Effect: Rounds imported terrain contours with shared mesh vertices fitted to the source image. Preserves tile types, water, resources and narrow passages; independent of cell-color cleanup.

Consumer: ImageMapImportSystem.cs, TerrainContourSmoothing.cs

`ImageContourSmoothingStrength`

Applies when: Image

Effect: Image Contour Smoothing Strength setting; applied by the consumer below.

Consumer: ImageMapImportSystem.cs, TerrainContourSmoothing.cs

`ImageOriginTopLeft`

Applies when: Image

Effect: Image Origin Top Left setting; applied by the consumer below.

Consumer: ImageMapImportSystem.cs

`ImageCellSampleRadius`

Applies when: Image

Effect: Sampling area around a cell. The simplified palette also enforces a minimum wider sample.

Consumer: ImageMapImportSystem.cs

`ImageColorTolerance`

Applies when: Image

Effect: Tolerance used to match image colors against palette or fallback reference colors.

Consumer: ImageMapImportSystem.cs

`EnableImageRegionSegmentation`

Applies when: Image; segmentation enabled and no simplified palette

Effect: Ink/boundary region voting. Simplify Image Layers with a palette bypasses this pipeline.

Consumer: ImageMapImportSystem.cs

`ImageRegionInkValueThreshold`

Applies when: Image; segmentation enabled and no simplified palette

Effect: Maximum pixel value classified as dark region ink.

Consumer: ImageMapImportSystem.cs

`ImageRegionInkSaturationThreshold`

Applies when: Image; segmentation enabled and no simplified palette

Effect: Saturation threshold for region ink recognition.

Consumer: ImageMapImportSystem.cs

`ImageRegionBoundaryColor`

Applies when: Image; segmentation enabled and no simplified palette

Effect: Reference boundary ink color.

Consumer: ImageMapImportSystem.cs

`ImageRegionBoundaryColorTolerance`

Applies when: Image; segmentation enabled and no simplified palette

Effect: Tolerance for the boundary color mask.

Consumer: ImageMapImportSystem.cs

`ImageRegionBoundaryExpansion`

Applies when: Image; segmentation enabled and no simplified palette

Effect: Pixel dilation of the detected region boundary mask.

Consumer: ImageMapImportSystem.cs

`ImageMinimumRegionPixelCount`

Applies when: Image; segmentation enabled and no simplified palette

Effect: Smallest connected pixel region considered for a cell-region override.

Consumer: ImageMapImportSystem.cs

`ImageMinimumRegionCellVoteRatio`

Applies when: Image; segmentation enabled and no simplified palette

Effect: Minimum dominant-region votes required to override local sampling.

Consumer: ImageMapImportSystem.cs

`ImageLargeWaterRegionOverrideMaxRatio`

Applies when: Image; segmentation enabled and no simplified palette

Effect: Limit for large-water region overrides relative to the full image.

Consumer: ImageMapImportSystem.cs

`ImageLargeWaterRegionLocalKeepRatio`

Applies when: Image; segmentation enabled and no simplified palette

Effect: Local water evidence needed to keep a large-water region decision.

Consumer: ImageMapImportSystem.cs

`EnableImageSurfaceSmoothing`

Applies when: Image

Effect: Cell-type cleanup after classification, independent of mesh contour smoothing.

Consumer: ImageMapImportSystem.cs

`ImageSurfaceSmoothingIterations`

Applies when: Image; Image Surface Smoothing

Effect: Number of classification cleanup passes.

Consumer: ImageMapImportSystem.cs

`EnableImageGroundRelief`

Applies when: Image

Effect: Adds a subtle, deterministic low-frequency elevation relief to imported ground cells.

Consumer: MapDefinition.cs, ImageMapImportSystem.cs

`ImageGroundReliefStrength`

Applies when: Image; Image Ground Relief

Effect: Image Ground Relief Strength setting; applied by the consumer below.

Consumer: MapDefinition.cs, ImageMapImportSystem.cs

`ImageGroundReliefScale`

Applies when: Image; Image Ground Relief

Effect: Image Ground Relief Scale setting; applied by the consumer below.

Consumer: MapDefinition.cs, ImageMapImportSystem.cs

`ImageWaterCleanupNeighborCount`

Applies when: Image; Image Surface Smoothing

Effect: Water neighbors required by cell cleanup.

Consumer: ImageMapImportSystem.cs

`ImageWaterIslandCleanupMaxArea`

Applies when: Image

Effect: Largest small land patch considered for water-island cleanup.

Consumer: ImageMapImportSystem.cs

`ImageWaterIslandCleanupMinWaterBorderRatio`

Applies when: Image

Effect: Required surrounding-water ratio before removing a small patch.

Consumer: ImageMapImportSystem.cs

`ImageMinimumSurfaceVoteRatio`

Applies when: Image

Effect: Votes needed to recognize a sampled non-mountain surface.

Consumer: ImageMapImportSystem.cs

`ImageMinimumMountainVoteRatio`

Applies when: Image; Has Mountain Zones

Effect: Votes needed to recognize a mountain patch.

Consumer: ImageMapImportSystem.cs

`ImportImageResources`

Applies when: Image

Effect: Imports painted gold/oil markers from the source image. Disabled by default so strategic resources can be placed manually.

Consumer: ImageMapImportSystem.cs

`ImageOilFieldVoteRatio`

Applies when: Image; Import Image Resources

Effect: Votes needed to import an oil marker.

Consumer: ImageMapImportSystem.cs

`ImageGoldMineVoteRatio`

Applies when: Image; Import Image Resources

Effect: Votes needed to import a gold marker.

Consumer: ImageMapImportSystem.cs

`ImageInkValueThreshold`

Applies when: Image

Effect: Threshold excluding dark local ink from terrain-color votes.

Consumer: ImageMapImportSystem.cs

`ImagePaperSaturationThreshold`

Applies when: Image

Effect: Low saturation threshold for detecting empty paper.

Consumer: ImageMapImportSystem.cs

`ImagePaperValueThreshold`

Applies when: Image

Effect: High value threshold for detecting empty paper.

Consumer: ImageMapImportSystem.cs

`ImportedMountainHeight`

Applies when: Image; Has Mountain Zones

Effect: Peak height for mountains painted in an imported image. Procedural mountain settings do not change their authored footprint.

Consumer: ImageMapImportSystem.cs, MountainMassifGenerator.cs

`ImageGrassColor`

Applies when: Image; no Image Layer Set

Effect: Fallback reference for grass recognition without a palette.

Consumer: ImageMapImportSystem.cs

`ImageDirtForestColor`

Applies when: Image; no Image Layer Set

Effect: Fallback reference for forest recognition without a palette.

Consumer: ImageMapImportSystem.cs

`ImageBeachColor`

Applies when: Image; no Image Layer Set

Effect: Fallback beach reference; imports as grass unless Beach As Dirt Path is enabled.

Consumer: ImageMapImportSystem.cs

`ImageWaterColor`

Applies when: Image; no Image Layer Set

Effect: Fallback reference for water recognition without a palette.

Consumer: ImageMapImportSystem.cs

`ImageDirtPathColor`

Applies when: Image; no Image Layer Set

Effect: Fallback reference for painted dirt paths without a palette.

Consumer: ImageMapImportSystem.cs

`ImportImageBeachAsDirtPath`

Applies when: Image; no Image Layer Set

Effect: Classifies fallback beach paint as DirtPath instead of grass.

Consumer: ImageMapImportSystem.cs

`ImageMountainColor`

Applies when: Image; no Image Layer Set

Effect: Fallback reference for mountain recognition without a palette.

Consumer: ImageMapImportSystem.cs

`ImageOilFieldColor`

Applies when: Image; no Image Layer Set; Import Image Resources

Effect: Fallback oil-marker reference without a palette.

Consumer: ImageMapImportSystem.cs

`ImageGoldMineColor`

Applies when: Image; no Image Layer Set; Import Image Resources

Effect: Fallback gold-marker reference without a palette.

Consumer: ImageMapImportSystem.cs

Strategic Layout

`AdditionalMainBases`

Applies when: Procedural

Effect: Extra main-base anchors in addition to Player Count.

Consumer: BasePlacementSystem.cs

`ExpansionsPerPlayer`

Applies when: Procedural

Effect: Requested owned expansions per main base.

Consumer: ExpansionPlacementSystem.cs, FairnessValidationSystem.cs

`ContestedExpansionCount`

Applies when: Procedural

Effect: Requested neutral contested expansion anchors. Seeded whole-layout planning scatters sites over free terrain, balances estimated access from player bases across the complete set, and favors underused sectors instead of fixed player medians. Crowded maps may fit fewer anchors.

Consumer: ExpansionPlacementSystem.cs

`GoldMinesPerPlayer`

Applies when: Procedural

Effect: Starting gold mines per main base. Expansions also request their own mine.

Consumer: ResourcePlacementSystem.cs, FairnessValidationSystem.cs

`OilFieldsPerPlayer`

Applies when: Procedural

Effect: Requested starting offshore oil fields; not generated on resolved River layouts.

Consumer: ResourcePlacementSystem.cs, FairnessValidationSystem.cs

`NeutralOilFieldCount`

Applies when: Procedural

Effect: Requested additional neutral offshore oil fields, outside River layouts.

Consumer: ResourcePlacementSystem.cs

`ChokePoints`

Applies when: Procedural

Effect: Strategic bottlenecks through continuous forest/rock dividers; 0 disables them. Multiple openings are spaced across broad approaches and share land roads. Tiny pockets and crowded optional gates are skipped. Uses final relief, ramps and bridges while preserving settlement access and plateau tops. Lake passages stay dry; Islands creates variable-width straits between irregular low or wooded islands, with optional rock cores. Nearby neutral expansion islands can join without moving resources or adding expansions; player coastlines stay open. Constrained layouts can need extra access openings.

Consumer: StrategicChokePlanner.cs

`NavalZones`

Applies when: Procedural

Effect: Additional water features, after the selected layout. Zero keeps the primary feature.

Consumer: MapDefinition.cs, WaterGenerator.cs through AdditionalNavalZones

Terrain Features

`HasWater`

Applies when: Both modes

Effect: Procedural water switch. Image import automatically enables it when water is classified.

Consumer: CliffGenerator.cs, MapGeneratorTester.cs

`HasBridges`

Applies when: Procedural; respective feature enabled

Effect: Safe bridge planning perpendicular to the local course on resolved River layouts, including oblique path crossings. Dry approaches reconnect paths to shifted landings; the river remains underneath.

Consumer: MapGeneratorTester.cs, WaterGenerator.cs

`HasCliffs`

Applies when: Both modes

Effect: Global cliff details. Plateau walls can still be decorated with this switch disabled.

Consumer: CliffGenerator.cs, MapGeneratorTester.cs

`HasPlateaus`

Applies when: Procedural; respective feature enabled

Effect: Builds strategic plateaus and their access ramps. With HasWater and WaterLayout = Lake, contested expansions retain ordinary ground relief to form low islands.

Consumer: CliffGenerator.cs, MapGeneratorTester.cs

`GenerateNaturalExpansionPlateaus`

Applies when: Procedural; respective feature enabled

Effect: When enabled, natural expansions receive their own lower plateau and a shared transition from the main base. Disable it to keep naturals on regular ground.

Consumer: MapDefinition.cs, WaterGenerator.cs

`HasForestZones`

Applies when: Procedural; respective feature enabled

Effect: Procedural forest surface patches. Does not erase forests painted in an image.

Consumer: TerrainMeshGenerator.cs

`HasMountainZones`

Applies when: Both modes

Effect: Procedural mountain placement, or recognition/shaping of painted image mountains.

Consumer: CliffGenerator.cs, MapGeneratorTester.cs

`EnablePropsGeneration`

Applies when: Both modes

Effect: Enable Props Generation setting; applied by the consumer below.

Consumer: MapGeneratorTester.cs

`GrassPrefabs`

Applies when: Both modes; grass enabled with no authored grass rule

Effect: Explicit fallback grass library. Editor defaults are prepared as serialized scene references; runtime-created definitions supply this library or their own grass rules.

Consumer: MapDefinition.cs

`PropConfigs`

Applies when: Both modes; Props enabled; non-empty array overrides biome configs

Effect: Optional per-map prop placement override. When empty, configs are resolved from the active BiomeSetting as before.

Consumer: BiomeSetting.cs, MapDefinition.cs

`PlateauReliefScale`

Applies when: Procedural; respective feature enabled

Effect: Vertical relief multiplier for strategic plateau tiers.

Consumer: HeightMapGenerator.cs, StrategicPlateauGenerator.cs

`PlateauFootprintScale`

Applies when: Procedural; respective feature enabled

Effect: Scales the buildable footprint around main bases and expansions. Values above 1 create broader plateau tops; available map space and nearby strategic zones can still limit the final size.

Consumer: BasePlacementSystem.cs, ExpansionPlacementSystem.cs

`PlayerPlateauBorderSafetyDistance`

Applies when: Procedural; respective feature enabled

Effect: Minimum distance in grid cells between a player plateau's outer cliff border and the map boundary. Zero allows the plateau to touch the boundary.

Consumer: MapDefinition.cs, BasePlacementSystem.cs

`PlateauContourIrregularity`

Applies when: Procedural; respective feature enabled

Effect: Low-frequency contour variation. Higher values create stronger lobes and recesses without cell-by-cell noise.

Consumer: BasePlacementSystem.cs, StrategicPlateauGenerator.cs

`MainBaseRampCount`

Applies when: Procedural; main plateau; enough safe space for distinct ramps

Effect: Distinct accesses distributed around the whole main plateau (1-4), including with an attached natural. The shared transition counts as one access. Unsafe directions are replaced with legal alternatives; insufficient space can limit the count.

Consumer: MapDefinition.cs, StrategicPlateauGenerator.cs

`PlateauRampWidth`

Applies when: Procedural; respective feature enabled

Effect: Width of planned plateau access ramps in grid cells.

Consumer: TerrainMeshGenerator.cs, ExpansionPlacementSystem.cs

`PlateauRampLength`

Applies when: Procedural; plateaus enabled

Effect: Length of the added downhill volume in cells (default 5, range 2-18). Larger values make a longer, gentler descent. A small added landing joins the plateau outline; original plateau faces are retained.

Consumer: PlateauAccessRamp.cs

`PlateauEdgeThickness`

Applies when: Procedural; respective feature enabled

Effect: Thickness of the non-walkable cliff border around plateau tops.

Consumer: BasePlacementSystem.cs, ExpansionPlacementSystem.cs

`NaturalExpansionSecondAccess`

Applies when: Procedural; natural plateau; enough safe space for two exterior exits

Effect: Creates two distinct exterior exits from each natural plateau instead of one, while retaining the access toward its main base. Requires enough safe space for separate ramps.

Consumer: StrategicPlateauGenerator.cs

`RampDirtPathApproachLength`

Applies when: Procedural; respective feature enabled

Effect: Length in cells of the worn DirtPath area generated before and after every ramp.

Consumer: TerrainMeshGenerator.cs

`RampDirtPathApproachWidth`

Applies when: Procedural; respective feature enabled

Effect: Width multiplier for the worn DirtPath approaches generated at both ends of every ramp.

Consumer: TerrainMeshGenerator.cs

`ForestDensityMultiplier`

Applies when: Procedural; respective feature enabled

Effect: Target forest coverage as a percentage, including values between zero and five. Strategic clearings can reduce the actual coverage.

Consumer: MapDefinition.cs, TerrainMeshGenerator.cs

`ForestZoneCountMultiplier`

Applies when: Procedural; respective feature enabled

Effect: Spatial fragmentation at comparable density: 1 gives a few broad woods; 10 gives many smaller groves. The seed fixes the pattern. Strategic clearings, high coverage and small maps can split or merge patches.

Consumer: TerrainMeshGenerator.cs

`PlateauForestDensityMultiplier`

Applies when: Procedural; plateaus + forests; safe plateau interior

Effect: Controls forest pockets on safe plateau tops as a proportion of the global forest coverage. Zero disables plateau forests; build pads, ramps and strategic paths remain clear.

Consumer: TerrainMeshGenerator.cs

`PlayerBaseForestClearRadius`

Applies when: Procedural; forests or resolved River; river clearance also applies without forests

Effect: Open ground radius around each main base. Clears forests and keeps procedural river channels outside the whole clearing, even with forests disabled. Existing base and plateau footprints stay protected. Also editable under River shape.

Consumer: TerrainMeshGenerator.cs, RiverSettlementClearance.cs

`ExpansionForestClearRadius`

Applies when: Procedural; forests or resolved River; river clearance also applies without forests

Effect: Open ground radius around each owned, natural or contested expansion. Clears forests and keeps procedural rivers outside the clearing. Existing expansion and plateau footprints stay protected. Also editable under River shape.

Consumer: TerrainMeshGenerator.cs, RiverSettlementClearance.cs

`MountainZoneCount`

Applies when: Procedural; Has Mountain Zones; legal footprint and clear routes

Effect: Base number of independent massifs per 80x80 land cells at density 1. The total follows land area and Mountain Density Multiplier; protected space and connectivity can reduce it.

Consumer: HeightMapGenerator.cs, MountainMassifGenerator.cs

`MountainDensityMultiplier`

Applies when: Procedural; Has Mountain Zones; legal footprint and clear routes

Effect: Multiplies the area-scaled number of mountain formations (0.25-4; default 3). Use 0.5-1 for scattered ridges, 3-4 for a mountainous map. The footprint cap and protected travel buffers remain independent.

Consumer: MapDefinition.cs, MountainMassifGenerator.cs

`MountainFalloff`

Applies when: Procedural; Has Mountain Zones; legal footprint and clear routes

Effect: Controls how quickly each ridge rises from its foothills.

Consumer: HeightMapGenerator.cs, MountainMassifGenerator.cs

`MountainMaxHeight`

Applies when: Procedural; Has Mountain Zones; legal footprint and clear routes

Effect: Mountain Max Height setting; applied by the consumer below.

Consumer: HeightMapGenerator.cs, MountainMassifGenerator.cs

`MountainChainWidth`

Applies when: Procedural; Has Mountain Zones; legal footprint and clear routes

Effect: Width multiplier for massifs and ridges. Their two sides vary independently for an asymmetric silhouette.

Consumer: HeightMapGenerator.cs, MountainMassifGenerator.cs

`MountainMaxArea`

Applies when: Procedural; Has Mountain Zones; legal footprint and clear routes

Effect: Maximum footprint, including low foothills, for each mountain formation.

Consumer: MapDefinition.cs, HeightMapGenerator.cs

`MountainRiseSharpness`

Applies when: Procedural; Has Mountain Zones; legal footprint and clear routes

Effect: Higher values sharpen the rocky crest while keeping a low foothill transition.

Consumer: MapDefinition.cs, HeightMapGenerator.cs

`MountainPathClearance`

Applies when: Procedural; Has Mountain Zones; legal footprint and clear routes

Effect: Minimum open buffer reserved around strategic routes, choke corridors, ramps and imported DirtPaths.

Consumer: MapDefinition.cs, MountainMassifGenerator.cs

`EnableMountainCliffDetails`

Applies when: Both modes

Effect: Embeds seeded cliff faces along mountain ridges and slopes, with slight lean and joined groups. Independent of Has Cliffs. Full tilted renderer/collider bounds remain inside safe mountain footprints.

Consumer: CliffGenerator.cs, MapGeneratorTester.cs

`MountainCliffDetailScale`

Applies when: Both modes; Has Mountain Zones + Mountain Cliff Details + eligible cliff prefab

Effect: Size multiplier for buried mountain cliff formations. Unsafe placements are skipped or reduced to preserve neighboring terrain and passages.

Consumer: CliffGenerator.cs

`MountainCliffMaterial`

Applies when: Both modes; Has Mountain Zones + Mountain Cliff Details + eligible cliff prefab

Effect: Optional material override for embedded mountain cliff instances. Empty uses Mountain Material or its biome default; if neither resolves, prefab materials remain. Does not modify the source prefab.

Consumer: MapDefinition.cs

Shared Scene Settings

`CellSize`

Applies when: Both modes

Effect: Cell Size setting; applied by the consumer below.

Consumer: CliffGenerator.cs, MapGeneratorTester.cs

`EnableMeshOptimization`

Applies when: Both modes

Effect: Shares compatible vertices and edges, groups surfaces with the same material and Unity layer, and uses 16-bit indices when possible. Custom UV/normal seams are preserved.

Consumer: TerrainMeshGenerator.cs, WaterGenerator.cs

`EnableVertexPositionNoise`

Applies when: Both modes

Effect: Small vertex displacement; planar jitter yields to shared contour smoothing.

Consumer: CliffGenerator.cs, TerrainMeshGenerator.cs

`EnableProceduralContourSmoothing`

Applies when: Procedural

Effect: Rounds procedural forest, mountain and shoreline contours with shared mesh vertices. Preserves gameplay cells, resources, ramps and reserved passages. Uses contour displacement instead of random planar vertex jitter.

Consumer: TerrainContourSmoothing.cs

`ProceduralContourSmoothingStrength`

Applies when: Procedural

Effect: Procedural Contour Smoothing Strength setting; applied by the consumer below.

Consumer: TerrainContourSmoothing.cs

`EnablePathSmoothing`

Applies when: Both modes

Effect: Refines road contours. Curved procedural roads, essential settlement access and local pointed town branches exist with this option on or off; it does not choose a different network.

Consumer: TerrainMeshGenerator.cs

`PathConnectionCount`

Applies when: Generated land-path network; eligible anchors/roads; authored image surfaces take priority

Effect: Essential settlement access at 1; higher values add alternative links between settlements. Safe terrain and reachable junctions limit actual links.

Consumer: TerrainMeshGenerator.cs

`RandomLongPathCount`

Applies when: Generated land-path network; eligible anchors/roads; authored image surfaces take priority

Effect: Optional long roads crossing the map. Keep at zero for a sparse natural network.

Consumer: TerrainMeshGenerator.cs

`PathBranchCount`

Applies when: Generated land-path network; eligible anchors/roads; authored image surfaces take priority

Effect: Maximum extra decorative branches, in addition to local town streets. The generated extra count is randomized.

Consumer: TerrainMeshGenerator.cs

`WanderingPathCount`

Applies when: Procedural

Effect: Number of additional meandering trails, independent of choke points and strategic roads. They can open narrow passages through forest. Zero disables them.

Consumer: MapDefinition.cs, TerrainMeshGenerator.cs

Terrain Materials

`GrassMaterial`

Applies when: Both modes; matching surface; direct material overrides biome

Effect: Grass Material setting; applied by the consumer below.

Consumer: MapDefinition.cs, MapGenerationPreview.cs

`PlateauMaterial`

Applies when: Plateau meshes; direct material overrides biome; otherwise grass fallback

Effect: Optional top material for strategic plateaus. Falls back to Grass Material.

Consumer: MapDefinition.cs

`DirtPathMaterial`

Applies when: Both modes; matching surface; direct material overrides biome

Effect: Dirt Path Material setting; applied by the consumer below.

Consumer: MapDefinition.cs

`DirtForestMaterial`

Applies when: Both modes; matching surface; direct material overrides biome

Effect: Dirt Forest Material setting; applied by the consumer below.

Consumer: MapDefinition.cs

`UnderwaterMaterial`

Applies when: Both modes; matching surface; direct material overrides biome

Effect: Underwater Material setting; applied by the consumer below.

Consumer: MapDefinition.cs

`EnvMaterial`

Applies when: Both modes; matching surface; direct material overrides biome

Effect: Env Material setting; applied by the consumer below.

Consumer: MapDefinition.cs

`MountainMaterial`

Applies when: Both modes; matching surface; direct material overrides biome

Effect: Mountain Material setting; applied by the consumer below.

Consumer: MapDefinition.cs

`ImageLayerMaterials`

Applies when: Image; matching palette material index; direct array overrides biome entry

Effect: Image Layer Materials setting; applied by the consumer below.

Consumer: MapDefinition.cs

Water Settings

`RiverBridgePrefab`

Applies when: Procedural; Has Water + Has Bridges + resolved River; safe footprint

Effect: Optional map override; otherwise uses BiomeSetting.RiverBridgePrefab, then the Editor fallback. The deck crosses perpendicular to the local river course, fits to the first fully dry shore sections and uses ordinary renderers. Both full-width ends must contact the generated ground; invalid placements are removed. Paths reconnect to the dry bridge landings. Unoccupied shore fragments of at most four cells can remain separate, with a warning; bases, expansions, choke points and larger islands remain required.

Consumer: MapDefinition.cs, MapGeneratorTester.cs

`RiverBridgeWidth`

Applies when: Procedural; Has Water + Has Bridges + resolved River; safe footprint

Effect: River Bridge Width setting; applied by the consumer below.

Consumer: PlateauConnectivityValidator.cs, RiverBridgeGenerator.cs

`RiverBridgeMaxSpan`

Applies when: Procedural; Has Water + Has Bridges + resolved River; safe footprint

Effect: Maximum distance between bridge landings, in grid cells. Bridges follow the local river normal, with a small tolerance for grid-aligned landing cells. Clear banks may have a gentle height difference.

Consumer: RiverBridgeGenerator.cs, RiverPathCrossingPlanner.cs

`WaterMaterial`

Applies when: Both modes; water present; safety and shoreline constraints

Effect: Water Material setting; applied by the consumer below.

Consumer: MapDefinition.cs

`WaterMinimapMaterial`

Applies when: Both modes; Enable Minimap Generation; water present; safety and shoreline constraints

Effect: Material for the optional water rendering copy; inactive when minimap generation is disabled.

Consumer: MapDefinition.cs

`WaterHeight`

Applies when: Automatic output / read-only in generator Inspector

Effect: Fixed water surface at -0.5 before Map Altitude. Normalized during water generation.

Consumer: MapGeneratorTester.cs, ResourcePlacementSystem.cs

`WaterFloorHeight`

Applies when: Both modes; water present; safety and shoreline constraints

Effect: Reference deep seabed height; the deepest configured depth can lower it further.

Consumer: TerrainMeshGenerator.cs, WaterGenerator.cs

`EnableWaterFloorNoise`

Applies when: Water cells; water-floor noise enabled

Effect: Enable Water Floor Noise setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`WaterFloorNoiseStrength`

Applies when: Water cells; water-floor noise enabled

Effect: Water Floor Noise Strength setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`WaterFloorNoiseScale`

Applies when: Water cells; water-floor noise enabled

Effect: Water Floor Noise Scale setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`WaterFloorNoiseSmoothingIterations`

Applies when: Water cells; water-floor noise enabled

Effect: Water Floor Noise Smoothing Iterations setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`WaterVisualBorder`

Applies when: Both modes; water present; safety and shoreline constraints

Effect: Water Visual Border setting; applied by the consumer below.

Consumer: TerrainMeshGenerator.cs, WaterGenerator.cs

`RiverWidth`

Applies when: Procedural; resolved River or EdgeRiver

Effect: River Width setting; applied by the consumer below.

Consumer: WaterGenerator.cs, RiverSettlementClearance.cs

`CurveStrength`

Applies when: Procedural; resolved River or EdgeRiver

Effect: Curve Strength setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`ControlPointCount`

Applies when: Procedural; resolved River or EdgeRiver

Effect: Control Point Count setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`IslandRadiusMin`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Island Radius Min setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`IslandRadiusMax`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Island Radius Max setting; applied by the consumer below.

Consumer: WaterGenerator.cs, BasePlacementSystem.cs

`IslandContourIrregularity`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Controls broad coastline deformation for procedural islands: coves, asymmetric lobes and tapered peninsulas. Strategic footprints remain protected.

Consumer: MapDefinition.cs, WaterGenerator.cs

`IslandMinimumWaterGap`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Minimum navigable water gap, in grid cells, reserved between separate procedural island footprints. Six keeps islands visually close without merging them.

Consumer: MapDefinition.cs, WaterGenerator.cs

`NaturalIslandRadiusMin`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Natural Island Radius Min setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`NaturalIslandRadiusMax`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Natural Island Radius Max setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`IsolatedExpansionIslandRadiusMin`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Isolated Expansion Island Radius Min setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`IsolatedExpansionIslandRadiusMax`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Isolated Expansion Island Radius Max setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`ContestedIslandRadiusMin`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Contested Island Radius Min setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`ContestedIslandRadiusMax`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Contested Island Radius Max setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`BonusIslandRadiusMin`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Bonus Island Radius Min setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`BonusIslandRadiusMax`

Applies when: Procedural; resolved Islands; enough space and corresponding anchors/features

Effect: Bonus Island Radius Max setting; applied by the consumer below.

Consumer: WaterGenerator.cs

`LakeRadiusMin`

Applies when: Procedural; Lake feature; radius capped by strategic space

Effect: Lower bound of the central lake radius in grid cells. The seeded radius is drawn between Min and Max; equal bounds request a fixed radius. Only the map footprint limits large values. Protected bases, plateaus and ramp landings trim the lake locally. Additional lakes use a smaller radius.

Consumer: WaterGenerator.cs

`LakeRadiusMax`

Applies when: Procedural; Lake feature; radius capped by strategic space

Effect: Upper bound of the central lake radius in grid cells, independently of Min. Set equal bounds for a fixed radius. The physical map limit is (Map Size - 2) / 2; strategic dry areas trim the shoreline locally rather than shrinking the whole lake.

Consumer: WaterGenerator.cs

`MinWaterDepth`

Applies when: Both modes; water present; safety and shoreline constraints

Effect: Shallow seabed target; safety keeps it below the fixed water surface.

Consumer: WaterGenerator.cs

`MaxWaterDepth`

Applies when: Both modes; water present; safety and shoreline constraints

Effect: Deep seabed target, combined with Water Floor Height by choosing the deepest value.

Consumer: WaterGenerator.cs

`BeachHeightInner`

Applies when: Both modes; water present; safety and shoreline constraints

Effect: Height of the inner beach transition, subject to dry landing safety.

Consumer: WaterGenerator.cs

`BeachHeightOuter`

Applies when: Both modes; water present; safety and shoreline constraints

Effect: Height of the outer beach transition and buildable shore-access checks.

Consumer: CliffGenerator.cs, TerrainMeshGenerator.cs

Cliff Settings

`CliffPrefabs`

Applies when: Global cliffs, plateau walls and ramp sides

Effect: Wall prefab override; empty follows the biome.

Consumer: MapDefinition.cs

`MountainCliffPrefabs`

Applies when: Mountain outcrops

Effect: Independent crest prefab override; empty follows the biome, then the legacy cliff fallback for older presets.

Consumer: MapDefinition.cs

`CliffParent`

Applies when: Automatic output / read-only in generator Inspector

Effect: Automatically assigned Env/Cliffs output. Direct CliffGenerator calls can use an explicit parent.

Consumer: CliffGenerator.cs, MapGeneratorTester.cs

`CliffHeightThreshold`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Minimum decorated height break, capped at the terrain face split threshold (0.55).

Consumer: CliffGenerator.cs

`CliffEdgeOffset`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Offsets the center of cliff details from the steep terrain face toward the low ground.

Consumer: CliffGenerator.cs

`CliffVerticalOffset`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Raises cliff details relative to the center of the abrupt face, increasing their visible overhang above the high ground.

Consumer: CliffGenerator.cs

`CliffHeightScale`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Extra vertical coverage. Legacy values below 1.15 are upgraded automatically so cliff details extend beneath the complete abrupt face.

Consumer: CliffGenerator.cs

`CliffSegmentOverlap`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Overlap between consecutive cliff details and at corners, in world units.

Consumer: CliffGenerator.cs

`CliffTilesPerPrefab`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Preferred number of terrain edge tiles covered by one cliff prefab. Twelve strongly reduces object count; curvature and concavity safeguards split unsafe spans automatically.

Consumer: CliffGenerator.cs

`CliffDepthScale`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Scales prefab depth across the terrain break without changing its wall length. Lower values reduce visual overlap over plateau tops and low ground.

Consumer: CliffGenerator.cs

`RampCliffDepthScale`

Applies when: Ramp-side cliff fitting; eligible prefab

Effect: Width multiplier for low ramp borders. Default width: about one third of a cell. Their tops only slightly protrude above the ramp; the remaining height is embedded, including with map altitude.

Consumer: CliffGenerator.cs

`RampCliffInset`

Applies when: Plateau ramp-side cliff fitting; eligible prefab

Effect: World-space overlap of each ramp rail beneath the ramp edge. The 0.32 default keeps the prefab tight against the complete descent without materially narrowing its traversable center.

Consumer: CliffGenerator.cs, MapDefinition.cs

`CliffVerticalOverlap`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Extra vertical overlap used to hide the terrain above and below a cliff prefab.

Consumer: CliffGenerator.cs

`CliffRunHeightTolerance`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Maximum high/low height variation merged into one straight cliff run.

Consumer: CliffGenerator.cs

`CliffPrefabLengthAxis`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Local prefab axis treated as its horizontal wall length. Auto selects the largest horizontal extent.

Consumer: CliffGenerator.cs, MapDefinition.cs

`MarkCliffCellsUnbuildable`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Marks cells touching generated cliff walls as non-buildable.

Consumer: CliffGenerator.cs

`CliffRotationOffsetY`

Applies when: Generated cliff details; global cliffs, plateau walls or mountain outcrops

Effect: Cliff Rotation Offset Y setting; applied by the consumer below.

Consumer: CliffGenerator.cs

Resource Settings

`GoldMinePrefab`

Applies when: Resources present; procedural requests or imported resource markers

Effect: Gold Mine Prefab setting; applied by the consumer below.

Consumer: MapDefinition.cs

`OilFieldPrefab`

Applies when: Resources present; procedural requests or imported resource markers

Effect: Oil Field Prefab setting; applied by the consumer below.

Consumer: MapDefinition.cs

`ResourceParent`

Applies when: Automatic output / read-only in generator Inspector

Effect: Automatically assigned Players/Nature output. Direct resource-system calls can use an explicit parent.

Consumer: MapGeneratorTester.cs, ResourcePlacementSystem.cs

`GoldMineMinDistanceFromAnchor`

Applies when: Procedural

Effect: Gold Mine Min Distance From Anchor setting; applied by the consumer below.

Consumer: ResourcePlacementSystem.cs

`GoldMineMaxDistanceFromAnchor`

Applies when: Procedural

Effect: Gold Mine Max Distance From Anchor setting; applied by the consumer below.

Consumer: ResourcePlacementSystem.cs, StrategicPlateauGenerator.cs

`OilFieldMinDistanceFromAnchor`

Applies when: Procedural

Effect: Oil Field Min Distance From Anchor setting; applied by the consumer below.

Consumer: ResourcePlacementSystem.cs

`OilFieldMaxDistanceFromAnchor`

Applies when: Procedural

Effect: Oil Field Max Distance From Anchor setting; applied by the consumer below.

Consumer: ResourcePlacementSystem.cs

Gameplay Constraints

`MinimumRushDistance`

Applies when: Procedural

Effect: Requested minimum base spacing and the rush-distance score floor; dense maps may need relaxed placement.

Consumer: BasePlacementSystem.cs, FairnessValidationSystem.cs

`ExpansionFairnessTolerance`

Applies when: Procedural

Effect: Relative expansion-access distance spread accepted by the gameplay score.

Consumer: FairnessValidationSystem.cs

`ResourceFairnessTolerance`

Applies when: Procedural

Effect: Relative starting-resource access distance spread accepted by the resource score; does not move resources.

Consumer: FairnessValidationSystem.cs

Generation

`Seed`

Applies when: Both modes

Effect: Deterministic generation seed; overwritten for each run when Use Random Seed is enabled.

Consumer: CliffGenerator.cs, MapGeneratorTester.cs

`UseRandomSeed`

Applies when: Both modes

Effect: Chooses a new seed for each generation; disable for meaningful comparisons.

Consumer: [RuntimeMapGenerationExample.cs](../../Assets/SorolifRTSMapGenerator/Examples/Runtime Generation/RuntimeMapGenerationExample.cs#L45), MapRulesSystem.cs

Screenshot Settings

`GenerateScreenshotAfterGeneration`

Applies when: Screenshot capture

Effect: Captures a top-down PNG after a successful map generation.

Consumer: MapGeneratorTester.cs

`ScreenshotOutputFolder`

Applies when: Screenshot capture

Effect: Absolute folder or relative folder. Relative paths use the Unity project root in the Editor and Application.persistentDataPath in a build.

Consumer: MapScreenshotCapture.cs

`ScreenshotWidth`

Applies when: Screenshot capture

Effect: Exported PNG width, independently of map resolution.

Consumer: MapScreenshotCapture.cs

`ScreenshotHeight`

Applies when: Screenshot capture

Effect: Exported PNG height, independently of map resolution.

Consumer: MapScreenshotCapture.cs

`ScreenshotPadding`

Applies when: Screenshot capture

Effect: Extra space around the rendered map as a ratio of its visible bounds.

Consumer: MapScreenshotCapture.cs

`ScreenshotViewMode`

Applies when: Screenshot capture

Effect: Top Down preserves a map-like orthographic view. Isometric reveals terrain height while keeping an orthographic projection.

Consumer: MapScreenshotCapture.cs

`ScreenshotIsometricPitch`

Applies when: Screenshot capture; Isometric view

Effect: Downward camera pitch used by Isometric screenshot mode.

Consumer: MapScreenshotCapture.cs

`ScreenshotIsometricYaw`

Applies when: Screenshot capture; Isometric view

Effect: Horizontal rotation used by Isometric screenshot mode.

Consumer: MapScreenshotCapture.cs

`ScreenshotBackgroundColor`

Applies when: Screenshot capture

Effect: Screenshot Background Color setting; applied by the consumer below.

Consumer: MapScreenshotCapture.cs

`ScreenshotTransparentBackground`

Applies when: Screenshot capture

Effect: Screenshot Transparent Background setting; applied by the consumer below.

Consumer: MapScreenshotCapture.cs

`ScreenshotLayerMask`

Applies when: Screenshot capture

Effect: Unity layers included by the capture camera; affects the export, not generation.

Consumer: MapScreenshotCapture.cs

`ScreenshotOverwriteExisting`

Applies when: Screenshot capture

Effect: Overwrites the map/seed image, or appends a timestamp on collision.

Consumer: MapScreenshotCapture.cs

Biome, decoration and image-palette fields

`BiomeSetting`

Parameter: `BiomeType`

Usage: Type used to filter compatible presets in the Editor.

`BiomeSetting`

Parameter: `BorderPrefab`

Usage: Optional map-border prefab instantiated and fitted by the generator.

`BiomeSetting`

Parameter: `BorderBoxMaterial`

Usage: Material applied to the generated border renderer.

`BiomeSetting`

Parameter: `GrassMaterial`

Usage: Default GrassMaterial reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `PlateauMaterial`

Usage: Default PlateauMaterial reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `DirtPathMaterial`

Usage: Default DirtPathMaterial reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `DirtForestMaterial`

Usage: Default DirtForestMaterial reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `UnderwaterMaterial`

Usage: Default UnderwaterMaterial reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `WaterMaterial`

Usage: Default WaterMaterial reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `WaterMinimapMaterial`

Usage: Default WaterMinimapMaterial reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `EnvMaterial`

Usage: Default EnvMaterial reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `MountainMaterial`

Usage: Default MountainMaterial reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `ImageLayerMaterials`

Usage: Default ImageLayerMaterials reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `RampLayerName`

Usage: Hidden legacy field, retained for saved-data compatibility and ignored. Ramp surfaces use the shared map Walkable layer (`BridgeLayerName`).

`BiomeSetting`

Parameter: `ShowInBiomeSelector`

Usage: Offers this preset and biome in Editor selectors. Enabled for new presets; hiding does not disable explicit references or runtime generation.

`BiomeSetting`

Parameter: `CliffPrefabs`

Usage: Default wall/plateau prefabs, used when the map has no direct override.

`BiomeSetting`

Parameter: `MountainCliffPrefabs`

Usage: Default mountain crests, independently of wall/plateau prefabs. Older empty presets keep the cliff-list fallback.

`BiomeSetting`

Parameter: `GoldMinePrefab`

Usage: Default GoldMinePrefab reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `OilFieldPrefab`

Usage: Default OilFieldPrefab reference, used when the map has no direct override and the corresponding feature is present.

`BiomeSetting`

Parameter: `configs`

Usage: Default decoration rules unless MapDefinition.PropConfigs is non-empty.

`InstConfig`

Parameter: `prefab`

Usage: Prefab to instantiate on a matching surface patch.

`InstConfig`

Parameter: `color`

Usage: Standalone Scan By Color only. Normal generation uses Prefix.

`InstConfig`

Parameter: `Prefix`

Usage: Case-sensitive generated surface prefix, for example DirtForest_.

`InstConfig`

Parameter: `tilePercent`

Usage: Placement probability (0-100) per eligible patch. Footprint and altitude checks can reject placement.

`InstConfig`

Parameter: `Scale`

Usage: Patch size in tiles, not object transform scale. Zero uses the one-tile pass; -1 uses the final detail pass.

`InstConfig`

Parameter: `RandomRotation`

Usage: Seeded random yaw; world objects can rotate their children.

`InstConfig`

Parameter: `IsWorldObject`

Usage: Checks occupancy and uses the world-object footprint tolerance.

`InstConfig`

Parameter: `Layer`

Usage: Unity layer assigned to the instantiated root; range 0-31.

`InstConfig`

Parameter: `HasMaxAltitude`

Usage: Enables rejection when the sampled footprint exceeds MaxAltitude.

`InstConfig`

Parameter: `MaxAltitude`

Usage: Maximum sampled footprint altitude. In generated maps, evaluated before the continuous Map Altitude offset.

`MapGeneratorImageLayer`

Parameter: `SourceImageNamePatterns`

Usage: Editor palette suggestion by normalized source-image name; it does not override an explicitly selected palette.

`MapGeneratorImageLayer`

Parameter: `Layers`

Usage: Palette entries used for image classification.

`ImageImporterLayer`

Parameter: `Color`

Usage: Reference color matched against image samples.

`ImageImporterLayer`

Parameter: `UnityLayer`

Usage: Unity layer assigned to the imported surface mesh.

`ImageImporterLayer`

Parameter: `Prefix`

Usage: Surface/resource identity: grass, forest, path, water, mountain, oil or gold tokens.

`ImageImporterLayer`

Parameter: `BiomeMaterialId`

Usage: Material array index; falls back to a material matching the resolved surface.

Visual examples

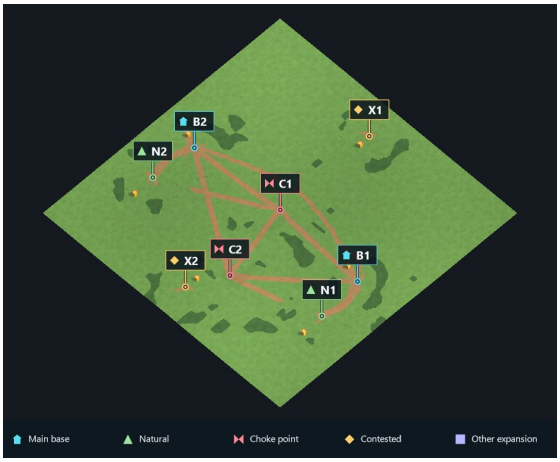
Visual examples are produced from real Unity generations with fixed seeds. The publishing workspace contains the complete comparison gallery and its per-case recipes, diagnostics, geometry statistics and PNGs. A clean source image will not necessarily demonstrate a cleanup or recognition threshold; its effect requires ambiguous/noisy input.

See the integrated How To Use topic **Parameters and conditions** for a concise guide, available in English and French. Screenshot exports and future release artifacts remain separate from the frozen rc9 archives.

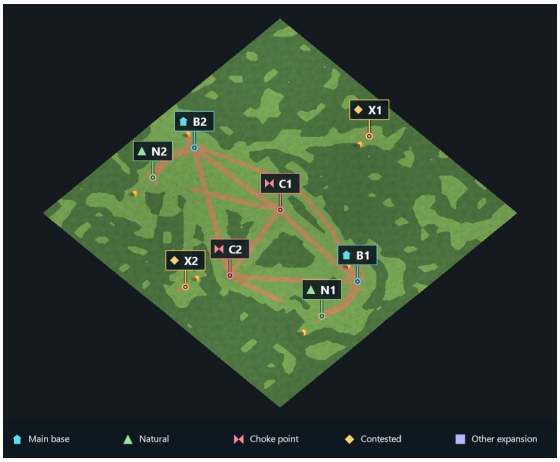
Strategic markers: **B** main base (house), **N** natural expansion (triangle), **C** choke point (bow tie), **X** contested expansion (diamond), **E** other expansion (square). A small ring marks the actual zone center, connected to its label. These documentation overlays do not change the generated map or the screenshot API.

R marks the center of each connected ramp area in the base/plateau close-ups.

ForestDensityMultiplier

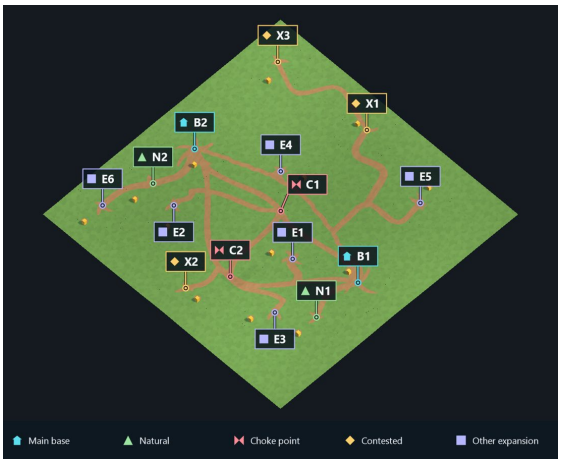


ForestDensityMultiplier = 4

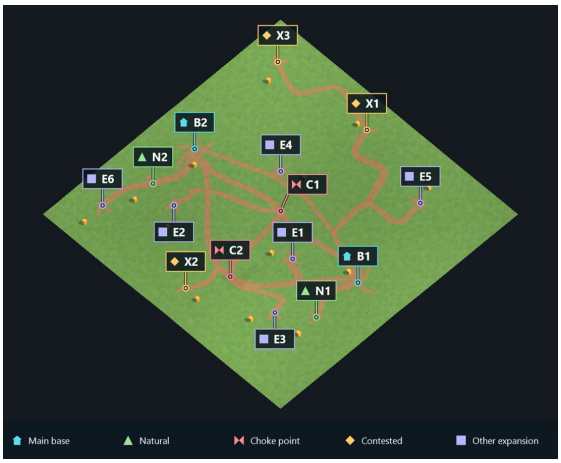


ForestDensityMultiplier = 65

EnablePathSmoothing



EnablePathSmoothing = False



EnablePathSmoothing = True

Close-up of the main base and its natural, using the same camera for both values. **R** marks connected ramp areas. Full-map context and diagnostics are available in the publishing gallery.

ImportedMountainHeight

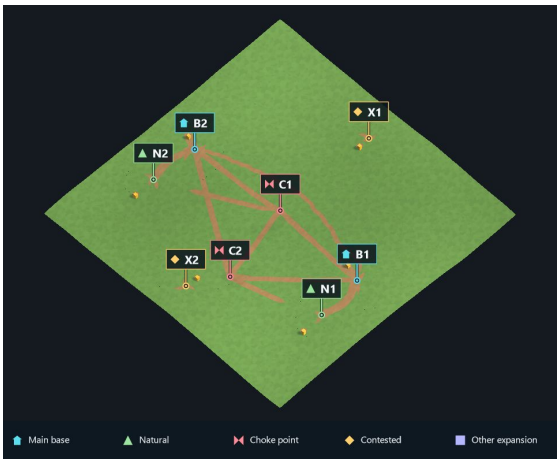


ImportedMountainHeight = 2.5

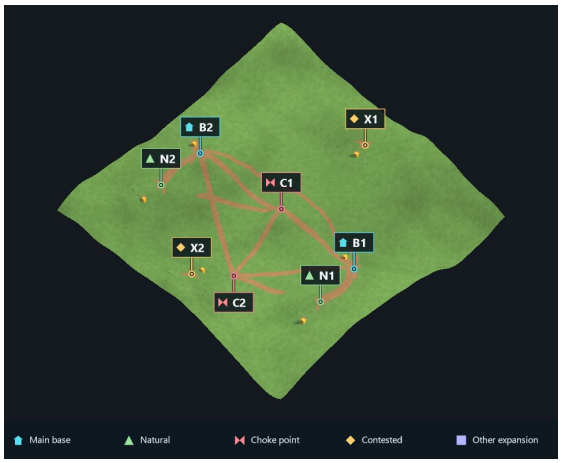


ImportedMountainHeight = 6

AltitudeMaxHeight

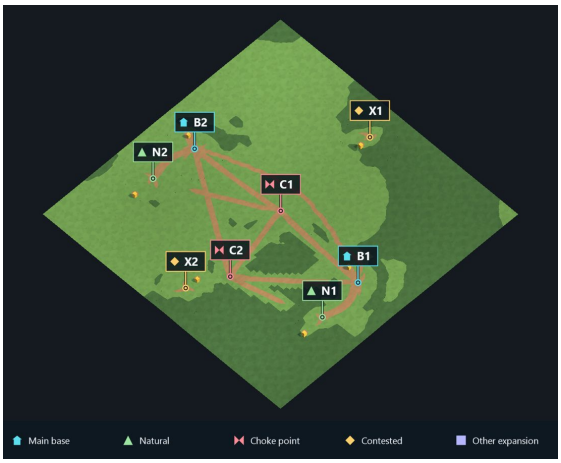


AltitudeMaxHeight = 2

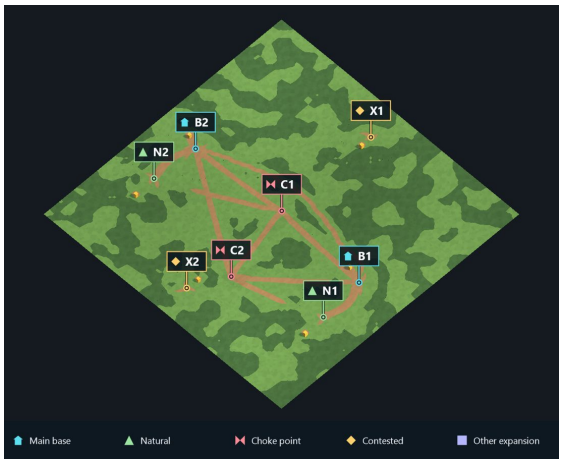


AltitudeMaxHeight = 12

ForestZoneCountMultiplier



ForestZoneCountMultiplier = 1



ForestZoneCountMultiplier = 10

MainBaseRampCount



MainBaseRampCount = 1

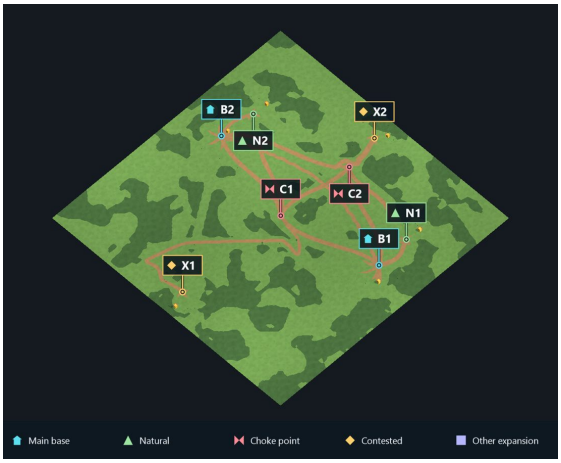


MainBaseRampCount = 4

The example uses no player expansions, a footprint scale of 1.8 and a ramp width of 3 to expose the access bands. **R** marks connected ramp areas on B1. MainBaseRampCount distributes accesses around the entire main plateau. An attached natural retains its shared transition, counted as one of the requested accesses; additional accesses use the

other sides. Obstacles can limit the count; access bands stay separate.

HasMountainZones



HasMountainZones = False



HasMountainZones = True

Whole-map comparison at a fixed camera. Mountain formations follow land area and density, spread across central and outer sectors, and preserve connected open terrain. The displayed coverage counts blocked mountain cells; walkable foothills are additional relief. Image mode retains the painted footprint.

MountainDensityMultiplier



MountainDensityMultiplier = 0.5



MountainDensityMultiplier = 3

Whole-map comparison at a fixed camera. Mountain formations follow land area and density, spread across central and outer sectors, and preserve connected open terrain. The displayed coverage counts blocked mountain cells; walkable foothills are additional relief. Image mode retains the painted footprint.

MountainCliffMaterial

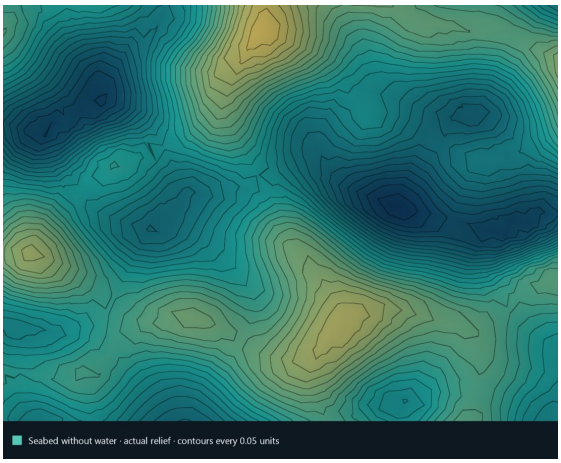


MountainCliffMaterial = None

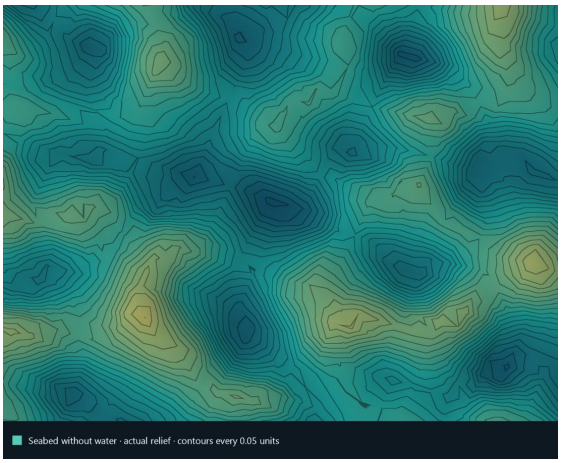


MountainCliffMaterial = Desert_Mountain

EnableWaterFloorNoise



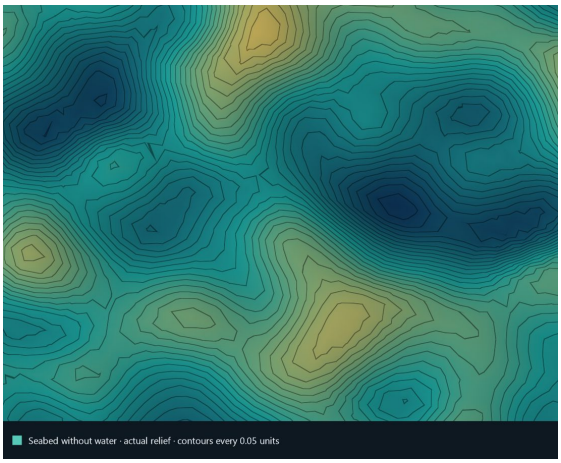
EnableWaterFloorNoise = False



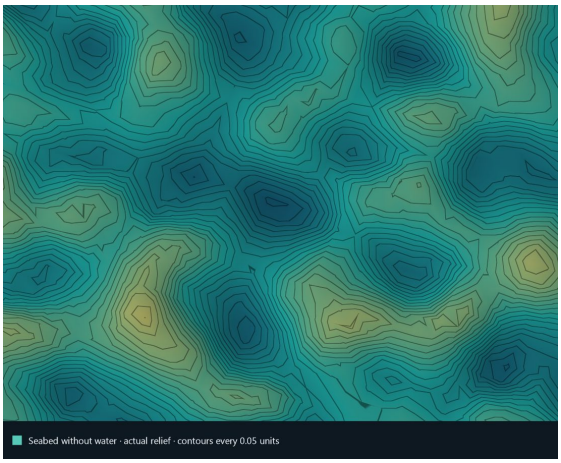
EnableWaterFloorNoise = True

Fixed close-up of the actual seabed, with the water surface hidden, height colors and contours every 0.05 units. Both values share the same palette, lighting and camera; vertical scale is unchanged. The publishing gallery also offers the normal water-surface view and an actual depth profile.

WaterFloorNoiseStrength



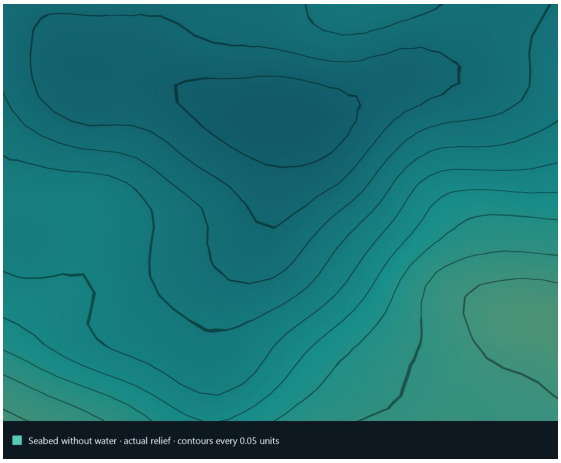
WaterFloorNoiseStrength = 0



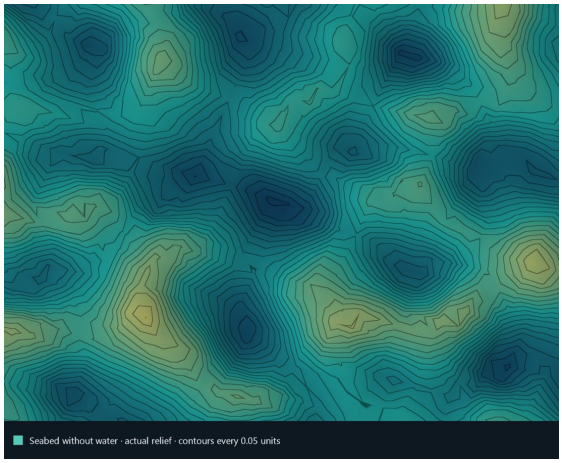
WaterFloorNoiseStrength = 1

Fixed close-up of the actual seabed, with the water surface hidden, height colors and contours every 0.05 units. Both values share the same palette, lighting and camera; vertical scale is unchanged. The publishing gallery also offers the normal water-surface view and an actual depth profile.

WaterFloorNoiseScale



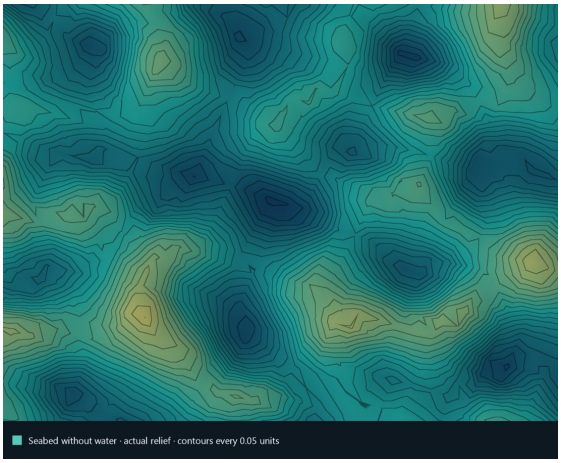
WaterFloorNoiseScale = 0.02



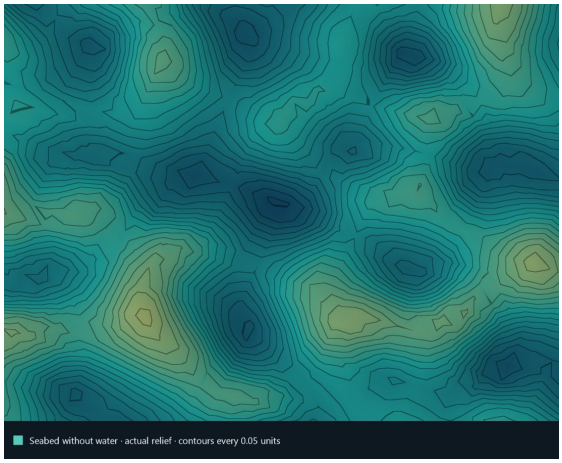
WaterFloorNoiseScale = 0.18

Fixed close-up of the actual seabed, with the water surface hidden, height colors and contours every 0.05 units. Both values share the same palette, lighting and camera; vertical scale is unchanged. The publishing gallery also offers the normal water-surface view and an actual depth profile.

WaterFloorNoiseSmoothingIterations



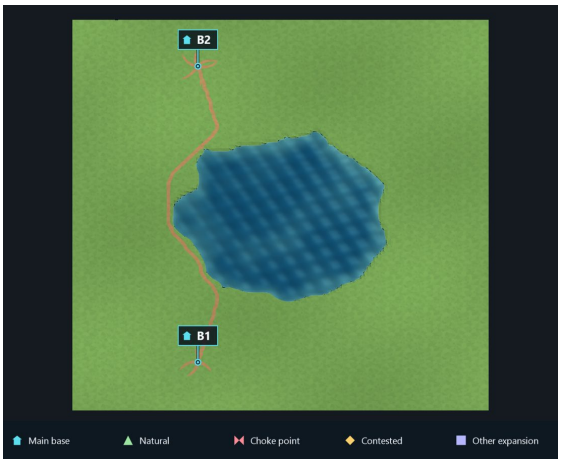
WaterFloorNoiseSmoothingIterations = 0



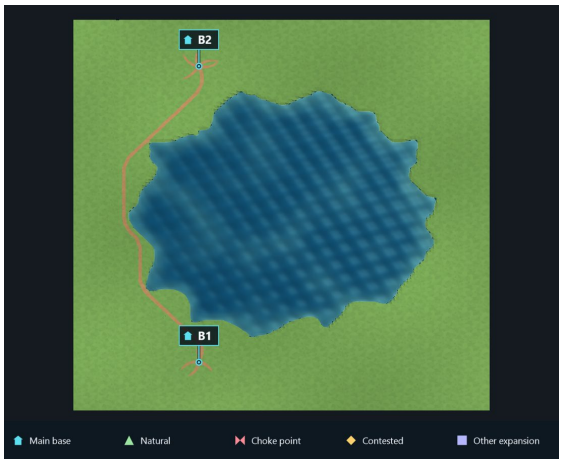
WaterFloorNoiseSmoothingIterations = 3

Fixed close-up of the actual seabed, with the water surface hidden, height colors and contours every 0.05 units. Both values share the same palette, lighting and camera; vertical scale is unchanged. The publishing gallery also offers the normal water-surface view and an actual depth profile.

LakeRadiusMin



LakeRadiusMin = 1

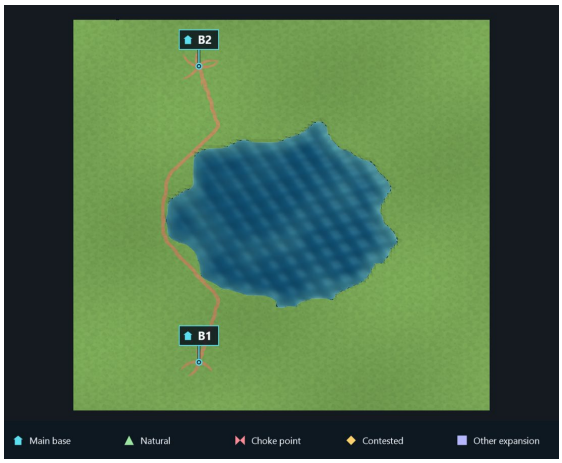


LakeRadiusMin = 42

LakeRadiusMax



LakeRadiusMax = 8



LakeRadiusMax = 50

ChokePoints



ChokePoints = 3



ChokePoints = 6

Islands layout with forests and props enabled, mountains disabled. The same seed and camera show irregular low islands and variable-width straits; only Choke Points changes. Nearby neutral expansion islands retain their coordinates and resources. **C** marks navigable straits and **X** contested expansions. Here 6 requested produces 5

safely spaced passages. Each passage works alone, and closing every passage prevents inter-player sea travel.

PlateauRampWidth



PlateauRampWidth = 3



PlateauRampWidth = 5

Close-up of the main base and its natural, using the same camera for both values. **R** marks connected ramp areas. Full-map context and diagnostics are available in the publishing gallery.

RampDirtPathApproachLength



RampDirtPathApproachLength = 2



RampDirtPathApproachLength = 8

Close-up of the main base and its natural, using the same camera for both values. **R** marks connected ramp areas. Full-map context and diagnostics are available in the publishing gallery.

RampDirtPathApproachWidth



RampDirtPathApproachWidth = 0.75



RampDirtPathApproachWidth = 2

Close-up of the main base and its natural, using the same camera for both values. **R** marks connected ramp areas. Full-map context and diagnostics are available in the publishing gallery.

RampCliffDepthScale



RampCliffDepthScale = 0.5



RampCliffDepthScale = 2

Close-up of the main base and its natural, using the same camera for both values. **R** marks connected ramp areas. Full-map context and diagnostics are available in the publishing gallery.

RampCliffInset



RampCliffInset = 0



RampCliffInset = 0.8

Close-up of the main base and its natural, using the same camera for both values. **R** marks connected ramp areas. Full-map context and diagnostics are available in the publishing gallery.

GenerateNaturalExpansionPlateaus



GenerateNaturalExpansionPlateaus = False



GenerateNaturalExpansionPlateaus = True

Close-up of the main base and its natural, using the same camera for both values. **R** marks connected ramp areas. Full-map context and diagnostics are available in the publishing gallery.

PlateauReliefScale



PlateauReliefScale = 0.4



PlateauReliefScale = 1.4

Close-up of the main base and its natural, using the same camera for both values. **R** marks connected ramp areas. Full-map context and diagnostics are available in the publishing gallery.

NaturalExpansionSecondAccess



NaturalExpansionSecondAccess = False



NaturalExpansionSecondAccess = True

Close-up of the main base and its natural, using the same camera for both values. **R** marks connected ramp areas. Full-map context and diagnostics are available in the publishing gallery.

False creates one exterior exit, true requests two distinct exterior exits, while retaining the main-facing access. A choke exit does not suppress the additional exit. Adequate safe terrain is required.