Remoscape v1.3.1

User's Guide

Remograph

Remoscape v1.3.1 User's Guide

Remograph AB

Website: www.remograph.com

E-mail:
info@remograph.com
sales@remograph.com
support@remograph.com

The terms of service are available on the Remoscape website.

This user's guide

You can read this user's guide from cover to cover to get to know Remoscape before starting to use it. The Settings and Export chapters might be a bit detailed though, so it is recommended to browse them quickly at first and then use them as a reference when using the service.

Or you can just dive in and start using Remoscape directly, learning by doing. It is meant to be intuitive and most of the information in this user's guide is also available behind small clickable help icons like this: ⓘ

Overview

General

Remoscape is an automated, user-friendly and cost-effective cloud service generating 3D landscapes for realtime visualization. The landscapes can be used as visual databases for simulators, realistic environments for games or correlated 3D maps for web applications. Remoscape is available as an online cloud service or as a locally installed software.

This cloud service allows you to upload your geodata, configure some intuitive settings and then just let the cloud do the rest. You will be notified by e-mail when the landscape is generated and the service lets you view it directly in your browser. Then you can export it to any of the file formats below.

To help you along the way, there are small clickable help icons like this to describe the different features of the service: ⓘ

There is also a Remoscape User's Guide (the one you are currently reading) available from the Help menu.

These are the main advantages of the Remoscape cloud service:

- You do not have to use your own powerful and expensive hardware, just leave it to the cloud. You can control Remoscape from a simple web browser.
- Thanks to an intuitive and user-friendly service, you do not have to go through weeks of training. Compared to other desktop products on the market, Remoscape may not offer the same generic functionality but it will produce realistic environments with its own content, based on your simple configuration and the location of your geodata.
- You can preview the generated landscape directly in the browser, before exporting to different file formats.
- The cloud service and the generated landscapes are available from anywhere, just log in to the service from any client.
- You do not have to purchase an expensive desktop product license, just pay as you go with the cloud service. See more in the Pricing section.

Remoscape works in most browsers, but Google Chrome is the recommended optimal browser, especially to ensure rendering performance in the 3D view.

File formats

The following file formats are currently supported for export:

- 3D Tiles (Cesium, Unreal Engine, Unity, etc.)
- MAK MetaFlight (VR-Vantage, VR-Forces)
- MetaFlight
- OpenFlight
- OpenSceneGraph

The following geodata file formats are currently supported for import:

- Raster
  - GeoTIFF
  - DTED
  - ECW
  - JPEG
  - JPEG 2000
  - DEM
- Vector
  - ESRI Shapefile
  - GeoPackage
- Laser
  - LAS
  - LAZ

More import and export formats are being developed and will be added in future versions of Remoscape.

Company

Remoscape is developed by **Remograph**. Read more about the company on the Remograph website.

If you need help tailoring your landscape to your unique system and requirements, Remograph is here for you. It can mean anything from finding the optimal settings or preparing your geodata, to actually implementing new functionality in the cloud service. Contact us at remoscape@remograph.com for more information.

Security

With many of our customers in the defense industry, we are aware of the data security concerns that users might have. We have therefore gathered some information about this below:

Website and server encryption

Remoscape is served over a TLS-encrypted HTTPS session, safely encrypting all communication with the service and all data uploading.

Passwords

User passwords are hashed using one of the strongest password hash functions available.

Data storage

All uploaded data, generated internal data and exported models are stored on the Microsoft Azure cloud storage, using a strong 256-bit AES encryption.

On-premise installation

If the security measures taken and described above are not enough for your business, we also offer Remoscape for on-premise installation on local area networks. Please contact sales@remograph.com for more information.

Third-party dependencies

Here are the licenses of all third-party dependencies used in Remoscape.

Sections

This chapter describes the different sections of the Remoscape cloud service website. They can be reached from the top-right menu.

Projects

This section is only available if you are logged in to the service.

At the top of this page is the **Create Project** button. It will let you create a new project. Just choose its name (which has to be unique among your own current projects) and click the Create button. You will then be sent to the Project page described in the next chapter.

Below the Create Project button, you will find your projects (if any) listed with their current status, the original date of creation and the date for the latest modification. The list is sorted by the latest modification date. Click on a row to go to the project page described in the next chapter.

Account

This section is only available if you are logged in to the service.

Here you will find the information that you entered when signing up to the service. At the moment you will need to contact Remograph at support@remograph.com if you want to change your information. It will be editable in an upcoming version of Remoscape.

About

This section gives you general information about the service, a link to the Remoscape release notes, an introduction video and a number of screenshots from example landscapes.

Pricing

This is the price list of the service.

Terms

This section provides the terms of service.

Privacy

This section provides the privacy policy of the service.

Help

This links to the user's guide PDF that you are reading.

Log in

This will let you log in to the service. It is obviously not available if you are already logged in.

Sign Up

This will let you sign up to the service. After entering all necessary information and accepting the terms of service, there will be a confirmation e-mail sent to you. Use the link in this e-mail to create your account. The Sign Up link is obviously not available if you are already logged in.

Log out

This will let you log out from the service. It is obviously not available if you are not logged in.

Project

Remove and Rename Project

Remove

Use Remove to delete the project.

Rename

Use Rename to give the project a new name. The project name header will turn editable and the new name is confirmed with the Enter key. Remember that the new project name has to be unique among your own projects.

Status

At the top center of the project page is the project status bar.

This bar shows the current stage of the project, as described below. You will be able to reset back to the Configure stage with the Restart button, improving your current configuration.

Configure

This is the first stage of a project, where you will be able to add data source files and specify some intuitive settings. Do not worry, there will be more of these small help icons to help you along the way: ⓘ

The configuration is described in detail in the **Configure** chapter.

When the configuration is complete, click the Upload button. The geodata you provided is now assessed as either **Small**, **Medium** or **Large**, with respect to its size, number of features, etc. (see more in the Pricing section). You will be able to pay by credit card and the amount depends on the project size. Or if you have agreed with Remograph to be invoiced, the upload will start immediately.

Upload

The project is in this stage while uploading the files. From now on you can consider this a "fire-and-forget" project. You will be notified by e-mail on the progress of the project.

Just make sure to leave the browser tab open and active while uploading, browsers tend to upload slowly if left in the background. We make sure to keep the computer from sleeping.

Convert

From now on, you can close the browser and turn off your computer if you want.

A process server is automatically started for your project (it might have to wait for other projects if the number of process servers is maxed out) and your data sources are merged, split and re-projected as required.

Generate

With all data sources converted as required, this stage is automatically activated and the process server starts generating your landscape.

You will be able to navigate through your uploaded and converted data sources in the map view while generating, and see the progress of the blocks that the landscape has been divided into.

View

When the landscape has been completed, you are notified by e-mail and this stage will be activated. In the View stage you will be able to preview your landscape in the 3D view, next to the map view described above.

After previewing you can choose to export to different file formats with the Export button above the map and 3D views. Contact Remograph if you want to use the landscape in a web tool. You can reset back to the Configure stage with the Restart button in order to improve your configuration and restart.

Export

By choosing to export to one of the supported file formats with the Export in the top-right corner, specifying export settings and pushing the Start Export button, the project will be in this stage while exporting and then go back to the View stage.

The export is described in detail in the **Export** chapter.

You will be notified by e-mail when the export has been completed and you can download your landscape.

Downloads

If you have exported this project before, here are the recent exports listed per format and sorted by date. Each link will download a ZIP archive.

Note that large downloads may fail occasionally in Chrome. Just choose Resume on the download if this happens.

Restart

You can reset your project back to the Configure stage with the Restart button in the top right corner, to the left of the Export button. This will let you improve your configuration by changing some settings, remove or add some data.

There are 3 restarts included in each project. Restarting a fourth time will require a new payment, unlocking another 3 restarts.

Note that even within the 3 including restarts, adding new data can result in the project being assessed according to a higher price level than before, requiring a difference payment. So if you for example paid for a Medium landscape and added data so that it is now a Large landscape, you will pay the difference between the cost for a Large landscape and the cost for a Medium landscape.

Configure

This chapter describes the main activity you will engage in when working with Remoscape, namely the project configuration. Here you will be able to add data source files and specify some intuitive settings.

Click the Add Data Source button to add a new data source.

A data source group will be created and expanded. Specify the type of the data source with the **Type** drop-down menu and provide the files with the **Choose files** button, both described below.

Type

Selects the type of data source. The different types are described below, together with their supported file formats. More types and file formats will be added.

NOTE: It is highly recommended to upload a bit more data than needed, spatially. Since the landscape will consist of blocks whose size is calculated as a multiple of texture size and imagery resolution, the landscape will seldom cover your geographic data completely.

Elevation

Raster

Elevation rasters generating the terrain mesh in multiple levels of detail.

Multiple resolutions are supported, using elevation rasters with better resolution where available.

NOTE: The landscape area will be calculated to cover the smallest of the total elevation raster and the total imagery raster. So if you want to use imagery outside the elevation, please add a low-resolution elevation raster there.

For performance reasons, resolution will currently be limited to 10m per pixel at best, regardless of the original raster resolution. This will of improve in upcoming versions of Remoscape.

Here are the supported elevation raster file formats:

- GeoTIFF (.tif / .tif + .tfw + .prj)
- DTED level 0 (.dt0)
- DTED level 1 (.dt1)
- DTED level 2 (.dt2)
- Enhanced Compression Wavelet (.ecw)
- Geo-referenced JPG (.jpg + .aux.xml)
- Geo-referenced JPEG 2000 (.jp2)
- USGS Digital Elevation Model (.dem)

Laser

Elevation point clouds, typically airborne LiDAR data, generating terrain mesh, tree heights and building roof shapes. This will replace any elevation raster.

Here are the supported point cloud file formats:

- ASPRS LASer (.las / .laz)

Imagery

Image

Geo-specific imagery, typically satellite images or aerial photographs.

Multiple resolutions are supported, using imagery with better resolution where available.

NOTE: The landscape area will be calculated to cover the smallest of the total elevation raster and the total imagery raster. So if you want to use imagery outside the elevation, please add a low-resolution elevation raster there.

If metadata including camera/sensor position is given, this will be used to offset the imagery on building roofs to compensate for perspective.

To improve the perceived level of detail and achieve visual cues on low altitudes, a generic detail texture will be mixed with this imagery in the generated landscape.

Here are the supported image file formats:

- GeoTIFF (.tif / .tif + .tfw + .prj)
- Enhanced Compression Wavelet (.ecw)
- Geo-referenced JPG (.jpg + .aux.xml)
- Geo-referenced JPEG 2000 (.jp2)
- Esri Shape (.shp + .shx + .dbf + .prj) or GeoPackage (.gpkg) - metadata specifying camera/sensor position (not necessary but helps with terrain-textured building roofs)

Features

Buildings

Building 2D footprints generating flat roof buildings with a controlled random number of floors, or realistically shaped roofs if laser data is available. Roofs will be textured from terrain imagery if available. Walls will be textured with randomly chosen facade textures or simply gray if specified.

Here are the supported building file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Geotypical

Vector area features generating geotypical imagery from land cover types, if there is no geo-specific imagery such as aerial photographs or satellite images available.

Here are the supported geotypical file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Masts

Vector point features generating masts or communication towers.

Here are the supported mast file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Mixed Vector

Vector data with multiple types in the same file, separated by a specified field. Other vector data sources described here can be added as types by values. This is described in more detail in the **Mixed Vector type** chapter.

Here are the supported mixed vector file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Power Lines

Vector line features generating power lines with pylons and wires.

Here are the supported power line file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Railroads

Vector line features generating railroads, cut out in the landscape, flattened laterally and textured.

Here are the supported railroad file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Roads

Vector line features generating roads, cut out in the landscape, flattened laterally and possibly textured.

Here are the supported road file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Street Lights

Vector point features generating street light models with lightmaps on the geometry below. Note that an alternative way of creating street lights is to distribute them along roads with the Street Lights setting in the Road type.

Here are the supported street light file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Urban Areas

Vector area features representing urban areas. These areas are used to filter street lights automatically placed along roads, so that there will only be street lights within built-up areas.

Note that urban areas will only affect street lights placed along roads, that is if the Street Lights setting is used for roads. They will not affect point features with the Street Lights type.

Also note that these urban areas are not related to the urban land cover type used with the Geotypical data source type, which is assumed to use a higher level of detail than these areas.

Urban areas may be used for other purposes in future versions.

Here are the supported urban area file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Vegetation

Vector area or point features generating trees, randomly distributed or according to given laser data.

For area features, trees will be placed with a random distribution and height according to specified density and height. Or if laser data is available, both positions and heights will be determined from laser points.

For point features, trees will be placed in the exact point coordinates. Height is treated as described for areas above.

If there is no geo-specific imagery available, all vegetation area features will also be automatically used for geotypically texturing the terrain surface.

Here are the supported vegetation file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Water

Vector area features generating water, cut out in the landscape and bump-mapped.

If there is no geo-specific imagery available, all water area features will also be automatically used for geotypically texturing the terrain surface.

Here are the supported water file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Wind Turbines

Vector point features generating wind turbine models.

Here are the supported wind turbine file formats:

- Esri Shape (.shp + .shx + .dbf + .prj)
- GeoPackage (.gpkg)

Geometry

Model

Ready-made 3D model representing for example high-res parts of the landscape or separate buildings.

The model needs to be defined in locally projected coordinates in meters, with its origin in the corresponding latitude, longtitude given by settings. For now, the Z axis has to be pointing upwards, X to the east and Y to the north.

Make sure that the model is shaded with valid normals so that it will react to the lighting conditions correctly. For now we highly recommend **texturing all surfaces** of a model since untextured surfaces might appear incorrect in the web viewer and the exported landscapes.

Here are the supported model file formats:

- ZIP file including the model and textures.
- Model file formats in ZIP:
  - 3D Studio (.3ds)
  - Collada (.dae)
  - Autodesk FBX (.fbx)
  - OpenFlight (.flt)
  - Wavefront OBJ (.obj)
  - OpenSceneGraph (.osg/.ive/.osga/.osgb/.osgt/.osgx, up to version 3.6.4)
  - Stanford Triangle Format (.ply)
- Texture file formats in ZIP:
  (currently only .jpg and .png will work in the View stage visualization)
  - SGI image file format (.rgb/.rgba/.int/.inta)
  - Microsoft Windows bitmap (.bmp)
  - Direct Draw Surface (.dds)
  - Graphics Interchange Format (.gif)
  - Joint Photographic Experts Group (.jpg)
  - Portable Network Graphics (.png)
  - Tagged Image File Format (.tif)

Choose files

Selects the files to be uploaded. The description below is general for all data source types and not specific to the type you may have chosen.

NOTE: It is highly recommended to upload a bit more data than needed, spatially. Since the landscape will consist of blocks whose size is calculated as a multiple of texture size and imagery resolution, the landscape will seldom cover your geographic data

completely.

For multiple files, for example the files belonging to a shape file, a non-self-contained GeoTIFF with its world and projection files, a geo-referenced JPG image with its .aux.xml geo-referencing file or both raster files and metadata vector files for imagery, make a multiple selection in the file dialog.

The Esri Shape file format requires multiple files, see below for a description of all files needed for one vector data source:

- .shp - The shape geometry.
- .shx - The spatial index.
- .dbf - The table data with fields and values.
- .prj - The projection file specifying the coordinate system.

The GeoPackage format consists of one file:

- .gpkg - The GeoPackage file with its database.

After choosing a GeoPackage .gpkg file, you will have to choose which table to use from the package. Only one table can be chosen per data source, so add another data source if you want to use more tables from one package. Only one copy of the same GeoPackage file will be uploaded.

The GeoTIFF file format can either be self-contained or require separate world and projection files, as described below:

- .tif - The raster image. Either self-contained including world offset and projection, or requiring separate files below.
- .tfw - A world file specifying offset and real world pixel size. Only required if the .tif file does not contain this information. Has the same first name as the .tif file.
- .prj - A projection file specifying the coordinate system. Has the same first name as the .tif file.

The geo-referenced JPG file format requires a separate geo-reference file as described below:

- .jpg - The raster image.
- .aux.xml - An XML file specifying the coordinate system and the geographic transformation. Has the same first name as the .jpg file.

Ready-made models have to be compressed into a ZIP file. Within the ZIP archive the following file formats are supported:

- .3ds - 3D Studio geometry file.
- .dae - Collada geometry file.
- .fbx - Autodesk FBX geometry file.
- .flt - OpenFlight geometry file.
- .obj - Wavefront OBJ geometry file.
- .osg/.ive/.osga/.osgb/.osgt/.osgx - OpenSceneGraph geometry files (up to version 3.6.4).
- .ply - Stanford Triangle Format geometry file.
- .attr - OpenFlight texture attribute file.
- .rgb/.rgba/.int/.inta - SGI image file format texture file.
- .bmp - Microsoft Windows bitmap texture file.
- .dds - Direct Draw Surface texture file.
- .gif - Graphics Interchange Format texture file.
- .jpg - Joint Photographic Experts Group texture file.

- .png - Portable Network Graphics texture file.
- .tif - Tagged Image File Format texture file.

All data sources except Models have to specify their coordinate system somehow, with a .prj file, internally in a GeoTIFF, with an .aux.xml file for JPG files or manually for LAS files. For Models, the WGS84 longitude and latitude are given manually in decimal degrees. The most suitable WGS84 UTM zone will be chosen automatically for the landscape and all data will be correctly re-projected.

Mixed Vector type

The **Mixed Vector** type is a powerful and frequently used feature that deserves a chapter of its own.

It is used for vector data with multiple types in the same file, separated by different attributes. By choosing a field in a shape file or GeoPackage and what its different values mean, you can map the values to different types, using the controls described below.

The available fields of the chosen shape file or GeoPackage are automatically fetched directly after you have chosen the files, or in the case of a GeoPackage, when you have chosen the table.

Mixed Vector Field

Sets the field of the mixed vector layer holding the values corresponding to different data source types.

When you have chosen which field that controls the type, click the **Add Type By Value** button for each type that you want to map. The controls appearing in each mapping are described below.

Mixed Vector Value

Sets the value of the chosen field, representing this type.

Mixed Vector Type

Sets the feature type tied to the selected field value. The different feature types are described in the Features sub-section under the Type section in the Configure chapter.

Settings

This chapter describes all the settings available for the different data source types.

Elevation

Raster

No settings for this type.

Laser

Coordinate Reference System

Sets the coordinate system in which chosen LAS files are represented.

Since the LAS format lacks a standard way of specifying the coordinate system, this has to be specified manually. Geographic (non-projected) coordinate systems are first in the list, projected coordinate systems last.

Open the drop-down list and start writing the name of your desired coordinate system to navigate to it faster.

Imagery

Image

Easting Field

Specifies the field in the metadata shape file or GeoPackage representing the easting coordinate of the camera/sensor position, in the coordinate system specified by the shape file or GeoPackage, or the image. This will be used to offset the imagery on building roofs to compensate for perspective, if terrain texture is used for roofs.

A metadata shape file or GeoPackage contains area features representing parts of the ortho-mosaic imagery belonging to different original images. These area features have values for the camera/sensor positions in the specified fields.

The Easting Field is filled with all available fields when a metadata shape file or GeoPackage has been chosen. It will not be used if no metadata is given.

Northing Field

Specifies the field in the metadata shape file or GeoPackage representing the northing coordinate of the camera/sensor position, in the coordinate system specified by the shape file or GeoPackage. This will be used to offset the imagery on building roofs to compensate for perspective, if terrain texture is used for roofs.

A metadata shape file or GeoPackage contains area features representing parts of the ortho-mosaic imagery belonging to different original images. These area features have

values for the camera/sensor positions in the specified fields.

The Northing Field is filled with all available fields when a metadata shape file or GeoPackage has been chosen. It will not be used if no metadata is given.

Altitude Field

Specifies the field in the metadata shape file or GeoPackage representing the altitude of the camera/sensor position, in meters above the ocean. This will be used to offset the imagery on building roofs to compensate for perspective, if terrain texture is used for roofs.

A metadata shape file or GeoPackage contains area features representing parts of the ortho-mosaic imagery belonging to different original images. These area features have values for the camera/sensor positions in the specified fields.

The Altitude Field is filled with all available fields when a metadata shape file or GeoPackage has been chosen. It will not be used if no metadata is given.

Features

Buildings

Min Floors

Sets the minimum number of building floors. If no laser data or Floors Field setting is available, building heights will be chosen randomly between given minimum and maximum number of floors. Each floor is 3 meters.

Max Floors

Sets the maximum number of building floors. If no laser data or Floors Field setting is available, building heights will be chosen randomly between given minimum and maximum number of floors. Each floor is 3 meters.

Floors Field

Specifies an optional field in the shape file or GeoPackage representing the number of floors for each building. Each floor is 3 meters.

This is only used if no laser data is available that otherwise determines the building height and the roof shape.

If neither laser data nor a floors field is available, the building height will be determined by the Min/Max Floors settings.

Roof Type

Sets the type of roof used if no laser data is available. The different types are explained below:

Flat		
Gable		
Hip		
Gambrel		
Mansard		
Flat gambrel		
Flat mansard		
Mixed	Random mix of the above	

Note that for complex building footprints these types of roofs may not be possible and a flat roof will be created instead. The specified roof type will not be used if laser data is available, in which case the laser point cloud will decide the roof type instead.

Roof Texture

Sets the type of roof texture used.

- **Random:** Applies specific roof textures chosen randomly between tiles (sloping roofs), metal (flat and sloping roofs) and bitumen (flat roofs).
- **Terrain:** Reuses the terrain texture beneath the building, including its detail texture. This requires good correlation between building vector data and imagery. Specifying image easting, northing and altitude fields will offset the imagery to fit the roofs in a better way. The result might have a bad impact on performance and stability in large landscapes since it uses high-res terrain textures with a larger LOD range than the terrain itself, but it can also give a very natural and varied appearance.

Random Facades

Enables the use of randomly chosen facade textures. If disabled, facades will be gray.

Geotypical

Land Cover

Sets the type of land cover represented by this type. Areas not covered by geotypical features will be textured with a generic geotypical texture.

If water or vegetation data sources are available, they will automatically be used for geotypical texturing as well, if there is no geo-specific imagery. Therefore they are not separately configurable here.

Note that the Urban type is not related to the Urban Areas data source type, which is assumed to use a lower level of detail and will only affect street lights.

- Farmland
- Glacier (ice or snow)
- Grassland
- Marsh firm
- Marsh wet
- Rock (mountain or other bare rock)
- Sand (desert, beach or dunes)
- Urban (asphalt or concrete)

Masts

Mast Height

Sets the height of the generated masts or communication towers (low, medium or high).

Power Lines

Type

Sets the type of power line generated. All types will have three wires between the pylons.

- Low voltage (small)
- Medium voltage (medium)
- High voltage (large)

Railroads

Track Gauge

Sets the track gauge, that is the distance in millimeters between the inside surfaces of the rails. The default is 1435mm which is the most widely used track gauge around the world.

Bridge

(experimental)

Specifies if the railroad features represent bridge overpass, underpass or elevated. If this is set to Overpass, the features will be built as bridges. If it is set to Underpass, any roads and railroads above the underpass will be built as bridges. If it is set to Elevated, it will be relatively elevated over the terrain, meant for longer elevated railroads.

NOTE: It is generally advised against making underpass railroads electrified, since the overhead lines might cross the bridge in an unnatural way.

Electrified

Enables the creation of overhead lines following the railroad.

NOTE: It is generally advised against making underpass railroads electrified, since the overhead lines might cross the bridge in an unnatural way.

Double-track

Enables double-track railroad. The distance between the two tracks is linearly scaled with the track gauge, with 4.5m for a standard track gauge of 1435mm.

Roads

Type

Sets the type of road generated:

- Asphalt (can use center and edge lines)
- Gravel (will not use center and edge lines)
- None (will use terrain imagery, preferred for high-res terrain imagery)

Width

Sets the width of the generated roads in meters. The roads will be flattened laterally along this width and textured according to the given type.

Bridge

(experimental)

Specifies if the road features represent bridge overpass, underpass or elevated. If this is set to Overpass, the features will be built as bridges. If it is set to Underpass, any roads and railroads above the underpass will be built as bridges. If it is set to Elevated, it will be relatively elevated over the terrain, meant for longer elevated roads.

Center Lines

Specifies the type of road center lines, if any. This will only be used for asphalt roads.

- None (no center lines)
- Solid (solid straight line in the middle of the road)
- Broken (broken/dashed lines in the middle of the road)

Edge Lines

Specifies the type of road edge lines, if any. This will only be used for asphalt roads.

- None (no edge lines)
- Solid (solid straight lines in the left and right edge of the road)
- Broken (broken/dashed lines in the left and right edge of the road)

Street Lights

Enables and sets the height of street lights along the roads (none, low, medium, high or auto).

Default is auto, which means that sparse high street lights are placed along wide asphalt roads, dense and low street lights are placed along narrow asphalt roads and no street lights are placed along gravel roads.

If Urban Areas are configured and uploaded, they will make sure that street lights configured this way will only be placed within built-up areas. This way you will not have to split the road features into parts with street lights and parts without. Note that the urban areas will not affect street lights configured from point features with the Street Lights type.

Staggered Street Lights

Enables the use of bilaterally staggered street lights, alternating from side to side of the road.

Street Lights

Height

Sets the height of the generated street lights (low, medium or high).

Direction Field

Specifies an optional field in the shape file or GeoPackage representing the direction of each street light. The settings below determine how the direction is interpreted.

If no direction field is present, street lights will be rotated to face the nearest road, if there are road lines available.

Direction Units

Specifies the units (degrees or radians) that the direction field is given in.

North Direction

Specifies the direction that corresponds to north in the direction field, given in the chosen units. So if 0 degrees means north, this setting should be 0. Or if PI/2 radians means north, it can be set to 1.570796.

Counter-clockwise

Specifies if the direction field contains directions in a counter-clockwise order or not.

Urban Areas

No settings for this type.

Vegetation

Type

Sets the type of vegetation generated. Exactly what types of trees that are selected depends on in which **continent and biome** your landscape is located. The currently supported vegetation types are:

- Mixed trees (coniferous and deciduous combined)
- Coniferous (for example pine, spruce or cypress)
- Deciduous (for example birch, acacia or palm tree)

Density

Sets the density of the generated vegetation (sparse, medium or dense). The different densities are relative to the type of vegetation.

The density setting is only used for area features and if laser data is not available.

Height

Sets the height of the generated vegetation (low, medium or high). The different heights are relative to the type of vegetation.

The height setting is only used if laser data is not available.

Water

Watercourse Width

Sets the width in meters of generated watercourses. This will only be used for line features.

Wind Turbines

Size

Sets the size of the generated wind turbines (small, medium or large).

Random Direction

Enables random direction of the generated wind turbines. If disabled, direction will be to the west.

Geometry

Model

Latitude

Sets the latitude (northing) in WGS84 decimal degrees, corresponding to y=0 in the model local coordinates.

The latitude has positive values for the northern hemisphere and negative for the southern hemisphere. Valid numbers are from -90 to 90.

Longitude

Sets the longitude (easting) in WGS84 decimal degrees, corresponding to x=0 in the model local coordinates.

The longitude has positive values for the eastern hemisphere and negative for the western hemisphere. Valid numbers are from -180 to 180.

Altitude

Sets the altitude in meters above the ocean, corresponding to z=0 in the model local coordinates.

If not given, altitude will be measured from the terrain below.

Set to 0 if the model has absolute altitude.

Stitch

Specifies if the model has a continuous footprint to which terrain can be stitched.

If the model should be stitched in the landscape, make sure it has a surrounding border of naked edges and that it is not closed with a bottom. This is what is referred to as a continuous footprint above.

If stitch is disabled, the model will be placed on top of the terrain which will be pushed down to prevent it from bleeding through the model.

Units

Specifies the units in which an imported model is defined. Default is meters.

Note that any units specified within the model are ignored, these are often incorrect and can not be trusted.

Map View

The map view lets you view your uploaded and converted data while generating and when completed.

The view is navigated by panning with the left mouse button and zooming with the mouse wheel or by clicking the + and - buttons.

Below is a legend of the map symbols used:


Coniferous vegetation


Deciduous vegetation


Mixed trees vegetation


Buildings


Water


Watercourses


Roads


Road bridges


Railroads


Railroad bridges


Power lines


Masts


Wind turbines


Geoypical farmland


Geotypical urban

Geotypical grassland


Geotypical marsh, firm


Geotypical marsh, wet

Geotypical sand (desert, beach or dunes)

Geotypical rock (mountain or other bare rock)

Geotypical glacier (ice)

The landscape coordinate system is presented in the bottom left corner.

The home button zooms to the total extent of the map.

With the layers button you can choose whether to show the elevation or imagery raster as base map and toggle vector layers for the different data sources on and off. You can also choose to toggle the grid on and off.

Vector data source layers are active at different zoom levels, to avoid cluttering the map. When zoomed further away from the map, certain layers will be deactivated and grayed out in the layers list.

The grid shows the blocks that the landscape is divided into. Yellow blocks are currently generating, green blocks are completed and transparent blocks are waiting to be generated. A red block represents a fatal error, Remograph will be notified automatically and get back to you with support.

3D View

The 3D view lets you view your generated landscape, as a preview before downloading it in different file formats or using it in a web tool.

Remograph recommends **Google Chrome** as the optimal browser to ensure rendering performance in the 3D view.

NOTE: The 3D view applies the following simplifications of the landscape in order to maintain rendering performance and compensate for limitations of the browser. Future versions will let the user control the detail level.

- Textures are currently half the size side-wise, resulting in 1/4 of the pixels compared to exported landscapes. This will improve in future versions thanks to further optimization and improvements of for example compressed textures.
- Trees are presented with half the density compared to exported landscapes.
- The LOD scale is adjusted, meaning that details will be visible further away in the exported landscapes. This can also make for example trees appear in low resolution even if you are close to them, if they are near a sub-tile border.

The view is navigated by rotating with the left mouse button, panning with the right mouse button and zooming with the mouse wheel. Double-clicking on a spot in the landscape will fly smoothly to that spot. By middle-clicking on a position and dragging, the view will move accordingly so that the clicked position will follow the mouse cursor.

Or put differently in this summary:

- **Rotate:** Left mouse button
- **Pan:** Right mouse button
- **Zoom:** Mouse wheel
- **Fly to position:** Double-click
- **Drag view:** Middle mouse button

Corresponding gestures on a touch device are one-finger drag to rotate, two-finger drag to pan and pinch to zoom, while double-tapping will fly smoothly to the indicated spot.

More detailed levels will be streamed into view when closing up on an area. While loading new levels, a loading spinner in the top right corner will be visible. The performance of this streaming will depend on your bandwidth and computer.

The 3D view can be presented in full-screen with the square icon in the bottom right corner.

The sun, a clear sky and stars are visualized correctly depending on time-of-day and world location. See below how the time-of-day can be controlled. A blue haze is added for realism.

These are the supported keyboard commands:

- **Key space:** Zooms to the total extent of the landscape
- **Key 1:** Enables orbit navigation mode (default, described above)
- **Key 2:** Enables fly navigation mode (left-click rotates, WASD moves, wheel changes speed)
- **Key 3:** Enables drive navigation mode (left-click rotates, WASD moves, wheel changes speed)

- **Key +:** Moves time forward one hour
- **Key -:** Moves time backwards one hour
- **Key Shift +:** Moves time forward 10 minutes
- **Key Shift -:** Moves time backwards 10 minutes

Export

To export a landscape, click the Export button in the top-right corner of the project page when it is in the View stage.

The page will switch to show the different export settings, which are described below.

There are 3 exports included in each project. Exporting a fourth time will require a new payment, unlocking another 3 exports.

These are the currently supported export formats:

- 3D Tiles (Cesium, Unreal Engine, Unity, etc.)
- MAK MetaFlight (VR-Vantage, VR-Forces)
- MetaFlight
- OpenFlight
- OpenSceneGraph

More export formats will be added in upcoming versions of Remoscape.

Export Format

Sets the file format used to export the landscape.

Each format has different settings appearing to the right when choosing the format. **It is highly recommended to learn and choose these settings wisely before pressing the Export button.**

These are the currently supported export formats (more will be added, please contact support@remograph.com for requests):

3D Tiles

3D Tiles is a format created and supported by the Cesium geospatial platform. The format has become an OGC (Open Geospatial Consortium) standard and it is supported in for example CesiumJS, Unreal Engine, Unity, NVIDIA Omniverse and Carmenta.

Make sure to read the instructions in the **README.txt** file in the exported folder.

The coordinate system is always geocentric for the 3D Tiles export, but you can choose the vertical datum that defines how to interpret the current elevations in the conversion to geocentric coordinates.

NOTE: <Name> used below refers to the project name, simplified with hyphens instead of spaces and without any special characters.

When exported, the zip file can be uploaded directly to **Cesium ion**. Just select the zip file, set **What kind of data is this?** to **3D Tiles** and **Main JSON File** to **<Name>_3DTiles/tileset.json**.

You can use the code provided in **cesium.js** (included in the zipped export folder) in **Cesium Sandcastle** to clip the Cesium terrain around your landscape and place the camera for an overview of your landscape. Refer to the comments in **cesium.js** for how to modify the script with your access token and asset ID. You can reach Cesium

Sandcastle from the **Open complete code example** in **Cesium ion**.

To use the 3D Tiles export in **Unreal Engine**, first download and install the Cesium for Unreal plugin from the UE Marketplace and follow the instructions and learn the basics from the Cesium for Unreal learning center. Then unzip the downloaded export in a suitable folder, create a **Blank 3D Tiles Tileset** in your Cesium for Unreal project, set its **Source** to **From URL** and set the **URL** to a file URL pointing to your tileset file, for example **file:///C:/Users/yourname/Downloads/<Name>_3DTiles/tileset.json**

If you experience strange flickering in Unreal Engine, this is a known issue with certain versions of the Cesium for Unreal plugin and can be fixed by enabling **Render CustomDepth Pass** on the tileset under **Rendering -> Advanced**. Alternatively, **Occlusion Culling** can be disabled in **Project Settings**.

To use the 3D Tiles export in **Unity**, first install and start using the Cesium for Unity plugin by following the Cesium for Unity Quickstart on the Cesium website, if you have not already. Then add a **Blank 3D Tileset**, set its **Tileset Source** to **From URL** and set the URL to a local path to your tileset, for example
C:/Users/yourname/Downloads/<Name>_3DTiles/tileset.json

Note that the 3D Tiles export does not yet support lightmaps, so street lights will not illuminate the surrounding landscape.

MAK MetaFlight

This is the recommended export format to use with products from MAK Technologies, for example VR-Vantage. It uses the MetaFlight format with some specific settings and it will also write control files used by MAK products. Make sure to read the instructions in the **README.txt** file in the exported folder.

NOTE: <Name> used below refers to the project name, simplified with hyphens instead of spaces and without any special characters.

The created MAK files are:
(note the difference between **.mft** for MetaFlight and **.mtf** for MAK Terrain Format)

- Specific MetaFlight **.mft** files prepared to work with compiled **.medf** files (see below).
- **<Name>_MAK_MetaFlight.mtf** referring these **.mft** files and setting various optimization parameters.
- **<Name>_MAK_MetaFlight_vantage2.mtf** for backwards compatibility with older VR-Vantage 2.x, pointing to the **userData** folder instead of the new **SharedData** folder.
- **<Name>_MAK_MetaFlight.osrx** defining an automatic initial view for the landscape. The reason for this specific initial view is that the default initial view in Vantage might put the camera at the edge of the landscape, looking away from it.

The following MetaFlight export settings are implicitly set for the MAK MetaFlight export format, to optimize the appearance in for example VR-Vantage:

- **Detail Textures:** Modulate
- **Texture Compression:** off (will be compressed at conversion later)
- **Lightmap:** on
- **Water Mask:** off (using water gloss map instead)
- **Water Gloss Map:** on

When exported and downloaded, perform the following steps (also available in a **README.txt** file in the exported folder):

1. Extract the exported **<Name>_MAK_MetaFlight.zip** archive.
2. Convert the export to MAK's optimized **.medf** format (**medfTool.exe** is normally located in the **bin64** folder of the installed MAK folder):
   `medfTool.exe -p <num_processes> -m jpg -m png -m dds -x flt --directory <extracted_folder>`
3. Move the extracted and converted folder to the **SharedData/<version>/latest/TerrainData/Terrain** folder for VR-Vantage 3.x, or the **userData/terrains** folder in the MAK product folder for older VR-Vantage 2.x.
4. In VR-Vantage 3.x it might be convenient to also move **<Name>_MetaFlight.mtf** to the **SharedData/<version>/latest/TerrainData/TerrainConfiguration** folder.
5. Load **<Name>_MAK_MetaFlight.mtf** (or **<Name>_MetaFlight_vantage2.mtf** for 2.x) in **VR-Vantage.**

Or if you want to load the landscape into a larger terrain, for example the MAK Earth terrain, replace step 5 with using the **Add Terrain Patch** feature to load all the medf-specific MetaFlight files. Load **terrain_medf.mft** separately first, with **Use Indirect Rendering** disabled. Then load all the other ***_medf.mft** files with **Use Indirect Rendering** enabled.

In **<Name>_MAK_MetaFlight.mtf** there is a **maxPagedLODs** parameter in the **myOptionalParameters** tag. This parameter is set to 0 as default from Remoscape, which means that tiles will be paged out when they are no longer within the view frustum. Depending on the hardware, this parameter can be raised to a higher value for an improved experience.

MetaFlight

MetaFlight is a high-level format arranging OpenFlight files geographically using XML files. The exported folder will contain at least a terrain.mft file, using an XML format file to define the landscape Tiles will be arranged in sub-folders.

There will also be separate .mft files for different types of features in the landscape, for example buildings.mft and vegetation.mft.

OpenFlight

OpenFlight has been a de-facto standard in the visual simulation industry for decades. It is mainly used as a file format for 3D modeling tools, for example Remo 3D, but also as an exchange format between different systems. Its verbosity and general design does not make it suitable for runtime loading, so this export is mainly for developers who want to apply custom conversion to their own formats.

You can choose to enable different masks, texture maps and instanced geometry in the exported landscape, but this will require custom development to enable support in the target runtime rendering software or image generator.

The OpenFlight export also contains the lower levels of detail produced recursively by merging and decimating 2-by-2 blocks down to one lowest level-of-detail. However, custom development is required to refer these properly in the target runtime rendering software. The master file is called **master.flt** and the LOD nodes referring the next level are named according to the file they are referring, recursively through the hierarchy and in the absence of paged level-of-detail nodes.

Separate block files with features like vegetation, power lines and street lights are referred in the same way from LOD nodes named according to the feature files from the lower level-of-detail file.

The landscape is located according to the coordinate system and offset given by the included **georef.txt** file.

OpenSceneGraph

Many visual simulation systems and image generators are based on OpenSceneGraph. This export will use either the binary OSGB format, the ASCII OSG format or the binary IVE format, all compatible with OpenSceneGraph 3.6.4 and later.

You can choose to enable a water mask and to replace the static water bump-map with a dynamic video bump-map, but this will require custom development to enable support in the target runtime rendering software or image generator. Optimized instanced geometry is enabled by default, since it is self-contained and will work without custom development.

The OpenSceneGraph export is self-contained with respect to the lower levels of detail produced recursively by merging and decimating 2-by-2 blocks down to one lowest level-of-detail, since these are recursively referred as PagedLod nodes from the .osgb/.osg/.ive files. This is useful and often necessary to keep good rendering performance in large landscapes. It is also used in Remoscape's 3D view of the landscape. The master file referring the higher levels of detail recursively is **master.osgb/.osg/.ive**. Load only master.osgb/.osg/.ive in your OpenSceneGraph application to view the complete landscape.

The landscape is located according to the coordinate system and offset given by the included **georef.txt** file. It also has descriptions on the top node including a WKT representation of the coordinate system and the offset in X and Y:

```
"WKT Coordinate System : PROJCS[...]"
"X Origin : <easting_offset_in_meters>"
"Y Origin : <northing_offset_in_meters>"
```

To improve rendering performance, texture compression is enabled by default in the OpenSceneGraph export. This will convert terrain textures to DXT-compressed .dds files. The DXT compression (or actually DXTC as part of the S3 texture compression standard) allows for decompressing on the GPU which is much more effective than on the CPU. By disabling the "Enable texture compression" setting, the terrain textures will be stored as .jpg files instead.

In the top-level master.osgb/.osg/.ive file there is a group node called "static" which contains one infinitesimally small triangle per static texture. Static textures are applied to for example vegetation or buildings and re-used throughout the landscape. This static node will help the your OpenSceneGraph viewer to load these static textures once and for all at startup, not having to upload them to the GPU again for each time they are needed. To make this work you will have to use the following code in your OpenSceneGraph viewer to setup caching of images and sharing of textures, before loading master.osgb/.osg/ive:

```
osgDB::ReaderWriter::Options *options = osgDB::Registry::instance()->getOptions();
if (!options) options = new osgDB::ReaderWriter::Options();
options->setObjectCacheHint(osgDB::Options::CACHE_IMAGES);
osgDB::Registry::instance()->setOptions(options);
osgDB::SharedStateManager *sharedStateManager = new osgDB::SharedStateManager(
  osgDB::SharedStateManager::SHARE_STATIC_TEXTURES |
osgDB::SharedStateManager::SHARE_UNSPECIFIED_TEXTURES
);
osgDB::Registry::instance()->setSharedStateManager(sharedStateManager);
```

Then make sure to start sharing textures after loading master.osgb/.osg/.ive:

```
sharedStateManager->share(node);
```

Finally, make sure to avoid optimizing textures further in any optimization as this will cause problems for the SharedStateManager. So if you are optimizing, use the following options:

```
osgUtil::Optimizer optimizer;
optimizer.optimize(node,
   osgUtil::Optimizer::DEFAULT_OPTIMIZATIONS &
   ~osgUtil::Optimizer::OPTIMIZE_TEXTURE_SETTINGS
);
```

Start Export

Click this button to start the export process according to your specified export settings. When exported, you will receive an e-mail and the exported landscape will be available for download below the map view.

There are 3 exports included in each project. Exporting a fourth time will require a new payment, unlocking another 3 exports.

Export Settings

Here are descriptions of all export settings. Some of them are specific to one export format while others are used for multiple formats.

Vertical Datum

Specifies the vertical datum, that is if the elevation from raster or laser is defined as height above the WGS84 ellipsoid or above the EGM84, EGM96 or EGM2008 geoid.

NOTE: This is only used if Coordinate System is set to Geocentric, which is always the case for the 3D Tiles export.

If elevations are originally from for example DTED or SRTM, they are heights above the EGM96 geoid.

Since EGM96 is a common vertical datum, this is the default for all exports except MAK MetaFlight where Ellipsoid WGS84 often works better.

Duplicates

This setting controls how frequently used and duplicated geometry is exported, for example vegetation, power line pylons and street lights. For MetaFlight and MAK MetaFlight exports, externals are always used.

- **Instanced:** Optimized instances allowing for rendering of a large amount of objects. **(default for 3D Tiles and OpenSceneGraph)**.

- **Cloned:** Copied inline geometry. For vegetation, only low resolution is used due to the vast amount of objects, appearing as vertical alpha-textured crossing polygons. **(default for OpenFlight)**.

- **External:** External references pointing to separate OpenFlight files. **(not available for 3D Tiles and OpenSceneGraph)**.

Instanced support

The instanced support is differently implemented depending on export format:

3D Tiles

For 3D Tiles, instanced geometry is stored using the I3DM format, so the target runtime system needs to have support for this. There are systems supporting 3D Tiles but not I3DM and for these you will have to use Cloned instead.

OpenSceneGraph

For OpenSceneGraph, the instanced geometry is self-contained and does not need any custom development. It can however be disabled if your image generator does not support OpenGL 4 or instanced geometry.

OpenFlight

For OpenFlight, being an export format used for further development, instancing is represented in two different ways that can both be devloped further with. One representation stores instance transformations in JSON files while the other uses PNG textures. Common to the two representations is that there is one group node per sub-tile and type, containing the geometry template for that type placed in the origin. If applicable there are two LOD nodes with low and high levels there. The instance transformations are stored in files named according to the first part of the group name, for example `block1_0_removeg_maple`. The rest of the group name, for example `block1_0_removeg_maple_tile2_1_1770-2925` specifies that it represents sub-tile 2,1 (horizontal index 2, vertical 1, both starting at 0 from the lower left sub-tile). It also points to the interval from zero-based index 1770 up to and including index 2925 in the JSON file or PNG texture.

JSON

Either you read the corresponding JSON files (for example `block1_0_removeg_maple.json`) and apply the transformations accordingly while implementing the instancing support of your target runtime system. The transformations are stored in the simplest JSON format possible, as arrays of 5 numbers (x,y,z, rotation and scale). This choice is a compromise between ease of parsing and optimal storage. The rotation is given as counter-clockwise degrees.

PNG

Or you choose a slightly more complicated but also more self-contained solution with an included GLSL shader, two PNG textures (on multi-texture layers 5 and 6) per type and block storing instance transformations and a number of uniforms noted in OpenFlight comments.

Transformation textures come in pairs per block and type, to be able to use 8 bytes for encoding of x,y,z translation, rotation around the z (vertical) axis and scale. Since shaders and textures are applied to OpenFlight polygons and can not be applied to group nodes, node comments are used instead to specify shaders, number of instances, samplers and uniforms. This is a description of the OpenFlight node comments:

```
<vertex shader>
<fragment shader>
<number of instances>
<base texture sampler2d uniform>
<transformation texture 1 sampler2d uniform>
<transformation texture 2 sampler2d uniform>
<sub-tile offset into the transformation textures int uniform>
<transformation texture size (width = height) int uniform>
<sub-tile bounding box minimum corner vec3 uniform>
<sub-tile bounding box maximum corner vec3 uniform>
<lowres flag uniform>
```

For example:

```
instances.vert
instances.frag
40
sampler2D baseTexture 0
sampler2D coordsBuffer1 5 block1_0_removeg_maple_1.png
sampler2D coordsBuffer2 6 block1_0_removeg_maple_2.png
int offset 327
int bufferSize 32
vec3 bboxMin -1027.7211880174 -1022.8984899896 0
vec3 bboxMax -447.14532484507 -414.88701091792 18.730845367146
int lowres 0
```

For vegetation, each instance has a low and a high level of detail (LOD). The disabled (0) lowres flag at the top level as included in the list above is overridden as enabled (1) in the low-res LOD child in a `int lowres 1` comment, also meant to be translated to a uniform when loading or converting. For other instances that do not have low and high levels, lowres is always 0.

The included GLSL shaders are written for OpenGL 4.2 and assume availability of variables like `gl_InstanceID, gl_ModelViewProjectionMatrix, gl_MultiTexCoord0, gl_LightSource, gl_TexCoord`, etc.

Alpha Mode

The 3D Tiles export can use the GLTF alpha modes Mask or Blend when creating semi-transparent textures. Default is Mask and if your target system has problems with transparency you can try Blend instead.

Coordinate System

Specifies the coordinate system used in the exported landscape:

- **UTM:** Universal Transverse Mercator projection in a suitable zone on the WGS84 ellipsoid with a suitable specified offset specified in OpenFlight and MetaFlight files.
- **Geocentric:** ECEF (Earth-Centered, Earth-Fixed) coordinate system with its origin in the Earth's center of mass, X axis through longitude 0 degrees at the equator, Y axis through longitude 90 degrees at the equator and Z axis through the north pole.

Detail textures

Sets the type of detail textures used in the exported landscape. Detail textures are repeated multi-textures mapped on layer 1 with different detailed appearance depending on if the polygon represents for example terrain or a road.

The default detail texture type visible in the 3D view is Add, using the ADD texture environment type. This means that the detail textures are dark with low values which are added to the base texture.

Choose Modulate if your target runtime rendering software doesn't support the ADD texture environment type. This will instead use the MODULATE texture environment type which means that the detail textures are bright with high values which are multiplied with the base texture.

Texture Compression

Enables DXT compression (or actually DXTC as part of the S3 texture compression standard) on terrain textures, stored in .dds image files. This will improve the rendering performance of the landscape, since the decompression is performed on the GPU. If disabled, terrain textures will be stored as .jpg image files.

Note that this will require support for loading DXT compressed .dds files in the software used to visualize your landscape.

None of this applies to textures for vegetation, buildings or other models and features.

Lightmap

Enables the use of a texture-mapped lightmap texture in the exported landscape. This is used to visualize areas lit by for example street lights.

The geo-specific lightmap for a block file is a JPG texture mapped on multi-texture layer 4 of the terrain polygons, or specially treated for MAK output as described below. It is named according to the block name with a _lightmap suffix, for example block0_0_lightmap.jpg. These lightmaps are also mapped on building facades.

The lightmap is black where there is no light and has the light color (usually white or yellow) in lit areas. A pixel shader can mix the base texture on layer 0 with this lightmap from layer 4 depending on time-of-day, sun elevation or ambient level.

For the MAK MetaFlight export, the lightmap will be replaced with a small white texture while the real lightmap will be written to a corresponding image with the _LMM suffix, to support lightmaps mapped together with both specific terrain textures and repeated road textures. The lightmap multi-texture will also be moved from layer 4 to layer 2, in order to avoid colliding with internally used multi-textures.

Make sure that the software used to visualize your landscape supports the use of these lightmaps before enabling this export setting! Otherwise it may always be mixed with the base texture, resulting in a very dark appearance even during daytime.

Water Mask

Enables the use of a water mask in the exported landscape. A water mask can be used by shaders to know where for example reflections, waves etc. should be rendered.

The water mask for a block file is a JPG texture mapped on multi-texture layer 2 of the terrain polygons. It is named according to the block name with a _water suffix, for example block0_0_water.jpg.

The water mask has white pixels where there is water and black pixels where there is no water. To accomplish a smooth appearance without aliasing, beaches are gray. A pixel shader can mix a water appearance of choice with the base texture on layer 0 depending on the mask value in layer 2.

Make sure that the software used to visualize your landscape supports the use of this water mask before enabling this export setting! It will require some custom development, otherwise it may always be mixed with the base texture, resulting in black terrain.

Water Gloss Map

Enables the export of a separate water gloss map for each terrain texture in the landscape. A gloss map can be used by the visualization software to determine glossy reflective surfaces, for example water reflecting the sky.

A water gloss map is a grayscale JPG texture with the same name as its corresponding terrain texture but with a _GLS suffix, for example block0_0_L3_X0_Y0_GLS.jpg. It has white pixels where there is water and black pixels where there is no water. To accomplish a smooth appearance without aliasing, beaches are gray.

Make sure that the software used to visualize your landscape supports the use of a gloss map with this naming convention. It will not do any harm if not supported though, since it is not mapped on to any terrain.

Water Bump-map

Specifies the type of water bump-map texture used in the exported landscape. This is used to accomplish a static or dynamic wave pattern in the water, reflecting light in the waves even if the geometry is flat.

The water bump-map texture is one repeatable JPG texture called water_bump.jpg (static) or one repeatable MP4 video called water_bump.mp4, mapped on multi-texture layer 3 of the separate water polygons cut out in the terrain in the best level of detail.

The bump-map is actually a **normal-map**, encoding pixel normals in its red, green and blue pixel values. A pixel shader can use these normals from layer 3 to calculate reflections, possibly with a smooth transition from a straight normal depending on the distance, to avoid a repeating pattern.

In the OpenSceneGraph export, an osgFX::BumpMapping node is created, reacting to the light source number 0. This can of course be modified by custom development. Note that for a dynamic bump-map, the ImageStream created by loading the MP4 video has to be started with the play method in the viewer application.

Make sure that the software used to visualize your landscape supports the use of the normal-map, and in the dynamic case the .mp4 file format, before setting anything else than None for this export setting! It may require some custom development, otherwise it may always be mixed with the base texture, resulting in incorrect colors. Or if you choose dynamic and the .mp4 file format or video textures in general are not supported at all, there may be a fatal error.

OpenSceneGraph File Format

Sets the file format used by the OpenSceneGraph export:

- **OSGB:** OpenSceneGraph extendable binary format **(default)**
- **OSG:** OpenSceneGraph ASCII format
- **IVE:** OpenSceneGraph native binary format

Wrapping up

Hopefully this user's guide has prepared you for using the Remoscape cloud service.

If you have any questions, feedback or suggestions, do not hesitate to contact Remograph at remoscape@remograph.com.