

LiDAR Point Cloud Plugin Reference

Reference for LiDAR Point Cloud plugin settings and other options



Project Settings


To access the LiDAR Point Cloud plugin settings, open the **Project Settings** window (menu: **Edit > Project Settings**).

The plugin settings can be accessed from the Project Settings window.

Octree

Setting	Description
Duplicate Handling	Determines how to handle duplicate points (distances less than 0.0001). You can choose from the following three options: <ul style="list-style-type: none">• Ignore: Keeps any duplicates found.• Select First: Keeps the first point and skips any further duplicates.• Select Brighter: Selects the brightest of the duplicates
Max Distance for Duplicate	Maximum distance between two points for them to be considered a duplicate of the other.
Max Bucket Size	Maximum number of unallocated points to keep inside the node before they need to be converted into a full child node. Lower values will provide finer LOD control at the expense of system RAM and CPU time.
Node Grid Resolution	Virtual grid resolution to divide the node into. Lower values will provide finer LOD control at the expense of system RAM and CPU time.


Performance

Setting	Description
Multithreading Insertion Batch Size	Determines the maximum number of points to process in a single batch when using multi-threading.
Use Async Import	When enabled, the editor imports the point clouds in the background, without blocking the main thread.
Prioritize Active Viewport	When enabled, Unreal Engine allocates a larger portion of the available point budget to the viewport that currently has focus.
Cached Node Lifetime	The amount of time a node will be kept in RAM after it is no longer visible. Large values are more likely to avoid reloads from storage at the cost of increased RAM usage.
Release Asset After Saving	If enabled, Unreal Engine will automatically release system memory used by an asset once that asset has been saved and will use asset streaming instead. This is useful to avoid memory blockings when dealing with large datasets.
Release Asset After Cooking	If enabled, Unreal Engine will automatically release system memory used by an asset once that asset has been cooked and will use asset streaming instead. This is useful to avoid memory blockings when dealing with large datasets.
Use Render Data Smoothing	If enabled, render data generation is spread across multiple frames to avoid freezes. With this option enabled, you can also set the Render Data Smoothing Max Frametime option, which controls how much of the frame time should be spent on render data generation.
Use Fast Rendering	<div>Enabling this setting can significantly improve runtime performance (by up to 300% in some cases), but it quadruples VRAM usage. This setting is enabled by default.</div> <div><div></div><div>You should disable this option during offline video recordings with very high point budgets, where runtime performance is not as important. The increased VRAM usage can quickly saturate the GPU and eventually cause crashes.</div></div>

Collision

Setting	Description
Meshing Batch Size	Determines the size of per-thread data for the meshing algorithm.

Automation

Setting	Description
Auto Center On Import	<p>If enabled, automatically centers the cloud on import.</p> <div>  <p>Preserving original coordinates may cause noticeable precision loss if the values are too large. If you experience point 'banding' effect, re-import your cloud with centering enabled.</p> </div>
Auto Calculate Normals on Import	If enabled, automatically calculates normals after a point cloud is imported successfully.
Auto Build Collision on Import	If enabled, automatically builds collision after a point cloud is imported successfully.

Import / Export


Setting	Description
Import Scale	On import, meters are converted to Unreal Units (UU), where 1 UU = 1 cm. To use a custom import scale, change this value.
Export Scale	On export, Unreal Units are converted back to meters. To use a custom export scale, change this value.

Details Panel

Appearance

The appearance of point clouds is defined by Actors. This relationship is similar to how Static Mesh Actors represent Static Mesh Assets.

You can change the appearance of a point cloud in your Level from the **Appearance** section of the **Details Panel**. Here, you can adjust lighting, point size, rendering, collision, and a number of other settings.

Setting	Description
Point Size	<p>Changes the size of all points. Lower point densities (where a lower point budget is selected) can be somewhat mitigated by increasing this value and the Point Size Bias. If this option is set to 0, each point will use a fixed size of 1 px, disregarding camera distance and data density fluctuations.</p>
Scaling Method	<p>Determines how points in the point cloud will scale. You can choose from the following options:</p> <ul style="list-style-type: none">• Per Node: Points are scaled based on the estimated density of their containing node.• Per Node Adaptive: Similar to Per Node scaling, but density is calculated adaptively based on the current view.• Per Point: Points are scaled based on their individual calculated depth.• Fixed Screen Size: Sprites are rendered using a screen-space scaling method. <p>Hover your mouse over each of these options in the Details panel for more information about their specific use cases.</p>
Gap Filling Strength	<p>Uses a custom Material to enlarge points and render them as paraboloids to minimize visible overlap. This incurs a performance cost of around ~0.7 ms / 1 million points.</p>
Color Source	<p>Choose whether to use the colors embedded in the point cloud data or color the point cloud based on Position, Elevation or Classification.</p> <p>If you choose Elevation, you can define which colors should be used for the highest and lowest elevation. The rest of the points will be colored using a gradient between these two values.</p> <p>If you choose Classification, you can further define which colors should be used in the Classification Colors array in this section.</p>
Point Size Bias	<p>Affects how point scaling is performed per LOD. This is related to the r.LidarBaseLODImportance console variable.</p> <div><p>If the asset budget is small and the number of overall components is high, point cloud assets can disappear and pop-in. To prevent this, the LOD system actively attempts to allocate at least the minimum quality LOD for assets that are far away from the camera.</p></div>

Rendering

You can set a number of rendering-specific options from the **Rendering** section of the **Details** panel.

Setting	Description
Use Frustum Culling	When this option is enabled, points outside of the visible frustum will not be rendered. Disabling it can reduce data streaming delays when shooting cinematics.

Console Variables

Use these console variables to define other point cloud parameters.

Variable	Default Value	Description
<code>r.LidarBaseLODImportance</code>	0.1	Defines the importance of tiles in the rendering budget depending on the distance from camera. This variable controls how important it is to display at least the lowest quality tile as opposed to spending the budget on what's closest to the camera
<code>r.LidarPointBudget</code>	1000000	Defines the global point rendering budget of the viewports. This value represents the total number of points displayed on-screen at the same time.
<code>r.LidarTargetFPS</code>	59	This setting will continually tweak the point cloud budget to maintain overall fps of the specified value. If <code>r.LidarPointBudget</code> is set to a value greater than 0, target fps is disabled.
<code>r.LidarIncrementalBudget</code>		If set to true, this will automatically increase point budget to very high values as long as the viewport remains stationary. The budget will reset to the previously set value when the viewport moves.
<code>r.LidarScreenCenterImportance</code>	0	Defines the importance of tiles in the rendering budget based on where the camera is looking. A higher value will force lower LODs on the sides of the camera frustum.
<code>stat lidarpointcloud</code>		Shows the statistics of the currently rendered point cloud data.