Nature Growth System: User Manual

1. Overview

Nature Growth System (aka NGSys) provides two primary tools for building trees that can grow at runtime:

- **Tree Designer:** A mode you enter directly in the level. You draw a tree's trunk, branches, roots, and foliage in place, tune its look, and export it as a mesh, a static prop, or a fully "growable" actor.
- **Growth Baker:** A standalone window that takes any rigged tree (a Skeletal Mesh) and generates the growth-timing data and animation asset it needs to grow at runtime.

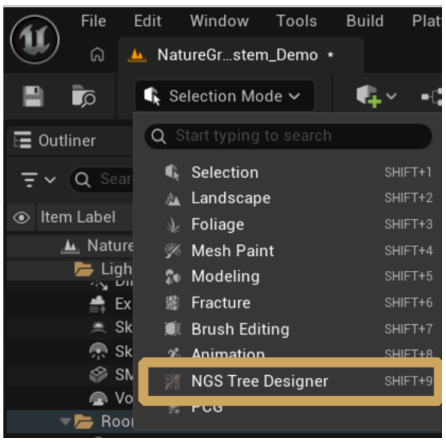
Tip: The *Growth Baker* does not care where the tree came from - all it needs is a skeleton whose bones follow the tree's structure. It should work with a Skeletal mesh exported from:

- Epic's Procedural Vegetation Editor
- Quixel MegaPlants
- Tree from Blender (as long as it's rigged properly)
- your own rig built in NGSys *Tree Designer*

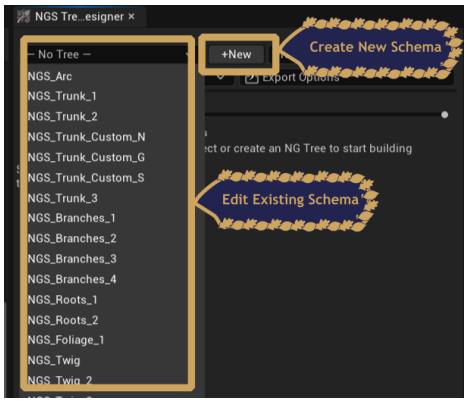
2. The Tree Design Process

A typical tree creation workflow, from start to finish:

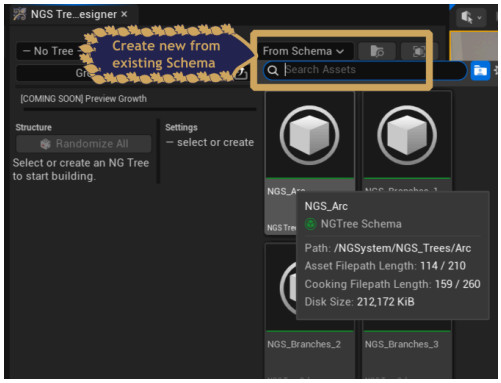
1. **Start a tree.** Open the Tree Designer mode



   either create a brand-new tree or pick one already placed in the level.



You can also spawn a tree from a previously saved design.



Note: creating a new tree based on the Existing Asset will result in copying that asset!
Alternatively you can take the existing NGTree asset and drag it to Scene.

2. **Shape the trunk.** Build the trunk using the parametric shape controls (height, lean, segment count, a light wobble) and generate it with one click:

Or hand-place its points directly in the viewport by selecting the Anchor and pressing ***N*** button you can expand the trunk or branch with the new anchor .

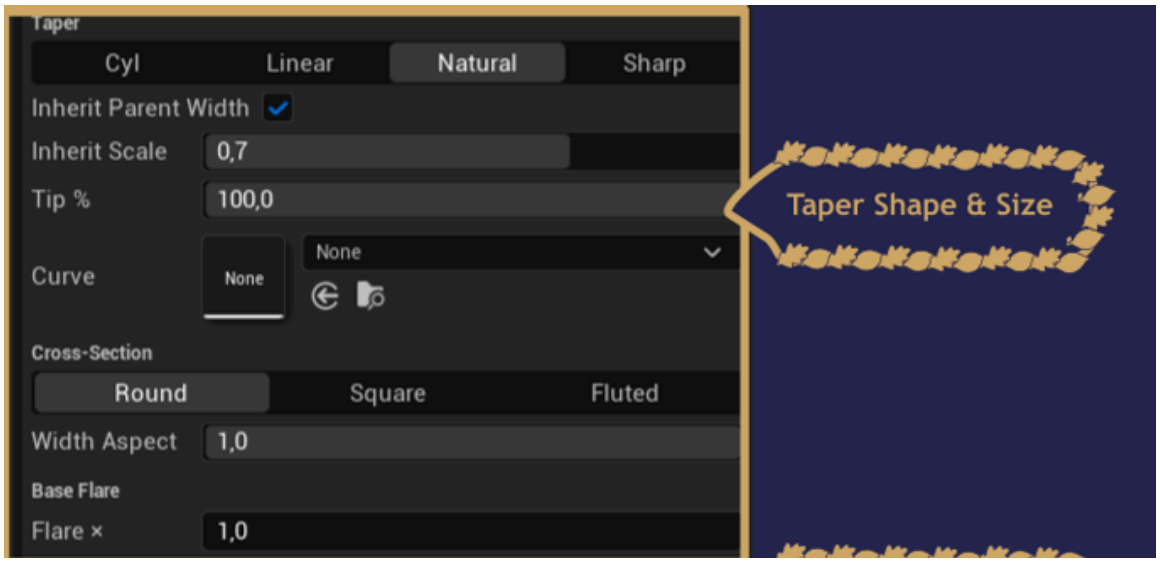
You can switch between the two methods at any time.

3. **Add canopy branches.** Working level by level (first-order branches off the trunk, second-order off those, and so on), either hand-draw individual branches or auto-populate a whole level at once from a handful of distribution settings (how many, how they spread, how much they angle upward).

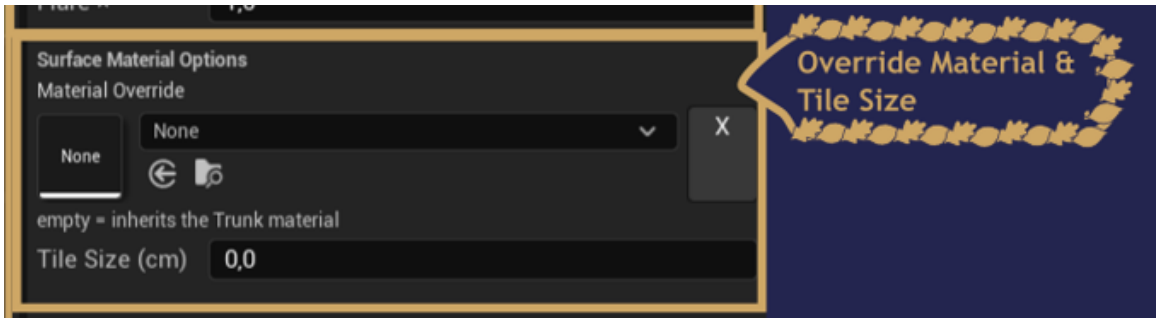
Auto-populated levels can be reshuffled at any time without disturbing anything you drew by hand.

4. **Add roots.** Work the same way as canopy branches, growing downward from the trunk instead of upward.

5. **Refine the look per level.** For the trunk and for each branch/root level, set how the wood tapers from base to tip, its cross-section shape, and its material.



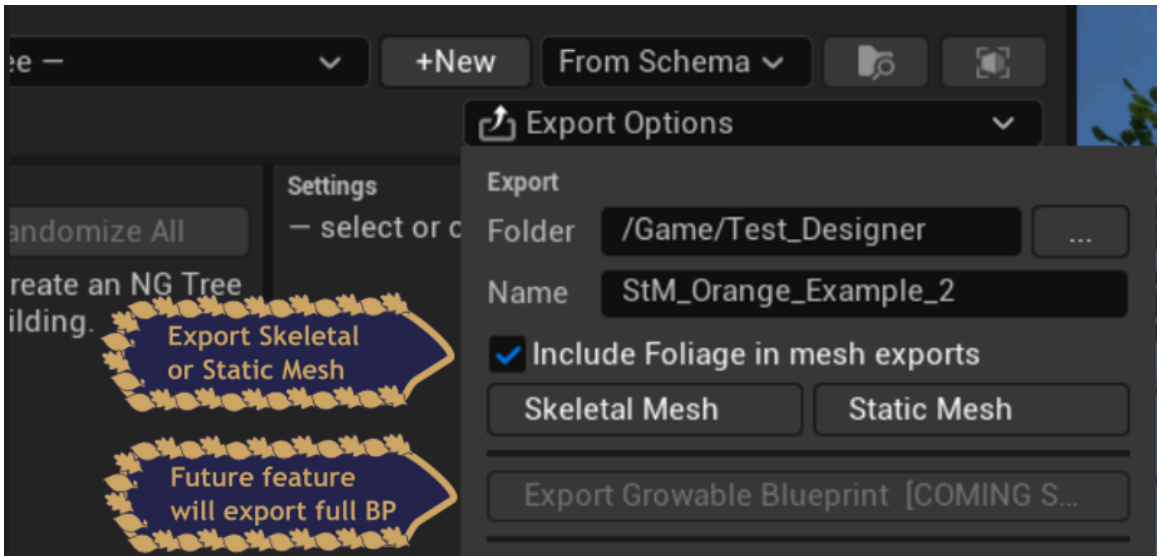
A branch level can use the trunk's material or its own.



6. **Add foliage.** Scatter leaves, fruit, or flowers onto chosen branch levels, controlling density, size variation, and where along the branch they appear.

7. **Export.** Produce whichever output you need: a skinned Skeletal Mesh (the one to use if you want the tree to grow - hand it to the Growth Baker next) or a non-animated Static

Mesh prop.



. Two core principles carry through the whole process:

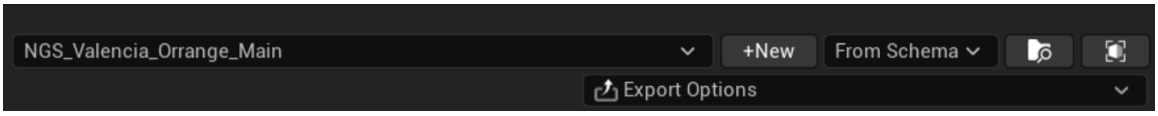
- **Nothing is destructive by accident:** Auto-populated content can always be reshuffled (Randomize) or fully regenerated (Populate) without losing anything you placed by hand.
- **The tree behind the scenes is always one file:** Its design asset acts as the core file. The actor in the level is just a live view onto it. Copying or sharing that one file shares the whole tree.

NOTE: THE NGS_TREE ASSET IS NOT SUPPOSED TO BE USED IN RUNTIME! IT'S A DESIGNER REPRESENTATION ONLY! MAKE SURE YOU EXPORT IT FIRST!

3. Tree Designer: Fields reference

3.1 Header Bar

It allows you to:

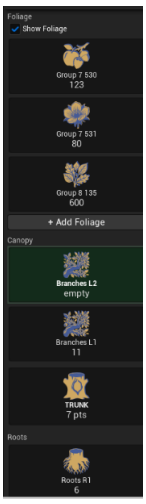


- Switch which tree in the level you're editing.
- Start a brand-new tree.
- Spawn a tree from a previously saved design (an independent copy, not a shared link).
- Jump to that design's file in the Content Browser.

- Recenter the viewport on the tree you're editing (also available as a keyboard shortcut).

3.3 Structure Toolbar

A vertical strip of tiles, one per part of the tree, arranged the way a tree actually reads: foliage groups at the top, canopy branch levels below that (highest level first), the trunk in the middle, and root levels below the trunk.



- Each tile shows how much exists there (branch count, 'empty', or 'draw' if nothing has been placed yet).
- Clicking one both selects that part of the tree and switches what the Settings Panel shows.
- A global Randomize All button above the list reshuffles every auto-populated level in the whole tree at once.

The following sections detail every setting available within the *Tree Designer*, categorized by your current selection. The tool strictly enforces the ranges and default values listed below.

3.4 Trunk

Control	Description
Stump Width (cm)	Applies an additional radius at the base to simulate root-flare. 0 disables the effect.
Height (cm)	The vertical extent of the trunk, terminating at the highest anchor point.
Segments	Defines the number of anchors constructing the trunk; this value is synced to the bone count.
Bend (dial)	A 2D drag-pad defining lean intensity and direction. Double-click to zero out.
Bend Start	Sets the height where lean begins. 0 bends from the root; 1 affects only the tip.
Jitter Anchor Locations	Adds random lateral variation per anchor, increasing toward the top. 0 is perfectly straight.

Control	Description
Seed	The reproducible seed for anchor jitter; the same value ensures the same trunk shape.
Generate Trunk	Executes or refreshes the trunk construction based on current parameters.

If the trunk has been generated solely via sliders, it will update live as you adjust values. However, once you manually modify a trunk anchor in the viewport, the tool will require confirmation before overwriting your manual edits with slider changes.

3.5 Individual Branch or Point

Control	Description
Bone Count	Sets export resampling (2–64, default 4). Higher counts improve bending smoothness at the cost of bone density.
Position	Displays local coordinates. Use viewport gizmos for interactive movement.

Individual anchors do not possess width controls; thickness is derived from the level's *Taper* settings. Select the branch itself to modify its overall form.

3.6 Branch/root Level Auto-Drawing

Note: that canopy branches and roots operate as a unified system across two different domains, sharing the same underlying controls.

These parameters are available when a branch or root level is selected. **Populate** constructs branches based on these settings while preserving hand-drawn geometry;
Randomize generates a fresh iteration using a new seed.

Control	Description
Distribution (Uniform/Spiral/Random/Curve)	Spacing logic along the parent. Spiral utilizes the golden angle to prevent branch stacking.
Count	Quantity of branches produced by the Populate command.
Seed	Ensures consistency; reshuffled via Randomize.
Attach Min/Max	The placement range on the parent; roots typically cluster at the very base.
Axil Angle (°)	Branch angle. Canopy measures from the parent direction; roots measure from straight down.

Control	Description
Tropism	Curvature toward sky (+) or ground (-). Extreme values fold branches vertically.
Length Scale	Global length multiplier for the entire level.
Length Jitter	Randomized variation in branch length relative to base scale.
Angle Jitter (°)	Random deviation applied around the base Axil Angle.
Jitter Anchor Locations	Adds random lateral variation per anchor, increasing toward the top. 0 is perfectly straight.
Density Curve	Determines branch clustering when Distribution is set to Curve.
Populate	Triggers level generation without affecting manual branches.
Randomize	Increments the seed and rebuilds the active level.

3.7 Taper & Cross-Section

Access these profile settings from the level settings; they govern the entire level's shape across trunks, canopy branches, and roots.

Control	Description
Preset (Cyl/Linear/Natural/Sharp)	Governs the base-to-tip silhouette. Natural maintains girth at the base.
Base Width (cm)	Radius at the branch start; used when inheritance is disabled.
Tip %	Tip radius relative to the base width.
Inherit Parent Width	Matches the branch base to the parent's radius at the attachment point.
Inherit Scale	Initial child thickness as a fraction of the parent width.
Curve	Optional asset to override presets with custom taper shapes.
Cross-Section (Round/Square/Fluted)	The tube's outline profile. Square simulates woody stems; Fluted adds ridges.
Width Aspect	Adjusts cross-section symmetry; lower values flatten the shape.
Flare ×	Widens the junction where the branch joins the parent.
Flare Length	Defines the transition distance from flare back to normal taper.

3.8 Surface Material

Material settings can be overridden by the level

Control	Description
Material (Trunk)	The primary material inherited by all parts of the tree.

Control	Description
Material Override	Level-specific material; use "X" to revert to trunk inheritance.
Tile Size (Trunk)	World units per tile; prevents stretching on elongated meshes.
Tile Size (Level)	Unique tile density for the level; 0 defaults to trunk settings.

When not overridden the levels will inherit the Trunk's Material settings.

3.9 Foliage Groups

Each Foliage Group represents a unique scatter of a single mesh asset. You may add multiple number of groups to a single tree design.

Control	Description
Static Mesh asset	The asset to be scattered.
Type (Foliage/Fruit/Flower)	Categorization for UI icons; does not impact scattering logic.
Count	Total number of instances to distribute.
Seed	The unique identifier for the scatter pattern.
Scale Range	Randomized uniform scale boundaries for instances.
Zone: Length	The attachment span on branches; defaults to the outer half.
Zone: Tips Only	Overrides length ranges to place items strictly at branch ends.
Align To Surface	Orients items to the branch surface; disabling this uses mesh-authored angles.
Axil Angle	Angle relative to bark: 90 is perpendicular, <90 leans forward.
Jitter Range	Random angular offset to break mechanical uniformity.
Foliage Tropism	Bends instances toward sky (+) or ground (-).
Rendering Mode (ISM / HISM)	Determines rendering method for exported actors.
Nanite Toggle	Enables Nanite on the mesh; note that this modifies the source asset.
Welding LOD Source (Source/Render)	Specifies which LOD data is utilized for mesh-weld exports.
Welding LOD Index	The target detail level for welding; higher values reduce memory footprint.
Randomize/Delete	Reshuffles or removes the specific scatter group.

Export settings are primarily relevant when foliage must be **welded** into a single mesh. If Nanite Assemblies are active, the tool ignores these during Skeletal Mesh export and attaches foliage at full detail.
Note: These parameters do not impact the viewport or growable actor exports.

It is highly advised not to export the Static Mesh with the big amount of foliage instances and welding them is a high performance task and may not succeed.
In future release Plugin will support the Export to Growable Blueprint with the ISM Foliage Component

3.10 Selection and the "Add" Key

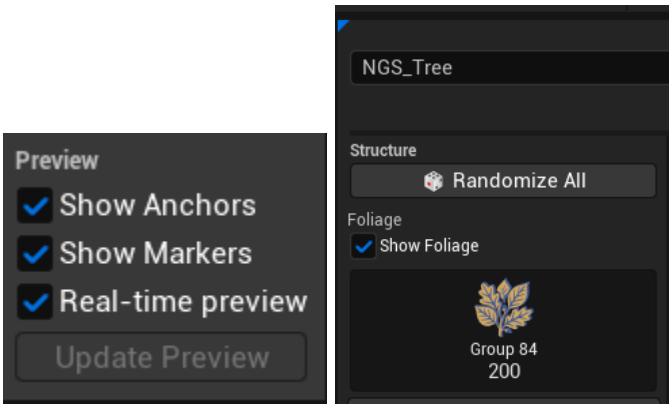
To place the Geometry one must select the Anchor point of the Trunk/Branch/Root and click "N" button.
Based on selection following will happen:

- **Tip point selected:** Extends the branch.
- **Mid-branch point selected:** Inserts a new point between two existing ones.
- **Valid parent point selected (while level is active):** Starts a brand-new branch from that point.

Deleting removes the selected point (or the whole branch, if selected); clearing the selection is one keypress.

3.11 Preview controls

Use the "Show Foliage", "Show Anchors", "Show Markers" options to clear up the view during the design process.



"Real-time preview" option allows disabling the auto update of the Tree and changes the rendering to an ondemand way through the "Update Preview" button.

4. Growth Baker: Settings & Configuration

4.1 Purpose

The ***Growth Baker*** turns a rigged tree (Skeletal Mesh) into a tree that can grow at runtime. It reads the tree's skeleton, calculates a growth order (trunk first, tips last), and produces the data and animation asset needed to drive growth from a single 0-1 value.

Skeleton Requirements: the bones must follow the tree's structure, parented from the base outward (trunk is a chain, branches hang off it).

4.2 Step-by-Step Baking

1. Provide **Skeletal Mesh**: Pick the tree to bake. (This is the only required input).
2. **Process Mesh**: Confirms the skeleton and reports bone count/depth. Meshes with fewer than two bones are rejected.
3. (*Optional*) **Load Settings** from Preset: Pick a previously baked growth preset to pre-fill settings.
4. Set Parameters & Output: Define growth parameters (see 4.3) and select an Output Folder (must start with /Game/.
5. **Bake Growth Animation**: Produces the three output assets (see 4.5).

(Settings are remembered between sessions).

4.3 Growth Parameters

Parameters are divided into two categories: Bake-time (requires a re-bake to change) and Runtime (editable anytime on the preset).

Bake-Time Settings:

Setting	Description
Growth Timing Curve	Optional. Sets whether the tree grows at a steady pace or speeds up and slows down along the way. Empty = steady, even growth. Add a curve to start slow and finish fast, or the other way round.
Per-Bone Scale Span	How long a single piece of the tree takes to reach full size. Small = pieces grow one after another, like a wave running up the tree. Large = large parts of the tree swell together.

Runtime Settings (Reusable Preset):

Setting	Description
Level Emerge Duration	How long each stage takes to appear: the trunk first, then the main branches, then the smaller ones. Small = each stage pops in quickly, one clearly after the other. Large = the stages overlap and grow together. This is how LONG a stage takes. Easing Exponent below is how it speeds up and slows down.
Easing Exponent	Whether a piece grows at a constant speed or eases in and out. 1 = constant speed 2 = gentle start and finish, looks natural (default) 4 or more = holds back, then shoots out.
Hierarchy Timing Curve	The order the tree builds itself in: trunk first, then the main branches, then the smaller ones With the default settings a half-grown tree already looks like a small complete tree, and simply gets fuller as it grows.
Branch Growth Stagger	Whether parts of the same size appear all at once or ripple outwards. 0 = they all pop in together 1 = the far ends lag behind, giving a visible wave (default)
Hierarchy Girth Multiplier Curve	Optional. Sets how thick each stage is compared with the trunk. Use it to keep a heavy trunk with fine twigs. When set, it replaces the single Falloff knob below. Empty = use Hierarchy Girth Falloff instead.
Hierarchy Girth Falloff	How much thinner the tree gets towards the twigs, as one simple knob. 1 = twigs as thick as the trunk. 0.5 = the finest parts are half the thickness of the trunk. 0.2 = very fine twigs Ignored when the curve above is set.
Unfurl Duration	How long a branch takes to swing out from its surface into final position. 0 = instant pop.
Unfurl Levels	Which branch levels receive the swing-out treatment (defaults to first-order

Setting	Description
	branches). 0 = trunk, 1 = main branches, 2 = smaller branches, etc.
Grow From Tip	Whether the outer tips lead the growth wave)or the base does. On (default) = the far end leads, so growth sweeps back towards the trunk." Off = the end nearest the trunk leads and growth sweeps outwards.

4.4 How Growth Order Is Calculated

The Baker reads the growth order directly from the mesh's skeleton:

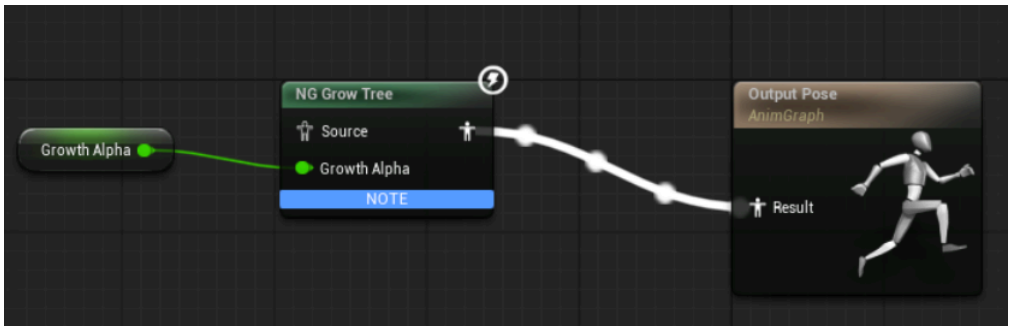
- **Timing:** Based on distance along the bone chain from the base. Parts equidistant from the base grow together.
- **Trunk Path:** Followed from the first bone. At every fork, the child best continuing the current direction stays the trunk; others become branches.

If a tree grows in a visibly wrong order (e.g., an unusual rig where the trunk stops short but a low branch goes upward), inspect the forks in the skeleton. "*Process Mesh*" allows you to sanity-check bone count before baking.

4.5 Output Assets

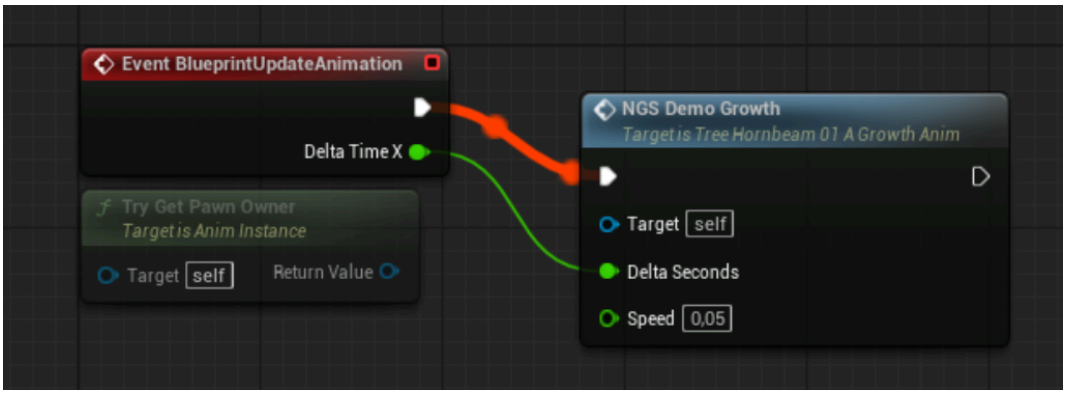
Baking produces three assets in your target folder:

1. **_GrowthPreset:** Reusable runtime settings (from 4.3), shareable across trees.
2. **_GrowthData:** The per-part growth order baked specifically for this tree, plus a record of its bake-time settings.
3. **_GrowthAnim:** An animation blueprint wired to grow the mesh using the data and preset. Drive its single growth value (0 to 1) via timeline, gameplay event, etc.



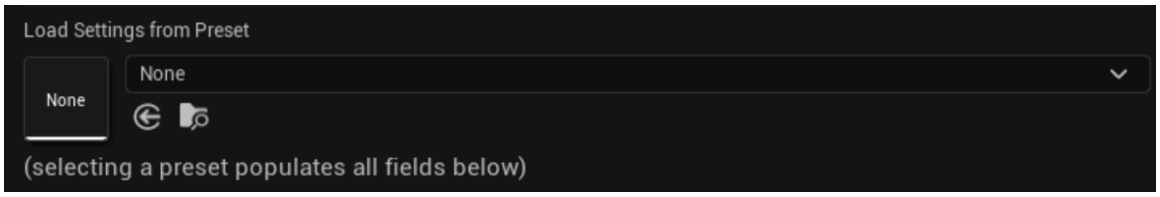
Included Demo: Every _GrowthAnim includes a ready-made demo. Open it, find NGS_DemoGrowth under Functions, drag it into Event Blueprint Update Animation, connect Delta Seconds, and set a Speed (0.15 is ~7 seconds each way). The tree will grow, reverse,

and repeat. Use this as reference for your own logic.



4.6 Convenience Features

- Load Settings from Preset allows you to populate all fields from a known-good baked preset, avoiding manual re-entry.



- All settings and output folders are automatically saved between sessions.

5. Known Limitations

Several UI controls in the current Tree Designer build are visible but inactive. These are intentional previews of upcoming features, not bugs:

- **Canopy Form (Trunk Shape settings):** This feature is being developed, thus disabled in the editor and marked as COMING SOON.
- **Export Growable Blueprint:** This feature is being developed. It will provide the ability to generate the BP with the animation hooked up to the Logic. The Demo level contains the example Blueprint of how this can be done.