

Small worlds. Real craft.

Eight stylized props went from a written brief to game-ready assets that run live in Unity. This is how each step worked.



The finished props in a Unity 6 village: cottage, crates, scooter, wheelbarrow and dwarf.

From a sentence

GPT Image 2.5 drew concept art, Meshy 7.1 built a detailed 3D model from each picture, and Claude Opus 5.5 3D rebuilt the meshes for a game.

Into a running scene

Parts were split so they could move, then the props were rendered in Unity 6 and assembled into a village with a generated sky, textures and trees.

Start with a sentence.

Every prop began as a written brief. A picture came first, then a heavy, detailed 3D model that keeps everything the picture shows.



Concept images for five of the props, drawn by GPT Image 2.5: cottage, windmill, scooter, helmet and wheelbarrow.

Concept art

GPT Image 2.5 drew each prop from a written brief. The set is a cottage, a windmill, a scooter, a wheelbarrow, a crate, a miner's helmet, a pickaxe and a dwarf.

A 3D model per picture

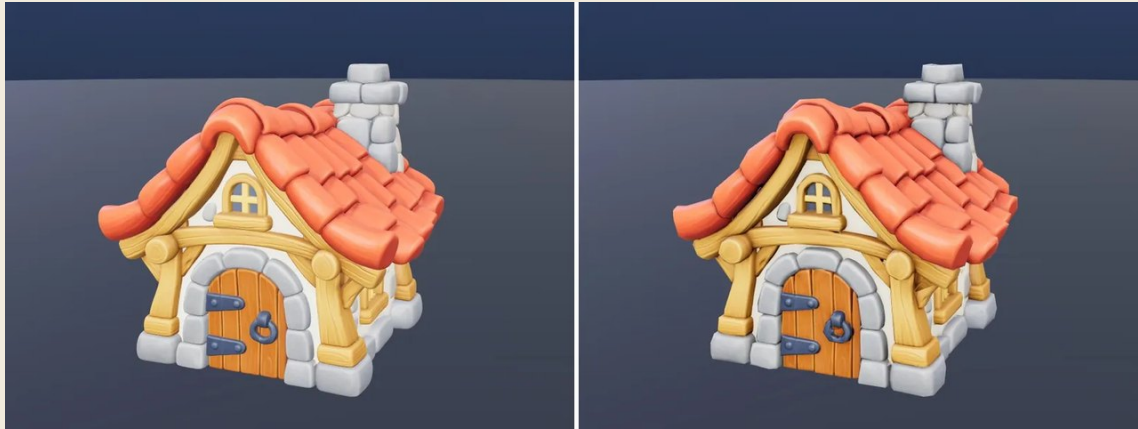
Meshy 7.1 turned each approved image into a textured 3D model with 4k geometry, 4k PBR textures and remeshing switched off, so every detail survived.

The price of detail

Detail is weight. The cottage came back from Meshy 7.1 at 834,978 triangles and the windmill at 961,272, far too heavy for a game scene.

Same cottage, 167 times lighter.

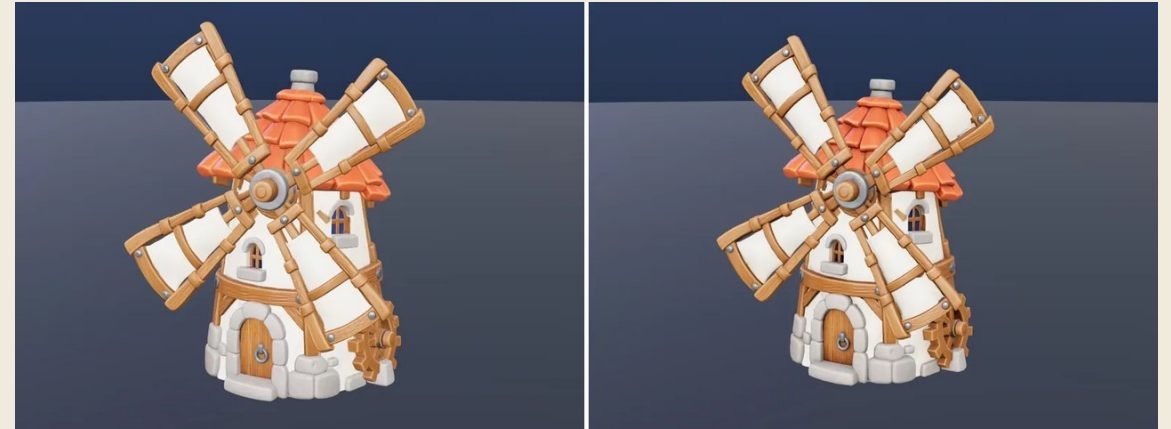
Claude Opus 5.5 3D rebuilt each mesh from a written brief instead of decimating it. Left: the original Meshy 7.1 mesh. Right: the optimized mesh.



Cottage: 834,978 triangles to 4,990.

What the cottage brief asked for

Preserve the silhouette, the curved roof edge, the large roof tile relief, the timber frame, the chimney, the arched door and the windows. Bake wood grain, stone joints and small tile details into the textures. Budget: 5,000 triangles. Result: 4,990.



Windmill: 961,272 triangles to 19,934.

What the windmill brief asked for

Preserve the tower and roof, exactly four sails with their frames, the central hub, the side gear, the door and the windows, and keep the spaces between blades open. Budget: 20,000 triangles. Result: 19,934.

The picture barely changes.

Each pair uses the same camera, light and angle: the original Meshy 7.1 mesh on the left, the optimized mesh on the right. The scooter body was the first optimization pass; the delivered scooter was later rebuilt in separate parts at 9,856 triangles.



Scooter body: 780,858 triangles to 15,990.



Wheelbarrow body: 340,900 triangles to 524.



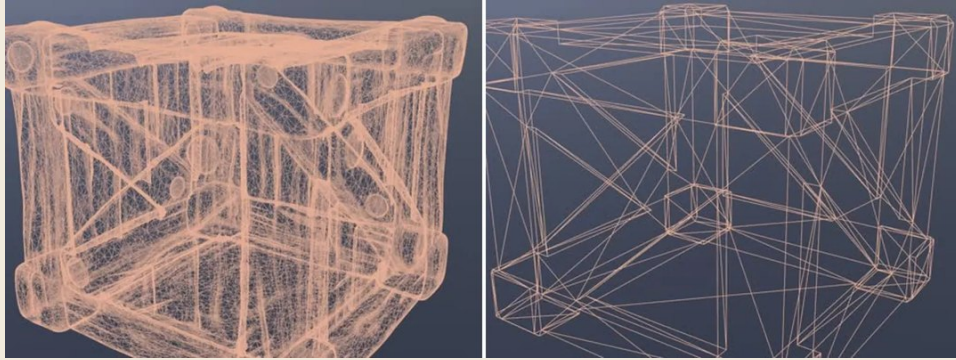
Dwarf head: 631,154 triangles to 6,891, eyes included.



Helmet: 399,914 triangles to 2,980.

Light, and breakable.

The crate shows both ends of the budget: a 372 triangle version that keeps its wood and rivets, and a hollow version that breaks into 26 pieces.



Wireframes: 595,340 triangles (left), 372 (right).

About 1,600 times fewer

The high poly crate has 595,340 triangles and the low poly crate has 372. Wood grain and rivets stay readable because they live in the textures.



Crate in the Unity village.

26 hollow pieces

GPT-6 Astra 3D produced a hollow crate split into 26 independent fragments, 4,116 triangles in all. A rigid body simulation was run once in Blender and baked into keyframes.



The same crate, a moment after the break.

The same break every time

Because the motion is baked, nothing simulates at runtime. The pieces fly out and settle, then rebuild into the crate, so the loop repeats exactly.

Split it so it can move.

The same models that optimized the meshes also prepared them for animation, by splitting each moving part from the rest.



The rotor turns, the tower stays still.

Windmill

Opus 5.5 3D split the rotor from the tower, set its origin on the shaft and baked an eight second, 360 degree turn.



Scooter: body, steering and wheels.

Scooter

It split the scooter into a body, a steering assembly that pivots on the real inclined axis, and two wheels. The delivered scooter has 9,856 triangles.



Helmet lamp off.



Helmet lamp on.

Helmet lamp

It also gave the lamp lens an emissive material. In a dark studio the lamp flickers on, switches off and flickers on again.

A character, not just a prop.

The dwarf is a skinned character with eyes that move, a helmet that lifts off and a pickaxe that leaves his hand. His new head came through the same chain as the props.



Head: 631,154 to 6,891 triangles.



A new head, same chain

GPT Image 2.5 drew the concept, Meshy 7.1 built the model at 631,154 triangles, and Claude Opus 5.5 3D rebuilt it: 6,387 triangles for the head and 252 for each eye.

21,323 skinned triangles

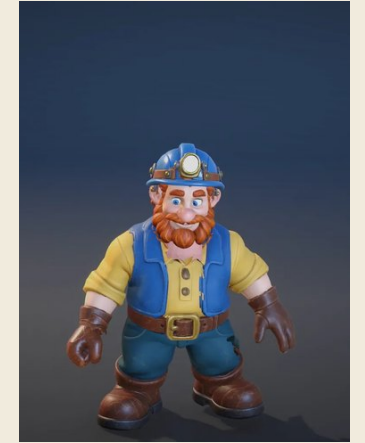
Body 13,992, head 6,387, eyes 504 and neck 440. The rig and skin weights were refined in Blender and Unity; no generative model is credited with the rigging.



Idle, helmet on.



Helmet lifted off.



Pickaxe released.

The right hand

A closed-fist glove was generated separately with GPT Image 2.5 and Meshy 7.1 (3,096 triangles) and attached to the rig, with the pickaxe held through it.

Then they meet in a scene.

The props were captured in Unity 6 and assembled into a village. At rest the scene holds 798,886 triangles across 1,133 mesh renderers, counted live in the film.



The village in Unity 6: the cottage, crates and the trees that surround them.

A generated sky

Skybox Flux generated the sky panorama that surrounds the village.

Generated ground

Scenario Texture made the grass and road textures, and PATINA Image to Maps made their normal maps.

Generated trees

Claude Opus 5.5 3D and Meshy 7.1 made the trees that fill the scene.

Sky, grass and road.

These generated images and textures are the ground the props stand on.



Sky panoramas generated with Skybox Flux.



Grass and road textures, made with Scenario Texture.

Rendered in Unity 6

The props and the village were captured in Unity 6 with the Universal Render Pipeline, real-time shadows and ambient occlusion.

Counts you can check

Before and after triangle counts were read from the GLB files. The village count in the film is read from the scene, frame by frame.

Start with your own sentence.

The same route works for any small world: picture, 3D model, rebuild from a brief, split the moving parts, render it in an engine.



Props from the Unity lineup: crate, dwarf, scooter and cottage.

Run the route yourself

Scenario MCP lets an agent run these models and fetch the results. The agent skill that ships with this Exploration packages the route.

mcp.scenario.com

Not part of this account

Credit costs, timings and ARKit blendshape compatibility are not covered. Who produced the 372 triangle crate is not recorded.

Explore Scenario

Models for images, 3D, video and audio, plus workflows and an API.

scenario.com