Prop Maker: Wood Bridge

1. Setup

This asset requires very minimal to no setup. Simply import it via the Package Manager and you'll have everything you need for your project. Note that the included materials use shaders made for the built-in render pipeline, so you need to upgrade your materials if you use URP or HDRP.

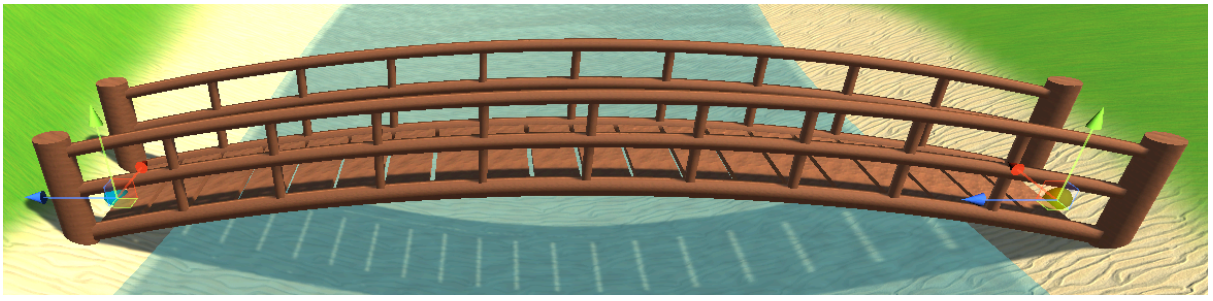
Upgrade materials in URP
Upgrade materials in HDRP

2. Create your first bridge

There are two ways you can go about creating your first bridge. The easiest way is to drag and drop one of the Wood Bridge prefabs into your scene. The other is to create a new Game Object and add the Wood Bridge component. This will automatically add all other necessary components to the Game Object. Note that the bridge will lack materials and which must be added manually. Once you have a bridge, press the "Edit Bridge" button in the inspector to get started.

3. Learn to use the handles

While the bridge is in edit mode two handles that you're probably familiar with will appear at each end of the bridge. These handles look like and work like Unity's position tool and moving them moves the start and end points of your bridge. Play around with the bridge until you feel like you understand how the handles work. Once you feel comfortable using the handles you can move on to the inspector where the real fun begins.



4. Learn to use the inspector

In the inspector, you'll find the WoodBridge component. To keep things simple the Wood Bridge editor has been divided into six tabs. To switch tabs simply click on the tab you wish to see. The current tab will be highlighted in blue. Some parts of the editor will be accessible from all tabs to make it easy. An important thing to remember when working

with procedural assets is that some values can generate weird results so make sure to play around with the settings to get a good-looking result.

Auto Update (Bridge.autoUpdate)
True: Automatically updates the bridge when a change is made
False: Only update the bridge when the "Update Bridge" button is pressed.

Auto Center (Bridge.autoCenter)
True: Automatically move the game object's position to the middle of the bridge.
False: Don't recenter the object unless the Recenter button is pressed.

Edit Orientation (Bridge.editOrientation)
World: The position handles will point towards the world forward, right, and up.
Local XZ: The position handle will point to local forward and right but world up.
Local: The position handles will point towards the local forward, right, and up.

4.1 Shape

Start Point (Bridge.start)
The point the bridge is built from. Can be seen as a yellow sphere in the Scene.

End Point (Bridge.end)
The point the bridge is built towards. Represented by a cyan sphere in the Scene.

Width (RopeBridge.bridgeWidth)
The distance from one edge of the bridge to the other.

Fix Height (RopeBridge.fixHeight)
True: The height of the parabola is controlled by the user. (Recommended for bigger bridges)
False: The height of the parabola is influenced by the Flatness parameter.

Fixed Height (RopeBridge.fixedHeight)
The predetermined height of the parabola if height is fixed.

Flatness (RopeBridge.flatness)
Influences how flat the parabola should be if the height isn't fixed..

4.2 Frame

Soft Edges (WoodBridge. softShadeFrame)
True: Soft shade the edges of the frame.
False: Flat shade the edges of the frame.

Height (WoodBridge.frameHeight)
The height of the frame.

Width (WoodBridge.frameWidth)
The width of the frame.

4.3 Boards

Width (WoodBridge. boardWidth)
The width of the boards.

Height (WoodBridge.boardHeight)
The height of the boards.

Spacing (WoodBridge.spacing)
The spacing between boards.

Padding (WoodBridge.boardPadding)
The size of the empty space before and after the boards.

4.4 Supports

Generate Supports (WoodBridge.generateSupports)
True: Generates a cylinder in each corner of the bridge.
False: Skips this, and any other support-related steps.

Resolution(WoodBridge. supportResolution)
The number of corners on the supports.

Ground Depth (WoodBridge.supportGroundDepth)
How far below the bridge the supports starts.

Radius (WoodBridge.supportRadius)
The radius of the supports.

Height (WoodBridge.support height)
Height of the supports.

4.5 Railing

Generate Railings (WoodBridge.generateRailings)
True: Creates a railing.
False: Don't create anything railing related.

Soft Shade (WoodBridge.softShadeRailing)
True: Soft shade the edges of the railing.
False: Flat shade the edges of the railing.

Height(WoodBridge.railingHeight)
Height of the railing.

Size Offset (WoodBridge.postSizeOffset)
The distance between the edge of the frame and the edge of the posts.

Boards Between Posts (WoodBridge.boardsBetweenPosts)
The number of boards between each post in the railing.

Top Width (WoodBridge.topBeamWidth)
The width of the top beam of the railing.

Top Height (WoodBridge.topBeamHeight)
The height of the top beam of the railing.

Generate Secondary Beam (WoodBridge.generateMiddleBeam)
True: Generate a second beam at a given height in the railing.
False: Don't generate a second beam.

Secondary Height (WoodBridge.middle)
The height where the secondary beam is placed (ranged from 0 - 1).

Middle Width (WoodBridge.middleBeamWidth)
The width of the middle beam.

Middle Height (WoodBridge.middleBeamHeight)
The height of the middle beam.

4.5 Mesh Info

Vertex Count
The number of vertices that makes up the mesh

Triangle Count
The total number of triangles across all sub-meshes.

Mesh Name (Prop.meshName)
The name of the mesh that will be shown in the inspector and if the mesh is exported

Save Mesh As Asset
Allows you to save the mesh as an .asset file so it can be easily reused by other game objects or exported into new projects. If you want to export the mesh as a .fbx file use the FBX Exporter Package.