



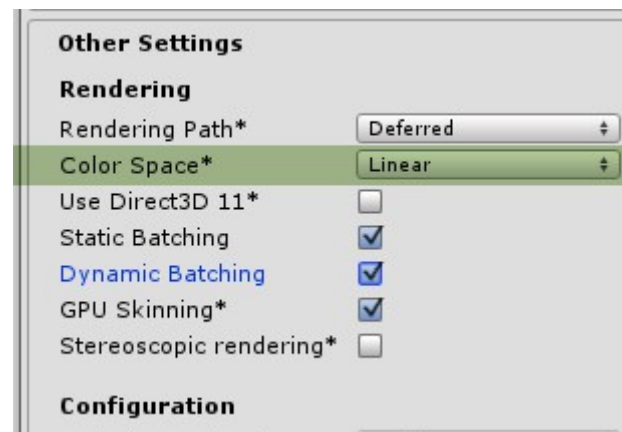
- | | |
|---|--|
| 1. Quick Introduction | 10. Modular Kits – Greathall |
| 2. Project Settings | 11. Modular Kits – Hut2 |
| 3. Grid and Snap Settings | 12. Modular Kits – Towers |
| 4. Sample Scenes | 13. Modular Kits – Fortification |
| 5. Folder Organization and Structure | 14. Buildings – Misc |
| 6. PreAssembled Buildings | 15. Terrain – Ice and Rocks |
| 7. Modular Kits – Keep | 16. Textures |
| 8. Modular Kits – Hut1 and Storehouse | 17. Material/Shader |
| 9. Modular Kits – Forge | 18. Performance |

Quick Introduction

Arctic castle is a fantasy/medieval themed environment asset package for unity. Package contains modular architecture, rocks, icebergs and various natural formations as well as some foliage, props and misc detail meshes. Visually package revolves around snowy/arctic theme but since snow is shader-based and not baked in textures, significant chunk of assets can work in different context as well.

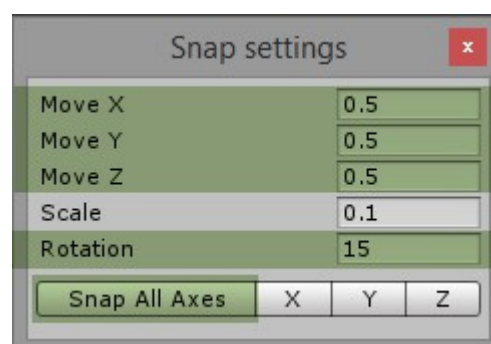
Project Settings

Asset relies on new PBR shader introduced with Unity 5, which looks and works best in Linear color space. So, it is recommended to go to Edit->Project Settings->Player and switch color space from gamma to Linear.

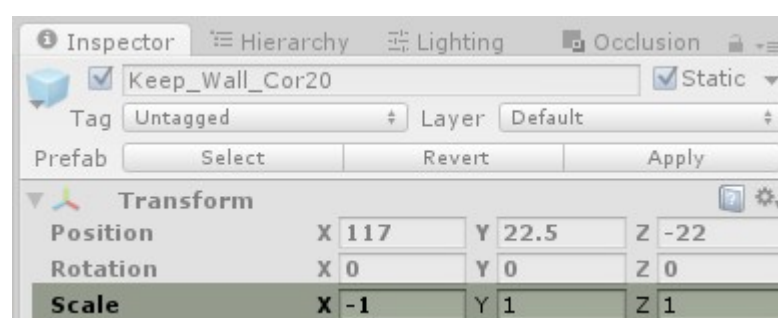


Grid And Snap Settings

Before using modular kits to assemble buildings, grid and snap settings have to be set first. Access it from Edit->Snap settings and set MoveX, MoveY and MoveZ steps to 0.5 units (50 cm). Use increment of 5 degrees for rotation.



If you want to mirror modules, you can simply use scaling value of -1 on respective axes.



Sample Scenes

/ArcticCastle/Scenes/Scene_A.unity and *Scene_B.unity* are sample environments that show all assets in action. Checking these is the quickest way to find out about look, quality and style of assets as well as get sense how they are broken down in terms of modularity and how they fit together.



Folder Organization and Structure

Folder structure of package is pretty self-explanatory. Most of content you are looking for is in Prefabs folder. It contains following subgroups

- ***Atmospherics*** - various atmospheric effects - these can be snow particles, fire, mist etc.
- ***Backgrounds*** - large background assets, such as distant mountains, ocean etc
- ***Buildings*** - containing modular kits used to assemble buildings
- ***/Buildings/PreAssembled/*** - contains preassembled buildings that were constructed using modular kits.
- ***Foliage*** - containing trees and shrubs
- ***Helpers*** - various helper objects, such as human figure for scale comparison purposes as well as collision blocker meshes
- ***Props*** - small detail objects, such as crates barrels, torches etc.
- ***Terrain*** - contains cliffs, rocks, icebergs and various large and small natural formatons that are not foliage.



PreAssembled Buildings

/Prefabs/Buildings/PreAssembled/ folder contains pre-built structures that were assembled from modular kits. These can be simply drag&dropped in your scene or used as basis which you modify, tweak and detail as required by your project. You can also check */Scenes/PreAssembled_01.unity/* scene to get quick visual overview of all preassembled buildings in one scene.



Using Modular Kits

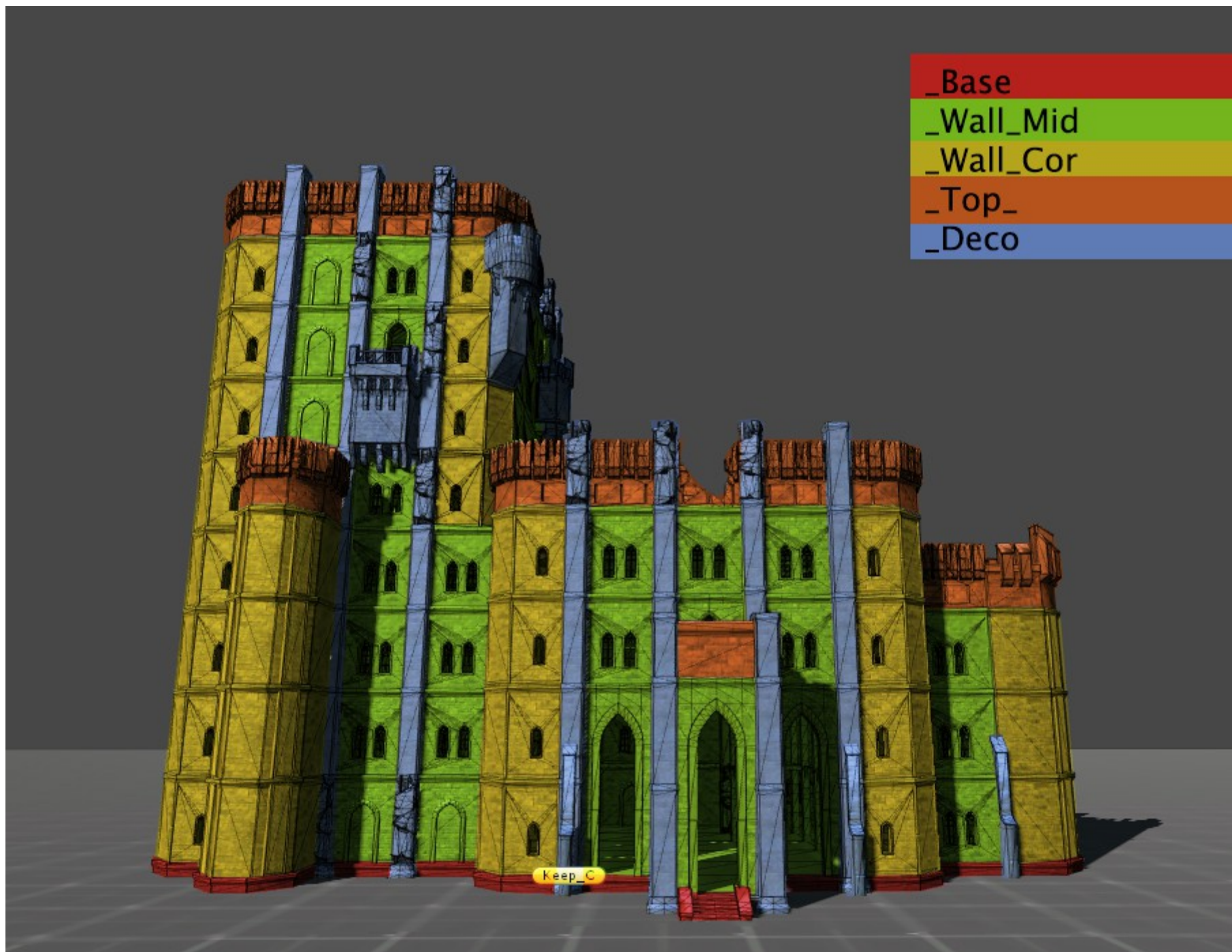
If Preassembled buildings are not sufficient your project or you simply prefer more hands-on approach and want additional control, you can always discard them and assemble your own ones from scratch using Modular kits. These kits are *Greathall*, *Hut1* and *Hut2*, *Storehouse*, *Forge*, *Keep* and fortification as well as *Watchtowers*. Rules of their usage are outlined below.

**Note: if modularity breakdowns below do not provide sufficient information, you can always check Preassembled buildings or sample scenes to see how these modules fit together in practice.*

Modular Kits – Keep

/Prefabs/Buildings/Keep/ is largest modular kit of package. It can be used to assemble small, medium or large sized keeps and castles. In order to help identify function and relative placement of each module, they are broken down into several categories, separated by prefixes. These are

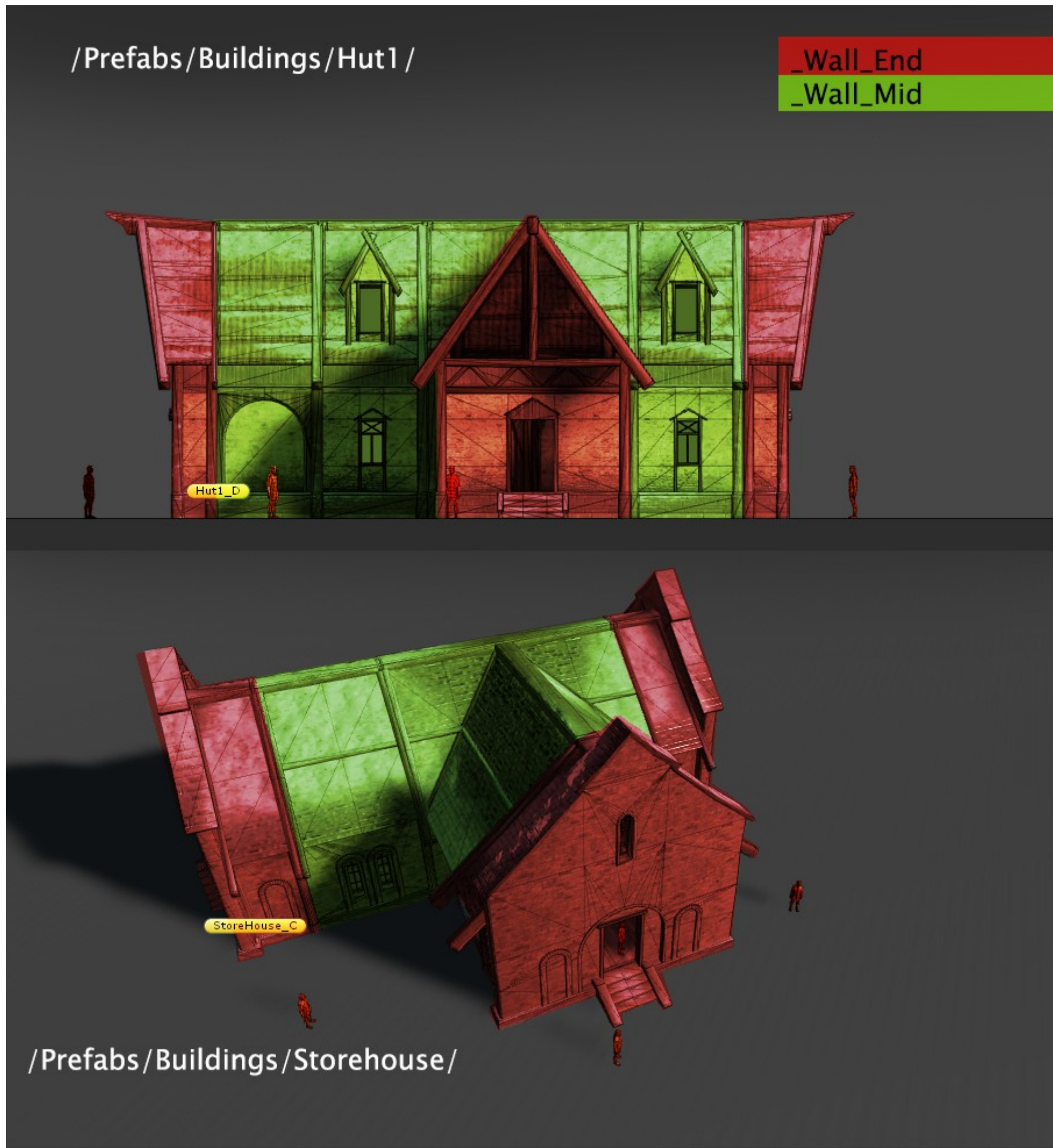
- **_Base** – use these to form base of building. copied horizontally
- **_Floor** – floor modules. aside from interior floors, these can be used to form roofs or terraces. copied horizontally.
- **_Wall_Mid** – main walls of buildings. These can be plain walls as well as ones with windows or door frames. Copy them horizontally or vertically to form width and height of buildins.
- **_Wall_Cor** – corner modules, used alongside walls and stacked vertically to form height of building. Of course these can also be flipped horizontally.
- **_Wall_Int** – Interior walls. After outer shell and floors of building are formed, you can use these modules to subdivide interior areas into corridors and rooms.
- **_Top_Mid, _Top_Cor** – these form topmost part of walls. Place _Top_Mid modules on top of _Wall_Mid-s. _Top_Cor modules work with _Wall_Cor.
- **_Deco** – Various stone and wooden support collumns, extensions, handrails, buttreses and other decorative modules. These are used after main shell of building is formed to add some visual interest, help break up boring areas and mask transitions between other modules.
- **_Staircase** – Interior staircases for keep.
- **_Door** – Door modules.



Modular Kits – Hut1 and Storehouse

Forge, Hut1 and Storehouse modules share same level of modularity. Generally these are one or two story structures and modules are distributed horizontally. Modules include

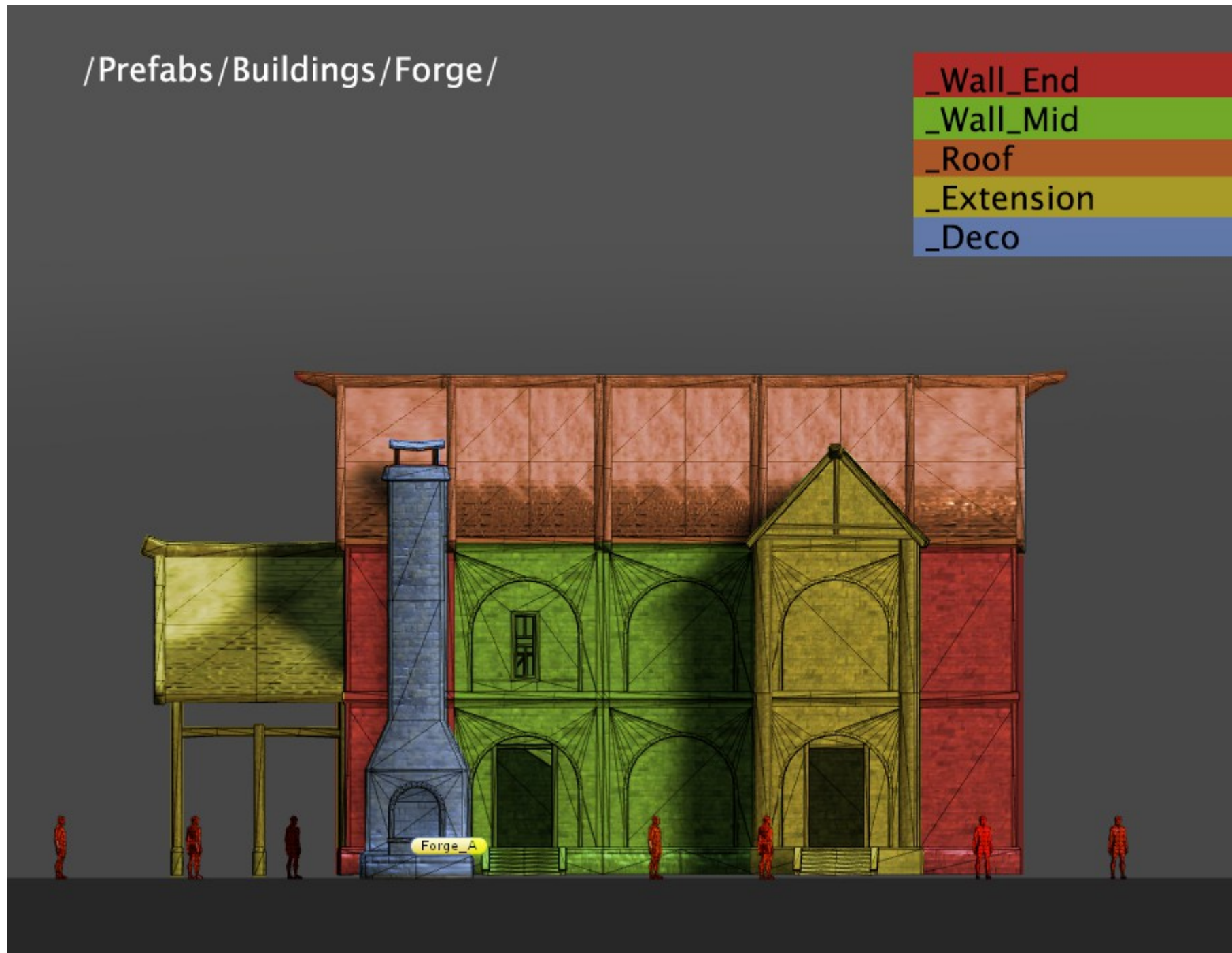
- *_Wall_Mid* – Start with these to form middle part of buildings. Copied horizontally
- *_Wall_End* – cap *_Wall_Mid* modules with these
- *_Extension* – generally used after main structure of buildings is formed.
- *_Deco* – secondary, decorative modules that can be used to break up and detalize buildings
- *_Column* – column modules. These are used as support pieces as well as for masking transitions between modules
- *_Door* – doors



Modular Kits – Forge

Modular elements for forge consist of these sub-groups

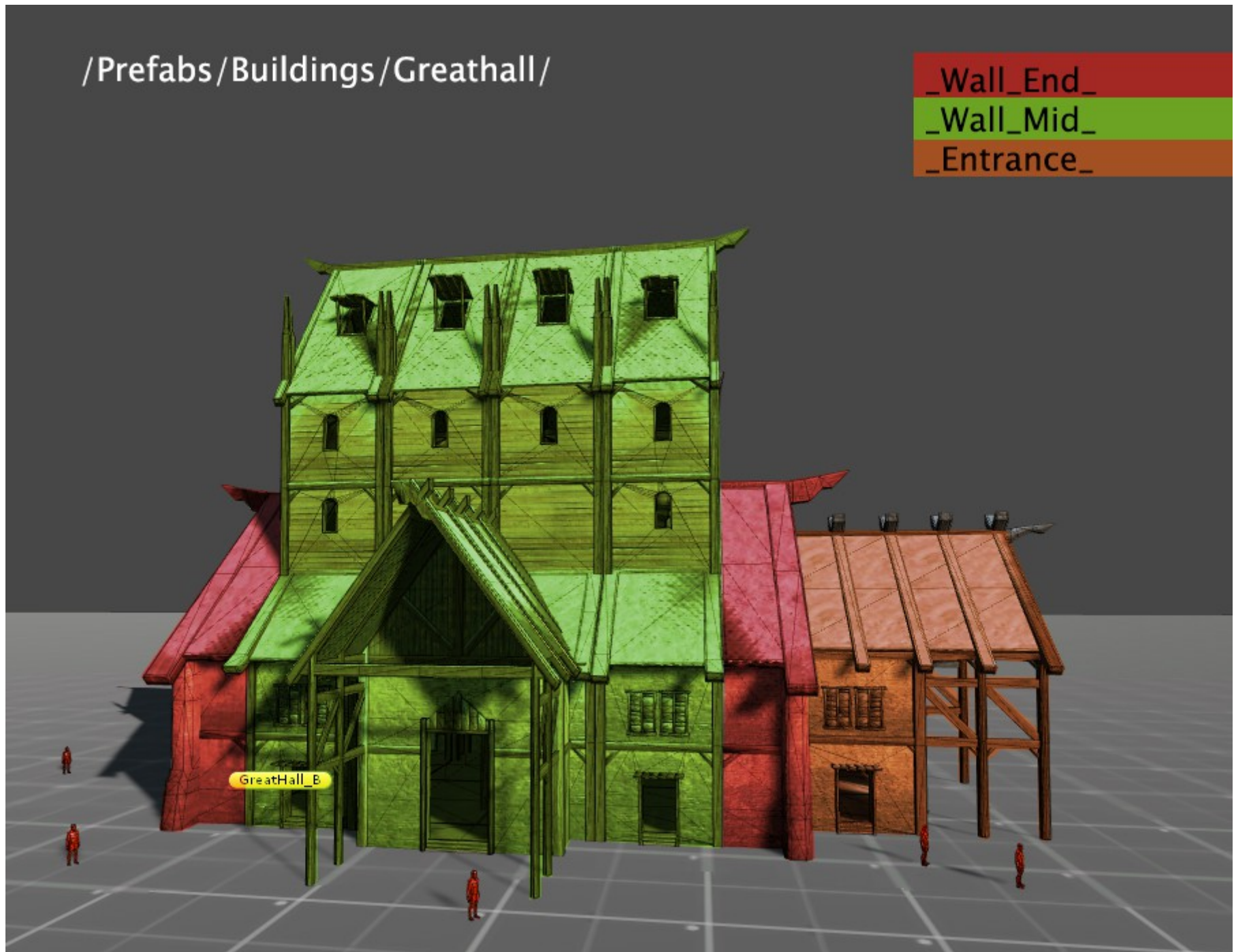
- *_Wall_Mid* – Copy these horizontally to form main walls of forge
- *_Wall_End* – Copy these horizontally to end/cap walls
- *_Roof* – roofing modules. copied horizontally
- *_Extension* – extension modules for forge.
- *_Deco* – decorative pieces



Modular Kits – Greathall

Modularity breakdown of Greathall is very similar to Storehouse and consists of:

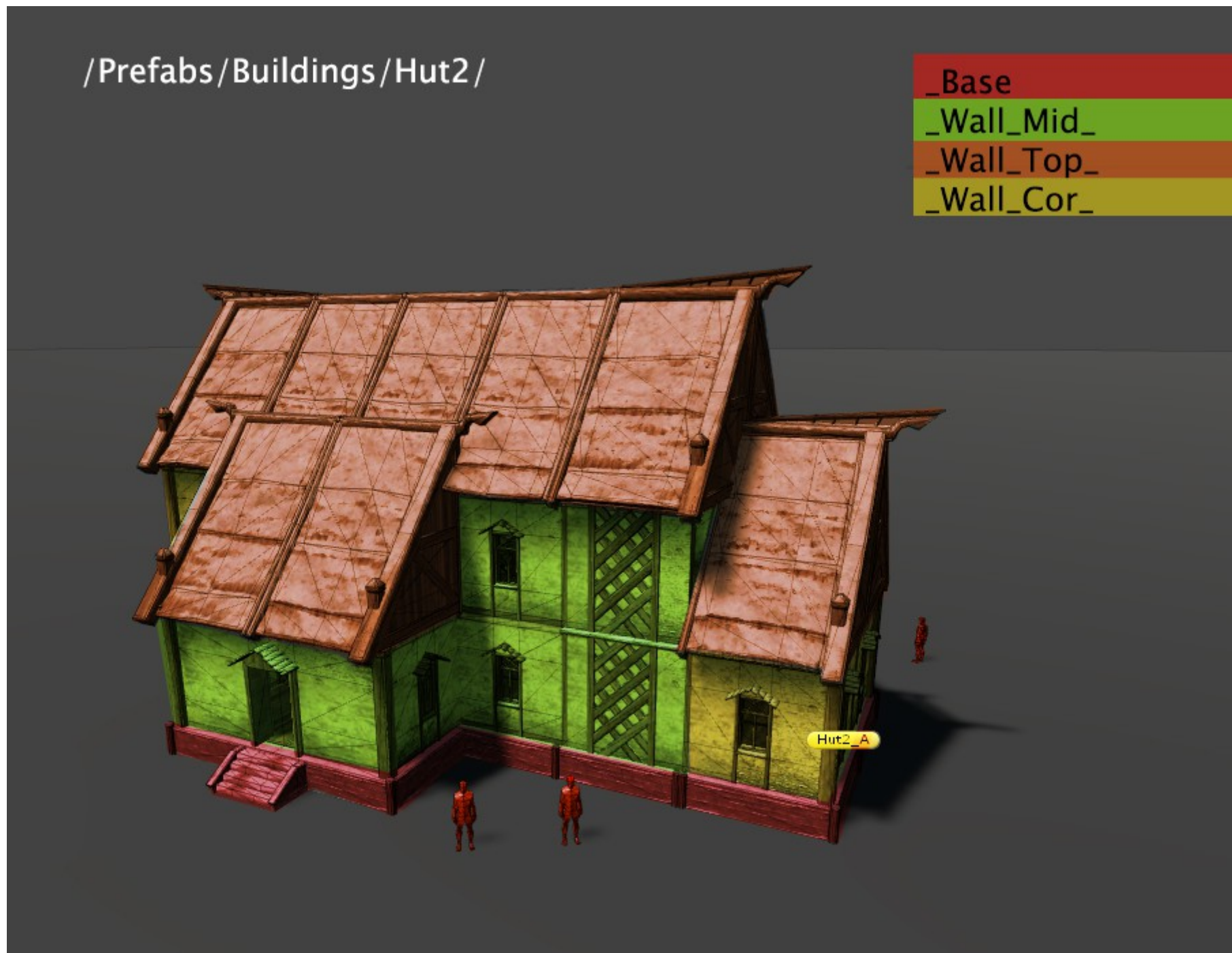
- *_Wall_Mid_* – for forming main parts of building
- *_Wall_End_* – for capping walls
- *_Entrance_* – this module is has more of a decorative function and is almost always used with one of *_Wall_End_* modules to form little entrance area.
- *_Door_* – large and small door modules.



Modular Kits – Hut2

Modularity breakdown of Hut2 is very similar to Keep. It consists of following modules:

- **_Base_** – these are form base of building. Includes _Base_Mid, _Base_Cor and _Base_Stairs modules
- **_Wall_Mid_** – main wall modules. Copied horizontally and vertically to form width and height of buildings.
- **_Wall_Cor_** – used to form building corners.
- **_Top_** – these are used in conjunction with walls and form topmost parts and roof of building. Copied horizontally on top of _Wall_Mid and _Wall_Cor modules.
- **_Floor_** – floor modules. Copied horizontally
- **_Deco_** – decorative modules that do not participate in forming floor plan of building, but serve as decorative and support pieces. These can be handrails, columns, doors etc.



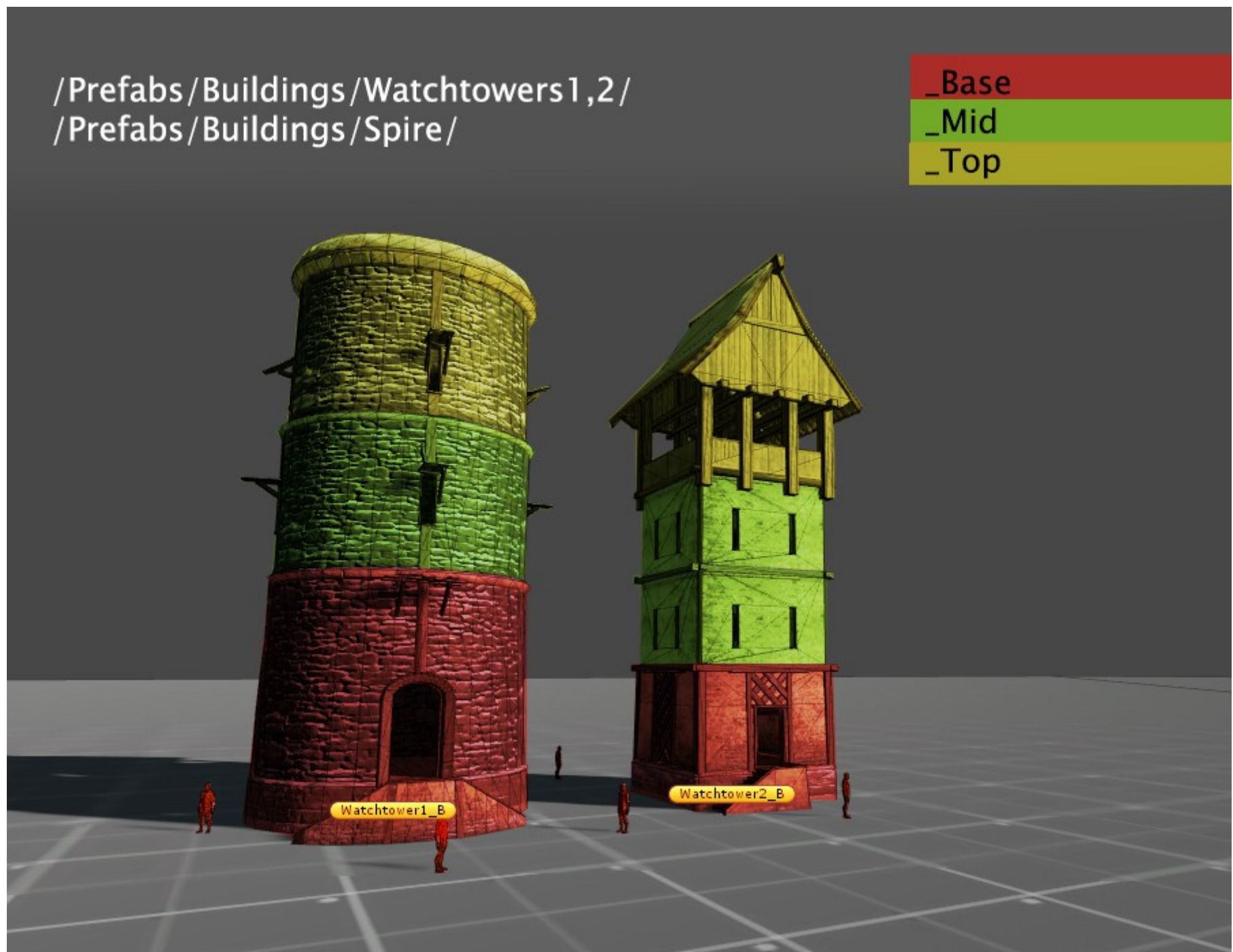
Modular Kits – Towers

/Prefabs/Buildings/Watchtowers 1 and 2 as well as */Buildings/Spire/* fall into this category. These are simplest to assemble and consist of:

_Base – forms base for towers

Mid – middle parts. stack them vertically to form towers of desired height

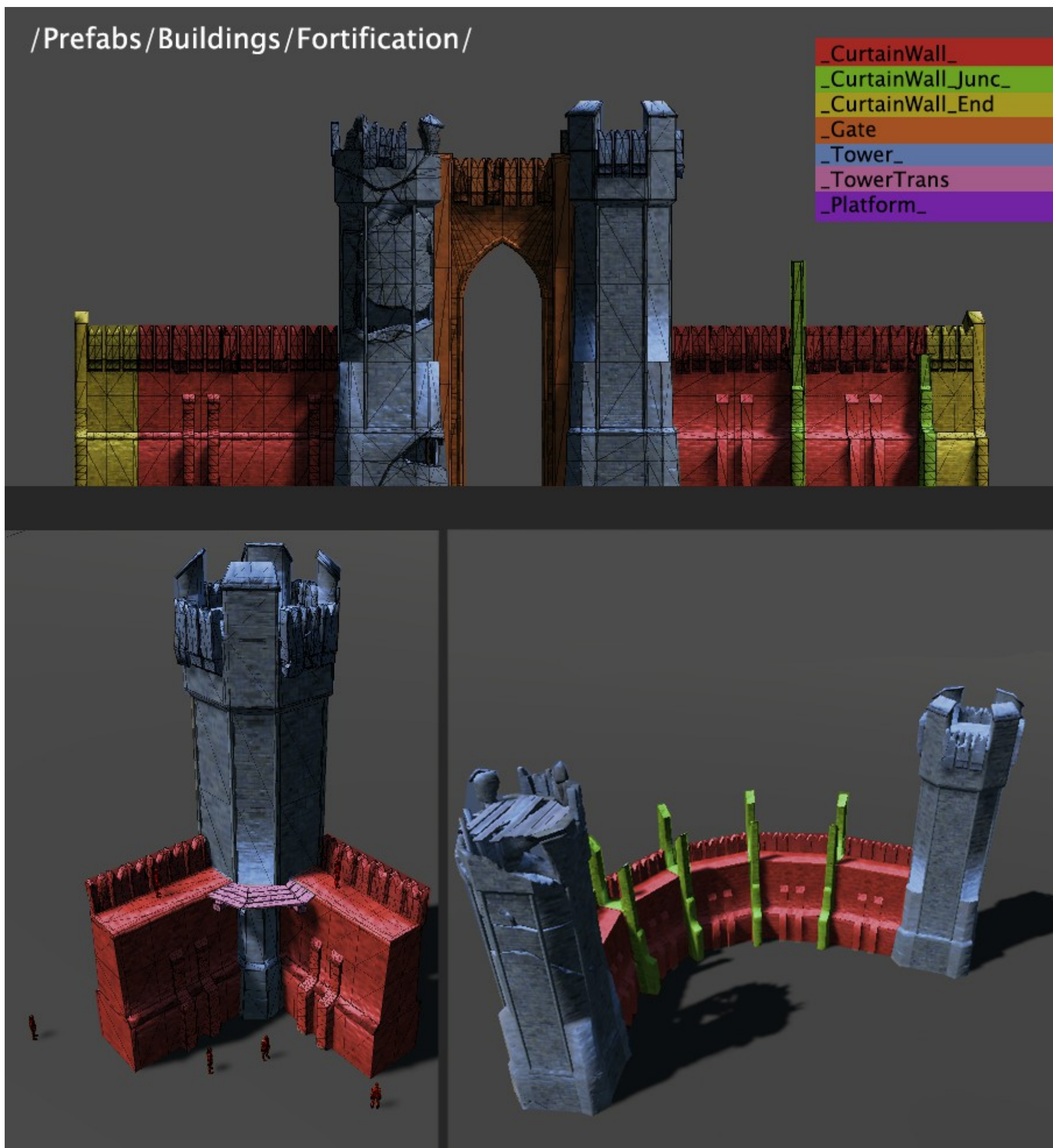
_Top – topmost parts of towers



Modular Kits – Fortification

/Prefabs/Buildings/Fortification/ houses all structures that are part of outer castle fortification. These are towers, curtainwalls, bastions, gates and bridges. Fort modules are broken down into these subcategories:

- *_CurtainWall_* – curtain walls of fort. Copy them horizontally to form outer wall of fort. Cap them with *_CurtainWall_End* modules
- *_CurtainWall_Junc_* – junction modules for curtainwalls. These are used as transition pieces between wall modules and are used for masking mesh intersections.
- *_Tower_* – tower modules. Use these between curtainwall modules to provide visual break points as well as help redirect flow of walls.
- *_TowerTrans_* – these modules serve as transition pieces from towers to curtainwalls
- *_Gate_* – gate modules. Best used between tower modules to form entrance to fortification.
- *_Platform_* – distribute these modules horizontally to form podium-like formations that can be used as pedestals for buildings or as is.
- *_Bridge_* – bridge module. Used in conjunction with gate modules



Buildings/Misc

/Prefabs/Buildings/Misc/ folder does not contain actual buildings, but manmade objects that are often used together with buildings. These are fences, scaffolding, palisades, bridges, docks, tents etc. These are very good for forming additional infrastructure, scaffolding and walkways around buildings as well as provide nice visual breakup and detail

Terrain – Ice and Rocks

/Prefabs/Terrain/Ice/ and */Prefabs/Terrain/Rocks/* folders contain large and small rock and ice assets. Their usage has no rules whatsoever and can be scaled, rotated and clumped together to form rock and ice formations of desired shape and size. For example mountains in sample scenes are formed by jamming rock and cliff assets inside hand-sculpted terrain.



Textures

ArcticCastle/Textures folder houses all textures used in this project. Textures are categorized by using prefixes. These are:

- *ATM_* – textures used for particles and various atmospheric effects
- *BLD_* – textures for buildings
- *BGR_* – textures for background assets as well as skyboxes.
- *ICE_* – baked and tiling textures for Ice formations
- *ROCK_* – textures for cliffs, stones and various rock formations
- *PROP_* – textures that are used for props
- *GR_* – ground textures, mainly used for terrains. These are dirt, snow, gravel, pebbles etc. Tiling
- *DET_* – detail textures, mainly used as detailNormals or as parts of terrain. These are also tiling.
- *FOL_* – Foliage textures – branches, leaves, grass etc
- *MISC_* – everything that does not fit in above categories, including various helper/debug textures

Texture suffixes indicate actual type of texture map. For example:

- *TextureName_A* – Albedo/Diffuse map (RGB)
- *TextureName_AO* – Ambient Occlusion map.
- *TextureName_MS* – Metalness (R) and smoothness (A) map. Objects that are metallic are marked as white in R channel. Smoothness determines how smooth objects are (white = polished, black = rough)
- *TextureName_N* – Normal map.

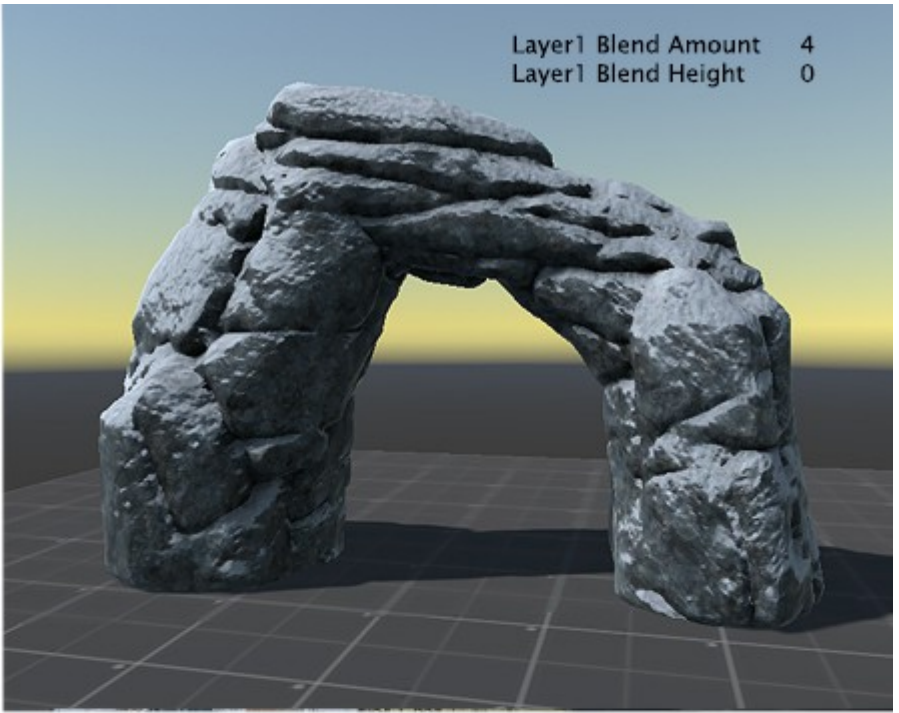
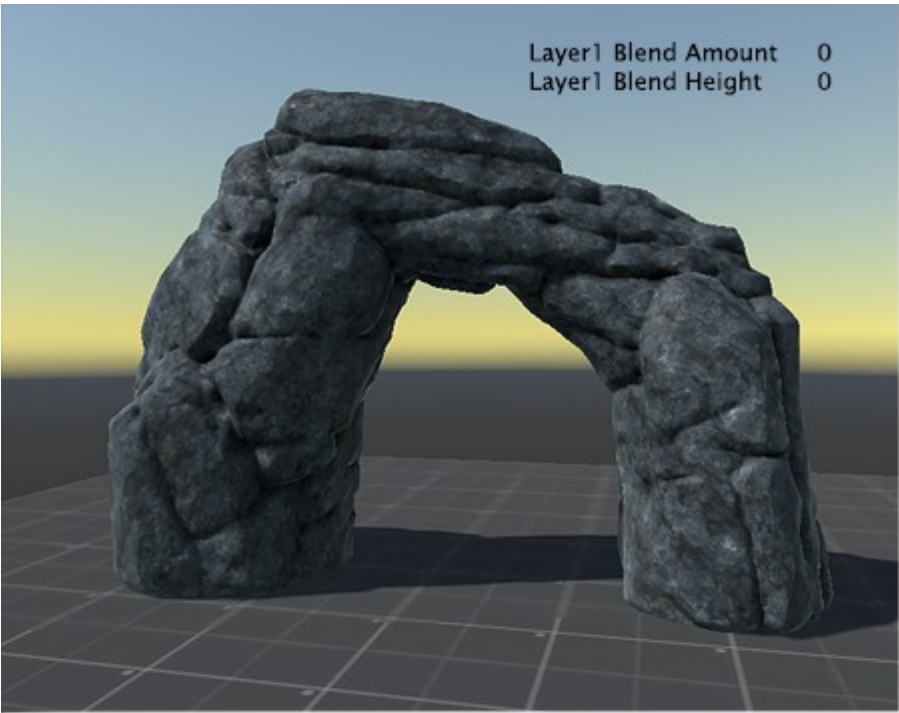
Material/Shader

SURFPBR_TopBlend is a surface shader that is used on 99% of opaque objects in this package. It is mainly used to blend tiling snow on top of up-facing portions of geometry, but it can pretty much be used to blend any kind of texture that shares same distribution pattern as snow – including moss, dirt, dust etc.



Shader uses Unity's Standard PBR lighting model as basis and expects metalness/smoothness maps for reflectivity. Following is a detailed breakdown of material components:

- **Base Albedo (RGB)** – base albedo/diffuse map
- **Base Metalness (R) Smoothness (A)** – metalness/smoothness map for base. As in case of Unity's standard shader, metalness is stored in Red channel of RGBA texture and smoothness is stored in alpha channel.
- **Base Normal** – base normal map
- **Base AO** – base ambient occlusion map. This can be left blank if you do not have AO map.
- **Layer1 Albedo (RGB) , Smoothness (A)** – diffuse map for layer1 (blend layer). This can be tiling snow, moss, dirt texture that you wish to blend on top of existing textures. Blending happens only on portions of geometry that face upwards
- **Layer1 Metalness** – Metalness slider for Layer1. It is mainly included for artistic control and in 99% of cases should be left to zero.
- **Layer1 Normal** – normal map for layer1
- **Layer1 Tiling** – Tiling multiplier for Layer1. Higher values mean more tiling
- **Layer1 Blend Amount, Layer1 Blend Height** – Use these two inputs together to fine tune how layer1 blends with base texture. Blend height can take both positive and negative values
- **Ignore Vcol Mask** – by default blending does not happen on areas of geometry that have vertices painted black, even if they face upwards. This control allows shader to ignore vertex mask.
- **Detail Normal** – detail normal map. This can be used to add fine grain on top of existing normal maps to create illusion of detail
- **DetailNormal Intensity, DetailNormal Tiling** – tiling and intensity controls for detail normal map



Performance

In order to estimate performance it is recommended to check [webplayer build](#) on target hardware. There is a FPS counter near top-right corner that displays both average FPS and frame time in milliseconds.

At the moment package is not optimized for mobile devices.

