

uConstruct

1.3

Generated by Doxygen 1.8.11

Contents

| | | |
|----------|---|-----------|
| 1 | Namespace Index | 1 |
| 1.1 | Packages | 1 |
| 2 | Hierarchical Index | 3 |
| 2.1 | Class Hierarchy | 3 |
| 3 | Class Index | 7 |
| 3.1 | Class List | 7 |
| 4 | Namespace Documentation | 11 |
| 4.1 | uConstruct Namespace Reference | 11 |
| 4.1.1 | Enumeration Type Documentation | 13 |
| 4.1.1.1 | PlacingRestrictionType | 13 |
| 4.2 | uConstruct.CodeGenerator Namespace Reference | 13 |
| 4.3 | uConstruct.Conditions Namespace Reference | 13 |
| 4.4 | uConstruct.Core Namespace Reference | 14 |
| 4.5 | uConstruct.Core.AOI Namespace Reference | 14 |
| 4.6 | uConstruct.Core.Blueprints Namespace Reference | 14 |
| 4.7 | uConstruct.Core.Manager Namespace Reference | 14 |
| 4.8 | uConstruct.Core.Physics Namespace Reference | 15 |
| 4.9 | uConstruct.Core.PrefabDatabase Namespace Reference | 15 |
| 4.10 | uConstruct.Core.Saving Namespace Reference | 15 |
| 4.11 | uConstruct.Core.Templates Namespace Reference | 16 |
| 4.12 | uConstruct.Core.Threading Namespace Reference | 16 |
| 4.13 | uConstruct.Demo Namespace Reference | 16 |
| 4.14 | uConstruct.Extensions Namespace Reference | 17 |
| 4.15 | uConstruct.Extensions.PCloudExtension Namespace Reference | 17 |
| 4.16 | uConstruct.Sockets Namespace Reference | 17 |

| | | |
|----------|---|-----------|
| 5 | Class Documentation | 19 |
| 5.1 | uConstruct.Core.AOI.AOIManager Class Reference | 19 |
| 5.1.1 | Detailed Description | 20 |
| 5.1.2 | Member Function Documentation | 20 |
| 5.1.2.1 | AddFinder(BaseAOIFinder value) | 20 |
| 5.1.2.2 | AddTarget(BaseAOITarget value) | 20 |
| 5.1.2.3 | ComputeAOI(BaseAOIFinder finder) | 20 |
| 5.1.2.4 | HandleAOI(BaseAOITarget target, BaseAOIFinder finder, bool isUnityThread) | 20 |
| 5.1.2.5 | RemoveFinder(BaseAOIFinder value) | 21 |
| 5.1.2.6 | RemoveTarget(BaseAOITarget value) | 21 |
| 5.1.2.7 | UpdateAOI(BaseAOIFinder finder) | 21 |
| 5.1.2.8 | UpdatePositions() | 21 |
| 5.1.3 | Member Data Documentation | 21 |
| 5.1.3.1 | finders | 21 |
| 5.1.3.2 | targets | 21 |
| 5.2 | uConstruct.Core.AOI.BaseAOIFinder Class Reference | 22 |
| 5.2.1 | Detailed Description | 22 |
| 5.2.2 | Member Function Documentation | 22 |
| 5.2.2.1 | OnDisable() | 22 |
| 5.2.2.2 | OnDrawGizmos() | 22 |
| 5.2.2.3 | OnEnable() | 23 |
| 5.2.2.4 | Update() | 23 |
| 5.2.2.5 | UpdateAOI() | 23 |
| 5.2.3 | Member Data Documentation | 23 |
| 5.2.3.1 | aoiPosition | 23 |
| 5.2.3.2 | oldPos | 23 |
| 5.2.3.3 | radius | 23 |
| 5.3 | uConstruct.Core.AOI.BaseAOITarget Class Reference | 23 |
| 5.3.1 | Detailed Description | 24 |
| 5.3.2 | Member Function Documentation | 24 |

| | | |
|----------|--|----|
| 5.3.2.1 | HandleAOI(BaseAOIFinder finder, bool _inRange) | 24 |
| 5.3.2.2 | InZone(Vector3 finderPos, float radius) | 24 |
| 5.3.2.3 | OnDisable() | 25 |
| 5.3.2.4 | OnEnable() | 25 |
| 5.3.3 | Member Data Documentation | 25 |
| 5.3.3.1 | aoiPosition | 25 |
| 5.3.3.2 | inRange | 25 |
| 5.3.4 | Property Documentation | 25 |
| 5.3.4.1 | useMultiThreadZoneSearch | 25 |
| 5.4 | uConstruct.BaseBuilding Class Reference | 25 |
| 5.4.1 | Detailed Description | 29 |
| 5.4.2 | Member Function Documentation | 29 |
| 5.4.2.1 | ActivateColliders(bool value) | 29 |
| 5.4.2.2 | ActivateConditions(bool value, bool force) | 29 |
| 5.4.2.3 | ActivateSnapPoints(bool value) | 30 |
| 5.4.2.4 | ActivateSockets(bool value, bool forced) | 30 |
| 5.4.2.5 | AddCondition(BaseCondition condition) | 30 |
| 5.4.2.6 | AddSocket(BaseSocket socket) | 30 |
| 5.4.2.7 | AddTemplate(GameObject template) | 30 |
| 5.4.2.8 | AssignOriginalColors() | 31 |
| 5.4.2.9 | Awake() | 31 |
| 5.4.2.10 | BuildingDeattached() | 31 |
| 5.4.2.11 | BuildingDestroyed() | 31 |
| 5.4.2.12 | BuildingGroupChanged(BaseBuildingGroup group) | 31 |
| 5.4.2.13 | BuildingPlaced() | 31 |
| 5.4.2.14 | CheckConditions() | 31 |
| 5.4.2.15 | CreateCondition(string name, SocketPositionAnchor anchor, System.Type condition) | 31 |
| 5.4.2.16 | CreateSnapPoint(string name, SocketPositionAnchor anchor, BuildingType type) | 32 |
| 5.4.2.17 | CreateSocket(string name, SocketPositionAnchor socketAnchor, GameObject previewGameObject, BuildingType receive, PlacingRestrictionType restriction) | 32 |

| | | |
|----------|--|----|
| 5.4.2.18 | DeAttachBuilding() | 32 |
| 5.4.2.19 | DestroyBuilding() | 33 |
| 5.4.2.20 | EnableRenderings(bool value) | 33 |
| 5.4.2.21 | GroupBuildingAdded(BaseBuilding building) | 33 |
| 5.4.2.22 | GroupBuildingRemoved(BaseBuilding building) | 33 |
| 5.4.2.23 | HandleFreePlace(UCPhysicsHit hit, BaseSocket socket) | 33 |
| 5.4.2.24 | HandleMaterial(BuildingMaterialData mat) | 34 |
| 5.4.2.25 | HandlePlacing(UCPhysicsHitsArray hits) | 34 |
| 5.4.2.26 | HandlePlacing(params RaycastHit[] hits) | 34 |
| 5.4.2.27 | HandleSnapPlace(UCPhysicsHit hit, BaseSocket socket) | 34 |
| 5.4.2.28 | InitiateBuildingData() | 35 |
| 5.4.2.29 | LostSnapToSocket(BaseSocket socket) | 35 |
| 5.4.2.30 | Pack() | 35 |
| 5.4.2.31 | PlaceBuilding() | 35 |
| 5.4.2.32 | RemoveTemplate(Template template) | 35 |
| 5.4.2.33 | ResetMaterialColors() | 35 |
| 5.4.2.34 | ReturnParent(string parentName, bool createlfNotFound) | 35 |
| 5.4.2.35 | ReturnPosition(SocketPositionAnchor anchor, Transform transform) | 36 |
| 5.4.2.36 | ReturnSocket(Vector3 pos, BuildingType?targetType) | 36 |
| 5.4.2.37 | SnappedToSocket(BaseSocket socket) | 36 |
| 5.4.3 | Member Data Documentation | 37 |
| 5.4.3.1 | _uid | 37 |
| 5.4.3.2 | globalCount | 37 |
| 5.4.4 | Property Documentation | 37 |
| 5.4.4.1 | ignore | 37 |
| 5.4.4.2 | priority | 37 |
| 5.5 | uConstruct.BaseBuildingBatcher Class Reference | 37 |
| 5.5.1 | Detailed Description | 38 |
| 5.5.2 | Member Function Documentation | 38 |
| 5.5.2.1 | DestroyBuilding() | 38 |

| | | |
|----------|---|----|
| 5.5.2.2 | GetBatchedBuilding(Vector3 point) | 38 |
| 5.6 | uConstruct.BaseBuildingGroup Class Reference | 38 |
| 5.6.1 | Detailed Description | 40 |
| 5.6.2 | Member Function Documentation | 40 |
| 5.6.2.1 | AddBuilding(BaseBuilding building) | 40 |
| 5.6.2.2 | AOIGroup(bool value) | 40 |
| 5.6.2.3 | Awake() | 41 |
| 5.6.2.4 | Batch(bool value, BaseBuilding updatedBuilding, bool added) | 41 |
| 5.6.2.5 | CreateGroup< T >(Vector3 pos) | 41 |
| 5.6.2.6 | DestroyGroup() | 41 |
| 5.6.2.7 | EnableGroupSockets(bool value, bool force) | 41 |
| 5.6.2.8 | GetBuildings() | 42 |
| 5.6.2.9 | GetUpdatedBatchData(BaseBuilding building, bool add) | 42 |
| 5.6.2.10 | HandleOccupiedSockets(BaseBuilding building) | 42 |
| 5.6.2.11 | IsGroupSocketOccoupled(BaseSocket socketInstance) | 42 |
| 5.6.2.12 | PopulateBatchedFilters(BaseBuilding building, bool Add) | 42 |
| 5.6.2.13 | RemoveBuilding(BaseBuilding building) | 43 |
| 5.6.2.14 | ReturnBatchedBuilding(Vector3 pos) | 43 |
| 5.6.2.15 | Save() | 43 |
| 5.6.3 | Property Documentation | 43 |
| 5.6.3.1 | buildings | 43 |
| 5.7 | uConstruct.Conditions.BaseCondition Class Reference | 43 |
| 5.7.1 | Detailed Description | 44 |
| 5.7.2 | Member Function Documentation | 44 |
| 5.7.2.1 | Awake() | 44 |
| 5.7.2.2 | CheckCondition() | 44 |
| 5.7.2.3 | OnDrawGizmos() | 45 |
| 5.7.2.4 | Pack() | 45 |
| 5.7.3 | Member Data Documentation | 45 |
| 5.7.3.1 | rootBuilding | 45 |

| | | |
|----------|--|----|
| 5.7.4 | Property Documentation | 45 |
| 5.7.4.1 | DisableOnPlace | 45 |
| 5.7.4.2 | ignore | 45 |
| 5.7.4.3 | priority | 45 |
| 5.8 | uConstruct.Sockets.BaseSnapPoint Class Reference | 46 |
| 5.8.1 | Detailed Description | 47 |
| 5.8.2 | Member Function Documentation | 47 |
| 5.8.2.1 | AnchoredPosition(Vector3 renderCenter, Vector3 renderSize, Vector3 origin) | 47 |
| 5.8.2.2 | Awake() | 48 |
| 5.8.2.3 | OnDrawGizmos() | 48 |
| 5.8.2.4 | ReturnClosest(BaseSnapPoint[] points, Vector3 pointInfluence, BuildingType type) | 48 |
| 5.8.2.5 | ReturnDistance(Vector3 pos) | 48 |
| 5.8.2.6 | Snap(Transform owner) | 48 |
| 5.9 | uConstruct.Sockets.BaseSocket Class Reference | 49 |
| 5.9.1 | Detailed Description | 50 |
| 5.9.2 | Member Function Documentation | 51 |
| 5.9.2.1 | Awake() | 51 |
| 5.9.2.2 | BuildingSnapped(bool value) | 51 |
| 5.9.2.3 | Create() | 51 |
| 5.9.2.4 | EnableSocket(bool value) | 51 |
| 5.9.2.5 | ForceEnable(bool enable) | 51 |
| 5.9.2.6 | GetTransform() | 51 |
| 5.9.2.7 | GloballyEnableSockets(bool value) | 52 |
| 5.9.2.8 | InitiateComponents(GameObject previewObject, Vector3 colliderScale) | 52 |
| 5.9.2.9 | IsFit(BaseBuilding building, PlacingRestrictionType buildingPlaceType) | 52 |
| 5.9.2.10 | OccupySocket(bool value) | 52 |
| 5.9.2.11 | Pack() | 52 |
| 5.9.2.12 | RenderEditor() | 53 |
| 5.9.2.13 | Update() | 53 |
| 5.9.3 | Member Data Documentation | 53 |

| | | |
|----------|--|----|
| 5.9.3.1 | drawIndividual | 53 |
| 5.9.3.2 | isHoverTarget | 53 |
| 5.9.4 | Property Documentation | 53 |
| 5.9.4.1 | priority | 53 |
| 5.10 | uConstruct.Core.Saving.BaseUCSaveData Class Reference | 53 |
| 5.10.1 | Detailed Description | 54 |
| 5.10.2 | Member Function Documentation | 54 |
| 5.10.2.1 | Load(BaseUCSaveData data) | 54 |
| 5.11 | uConstruct.BatchClass Class Reference | 54 |
| 5.11.1 | Detailed Description | 55 |
| 5.11.2 | Member Function Documentation | 55 |
| 5.11.2.1 | Contains(Material[] materials) | 55 |
| 5.12 | uConstruct.BatchData Class Reference | 55 |
| 5.12.1 | Detailed Description | 56 |
| 5.12.2 | Member Function Documentation | 56 |
| 5.12.2.1 | Add(BatchClass value) | 56 |
| 5.12.2.2 | Batchable(Material[] data, int shapeVertexCount) | 56 |
| 5.12.2.3 | Remove(BatchClass value) | 56 |
| 5.13 | uConstruct.BatchExtensions Class Reference | 57 |
| 5.13.1 | Detailed Description | 57 |
| 5.14 | uConstruct.BatchUtility Class Reference | 57 |
| 5.14.1 | Detailed Description | 57 |
| 5.14.2 | Member Function Documentation | 57 |
| 5.14.2.1 | CompileInitialBatchData(MeshFilter[] batch, bool value) | 57 |
| 5.14.2.2 | HandleRenders(MeshFilter filter, bool value) | 58 |
| 5.14.2.3 | isVertexOverLimit(int amount) | 58 |
| 5.14.2.4 | UpdateBatchData(MeshFilter[] filters, bool Add, ref BatchData batchData) | 58 |
| 5.15 | uConstruct.Core.Blueprints.Blueprint Class Reference | 59 |
| 5.15.1 | Detailed Description | 60 |
| 5.15.2 | Member Function Documentation | 60 |

| | | |
|----------|---|----|
| 5.15.2.1 | AddField(BlueprintField field) | 60 |
| 5.15.2.2 | CreateBlueprint() | 60 |
| 5.15.2.3 | Delete() | 60 |
| 5.15.2.4 | GetPath(string name) | 60 |
| 5.15.2.5 | RemoveField(BlueprintField field) | 60 |
| 5.15.2.6 | Save() | 61 |
| 5.15.3 | Member Data Documentation | 61 |
| 5.15.3.1 | BLUEPRINT_ASSET_FIRST | 61 |
| 5.15.3.2 | blueprintName | 61 |
| 5.15.3.3 | fields | 61 |
| 5.15.4 | Property Documentation | 61 |
| 5.15.4.1 | BLUEPRINT_ASSET_PATH | 61 |
| 5.15.4.2 | selfPath | 61 |
| 5.16 | uConstruct.Core.Blueprints.BlueprintData Class Reference | 61 |
| 5.16.1 | Detailed Description | 62 |
| 5.17 | BlueprintEditEditor Class Reference | 62 |
| 5.18 | BlueprintEditor Class Reference | 63 |
| 5.19 | uConstruct.Core.Blueprints.BlueprintField Class Reference | 63 |
| 5.19.1 | Detailed Description | 64 |
| 5.19.2 | Constructor & Destructor Documentation | 64 |
| 5.19.2.1 | BlueprintField(BuildingType type) | 64 |
| 5.19.2.2 | BlueprintField(BuildingType type, GameObject source) | 65 |
| 5.19.3 | Member Function Documentation | 65 |
| 5.19.3.1 | Contains(BlueprintField field, List< BlueprintField > fields) | 65 |
| 5.19.3.2 | HandlePivot(GameObject go) | 65 |
| 5.19.3.3 | OnAfterDeserialize() | 65 |
| 5.19.3.4 | OnBeforeSerialize() | 66 |
| 5.19.3.5 | Pack(GameObject target) | 66 |
| 5.19.3.6 | UnPack(GameObject target, bool saveToPrefab) | 66 |
| 5.19.4 | Member Data Documentation | 66 |

| | | |
|----------|--|----|
| 5.19.4.1 | data | 66 |
| 5.19.4.2 | dataBytes | 66 |
| 5.19.4.3 | target | 66 |
| 5.19.4.4 | type | 66 |
| 5.19.5 | Property Documentation | 67 |
| 5.19.5.1 | name | 67 |
| 5.20 | uConstruct.BuildingBlueprintData Class Reference | 67 |
| 5.21 | uConstruct.BuildingEditor Class Reference | 67 |
| 5.22 | uConstruct.Core.AOI.BuildingGroupAOITarget Class Reference | 68 |
| 5.22.1 | Detailed Description | 69 |
| 5.22.2 | Member Function Documentation | 69 |
| 5.22.2.1 | GroupBuildingAdded(BaseBuilding building) | 69 |
| 5.22.2.2 | GroupBuildingRemoved(BaseBuilding building) | 70 |
| 5.22.2.3 | HandleAOI(BaseAOIFinder finder, bool _inRange) | 70 |
| 5.22.2.4 | InZone(Vector3 finderPos, float radius) | 70 |
| 5.22.2.5 | OnDisable() | 70 |
| 5.22.2.6 | OnDrawGizmos() | 70 |
| 5.22.2.7 | OnEnable() | 71 |
| 5.22.3 | Member Data Documentation | 71 |
| 5.22.3.1 | buildingGroup | 71 |
| 5.22.3.2 | maxPointOnGroup | 71 |
| 5.22.3.3 | radiusAdjuster | 71 |
| 5.22.4 | Property Documentation | 71 |
| 5.22.4.1 | correctPosition | 71 |
| 5.22.4.2 | totalVectors | 71 |
| 5.23 | uConstruct.Core.Saving.BuildingGroupSaveData Class Reference | 71 |
| 5.23.1 | Detailed Description | 72 |
| 5.23.2 | Member Function Documentation | 72 |
| 5.23.2.1 | Load(BaseUCSaveData _data) | 72 |
| 5.23.3 | Member Data Documentation | 73 |

| | | |
|-----------|---|----|
| 5.23.3.1 | initialBuildingAction | 73 |
| 5.24 | uConstruct.BuildingMaterialData Struct Reference | 73 |
| 5.25 | uConstruct.Demo.BuildingPlacer Class Reference | 73 |
| 5.25.1 | Detailed Description | 75 |
| 5.25.2 | Member Function Documentation | 75 |
| 5.25.2.1 | ApplyControlsToDemoUI() | 75 |
| 5.25.2.2 | Awake() | 75 |
| 5.25.2.3 | CreateBuildingInstance(GameObject building) | 75 |
| 5.25.2.4 | DestroyBuilding(BaseBuilding building, RaycastHit hit) | 76 |
| 5.25.2.5 | DestroyCurrentBuilding() | 76 |
| 5.25.2.6 | GetInputs() | 76 |
| 5.25.2.7 | HandlePlacingResults(BaseBuilding building, bool results) | 76 |
| 5.25.2.8 | PlaceBuilding() | 76 |
| 5.25.2.9 | ResetBuildingInstance() | 76 |
| 5.25.2.10 | ReturnAlphaKey(int key) | 76 |
| 5.25.2.11 | Start() | 77 |
| 5.25.2.12 | Update() | 77 |
| 5.25.3 | Member Data Documentation | 77 |
| 5.25.3.1 | _defaultLockCursor | 77 |
| 5.25.3.2 | _destroyBuildings | 77 |
| 5.25.3.3 | _rotatedWithPlayer | 77 |
| 5.25.3.4 | _rotationValue | 77 |
| 5.26 | uConstruct.Core.Saving.BuildingSaveData Class Reference | 77 |
| 5.26.1 | Detailed Description | 78 |
| 5.27 | uConstruct.CodeGenerator.BuildingTypesCodeGenerator Class Reference | 78 |
| 5.28 | uConstruct.CodeGenerator.BuildingTypesCodeGeneratorEditor Class Reference | 79 |
| 5.29 | uConstruct.Conditions.CheckForBuilding_BlueprintData Class Reference | 79 |
| 5.30 | uConstruct.Conditions.CheckForBuildingCondition Class Reference | 80 |
| 5.30.1 | Detailed Description | 80 |
| 5.30.2 | Member Function Documentation | 81 |

| | | |
|----------|---|----|
| 5.30.2.1 | CheckCondition() | 81 |
| 5.30.2.2 | OnDrawGizmos() | 81 |
| 5.30.2.3 | Pack() | 81 |
| 5.31 | CheckForBuildingsEditor Class Reference | 81 |
| 5.32 | uConstruct.Conditions.CheckForCollision_BlueprintData Class Reference | 82 |
| 5.33 | uConstruct.Conditions.CheckForCollisionCondition Class Reference | 82 |
| 5.33.1 | Detailed Description | 83 |
| 5.33.2 | Member Function Documentation | 83 |
| 5.33.2.1 | Awake() | 83 |
| 5.33.2.2 | CheckCondition() | 83 |
| 5.33.2.3 | Pack() | 84 |
| 5.34 | uConstruct.Conditions.CheckForGround_BlueprintData Class Reference | 84 |
| 5.35 | uConstruct.Conditions.CheckForGroundCondition Class Reference | 84 |
| 5.35.1 | Detailed Description | 85 |
| 5.35.2 | Member Function Documentation | 85 |
| 5.35.2.1 | Awake() | 85 |
| 5.35.2.2 | CheckCondition() | 86 |
| 5.35.2.3 | OnDrawGizmos() | 86 |
| 5.35.2.4 | Pack() | 86 |
| 5.36 | uConstruct.Demo.DemoUI Class Reference | 86 |
| 5.36.1 | Detailed Description | 87 |
| 5.36.2 | Member Function Documentation | 87 |
| 5.36.2.1 | AddControl(string name) | 87 |
| 5.36.2.2 | Awake() | 87 |
| 5.36.2.3 | Inspect(string text) | 87 |
| 5.36.2.4 | ResetControl() | 88 |
| 5.36.3 | Member Data Documentation | 88 |
| 5.36.3.1 | controls | 88 |
| 5.36.3.2 | controlsCount | 88 |
| 5.36.3.3 | instance | 88 |

| | | |
|----------|---|----|
| 5.37 | uConstruct.Extensions.ExtensionsEditor Class Reference | 88 |
| 5.38 | uConstruct.FlagsHelper Class Reference | 89 |
| 5.38.1 | Detailed Description | 89 |
| 5.38.2 | Member Function Documentation | 89 |
| 5.38.2.1 | IsBitSet(BuildingType values, BuildingType value) | 89 |
| 5.38.2.2 | IsBitSet< T >(T values, T value) | 90 |
| 5.38.2.3 | isInsideMask(int GameObjectLayer, LayerMask mask) | 90 |
| 5.39 | uConstruct.Conditions.HeightsData Class Reference | 90 |
| 5.40 | uConstruct.Core.Blueprints.IBlueprintItem Interface Reference | 91 |
| 5.40.1 | Detailed Description | 91 |
| 5.40.2 | Member Function Documentation | 91 |
| 5.40.2.1 | Pack() | 91 |
| 5.40.3 | Property Documentation | 92 |
| 5.40.3.1 | priority | 92 |
| 5.41 | uConstruct.IBuilding Interface Reference | 92 |
| 5.41.1 | Detailed Description | 92 |
| 5.41.2 | Member Function Documentation | 92 |
| 5.41.2.1 | DestroyBuilding() | 92 |
| 5.42 | uConstruct.IPlacingModifier Interface Reference | 92 |
| 5.43 | uConstruct.Core.Templates.ITemplateObject Interface Reference | 93 |
| 5.44 | uConstruct.Core.Threading.IThreadTask Interface Reference | 93 |
| 5.44.1 | Detailed Description | 93 |
| 5.45 | uConstruct.Core.Physics.IUTCPysicsIgnored Interface Reference | 94 |
| 5.45.1 | Detailed Description | 94 |
| 5.46 | uConstruct.LayersData Class Reference | 94 |
| 5.46.1 | Detailed Description | 95 |
| 5.46.2 | Property Documentation | 95 |
| 5.46.2.1 | BuildingLayers | 95 |
| 5.46.2.2 | BuildingMask | 95 |
| 5.46.2.3 | DefaultBuildingLayer | 96 |

| | | |
|----------|--|-----|
| 5.46.2.4 | DefaultBuildingLayerString | 96 |
| 5.46.2.5 | DefaultSocketLayer | 96 |
| 5.46.2.6 | DefaultSocketLayerString | 96 |
| 5.46.2.7 | SocketLayers | 96 |
| 5.46.2.8 | SocketMask | 96 |
| 5.47 | uConstruct.LayersEditor Class Reference | 96 |
| 5.48 | uConstruct.PhysicsObjectEditor Class Reference | 97 |
| 5.49 | uConstruct.Extensions.PCloudExtension.PlayerInstantiator Class Reference | 98 |
| 5.50 | uConstruct.Core.PrefabDatabase.PrefabData Class Reference | 98 |
| 5.50.1 | Detailed Description | 98 |
| 5.51 | uConstruct.Core.PrefabDatabase.PrefabDatabaseEditor Class Reference | 98 |
| 5.51.1 | Member Function Documentation | 99 |
| 5.51.1.1 | UpdateDB() | 99 |
| 5.52 | uConstruct.Core.PrefabDatabase.PrefabDB Class Reference | 99 |
| 5.52.1 | Detailed Description | 100 |
| 5.52.2 | Member Function Documentation | 100 |
| 5.52.2.1 | AddToDB(GameObject go, int UID) | 100 |
| 5.52.2.2 | AddToDB(GameObject go) | 100 |
| 5.52.2.3 | Contains(int uid) | 101 |
| 5.52.2.4 | GetGO(int prefabID) | 101 |
| 5.52.2.5 | GetGO(BuildingType type) | 101 |
| 5.52.2.6 | RemoveFromDB(GameObject go) | 101 |
| 5.52.2.7 | ResetDB() | 102 |
| 5.52.2.8 | ReturnUID() | 102 |
| 5.52.2.9 | ReturnUID(int initial) | 102 |
| 5.53 | uConstruct.Core.PrefabDatabase.PrefabDBCustomEditor Class Reference | 102 |
| 5.54 | uConstruct.PreviewBuilding Class Reference | 103 |
| 5.54.1 | Detailed Description | 103 |
| 5.54.2 | Member Function Documentation | 103 |
| 5.54.2.1 | ApplyChangesToPrefab(GameObject prefab) | 103 |

| | | |
|----------|--|-----|
| 5.54.2.2 | FitToLocalSpace() | 103 |
| 5.54.3 | Member Data Documentation | 103 |
| 5.54.3.1 | previewPrefab | 104 |
| 5.55 | uConstruct.PreviewBuildingEditor Class Reference | 104 |
| 5.56 | uConstruct.PropertyCreatorEditor Class Reference | 104 |
| 5.57 | uConstruct.Conditions.TerrainModificationCondition.RestoreData Class Reference | 105 |
| 5.58 | uConstruct.Core.Saving.SaveDrawer Class Reference | 105 |
| 5.59 | uConstruct.SavingEditor Class Reference | 106 |
| 5.60 | uConstruct.Core.Saving.SerializeableQuaternion Class Reference | 106 |
| 5.60.1 | Detailed Description | 107 |
| 5.61 | uConstruct.Core.Saving.SerializeableVector3 Class Reference | 107 |
| 5.61.1 | Detailed Description | 107 |
| 5.62 | uConstruct.Sockets.SnapPointEditor Class Reference | 107 |
| 5.63 | uConstruct.Sockets.SocketBuildingData Class Reference | 108 |
| 5.64 | uConstruct.SocketEditor Class Reference | 108 |
| 5.65 | uConstruct.Sockets.SOverlapThreshold Class Reference | 109 |
| 5.66 | uConstruct.Core.Templates.Template Class Reference | 110 |
| 5.67 | uConstruct.Core.Templates.TemplateCreatorEditor Class Reference | 110 |
| 5.68 | uConstruct.Core.Templates.TemplateCreationData Class Reference | 111 |
| 5.69 | uConstruct.Core.Templates.TemplateMenuEditor Class Reference | 111 |
| 5.70 | uConstruct.Core.Templates.TemplateObjectSelection Class Reference | 112 |
| 5.71 | uConstruct.Core.Templates.TemplateSelectionWindow Class Reference | 112 |
| 5.72 | uConstruct.Core.Templates.TemplateUtility Class Reference | 112 |
| 5.72.1 | Member Function Documentation | 113 |
| 5.72.1.1 | GenerateTemplate(string name, BaseBuilding building, ITemplateObject[] templateTargets, bool copy) | 113 |
| 5.72.2 | Member Data Documentation | 113 |
| 5.72.2.1 | RESOURCES_PATH | 113 |
| 5.72.3 | Property Documentation | 113 |
| 5.72.3.1 | PREFAB_PATH | 113 |
| 5.73 | uConstruct.Conditions.TerrainModification_BlueprintData Class Reference | 114 |

| | | |
|----------|---|-----|
| 5.74 | uConstruct.Conditions.TerrainModificationCondition Class Reference | 114 |
| 5.74.1 | Detailed Description | 116 |
| 5.74.2 | Member Function Documentation | 116 |
| 5.74.2.1 | AddModificationData(Terrain terrain, float[,] heights) | 116 |
| 5.74.2.2 | AddModificationData(Terrain terrain, int[,] details, int layersIndex) | 116 |
| 5.74.2.3 | Awake() | 116 |
| 5.74.2.4 | CheckCondition() | 116 |
| 5.74.2.5 | OnDrawGizmos() | 117 |
| 5.74.2.6 | Pack() | 117 |
| 5.74.2.7 | RestoreTerrainModifications() | 117 |
| 5.74.2.8 | RevertModifications() | 117 |
| 5.75 | uConstruct.Conditions.TerrainModificationData Class Reference | 117 |
| 5.76 | uConstruct.Core.Threading.ThreadManager Class Reference | 118 |
| 5.76.1 | Detailed Description | 118 |
| 5.76.2 | Member Function Documentation | 118 |
| 5.76.2.1 | RunOnUConstructThread(IThreadTask action) | 118 |
| 5.76.2.2 | RunOnUnityThread(IThreadTask action) | 119 |
| 5.76.2.3 | RunThread() | 119 |
| 5.76.2.4 | StartThread() | 119 |
| 5.76.2.5 | StopThread() | 119 |
| 5.76.2.6 | UnityThread() | 119 |
| 5.76.2.7 | UpdateThread() | 119 |
| 5.76.3 | Member Data Documentation | 119 |
| 5.76.3.1 | _enabled | 119 |
| 5.76.3.2 | isRunning | 119 |
| 5.76.3.3 | isUpdate | 120 |
| 5.76.3.4 | thread | 120 |
| 5.76.3.5 | UConstructThreadActions | 120 |
| 5.76.3.6 | UnityThreadQueuedActions | 120 |
| 5.77 | uConstruct.Core.Threading.ThreadTask Class Reference | 120 |

| | |
|---|-----|
| 5.77.1 Detailed Description | 120 |
| 5.78 uConstruct.Core.Threading.ThreadTask Class Reference | 120 |
| 5.78.1 Detailed Description | 120 |
| 5.79 uConstruct.Core.Threading.ThreadTask Class Reference | 121 |
| 5.79.1 Detailed Description | 121 |
| 5.80 uConstruct.Core.Threading.ThreadTask Class Reference | 121 |
| 5.80.1 Detailed Description | 121 |
| 5.81 uConstruct.TransformExtensions Class Reference | 121 |
| 5.81.1 Detailed Description | 122 |
| 5.81.2 Member Function Documentation | 122 |
| 5.81.2.1 FloatPercisionEquals(this Vector3 a, Vector3 b) | 122 |
| 5.81.2.2 GetBackwards(this Transform transform) | 122 |
| 5.81.2.3 GetDown(this Transform transform) | 122 |
| 5.81.2.4 GetForward(this Transform transform) | 123 |
| 5.81.2.5 GetLeft(this Transform transform) | 124 |
| 5.81.2.6 GetRendererCenter(this Transform transform) | 124 |
| 5.81.2.7 GetRendererSize(this Transform transform) | 124 |
| 5.81.2.8 GetRenderersSum(this Transform transform) | 125 |
| 5.81.2.9 GetRight(this Transform transform) | 125 |
| 5.81.2.10 GetUp(this Transform transform) | 125 |
| 5.81.2.11 Subside(this Quaternion a, Quaternion b) | 125 |
| 5.82 uConstruct.UC_EditorUtility Class Reference | 126 |
| 5.82.1 Member Function Documentation | 126 |
| 5.82.1.1 DisplayScriptField(Editor editor) | 126 |
| 5.83 uConstruct.Core.Manager.UCCallbacksManager Class Reference | 126 |
| 5.83.1 Detailed Description | 127 |
| 5.83.2 Member Function Documentation | 127 |
| 5.83.2.1 AddApplicationQuitAction(System.Action action) | 127 |
| 5.83.2.2 Awake() | 128 |
| 5.83.2.3 CreateAndInitialize() | 128 |

| | | |
|-----------|--|-----|
| 5.83.2.4 | OnApplicationQuit() | 128 |
| 5.83.2.5 | Start() | 128 |
| 5.83.2.6 | Update() | 128 |
| 5.83.3 | Member Data Documentation | 128 |
| 5.83.3.1 | ProjectName | 128 |
| 5.83.4 | Property Documentation | 128 |
| 5.83.4.1 | ProjectPath | 128 |
| 5.84 | uConstruct.Demo.uConstruct_FirstPersonController Class Reference | 129 |
| 5.85 | uConstruct.Demo.uConstruct_MouseLook Class Reference | 130 |
| 5.86 | uConstruct.UConstructManager Class Reference | 130 |
| 5.87 | uConstruct.Core.Physics.UCPhysics Class Reference | 131 |
| 5.87.1 | Detailed Description | 132 |
| 5.87.2 | Member Function Documentation | 132 |
| 5.87.2.1 | AddPhysicsObject(UCPhysicsObject pObject) | 132 |
| 5.87.2.2 | Raycast(Vector3 origin, Vector3 direction, out UCPhysicsHit hit, float distance, int mask, float offset) | 132 |
| 5.87.2.3 | Raycast(Vector3 origin, Vector3 direction, out UCPhysicsHit hit, float distance, int mask) | 132 |
| 5.87.2.4 | Raycast(Vector3 origin, Vector3 direction, out UCPhysicsHit hit, float distance) | 133 |
| 5.87.2.5 | Raycast(Ray ray, out UCPhysicsHit hit, float distance, bool TakeUnityPhysics↔ IntoAccount, Transform target) | 133 |
| 5.87.2.6 | Raycast(Ray ray, out UCPhysicsHit hit, float distance, int mask) | 133 |
| 5.87.2.7 | RaycastAll(Vector3 origin, Vector3 direction, float distance, int mask, float offset) | 134 |
| 5.87.2.8 | RaycastAll(Ray ray, float distance, int mask) | 134 |
| 5.87.2.9 | RaycastAll(Ray ray, float distance, int mask, float offset) | 134 |
| 5.87.2.10 | RemovePhysicsObject(UCPhysicsObject pObject) | 135 |
| 5.88 | uConstruct.Core.Physics.UCPhysicsHit Class Reference | 135 |
| 5.88.1 | Detailed Description | 135 |
| 5.88.2 | Member Function Documentation | 135 |
| 5.88.2.1 | Convert(RaycastHit hit) | 135 |
| 5.89 | uConstruct.Core.Physics.UCPhysicsHitsArray Class Reference | 136 |
| 5.89.1 | Detailed Description | 136 |

| | | |
|----------|---|-----|
| 5.90 | uConstruct.Core.Physics.UCPhysicsObject Class Reference | 136 |
| 5.90.1 | Detailed Description | 137 |
| 5.90.2 | Member Function Documentation | 137 |
| 5.90.2.1 | DrawShape(Matrix4x4 matrix, bool selected) | 137 |
| 5.90.2.2 | OnDisable() | 138 |
| 5.90.2.3 | OnDrawGizmos() | 138 |
| 5.90.2.4 | OnEnable() | 138 |
| 5.90.2.5 | Raycast(Ray ray, out UCPhysicsHit _hit, LayerMask mask) | 138 |
| 5.90.2.6 | Start() | 138 |
| 5.90.2.7 | UpdateBounds(Vector3 center, Vector3 size) | 138 |
| 5.91 | uConstruct.Core.Saving.UCSavedItem Interface Reference | 139 |
| 5.91.1 | Detailed Description | 139 |
| 5.91.2 | Member Function Documentation | 139 |
| 5.91.2.1 | Save() | 139 |
| 5.92 | uConstruct.Core.Saving.UCSavingManager Class Reference | 139 |
| 5.92.1 | Detailed Description | 141 |
| 5.92.2 | Member Function Documentation | 141 |
| 5.92.2.1 | DeserializeStream(Stream stream) | 141 |
| 5.92.2.2 | DestoryDataOnCurrentScene() | 141 |
| 5.92.2.3 | Load() | 141 |
| 5.92.2.4 | LoadExternalData(Stream stream) | 141 |
| 5.92.2.5 | ReturnSaveData() | 141 |
| 5.92.2.6 | Save() | 142 |
| 5.92.2.7 | Serialize(Stream stream, out BaseUCSaveData[] data) | 142 |
| 5.92.2.8 | Serialize(Stream stream) | 142 |
| 5.92.3 | Member Data Documentation | 142 |
| 5.92.3.1 | fileFormat | 142 |
| 5.92.3.2 | fileName | 142 |
| 5.92.4 | Property Documentation | 142 |
| 5.92.4.1 | dataPath | 142 |
| 5.92.4.2 | enabled | 143 |
| 5.92.4.3 | folderPath | 143 |
| 5.93 | uConstruct.UCSettingAttribute Class Reference | 143 |
| 5.94 | uConstruct.UCSettingCategory Class Reference | 144 |
| 5.95 | uConstruct.UCSettings Class Reference | 144 |
| 5.96 | uConstruct.UCSettingsEditor Class Reference | 145 |

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

| | |
|---|----|
| uConstruct | 11 |
| uConstruct.CodeGenerator | 13 |
| uConstruct.Conditions | 13 |
| uConstruct.Core | 14 |
| uConstruct.Core.AOI | 14 |
| uConstruct.Core.Blueprints | 14 |
| uConstruct.Core.Manager | 14 |
| uConstruct.Core.Physics | 15 |
| uConstruct.Core.PrefabDatabase | 15 |
| uConstruct.Core.Saving | 15 |
| uConstruct.Core.Templates | 16 |
| uConstruct.Core.Threading | 16 |
| uConstruct.Demo | 16 |
| uConstruct.Extensions | 17 |
| uConstruct.Extensions.PCloudExtension | 17 |
| uConstruct.Sockets | 17 |

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

| | |
|---|-----|
| uConstruct.Core.AOI.AOIManager | 19 |
| Attribute | |
| uConstruct.UCSettingAttribute | 143 |
| uConstruct.Core.Saving.BaseUCSaveData | 53 |
| uConstruct.Core.Saving.BuildingGroupSaveData | 71 |
| uConstruct.BatchClass | 54 |
| uConstruct.BatchData | 55 |
| uConstruct.BatchExtensions | 57 |
| uConstruct.BatchUtility | 57 |
| uConstruct.Core.Blueprints.BlueprintData | 61 |
| uConstruct.BuildingBlueprintData | 67 |
| uConstruct.Conditions.CheckForBuilding_BlueprintData | 79 |
| uConstruct.Conditions.CheckForCollision_BlueprintData | 82 |
| uConstruct.Conditions.CheckForGround_BlueprintData | 84 |
| uConstruct.Conditions.TerrainModification_BlueprintData | 114 |
| uConstruct.Sockets.SocketBuildingData | 108 |
| uConstruct.BuildingMaterialData | 73 |
| uConstruct.Core.Saving.BuildingSaveData | 77 |
| uConstruct.CodeGenerator.BuildingTypesCodeGenerator | 78 |
| Editor | |
| CheckForBuildingsEditor | 81 |
| uConstruct.BuildingEditor | 67 |
| uConstruct.Core.PrefabDatabase.PrefabDBCustomeEditor | 102 |
| uConstruct.PhysicsObjectEditor | 97 |
| uConstruct.SocketEditor | 108 |
| uConstruct.PreviewBuildingEditor | 104 |
| uConstruct.SavingEditor | 106 |
| uConstruct.Sockets.SnapPointEditor | 107 |
| EditorWindow | |
| BlueprintEditEditor | 62 |
| BlueprintEditor | 63 |
| uConstruct.CodeGenerator.BuildingTypesCodeGeneratorEditor | 79 |
| uConstruct.Core.PrefabDatabase.PrefabDatabaseEditor | 98 |
| uConstruct.Core.Templates.TemplateCreatorEditor | 110 |
| uConstruct.Core.Templates.TemplateCreatorEditor | 110 |

| | |
|--|-----|
| uConstruct.Core.Templates.TemplateCreatorEditor | 110 |
| uConstruct.Core.Templates.TemplateCreatorEditor | 110 |
| uConstruct.Core.Templates.TemplateMenuEditor | 111 |
| uConstruct.Core.Templates.TemplateSelectionWindow | 112 |
| uConstruct.Extensions.ExtensionsEditor | 88 |
| uConstruct.LayersEditor | 96 |
| uConstruct.PropertyCreatorEditor | 104 |
| uConstruct.UConstructManager | 130 |
| uConstruct.UCSettingsEditor | 145 |
| uConstruct.FlagsHelper | 89 |
| uConstruct.Conditions.HeightsData | 90 |
| uConstruct.Core.Blueprints.IBlueprintItem | 91 |
| uConstruct.BaseBuilding | 25 |
| uConstruct.Conditions.BaseCondition | 43 |
| uConstruct.Conditions.CheckForBuildingCondition | 80 |
| uConstruct.Conditions.CheckForCollisionCondition | 82 |
| uConstruct.Conditions.CheckForGroundCondition | 84 |
| uConstruct.Conditions.TerrainModificationCondition | 114 |
| uConstruct.Sockets.BaseSocket | 49 |
| uConstruct.IBuilding | 92 |
| uConstruct.BaseBuilding | 25 |
| uConstruct.BaseBuildingBatcher | 37 |
| uConstruct.IPlacingModifier | 92 |
| uConstruct.Sockets.BaseSocket | 49 |
| ISerializationCallbackReceiver | |
| uConstruct.Core.Blueprints.BlueprintField | 63 |
| uConstruct.Core.Templates.ITemplateObject | 93 |
| uConstruct.Conditions.BaseCondition | 43 |
| uConstruct.Sockets.BaseSocket | 49 |
| uConstruct.Core.Threading.IThreadTask | 93 |
| uConstruct.Core.Threading.ThreadTask< T, T1, T2 > | 121 |
| uConstruct.Core.Threading.ThreadTask< T, T1, T2 > | 121 |
| uConstruct.Core.Threading.ThreadTask< T, T1, T2 > | 121 |
| uConstruct.Core.Threading.ThreadTask< T, T1, T2 > | 121 |
| uConstruct.Core.Physics.IUTCPPhysicsIgnored | 94 |
| uConstruct.BaseBuilding | 25 |
| uConstruct.Conditions.BaseCondition | 43 |
| MonoBehaviour | |
| uConstruct.BaseBuilding | 25 |
| uConstruct.BaseBuilding | 25 |
| uConstruct.BaseBuildingBatcher | 37 |
| uConstruct.Conditions.BaseCondition | 43 |
| uConstruct.Conditions.BaseCondition | 43 |
| uConstruct.Core.AOI.BaseAOIFinder | 22 |
| uConstruct.Core.AOI.BaseAOITarget | 23 |
| uConstruct.Core.AOI.BuildingGroupAOITarget | 68 |
| uConstruct.BaseBuildingGroup | 38 |
| uConstruct.Core.Manager.UCCallbacksManager | 126 |
| uConstruct.Core.Physics.UCPhysicsObject | 136 |
| uConstruct.Sockets.BaseSocket | 49 |
| uConstruct.Sockets.BaseSocket | 49 |
| uConstruct.Core.Saving.SaveDrawer | 105 |
| uConstruct.Core.Templates.Template | 110 |
| uConstruct.Core.Templates.TemplateUtility | 112 |
| uConstruct.Demo.BuildingPlacer | 73 |
| uConstruct.Demo.DemoUI | 86 |

| | |
|--|-----|
| uConstruct.Demo.uConstruct_FirstPersonController | 129 |
| uConstruct.PreviewBuilding | 103 |
| uConstruct.Sockets.BaseSnapPoint | 46 |
| uConstruct.Core.PrefabDatabase.PrefabData | 98 |
| PunBehaviour | |
| uConstruct.Extensions.PCloudExtension.PlayerInstantiator | 98 |
| uConstruct.Conditions.TerrainModificationCondition.RestoreData | 105 |
| ScriptableObject | |
| uConstruct.Core.Blueprints.Blueprint | 59 |
| uConstruct.Core.PrefabDatabase.PrefabDB | 99 |
| uConstruct.LayersData | 94 |
| uConstruct.UCSettings | 144 |
| uConstruct.Core.Saving.SerializeableQuaternion | 106 |
| uConstruct.Core.Saving.SerializeableVector3 | 107 |
| uConstruct.Sockets.SOverlapThreshold | 109 |
| uConstruct.Core.Templates.TemplateCreationData | 111 |
| uConstruct.Core.Templates.TemplateObjectSelection | 112 |
| uConstruct.Conditions.TerrainModificationData | 117 |
| uConstruct.Core.Threading.ThreadManager | 118 |
| uConstruct.Core.Threading.ThreadTask< T, T1 > | 121 |
| uConstruct.Core.Threading.ThreadTask | 121 |
| uConstruct.Core.Threading.ThreadTask< T > | 121 |
| uConstruct.TransformExtensions | 121 |
| uConstruct.UC_EditorUtility | 126 |
| uConstruct.Demo.uConstruct_MouseLook | 130 |
| uConstruct.Core.Physics.UCPhysics | 131 |
| uConstruct.Core.Physics.UCPhysicsHit | 135 |
| uConstruct.Core.Physics.UCPhysicsHitsArray | 136 |
| uConstruct.Core.Saving.UCSavedItem | 139 |
| uConstruct.BaseBuildingGroup | 38 |
| uConstruct.Core.Saving.UCSavingManager | 139 |
| uConstruct.UCSettingCategory | 144 |

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

| | |
|---|----|
| uConstruct.Core.AOI.AOIManager | |
| The AOI Class that handles all of the AOI management. | 19 |
| uConstruct.Core.AOI.BaseAOIFinder | |
| A base aoi finder class | 22 |
| uConstruct.Core.AOI.BaseAOITarget | |
| Base AOITarget class. | 23 |
| uConstruct.BaseBuilding | |
| A base building script that has all building methods. Incase of making an another type of use for the building like a mine/ a networked building please inherite from this class. | 25 |
| uConstruct.BaseBuildingBatcher | |
| A class that is attached to the bathed collider. used to contain data about the group we are batching. | 37 |
| uConstruct.BaseBuildingGroup | |
| A base class for groups. Handles all group management. incase of doing another use of groups please inherite from this class. | 38 |
| uConstruct.Conditions.BaseCondition | |
| A base condition that should be inherited from when creating conditions. | 43 |
| uConstruct.Sockets.BaseSnapPoint | |
| Snap points are points on your building which are used for dynamically choosing an anchor for placing the building based on distances. | 46 |
| uConstruct.Sockets.BaseSocket | |
| Base class for sockets. inherite from this class if you want to do any code adjustments. | 49 |
| uConstruct.Core.Saving.BaseUCSaveData | |
| A base class for saving data, inherite from this class when ever you want to create a custom save data | 53 |
| uConstruct.BatchClass | |
| A class that handles all batch data | 54 |
| uConstruct.BatchData | |
| Our batch data | 55 |
| uConstruct.BatchExtensions | |
| Extension class for the mesh class. | 57 |
| uConstruct.BatchUtility | |
| This class handles the batching mechanic of uConstruct . Can be used with other system as well if needed. | 57 |

| | |
|--|----|
| uConstruct.Core.Blueprints.Blueprint | |
| Blueprints are a set of data that allows you to quickly create a set of data that can be applied on any kind of a building with not efforts. | 59 |
| uConstruct.Core.Blueprints.BlueprintData | |
| A serializeable data class that needs to be inherited from on any data that can be serialized into the blueprint system. | 61 |
| BlueprintEditEditor | 62 |
| BlueprintEditor | 63 |
| uConstruct.Core.Blueprints.BlueprintField | |
| Blueprint field holds data about the blueprint. | 63 |
| uConstruct.BuildingBlueprintData | 67 |
| uConstruct.BuildingEditor | 67 |
| uConstruct.Core.AOI.BuildingGroupAOITarget | |
| The building version of AOITarget | 68 |
| uConstruct.Core.Saving.BuildingGroupSaveData | |
| This is a class that holds data for all the group save data. used for saving groups. | 71 |
| uConstruct.BuildingMaterialData | 73 |
| uConstruct.Demo.BuildingPlacer | |
| A demo script that comes with the asset to place buildings. | 73 |
| uConstruct.Core.Saving.BuildingSaveData | |
| Save data class for the group | 77 |
| uConstruct.CodeGenerator.BuildingTypesCodeGenerator | 78 |
| uConstruct.CodeGenerator.BuildingTypesCodeGeneratorEditor | 79 |
| uConstruct.Conditions.CheckForBuilding_BlueprintData | 79 |
| uConstruct.Conditions.CheckForBuildingCondition | |
| A basic condition that comes with the asset, checks if there is an building that you specify in the editor infront of the condition in the distance specified. | 80 |
| CheckForBuildingsEditor | 81 |
| uConstruct.Conditions.CheckForCollision_BlueprintData | 82 |
| uConstruct.Conditions.CheckForCollisionCondition | |
| This class is a built-in condition that comes with the asset. it checks for any collision while placing the object, to make sure you arent placing buildings inside buildings and so on. | 82 |
| uConstruct.Conditions.CheckForGround_BlueprintData | 84 |
| uConstruct.Conditions.CheckForGroundCondition | |
| A basic built-in condition that checks if the building has ground. if it doesnt it will add gravity to the object and remove him from the group (at the end, destroy it). | 84 |
| uConstruct.Demo.DemoUI | |
| A simple demo class that handles the building placer controls | 86 |
| uConstruct.Extensions.ExtensionsEditor | 88 |
| uConstruct.FlagsHelper | |
| Some helper classes for bitmasks | 89 |
| uConstruct.Conditions.HeightsData | 90 |
| uConstruct.Core.Blueprints.IBlueprintItem | |
| An interface that each one of the blueprinted items should have. | 91 |
| uConstruct.IBuilding | |
| An interface for all buildings. | 92 |
| uConstruct.IPlacingModifier | 92 |
| uConstruct.Core.Templates.ITemplateObject | 93 |
| uConstruct.Core.Threading.IThreadTask | |
| A thread task interface. Implement on any customely created thread task. | 93 |
| uConstruct.Core.Physics.IUTCPysicsIgnored | |
| Ignore all physics on this script. | 94 |
| uConstruct.LayersData | |
| A class that contains information about custom layers data of the asset. this data is used when initiating layers assigning and so on. | 94 |
| uConstruct.LayersEditor | 96 |
| uConstruct.PhysicsObjectEditor | 97 |
| uConstruct.Extensions.PCloudExtension.PlayerInstantiator | 98 |

| | |
|--|-----|
| uConstruct.Core.PrefabDatabase.PrefabData | |
| Holds all the data for a prefab | 98 |
| uConstruct.Core.PrefabDatabase.PrefabDatabaseEditor | 98 |
| uConstruct.Core.PrefabDatabase.PrefabDB | |
| This class handles all prefab database in the system. | 99 |
| uConstruct.Core.PrefabDatabase.PrefabDBCustomEditor | 102 |
| uConstruct.PreviewBuilding | |
| A class that is attached to the socket preview object to contain data about the prefab and apply changes to the prefab. | 103 |
| uConstruct.PreviewBuildingEditor | 104 |
| uConstruct.PropertyCreatorEditor | 104 |
| uConstruct.Conditions.TerrainModificationCondition.RestoreData | 105 |
| uConstruct.Core.Saving.SaveDrawer | 105 |
| uConstruct.SavingEditor | 106 |
| uConstruct.Core.Saving.SerializeableQuaternion | |
| A serializeable version of quaternion | 106 |
| uConstruct.Core.Saving.SerializeableVector3 | |
| A serializeable version of the vector3 | 107 |
| uConstruct.Sockets.SnapPointEditor | 107 |
| uConstruct.Sockets.SocketBuildingData | 108 |
| uConstruct.SocketEditor | 108 |
| uConstruct.Sockets.SOverlapThreshold | 109 |
| uConstruct.Core.Templates.Template | 110 |
| uConstruct.Core.Templates.TemplateCreatorEditor | 110 |
| uConstruct.Core.Templates.TemplateCreationData | 111 |
| uConstruct.Core.Templates.TemplateMenuEditor | 111 |
| uConstruct.Core.Templates.TemplateObjectSelection | 112 |
| uConstruct.Core.Templates.TemplateSelectionWindow | 112 |
| uConstruct.Core.Templates.TemplateUtility | 112 |
| uConstruct.Conditions.TerrainModification_BlueprintData | 114 |
| uConstruct.Conditions.TerrainModificationCondition | |
| This condition is a built-in condition that will clean details around you on place. Should be used for stuff like foundations. | 114 |
| uConstruct.Conditions.TerrainModificationData | 117 |
| uConstruct.Core.Threading.ThreadManager | |
| This class handles the multi-threading mechanics of uConstruct | 118 |
| uConstruct.Core.Threading.ThreadTask< T, T1 > | |
| A thread task that takes 2 parameters. | |

Template Parameters

| | |
|-----------|--------|
| <i>T</i> | Type 1 |
| <i>T1</i> | Type 2 |

121

| | |
|---|-----|
| uConstruct.Core.Threading.ThreadTask | |
| A thread task that takes no parameters. | 121 |
| uConstruct.Core.Threading.ThreadTask< T > | |
| A thread task that takes 1 parameter. | |

Template Parameters

| | |
|----------|--------|
| <i>T</i> | Type 1 |
|----------|--------|

121

| | |
|---|--|
| uConstruct.Core.Threading.ThreadTask< T, T1, T2 > | |
| A thread task that takes 3 parameters. | |

Template Parameters

| | |
|-----------|--------|
| <i>T</i> | Type 1 |
| <i>T1</i> | Type 2 |
| <i>T2</i> | Type 3 |

| | |
|---|-----|
| | 121 |
| uConstruct.TransformExtensions | |
| Extension methods for the transform | 121 |
| uConstruct.UC_EditorUtility | 126 |
| uConstruct.Core.Manager.UCCallbacksManager | |
| This class needs to be initiated on the start on the game and it handles loading and saving, it has control over all unity callbacks and you can use it to add some static OnApplicationQuit callbacks. | 126 |
| uConstruct.Demo.uConstruct_FirstPersonController | 129 |
| uConstruct.Demo.uConstruct_MouseLook | 130 |
| uConstruct.UConstructManager | 130 |
| uConstruct.Core.Physics.UCPhysics | |
| This class handles all custom physics. | 131 |
| uConstruct.Core.Physics.UCPhysicsHit | |
| A class that holds the data for the hit data | 135 |
| uConstruct.Core.Physics.UCPhysicsHitsArray | |
| An custom array that holds all ray results in an array | 136 |
| uConstruct.Core.Physics.UCPhysicsObject | |
| This is a base class for a UCPhysicsObject . Every class that inherits this class will be counted in the physics system. | 136 |
| uConstruct.Core.Saving.UCSavedItem | |
| An interface that each saveable object in the scene needs to have. | 139 |
| uConstruct.Core.Saving.UCSavingManager | |
| This class handles all the saving management of the asset. | 139 |
| uConstruct.UCSettingAttribute | 143 |
| uConstruct.UCSettingCategory | 144 |
| uConstruct.UCSettings | 144 |
| uConstruct.UCSettingsEditor | 145 |

Chapter 4

Namespace Documentation

4.1 uConstruct Namespace Reference

Namespaces

Classes

- class [BaseBuilding](#)
A base building script that has all building methods. Incase of making an another type of use for the building like a mine/ a networked building please inherit from this class.
- class [BaseBuildingBatcher](#)
A class that is attached to the batched collider. used to contain data about the group we are batching.
- class [BaseBuildingGroup](#)
A base class for groups. Handles all group management. incase of doing another use of groups please inherit from this class.
- class [BatchClass](#)
A class that handles all batch data
- class [BatchData](#)
Our batch data
- class [BatchExtensions](#)
Extension class for the mesh class.
- class [BatchUtility](#)
This class handles the batching mechanic of [uConstruct](#). Can be used with other system as well if needed.
- class [BuildingBlueprintData](#)
- class [BuildingEditor](#)
- struct [BuildingMaterialData](#)
- class [FlagsHelper](#)
Some helper classes for bitmasks
- interface [IBuilding](#)
An interface for all buildings.
- interface [IPlacingModifier](#)
- class [LayersData](#)
A class that contains information about custom layers data of the asset. this data is used when initiating layers assigning and so on.
- class [LayersEditor](#)
- class [PhysicsObjectEditor](#)

- class [PreviewBuilding](#)

A class that is attached to the socket preview object to contain data about the prefab and apply changes to the prefab.

- class [PreviewBuildingEditor](#)
- class [PropertyCreatorEditor](#)
- class [SavingEditor](#)
- class [SocketEditor](#)
- class [TransformExtensions](#)

Extension methods for the transform

- class [UC_EditorUtility](#)
- class [UConstructManager](#)
- class [UCSettingAttribute](#)
- class [UCSettingCategory](#)
- class [UCSettings](#)
- class [UCSettingsEditor](#)

Enumerations

- enum [PlacingRestrictionType](#) { **SocketBased** = 1, **FreelyBased** = 2 }

An enum that contains the 2 placing types that are allowed for a building/ socket.

- enum **SocketPositionAnchor** { **Right**, **Left**, **Forward**, **ForwardRight**, **ForwardLeft**, **Back**, **BackRight**, **BackLeft**, **Up**, **Down**, **Center** }
- enum **Axis** { **X**, **Y**, **Z** }
- enum **SavingPathType** { **Persistent**, **Data** }
- enum **UCSettingCategories** { **Saving**, **General**, **Conditions**, **Sockets**, **Blueprints**, **Utilities**, **Templates**, **Prefabs**, **Physics**, **Threading**, **Batching** }
- enum **ModifierType** { **Condition**, **SnapPoint**, **Socket** }

Functions

- delegate void **OnSnappedToSocket** ([BaseSocket](#) socket)
- delegate void **OnLostSnapToSocket** ([BaseSocket](#) socket)
- delegate void **OnPlacedOnChanged** ([BaseBuilding](#) oldValue, [BaseBuilding](#) newValue)
- delegate void **OnBuildingGroupChanged** ([BaseBuildingGroup](#) group)
- delegate void **OnNetworkInstanceLoaded** ([BaseBuilding](#) building)
- delegate void **OnNetworkInstancePacked** ([BaseBuilding](#) building)
- delegate void **OnPlaced** ()
- delegate void **OnDestroy** ()
- delegate void **OnDeattach** ()
- delegate void **OnMaterialColorChanged** ([BuildingMaterialData](#) color)
- delegate void **OnHealthChanged** ()
- delegate void **BuildingAdded** ([BaseBuilding](#) building)
- delegate void **BuildingRemoved** ([BaseBuilding](#) building)
- delegate void **GroupBuildingDestroyed** ([BaseBuilding](#) building)
- delegate void **BuildingGroupCreated** ([BaseBuildingGroup](#) group)
- delegate void **OnBatchDone** (GameObject go, Mesh mesh)
- delegate void **BatchedGroup** (bool value)

4.1.1 Enumeration Type Documentation

4.1.1.1 enum uConstruct.PlacingRestrictionType [strong]

An enum that contains the 2 placing types that are allowed for a building/ socket.

4.2 uConstruct.CodeGenerator Namespace Reference

Classes

- class [BuildingTypesCodeGenerator](#)
- class [BuildingTypesCodeGeneratorEditor](#)

4.3 uConstruct.Conditions Namespace Reference

Classes

- class [BaseCondition](#)
A base condition that should be inherited from when creating conditions.
- class [CheckForBuilding_BlueprintData](#)
- class [CheckForBuildingCondition](#)
A basic condition that comes with the asset, checks if there is an building that you specify in the editor infront of the condition in the distance specified.
- class [CheckForCollision_BlueprintData](#)
- class [CheckForCollisionCondition](#)
This class is a built-in condition that comes with the asset. it checks for any collision while placing the object, to make sure you arent placing buildings inside buildings and so on.
- class [CheckForGround_BlueprintData](#)
- class [CheckForGroundCondition](#)
A basic built-in condition that checks if the building has ground. if it doesnt it will add gravity to the object and remove him from the group (at the end, destroy it).
- class [HeightsData](#)
- class [TerrainModification_BlueprintData](#)
- class [TerrainModificationCondition](#)
This condition is a built-in condition that will clean details around you on place. Should be used for stuff like foundations.
- class [TerrainModificationData](#)

Enumerations

- enum **DetectionType** { **Raycast**, **Sphere** }
- enum **TerrainModificationType** { **ClearDetails**, **FlattenHeight** }

4.4 uConstruct.Core Namespace Reference

Namespaces

4.5 uConstruct.Core.AOI Namespace Reference

Classes

- class [AOIManager](#)
The AOI Class that handles all of the AOI management.
- class [BaseAOIFinder](#)
A base aoI finder class
- class [BaseAOITarget](#)
Base AOITarget class.
- class [BuildingGroupAOITarget](#)
The building version of AOITarget

4.6 uConstruct.Core.Blueprints Namespace Reference

Classes

- class [Blueprint](#)
Blueprints are a set of data that allows you to quickly create a set of data that can be applied on any kind of a building with not efforts.
- class [BlueprintData](#)
A serializeable data class that needs to be inherited from on any data that can be serialized into the blueprint system.
- class [BlueprintField](#)
Blueprint field holds data about the blueprint.
- interface [IBlueprintItem](#)
An interface that each one of the blueprinted items should have.

4.7 uConstruct.Core.Manager Namespace Reference

Classes

- class [UCCallbacksManager](#)
This class needs to be initiated on the start on the game and it handles loading and saving, it has control over all unity callbacks and you can use it to add some static OnApplicationQuit callbacks.

4.8 uConstruct.Core.Physics Namespace Reference

Classes

- interface [IUTCPysicsIgnored](#)
Ignore all physics on this script.
- class [UCPhysics](#)
This class handles all custom physics.
- class [UCPhysicsHit](#)
A class that holds the data for the hit data
- class [UCPhysicsHitsArray](#)
An custom array that holds all ray results in an array
- class [UCPhysicsObject](#)
This is a base class for a [UCPhysicsObject](#). Every class that inherits this class will be counted in the physics system.

4.9 uConstruct.Core.PrefabDatabase Namespace Reference

Classes

- class [PrefabData](#)
Holds all the data for a prefab
- class [PrefabDatabaseEditor](#)
- class [PrefabDB](#)
This class handles all prefab database in the system.
- class [PrefabDBCustomeEditor](#)

4.10 uConstruct.Core.Saving Namespace Reference

Classes

- class [BaseUCSaveData](#)
A base class for saving data, inherite from this class when ever you want to create a custom save data
- class [BuildingGroupSaveData](#)
This is a class that holds data for all the group save data. used for saving groups.
- class [BuildingSaveData](#)
Save data class for the group
- class [SaveDrawer](#)
- class [SerializeableQuaternion](#)
A serializeable version of quaternion
- class [SerializeableVector3](#)
A serializeable version of the vector3
- interface [UCSavedItem](#)
An interface that each saveable object in the scene needs to have.
- class [UCSavingManager](#)
This class handles all the saving management of the asset.

Functions

- delegate void **OnBuildingLoaded** ([BaseBuilding](#) building)
- delegate void **OnBuildingSaving** ([BuildingSaveData](#) data)
- delegate void **SavingProcessComplete** ()
- delegate void **LoadingProcessComplete** ()

4.11 uConstruct.Core.Templates Namespace Reference

Classes

- interface [ITemplateObject](#)
- class [Template](#)
- class [TemplateCreatorEditor](#)
- class [TemplateCreationData](#)
- class [TemplateMenuEditor](#)
- class [TemplateObjectSelection](#)
- class [TemplateSelectionWindow](#)
- class [TemplateUtility](#)

4.12 uConstruct.Core.Threading Namespace Reference

Classes

- interface [IThreadTask](#)
A thread task interface. Implement on any customely created thread task.
- class [ThreadManager](#)
This class handles the multi-threading mechanics of [uConstruct](#).
- class [ThreadTask](#)
A thread task that takes no parameters.

4.13 uConstruct.Demo Namespace Reference

Classes

- class [BuildingPlacer](#)
A demo script that comes with the asset to place buildings.
- class [DemoUI](#)
A simple demo class that handles the building placer controls
- class [uConstruct_FirstPersonController](#)
- class [uConstruct_MouseLook](#)

4.14 uConstruct.Extensions Namespace Reference

Namespaces

Classes

- class [ExtensionsEditor](#)

4.15 uConstruct.Extensions.PCloudExtension Namespace Reference

Classes

- class [PlayerInstantiator](#)

4.16 uConstruct.Sockets Namespace Reference

Classes

- class [BaseSnapPoint](#)
Snap points are points on your building which are used for dynamically choosing an anchor for placing the building based on distances.
- class [BaseSocket](#)
Base class for sockets. inherit from this class if you want to do any code adjustments.
- class [SnapPointEditor](#)
- class [SocketBuildingData](#)
- class [SOverlapThreshold](#)

Functions

- delegate void **OnPreviewObjectChanged** (GameObject go)

Chapter 5

Class Documentation

5.1 uConstruct.Core.AOI.AOIManager Class Reference

The [AOI](#) Class that handles all of the [AOI](#) management.

Static Public Member Functions

- static void [AddFinder](#) ([BaseAOIFinder](#) value)
Add a finder to the finder list
- static void [RemoveFinder](#) ([BaseAOIFinder](#) value)
Remove a finder from the finder list
- static void [AddTarget](#) ([BaseAOITarget](#) value)
Add a target to the target list
- static void [RemoveTarget](#) ([BaseAOITarget](#) value)
Remove a target from the target list
- static void [UpdateAOI](#) ([BaseAOIFinder](#) finder)
Update the [AOI](#) of the finder.

Static Private Member Functions

- static void [UpdatePositions](#) ()
This will update position on the finders and targets so the thread can read that.
- static void [ComputeAOI](#) ([BaseAOIFinder](#) finder)
Compute [AOI](#) for a certain finder
- static void [HandleAOI](#) ([BaseAOITarget](#) target, [BaseAOIFinder](#) finder, bool isUnityThread)
Handle the [AOI](#).

Static Private Attributes

- static List< [BaseAOIFinder](#) > [finders](#) = new List<[BaseAOIFinder](#)>()
[AOI](#) finders
- static List< [BaseAOITarget](#) > [targets](#) = new List<[BaseAOITarget](#)>()
[AOI](#) targets

5.1.1 Detailed Description

The [AOI](#) Class that handles all of the [AOI](#) management.

5.1.2 Member Function Documentation

5.1.2.1 `static void uConstruct.Core.AOI.AOIManager.AddFinder (BaseAOIFinder value) [static]`

Add a finder to the finder list

Parameters

| | |
|--------------|----------------------------|
| <i>value</i> | the finder you want to add |
|--------------|----------------------------|

5.1.2.2 `static void uConstruct.Core.AOI.AOIManager.AddTarget (BaseAOITarget value) [static]`

Add a target to the target list

Parameters

| | |
|--------------|----------------------------|
| <i>value</i> | the target you want to add |
|--------------|----------------------------|

5.1.2.3 `static void uConstruct.Core.AOI.AOIManager.ComputeAOI (BaseAOIFinder finder) [static], [private]`

Compute [AOI](#) for a certain finder

Parameters

| | |
|---------------|---|
| <i>finder</i> | the finder you want to compute AOI for. |
|---------------|---|

5.1.2.4 `static void uConstruct.Core.AOI.AOIManager.HandleAOI (BaseAOITarget target, BaseAOIFinder finder, bool isUnityThread) [static], [private]`

Handle the [AOI](#).

Parameters

| | |
|----------------------|---------------------------------------|
| <i>target</i> | our target |
| <i>finder</i> | our finder |
| <i>isUnityThread</i> | is this executed from unity's thread. |

5.1.2.5 `static void uConstruct.Core.AOI.AOIManager.RemoveFinder (BaseAOIFinder value) [static]`

Remove a finder from the finder list

Parameters

| | |
|--------------|-------------------------------|
| <i>value</i> | the finder you want to remove |
|--------------|-------------------------------|

5.1.2.6 `static void uConstruct.Core.AOI.AOIManager.RemoveTarget (BaseAOITarget value) [static]`

Remove a target from the target list

Parameters

| | |
|--------------|-------------------------------|
| <i>value</i> | the target you want to remove |
|--------------|-------------------------------|

5.1.2.7 `static void uConstruct.Core.AOI.AOIManager.UpdateAOI (BaseAOIFinder finder) [static]`

Update the [AOI](#) of the finder.

Parameters

| | |
|---------------|--|
| <i>finder</i> | the finder you want to update the AOI zone of. |
|---------------|--|

5.1.2.8 `static void uConstruct.Core.AOI.AOIManager.UpdatePositions () [static],[private]`

This will update position on the finders and targets so the thread can read that.

5.1.3 Member Data Documentation

5.1.3.1 `List<BaseAOIFinder> uConstruct.Core.AOI.AOIManager.finders = new List<BaseAOIFinder>() [static],[private]`

[AOI](#) finders

5.1.3.2 `List<BaseAOITarget> uConstruct.Core.AOI.AOIManager.targets = new List<BaseAOITarget>() [static],[private]`

[AOI](#) targets

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/AOI/AOIManager.cs

5.2 uConstruct.Core.AOI.BaseAOIFinder Class Reference

A base aoi finder class

Inheritance diagram for uConstruct.Core.AOI.BaseAOIFinder:

Public Member Functions

- virtual void [OnEnable](#) ()
Add the finder to the list
- virtual void [OnDisable](#) ()
Remove the finder from the list
- virtual void [UpdateAOI](#) ()
Update the [AOI](#) of the finder.
- virtual void [OnDrawGizmos](#) ()
Draw gizmos
- virtual void [Update](#) ()
Calls the update on the [AOI](#).

Public Attributes

- float [radius](#) = 20f
our searching radius
- Vector3 [aoiPosition](#)
Position that is updated by the [AOIManager](#) and used by a different thread

Private Attributes

- Vector3 [oldPos](#) = -Vector3.one
our old position, used for checking movement and update [AOI](#) only when needed.

5.2.1 Detailed Description

A base aoi finder class

5.2.2 Member Function Documentation

5.2.2.1 virtual void uConstruct.Core.AOI.BaseAOIFinder.OnDisable () [virtual]

Remove the finder from the list

5.2.2.2 virtual void uConstruct.Core.AOI.BaseAOIFinder.OnDrawGizmos () [virtual]

Draw gizmos

5.2.2.3 virtual void uConstruct.Core.AOI.BaseAOIFinder.OnEnable () [virtual]

Add the finder to the list

5.2.2.4 virtual void uConstruct.Core.AOI.BaseAOIFinder.Update () [virtual]

Calls the update on the [AOI](#).

5.2.2.5 virtual void uConstruct.Core.AOI.BaseAOIFinder.UpdateAOI () [virtual]

Update the [AOI](#) of the finder.

5.2.3 Member Data Documentation

5.2.3.1 Vector3 uConstruct.Core.AOI.BaseAOIFinder.aoiPosition

Position that is updated by the [AOIManager](#) and used by a different thread

5.2.3.2 Vector3 uConstruct.Core.AOI.BaseAOIFinder.oldPos = -Vector3.one [private]

our old position, used for checking movement and update [AOI](#) only when needed.

5.2.3.3 float uConstruct.Core.AOI.BaseAOIFinder.radius = 20f

our searching radius

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/AOI/BaseAOIFinder.cs

5.3 uConstruct.Core.AOI.BaseAOITarget Class Reference

Base AOITarget class.

Inheritance diagram for uConstruct.Core.AOI.BaseAOITarget:

Public Member Functions

- virtual void [HandleAOI](#) ([BaseAOIFinder](#) finder, bool _inRange)
Handle [AOI](#) Change
- virtual bool [InZone](#) (Vector3 finderPos, float radius)
Are we in zone of this target?

Public Attributes

- bool [inRange](#) = true
Are we in range of any of the finders
- Vector3 [aoiPosition](#)
Position that is updated by the [AOIManager](#) and used by a different thread

Protected Member Functions

- virtual void [OnEnable](#) ()
Add the target to the targets list
- virtual void [OnDisable](#) ()
Remove the target from the targets list

Properties

- virtual bool [useMultiThreadZoneSearch](#) [get]
Will the system use multi-threading to choose whether this target is in range of a finder or not.

5.3.1 Detailed Description

Base AOITarget class.

5.3.2 Member Function Documentation

5.3.2.1 virtual void `uConstruct.Core.AOI.BaseAOITarget.HandleAOI (BaseAOIFinder finder, bool _inRange)` [virtual]

Handle [AOI](#) Change

Parameters

| | |
|-----------------|--|
| <i>finder</i> | The finder that we got in range/ out of range of. |
| <i>_inRange</i> | Are we in range of the finder or out of range of the finder? |

Reimplemented in [uConstruct.Core.AOI.BuildingGroupAOITarget](#).

5.3.2.2 virtual bool `uConstruct.Core.AOI.BaseAOITarget.InZone (Vector3 finderPos, float radius)` [virtual]

Are we in zone of this target?

Parameters

| | |
|------------------|---------------------|
| <i>finderPos</i> | Our finder position |
| <i>radius</i> | Finder's radius |

Returns

Are we in range ?

Reimplemented in [uConstruct.Core.AOI.BuildingGroupAOITarget](#).

5.3.2.3 virtual void uConstruct.Core.AOI.BaseAOITarget.OnDisable () [protected], [virtual]

Remove the target from the targets list

Reimplemented in [uConstruct.Core.AOI.BuildingGroupAOITarget](#).

5.3.2.4 virtual void uConstruct.Core.AOI.BaseAOITarget.OnEnable () [protected], [virtual]

Add the target to the targets list

Reimplemented in [uConstruct.Core.AOI.BuildingGroupAOITarget](#).

5.3.3 Member Data Documentation

5.3.3.1 Vector3 uConstruct.Core.AOI.BaseAOITarget.aoiPosition

Position that is updated by the [AOIManager](#) and used by a different thread

5.3.3.2 bool uConstruct.Core.AOI.BaseAOITarget.inRange = true

Are we in range of any of the finders

5.3.4 Property Documentation

5.3.4.1 virtual bool uConstruct.Core.AOI.BaseAOITarget.useMultiThreadZoneSearch [get]

Will the system use multi-threading to choose whether this target is in range of a finder or not.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/AOI/BaseAOITarget.cs

5.4 uConstruct.BaseBuilding Class Reference

A base building script that has all building methods. Incase of making an another type of use for the building like a mine/ a networked building please inherit from this class.

Inheritance diagram for uConstruct.BaseBuilding:

Public Member Functions

- void [ActivateSockets](#) (bool value, bool forced)
Change socket's state, disabled or enabled (used on init, and also can be used for AOI [disable all sockets when far away from the building]).
- void [ActivateConditions](#) (bool value, bool force)
Change condition's state, disabled or enabled (used on init, and also can be used for AOI [disable all conditions when far away from the building to stay away physics limit]).
- void [ActivateColliders](#) (bool value)
Activate the colliders in the building.
- void [ActivateSnapPoints](#) (bool value)
Activate the snap points in the building.
- virtual bool [HandlePlacing](#) ([UCPhysicsHitsArray](#) hits)
Called from the player controller, called to check the placing of this building and see if it fits the target or not, receives a raycastHit.
- virtual void [HandleSnapPlace](#) ([UCPhysicsHit](#) hit, [BaseSocket](#) socket)
Handle the snap placement of the building.
- virtual void [HandleFreePlace](#) ([UCPhysicsHit](#) hit, [BaseSocket](#) socket)
Handle the free placement of the building.
- virtual bool [HandlePlacing](#) (params [RaycastHit](#)[] hits)
Another way to run Handle placing method but with raycast hits instead of the custom physics library.
- virtual void [PlaceBuilding](#) ()
A method that handles the placing of the building, this is a virtual method so you can implement your 3d party libraries implementation here.
- virtual void [DeAttachBuilding](#) ()
Handles the building deattaching, removes it from the group and updates the sockets that it is no longer snapped.
- virtual void [AssignOriginalColors](#) ()
Assign the original colors of the materials
- virtual void [ResetMaterialColors](#) ()
Reset the materials color to initial colors, used to reset preview material after placement.
- virtual void [HandleMaterial](#) ([BuildingMaterialData](#) mat)
Changing all the materials into a chosen mat.
- virtual bool [CheckConditions](#) ()
Check for all the conditions in the object and make sure they all return true.
- virtual [BaseSocket](#) [ReturnSocket](#) ([Vector3](#) pos, [BuildingType](#)?targetType)
Returns a socket from the building by position.
- virtual void [DestroyBuilding](#) ()
Destroy this building
- virtual void [BuildingGroupChanged](#) ([BaseBuildingGroup](#) group)
Called when the building group of the building changes
- virtual void [GroupBuildingAdded](#) ([BaseBuilding](#) building)
Called when ever a building was created in the group/ added to the group.
- virtual void [GroupBuildingRemoved](#) ([BaseBuilding](#) building)
Called when ever a building was created in the group/ added to the group.
- virtual void [EnableRenderings](#) (bool value)
Enable all building's renderings, used mostly to restore to its old state before being batched.

Parameters

| | |
|-------|---------|
| value | Enable? |
|-------|---------|

- virtual void [BuildingDeattached](#) ()

Called when the building is deattached.

- void [AddTemplate](#) (GameObject template)
Add a template
- void [RemoveTemplate](#) ([Template](#) template)
Remove a template
- [BaseSocket CreateSocket](#) (string name, SocketPositionAnchor socketAnchor, GameObject previewGame↵Object, BuildingType receive, [PlacingRestrictionType](#) restriction)
Create a socket, can be used on both runtime and editor.
- [BaseCondition CreateCondition](#) (string name, SocketPositionAnchor anchor, System.Type condition)
Create a condition, can be used on both runtime and editor.
- [BaseSnapPoint CreateSnapPoint](#) (string name, SocketPositionAnchor anchor, BuildingType type)
Create a condition, can be used on both runtime and editor.
- void [AddSocket](#) ([BaseSocket](#) socket)
Add a socket to the building.
- void [AddCondition](#) ([BaseCondition](#) condition)
Add a condition to the building.
- Vector3 [ReturnPosition](#) (SocketPositionAnchor anchor, Transform transform)
Return a position based upon an anchor
- Transform [ReturnParent](#) (string parentName, bool createlfNotFound)
Return the parent of the specific condition/socket.
- virtual [BlueprintData Pack](#) ()
Pack our building data

Static Public Member Functions

- static void **CallPack** ([BaseBuilding](#) building)
- static void **CallLoad** ([BaseBuilding](#) building)

Public Attributes

- BuildingType **buildingType**
- [PlacingRestrictionType](#) **placingRestrictionType**
- bool **rotateWithSlope** = false
- bool **rotateToFit** = false
- Axis **rotateAxis** = Axis.X
- float **rotateThreshold** = 90f
- int **rotationSteps** = 4
- int **_health** = 100
- [BaseSnapPoint](#) **currentSnapPoint**
- [BaseSocket](#) **SnappedTo**
- int **prefabID** = -1
- List< [Template](#) > **templates** = new List<[Template](#)>()
- const string **SocketParentName** = "Sockets"
- const string **ConditionsParentName** = "Conditions"
- const string **SnapPointParentName** = "Snap Points"

Protected Member Functions

- virtual void [Awake](#) ()
Called on awake, initiates all methods. Please note this is also called on editor.
- virtual void [InitiateBuildingData](#) ()
Calls when creating or placing this building, used to disappear sockets on need or reappear them on need.
- virtual void [SnappedToSocket](#) ([BaseSocket](#) socket)
An event callback thats called when the building is snapped to a socket
- virtual void [LostSnapToSocket](#) ([BaseSocket](#) socket)
An event callback thats called when the building lost snap to a socket.
- virtual void [BuildingPlaced](#) ()
An event callback thats called when building is placed
- virtual void [BuildingDestroyed](#) ()
An event callback thats called when the building is destroyed

Protected Attributes

- [BaseBuildingGroup](#) **_buildingGroup**

Properties

- bool **isSocketPlaceType** [get]
- bool **isFreePlaceType** [get]
- bool **isPlaced** [get]
- bool **batchBuilding** [get, set]
- bool **isBeingPlaced** [get, set]
- int **maxHealth** [get, set]
- virtual int **health** [get, set]
- [BaseSocket](#)[] **sockets** [get, set]
- [BaseSnapPoint](#)[] **snapPoints** [get, set]
- [BaseBuilding](#) **placedOn** [get, set]
- [BaseCondition](#)[] **conditions** [get, set]
- [BaseBuildingGroup](#) **buildingGroup** [get, set]
- bool **hasGroup** [get]
- int **uid** [get, set]
- virtual int [priority](#) [get]
Our priority.
- bool [ignore](#) [get]
Will our building be ignored?

Events

- OnSnappedToSocket **OnSnappedToSocketEvent**
- OnLostSnapToSocket **OnLostSnapToSocketEvent**
- OnPlacedOnChanged **OnPlacedOnChangedEvent**
- OnPlaced **OnPlacedEvent**
- OnDestroy **OnDestroyEvent**
- OnDeattach **OnDeattachEvent**
- OnBuildingGroupChanged **OnGroupChangedEvent**
- OnMaterialColorChanged **OnMaterialColorChangedEvent**
- OnHealthChanged **OnHealthChangedEvent**
- static OnNetworkInstancePacked **OnNetworkInstancePackedEvent**
- static OnNetworkInstanceLoaded **OnNetworkInstanceLoadedEvent**

Private Attributes

- bool **_batchBuilding**
- MeshRenderer[] **meshRenderers**
- Collider[] **colliders**
- List< [BuildingMaterialData](#) > **originalMats** = new List<[BuildingMaterialData](#)>()
- bool **_isBeingPlaced** = true
- int **_maxHealth** = 100
- [BaseSocket](#)[] **_sockets** = new [BaseSocket](#)[0]
- [BaseSnapPoint](#)[] **_snapPoints** = new [BaseSnapPoint](#)[0]
- [BaseBuilding](#) **_placedOn**
- [BaseCondition](#)[] **_conditions**
- int **_uid** = -1

A uid which is used to locate this building. persists through saving BUT, DOESNT persist through network!. (networkID -> networking persisted id).

Static Private Attributes

- static int [globalCount](#)

The global count of the assignable uid.

5.4.1 Detailed Description

A base building script that has all building methods. Incase of making an another type of use for the building like a mine/ a networked building please inherit from this class.

A base building script that has all building methods. This class will handle the creation of sockets, conditions and snap points runtime/editor wise.

5.4.2 Member Function Documentation

5.4.2.1 void uConstruct.BaseBuilding.ActivateColliders (bool *value*)

Activate the colliders in the building.

Parameters

| | |
|--------------|--|
| <i>value</i> | Enable the colliders or disable them ? |
|--------------|--|

5.4.2.2 void uConstruct.BaseBuilding.ActivateConditions (bool *value*, bool *force*)

Change condition's state, disabled or enabled (used on init, and also can be used for AOI [disable all conditions when far away from the building to stay away physics limit]).

Parameters

| | |
|--------------|--|
| <i>value</i> | |
|--------------|--|

5.4.2.3 void uConstruct.BaseBuilding.ActivateSnapPoints (bool *value*)

Activate the snap points in the building.

Parameters

| | |
|--------------|---|
| <i>value</i> | Enable the snappoints or disable them ? |
|--------------|---|

5.4.2.4 void uConstruct.BaseBuilding.ActivateSockets (bool *value*, bool *forced*)

Change socket's state, disabled or enabled (used on init, and also can be used for AOI [disable all sockets when far away from the building]).

Parameters

| | |
|--------------|--|
| <i>value</i> | |
|--------------|--|

5.4.2.5 void uConstruct.BaseBuilding.AddCondition (**BaseCondition** *condition*)

Add a condition to the building.

Parameters

| | |
|------------------|--|
| <i>condition</i> | |
|------------------|--|

5.4.2.6 void uConstruct.BaseBuilding.AddSocket (**BaseSocket** *socket*)

Add a socket to the building.

Parameters

| | |
|---------------|--|
| <i>socket</i> | |
|---------------|--|

5.4.2.7 void uConstruct.BaseBuilding.AddTemplate (**GameObject** *template*)

Add a template

Parameters

| | |
|-----------------|----------------------|
| <i>template</i> | What template to add |
|-----------------|----------------------|

5.4.2.8 virtual void uConstruct.BaseBuilding.AssignOriginalColors () [virtual]

Assign the original colors of the materials

5.4.2.9 virtual void uConstruct.BaseBuilding.Awake () [protected],[virtual]

Called on awake, initiates all methods. Please note this is also called on editor.

5.4.2.10 virtual void uConstruct.BaseBuilding.BuildingDeattached () [virtual]

Called when the building is deattached.

5.4.2.11 virtual void uConstruct.BaseBuilding.BuildingDestroyed () [protected],[virtual]

An event callback thats called when the building is destroyed

5.4.2.12 virtual void uConstruct.BaseBuilding.BuildingGroupChanged (**BaseBuildingGroup** *group*) [virtual]

Called when the building group of the building changes

Parameters

| | |
|--------------|------------------------|
| <i>group</i> | the new building group |
|--------------|------------------------|

5.4.2.13 virtual void uConstruct.BaseBuilding.BuildingPlaced () [protected],[virtual]

An event callback thats called when building is placed

5.4.2.14 virtual bool uConstruct.BaseBuilding.CheckConditions () [virtual]

Check for all the conditions in the object and make sure they all return true.

Returns

Are all conditions meet ?

5.4.2.15 **BaseCondition** uConstruct.BaseBuilding.CreateCondition (string *name*, SocketPositionAnchor *anchor*, System.Type *condition*)

Create a condition, can be used on both runtime and editor.

Parameters

| | |
|------------------|--|
| <i>name</i> | The name of the condition, can be changed later on. |
| <i>anchor</i> | What anchor will this condition be created on, used for mostly editor. |
| <i>condition</i> | What condition this condition will have? cant use an abstract type. |

Returns

5.4.2.16 **BaseSnapPoint** uConstruct.BaseBuilding.CreateSnapPoint (*string name*, *SocketPositionAnchor anchor*, *BuildingType type*)

Create a condition, can be used on both runtime and editor.

Parameters

| | |
|---------------|--|
| <i>name</i> | The name of the condition, can be changed later on. |
| <i>anchor</i> | What anchor will this condition be created on, used for mostly editor. |
| <i>type</i> | What buildings will this snap point receive? |

Returns

5.4.2.17 **BaseSocket** uConstruct.BaseBuilding.CreateSocket (*string name*, *SocketPositionAnchor socketAnchor*, *GameObject previewGameObject*, *BuildingType receive*, *PlacingRestrictionType restriction*)

Create a socket, can be used on both runtime and editor.

Parameters

| | |
|--------------------------|--|
| <i>name</i> | The name of the socket, can be changed later on. |
| <i>socketAnchor</i> | What anchor will this socket be created on, used for mostly editor. |
| <i>previewGameObject</i> | Use a preview game object for the socket creation, used mostly for editor. |
| <i>receive</i> | what kind of buildings will this socket receive?, can be changed later on. |
| <i>restriction</i> | what placing type will this socket have?, can be changed later on. |

Returns

Returns the created socket.

5.4.2.18 **virtual void** uConstruct.BaseBuilding.DeAttachBuilding () [virtual]

Handles the building deattaching, removes it from the group and updates the sockets that it is no longer snapped.

5.4.2.19 virtual void uConstruct.BaseBuilding.DestroyBuilding () [virtual]

Destroy this building

Parameters

| | |
|------------|-----------------------------|
| <i>hit</i> | The hit data of the raycast |
|------------|-----------------------------|

Implements [uConstruct.IBuilding](#).

5.4.2.20 virtual void uConstruct.BaseBuilding.EnableRenderings (bool *value*) [virtual]

Enable all building's renderings, used mostly to restore to its old state before being batched.

Parameters

| | |
|--------------|---------|
| <i>value</i> | Enable? |
|--------------|---------|

5.4.2.21 virtual void uConstruct.BaseBuilding.GroupBuildingAdded (BaseBuilding *building*) [virtual]

Called when ever a building was created in the group/ added to the group.

Parameters

| | |
|-----------------|--------------|
| <i>building</i> | the building |
|-----------------|--------------|

5.4.2.22 virtual void uConstruct.BaseBuilding.GroupBuildingRemoved (BaseBuilding *building*) [virtual]

Called when ever a building was created in the group/ added to the group.

Parameters

| | |
|-----------------|--------------|
| <i>building</i> | the building |
|-----------------|--------------|

5.4.2.23 virtual void uConstruct.BaseBuilding.HandleFreePlace (UCPHysicsHit *hit*, BaseSocket *socket*) [virtual]

Handle the free placement of the building.

Parameters

| | |
|---------------|------------|
| <i>hit</i> | hit data |
| <i>socket</i> | Our socket |

5.4.2.24 `virtual void uConstruct.BaseBuilding.HandleMaterial (BuildingMaterialData mat)` `[virtual]`

Changing all the materials into a chosen mat.

Parameters

| | |
|--------------|-------------------------------------|
| <i>color</i> | the material you want to change to. |
|--------------|-------------------------------------|

5.4.2.25 `virtual bool uConstruct.BaseBuilding.HandlePlacing (UCPhysicsHitsArray hits)` `[virtual]`

Called from the player controller, called to check the placing of this building and see if it fits the target or not, receives a raycastHit.

Parameters

| | |
|------------|---|
| <i>hit</i> | A raycast hit generated from a raycast that is being called on the player/ from where this script is called from. |
|------------|---|

Returns

is this building placeable ?, can be used to change the material to a certain color/ what ever.

5.4.2.26 `virtual bool uConstruct.BaseBuilding.HandlePlacing (params RaycastHit[] hits)` `[virtual]`

Another way to run Handle placing method but with raycast hits instead of the custom physics library.

Parameters

| | |
|-------------|-------------|
| <i>hits</i> | hits array. |
|-------------|-------------|

Returns

did we place it correctly ?

5.4.2.27 `virtual void uConstruct.BaseBuilding.HandleSnapPlace (UCPhysicsHit hit, BaseSocket socket)`
`[virtual]`

Handle the snap placement of the building.

Parameters

| | |
|---------------|------------|
| <i>hit</i> | hit data |
| <i>socket</i> | Our socket |

5.4.2.28 `virtual void uConstruct.BaseBuilding.InitiateBuildingData () [protected], [virtual]`

Calls when creating or placing this building, used to disappear sockets on need or reappear them on need.

5.4.2.29 `virtual void uConstruct.BaseBuilding.LostSnapToSocket (BaseSocket socket) [protected], [virtual]`

An event callback thats called when the building lost snap to a socket.

Parameters

| | |
|---------------|----------------------------|
| <i>socket</i> | the socket we lost snap to |
|---------------|----------------------------|

5.4.2.30 `virtual BlueprintData uConstruct.BaseBuilding.Pack () [virtual]`

Pack our building data

Returns

our building data

Implements [uConstruct.Core.Blueprints.IBlueprintItem](#).

5.4.2.31 `virtual void uConstruct.BaseBuilding.PlaceBuilding () [virtual]`

A method that handles the placing of the building, this is a virtual method so you can implement your 3d party libraries implementation here.

5.4.2.32 `void uConstruct.BaseBuilding.RemoveTemplate (Template template)`

Remove a template

Parameters

| | |
|-----------------|-------------------------|
| <i>template</i> | what template to remove |
|-----------------|-------------------------|

5.4.2.33 `virtual void uConstruct.BaseBuilding.ResetMaterialColors () [virtual]`

Reset the materials color to initial colors, used to reset preview material after placement.

5.4.2.34 `Transform uConstruct.BaseBuilding.ReturnParent (string parentName, bool createIfNotFound)`

Return the parent of the specific condition/socket.

Parameters

| | |
|-------------------------|--|
| <i>isSocket</i> | Is this a socket?. |
| <i>createIfNotFound</i> | Create an instance if parent cannot be found |

Returns

The parent of the socket/ condition.

5.4.2.35 Vector3 uConstruct.BaseBuilding.ReturnPosition (SocketPositionAnchor *anchor*, Transform *transform*)

Return a position based upon an anchor

Parameters

| | |
|---------------|--------------------|
| <i>anchor</i> | what anchor to use |
|---------------|--------------------|

Returns

A position based upon the anchor.

5.4.2.36 virtual BaseSocket uConstruct.BaseBuilding.ReturnSocket (Vector3 *pos*, BuildingType? *targetType*) [virtual]

Returns a socket from the building by position.

Parameters

| | |
|-------------------|---|
| <i>pos</i> | The position of the socket |
| <i>targetType</i> | What is the targeted building type? this will make sure you only check the right sockets and save perf. |

Returns

The socket that is placed on this position

5.4.2.37 virtual void uConstruct.BaseBuilding.SnappedToSocket (BaseSocket *socket*) [protected],[virtual]

An event callback thats called when the building is snapped to a socket

Parameters

| | |
|---------------|--------------------------|
| <i>socket</i> | the socket we snapped to |
|---------------|--------------------------|

5.4.3 Member Data Documentation

5.4.3.1 `int uConstruct.BaseBuilding._uid = -1` `[private]`

A uid which is used to locate this building. persists through saving BUT, DOESNT persist through network!. (networkID -> networking persisted id).

5.4.3.2 `int uConstruct.BaseBuilding.globalCount` `[static], [private]`

The global count of the assignable uid.

5.4.4 Property Documentation

5.4.4.1 `bool uConstruct.BaseBuilding.ignore` `[get]`

Will our building be ignored?

5.4.4.2 `virtual int uConstruct.BaseBuilding.priority` `[get]`

Our priority.

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/BaseBuilding.cs`

5.5 uConstruct.BaseBuildingBatcher Class Reference

A class that is attached to the bathed collider. used to contain data about the group we are batching.

Inheritance diagram for `uConstruct.BaseBuildingBatcher`:

Public Member Functions

- `void DestroyBuilding ()`
Destroy the building
- `BaseBuilding GetBatchedBuilding (Vector3 point)`
Get our batch building from the group

Public Attributes

- `BaseBuildingGroup group`

5.5.1 Detailed Description

A class that is attached to the bathed collider. used to contain data about the group we are batching.

5.5.2 Member Function Documentation

5.5.2.1 void `uConstruct.BaseBuildingBatcher.DestroyBuilding` ()

Destroy the building

Implements [uConstruct.IBuilding](#).

5.5.2.2 `BaseBuilding` `uConstruct.BaseBuildingBatcher.GetBatchedBuilding` (`Vector3 point`)

Get our batch building from the group

Parameters

| | |
|--------------|-------------------------|
| <i>point</i> | a point on the building |
|--------------|-------------------------|

Returns

our batched building instance

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/Batch/BaseBuildingBatcher.cs`

5.6 `uConstruct.BaseBuildingGroup` Class Reference

A base class for groups. Handles all group management. incase of doing another use of groups please inherit from this class.

Inheritance diagram for `uConstruct.BaseBuildingGroup`:

Public Member Functions

- virtual void [DestroyGroup](#) ()
Destroys the group.
- virtual void [AddBuilding](#) ([BaseBuilding](#) building)
Add a building to the group.
- virtual void [RemoveBuilding](#) ([BaseBuilding](#) building)
Remove a building from the group.

- virtual void [AOIGroup](#) (bool value)
- void [HandleOccupiedSockets](#) ([BaseBuilding](#) building)
This method will handle all the building sockets and check if one of the buildings in the group are on that socket.
- bool [IsGroupSocketOccoupied](#) ([BaseSocket](#) socketInstance)
Is the socket occoupied in the group buildings?
- void [EnableGroupSockets](#) (bool value, bool force)
Enable all of the sockets in the group.
- void [PopulateBatchedFilters](#) ([BaseBuilding](#) building, bool Add)
Populate the batch filters.
- virtual void [Batch](#) (bool value, [BaseBuilding](#) updatedBuilding, bool added)
Batch buildings
- virtual [BaseBuilding](#) [ReturnBatchedBuilding](#) (Vector3 pos)
Return a building from a point inside the group.
- [BaseUCSaveData](#) [Save](#) ()
Save all group data into a binary file.

Static Public Member Functions

- static T [CreateGroup](#)< T > (Vector3 pos)
A generic method to create a building group.

Public Attributes

- List< [BaseBuilding](#) > **groupBuildings** = new List<[BaseBuilding](#)>()

Protected Member Functions

- virtual void [GetBuildings](#) ()
Assign all the buildings in your group (get components in childrens) and assign them.
- MeshFilter[] [GetUpdatedBatchData](#) ([BaseBuilding](#) building, bool add)
Get the batch data

Properties

- static List< [BaseBuildingGroup](#) > **groups** [get]
- static int **lastID** [get, set]
- List< [BaseBuilding](#) > **buildings** [get]
Get the group buildings

Events

- BuildingAdded **OnBuildingAddedEvent**
- BuildingRemoved **OnBuildingRemovedEvent**
- GroupBuildingDestroyed **OnGroupBuildingDestroyed**
- BatchedGroup **OnGroupBatchedEvent**
- static OnBatchDone **OnBatchDoneEvent**
- static OnBuildingGroupChanged **OnBuildingGroupCreatedEvent**

Private Member Functions

- void [Awake](#) ()
Awake.

Private Attributes

- bool **initializedBatch**
- [BatchData](#) **batchData**
- HashSet< MeshFilter > **batchedFilters** = new HashSet<MeshFilter>()
- List< Transform > **batchInstances** = new List<Transform>()

Static Private Attributes

- static List< [BaseBuildingGroup](#) > **_groups** = new List<[BaseBuildingGroup](#)>()
- static int **_lastID**

5.6.1 Detailed Description

A base class for groups. Handles all group management. incase of doing another use of groups please inherite from this class.

5.6.2 Member Function Documentation

5.6.2.1 virtual void uConstruct.BaseBuildingGroup.AddBuilding ([BaseBuilding](#) *building*) [virtual]

Add a building to the group.

Parameters

| | |
|-----------------|--------------|
| <i>building</i> | the building |
|-----------------|--------------|

Add this building to the event count

5.6.2.2 virtual void uConstruct.BaseBuildingGroup.AOIGroup (bool *value*) [virtual]

Toggle AOI state on the group, if value is true, it will disable all sockets in the group to save performance and avoid the physics limit.

Use case : when a player is more than 20 meters from the group enable AOI cause he wont build in this area so no need to keep sockets alive.

Parameters

| | |
|--------------|---|
| <i>value</i> | Should it enable AOI or not ? if set to true then all sockets on all buildings in the group will be disabled if set to false then they will be enabled. |
|--------------|---|

5.6.2.3 void uConstruct.BaseBuildingGroup.Awake () [private]

Awake.

5.6.2.4 virtual void uConstruct.BaseBuildingGroup.Batch (bool *value*, BaseBuilding *updatedBuilding*, bool *added*)
[virtual]

Batch buildings

Parameters

| | |
|--------------|----------------------|
| <i>value</i> | Batch the building ? |
|--------------|----------------------|

5.6.2.5 static T uConstruct.BaseBuildingGroup.CreateGroup< T > (Vector3 *pos*) [static]

A generic method to create a building group.

Template Parameters

| | |
|----------|---|
| <i>T</i> | The class of the group, cant be the base group (cant be abstract) |
|----------|---|

Parameters

| | |
|------------|---|
| <i>pos</i> | The position that the group will be on, so if its the first building in the group just give it the building position. |
|------------|---|

Returns

Returns the created group instance.

Type Constraints

***T* : BaseBuildingGroup**

5.6.2.6 virtual void uConstruct.BaseBuildingGroup.DestroyGroup () [virtual]

Destroys the group.

5.6.2.7 void uConstruct.BaseBuildingGroup.EnableGroupSockets (bool *value*, bool *force*)

Enable all of the sockets in the group.

Parameters

| | |
|--------------|--------------------|
| <i>value</i> | Disable or enable? |
| <i>force</i> | Force? |

5.6.2.8 `virtual void uConstruct.BaseBuildingGroup.GetBuildings ()` `[protected]`, `[virtual]`

Assign all the buildings in your group (get components in childrens) and assign them.

5.6.2.9 `MeshFilter [] uConstruct.BaseBuildingGroup.GetUpdatedBatchData (BaseBuilding building, bool add)`
`[protected]`

Get the batch data

Parameters

| | |
|-----------------|--|
| <i>building</i> | |
|-----------------|--|

Returns

5.6.2.10 `void uConstruct.BaseBuildingGroup.HandleOccupiedSockets (BaseBuilding building)`

This method will handle all the building sockets and check if one of the buildings in the group are on that socket.

Parameters

| | |
|-----------------|---|
| <i>building</i> | the building you want to apply sockets check for. |
|-----------------|---|

5.6.2.11 `bool uConstruct.BaseBuildingGroup.IsGroupSocketOccoupied (BaseSocket socketInstance)`

Is the socket occopied in the group buildings?

Parameters

| | |
|-----------------------|--|
| <i>socketInstance</i> | the instance of the socket you are checking. |
|-----------------------|--|

Returns

5.6.2.12 `void uConstruct.BaseBuildingGroup.PopulateBatchedFilters (BaseBuilding building, bool Add)`

Populate the batch filters.

Parameters

| | |
|-----------------|-------------------------------|
| <i>building</i> | the building |
| <i>Add</i> | are we adding ? or removing ? |

5.6.2.13 `virtual void uConstruct.BaseBuildingGroup.RemoveBuilding (BaseBuilding building)` [virtual]

Remove a building from the group.

Parameters

| | |
|-----------------|--------------|
| <i>building</i> | The building |
|-----------------|--------------|

5.6.2.14 `virtual BaseBuilding uConstruct.BaseBuildingGroup.ReturnBatchedBuilding (Vector3 pos)` [virtual]

Return a building from a point inside the group.

Parameters

| | |
|------------|---------------|
| <i>pos</i> | The hit point |
|------------|---------------|

Returns

The building that contains this point, used for batching.

5.6.2.15 `BaseUCSaveData uConstruct.BaseBuildingGroup.Save ()`

Save all group data into a binary file.

Returns

The saved data

Implements [uConstruct.Core.Saving.UCSavedItem](#).

5.6.3 Property Documentation

5.6.3.1 `List<BaseBuilding> uConstruct.BaseBuildingGroup.buildings` [get]

Get the group buildings

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/BaseBuildingGroup.cs

5.7 uConstruct.Conditions.BaseCondition Class Reference

A base condition that should be inherited from when creating conditions.

Inheritance diagram for uConstruct.Conditions.BaseCondition:

Public Member Functions

- virtual bool [CheckCondition](#) ()
Called when the building is being placed, checks for the condition.
- virtual void [OnDrawGizmos](#) ()
Called when gizmos is drawing, can be used to debug your condition.
- virtual void [Awake](#) ()
Called on awake to make sure rootParent isnt null
- virtual [BlueprintData Pack](#) ()
Pack our building data
- Transform [GetTransform](#) ()

Public Attributes

- [BaseBuilding](#) [rootBuilding](#)
The building of this condition

Properties

- virtual bool [DisableOnPlace](#) [get]
Will this condition be disabled when placing the building
- virtual bool [ignore](#) [get]
Ignore physics on this condition ?
- virtual int [priority](#) [get]
Our priority.

5.7.1 Detailed Description

A base condition that should be inherited from when creating conditions.

A partial class for conditions that handles templates

5.7.2 Member Function Documentation

5.7.2.1 virtual void [uConstruct.Conditions.BaseCondition.Awake](#) () [virtual]

Called on awake to make sure rootParent isnt null

Reimplemented in [uConstruct.Conditions.TerrainModificationCondition](#), [uConstruct.Conditions.CheckForCollisionCondition](#), and [uConstruct.Conditions.CheckForGroundCondition](#).

5.7.2.2 virtual bool [uConstruct.Conditions.BaseCondition.CheckCondition](#) () [virtual]

Called when the building is being placed, checks for the condition.

Returns

Is the condition applied?

Reimplemented in [uConstruct.Conditions.TerrainModificationCondition](#), [uConstruct.Conditions.CheckForCollisionCondition](#), [uConstruct.Conditions.CheckForGroundCondition](#), and [uConstruct.Conditions.CheckForBuildingCondition](#).

5.7.2.3 virtual void uConstruct.Conditions.BaseCondition.OnDrawGizmos () [virtual]

Called when gizmos is drawing, can be used to debug your condition.

Reimplemented in [uConstruct.Conditions.TerrainModificationCondition](#), [uConstruct.Conditions.CheckForGroundCondition](#), and [uConstruct.Conditions.CheckForBuildingCondition](#).

5.7.2.4 virtual BlueprintData uConstruct.Conditions.BaseCondition.Pack () [virtual]

Pack our building data

Returns

our building data

Implements [uConstruct.Core.Blueprints.IBlueprintItem](#).

Reimplemented in [uConstruct.Conditions.TerrainModificationCondition](#), [uConstruct.Conditions.CheckForCollisionCondition](#), [uConstruct.Conditions.CheckForGroundCondition](#), and [uConstruct.Conditions.CheckForBuildingCondition](#).

5.7.3 Member Data Documentation

5.7.3.1 BaseBuilding uConstruct.Conditions.BaseCondition.rootBuilding

The building of this condition

5.7.4 Property Documentation

5.7.4.1 virtual bool uConstruct.Conditions.BaseCondition.DisableOnPlace [get]

Will this condition be disabled when placing the building

5.7.4.2 virtual bool uConstruct.Conditions.BaseCondition.ignore [get]

Ignore physics on this condition ?

5.7.4.3 virtual int uConstruct.Conditions.BaseCondition.priority [get]

Our priority.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BaseCondition.cs

5.8 uConstruct.Sockets.BaseSnapPoint Class Reference

Snap points are points on your building which are used for dynamically choosing an anchor for placing the building based on distances.

Inheritance diagram for uConstruct.Sockets.BaseSnapPoint:

Public Member Functions

- virtual Vector3 [AnchoredPosition](#) (Vector3 renderCenter, Vector3 renderSize, Vector3 origin)
Get the resulted anchored position

Parameters

| | |
|--------|---------------------------------|
| origin | <i>the origin of the anchor</i> |
|--------|---------------------------------|

*Returns**the resulting anchor.*

- virtual float [ReturnDistance](#) (Vector3 pos)

*Return our distance from the target.**Parameters*

| | |
|--------|-------------------|
| target | <i>our target</i> |
|--------|-------------------|

*Returns**distance to our target*

- virtual void [Snap](#) (Transform owner)

*Snap this point and stash it.***Static Public Member Functions**

- static [BaseSnapPoint ReturnClosest](#) ([BaseSnapPoint](#)[] points, Vector3 pointInfluence, BuildingType type)

*Return the closest point to the target from the points.***Public Attributes**

- [BaseBuilding](#) **building**
- BuildingType **receiveType**

Protected Member Functions

- virtual void [Awake](#) ()

*Initialize snap point.***Private Member Functions**

- void [OnDrawGizmos](#) ()

*Draw gizmos***5.8.1 Detailed Description**

Snap points are points on your building which are used for dynamically choosing an anchor for placing the building based on distances.

5.8.2 Member Function Documentation

- 5.8.2.1 virtual Vector3 uConstruct.Sockets.BaseSnapPoint.AnchoredPosition (Vector3 *renderCenter*, Vector3 *renderSize*, Vector3 *origin*) [virtual]

Get the resulted anchored position

Parameters

| | |
|---------------|--------------------------|
| <i>origin</i> | the origin of the anchor |
|---------------|--------------------------|

Returns

the resulting anchor.

5.8.2.2 `virtual void uConstruct.Sockets.BaseSnapPoint.Awake () [protected],[virtual]`

Initialize snap point.

5.8.2.3 `void uConstruct.Sockets.BaseSnapPoint.OnDrawGizmos () [private]`

Draw gizmos

5.8.2.4 `static BaseSnapPoint uConstruct.Sockets.BaseSnapPoint.ReturnClosest (BaseSnapPoint[] points, Vector3 pointInfluence, BuildingType type) [static]`

Return the closest point to the target from the points.

Parameters

| | |
|---------------|------------------|
| <i>points</i> | our snap points. |
|---------------|------------------|

Returns

closest snap point to the target.

5.8.2.5 `virtual float uConstruct.Sockets.BaseSnapPoint.ReturnDistance (Vector3 pos) [virtual]`

Return our distance from the target.

Parameters

| | |
|---------------|------------|
| <i>target</i> | our target |
|---------------|------------|

Returns

distance to our target

5.8.2.6 `virtual void uConstruct.Sockets.BaseSnapPoint.Snap (Transform owner) [virtual]`

Snap this point and stash it.

Parameters

| | |
|--------------|----------------------|
| <i>owner</i> | who do we belong to? |
|--------------|----------------------|

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Sockets/Base/BaseSnapPoint.cs

5.9 uConstruct.Sockets.BaseSocket Class Reference

Base class for sockets. inherit from this class if you want to do any code adjustments.

Inheritance diagram for uConstruct.Sockets.BaseSocket:

Public Member Functions

- virtual bool **IsFit** (**BaseBuilding** building, **PlacingRestrictionType** buildingPlaceType)
Handle the building that is hovering on the socket.
- virtual void **Awake** ()
Calls on awake, sets up values and adds the socket to the global sockets collection.
- virtual void **BuildingSnapped** (bool value)
Calls when building was snapped into this socket, if true then it will disable this socket. if false it will re-enable it.
- virtual void **InitiateComponents** (GameObject previewObject, Vector3 colliderScale)
This will take 2 initial parameters and scale up the socket, and use the parameters to determine its values.
- void **OccupySocket** (bool value)
Occupy a socket.
- virtual void **ForceEnable** (bool enable)
Force enable or disable the socket, this will block the ability to normally enable sockets.
- virtual void **EnableSocket** (bool value)
Enable the socket
- virtual void **Update** ()
Initialize runtime socket preview.
- Transform **GetTransform** ()
Get our transform
- virtual **BlueprintData Pack** ()
Pack our building data
- void **RenderEditor** ()
Render our editor. (EDITOR ONLY).
- void **Create** ()
Create our instance.

Static Public Member Functions

- static void **GloballyEnableSockets** (bool value)
Globally enable all of the sockets in the scene

Public Attributes

- BuildingType **receiveType**
- [PlacingRestrictionType](#) **placingType** = PlacingRestrictionType.FreelyBased
- bool **isHoverTarget**
Will building be able to hover on this socket ? (mainly used for flat sockets, terrains etc).
- bool **drawIndividual**
Draw this specific socket individually.
- [BaseBuilding](#) **building**

Static Public Attributes

- static bool **drawSockets** = false

Properties

- static Mesh **cubeMesh** [get]
- static Material **socketMat** [get]
- [PreviewBuilding](#) **previewInstance** [get, set]
- GameObject **PreviewObject** [get, set]
- bool **isActive** [get]
- bool **isOccupied** [get, set]
- bool **isForced** [get]
- bool **isSocketPlaceType** [get]
- bool **isFreePlaceType** [get]
- virtual int **priority** [get]
Our priority.

Events

- static OnPreviewObjectChanged **OnPreviewObjectChangedEvent**

Private Attributes

- [PreviewBuilding](#) **_previewInstance**
- GameObject **_previewObject**
- bool **_isOccupied** = false
- bool **_isForced** = false

Static Private Attributes

- static Mesh **_cubeMesh**
- static Material **_socketMat**

5.9.1 Detailed Description

Base class for sockets. inherit from this class if you want to do any code adjustments.

Handles socket's template and blueprints section

5.9.2 Member Function Documentation

5.9.2.1 virtual void uConstruct.Sockets.BaseSocket.Awake () [virtual]

Calls on awake, sets up values and adds the socket to the global sockets collection.

5.9.2.2 virtual void uConstruct.Sockets.BaseSocket.BuildingSnapped (bool *value*) [virtual]

Calls when building was snapped into this socket, if true then it will disable this socket. if false it will re-enable it.

Parameters

| | |
|--------------|---------------------------------------|
| <i>value</i> | Is a building snapped to the socket ? |
|--------------|---------------------------------------|

5.9.2.3 void uConstruct.Sockets.BaseSocket.Create ()

Create our instance.

Implements [uConstruct.IPlacingModifier](#).

5.9.2.4 virtual void uConstruct.Sockets.BaseSocket.EnableSocket (bool *value*) [virtual]

Enable the socket

Parameters

| | |
|--------------|-------------------|
| <i>value</i> | disable or enable |
|--------------|-------------------|

5.9.2.5 virtual void uConstruct.Sockets.BaseSocket.ForceEnable (bool *enable*) [virtual]

Force enable or disable the socket, this will block the ability to normally enable sockets.

Parameters

| | |
|---------------|--------------------|
| <i>enable</i> | enable or disable? |
|---------------|--------------------|

5.9.2.6 Transform uConstruct.Sockets.BaseSocket.GetTransform ()

Get our transform

Returns

our transform

Implements [uConstruct.Core.Templates.ITemplateObject](#).

5.9.2.7 static void `uConstruct.Sockets.BaseSocket.GloballyEnableSockets (bool value)` `[static]`

Globally enable all of the sockets in the scene

Parameters

| | |
|--------------|--------------------------------------|
| <i>value</i> | Enable the sockets, or disable them. |
|--------------|--------------------------------------|

5.9.2.8 virtual void `uConstruct.Sockets.BaseSocket.InitiateComponents (GameObject previewObject, Vector3 colliderScale)` `[virtual]`

This will take 2 initial parameters and scale up the socket, and use the parameters to determine its values.

Parameters

| | |
|----------------------|--|
| <i>previewObject</i> | The preview object, will be used in order to scale the socket. can be left null. |
| <i>colliderScale</i> | The collider scale. |

5.9.2.9 virtual bool `uConstruct.Sockets.BaseSocket.IsFit (BaseBuilding building, PlacingRestrictionType buildingPlaceType)` `[virtual]`

Handle the building that is hovering on the socket.

Parameters

| | |
|--------------------------|---|
| <i>building</i> | The building that is hovering on the socket now |
| <i>buildingPlaceType</i> | What is the building type ? |

Returns

5.9.2.10 void `uConstruct.Sockets.BaseSocket.OccupySocket (bool value)`

Occupy a socket.

Parameters

| | |
|--------------|----------------------|
| <i>value</i> | occupy or unoccupy ? |
|--------------|----------------------|

5.9.2.11 virtual `BlueprintData` `uConstruct.Sockets.BaseSocket.Pack ()` `[virtual]`

Pack our building data

Returns

our building data

Implements [uConstruct.Core.Blueprints.IBlueprintItem](#).

5.9.2.12 void uConstruct.Sockets.BaseSocket.RenderEditor ()

Render our editor. (EDITOR ONLY).

Implements [uConstruct.IPlacingModifier](#).

5.9.2.13 virtual void uConstruct.Sockets.BaseSocket.Update () [virtual]

Initialize runtime socket preview.

5.9.3 Member Data Documentation**5.9.3.1 bool uConstruct.Sockets.BaseSocket.drawIndividual**

Draw this specific socket individually.

5.9.3.2 bool uConstruct.Sockets.BaseSocket.isHoverTarget

Will building be able to hover on this socket ? (mainly used for flat sockets, terrains etc).

5.9.4 Property Documentation**5.9.4.1 virtual int uConstruct.Sockets.BaseSocket.priority [get]**

Our priority.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Socket/Base/BaseSocket.cs

5.10 uConstruct.Core.Saving.BaseUCSaveData Class Reference

A base class for saving data, inherite from this class when ever you want to create a custom save data

Inheritance diagram for uConstruct.Core.Saving.BaseUCSaveData:

Public Member Functions

- virtual void [Load](#) ([BaseUCSaveData](#) data)
Initiate loading of the data

Public Attributes

- string **GUID**

Properties

- virtual int **priority** [get]

5.10.1 Detailed Description

A base class for saving data, inherite from this class when ever you want to create a custom save data

5.10.2 Member Function Documentation

5.10.2.1 virtual void [uConstruct.Core.Saving.BaseUCSaveData.Load](#) ([BaseUCSaveData](#) *data*) [virtual]

Initiate loading of the data

Parameters

| | |
|-------------|----------|
| <i>data</i> | the data |
|-------------|----------|

Reimplemented in [uConstruct.Core.Saving.BuildingGroupSaveData](#).

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Saving/UCSavingManager.cs

5.11 [uConstruct.BatchClass](#) Class Reference

A class that handles all batch data

Public Member Functions

- **BatchClass** (Material[] _materials, List< MeshFilter > _filters)
- **BatchClass** (Material[] _materials)
- void **AddFilter** (MeshFilter filter, CombineInstance instance)
- void **RemoveFilter** (int index)
- bool [Containes](#) (Material[] materials)
Are the materials contained?

Public Attributes

- `Material[] Materials`
- `List< MeshFilter > Filters = new List<MeshFilter>()`
- `List< CombineInstance > combineInstances = new List<CombineInstance>()`
- `int totalVertexAmount = 0`

5.11.1 Detailed Description

A class that handles all batch data

5.11.2 Member Function Documentation

5.11.2.1 `bool uConstruct.BatchClass.Contains (Material[] materials)`

Are the materials contained?

Parameters

| | |
|------------------|---------------|
| <i>materials</i> | the materials |
|------------------|---------------|

Returns

are the materials contained?

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/BaseBuildingGroup.cs`

5.12 uConstruct.BatchData Class Reference

Our batch data

Public Member Functions

- [BatchClass Batchable](#) (`Material[] data`, `int shapeVertexCount`)
Can we get any more batches? and is the data exists?
- [BatchClass Add](#) ([BatchClass](#) value)
Add a building to the data
- `void Remove (BatchClass value)`
Remove a building from the data

Properties

- [BatchClass this\[int index\]](#) `[get]`
- `int Count` `[get]`

Private Attributes

- List< [BatchClass](#) > **renderesData** = new List<[BatchClass](#)>()

5.12.1 Detailed Description

Our batch data

5.12.2 Member Function Documentation

5.12.2.1 [BatchClass](#) uConstruct.BatchData.Add ([BatchClass](#) *value*)

Add a building to the data

Parameters

| | |
|--------------|----------------------|
| <i>value</i> | what building to add |
|--------------|----------------------|

Returns

instance of the building

5.12.2.2 [BatchClass](#) uConstruct.BatchData.Batchable ([Material](#)[] *data*, int *shapeVertexCount*)

Can we get any more batches? and is the data exists?

Parameters

| | |
|-------------------------|--------------------------------|
| <i>data</i> | our materials data |
| <i>shapeVertexCount</i> | our current shape vertex count |

Returns

instance of the batch data

5.12.2.3 void uConstruct.BatchData.Remove ([BatchClass](#) *value*)

Remove a building from the data

Parameters

| | |
|--------------|-------------------------|
| <i>value</i> | the instance to remove. |
|--------------|-------------------------|

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/Batch/BatchUtility.cs

5.13 uConstruct.BatchExtensions Class Reference

Extension class for the mesh class.

Static Public Member Functions

- static bool **IsMeshFull** (this Mesh mesh)

5.13.1 Detailed Description

Extension class for the mesh class.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/Batch/BatchUtility.cs

5.14 uConstruct.BatchUtility Class Reference

This class handles the batching mechanic of [uConstruct](#). Can be used with other system as well if needed.

Static Public Member Functions

- static [BatchData](#) **CompileInitialBatchData** (MeshFilter[] batch, bool value)
Return combined batched meshes (support submeshes).
- static void **UpdateBatchData** (MeshFilter[] filters, bool Add, ref [BatchData](#) batchData)
Update our batch data
- static bool **isVertexOverLimit** (int amount)
Are we over the vertex limit ?

Static Private Member Functions

- static Material[] **HandleRenders** (MeshFilter filter, bool value)
Handle the renders of the filter

5.14.1 Detailed Description

This class handles the batching mechanic of [uConstruct](#). Can be used with other system as well if needed.

5.14.2 Member Function Documentation

5.14.2.1 static [BatchData](#) uConstruct.BatchUtility.CompileInitialBatchData (MeshFilter[] batch, bool value) [static]

Return combined batched meshes (support submeshes).

Parameters

| | |
|--------------|---------------------------|
| <i>batch</i> | What MeshFilters to batch |
|--------------|---------------------------|

Returns

the batched meshes

5.14.2.2 `static Material [] uConstruct.BatchUtility.HandleRenders (MeshFilter filter, bool value) [static], [private]`

Handle the renders of the filter

Parameters

| | |
|---------------|---------------------|
| <i>filter</i> | the filter |
| <i>value</i> | enable or disable ? |

Returns

the renderer material data

5.14.2.3 `static bool uConstruct.BatchUtility.isVertexOverLimit (int amount) [static]`

Are we over the vertex limit ?

Parameters

| | |
|---------------|--------------------------|
| <i>amount</i> | our current vertex count |
|---------------|--------------------------|

Returns

5.14.2.4 `static void uConstruct.BatchUtility.UpdateBatchData (MeshFilter[] filters, bool Add, ref BatchData batchData) [static]`

Update our batch data

Parameters

| | |
|------------------|--|
| <i>filters</i> | what filters you want to add / remove |
| <i>Add</i> | Are we adding an instance ? or removing it ? |
| <i>batchData</i> | returns the edited batch data |

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/Batch/BatchUtility.cs

5.15 uConstruct.Core.Blueprints.Blueprint Class Reference

Blueprints are a set of data that allows you to quickly create a set of data that can be applied on any kind of a building with not efforts.

Inheritance diagram for uConstruct.Core.Blueprints.Blueprint:

Public Member Functions

- void **Save** ()
Save the blueprint, works only on editor.
- void **Delete** ()
Delete the blueprint, works only on editor.
- void **AddField** (BlueprintField field)
Add a field to the blueprint, works on both runtime and editor.
- void **RemoveField** (BlueprintField field)
Remove a field from the blueprint, works on both runtime and editor.

Static Public Member Functions

- static string **GetPath** (string name)
Get path to a certain name
- static **Blueprint CreateBlueprint** ()
Create a new blueprint, works on both runtime and editor.

Public Attributes

- const string **BLUEPRINT_ASSET_FIRST** = "uBlueprint_"
The first blueprint name.
- string **blueprintName** = "New Blueprint"
The blueprint name.
- List< **BlueprintField** > **fields** = new List<BlueprintField>()
The blueprint's fields.

Properties

- static string **BLUEPRINT_ASSET_PATH** [get]
A static path to the Assets blueprints folder.
- string **selfPath** [get]
Our path.

5.15.1 Detailed Description

[Blueprints](#) are a set of data that allows you to quickly create a set of data that can be applied on any kind of a building with not efforts.

5.15.2 Member Function Documentation

5.15.2.1 void uConstruct.Core.Blueprints.Blueprint.AddField (BlueprintField *field*)

Add a field to the blueprint, works on both runtime and editor.

Parameters

| | |
|--------------|--|
| <i>field</i> | |
|--------------|--|

5.15.2.2 static Blueprint uConstruct.Core.Blueprints.Blueprint.CreateBlueprint () [static]

Create a new blueprint, works on both runtime and editor.

Returns

Our newly created blueprint.

5.15.2.3 void uConstruct.Core.Blueprints.Blueprint.Delete ()

Delete the blueprint, works only on editor.

5.15.2.4 static string uConstruct.Core.Blueprints.Blueprint.GetPath (string *name*) [static]

Get path to a certain name

Parameters

| | |
|-------------|----------|
| <i>name</i> | the name |
|-------------|----------|

Returns

the path of that name.

5.15.2.5 void uConstruct.Core.Blueprints.Blueprint.RemoveField (BlueprintField *field*)

Remove a field from the blueprint, works on both runtime and editor.

Parameters

| | |
|--------------|--|
| <i>field</i> | |
|--------------|--|

5.15.2.6 void uConstruct.Core.Blueprints.Blueprint.Save ()

Save the blueprint, works only on editor.

5.15.3 Member Data Documentation

5.15.3.1 const string uConstruct.Core.Blueprints.Blueprint.BLUEPRINT_ASSET_FIRST = "uBlueprint_ "

The first blueprint name.

5.15.3.2 string uConstruct.Core.Blueprints.Blueprint.blueprintName = "New Blueprint"

The blueprint name.

5.15.3.3 List<BlueprintField> uConstruct.Core.Blueprints.Blueprint.fields = new List<BlueprintField>()

The blueprint's fields.

5.15.4 Property Documentation

5.15.4.1 string uConstruct.Core.Blueprints.Blueprint.BLUEPRINT_ASSET_PATH [static], [get]

A static path to the Assets blueprints folder.

5.15.4.2 string uConstruct.Core.Blueprints.Blueprint.selfPath [get], [private]

Our path.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Blueprints/Blueprint.cs

5.16 uConstruct.Core.Blueprints.BlueprintData Class Reference

A serializeable data class that needs to be inherited from on any data that can be serialized into the blueprint system.

Inheritance diagram for uConstruct.Core.Blueprints.BlueprintData:

Public Member Functions

- virtual void **UnPack** (GameObject target)

Public Attributes

- string **name**
- [SerializeableVector3](#) **position**
- [SerializeableQuaternion](#) **rotation**
- [SerializeableVector3](#) **scale**

5.16.1 Detailed Description

A serializeable data class that needs to be inherited from on any data that can be serialized into the blueprint system.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Blueprints/Blueprint.cs

5.17 BlueprintEditEditor Class Reference

Inheritance diagram for BlueprintEditEditor:

Static Public Member Functions

- static void **OpenWindow** ([BlueprintEditor](#) editor)

Private Member Functions

- void **OnGUI** ()
- void **DrawCreate** ()
- void **DrawEdit** ()

Private Attributes

- GUIStyle **invisibleButtonStyle**
- GUIStyle **boxStyle**
- [BlueprintEditor](#) **editor**
- GameObject **sourceGO**
- BuildingType **fieldType**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Blueprints/Editor/BlueprintEditor.cs

5.18 BlueprintEditor Class Reference

Inheritance diagram for BlueprintEditor:

Static Public Member Functions

- static void **OpenWindow** ()

Public Attributes

- const KeyCode **exportBtn** = KeyCode.LeftControl
- [Blueprint](#) **currentBlueprint**
- [BlueprintField](#) **currentField**

Private Member Functions

- void **LoadBlueprints** ()
- void **Update** ()
- void **OnGUI** ()
- void **DrawBlueprintsList** ()
- void **DrawBlueprintsEditor** ()

Private Attributes

- List< [Blueprint](#) > **loadedBlueprints** = new List<[Blueprint](#)>()
- GUIStyle **invisibleButtonStyle**
- GUIStyle **boxStyle**
- Vector2 **blueprintListScrollPos** = new Vector2()
- Vector2 **fieldsListScrollPos** = new Vector2()
- bool **exportBtnClicked**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Blueprints/Editor/BlueprintEditor.cs

5.19 uConstruct.Core.Blueprints.BlueprintField Class Reference

[Blueprint](#) field holds data about the blueprint.

Inheritance diagram for uConstruct.Core.Blueprints.BlueprintField:

Public Member Functions

- [BlueprintField](#) (BuildingType [type](#))
Create a new field instance
- [BlueprintField](#) (BuildingType [type](#), GameObject [source](#))
Create a new blueprint field.
- void [OnBeforeSerialize](#) ()
Customly serialize the inheritable data.
- void [OnAfterDeserialize](#) ()
Customly serialize the inheritable data.
- void [Pack](#) (GameObject [target](#))
Loads data from a certain GO.
- void [UnPack](#) (GameObject [target](#), bool [saveToPrefab](#))
Unpack data into a gameobject.

Static Public Member Functions

- static bool [Contains](#) ([BlueprintField](#) [field](#), List< [BlueprintField](#) > [fields](#))
Check if a field is contained.

Public Attributes

- List< [BlueprintData](#) > [data](#) = new List<[BlueprintData](#)>()
Unserialized data that is being serialized by a custom method as unity's scriptable object doesnt work well with inherited classes.
- GameObject [target](#)
Target for packaging, used in the editors.

Properties

- string [name](#) [get]
The name of the field.

Private Member Functions

- GameObject [HandlePivot](#) (GameObject [go](#))
Handle wrongly placed pivots.

Private Attributes

- byte[] [dataBytes](#)
Serialized bytes
- BuildingType [type](#)
our field type.

5.19.1 Detailed Description

[Blueprint](#) field holds data about the blueprint.

5.19.2 Constructor & Destructor Documentation

5.19.2.1 `uConstruct.Core.Blueprints.BlueprintField.BlueprintField (BuildingType type)`

Create a new field instance

Parameters

| | |
|-------------|----------------|
| <i>type</i> | the field type |
|-------------|----------------|

5.19.2.2 uConstruct.Core.Blueprints.BlueprintField.BlueprintField (BuildingType *type*, GameObject *source*)

Create a new blueprint field.

Parameters

| | |
|---------------|--|
| <i>type</i> | the type of the field |
| <i>source</i> | the source that the building will get data from. |

5.19.3 Member Function Documentation

5.19.3.1 static bool uConstruct.Core.Blueprints.BlueprintField.Contains (BlueprintField *field*, List< BlueprintField > *fields*) [static]

Check if a field is contained.

Parameters

| | |
|---------------|--------------------|
| <i>field</i> | our field |
| <i>fields</i> | the list of fields |

Returns

is this field already contained?

5.19.3.2 GameObject uConstruct.Core.Blueprints.BlueprintField.HandlePivot (GameObject *go*) [private]

Handle wrongly placed pivots.

Parameters

| | |
|-----------|---------------------|
| <i>go</i> | what object to fix? |
|-----------|---------------------|

Returns

the fixed result

5.19.3.3 void uConstruct.Core.Blueprints.BlueprintField.OnAfterDeserialize ()

Customly serialize the inheritable data.

5.19.3.4 void `uConstruct.Core.Blueprints.BlueprintField.OnBeforeSerialize ()`

Customly serialize the inheritable data.

5.19.3.5 void `uConstruct.Core.Blueprints.BlueprintField.Pack (GameObject target)`

Loads data from a certain GO.

Parameters

| | |
|---------------|-----------------|
| <i>target</i> | our targeted GO |
|---------------|-----------------|

5.19.3.6 void `uConstruct.Core.Blueprints.BlueprintField.UnPack (GameObject target, bool saveToPrefab)`

Unpack data into a gameobject.

Parameters

| | |
|---------------------|---|
| <i>target</i> | Our targeted gameobject |
| <i>saveToPrefab</i> | Save the changes into the prefab, if available. |

5.19.4 Member Data Documentation

5.19.4.1 List<BlueprintData> `uConstruct.Core.Blueprints.BlueprintField.data = new List<BlueprintData>()`

Unserialized data that is being serialized by a custom method as unity's scriptable object doesnt work well with inherited classes.

5.19.4.2 byte [] `uConstruct.Core.Blueprints.BlueprintField.dataBytes` [private]

Serialized bytes

5.19.4.3 GameObject `uConstruct.Core.Blueprints.BlueprintField.target`

Target for packaging, used in the editors.

5.19.4.4 BuildingType `uConstruct.Core.Blueprints.BlueprintField.type` [private]

our field type.

5.19.5 Property Documentation

5.19.5.1 string uConstruct.Core.Blueprints.BlueprintField.name [get]

The name of the field.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Blueprints/Blueprint.cs

5.20 uConstruct.BuildingBlueprintData Class Reference

Inheritance diagram for uConstruct.BuildingBlueprintData:

Public Member Functions

- **BuildingBlueprintData** ([BaseBuilding](#) building)
- override void **UnPack** (GameObject target)

Public Attributes

- BuildingType **buildingType**
- [PlacingRestrictionType](#) **placingRestrictionType**
- bool **batchBuilding**
- bool **rotateWithSlope**
- bool **rotateToFit**
- Axis **rotateAxis**
- float **rotateThreshold**
- int **rotationSteps**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/BaseBuilding.cs

5.21 uConstruct.BuildingEditor Class Reference

Inheritance diagram for uConstruct.BuildingEditor:

Public Member Functions

- override void **OnInspectorGUI** ()

Public Attributes

- [BaseBuilding](#) **script**

Private Member Functions

- void **OpenPropertyCreatingWindow** (string Name, ModifierType type)
- void **OnEnable** ()

Private Attributes

- bool **showSockets**
- bool **showTagEditor**
- bool **showConditions**
- GUIStyle **boxStyle**
- [BaseSocket](#) **socketData**
- [BaseSocket](#)[] **sockets**
- [BaseSnapPoint](#) **point**
- [BaseSnapPoint](#)[] **snapPoints**
- [BaseCondition](#) **conditionData**
- [BaseCondition](#)[] **conditions**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Buildings/BuildingEditor.cs

5.22 uConstruct.Core.AOI.BuildingGroupAOITarget Class Reference

The building version of AOITarget

Inheritance diagram for uConstruct.Core.AOI.BuildingGroupAOITarget:

Public Member Functions

- override void [HandleAOI](#) ([BaseAOIFinder](#) finder, bool _inRange)
Handle the [AOI](#) results
- override bool [InZone](#) (Vector3 finderPos, float radius)
Is the finder in our zone?

Public Attributes

- Vector3 **_totalVectors**
- float [maxPointOnGroup](#) = 0
Furthest point on the building group.

Protected Member Functions

- override void [OnEnable](#) ()
Add the target to our list
- override void [OnDisable](#) ()
Remove the target from the list

Properties

- Vector3 [totalVectors](#) [get, set]
Total vectors from the group buildings
- Vector3 [correctPosition](#) [get]
the center positon on the group

Private Member Functions

- void [GroupBuildingAdded](#) ([BaseBuilding](#) building)
Called when a building was added to our group. extend the radius.
- void [GroupBuildingRemoved](#) ([BaseBuilding](#) building)
Called when a building was removed from our group. shorten the radius.
- void [OnDrawGizmos](#) ()
Draw gizmos

Private Attributes

- [BaseBuildingGroup](#) [buildingGroup](#)
our building instance
- float [radiusAdjuster](#) = 2f
Used to define a new distance.

5.22.1 Detailed Description

The building version of AOITarget

5.22.2 Member Function Documentation

5.22.2.1 void uConstruct.Core.AOI.BuildingGroupAOITarget.GroupBuildingAdded ([BaseBuilding](#) *building*)
[private]

Called when a building was added to our group. extend the radius.

Parameters

| | |
|-----------------|--------------------|
| <i>building</i> | the added building |
|-----------------|--------------------|

5.22.2.2 void `uConstruct.Core.AOI.BuildingGroupAOITarget.GroupBuildingRemoved (BaseBuilding building)`
`[private]`

Called when a building was removed from our group. shorten the radius.

Parameters

| | |
|-----------------|----------------------|
| <i>building</i> | the removed building |
|-----------------|----------------------|

5.22.2.3 override void `uConstruct.Core.AOI.BuildingGroupAOITarget.HandleAOI (BaseAOIFinder finder, bool _inRange)`
`[virtual]`

Handle the [AOI](#) results

Parameters

| | |
|-----------------|--|
| <i>finder</i> | the finder that our results got changed of |
| <i>_inRange</i> | are we in range of the finder? |

Reimplemented from [uConstruct.Core.AOI.BaseAOITarget](#).

5.22.2.4 override bool `uConstruct.Core.AOI.BuildingGroupAOITarget.InZone (Vector3 finderPos, float radius)`
`[virtual]`

Is the finder in our zone?

Parameters

| | |
|------------------|-------------------|
| <i>finderPos</i> | the finder pos |
| <i>radius</i> | the finder radius |

Returns

Are we in range?

Reimplemented from [uConstruct.Core.AOI.BaseAOITarget](#).

5.22.2.5 override void `uConstruct.Core.AOI.BuildingGroupAOITarget.OnDisable ()` `[protected],[virtual]`

Remove the target from the list

Reimplemented from [uConstruct.Core.AOI.BaseAOITarget](#).

5.22.2.6 void `uConstruct.Core.AOI.BuildingGroupAOITarget.OnDrawGizmos ()` `[private]`

Draw gizmos

5.22.2.7 `override void uConstruct.Core.AOI.BuildingGroupAOITarget.OnEnable () [protected],[virtual]`

Add the target to our list

Reimplemented from [uConstruct.Core.AOI.BaseAOITarget](#).

5.22.3 Member Data Documentation

5.22.3.1 **BaseBuildingGroup** `uConstruct.Core.AOI.BuildingGroupAOITarget.buildingGroup [private]`

our building instance

5.22.3.2 `float uConstruct.Core.AOI.BuildingGroupAOITarget.maxPointOnGroup = 0`

Furthest point on the building group.

5.22.3.3 `float uConstruct.Core.AOI.BuildingGroupAOITarget.radiusAdjuster = 2f [private]`

Used to define a new distance.

5.22.4 Property Documentation

5.22.4.1 `Vector3 uConstruct.Core.AOI.BuildingGroupAOITarget.correctPosition [get]`

the center positon on the group

5.22.4.2 `Vector3 uConstruct.Core.AOI.BuildingGroupAOITarget.totalVectors [get],[set],[private]`

Total vectors from the group buildings

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/AOI/CustomAOI/BuildingGroupAOITarget.cs`

5.23 uConstruct.Core.Saving.BuildingGroupSaveData Class Reference

This is a class that holds data for all the group save data. used for saving groups.

Inheritance diagram for `uConstruct.Core.Saving.BuildingGroupSaveData`:

Public Member Functions

- override void [Load](#) ([BaseUCSaveData](#) _data)
Initiate loading of the data

Public Attributes

- List< [BuildingSaveData](#) > **buildingsData** = new List<[BuildingSaveData](#)>()

Static Public Attributes

- static System.Action< GameObject > [initialBuildingAction](#)
Run this action on the first loaded building on each group.

Events

- static OnBuildingLoaded **OnBuildingLoadedEvent**

Private Member Functions

- GameObject **LoadSpecificData** ([BuildingSaveData](#) building)

Private Attributes

- [BaseBuildingGroup](#) **instanceGroup**

Additional Inherited Members

5.23.1 Detailed Description

This is a class that holds data for all the group save data. used for saving groups.

5.23.2 Member Function Documentation

5.23.2.1 override void `uConstruct.Core.Saving.BuildingGroupSaveData.Load` (`BaseUCSaveData data`) `[virtual]`

Initiate loading of the data

Parameters

| | |
|-------------------|----------|
| <code>data</code> | the data |
|-------------------|----------|

Reimplemented from [uConstruct.Core.Saving.BaseUCSaveData](#).

5.23.3 Member Data Documentation

5.23.3.1 System.Action<GameObject> uConstruct.Core.Saving.BuildingGroupSaveData.initialBuildingAction [static]

Run this action on the first loaded building on each group.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Saving/BuidlingGroupSaveData.cs

5.24 uConstruct.BuildingMaterialData Struct Reference

Public Member Functions

- **BuildingMaterialData** (Material mat, Color col)

Public Attributes

- Material **material**
- Color **color**

The documentation for this struct was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/BaseBuilding.cs

5.25 uConstruct.Demo.BuildingPlacer Class Reference

A demo script that comes with the asset to place buildings.

Inheritance diagram for uConstruct.Demo.BuildingPlacer:

Public Member Functions

- virtual void [Awake](#) ()
Create and initialize the callbacks manager.
- virtual void [Start](#) ()
Initiaite the demo ui
- virtual void [ApplyControlsToDemoUI](#) ()
This method will apply our controls to the demo ui, if available.
- virtual void [Update](#) ()
Update the building cycle
- virtual void [GetInputs](#) ()
Get switch slot inputs, in order to use 3d parties like inventories etc you will want to inherite this method and make it empty.
- virtual void [CreateBuildingInstance](#) (GameObject building)
Create a new building instance.
- virtual void [ResetBuildingInstance](#) ()
Reset the current building instance = recreate.
- virtual void [DestroyCurrentBuilding](#) ()
Destroy the currently created building instance.
- virtual void [PlaceBuilding](#) ()
Place the building
- virtual void [DestroyBuilding](#) ([BaseBuilding](#) building, RaycastHit hit)
Destroy the current building
- virtual void [HandlePlacingResults](#) ([BaseBuilding](#) building, bool results)
Handle the placing results, so for example switch the building material color to Red/Green.
- virtual KeyCode [ReturnAlphaKey](#) (int key)
Return an KeyCode between 1-9.

Public Attributes

- [BaseBuilding](#) **currentBuilding**
- List< GameObject > **Buildings** = new List<GameObject>()
- AudioClip **placeBuildingSound**
- AudioClip **destroyBuildingSound**
- AudioSource **audioSource**

Properties

- [BaseBuilding](#) **currentlyInspectedBuilding** [get, set]
- int **currentSlot** [get, set]
- Camera **playerCamera** [get, set]
- float **placingDistance** [get, set]
- float **destroyDistance** [get, set]
- [BuildingMaterialData](#) **canBePlacedMat** [get, set]
- [BuildingMaterialData](#) **cantBePlacedMat** [get, set]
- bool **defaultLockCursor** [get, set]
- bool **LockCursor** [get, set]
- bool **rotateWithPlayer** [get, set]
- float **rotationValue** [get, set]
- bool **destroyBuildings** [get, set]
- float **rayOffset** [get, set]
- RayOrigin **rayOrigin** [get, set]
- virtual Ray **ray** [get]

Private Attributes

- [BaseBuilding](#) **_currentlyInspectedBuilding**
- int **_currentSlot** = -1
- Camera **_playerCamera**
- float **_placingDistance** = 20
- float **_destroyDistance** = 50
- [BuildingMaterialData](#) **_canBePlacedMat** = new [BuildingMaterialData](#)(null, Color.green)
- [BuildingMaterialData](#) **_cantBePlacedMat** = new [BuildingMaterialData](#)(null, Color.red)
- bool **_defaultLockCursor** = true

Whether the cursor will be locked on default.
- bool **_LockCursor**
- bool **_rotatedWithPlayer** = false

Rotate the placed buildings according to the player rotation Kind of like fallout 4 building style.
- float **_rotationValue** = 90

The rotation value of which the buildings will be rotated with the scroll wheel.
- bool **_destroyBuildings** = true

Destroy buildings with right mouse click.
- float **_rayOffset** = 0f
- RayOrigin **_rayOrigin** = RayOrigin.MidScreen

5.25.1 Detailed Description

A demo script that comes with the asset to place buildings.

5.25.2 Member Function Documentation

5.25.2.1 virtual void uConstruct.Demo.BuildingPlacer.ApplyControlsToDemoUI () [virtual]

This method will apply our controls to the demo ui, if available.

5.25.2.2 virtual void uConstruct.Demo.BuildingPlacer.Awake () [virtual]

Create and initialize the callbacks manager.

5.25.2.3 virtual void uConstruct.Demo.BuildingPlacer.CreateBuildingInstance (GameObject *building*) [virtual]

Create a new building instance.

Parameters

| | |
|-----------------|--|
| <i>building</i> | |
|-----------------|--|

5.25.2.4 `virtual void uConstruct.Demo.BuildingPlacer.DestroyBuilding (BaseBuilding building, RaycastHit hit)`
`[virtual]`

Destroy the current building

Parameters

| | |
|-----------------|---------------------|
| <i>building</i> | building instance |
| <i>hit</i> | our hit information |

5.25.2.5 `virtual void uConstruct.Demo.BuildingPlacer.DestroyCurrentBuilding ()` `[virtual]`

Destroy the currently created building instance.

5.25.2.6 `virtual void uConstruct.Demo.BuildingPlacer.GetInputs ()` `[virtual]`

Get switch slot inputs, in order to use 3d parties like inventories etc you will want to inherite this method and make it empty.

5.25.2.7 `virtual void uConstruct.Demo.BuildingPlacer.HandlePlacingResults (BaseBuilding building, bool results)`
`[virtual]`

Handle the placing results, so for example switch the building material color to Red/Green.

Parameters

| | |
|-----------------|--------------|
| <i>building</i> | our building |
| <i>results</i> | the results |

5.25.2.8 `virtual void uConstruct.Demo.BuildingPlacer.PlaceBuilding ()` `[virtual]`

Place the building

5.25.2.9 `virtual void uConstruct.Demo.BuildingPlacer.ResetBuildingInstance ()` `[virtual]`

Reset the current building instance = recreate.

5.25.2.10 `virtual KeyCode uConstruct.Demo.BuildingPlacer.ReturnAlphaKey (int key)` `[virtual]`

Return an KeyCode between 1-9.

Parameters

| | |
|------------|------------------------|
| <i>key</i> | our targeted key index |
|------------|------------------------|

Returns

5.25.2.11 `virtual void uConstruct.Demo.BuildingPlacer.Start () [virtual]`

Initiaite the demo ui

5.25.2.12 `virtual void uConstruct.Demo.BuildingPlacer.Update () [virtual]`

Update the building cycle

5.25.3 Member Data Documentation

5.25.3.1 `bool uConstruct.Demo.BuildingPlacer._defaultLockCursor = true [private]`

Whether the cursor will be locked on default.

5.25.3.2 `bool uConstruct.Demo.BuildingPlacer._destroyBuildings = true [private]`

Destroy buildings with right mouse click.

5.25.3.3 `bool uConstruct.Demo.BuildingPlacer._rotatedWithPlayer = false [private]`

Rotate the placed buildings according to the player rotation Kind of like fallout 4 building style.

5.25.3.4 `float uConstruct.Demo.BuildingPlacer._rotationValue = 90 [private]`

The rotation value of which the buildings will be rotated with the scroll wheel.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Demo/BuildingPlacer.cs

5.26 uConstruct.Core.Saving.BuildingSaveData Class Reference

Save data class for the group

Public Member Functions

- **BuildingSaveData** (Vector3 _pos, Quaternion _rot, int _placedOnUID, int _health, int _prefabID, int _↵uniqueID)

Public Attributes

- [SerializeableVector3](#) **pos**
- [SerializeableQuaternion](#) **rot**
- int **health**
- int **prefabID**
- int **uniqueID**
- int **placedOnUID**

Events

- static OnBuildingSaving **OnBuildingSavingEvent**

5.26.1 Detailed Description

Save data class for the group

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Saving/BuidlingGroupSaveData.cs

5.27 uConstruct.CodeGenerator.BuildingTypesCodeGenerator Class Reference

Static Public Member Functions

- static string **CheckForDuplications** (List< string > enumFields)
- static void **CompileAssembly** (List< string > enumFields)
- static void **GenerateCSharpCode** ()
- static List< string > **LoadEnumData** ()

Static Public Attributes

- static string **fileName** = "BuildingTypes.cs"
- static string **filePath** = Application.dataPath + "/UConstruct/Scripts/CodeGenerator/" + fileName
- static CodeCompileUnit **targetUnit**
- static CodeTypeDeclaration **targetClass**
- static CodeNamespace **CodeNamespace** = new CodeNamespace()

Static Private Member Functions

- static List< string > **UpdateSpaces** (List< string > enumFields)
- static int **ReturnBitValue** (CodeTypeMemberCollection members, int value)

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/CodeGenerator/Editor/BuildingTypesCodeGenerator.cs

5.28 uConstruct.CodeGenerator.BuildingTypesCodeGeneratorEditor Class Reference

Inheritance diagram for uConstruct.CodeGenerator.BuildingTypesCodeGeneratorEditor:

Static Public Member Functions

- static void **OpenWindow** ()

Static Public Attributes

- static List< string > **data**
- static [BuildingTypesCodeGeneratorEditor](#) **instance**
- static bool **removeButtonClicked**

Private Member Functions

- void **Update** ()
- void **OnGUI** ()

Static Private Member Functions

- static void **OnUnityCompiledScripts** ()

Private Attributes

- GUIStyle **boxStyle**
- Vector2 **scrollPos**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/CodeGenerator/Editor/BuildingTypesCodeGenerator↔
Editor.cs

5.29 uConstruct.Conditions.CheckForBuilding_BlueprintData Class Reference

Inheritance diagram for uConstruct.Conditions.CheckForBuilding_BlueprintData:

Public Member Functions

- **CheckForBuilding_BlueprintData** ([CheckForBuildingCondition](#) condition)
- override void **UnPack** (GameObject target)

Public Attributes

- BuildingType **buildings**
- float **distance**
- DetectionType **detectionMethod**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/CheckForBuildingCondition.cs

5.30 uConstruct.Conditions.CheckForBuildingCondition Class Reference

A basic condition that comes with the asset, checks if there is an building that you specify in the editor infront of the condition in the distance specified.

Inheritance diagram for uConstruct.Conditions.CheckForBuildingCondition:

Public Member Functions

- override bool [CheckCondition](#) ()
Called when the building is being placed, checks for the condition.
- override void [OnDrawGizmos](#) ()
Called when gizmos is drawing, can be used to debug your condition.
- override [BlueprintData Pack](#) ()
Pack our building data

Public Attributes

- BuildingType **buildings**
- float **distance** = 1
- DetectionType **detectionMethod** = DetectionType.Raycast

Private Member Functions

- bool **CheckSphere** ()
- bool **CheckHit** (Transform hit, Vector3 point)
- bool **CheckRay** ()

Additional Inherited Members

5.30.1 Detailed Description

A basic condition that comes with the asset, checks if there is an building that you specify in the editor infront of the condition in the distance specified.

5.30.2 Member Function Documentation

5.30.2.1 override bool uConstruct.Conditions.CheckForBuildingCondition.CheckCondition () [virtual]

Called when the building is being placed, checks for the condition.

Returns

Is the condition applied?

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.30.2.2 override void uConstruct.Conditions.CheckForBuildingCondition.OnDrawGizmos () [virtual]

Called when gizmos is drawing, can be used to debug your condition.

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.30.2.3 override BlueprintData uConstruct.Conditions.CheckForBuildingCondition.Pack () [virtual]

Pack our building data

Returns

our building data

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/CheckForBuilding↵
Condition.cs

5.31 CheckForBuildingsEditor Class Reference

Inheritance diagram for CheckForBuildingsEditor:

Public Member Functions

- override void **OnInspectorGUI** ()

Private Attributes

- [CheckForBuildingCondition](#) script

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Condition/CheckForBuildingsEditor.cs

5.32 uConstruct.Conditions.CheckForCollision_BlueprintData Class Reference

Inheritance diagram for uConstruct.Conditions.CheckForCollision_BlueprintData:

Public Member Functions

- **CheckForCollision_BlueprintData** ([CheckForCollisionCondition](#) condition)
- override void **UnPack** (GameObject target)

Public Attributes

- List< string > **allowedTags**
- [SerializeableVector3](#) **ceneter**
- [SerializeableVector3](#) **size**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/CheckForCollision↔Condition.cs

5.33 uConstruct.Conditions.CheckForCollisionCondition Class Reference

This class is a built-in condition that comes with the asset. it checks for any collision while placing the object, to make sure you arent placing buildings inside buildings and so on.

Inheritance diagram for uConstruct.Conditions.CheckForCollisionCondition:

Public Member Functions

- override void [Awake](#) ()
Called on awake to make sure rootParent isnt null
- override bool [CheckCondition](#) ()
Called when the building is being placed, checks for the condition.
- override [BlueprintData](#) [Pack](#) ()
Pack our building data

Public Attributes

- List< string > **allowedTags** = new List<string>()
- List< Collider > **collisions** = new List<Collider>()

Properties

- override bool **DisableOnPlace** [get]

Private Member Functions

- void **RemoveNullReferences** ()
- void **AddCollider** (Collider collider)
- void **RemoveCollider** (Collider collider)
- void **OnTriggerEnter** (Collider collision)
- void **OnTriggerExit** (Collider collision)

Private Attributes

- BoxCollider **_collider**
- Rigidbody **_rigid**

5.33.1 Detailed Description

This class is a built-in condition that comes with the asset. it checks for any collision while placing the object, to make sure you arent placing buildings inside buildings and so on.

5.33.2 Member Function Documentation

5.33.2.1 override void uConstruct.Conditions.CheckForCollisionCondition.Awake () [virtual]

Called on awake to make sure rootParent isnt null

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.33.2.2 override bool uConstruct.Conditions.CheckForCollisionCondition.CheckCondition () [virtual]

Called when the building is being placed, checks for the condition.

Returns

Is the condition applied?

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.33.2.3 override `BlueprintData` `uConstruct.Conditions.CheckForCollisionCondition.Pack ()` [virtual]

Pack our building data

Returns

our building data

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/CheckForCollisionCondition.cs`

5.34 `uConstruct.Conditions.CheckForGround_BlueprintData` Class Reference

Inheritance diagram for `uConstruct.Conditions.CheckForGround_BlueprintData`:

Public Member Functions

- **`CheckForGround_BlueprintData`** ([CheckForGroundCondition](#) condition)
- override void **`UnPack`** (GameObject target)

Public Attributes

- float **`destroyDelay`**

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/CheckForGroundCondition.cs`

5.35 `uConstruct.Conditions.CheckForGroundCondition` Class Reference

A basic built-in condition that checks if the building has ground. if it doesnt it will add gravity to the object and remove him from the group (at the end, destroy it).

Inheritance diagram for `uConstruct.Conditions.CheckForGroundCondition`:

Public Member Functions

- override bool [CheckCondition](#) ()
Called when the building is being placed, checks for the condition.
- override void [Awake](#) ()
Called on awake to make sure rootParent isnt null
- override void [OnDrawGizmos](#) ()
Called when gizmos is drawing, can be used to debug your condition.
- override [BlueprintData Pack](#) ()
Pack our building data

Public Attributes

- float **destroyDelay** = 10.0f

Properties

- override bool **DisableOnPlace** [get]

Private Member Functions

- void **InitiateCondition** ([BaseBuilding](#) building)
- void **AddGravity** ()

Private Attributes

- [BaseBuildingGroup](#) **buildingGroup**
- bool **calledAlready**

5.35.1 Detailed Description

A basic built-in condition that checks if the building has ground. if it doesnt it will add gravity to the object and remove him from the group (at the end, destroy it).

5.35.2 Member Function Documentation

5.35.2.1 override void uConstruct.Conditions.CheckForGroundCondition.Awake () [virtual]

Called on awake to make sure rootParent isnt null

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.35.2.2 override bool uConstruct.Conditions.CheckForGroundCondition.CheckCondition () [virtual]

Called when the building is being placed, checks for the condition.

Returns

Is the condition applied?

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.35.2.3 override void uConstruct.Conditions.CheckForGroundCondition.OnDrawGizmos () [virtual]

Called when gizmos is drawing, can be used to debug your condition.

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.35.2.4 override BlueprintData uConstruct.Conditions.CheckForGroundCondition.Pack () [virtual]

Pack our building data

Returns

our building data

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/CheckForGround↔
Condition.cs

5.36 uConstruct.Demo.DemoUI Class Reference

A simple demo class that handles the building placer controls

Inheritance diagram for uConstruct.Demo.DemoUI:

Public Member Functions

- void [AddControl](#) (string name)
Add a new control to the controls text
- void [ResetControl](#) ()
Reset the controls list
- void [Inspect](#) (string text)
Inspect a certain transform.

Static Public Attributes

- static [DemoUI instance](#)
Our instance.

Private Member Functions

- void [Awake](#) ()
Initialize Instance.

Private Attributes

- Text [controls](#)
Our ui text variable.
- Text **inspectedTarget**
- int [controlsCount](#)
Our current controls added count.

5.36.1 Detailed Description

A simple demo class that handles the building placer controls

5.36.2 Member Function Documentation

5.36.2.1 void uConstruct.Demo.DemoUI.AddControl (string *name*)

Add a new control to the controls text

Parameters

| | |
|-------------|-----------------------------|
| <i>name</i> | what that control presents. |
|-------------|-----------------------------|

5.36.2.2 void uConstruct.Demo.DemoUI.Awake () [private]

Initialize Instance.

5.36.2.3 void uConstruct.Demo.DemoUI.Inspect (string *text*)

Inspect a certain transform.

Parameters

| | |
|-------------|-------------------------|
| <i>text</i> | the name of the target. |
|-------------|-------------------------|

5.36.2.4 void `uConstruct.Demo.DemoUI.ResetControl` ()

Reset the controls list

5.36.3 Member Data Documentation

5.36.3.1 Text `uConstruct.Demo.DemoUI.controls` [private]

Our ui text variable.

5.36.3.2 int `uConstruct.Demo.DemoUI.controlsCount` [private]

Our current controls added count.

5.36.3.3 DemoUI `uConstruct.Demo.DemoUI.instance` [static]

Our instance.

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Demo/DemoUI.cs`

5.37 `uConstruct.Extensions.ExtensionsEditor` Class Reference

Inheritance diagram for `uConstruct.Extensions.ExtensionsEditor`:

Static Public Member Functions

- static void **Open** ()
- static void **HandleCompile** ()

Properties

- static bool **isOpen** [get]

Private Member Functions

- void **Init** ()
- void **OnGUI** ()

Private Attributes

- Dictionary< string, List< Extension > > **extensions** = new Dictionary<string, List<Extension>>()
- GUIStyle **invisibleButtonStyle**
- GUIStyle **boxStyle**
- Vector2 **scrollPos**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Extensions/Editor/ExtensionsEditor.cs

5.38 uConstruct.FlagsHelper Class Reference

Some helper classes for bitmasks

Static Public Member Functions

- static bool **IsBitSet**< T > (T values, T value)
Is the value contained inside the enum values?
- static bool **IsBitSet** (BuildingType values, BuildingType value)
Is the value contained inside the enum values?
- static bool **IsInsideMask** (int GameObjectLayer, LayerMask mask)
Is this layer inside the layer bitmask?

5.38.1 Detailed Description

Some helper classes for bitmasks

5.38.2 Member Function Documentation

5.38.2.1 static bool uConstruct.FlagsHelper.IsBitSet (BuildingType *values*, BuildingType *value*) [static]

Is the value contained inside the enum values?

Template Parameters

| | |
|----------|----------------------|
| <i>T</i> | the type of the enum |
|----------|----------------------|

Parameters

| | |
|---------------|--|
| <i>values</i> | values of the enum |
| <i>value</i> | the specific value you want to check if is included in the enum values |

Returns

is it assigned or not ?

5.38.2.2 `static bool uConstruct.FlagsHelper.IsBitSet< T > (T values, T value)` `[static]`

Is the value contained inside the enum values?

Template Parameters

| | |
|----------|----------------------|
| <i>T</i> | the type of the enum |
|----------|----------------------|

Parameters

| | |
|---------------|--|
| <i>values</i> | values of the enum |
| <i>value</i> | the specific value you want to check if is included in the enum values |

Returns

is it assigned or not ?

Type Constraints

T* : *struct

5.38.2.3 `static bool uConstruct.FlagsHelper.isInsideMask (int GameObjectLayer, LayerMask mask)` `[static]`

Is this layer inside the layer bitmask?

Parameters

| | |
|------------------------|-----------------------------|
| <i>GameObjectLayer</i> | the layer you want to check |
| <i>mask</i> | the mask |

Returns

is the layer inside the mask?

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/BaseBuilding.cs

5.39 uConstruct.Conditions.HeightsData Class Reference

Public Member Functions

- **HeightsData** (float _x, float _z, float _value)

Static Public Member Functions

- static float[] **returnArray** (List< [HeightsData](#) > list, int count)

Public Attributes

- float **x**
- float **z**
- float **value**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/TerrainModification↔
Condition.cs

5.40 uConstruct.Core.Blueprints.IBlueprintItem Interface Reference

An interface that each one of the blueprinted items should have.

Inheritance diagram for uConstruct.Core.Blueprints.IBlueprintItem:

Public Member Functions

- [BlueprintData Pack](#) ()
Pack our data

Properties

- int [priority](#) [get]
Our priority, this is used when ordering the "Pack" methods.

5.40.1 Detailed Description

An interface that each one of the blueprinted items should have.

5.40.2 Member Function Documentation

5.40.2.1 [BlueprintData](#) uConstruct.Core.Blueprints.IBlueprintItem.Pack ()

Pack our data

Returns

our data

Implemented in [uConstruct.BaseBuilding](#), [uConstruct.Sockets.BaseSocket](#), [uConstruct.Conditions.Terrain↔
ModificationCondition](#), [uConstruct.Conditions.CheckForCollisionCondition](#), [uConstruct.Conditions.CheckFor↔
GroundCondition](#), [uConstruct.Conditions.CheckForBuildingCondition](#), and [uConstruct.Conditions.BaseCondition](#).

5.40.3 Property Documentation

5.40.3.1 `int uConstruct.Core.Blueprints.IBlueprintItem.priority` [get]

Our priority, this is used when ordering the "Pack" methods.

The documentation for this interface was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Blueprints/Blueprint.cs`

5.41 `uConstruct.IBuilding` Interface Reference

An interface for all buildings.

Inheritance diagram for `uConstruct.IBuilding`:

Public Member Functions

- void `DestroyBuilding` ()
Destroy the building

5.41.1 Detailed Description

An interface for all buildings.

5.41.2 Member Function Documentation

5.41.2.1 `void uConstruct.IBuilding.DestroyBuilding ()`

Destroy the building

Implemented in `uConstruct.BaseBuilding`, and `uConstruct.BaseBuildingBatcher`.

The documentation for this interface was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/BaseBuilding.cs`

5.42 `uConstruct.IPlacingModifier` Interface Reference

Inheritance diagram for `uConstruct.IPlacingModifier`:

Public Member Functions

- void **RenderEditor** ()
- void **Create** ()

The documentation for this interface was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/IPlacingModifier.cs

5.43 uConstruct.Core.Templates.ITemplateObject Interface Reference

Inheritance diagram for uConstruct.Core.Templates.ITemplateObject:

Public Member Functions

- Transform **GetTransform** ()

The documentation for this interface was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/ITemplateObject.cs

5.44 uConstruct.Core.Threading.IThreadTask Interface Reference

A thread task interface. Implement on any customely created thread task.

Inheritance diagram for uConstruct.Core.Threading.IThreadTask:

Public Member Functions

- void **Invoke** ()

5.44.1 Detailed Description

A thread task interface. Implement on any customely created thread task.

The documentation for this interface was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/MultiThreading/ThreadManager.cs

5.45 uConstruct.Core.Physics.IUTCPysicsIgnored Interface Reference

Ignore all physics on this script.

Inheritance diagram for uConstruct.Core.Physics.IUTCPysicsIgnored:

Properties

- bool **ignore** [get]

5.45.1 Detailed Description

Ignore all physics on this script.

The documentation for this interface was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Physics/UCPhysics.cs

5.46 uConstruct.LayersData Class Reference

A class that contains information about custom layers data of the asset. this data is used when initiating layers assigning and so on.

Inheritance diagram for uConstruct.LayersData:

Public Attributes

- List< string > **_socketLayers** = new List<string>() { "BuildingSocket" }
- List< string > **_buildingLayers** = new List<string>() { "Building" }
- int **defaultSocketLayer**
- int **defaultBuildingLayer**
- const string **FILE_PATH** = "Data/" + "[LayersData](#)"

Properties

- static [LayersData](#) **instance** [get]
- static int [SocketMask](#) [get]
A mask that aims for all of the allowed socket layers. (used for raycasts).
- static int [BuildingMask](#) [get]
A mask that aims for all of the allowed building layers. (used for raycasts).
- static List< string > [SocketLayers](#) [get]
All the layers that can be used for [Sockets](#)
- static List< string > [BuildingLayers](#) [get]
All the layers that can be used for buildings
- static int [DefaultBuildingLayer](#) [get]
The default building layer that will be assigned to a building that doesnt have a layer that is contained inside the building layers list
- static int [DefaultSocketLayer](#) [get]
The default socket layer that will be assigned to a building that doesnt have a layer that is contained inside the socket layers list
- static string [DefaultBuildingLayerString](#) [get]
The default building layer that will be assigned to a building that doesnt have a layer that is contained inside the building layers list
- static string [DefaultSocketLayerString](#) [get]
The default socket layer that will be assigned to a building that doesnt have a layer that is contained inside the socket layers list

Static Private Attributes

- static [LayersData](#) **_instance**
- static int **_socketMask** = -999
- static int **_buildingMask** = -999

5.46.1 Detailed Description

A class that contains information about custom layers data of the asset. this data is used when initiating layers assigning and so on.

5.46.2 Property Documentation

5.46.2.1 List<string> uConstruct.LayersData.BuildingLayers [static], [get]

All the layers that can be used for buildings

5.46.2.2 int uConstruct.LayersData.BuildingMask [static], [get]

A mask that aims for all of the allowed building layers. (used for raycasts).

5.46.2.3 `int uConstruct.LayersData.DefaultBuildingLayer` `[static], [get]`

The default building layer that will be assigned to a building that doesnt have a layer that is contained inside the building layers list

5.46.2.4 `string uConstruct.LayersData.DefaultBuildingLayerString` `[static], [get]`

The default building layer that will be assigned to a building that doesnt have a layer that is contained inside the building layers list

5.46.2.5 `int uConstruct.LayersData.DefaultSocketLayer` `[static], [get]`

The default socket layer that will be assigned to a building that doesnt have a layer that is contained inside the socket layers list

5.46.2.6 `string uConstruct.LayersData.DefaultSocketLayerString` `[static], [get]`

The default socket layer that will be assigned to a building that doesnt have a layer that is contained inside the socket layers list

5.46.2.7 `List<string> uConstruct.LayersData.SocketLayers` `[static], [get]`

All the layers that can be used for [Sockets](#)

5.46.2.8 `int uConstruct.LayersData.SocketMask` `[static], [get]`

A mask that aims for all of the allowed socket layers. (used for raycasts).

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Data/LayersData.cs`

5.47 uConstruct.LayersEditor Class Reference

Inheritance diagram for uConstruct.LayersEditor:

Static Public Member Functions

- static void **OpenWindow** ()

Public Attributes

- const string **DATA_PATH** = "Data/" + "[LayersData](#)"

Static Public Attributes

- static [LayersData](#) **layersData**

Private Member Functions

- void **Init** ()
- void **Update** ()
- void **LoadResources** ()
- void **OnGUI** ()

Private Attributes

- GUIStyle **boxStyle**
- bool **removeButtonClicked**
- bool **showBuildingLayers**
- bool **showSocketLayers**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Core/LayersEditor.cs

5.48 uConstruct.PhysicsObjectEditor Class Reference

Inheritance diagram for uConstruct.PhysicsObjectEditor:

Public Member Functions

- virtual void **OnEnable** ()
- override void **OnInspectorGUI** ()

Private Attributes

- GUIStyle **_boxStyle**
- [UCPhysicsObject](#) **_script**
- SerializedProperty **m_usePhysics**
- SerializedProperty **m_center**
- SerializedProperty **m_size**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Core/PhysicsObjectEditor.cs

5.49 uConstruct.Extensions.PCloudExtension.PlayerInstantiator Class Reference

Inheritance diagram for uConstruct.Extensions.PCloudExtension.PlayerInstantiator:

Public Member Functions

- override void **OnJoinedRoom** ()

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Extensions/Extensions/Photon/Integration/Demo/Player↔Instantiator.cs

5.50 uConstruct.Core.PrefabDatabase.PrefabData Class Reference

Holds all the data for a prefab

Public Member Functions

- **PrefabData** (int _ID, GameObject _go)

Public Attributes

- int **ID** = -1
- GameObject **go** = null

5.50.1 Detailed Description

Holds all the data for a prefab

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/PrefabManager/PrefabDB.cs

5.51 uConstruct.Core.PrefabDatabase.PrefabDatabaseEditor Class Reference

Inheritance diagram for uConstruct.Core.PrefabDatabase.PrefabDatabaseEditor:

Static Public Member Functions

- static void **OpenWindow** ()

Static Private Member Functions

- static void **UpdateDB** ()
Update the prefab database

5.51.1 Member Function Documentation

5.51.1.1 static void uConstruct.Core.PrefabDatabase.PrefabDatabaseEditor.UpdateDB () [static],[private]

Update the prefab database

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/PrefabManager/Editor/PrefabDatabaseEditor.cs

5.52 uConstruct.Core.PrefabDatabase.PrefabDB Class Reference

This class handles all prefab database in the system.

Inheritance diagram for uConstruct.Core.PrefabDatabase.PrefabDB:

Public Member Functions

- void **AddToDB** (GameObject go, int UID)
Add an item to the database
- void **RemoveFromDB** (GameObject go)
Remove an prefab from the database
- void **ResetDB** ()
Reset the prefabs on the database
- bool **Contains** (int uid)
Does the prefab contains this prefab id?
- int **AddToDB** (GameObject go)
Add an item to the database
- GameObject **GetGO** (int prefabID)
Get a gameobject thats attached to this prefabID
- GameObject **GetGO** (BuildingType type)
Returns a building that has that specific type in it.
- int **ReturnUID** ()
Return a random id that isnt used

Properties

- static [PrefabDB](#) **instance** [get]
- List< [PrefabData](#) > **prefabs** [get]

Private Member Functions

- int [ReturnUID](#) (int initial)
Get ID that isnt in use

Private Attributes

- List< [PrefabData](#) > **_prefabs** = new List<[PrefabData](#)>()

Static Private Attributes

- static [PrefabDB](#) **_instance**

5.52.1 Detailed Description

This class handles all prefab database in the system.

5.52.2 Member Function Documentation

5.52.2.1 void uConstruct.Core.PrefabDatabase.PrefabDB.AddToDB (GameObject *go*, int *UID*)

Add an item to the database

Parameters

| | |
|------------|---------------------------------------|
| <i>go</i> | The GameObject you want to add |
| <i>UID</i> | The prefabID you want to assign to it |

5.52.2.2 int uConstruct.Core.PrefabDatabase.PrefabDB.AddToDB (GameObject *go*)

Add an item to the database

Parameters

| | |
|-----------|----------------|
| <i>go</i> | The gameObject |
|-----------|----------------|

Returns

Random prefabID

5.52.2.3 bool uConstruct.Core.PrefabDatabase.PrefabDB.Contains (int *uid*)

Does the prefab contains this prefab id?

Parameters

| | |
|------------|-------------------------|
| <i>uid</i> | the prefab uid to check |
|------------|-------------------------|

Returns

is it used?

5.52.2.4 GameObject uConstruct.Core.PrefabDatabase.PrefabDB.GetGO (int *prefabID*)

Get a gameobject thats attached to this prefabID

Parameters

| | |
|-----------------|--|
| <i>prefabID</i> | The prefab id you want to get an game object off |
|-----------------|--|

Returns

The game object this prefabID belongs to

5.52.2.5 GameObject uConstruct.Core.PrefabDatabase.PrefabDB.GetGO (BuildingType *type*)

Returns a building that has that specific type in it.

Parameters

| | |
|-------------|----------------------|
| <i>type</i> | type of the building |
|-------------|----------------------|

Returns

5.52.2.6 void uConstruct.Core.PrefabDatabase.PrefabDB.RemoveFromDB (GameObject *go*)

Remove an prefab from the database

Parameters

| | |
|-----------|-----------------------|
| <i>go</i> | what prefab to remove |
|-----------|-----------------------|

5.52.2.7 void `uConstruct.Core.PrefabDatabase.PrefabDB.ResetDB ()`

Reset the prefabs on the database

5.52.2.8 int `uConstruct.Core.PrefabDatabase.PrefabDB.ReturnUID ()`

Return a random id that isnt used

Returns

an random id

5.52.2.9 int `uConstruct.Core.PrefabDatabase.PrefabDB.ReturnUID (int initial)` [private]

Get ID that isnt in use

Parameters

| | |
|----------------|---|
| <i>initial</i> | the initial value, leave 0 if called first time |
|----------------|---|

Returns

unique id

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/PrefabManager/PrefabDB.cs

5.53 `uConstruct.Core.PrefabDatabase.PrefabDBCustomeEditor` Class Reference

Inheritance diagram for `uConstruct.Core.PrefabDatabase.PrefabDBCustomeEditor`:

Public Member Functions

- override void **OnInspectorGUI** ()

Private Attributes

- [PrefabDB](#) instance

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/PrefabManager/Editor/PrefabDBCustomeEditor.cs

5.54 uConstruct.PreviewBuilding Class Reference

A class that is attached to the socket preview object to contain data about the prefab and apply changes to the prefab.

Inheritance diagram for uConstruct.PreviewBuilding:

Public Member Functions

- void [ApplyChangesToPrefab](#) (GameObject prefab)
Apply changes to the prefab from the transform (rotation and scale).
- void [FitToLocalSpace](#) ()
Fits the transform to fit the parent scale. (changes localScale to 1,1,1)

Public Attributes

- GameObject [previewPrefab](#)
Our preview prefab.

5.54.1 Detailed Description

A class that is attached to the socket preview object to contain data about the prefab and apply changes to the prefab.

5.54.2 Member Function Documentation

5.54.2.1 void uConstruct.PreviewBuilding.ApplyChangesToPrefab (GameObject *prefab*)

Apply changes to the prefab from the transform (rotation and scale).

Parameters

| | |
|---------------|------------|
| <i>prefab</i> | our prefab |
|---------------|------------|

5.54.2.2 void uConstruct.PreviewBuilding.FitToLocalSpace ()

Fits the transform to fit the parent scale. (changes localScale to 1,1,1)

5.54.3 Member Data Documentation

5.54.3.1 GameObject `uConstruct.PreviewBuilding.previewPrefab`

Our preview prefab.

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/PreviewBuilding.cs`

5.55 `uConstruct.PreviewBuildingEditor` Class Reference

Inheritance diagram for `uConstruct.PreviewBuildingEditor`:

Public Member Functions

- override void **OnInspectorGUI** ()

Private Attributes

- [PreviewBuilding](#) script

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Buildings/PreviewBuildingEditor.cs`

5.56 `uConstruct.PropertyCreatorEditor` Class Reference

Inheritance diagram for `uConstruct.PropertyCreatorEditor`:

Public Member Functions

- void **Init** ([BuildingEditor](#) editorWindow, string startingName, GUIStyle style, ModifierType modifier)

Public Attributes

- ModifierType **creatingType** = ModifierType.Socket

Private Member Functions

- void **OnGUI** ()

Private Attributes

- [BuildingEditor](#) **buildingEditor**
- string **propertyName**
- SocketPositionAnchor **positionAnchor** = SocketPositionAnchor.Center
- GameObject **previewGameObject**
- BuildingType **receivesBuildings**
- [PlacingRestrictionType](#) **socketPlacingType**
- MonoScript **condition**
- BuildingType **targetType**
- GUIStyle **boxStyle**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Buildings/BuildingEditor.cs

5.57 uConstruct.Conditions.TerrainModificationCondition.RestoreData Class Reference

Public Attributes

- Dictionary< int, int[,]> **details** = new Dictionary<int, int[,]>()
- float[,] **heights**

Properties

- bool **hasHeights** [get]

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/TerrainModification↔Condition.cs

5.58 uConstruct.Core.Saving.SaveDrawer Class Reference

Inheritance diagram for uConstruct.Core.Saving.SaveDrawer:

Public Member Functions

- void **DrawSave** (List< [BuildingGroupSaveData](#) > stashedSavedData)
- void **Awake** ()

Private Attributes

- List< [BuildingGroupSaveData](#) > **groupSavingData** = new List<[BuildingGroupSaveData](#)>()

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Saving/SaveDrawer.cs

5.59 uConstruct.SavingEditor Class Reference

Inheritance diagram for uConstruct.SavingEditor:

Static Public Member Functions

- static void **ResetSave** ()

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Core/SavingEditor.cs

5.60 uConstruct.Core.Saving.SerializeableQuaternion Class Reference

A serializeable version of quaternion

Public Member Functions

- **SerializeableQuaternion** (float x, float y, float z, float w)

Static Public Member Functions

- static implicit **operator SerializeableQuaternion** (Quaternion data)
- static **operator Quaternion** ([SerializeableQuaternion](#) data)

Public Attributes

- float **x**

Private Attributes

- float **y**
- float **z**
- float **w**

5.60.1 Detailed Description

A serializeable version of quaternion

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Saving/UCSavingManager.cs

5.61 uConstruct.Core.Saving.SerializeableVector3 Class Reference

A serializeable version of the vector3

Public Member Functions

- **SerializeableVector3** (float x, float y, float z)

Static Public Member Functions

- static implicit **operator SerializeableVector3** (Vector3 data)
- static **operator Vector3** ([SerializeableVector3](#) data)
- static [SerializeableVector3](#) **operator-** ([SerializeableVector3](#) a, Vector3 b)

Public Attributes

- float **x**

Private Attributes

- float **y**
- float **z**

5.61.1 Detailed Description

A serializeable version of the vector3

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Saving/UCSavingManager.cs

5.62 uConstruct.Sockets.SnapPointEditor Class Reference

Inheritance diagram for uConstruct.Sockets.SnapPointEditor:

Public Member Functions

- override void **OnInspectorGUI** ()

Private Attributes

- [BaseSnapPoint](#) **script**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Buildings/SnapPointEditor.cs

5.63 uConstruct.Sockets.SocketBuildingData Class Reference

Inheritance diagram for uConstruct.Sockets.SocketBuildingData:

Public Member Functions

- **SocketBuildingData** ([BaseSocket](#) socket)
- override void **UnPack** (GameObject target)

Public Attributes

- BuildingType **receiveType** = BuildingType.Foundation
- [PlacingRestrictionType](#) **placingType** = PlacingRestrictionType.SocketBased
- [SerializeableVector3](#) **center**
- [SerializeableVector3](#) **size**
- bool **isHoverTarget**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Sockets/Base/BaseSocket.cs

5.64 uConstruct.SocketEditor Class Reference

Inheritance diagram for uConstruct.SocketEditor:

Public Member Functions

- override void **OnEnable** ()
- override void **OnInspectorGUI** ()

Public Attributes

- SerializedProperty **receiveType**
- SerializedProperty **placingType**
- SerializedProperty **isHoverTarget**
- SerializedProperty **drawIndividual**
- SerializedProperty **previewObject**

Private Member Functions

- void **OnDisable** ()
- void **OnPreviewChanged** (GameObject _target)

Private Attributes

- GUIStyle **boxStyle**
- [BaseSocket](#) **script**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Buildings/SocketEditor.cs

5.65 uConstruct.Sockets.SOverlapThreshold Class Reference

Static Public Member Functions

- static void **DetectOverlap** (Vector3 pos, [BaseSocket](#)[] targets)

Public Attributes

- readonly List< [BaseSocket](#) > **overlapping** = new List<[BaseSocket](#)>()

Static Private Member Functions

- static float **fastDistance** (Vector3 objA, Vector3 objB)

Static Private Attributes

- static Vector3 **tempPosition**
- static readonly Dictionary< Vector3, [SOverlapThreshold](#) > **overlaps** = new Dictionary<Vector3, [SOverlapThreshold](#)>()

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Sockets/ThresholdManager/SOverlapThreshold.cs

5.66 uConstruct.Core.Templates.Template Class Reference

Inheritance diagram for uConstruct.Core.Templates.Template:

Public Attributes

- [ITemplateObject](#)[] **templateObjects**
- string **templateName**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/Template.cs

5.67 uConstruct.Core.Templates.TemplateCreatorEditor Class Reference

Inheritance diagram for uConstruct.Core.Templates.TemplateCreatorEditor:

Public Member Functions

- void **Init** ([BuildingEditor](#) editorWindow)

Private Member Functions

- void **Update** ()
- void **OnGUI** ()
- void **DrawCreateWindow** ()
- void **DrawSelectionList** ()
- void **DrawTemplatesView** ()
- void **DrawToolBox** ()

Private Attributes

- [BuildingEditor](#) **buildingEditor**
- [BaseBuilding](#) **building**
- [TemplateCreationData](#) **onCreationTemplate** = null
- Vector2 **creationScroll** = new Vector2()
- Vector2 **scrollView** = new Vector2()

The documentation for this class was generated from the following files:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/Editor/TemplateEditor/TemplateEditor.cs
- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/Editor/TemplateEditor/Templates↔CreatingEditor.cs
- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/Editor/TemplateEditor/TemplatesList↔View.cs
- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/Editor/TemplateEditor/TemplateTool↔Box.cs

5.68 uConstruct.Core.Templates.TemplateCreationData Class Reference

Public Member Functions

- [ITemplateObject\[\]](#) **ReturnTemplates** ()
- **TemplateCreationData** (Transform transform)
- **TemplateCreationData** ([Template](#) copy, Transform root, string Name)

Public Attributes

- string **name**
- List< [TemplateObjectSelection](#) > **templateObjects** = new List<[TemplateObjectSelection](#)>()
- [Template](#) **editedTemplate**

Properties

- bool **markAll** [get, set]

Private Attributes

- bool **_markAll**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/Editor/TemplateEditor/Templates↵
CreatingEditor.cs

5.69 uConstruct.Core.Templates.TemplateMenuEditor Class Reference

Inheritance diagram for uConstruct.Core.Templates.TemplateMenuEditor:

Static Public Member Functions

- static void **Open** ()

Static Private Member Functions

- static void **UpdateBuilding** ([BaseBuilding](#) building)

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/Editor/TemplateEditor/TemplateEditor.cs

5.70 uConstruct.Core.Templates.TemplateObjectSelection Class Reference

Public Attributes

- [ITemplateObject](#) **templateObject**
- bool **chosen**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/Editor/TemplateEditor/Templates↔
CreatingEditor.cs

5.71 uConstruct.Core.Templates.TemplateSelectionWindow Class Reference

Inheritance diagram for uConstruct.Core.Templates.TemplateSelectionWindow:

Public Member Functions

- void **Init** ([BaseBuilding](#) building)

Properties

- GameObject **selectedTemplate** [get, set]

Private Member Functions

- void **OnGUI** ()

Private Attributes

- GameObject **_selectedTemplate**
- [BaseBuilding](#) **building**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/Editor/TemplateEditor/TemplateSelection↔
Window.cs

5.72 uConstruct.Core.Templates.TemplateUtility Class Reference

Inheritance diagram for uConstruct.Core.Templates.TemplateUtility:

Static Public Member Functions

- static GameObject [GenerateTemplate](#) (string name, [BaseBuilding](#) building, [ITemplateObject\[\]](#) templateTargets, bool copy)

Generate our templates for the building.

Public Attributes

- const string [RESOURCES_PATH](#) = "Templates/"
Our path to the resources folder. (Used with Resources class)

Properties

- static string [PREFAB_PATH](#) [get]
Our path to the prefab folder. (Used with prefab utility)

5.72.1 Member Function Documentation

5.72.1.1 static GameObject uConstruct.Core.Templates.TemplateUtility.GenerateTemplate (string name, BaseBuilding building, ITemplateObject[] templateTargets, bool copy) [static]

Generate our templates for the building.

Parameters

| | |
|------------------------|--|
| <i>name</i> | The name of the template |
| <i>building</i> | what building is the template created for |
| <i>templateTargets</i> | the template objects you want to template |
| <i>copy</i> | Auto-Assign the template into the building |

Returns

The generated template prefab

5.72.2 Member Data Documentation

5.72.2.1 const string uConstruct.Core.Templates.TemplateUtility.RESOURCES_PATH = "Templates/"

Our path to the resources folder. (Used with Resources class)

5.72.3 Property Documentation

5.72.3.1 string uConstruct.Core.Templates.TemplateUtility.PREFAB_PATH [static], [get]

Our path to the prefab folder. (Used with prefab utility)

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Templates/Editor/TemplateUtility.cs

5.73 uConstruct.Conditions.TerrainModification_BlueprintData Class Reference

Inheritance diagram for uConstruct.Conditions.TerrainModification_BlueprintData:

Public Member Functions

- **TerrainModification_BlueprintData** ([TerrainModificationCondition](#) condition)
- override void **UnPack** (GameObject target)

Public Attributes

- bool **revertOnDestroy**
- TerrainModificationType **modificationType**
- int **xScale**
- int **zScale**
- [SerializeableVector3](#) **offset**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/TerrainModification↔
Condition.cs

5.74 uConstruct.Conditions.TerrainModificationCondition Class Reference

This condition is a built-in condition that will clean details around you on place. Should be used for stuff like foundations.

Inheritance diagram for uConstruct.Conditions.TerrainModificationCondition:

Classes

- class [RestoreData](#)

Public Member Functions

- override bool [CheckCondition](#) ()
Called when the building is being placed, checks for the condition.
- override void [Awake](#) ()
Called on awake to make sure rootParent isnt null
- override void [OnDrawGizmos](#) ()
Called when gizmos is drawing, can be used to debug your condition.
- override [BlueprintData](#) [Pack](#) ()
Pack our building data

Public Attributes

- bool **revertOnDestroy** = true
- TerrainModificationType **modificationType**
- int **xScale** = 15
- int **zScale** = 15
- Vector3 **offset** = new Vector3()

Properties

- bool **isDetails** [get]
- bool **isHeight** [get]
- Vector3 **position** [get]
- override bool **DisableOnPlace** [get]

Private Member Functions

- void **HandleTerrainModifications** ()
- void **RestoreTerrainModifications** ()
Restore the terrain modification caused by this building only.

Static Private Member Functions

- static void **AddModificationData** (Terrain terrain, float[,] heights)
Add Heights To A Terrain
- static void **AddModificationData** (Terrain terrain, int[,] details, int layersIndex)
Add Details Data To Terrain
- static void **RevertModifications** ()
Revert the terrain modifications globally

Private Attributes

- List< [TerrainModificationData](#) > **savedDetailData** = new List<[TerrainModificationData](#)>()
- Terrain **terrain**
- int **detailIndex**
- int **xBase**
- int **zBase**
- TerrainData **terrainData**
- int[,] **details**
- float[,] **heights**
- Vector3 **HalfScale**
- Vector3 **terrainPoint**
- Vector3 **normalizedPos**

Static Private Attributes

- static Dictionary< Terrain, TerrainData > **globalDefaultData** = new Dictionary<Terrain, TerrainData>()
- static Dictionary< Terrain, [RestoreData](#) > **restoreData** = new Dictionary<Terrain, [RestoreData](#)>()
- static bool **restored**

5.74.1 Detailed Description

This condition is a built-in condition that will clean details around you on place. Should be used for stuff like foundations.

5.74.2 Member Function Documentation

5.74.2.1 `static void uConstruct.Conditions.TerrainModificationCondition.AddModificationData (Terrain terrain, float heights[,])` `[static], [private]`

Add Heights To A Terrain

Parameters

| | |
|----------------|----------------------------|
| <i>terrain</i> | Specific Terrain |
| <i>heights</i> | The heights of the terrain |

5.74.2.2 `static void uConstruct.Conditions.TerrainModificationCondition.AddModificationData (Terrain terrain, int details[,], int layersIndex)` `[static], [private]`

Add Details Data To Terrain

Parameters

| | |
|--------------------|--------------------|
| <i>terrain</i> | A specific terrain |
| <i>details</i> | terrain's details |
| <i>layersIndex</i> | detail's layer |

5.74.2.3 `override void uConstruct.Conditions.TerrainModificationCondition.Awake ()` `[virtual]`

Called on awake to make sure rootParent isnt null

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.74.2.4 `override bool uConstruct.Conditions.TerrainModificationCondition.CheckCondition ()` `[virtual]`

Called when the building is being placed, checks for the condition.

Returns

Is the condition applied?

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.74.2.5 `override void uConstruct.Conditions.TerrainModificationCondition.OnDrawGizmos () [virtual]`

Called when gizmos is drawing, can be used to debug your condition.

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.74.2.6 `override BlueprintData uConstruct.Conditions.TerrainModificationCondition.Pack () [virtual]`

Pack our building data

Returns

our building data

Reimplemented from [uConstruct.Conditions.BaseCondition](#).

5.74.2.7 `void uConstruct.Conditions.TerrainModificationCondition.RestoreTerrainModifications () [private]`

Restore the terrain modification caused by this building only.

5.74.2.8 `static void uConstruct.Conditions.TerrainModificationCondition.RevertModifications () [static], [private]`

Revert the terrain modifications globally

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/TerrainModification↔
Condition.cs

5.75 uConstruct.Conditions.TerrainModificationData Class Reference

Public Member Functions

- **TerrainModificationData** (int _xIndex, int _yIndex, int _detailLayer, int[,] _details, Terrain _terrain)
- **TerrainModificationData** (int _xIndex, int _yIndex, float[,] _heights, int xScale, int zScale, Terrain _terrain)

Public Attributes

- int **xIndex**
- int **yIndex**
- int **xScale**
- int **zScale**
- int[,] **details**
- float[,] **heights**
- int **layer**
- Terrain **terrain**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Conditions/BasicConditions/TerrainModification↔
Condition.cs

5.76 uConstruct.Core.Threading.ThreadManager Class Reference

This class handles the multi-threading mechanics of [uConstruct](#).

Static Public Member Functions

- static void [StartThread](#) ()
Start our thread
- static void [StopThread](#) ()
Stop the thread
- static void [UpdateThread](#) ()
Update thread
- static void [UnityThread](#) ()
Access to unity thread
- static void [RunOnUnityThread](#) (IThreadTask action)
Add an action to the unity thread
- static void [RunOnUConstructThread](#) (IThreadTask action)
Add an action to the [uConstruct](#) thread

Properties

- static bool **enabled** [get, set]

Static Private Member Functions

- static void [RunThread](#) ()
Run Thread

Static Private Attributes

- static bool [_enabled](#) = true
Will the system multi-thread calculations in order to remove overload from main thread?
- static List< [IThreadTask](#) > [UConstructThreadActions](#) = new List<IThreadTask>()
Queued thread actions
- static List< [IThreadTask](#) > [UnityThreadQueuedActions](#) = new List<IThreadTask>()
List of all queued unity thread actions
- static Thread [thread](#) = new Thread([RunThread](#))
Our thread instance.
- static bool [isRunning](#) = false
should the thread run
- static bool [isUpdate](#)
should the thread update

5.76.1 Detailed Description

This class handles the multi-threading mechanics of [uConstruct](#).

5.76.2 Member Function Documentation

- 5.76.2.1 static void [uConstruct.Core.Threading.ThreadManager.RunOnUConstructThread](#) (IThreadTask action)
[static]

Add an action to the [uConstruct](#) thread

Parameters

| | |
|---------------|------------|
| <i>action</i> | the action |
|---------------|------------|

5.76.2.2 `static void uConstruct.Core.Threading.ThreadManager.RunOnUnityThread (IThreadTask action) [static]`

Add an action to the unity thread

Parameters

| | |
|---------------|------------|
| <i>action</i> | the action |
|---------------|------------|

5.76.2.3 `static void uConstruct.Core.Threading.ThreadManager.RunThread () [static],[private]`

Run Thread

5.76.2.4 `static void uConstruct.Core.Threading.ThreadManager.StartThread () [static]`

Start our thread

5.76.2.5 `static void uConstruct.Core.Threading.ThreadManager.StopThread () [static]`

Stop the thread

5.76.2.6 `static void uConstruct.Core.Threading.ThreadManager.UnityThread () [static]`

Access to unity thread

5.76.2.7 `static void uConstruct.Core.Threading.ThreadManager.UpdateThread () [static]`

Update thread

5.76.3 Member Data Documentation

5.76.3.1 `bool uConstruct.Core.Threading.ThreadManager._enabled = true [static],[private]`

Will the system multi-thread calculations in order to remove overload from main thread?

5.76.3.2 `bool uConstruct.Core.Threading.ThreadManager.isRunning = false [static],[private]`

should the thread run

5.76.3.3 `bool uConstruct.Core.Threading.ThreadManager.isUpdate` `[static], [private]`

should the thread update

5.76.3.4 `Thread uConstruct.Core.Threading.ThreadManager.thread = new Thread(RunThread)` `[static], [private]`

Our thread instance.

5.76.3.5 `List<IThreadTask> uConstruct.Core.Threading.ThreadManager.UConstructThreadActions = new List<IThreadTask>()` `[static], [private]`

Queued thread actions

5.76.3.6 `List<IThreadTask> uConstruct.Core.Threading.ThreadManager.UnityThreadQueuedActions = new List<IThreadTask>()` `[static], [private]`

List of all queued unity thread actions

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/MultiThreading/ThreadManager.cs

5.77 uConstruct.Core.Threading.ThreadTask Class Reference

A thread task that takes no parameters.

5.77.1 Detailed Description

A thread task that takes no parameters.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/MultiThreading/ThreadManager.cs

5.78 uConstruct.Core.Threading.ThreadTask Class Reference

A thread task that takes no parameters.

5.78.1 Detailed Description

A thread task that takes no parameters.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/MultiThreading/ThreadManager.cs

5.79 uConstruct.Core.Threading.ThreadTask Class Reference

A thread task that takes no parameters.

5.79.1 Detailed Description

A thread task that takes no parameters.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/MultiThreading/ThreadManager.cs

5.80 uConstruct.Core.Threading.ThreadTask Class Reference

A thread task that takes no parameters.

5.80.1 Detailed Description

A thread task that takes no parameters.

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/MultiThreading/ThreadManager.cs

5.81 uConstruct.TransformExtensions Class Reference

Extension methods for the transform

Static Public Member Functions

- static Vector3 [GetUp](#) (this Transform transform)
The Up corner of the transform
- static Vector3 [GetDown](#) (this Transform transform)
The Down corner of the transform
- static Vector3 [GetRight](#) (this Transform transform)
The Right corner of the transform
- static Vector3 [GetLeft](#) (this Transform transform)
The Left corner of the transform
- static Vector3 [GetForward](#) (this Transform transform)
The Forward corner of the transform
- static Vector3 [GetBackwards](#) (this Transform transform)
The Back corner of the transform
- static bool [FloatPercisionEquals](#) (this Vector3 a, Vector3 b)
A comperasion extension method that checks for equality with hard float percision in account
- static Vector3 [GetRenderersSum](#) (this Transform transform)
Returns the sum size of all the renderers in the transform.
- static Vector3 [GetRendererCenter](#) (this Transform transform)
Returns solo renderers center, not encapsulated.
- static Vector3 [GetRendererSize](#) (this Transform transform)
Returns solo renderers size, not encapsulated.
- static Quaternion [Subside](#) (this Quaternion a, Quaternion b)
Subside 2 quaternions.

5.81.1 Detailed Description

Extension methods for the transform

5.81.2 Member Function Documentation

5.81.2.1 `static bool uConstruct.TransformExtensions.FloatPercisionEquals (this Vector3 a, Vector3 b) [static]`

A comperasion extension method that checks for equality with hard float percision in account

Parameters

| | |
|----------|---|
| <i>a</i> | the Vector3 instance |
| <i>b</i> | the second Vector3 you want to compare to |

Returns

Are they equal ?

5.81.2.2 `static Vector3 uConstruct.TransformExtensions.GetBackwards (this Transform transform) [static]`

The Back corner of the transform

Parameters

| | |
|------------------|---------------|
| <i>transform</i> | our transform |
|------------------|---------------|

Returns

BACK corner

5.81.2.3 `static Vector3 uConstruct.TransformExtensions.GetDown (this Transform transform) [static]`

The Down corner of the transform

Parameters

| | |
|------------------|---------------|
| <i>transform</i> | our transform |
|------------------|---------------|

Returns

DOWN corner

5.81.2.4 static Vector3 uConstruct.TransformExtensions.GetForward (this Transform *transform*) [static]

The Forward corner of the transform

Parameters

| | |
|------------------|---------------|
| <i>transform</i> | our transform |
|------------------|---------------|

Returns

FORWARD corner

5.81.2.5 `static Vector3 uConstruct.TransformExtensions.GetLeft (this Transform transform)` `[static]`

The Left corner of the transform

Parameters

| | |
|------------------|---------------|
| <i>transform</i> | our transform |
|------------------|---------------|

Returns

LEFT corner

5.81.2.6 `static Vector3 uConstruct.TransformExtensions.GetRendererCenter (this Transform transform)` `[static]`

Returns solo renderers center, not encapsulated.

Parameters

| | |
|------------------|--------------------|
| <i>transform</i> | transform instance |
|------------------|--------------------|

Returns

center of the chosen target.

5.81.2.7 `static Vector3 uConstruct.TransformExtensions.GetRendererSize (this Transform transform)` `[static]`

Returns solo renderers size, not encapsulated.

Parameters

| | |
|------------------|--------------------|
| <i>transform</i> | transform instance |
|------------------|--------------------|

Returns

size of the chosen target.

5.81.2.8 static Vector3 uConstruct.TransformExtensions.GetRenderersSum (this Transform *transform*) [static]

Returns the sum size of all the renderers in the transform.

Parameters

| | |
|------------------|--------------------|
| <i>transform</i> | transform instance |
|------------------|--------------------|

Returns

size sum of all renderers in the transform.

5.81.2.9 static Vector3 uConstruct.TransformExtensions.GetRight (this Transform *transform*) [static]

The Right corner of the transform

Parameters

| | |
|------------------|---------------|
| <i>transform</i> | our transform |
|------------------|---------------|

Returns

RIGHT corner

5.81.2.10 static Vector3 uConstruct.TransformExtensions.GetUp (this Transform *transform*) [static]

The Up corner of the transform

Parameters

| | |
|------------------|---------------|
| <i>transform</i> | our transform |
|------------------|---------------|

Returns

UP corner

5.81.2.11 static Quaternion uConstruct.TransformExtensions.Subside (this Quaternion *a*, Quaternion *b*) [static]

Subside 2 quaternions.

Parameters

| | |
|----------|-------------------------|
| <i>a</i> | our quaternion instance |
| <i>b</i> | subside from |

Returns

our subsided result

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Buildings/Base/BaseBuilding.cs

5.82 uConstruct.UC_EditorUtility Class Reference

Static Public Member Functions

- static bool [DisplayScriptField](#) (Editor editor)
Display a script field property, like the one unity draws.
- static void **DisplayObjectField** (System.Action< GameObject > onClose, bool allowSceneObjects)
- static void **OnGUI** ()

Static Private Attributes

- static GameObject **target**
- static System.Action< GameObject > **onClose**

5.82.1 Member Function Documentation

5.82.1.1 static bool uConstruct.UC_EditorUtility.DisplayScriptField (Editor editor) [static]

Display a script field property, like the one unity draws.

Parameters

| | |
|---------------|-----------------------|
| <i>editor</i> | The referenced editor |
|---------------|-----------------------|

Returns

did we change our script type?

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Utility/UC_EditorUtility.cs

5.83 uConstruct.Core.Manager.UCCallbacksManager Class Reference

This class needs to be initiated on the start on the game and it handles loading and saving, it has control over all unity callbacks and you can use it to add some static OnApplicationQuit callbacks.

Inheritance diagram for uConstruct.Core.Manager.UCCallbacksManager:

Public Member Functions

- void [AddApplicationQuitAction](#) (System.Action action)
Add an action to the application quit data

Static Public Member Functions

- static void [CreateAndInitialize](#) ()
Initialize and create an instance of the callbacks manager

Public Attributes

- const string [ProjectName](#) = "uConstruct"
Project name ([uConstruct](#) folder name).

Properties

- static string [ProjectPath](#) [get]
The found path to the project directory (based on the name provided on [ProjectName](#)).
- static [UCCallbacksManager](#) **instance** [get]

Private Member Functions

- void [OnApplicationQuit](#) ()
Called when application quits
- void [Awake](#) ()
Assign instance and start thread.
- void [Start](#) ()
Load data on start to avoid miss-order.
- void [Update](#) ()
Update the unity thread.

Private Attributes

- List< System.Action > **OnApplicationQuitActions** = new List<System.Action>()

Static Private Attributes

- static [UCCallbacksManager](#) **_instance**

5.83.1 Detailed Description

This class needs to be initiated on the start on the game and it handles loading and saving, it has control over all unity callbacks and you can use it to add some static OnApplicationQuit callbacks.

5.83.2 Member Function Documentation

5.83.2.1 void uConstruct.Core.Manager.UCCallbacksManager.AddApplicationQuitAction (System.Action action)

Add an action to the application quit data

Parameters

| | |
|---------------|------------|
| <i>action</i> | the action |
|---------------|------------|

5.83.2.2 `void uConstruct.Core.Manager.UCCallbacksManager.Awake () [private]`

Assign instance and start thread.

5.83.2.3 `static void uConstruct.Core.Manager.UCCallbacksManager.CreateAndInitialize () [static]`

Initialize and create an instance of the callbacks manager

5.83.2.4 `void uConstruct.Core.Manager.UCCallbacksManager.OnApplicationQuit () [private]`

Called when application quits

5.83.2.5 `void uConstruct.Core.Manager.UCCallbacksManager.Start () [private]`

Load data on start to avoid miss-order.

5.83.2.6 `void uConstruct.Core.Manager.UCCallbacksManager.Update () [private]`

Update the unity thread.

5.83.3 Member Data Documentation

5.83.3.1 `const string uConstruct.Core.Manager.UCCallbacksManager.ProjectName = "uConstruct"`

Project name ([uConstruct](#) folder name).

5.83.4 Property Documentation

5.83.4.1 `string uConstruct.Core.Manager.UCCallbacksManager.ProjectPath [static],[get]`

The found path to the project directory (based on the name provided on ProjectName).

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Manager/UCCallbacksManager.cs

5.84 uConstruct.Demo.uConstruct_FirstPersonController Class Reference

Inheritance diagram for uConstruct.Demo.uConstruct_FirstPersonController:

Public Attributes

- bool **getInputsMouse** = true

Private Member Functions

- void **Start** ()
- void **Update** ()
- void **PlayLandingSound** ()
- void **FixedUpdate** ()
- void **PlayJumpSound** ()
- void **ProgressStepCycle** (float speed)
- void **PlayFootStepAudio** ()
- void **GetInput** (out float speed)
- void **RotateView** ()
- void **OnControllerColliderHit** (ControllerColliderHit hit)

Private Attributes

- bool **m_IsWalking**
- float **m_WalkSpeed**
- float **m_RunSpeed**
- float **m_RunstepLenghten**
- float **m_JumpSpeed**
- float **m_StickToGroundForce**
- float **m_GravityMultiplier**
- [uConstruct_MouseLook](#) **m_MouseLook**
- bool **m_UseFovKick**
- float **m_StepInterval**
- AudioClip[] **m_FootstepSounds**
- AudioClip **m_JumpSound**
- AudioClip **m_LandSound**
- Camera **m_Camera**
- bool **m_Jump**
- float **m_YRotation**
- Vector2 **m_Input**
- Vector3 **m_MoveDir** = Vector3.zero
- CharacterController **m_CharacterController**
- CollisionFlags **m_CollisionFlags**
- bool **m_PreviouslyGrounded**
- float **m_StepCycle**
- float **m_NextStep**
- bool **m_Jumping**
- AudioSource **m_AudioSource**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Demo/PlayerScripts/uConstruct_FirstPersonController.cs

5.85 uConstruct.Demo.uConstruct_MouseLook Class Reference

Public Member Functions

- void **Init** (Transform character, Transform camera)
- void **LookRotation** (Transform character, Transform camera)

Public Attributes

- float **XSensitivity** = 2f
- float **YSensitivity** = 2f
- bool **clampVerticalRotation** = true
- float **MinimumX** = -90F
- float **MaximumX** = 90F
- bool **smooth**
- float **smoothTime** = 5f

Private Member Functions

- Quaternion **ClampRotationAroundXAxis** (Quaternion q)

Private Attributes

- Quaternion **m_CharacterTargetRot**
- Quaternion **m_CameraTargetRot**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Demo/PlayerScripts/uConstruct_MouseLook.cs

5.86 uConstruct.UConstructManager Class Reference

Inheritance diagram for uConstruct.UConstructManager:

Static Public Member Functions

- static void **OpenWindow** ()
- static void **UpdateCompilingDefines** ()

Private Member Functions

- void **OnGUI** ()
- void **ReadMeBtn** ()
- void **CreateMissingLayersBtn** ()

Static Private Member Functions

- static bool **CreateLayer** (SerializedObject layersManager, SerializedProperty layers, string layerName)
- static int **GetEmptyLayerIndex** (SerializedProperty layers)
- static bool **CheckIfLayerExists** (SerializedProperty layers, string layer)

Private Attributes

- Vector2 **_scrollPos** = Vector3.zero
- const string **UConstructDefine** = "UCONSTRUCT_PRESET"

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Core/uConstructManager.cs

5.87 uConstruct.Core.Physics.UCPhysics Class Reference

This class handles all custom physics.

Static Public Member Functions

- static void [AddPhysicsObject](#) ([UCPhysicsObject](#) pObject)
Add a physics object to the physics simulation
- static void [RemovePhysicsObject](#) ([UCPhysicsObject](#) pObject)
Remove a physics object from the physics simulation
- static [UCPhysicsHitsArray RaycastAll](#) (Vector3 origin, Vector3 direction, float distance, int mask, float offset)
Create a raycast
- static [UCPhysicsHitsArray RaycastAll](#) (Ray ray, float distance, int mask)
Creates a raycast
- static [UCPhysicsHitsArray RaycastAll](#) (Ray ray, float distance, int mask, float offset)
Creates a raycast
- static bool [Raycast](#) (Vector3 origin, Vector3 direction, out [UCPhysicsHit](#) hit, float distance, int mask, float offset)
Creates a raycast
- static bool [Raycast](#) (Vector3 origin, Vector3 direction, out [UCPhysicsHit](#) hit, float distance, int mask)
Creates a raycast
- static bool [Raycast](#) (Vector3 origin, Vector3 direction, out [UCPhysicsHit](#) hit, float distance)
Creates a raycast
- static bool [Raycast](#) (Ray ray, out [UCPhysicsHit](#) hit, float distance, bool TakeUnityPhysicsIntoAccount, Transform target)
Creates a raycast
- static bool [Raycast](#) (Ray ray, out [UCPhysicsHit](#) hit, float distance, int mask)
Creates a raycast

Static Public Attributes

- static readonly List< [UCPhysicsObject](#) > **physicsObjects** = new List<[UCPhysicsObject](#)>()

Static Private Attributes

- static [UCPhysicsObject](#) **currentPObject**
- static RaycastHit **rayHit**
- static Ray **ray**
- static [UCPhysicsHitsArray](#) **hits**

5.87.1 Detailed Description

This class handles all custom physics.

5.87.2 Member Function Documentation

5.87.2.1 static void `uConstruct.Core.Physics.UCPhysics.AddPhysicsObject (UCPhysicsObject pObject)` `[static]`

Add a physics object to the physics simulation

Parameters

| | |
|----------------|----------------------------|
| <i>pObject</i> | the object you want to add |
|----------------|----------------------------|

5.87.2.2 static bool `uConstruct.Core.Physics.UCPhysics.Raycast (Vector3 origin, Vector3 direction, out UCPhysicsHit hit, float distance, int mask, float offset)` `[static]`

Creates a raycast

Parameters

| | |
|------------------|------------------------------|
| <i>origin</i> | the origin of the raycast |
| <i>direction</i> | the direction of the raycast |
| <i>hit</i> | returns the hit data |
| <i>distance</i> | max distance |
| <i>mask</i> | layerMask |
| <i>offset</i> | raycast offset |

Returns

did we hit something?

5.87.2.3 static bool `uConstruct.Core.Physics.UCPhysics.Raycast (Vector3 origin, Vector3 direction, out UCPhysicsHit hit, float distance, int mask)` `[static]`

Creates a raycast

Parameters

| | |
|------------------|------------------------------|
| <i>origin</i> | the origin of the raycast |
| <i>direction</i> | the direction of the raycast |
| <i>hit</i> | returns the hit data |
| <i>distance</i> | max distance |
| <i>mask</i> | layerMask |

Returns

did we hit something?

5.87.2.4 `static bool uConstruct.Core.Physics.UCPhysics.Raycast (Vector3 origin, Vector3 direction, out UCPhysicsHit hit, float distance) [static]`

Creates a raycast

Parameters

| | |
|------------------|--------------------------|
| <i>origin</i> | The origin of the ray |
| <i>direction</i> | The direction of the ray |
| <i>hit</i> | The hit data of the ray |
| <i>distance</i> | max distance |

Returns

did we hit something?

5.87.2.5 `static bool uConstruct.Core.Physics.UCPhysics.Raycast (Ray ray, out UCPhysicsHit hit, float distance, bool TakeUnityPhysicsIntoAccount, Transform target) [static]`

Creates a raycast

Parameters

| | |
|-----------------|------------------------|
| <i>ray</i> | the ray of the raycast |
| <i>hit</i> | the hit data |
| <i>distance</i> | max distance |

Returns

5.87.2.6 `static bool uConstruct.Core.Physics.UCPhysics.Raycast (Ray ray, out UCPhysicsHit hit, float distance, int mask) [static]`

Creates a raycast

Parameters

| | |
|-----------------|------------------------|
| <i>ray</i> | the ray of the raycast |
| <i>hit</i> | returns the hit data |
| <i>distance</i> | max distance |
| <i>mask</i> | layerMask |

Returns

did we hit something?

5.87.2.7 static **UCPhysicsHitsArray** uConstruct.Core.Physics.UCPhysics.RaycastAll (**Vector3** *origin*, **Vector3** *direction*, float *distance*, int *mask*, float *offset*) [static]

Create a raycast

Parameters

| | |
|------------------|------------------------------|
| <i>origin</i> | The origin of the raycast |
| <i>direction</i> | The direction of the raycast |
| <i>distance</i> | max distance |
| <i>mask</i> | mask |
| <i>offset</i> | raycast offset |

Returns

The hits value

5.87.2.8 static **UCPhysicsHitsArray** uConstruct.Core.Physics.UCPhysics.RaycastAll (**Ray** *ray*, float *distance*, int *mask*) [static]

Creates a raycast

Parameters

| | |
|-----------------|------------------------|
| <i>ray</i> | The ray of the raycast |
| <i>distance</i> | Max distance |
| <i>mask</i> | LayerMask |

Returns

Returns the hits array

5.87.2.9 static **UCPhysicsHitsArray** uConstruct.Core.Physics.UCPhysics.RaycastAll (**Ray** *ray*, float *distance*, int *mask*, float *offset*) [static]

Creates a raycast

Parameters

| | |
|-----------------|------------------------|
| <i>ray</i> | The ray of the raycast |
| <i>distance</i> | Max distance |
| <i>mask</i> | LayerMask |
| <i>offset</i> | raycast offset |

Returns

Returns the hits array

5.87.2.10 `static void uConstruct.Core.Physics.UCPhysics.RemovePhysicsObject (UCPhysicsObject pObject)`
`[static]`

Remove a physics object from the physics simulation

Parameters

| | |
|----------------|-------------------------------|
| <i>pObject</i> | the object you want to remove |
|----------------|-------------------------------|

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Physics/UCPhysics.cs

5.88 uConstruct.Core.Physics.UCPhysicsHit Class Reference

A class that holds the data for the hit data

Public Member Functions

- void [Convert](#) (RaycastHit hit)
Convert a raycastHit to UCPhysicsHit

Public Attributes

- Transform **transform**
- Vector3 **point**
- Vector3 **normal** = -Vector3.one
- float **distance**

5.88.1 Detailed Description

A class that holds the data for the hit data

5.88.2 Member Function Documentation

5.88.2.1 `void uConstruct.Core.Physics.UCPhysicsHit.Convert (RaycastHit hit)`

Convert a raycastHit to [UCPhysicsHit](#)

Parameters

| | |
|------------|--|
| <i>hit</i> | |
|------------|--|

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Physics/UCPhysics.cs

5.89 uConstruct.Core.Physics.UCPhysicsHitsArray Class Reference

An custom array that holds all ray results in an array

Public Member Functions

- void **AddToList** ([UCPhysicsHit](#) hit)
- void **Sort** ()

Properties

- [UCPhysicsHit](#) **this**[int index] [get]
- int **Count** [get]

Private Attributes

- List< [UCPhysicsHit](#) > **_data**

5.89.1 Detailed Description

An custom array that holds all ray results in an array

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Physics/UCPhysics.cs

5.90 uConstruct.Core.Physics.UCPhysicsObject Class Reference

This is a base class for a [UCPhysicsObject](#). Every class that inherits this class will be counted in the physics system.

Inheritance diagram for uConstruct.Core.Physics.UCPhysicsObject:

Public Member Functions

- bool [Raycast](#) (Ray ray, out [UCPhysicsHit](#) _hit, LayerMask mask)
Raycast the physics object

Public Attributes

- Vector3 **_size** = Vector3.one
- bool **_usePhysics** = true

Static Public Attributes

- static Color **GizmosColor** = new Color(0, 0, 0, 0.2f)

Properties

- Vector3 **center** [get, set]
- Vector3 **size** [get, set]
- bool **usePhysics** [get, set]

Private Member Functions

- void [UpdateBounds](#) (Vector3 center, Vector3 size)
Update object's bounds
- void [Start](#) ()
Add to physics simulation and update bounds
- void [OnEnable](#) ()
Add object to physics simulation
- void [OnDisable](#) ()
Remove physics object from physics simulation
- void [OnDrawGizmos](#) ()
Draw gizmos
- void [DrawShape](#) (Matrix4x4 matrix, bool selected)
Draw the shape of the bounds
- bool **VerifyUnityCollisions** ([UCPhysicsHit](#) hit, Ray ray)

Private Attributes

- Vector3 **_center** = Vector3.zero
- Bounds **Bounds**
- [UCPhysicsHit](#) **hit** = new [UCPhysicsHit](#)()
- bool **inList** = false

5.90.1 Detailed Description

This is a base class for a [UCPhysicsObject](#). Every class that inherits this class will be counted in the physics system.

5.90.2 Member Function Documentation

5.90.2.1 void uConstruct.Core.Physics.UCPhysicsObject.DrawShape (Matrix4x4 matrix, bool selected) [private]

Draw the shape of the bounds

Parameters

| | |
|-----------------|-----------------------------------|
| <i>matrix</i> | the matrix of the bounds |
| <i>selected</i> | is the shape selected in heirachy |

5.90.2.2 `void uConstruct.Core.Physics.UCPhysicsObject.OnDisable () [private]`

Remove physics object from physics simulation

5.90.2.3 `void uConstruct.Core.Physics.UCPhysicsObject.OnDrawGizmos () [private]`

Draw gizmos

5.90.2.4 `void uConstruct.Core.Physics.UCPhysicsObject.OnEnable () [private]`

Add object to physics simulation

5.90.2.5 `bool uConstruct.Core.Physics.UCPhysicsObject.Raycast (Ray ray, out UCPhysicsHit _hit, LayerMask mask)`

Raycast the physics object

Parameters

| | |
|------------------|---------------|
| <i>origin</i> | ray origin |
| <i>direction</i> | ray direction |
| <i>_hit</i> | hit data |
| <i>distance</i> | max distance |
| <i>mask</i> | layerMask |

Returns

Did we hit something?

5.90.2.6 `void uConstruct.Core.Physics.UCPhysicsObject.Start () [private]`

Add to physics simulation and update bounds

5.90.2.7 `void uConstruct.Core.Physics.UCPhysicsObject.UpdateBounds (Vector3 center, Vector3 size) [private]`

Update object's bounds

Parameters

| | |
|---------------|--------------------------------------|
| <i>center</i> | The center of the bounds, worldspace |
| <i>size</i> | The size of the bounds, worldspace |

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Physics/UCPhysicsObject.cs

5.91 uConstruct.Core.Saving.UCSavedItem Interface Reference

An interface that each saveable object in the scene needs to have.

Inheritance diagram for uConstruct.Core.Saving.UCSavedItem:

Public Member Functions

- [BaseUCSaveData Save](#) ()
Save data

5.91.1 Detailed Description

An interface that each saveable object in the scene needs to have.

5.91.2 Member Function Documentation

5.91.2.1 BaseUCSaveData uConstruct.Core.Saving.UCSavedItem.Save ()

Save data

Returns

our save result

Implemented in [uConstruct.BaseBuildingGroup](#).

The documentation for this interface was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Saving/UCSavingManager.cs

5.92 uConstruct.Core.Saving.UCSavingManager Class Reference

This class handles all the saving management of the asset.

Static Public Member Functions

- static [BaseUCSaveData\[\] DeserializeStream](#) (Stream stream)
Deserialize a given stream
- static void [Load](#) ()
Load all data from files
- static void [LoadExternalData](#) (Stream stream)
Load an external data
- static void [Save](#) ()
Save all data into a file
- static Stream [Serialize](#) (Stream stream, out [BaseUCSaveData\[\]](#) data)
Serialize our save data.
- static Stream [Serialize](#) (Stream stream)
Serialize our save data.
- static void [DestoryDataOnCurrentScene](#) ()
Destroy all of the data objects on the current scene.

Static Public Attributes

- static string [fileName](#) = "UConstructData"
Saving file name (for example : saveData)
- static string [fileFormat](#) = "bin"
Saving file format (for example : bin)

Properties

- static string [dataPath](#) [get]
Full path to the current save
- static string [folderPath](#) [get]
The folder path to the saves.
- static bool [enabled](#) [get, set]
Will uConstruct save your buildings
- static bool [IsLoading](#) [get]
- static bool [renderVisualSave](#) [get, set]

Events

- static SavingProcessComplete [OnSavingProcessComplete](#)
- static LoadingProcessComplete [OnLoadingProcessComplete](#)

Static Private Member Functions

- static List< [BaseUCSaveData](#) > [ReturnSaveData](#) ()
Return all save data from the objects in the scene

Static Private Attributes

- static bool [_isLoading](#)
- static bool [_renderVisualSave](#)

5.92.1 Detailed Description

This class handles all the saving management of the asset.

5.92.2 Member Function Documentation

5.92.2.1 `static BaseUCSaveData [] uConstruct.Core.Saving.UCSavingManager.DeserializeStream (Stream stream)`
[static]

Deserialize a given stream

Parameters

| | |
|---------------|------------|
| <i>stream</i> | our stream |
|---------------|------------|

Returns

the deserialized Stream

5.92.2.2 `static void uConstruct.Core.Saving.UCSavingManager.DestroyDataOnCurrentScene ()` [static]

Destroy all of the data objects on the current scene.

5.92.2.3 `static void uConstruct.Core.Saving.UCSavingManager.Load ()` [static]

Load all data from files

5.92.2.4 `static void uConstruct.Core.Saving.UCSavingManager.LoadExternalData (Stream stream)` [static]

Load an external data

Parameters

| | |
|---------------|-----------|
| <i>stream</i> | our data. |
|---------------|-----------|

5.92.2.5 `static List<BaseUCSaveData> uConstruct.Core.Saving.UCSavingManager.ReturnSaveData ()` [static],
[private]

Return all save data from the objects in the scene

Returns

array of the saving data

5.92.2.6 `static void uConstruct.Core.Saving.UCSavingManager.Save () [static]`

Save all data into a file

5.92.2.7 `static Stream uConstruct.Core.Saving.UCSavingManager.Serialize (Stream stream, out BaseUCSaveData[] data) [static]`

Serialize our save data.

Parameters

| | |
|---------------|------------|
| <i>stream</i> | our stream |
| <i>data</i> | our data |

Returns

result data

5.92.2.8 `static Stream uConstruct.Core.Saving.UCSavingManager.Serialize (Stream stream) [static]`

Serialize our save data.

Parameters

| | |
|---------------|------------|
| <i>stream</i> | our stream |
|---------------|------------|

Returns

result data

5.92.3 Member Data Documentation

5.92.3.1 `string uConstruct.Core.Saving.UCSavingManager.fileFormat = "bin" [static]`

[Saving](#) file format (for example : bin)

5.92.3.2 `string uConstruct.Core.Saving.UCSavingManager.fileName = "UConstructData" [static]`

[Saving](#) file name (for example : saveData)

5.92.4 Property Documentation

5.92.4.1 `string uConstruct.Core.Saving.UCSavingManager.dataPath [static],[get]`

Full path to the current save

5.92.4.2 `bool uConstruct.Core.Saving.UCSavingManager.enabled` `[static], [get], [set]`

Will [uConstruct](#) save your buildings

5.92.4.3 `string uConstruct.Core.Saving.UCSavingManager.folderPath` `[static], [get]`

The folder path to the saves.

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Saving/UCSavingManager.cs`

5.93 uConstruct.UCSettingAttribute Class Reference

Inheritance diagram for `uConstruct.UCSettingAttribute`:

Public Member Functions

- **UCSettingAttribute** (UCSettingCategories category, string name)
- **UCSettingAttribute** (UCSettingCategories category, string name, string desc)
- object **Draw** (object instance)

Public Attributes

- UCSettingCategories **category**
- string **name**
- string **desc**

Properties

- GUIContent **content** `[get]`

Private Member Functions

- bool **CheckType** (System.Type type, System.Type target)

Private Attributes

- GUIContent **_content**

The documentation for this class was generated from the following file:

- `D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Manager/UCSettings.cs`

5.94 uConstruct.UCSettingCategory Class Reference

Public Member Functions

- **UCSettingCategory** (UCSettingCategories category)

Static Public Member Functions

- static **UCSettingCategory GetCategory** (UCSettingCategories category)

Public Attributes

- bool **show**
- UCSettingCategories **type**
- List< **UCSettingAttribute** > **attributes** = new List<**UCSettingAttribute**>()
- List< FieldInfo > **fields** = new List<FieldInfo>()

Properties

- static List< **UCSettingCategory** > **categories** [get]

Static Private Attributes

- static List< **UCSettingCategory** > **_categories**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Manager/UCSettings.cs

5.95 uConstruct.UCSettings Class Reference

Inheritance diagram for uConstruct.UCSettings:

Public Attributes

- const string **fileName** = "uConstructSettings"
- SavingPathType **UCSavingPathType**
- bool **UCSavingEnabled** = true
- bool **UCThreadingEnabled** = true
- bool **UCBatchingEnabled** = true
- int **UCBatchingLODLevels** = 3

Properties

- static [UCSettings](#) **instance** [get]

Static Private Attributes

- static [UCSettings](#) **_instance**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Core/Manager/UCSettings.cs

5.96 uConstruct.UCSettingsEditor Class Reference

Inheritance diagram for uConstruct.UCSettingsEditor:

Static Public Member Functions

- static void **Open** ()

Public Attributes

- [UCSettings](#) **_settings**

Properties

- [UCSettings](#) **settings** [get]

Private Member Functions

- void **OnGUI** ()

Private Attributes

- GUIStyle **invisibleButtonStyle**
- GUIStyle **boxStyle**
- Vector2 **scrollPos**

The documentation for this class was generated from the following file:

- D:/Projects/uConstruct/Assets/uConstruct/Scripts/Editor/Managers/UCSettingsEditor.cs

Index

- [_defaultLockCursor](#)
 - [uConstruct::Demo::BuildingPlacer, 77](#)
 - [_destroyBuildings](#)
 - [uConstruct::Demo::BuildingPlacer, 77](#)
 - [_enabled](#)
 - [uConstruct::Core::Threading::ThreadManager, 119](#)
 - [_rotatedWithPlayer](#)
 - [uConstruct::Demo::BuildingPlacer, 77](#)
 - [_rotationValue](#)
 - [uConstruct::Demo::BuildingPlacer, 77](#)
 - [_uid](#)
 - [uConstruct::BaseBuilding, 37](#)
 - [AOIGroup](#)
 - [uConstruct::BaseBuildingGroup, 40](#)
 - [ActivateColliders](#)
 - [uConstruct::BaseBuilding, 29](#)
 - [ActivateConditions](#)
 - [uConstruct::BaseBuilding, 29](#)
 - [ActivateSnapPoints](#)
 - [uConstruct::BaseBuilding, 30](#)
 - [ActivateSockets](#)
 - [uConstruct::BaseBuilding, 30](#)
 - [Add](#)
 - [uConstruct::BatchData, 56](#)
 - [AddApplicationQuitAction](#)
 - [uConstruct::Core::Manager::UCCallbacks↔
Manager, 127](#)
 - [AddBuilding](#)
 - [uConstruct::BaseBuildingGroup, 40](#)
 - [AddCondition](#)
 - [uConstruct::BaseBuilding, 30](#)
 - [AddControl](#)
 - [uConstruct::Demo::DemoUI, 87](#)
 - [AddField](#)
 - [uConstruct::Core::Blueprints::Blueprint, 60](#)
 - [AddFinder](#)
 - [uConstruct::Core::AOI::AOIManager, 20](#)
 - [AddModificationData](#)
 - [uConstruct::Conditions::TerrainModification↔
Condition, 116](#)
 - [AddPhysicsObject](#)
 - [uConstruct::Core::Physics::UCPhysics, 132](#)
 - [AddSocket](#)
 - [uConstruct::BaseBuilding, 30](#)
 - [AddTarget](#)
 - [uConstruct::Core::AOI::AOIManager, 20](#)
 - [AddTemplate](#)
 - [uConstruct::BaseBuilding, 30](#)
 - [AddToDB](#)
 - [uConstruct::Core::PrefabDatabase::PrefabDB, 100](#)
- [AnchoredPosition](#)
 - [uConstruct::Sockets::BaseSnapPoint, 47](#)
 - [aoiPosition](#)
 - [uConstruct::Core::AOI::BaseAOIFinder, 23](#)
 - [uConstruct::Core::AOI::BaseAOITarget, 25](#)
 - [ApplyChangesToPrefab](#)
 - [uConstruct::PreviewBuilding, 103](#)
 - [ApplyControlsToDemoUI](#)
 - [uConstruct::Demo::BuildingPlacer, 75](#)
 - [AssignOriginalColors](#)
 - [uConstruct::BaseBuilding, 30](#)
 - [Awake](#)
 - [uConstruct::BaseBuilding, 31](#)
 - [uConstruct::BaseBuildingGroup, 41](#)
 - [uConstruct::Conditions::BaseCondition, 44](#)
 - [uConstruct::Conditions::CheckForCollision↔
Condition, 83](#)
 - [uConstruct::Conditions::CheckForGround↔
Condition, 85](#)
 - [uConstruct::Conditions::TerrainModification↔
Condition, 116](#)
 - [uConstruct::Core::Manager::UCCallbacks↔
Manager, 128](#)
 - [uConstruct::Demo::BuildingPlacer, 75](#)
 - [uConstruct::Demo::DemoUI, 87](#)
 - [uConstruct::Sockets::BaseSnapPoint, 48](#)
 - [uConstruct::Sockets::BaseSocket, 51](#)
 - [BLUEPRINT_ASSET_FIRST](#)
 - [uConstruct::Core::Blueprints::Blueprint, 61](#)
 - [BLUEPRINT_ASSET_PATH](#)
 - [uConstruct::Core::Blueprints::Blueprint, 61](#)
 - [Batch](#)
 - [uConstruct::BaseBuildingGroup, 41](#)
 - [Batchable](#)
 - [uConstruct::BatchData, 56](#)
 - [BlueprintEditEditor, 62](#)
 - [BlueprintEditor, 63](#)
 - [BlueprintField](#)
 - [uConstruct::Core::Blueprints::BlueprintField, 64, 65](#)
 - [blueprintName](#)
 - [uConstruct::Core::Blueprints::Blueprint, 61](#)
 - [BuildingDeattached](#)
 - [uConstruct::BaseBuilding, 31](#)
 - [BuildingDestroyed](#)
 - [uConstruct::BaseBuilding, 31](#)
 - [buildingGroup](#)
 - [uConstruct::Core::AOI::BuildingGroupAOITarget, 71](#)

- BuildingGroupChanged
 - uConstruct::BaseBuilding, 31
- BuildingLayers
 - uConstruct::LayersData, 95
- BuildingMask
 - uConstruct::LayersData, 95
- BuildingPlaced
 - uConstruct::BaseBuilding, 31
- BuildingSnapped
 - uConstruct::Sockets::BaseSocket, 51
- buildings
 - uConstruct::BaseBuildingGroup, 43
- CheckCondition
 - uConstruct::Conditions::BaseCondition, 44
 - uConstruct::Conditions::CheckForBuilding↔
Condition, 81
 - uConstruct::Conditions::CheckForCollision↔
Condition, 83
 - uConstruct::Conditions::CheckForGround↔
Condition, 85
 - uConstruct::Conditions::TerrainModification↔
Condition, 116
- CheckConditions
 - uConstruct::BaseBuilding, 31
- CheckForBuildingsEditor, 81
- CompileInitialBatchData
 - uConstruct::BatchUtility, 57
- ComputeAOI
 - uConstruct::Core::AOI::AOIManager, 20
- Contains
 - uConstruct::BatchClass, 55
- Contains
 - uConstruct::Core::Blueprints::BlueprintField, 65
 - uConstruct::Core::PrefabDatabase::PrefabDB, 100
- controls
 - uConstruct::Demo::DemoUI, 88
- controlsCount
 - uConstruct::Demo::DemoUI, 88
- Convert
 - uConstruct::Core::Physics::UCPhysicsHit, 135
- correctPosition
 - uConstruct::Core::AOI::BuildingGroupAOITarget,
71
- Create
 - uConstruct::Sockets::BaseSocket, 51
- CreateAndInitialize
 - uConstruct::Core::Manager::UCCallbacks↔
Manager, 128
- CreateBlueprint
 - uConstruct::Core::Blueprints::Blueprint, 60
- CreateBuildingInstance
 - uConstruct::Demo::BuildingPlacer, 75
- CreateCondition
 - uConstruct::BaseBuilding, 31
- CreateGroup< T >
 - uConstruct::BaseBuildingGroup, 41
- CreateSnapPoint
 - uConstruct::BaseBuilding, 32
- CreateSocket
 - uConstruct::BaseBuilding, 32
- data
 - uConstruct::Core::Blueprints::BlueprintField, 66
- dataBytes
 - uConstruct::Core::Blueprints::BlueprintField, 66
- dataPath
 - uConstruct::Core::Saving::UCSavingManager, 142
- DeAttachBuilding
 - uConstruct::BaseBuilding, 32
- DefaultBuildingLayer
 - uConstruct::LayersData, 95
- DefaultBuildingLayerString
 - uConstruct::LayersData, 96
- DefaultSocketLayer
 - uConstruct::LayersData, 96
- DefaultSocketLayerString
 - uConstruct::LayersData, 96
- Delete
 - uConstruct::Core::Blueprints::Blueprint, 60
- DeserializeStream
 - uConstruct::Core::Saving::UCSavingManager, 141
- DestoryDataOnCurrentScene
 - uConstruct::Core::Saving::UCSavingManager, 141
- DestroyBuilding
 - uConstruct::BaseBuilding, 32
 - uConstruct::BaseBuildingBatcher, 38
 - uConstruct::Demo::BuildingPlacer, 75
 - uConstruct::IBuilding, 92
- DestroyCurrentBuilding
 - uConstruct::Demo::BuildingPlacer, 76
- DestroyGroup
 - uConstruct::BaseBuildingGroup, 41
- DisableOnPlace
 - uConstruct::Conditions::BaseCondition, 45
- DisplayScriptField
 - uConstruct::UC_EditorUtility, 126
- drawIndividual
 - uConstruct::Sockets::BaseSocket, 53
- DrawShape
 - uConstruct::Core::Physics::UCPhysicsObject, 137
- EnableGroupSockets
 - uConstruct::BaseBuildingGroup, 41
- EnableRenderings
 - uConstruct::BaseBuilding, 33
- EnableSocket
 - uConstruct::Sockets::BaseSocket, 51
- enabled
 - uConstruct::Core::Saving::UCSavingManager, 142
- fields
 - uConstruct::Core::Blueprints::Blueprint, 61
- fileFormat
 - uConstruct::Core::Saving::UCSavingManager, 142
- fileName
 - uConstruct::Core::Saving::UCSavingManager, 142
- finders

- uConstruct::Core::AOI::AOIManager, 21
- FitToLocalSpace
 - uConstruct::PreviewBuilding, 103
- FloatPercisionEquals
 - uConstruct::TransformExtensions, 122
- folderPath
 - uConstruct::Core::Saving::UCSavingManager, 143
- ForceEnable
 - uConstruct::Sockets::BaseSocket, 51
- GenerateTemplate
 - uConstruct::Core::Templates::TemplateUtility, 113
- GetBackwards
 - uConstruct::TransformExtensions, 122
- GetBatchedBuilding
 - uConstruct::BaseBuildingBatcher, 38
- GetBuildings
 - uConstruct::BaseBuildingGroup, 42
- GetDown
 - uConstruct::TransformExtensions, 122
- GetForward
 - uConstruct::TransformExtensions, 122
- GetGO
 - uConstruct::Core::PrefabDatabase::PrefabDB, 101
- GetInputs
 - uConstruct::Demo::BuildingPlacer, 76
- GetLeft
 - uConstruct::TransformExtensions, 124
- GetPath
 - uConstruct::Core::Blueprints::Blueprint, 60
- GetRendererCenter
 - uConstruct::TransformExtensions, 124
- GetRendererSize
 - uConstruct::TransformExtensions, 124
- GetRenderersSum
 - uConstruct::TransformExtensions, 124
- GetRight
 - uConstruct::TransformExtensions, 125
- GetTransform
 - uConstruct::Sockets::BaseSocket, 51
- GetUp
 - uConstruct::TransformExtensions, 125
- GetUpdatedBatchData
 - uConstruct::BaseBuildingGroup, 42
- globalCount
 - uConstruct::BaseBuilding, 37
- GloballyEnableSockets
 - uConstruct::Sockets::BaseSocket, 51
- GroupBuildingAdded
 - uConstruct::BaseBuilding, 33
 - uConstruct::Core::AOI::BuildingGroupAOITarget, 69
- GroupBuildingRemoved
 - uConstruct::BaseBuilding, 33
 - uConstruct::Core::AOI::BuildingGroupAOITarget, 69
- HandleAOI
 - uConstruct::Core::AOI::AOIManager, 20
- uConstruct::Core::AOI::BaseAOITarget, 24
- uConstruct::Core::AOI::BuildingGroupAOITarget, 70
- HandleFreePlace
 - uConstruct::BaseBuilding, 33
- HandleMaterial
 - uConstruct::BaseBuilding, 33
- HandleOccupiedSockets
 - uConstruct::BaseBuildingGroup, 42
- HandlePivot
 - uConstruct::Core::Blueprints::BlueprintField, 65
- HandlePlacing
 - uConstruct::BaseBuilding, 34
- HandlePlacingResults
 - uConstruct::Demo::BuildingPlacer, 76
- HandleRenders
 - uConstruct::BatchUtility, 58
- HandleSnapPlace
 - uConstruct::BaseBuilding, 34
- ignore
 - uConstruct::BaseBuilding, 37
 - uConstruct::Conditions::BaseCondition, 45
- inRange
 - uConstruct::Core::AOI::BaseAOITarget, 25
- InZone
 - uConstruct::Core::AOI::BaseAOITarget, 24
 - uConstruct::Core::AOI::BuildingGroupAOITarget, 70
- initialBuildingAction
 - uConstruct::Core::Saving::BuildingGroupSave←Data, 73
- InitiateBuildingData
 - uConstruct::BaseBuilding, 34
- InitiateComponents
 - uConstruct::Sockets::BaseSocket, 52
- Inspect
 - uConstruct::Demo::DemoUI, 87
- instance
 - uConstruct::Demo::DemoUI, 88
- IsBitSet
 - uConstruct::FlagsHelper, 89
- IsBitSet< T >
 - uConstruct::FlagsHelper, 90
- IsFit
 - uConstruct::Sockets::BaseSocket, 52
- IsGroupSocketOccupied
 - uConstruct::BaseBuildingGroup, 42
- isHoverTarget
 - uConstruct::Sockets::BaseSocket, 53
- isInsideMask
 - uConstruct::FlagsHelper, 90
- isRunning
 - uConstruct::Core::Threading::ThreadManager, 119
- isUpdate
 - uConstruct::Core::Threading::ThreadManager, 119
- isVertexOverLimit
 - uConstruct::BatchUtility, 58

- Load
 - uConstruct::Core::Saving::BaseUCSaveData, 54
 - uConstruct::Core::Saving::BuildingGroupSave↔Data, 72
 - uConstruct::Core::Saving::UCSavingManager, 141
- LoadExternalData
 - uConstruct::Core::Saving::UCSavingManager, 141
- LostSnapToSocket
 - uConstruct::BaseBuilding, 35
- maxPointOnGroup
 - uConstruct::Core::AOI::BuildingGroupAOITarget, 71
- name
 - uConstruct::Core::Blueprints::BlueprintField, 67
- OccupySocket
 - uConstruct::Sockets::BaseSocket, 52
- oldPos
 - uConstruct::Core::AOI::BaseAOIFinder, 23
- OnAfterDeserialize
 - uConstruct::Core::Blueprints::BlueprintField, 65
- OnApplicationQuit
 - uConstruct::Core::Manager::UCCallbacks↔Manager, 128
- OnBeforeSerialize
 - uConstruct::Core::Blueprints::BlueprintField, 65
- OnDisable
 - uConstruct::Core::AOI::BaseAOIFinder, 22
 - uConstruct::Core::AOI::BaseAOITarget, 25
 - uConstruct::Core::AOI::BuildingGroupAOITarget, 70
 - uConstruct::Core::Physics::UCPhysicsObject, 138
- OnDrawGizmos
 - uConstruct::Conditions::BaseCondition, 44
 - uConstruct::Conditions::CheckForBuilding↔Condition, 81
 - uConstruct::Conditions::CheckForGround↔Condition, 86
 - uConstruct::Conditions::TerrainModification↔Condition, 116
 - uConstruct::Core::AOI::BaseAOIFinder, 22
 - uConstruct::Core::AOI::BuildingGroupAOITarget, 70
 - uConstruct::Core::Physics::UCPhysicsObject, 138
 - uConstruct::Sockets::BaseSnapPoint, 48
- OnEnable
 - uConstruct::Core::AOI::BaseAOIFinder, 22
 - uConstruct::Core::AOI::BaseAOITarget, 25
 - uConstruct::Core::AOI::BuildingGroupAOITarget, 70
 - uConstruct::Core::Physics::UCPhysicsObject, 138
- PREFAB_PATH
 - uConstruct::Core::Templates::TemplateUtility, 113
- Pack
 - uConstruct::BaseBuilding, 35
 - uConstruct::Conditions::BaseCondition, 45
 - uConstruct::Conditions::CheckForBuilding↔Condition, 81
 - uConstruct::Conditions::CheckForCollision↔Condition, 83
 - uConstruct::Conditions::CheckForGround↔Condition, 86
 - uConstruct::Conditions::TerrainModification↔Condition, 117
 - uConstruct::Core::Blueprints::BlueprintField, 66
 - uConstruct::Core::Blueprints::IBlueprintItem, 91
 - uConstruct::Sockets::BaseSocket, 52
- PlaceBuilding
 - uConstruct::BaseBuilding, 35
 - uConstruct::Demo::BuildingPlacer, 76
- PlacingRestrictionType
 - uConstruct, 13
- PopulateBatchedFilters
 - uConstruct::BaseBuildingGroup, 42
- previewPrefab
 - uConstruct::PreviewBuilding, 103
- priority
 - uConstruct::BaseBuilding, 37
 - uConstruct::Conditions::BaseCondition, 45
 - uConstruct::Core::Blueprints::IBlueprintItem, 92
 - uConstruct::Sockets::BaseSocket, 53
- ProjectName
 - uConstruct::Core::Manager::UCCallbacks↔Manager, 128
- ProjectPath
 - uConstruct::Core::Manager::UCCallbacks↔Manager, 128
- RESOURCES_PATH
 - uConstruct::Core::Templates::TemplateUtility, 113
- radius
 - uConstruct::Core::AOI::BaseAOIFinder, 23
- radiusAdjuster
 - uConstruct::Core::AOI::BuildingGroupAOITarget, 71
- Raycast
 - uConstruct::Core::Physics::UCPhysics, 132, 133
 - uConstruct::Core::Physics::UCPhysicsObject, 138
- RaycastAll
 - uConstruct::Core::Physics::UCPhysics, 134
- Remove
 - uConstruct::BatchData, 56
- RemoveBuilding
 - uConstruct::BaseBuildingGroup, 43
- RemoveField
 - uConstruct::Core::Blueprints::Blueprint, 60
- RemoveFinder
 - uConstruct::Core::AOI::AOIManager, 20
- RemoveFromDB
 - uConstruct::Core::PrefabDatabase::PrefabDB, 101
- RemovePhysicsObject
 - uConstruct::Core::Physics::UCPhysics, 135
- RemoveTarget
 - uConstruct::Core::AOI::AOIManager, 21
- RemoveTemplate

- uConstruct::BaseBuilding, 35
- RenderEditor
 - uConstruct::Sockets::BaseSocket, 53
- ResetBuildingInstance
 - uConstruct::Demo::BuildingPlacer, 76
- ResetControl
 - uConstruct::Demo::DemoUI, 88
- ResetDB
 - uConstruct::Core::PrefabDatabase::PrefabDB, 101
- ResetMaterialColors
 - uConstruct::BaseBuilding, 35
- RestoreTerrainModifications
 - uConstruct::Conditions::TerrainModification↔Condition, 117
- ReturnAlphaKey
 - uConstruct::Demo::BuildingPlacer, 76
- ReturnBatchedBuilding
 - uConstruct::BaseBuildingGroup, 43
- ReturnClosest
 - uConstruct::Sockets::BaseSnapPoint, 48
- ReturnDistance
 - uConstruct::Sockets::BaseSnapPoint, 48
- ReturnParent
 - uConstruct::BaseBuilding, 35
- ReturnPosition
 - uConstruct::BaseBuilding, 36
- ReturnSaveData
 - uConstruct::Core::Saving::UCSavingManager, 141
- ReturnSocket
 - uConstruct::BaseBuilding, 36
- ReturnUID
 - uConstruct::Core::PrefabDatabase::PrefabDB, 102
- RevertModifications
 - uConstruct::Conditions::TerrainModification↔Condition, 117
- rootBuilding
 - uConstruct::Conditions::BaseCondition, 45
- RunOnUConstructThread
 - uConstruct::Core::Threading::ThreadManager, 118
- RunOnUnityThread
 - uConstruct::Core::Threading::ThreadManager, 119
- RunThread
 - uConstruct::Core::Threading::ThreadManager, 119
- Save
 - uConstruct::BaseBuildingGroup, 43
 - uConstruct::Core::Blueprints::Blueprint, 61
 - uConstruct::Core::Saving::UCSavedItem, 139
 - uConstruct::Core::Saving::UCSavingManager, 141
- selfPath
 - uConstruct::Core::Blueprints::Blueprint, 61
- Serialize
 - uConstruct::Core::Saving::UCSavingManager, 142
- Snap
 - uConstruct::Sockets::BaseSnapPoint, 48
- SnappedToSocket
 - uConstruct::BaseBuilding, 36
- SocketLayers
 - uConstruct::LayersData, 96
- SocketMask
 - uConstruct::LayersData, 96
- Start
 - uConstruct::Core::Manager::UCCallbacks↔Manager, 128
 - uConstruct::Core::Physics::UCPhysicsObject, 138
 - uConstruct::Demo::BuildingPlacer, 77
- StartThread
 - uConstruct::Core::Threading::ThreadManager, 119
- StopThread
 - uConstruct::Core::Threading::ThreadManager, 119
- Subside
 - uConstruct::TransformExtensions, 125
- target
 - uConstruct::Core::Blueprints::BlueprintField, 66
- targets
 - uConstruct::Core::AOI::AOIManager, 21
- thread
 - uConstruct::Core::Threading::ThreadManager, 120
- totalVectors
 - uConstruct::Core::AOI::BuildingGroupAOITarget, 71
- type
 - uConstruct::Core::Blueprints::BlueprintField, 66
- uConstruct, 11
 - PlacingRestrictionType, 13
- uConstruct.BaseBuilding, 25
- uConstruct.BaseBuildingBatcher, 37
- uConstruct.BaseBuildingGroup, 38
- uConstruct.BatchClass, 54
- uConstruct.BatchData, 55
- uConstruct.BatchExtensions, 57
- uConstruct.BatchUtility, 57
- uConstruct.BuildingBlueprintData, 67
- uConstruct.BuildingEditor, 67
- uConstruct.BuildingMaterialData, 73
- uConstruct.CodeGenerator, 13
- uConstruct.CodeGenerator.BuildingTypesCode↔Generator, 78
- uConstruct.CodeGenerator.BuildingTypesCode↔GeneratorEditor, 79
- uConstruct.Conditions, 13
- uConstruct.Conditions.BaseCondition, 43
- uConstruct.Conditions.CheckForBuilding_Blueprint↔Data, 79
- uConstruct.Conditions.CheckForBuildingCondition, 80
- uConstruct.Conditions.CheckForCollision_Blueprint↔Data, 82
- uConstruct.Conditions.CheckForCollisionCondition, 82
- uConstruct.Conditions.CheckForGround_BlueprintData, 84
- uConstruct.Conditions.CheckForGroundCondition, 84
- uConstruct.Conditions.HeightsData, 90
- uConstruct.Conditions.TerrainModification_Blueprint↔Data, 114
- uConstruct.Conditions.TerrainModificationCondition, 114

- uConstruct.Conditions.TerrainModificationCondition.↔
RestoreData, 105
- uConstruct.Conditions.TerrainModificationData, 117
- uConstruct.Core, 14
- uConstruct.Core.AOI.AOIManager, 19
- uConstruct.Core.AOI.BaseAOIFinder, 22
- uConstruct.Core.AOI.BaseAOITarget, 23
- uConstruct.Core.AOI.BuildingGroupAOITarget, 68
- uConstruct.Core.AOI, 14
- uConstruct.Core.Blueprints, 14
- uConstruct.Core.Blueprints.Blueprint, 59
- uConstruct.Core.Blueprints.BlueprintData, 61
- uConstruct.Core.Blueprints.BlueprintField, 63
- uConstruct.Core.Blueprints.IBlueprintItem, 91
- uConstruct.Core.Manager, 14
- uConstruct.Core.Manager.UCCallbacksManager, 126
- uConstruct.Core.Physics, 15
- uConstruct.Core.Physics.IUTCPPhysicsIgnored, 94
- uConstruct.Core.Physics.UCPhysics, 131
- uConstruct.Core.Physics.UCPhysicsHit, 135
- uConstruct.Core.Physics.UCPhysicsHitsArray, 136
- uConstruct.Core.Physics.UCPhysicsObject, 136
- uConstruct.Core.PrefabDatabase, 15
- uConstruct.Core.PrefabDatabase.PrefabDBCust↔
Editor, 102
- uConstruct.Core.PrefabDatabase.PrefabData, 98
- uConstruct.Core.PrefabDatabase.PrefabDatabase↔
Editor, 98
- uConstruct.Core.PrefabDatabase.PrefabDB, 99
- uConstruct.Core.Saving, 15
- uConstruct.Core.Saving.BaseUCSaveData, 53
- uConstruct.Core.Saving.BuildingGroupSaveData, 71
- uConstruct.Core.Saving.BuildingSaveData, 77
- uConstruct.Core.Saving.SaveDrawer, 105
- uConstruct.Core.Saving.SerializeableQuaternion, 106
- uConstruct.Core.Saving.SerializeableVector3, 107
- uConstruct.Core.Saving.UCSavedItem, 139
- uConstruct.Core.Saving.UCSavingManager, 139
- uConstruct.Core.Templates, 16
- uConstruct.Core.Templates.ITemplateObject, 93
- uConstruct.Core.Templates.Template, 110
- uConstruct.Core.Templates.TemplateCreatorEditor, 110
- uConstruct.Core.Templates.TemplateCreationData, 111
- uConstruct.Core.Templates.TemplateMenuEditor, 111
- uConstruct.Core.Templates.TemplateObjectSelection,
112
- uConstruct.Core.Templates.TemplateSelectionWindow,
112
- uConstruct.Core.Templates.TemplateUtility, 112
- uConstruct.Core.Threading, 16
- uConstruct.Core.Threading.IThreadTask, 93
- uConstruct.Core.Threading.ThreadManager, 118
- uConstruct.Core.Threading.ThreadTask, 120, 121
- uConstruct.Demo, 16
- uConstruct.Demo.BuildingPlacer, 73
- uConstruct.Demo.DemoUI, 86
- uConstruct.Demo.uConstruct_FirstPersonController,
129
- uConstruct.Demo.uConstruct_MouseLook, 130
- uConstruct.Extensions, 17
- uConstruct.Extensions.ExtensionsEditor, 88
- uConstruct.Extensions.PCloudExtension, 17
- uConstruct.Extensions.PCloudExtension.Player↔
Instantiator, 98
- uConstruct.FlagsHelper, 89
- uConstruct.IBuilding, 92
- uConstruct.IPlacingModifier, 92
- uConstruct.LayersData, 94
- uConstruct.LayersEditor, 96
- uConstruct.PhysicsObjectEditor, 97
- uConstruct.PreviewBuilding, 103
- uConstruct.PreviewBuildingEditor, 104
- uConstruct.PropertyCreatorEditor, 104
- uConstruct.SavingEditor, 106
- uConstruct.SocketEditor, 108
- uConstruct.Sockets, 17
- uConstruct.Sockets.BaseSnapPoint, 46
- uConstruct.Sockets.BaseSocket, 49
- uConstruct.Sockets.SOverlapThreshold, 109
- uConstruct.Sockets.SnapPointEditor, 107
- uConstruct.Sockets.SocketBuildingData, 108
- uConstruct.TransformExtensions, 121
- uConstruct.UC_EditorUtility, 126
- uConstruct.UCSettingAttribute, 143
- uConstruct.UCSettingCategory, 144
- uConstruct.UCSettings, 144
- uConstruct.UCSettingsEditor, 145
- uConstruct.UConstructManager, 130
- uConstruct::BaseBuilding
_uid, 37
- ActivateColliders, 29
- ActivateConditions, 29
- ActivateSnapPoints, 30
- ActivateSockets, 30
- AddCondition, 30
- AddSocket, 30
- AddTemplate, 30
- AssignOriginalColors, 30
- Awake, 31
- BuildingDeattached, 31
- BuildingDestroyed, 31
- BuildingGroupChanged, 31
- BuildingPlaced, 31
- CheckConditions, 31
- CreateCondition, 31
- CreateSnapPoint, 32
- CreateSocket, 32
- DeAttachBuilding, 32
- DestroyBuilding, 32
- EnableRenderings, 33
- globalCount, 37
- GroupBuildingAdded, 33
- GroupBuildingRemoved, 33
- HandleFreePlace, 33
- HandleMaterial, 33
- HandlePlacing, 34

- HandleSnapPlace, 34
- ignore, 37
- InitiateBuildingData, 34
- LostSnapToSocket, 35
- Pack, 35
- PlaceBuilding, 35
- priority, 37
- RemoveTemplate, 35
- ResetMaterialColors, 35
- ReturnParent, 35
- ReturnPosition, 36
- ReturnSocket, 36
- SnappedToSocket, 36
- uConstruct::BaseBuildingBatcher
 - DestroyBuilding, 38
 - GetBatchedBuilding, 38
- uConstruct::BaseBuildingGroup
 - AOIGroup, 40
 - AddBuilding, 40
 - Awake, 41
 - Batch, 41
 - buildings, 43
 - CreateGroup< T >, 41
 - DestroyGroup, 41
 - EnableGroupSockets, 41
 - GetBuildings, 42
 - GetUpdatedBatchData, 42
 - HandleOccupiedSockets, 42
 - IsGroupSocketOccouped, 42
 - PopulateBatchedFilters, 42
 - RemoveBuilding, 43
 - ReturnBatchedBuilding, 43
 - Save, 43
- uConstruct::BatchClass
 - Contains, 55
- uConstruct::BatchData
 - Add, 56
 - Batchable, 56
 - Remove, 56
- uConstruct::BatchUtility
 - CompileInitialBatchData, 57
 - HandleRenders, 58
 - isVertexOverLimit, 58
 - UpdateBatchData, 58
- uConstruct::Conditions::BaseCondition
 - Awake, 44
 - CheckCondition, 44
 - DisableOnPlace, 45
 - ignore, 45
 - OnDrawGizmos, 44
 - Pack, 45
 - priority, 45
 - rootBuilding, 45
- uConstruct::Conditions::CheckForBuildingCondition
 - CheckCondition, 81
 - OnDrawGizmos, 81
 - Pack, 81
- uConstruct::Conditions::CheckForCollisionCondition
 - Awake, 83
 - CheckCondition, 83
 - Pack, 83
- uConstruct::Conditions::CheckForGroundCondition
 - Awake, 85
 - CheckCondition, 85
 - OnDrawGizmos, 86
 - Pack, 86
- uConstruct::Conditions::TerrainModificationCondition
 - AddModificationData, 116
 - Awake, 116
 - CheckCondition, 116
 - OnDrawGizmos, 116
 - Pack, 117
 - RestoreTerrainModifications, 117
 - RevertModifications, 117
- uConstruct::Core::AOI::AOIManager
 - AddFinder, 20
 - AddTarget, 20
 - ComputeAOI, 20
 - finders, 21
 - HandleAOI, 20
 - RemoveFinder, 20
 - RemoveTarget, 21
 - targets, 21
 - UpdateAOI, 21
 - UpdatePositions, 21
- uConstruct::Core::AOI::BaseAOIFinder
 - aoiPosition, 23
 - oldPos, 23
 - OnDisable, 22
 - OnDrawGizmos, 22
 - OnEnable, 22
 - radius, 23
 - Update, 23
 - UpdateAOI, 23
- uConstruct::Core::AOI::BaseAOITarget
 - aoiPosition, 25
 - HandleAOI, 24
 - inRange, 25
 - InZone, 24
 - OnDisable, 25
 - OnEnable, 25
 - useMultiThreadZoneSearch, 25
- uConstruct::Core::AOI::BuildingGroupAOITarget
 - buildingGroup, 71
 - correctPosition, 71
 - GroupBuildingAdded, 69
 - GroupBuildingRemoved, 69
 - HandleAOI, 70
 - InZone, 70
 - maxPointOnGroup, 71
 - OnDisable, 70
 - OnDrawGizmos, 70
 - OnEnable, 70
 - radiusAdjuster, 71
 - totalVectors, 71
- uConstruct::Core::Blueprints::Blueprint

- AddField, 60
- BLUEPRINT_ASSET_FIRST, 61
- BLUEPRINT_ASSET_PATH, 61
- blueprintName, 61
- CreateBlueprint, 60
- Delete, 60
- fields, 61
- GetPath, 60
- RemoveField, 60
- Save, 61
- selfPath, 61
- uConstruct::Core::Blueprints::BlueprintField
 - BlueprintField, 64, 65
 - Contains, 65
 - data, 66
 - dataBytes, 66
 - HandlePivot, 65
 - name, 67
 - OnAfterDeserialize, 65
 - OnBeforeSerialize, 65
 - Pack, 66
 - target, 66
 - type, 66
 - UnPack, 66
- uConstruct::Core::Blueprints::IBlueprintItem
 - Pack, 91
 - priority, 92
- uConstruct::Core::Manager::UCCallbacksManager
 - AddApplicationQuitAction, 127
 - Awake, 128
 - CreateAndInitialize, 128
 - OnApplicationQuit, 128
 - ProjectName, 128
 - ProjectPath, 128
 - Start, 128
 - Update, 128
- uConstruct::Core::Physics::UCPhysics
 - AddPhysicsObject, 132
 - Raycast, 132, 133
 - RaycastAll, 134
 - RemovePhysicsObject, 135
- uConstruct::Core::Physics::UCPhysicsHit
 - Convert, 135
- uConstruct::Core::Physics::UCPhysicsObject
 - DrawShape, 137
 - OnDisable, 138
 - OnDrawGizmos, 138
 - OnEnable, 138
 - Raycast, 138
 - Start, 138
 - UpdateBounds, 138
- uConstruct::Core::PrefabDatabase::PrefabDatabase↔
 - Editor
 - UpdateDB, 99
- uConstruct::Core::PrefabDatabase::PrefabDB
 - AddToDB, 100
 - Contains, 100
 - GetGO, 101
 - RemoveFromDB, 101
 - ResetDB, 101
 - ReturnUID, 102
- uConstruct::Core::Saving::BaseUCSaveData
 - Load, 54
- uConstruct::Core::Saving::BuildingGroupSaveData
 - initialBuildingAction, 73
 - Load, 72
- uConstruct::Core::Saving::UCSavedItem
 - Save, 139
- uConstruct::Core::Saving::UCSavingManager
 - dataPath, 142
 - DeserializeStream, 141
 - DestoryDataOnCurrentScene, 141
 - enabled, 142
 - fileFormat, 142
 - fileName, 142
 - folderPath, 143
 - Load, 141
 - LoadExternalData, 141
 - ReturnSaveData, 141
 - Save, 141
 - Serialize, 142
- uConstruct::Core::Templates::TemplateUtility
 - GenerateTemplate, 113
 - PREFAB_PATH, 113
 - RESOURCES_PATH, 113
- uConstruct::Core::Threading::ThreadManager
 - _enabled, 119
 - isRunning, 119
 - isUpdate, 119
 - RunOnUConstructThread, 118
 - RunOnUnityThread, 119
 - RunThread, 119
 - StartThread, 119
 - StopThread, 119
 - thread, 120
 - UConstructThreadActions, 120
 - UnityThread, 119
 - UnityThreadQueuedActions, 120
 - UpdateThread, 119
- uConstruct::Demo::BuildingPlacer
 - _defaultLockCursor, 77
 - _destroyBuildings, 77
 - _rotatedWithPlayer, 77
 - _rotationValue, 77
 - ApplyControlsToDemoUI, 75
 - Awake, 75
 - CreateBuildingInstance, 75
 - DestroyBuilding, 75
 - DestroyCurrentBuilding, 76
 - GetInputs, 76
 - HandlePlacingResults, 76
 - PlaceBuilding, 76
 - ResetBuildingInstance, 76
 - ReturnAlphaKey, 76
 - Start, 77
 - Update, 77

- uConstruct::Demo::DemoUI
 - AddControl, [87](#)
 - Awake, [87](#)
 - controls, [88](#)
 - controlsCount, [88](#)
 - Inspect, [87](#)
 - instance, [88](#)
 - ResetControl, [88](#)
- uConstruct::FlagsHelper
 - IsBitSet, [89](#)
 - IsBitSet< T >, [90](#)
 - isInsideMask, [90](#)
- uConstruct::IBuilding
 - DestroyBuilding, [92](#)
- uConstruct::LayersData
 - BuildingLayers, [95](#)
 - BuildingMask, [95](#)
 - DefaultBuildingLayer, [95](#)
 - DefaultBuildingLayerString, [96](#)
 - DefaultSocketLayer, [96](#)
 - DefaultSocketLayerString, [96](#)
 - SocketLayers, [96](#)
 - SocketMask, [96](#)
- uConstruct::PreviewBuilding
 - ApplyChangesToPrefab, [103](#)
 - FitToLocalSpace, [103](#)
 - previewPrefab, [103](#)
- uConstruct::Sockets::BaseSnapPoint
 - AnchoredPosition, [47](#)
 - Awake, [48](#)
 - OnDrawGizmos, [48](#)
 - ReturnClosest, [48](#)
 - ReturnDistance, [48](#)
 - Snap, [48](#)
- uConstruct::Sockets::BaseSocket
 - Awake, [51](#)
 - BuildingSnapped, [51](#)
 - Create, [51](#)
 - drawIndividual, [53](#)
 - EnableSocket, [51](#)
 - ForceEnable, [51](#)
 - GetTransform, [51](#)
 - GloballyEnableSockets, [51](#)
 - InitiateComponents, [52](#)
 - IsFit, [52](#)
 - isHoverTarget, [53](#)
 - OccupySocket, [52](#)
 - Pack, [52](#)
 - priority, [53](#)
 - RenderEditor, [53](#)
 - Update, [53](#)
- uConstruct::TransformExtensions
 - FloatPercisionEquals, [122](#)
 - GetBackwards, [122](#)
 - GetDown, [122](#)
 - GetForward, [122](#)
 - GetLeft, [124](#)
 - GetRendererCenter, [124](#)
 - GetRendererSize, [124](#)
 - GetRenderersSum, [124](#)
 - GetRight, [125](#)
 - GetUp, [125](#)
 - Subside, [125](#)
- uConstruct::UC_EditorUtility
 - DisplayScriptField, [126](#)
- UConstructThreadActions
 - uConstruct::Core::Threading::ThreadManager, [120](#)
- UnPack
 - uConstruct::Core::Blueprints::BlueprintField, [66](#)
- UnityThread
 - uConstruct::Core::Threading::ThreadManager, [119](#)
- UnityThreadQueuedActions
 - uConstruct::Core::Threading::ThreadManager, [120](#)
- Update
 - uConstruct::Core::AOI::BaseAOIFinder, [23](#)
 - uConstruct::Core::Manager::UCCallbacks↔Manager, [128](#)
 - uConstruct::Demo::BuildingPlacer, [77](#)
 - uConstruct::Sockets::BaseSocket, [53](#)
- UpdateAOI
 - uConstruct::Core::AOI::AOIManager, [21](#)
 - uConstruct::Core::AOI::BaseAOIFinder, [23](#)
- UpdateBatchData
 - uConstruct::BatchUtility, [58](#)
- UpdateBounds
 - uConstruct::Core::Physics::UCPhysicsObject, [138](#)
- UpdateDB
 - uConstruct::Core::PrefabDatabase::Prefab↔DatabaseEditor, [99](#)
- UpdatePositions
 - uConstruct::Core::AOI::AOIManager, [21](#)
- UpdateThread
 - uConstruct::Core::Threading::ThreadManager, [119](#)
- useMultiThreadZoneSearch
 - uConstruct::Core::AOI::BaseAOITarget, [25](#)