proTile Map Editor

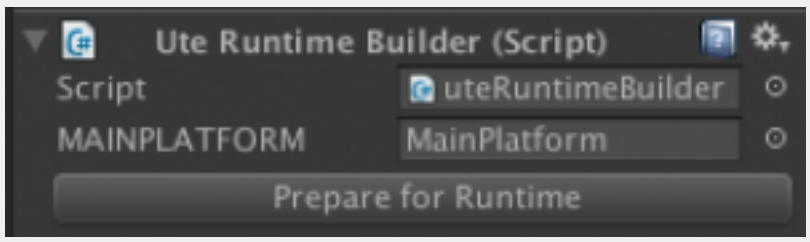
Unity Editor Extension

Runtime Scripting

Introduction

To start using Runtime Support create a GameObject and add **uteRuntimeBuilder** Component. Assign **MAINPLATFORM** (a starting point for building) if needed.

Click on “Prepare for Runtime” to generate all the necessary data from Tile-Editor for Runtime usage.



Class Tile

```
public string guidid; // Get object GUID ID
public GameObject ref_obj; // Get GameObject reference
public GameObject mainObject; // Get GameObject that you build and set to
RuntimeBuilder
public Texture2D preview; // Get Tile preview texture that is generated auto or
set in Tile-Editor
public string name; // Get Tile name
public string title; // Get Tile title
```

Class Category

```
public List<Tile> allTiles = new List<Tile>(); // Get the List of all Tile
objects in the Category
public string name; // Get the Category name
public string type; // Get the category type (static/dynamic)
```

Class uteRuntimeBuilder

```
runtimeBuilder = this.gameObject.GetComponent<uteRuntimeBuilder>(); // Assign
component
```

Functions

All functions are called from uteRuntimeBuilder Component.

Runtime Editor settings

- **void SetRaycastDistance(float distance)** – Set the maximum distance of Raycast when building.
- **void SetSnapOption(string option)** – Set snapping option (“auto”, “fixed” or “none”).
- **string GetSnapOption()** - Get current snapping option.
- **void DisableMouseInputForBuild()** - Disable mouse input (use if you want to build using keyboard or procedurally).
- **void EnableMouseInputForBuild()** – Enable mouse input.
- **void SetFixedRaycastPosition(Vector2 position)** – Set fixed Raycast position from screen point.
- **void DisableFixedRaycastPosition()** – Unset Fixed Raycast position to use Mouse Position.
- **Vector2 GetFixedRaycastPosition()** – Get fixed Raycast position.

Optimization

- **void Batch(bool AddMeshColliders, bool RemoveLeftOvers)** – Batching and Optimization for Runtime. Ability to generate colliders and clean up references.
- **void UnBatch()** – Remove Runtime batching and optimization (use only if you use RemoveLeftOvers=false in Batch). Good when want to continue building.
- **List<GameObject> GetBatchedObjects()** – Get the list of the Batched objects.

Building and Handling Runtime Editor

- **void EnableToBuild()** - Enable RuntimeEngine building.
- **void DisableToBuild()** - Disable RuntimeEngine building.
- **void SetBuildMode(BuildMode)** – Set BuildMode (BuildMode.Normal, BuildMode.Continuous, BuildMode.Mass).
- **BuildMode GetCurrentBuildMode()** - Get current BuildMode.
- **void SetCurrentTile(GameObject go)** – Set tile for building (usually you get it from Tile.mainObject).
- **void CancelCurrentTile()** - Cancel current tile for building.
- **void RotateCurrentTileRight()** – Rotate current tile Right.
- **void RotateCurrentTileLeft()** - Rotate current tile Left.
- **void RotateCurrentTileUp()** – Rotate current tile Up.
- **void RotateCurrentTileDown()** – Rotate current tile Down.
- **void RotateCurrentTileFlip()** – Flip current tile.
- **GameObject GetCurrentSelectedObject()** – Get current object that mouse is hovering.
- **GameObject GetCurrentTile()** - Get current tile that is used for building.
- **bool DestroyCurrentSelectedObject()** - Destroy tile that is hovered by mouse.
- **void MassBuildHeightUp()** - Increase height for Mass Build Mode by 1.
- **void MassBuildHeightDown()** – Reduce height for Mass Build Mode by 1.
- **void MassBuildResetHeight()** – Reset height for Mass Build Mode to 1.
- **void MassBuildCancel()** – Cancel Mass Build (which is in progress when dragging).

Create Tiles at Runtime

- **void AddTileInCategory(GameObject obj, string categoryName, string tileUniqueName, string tileTitle, Texture2D tilePreview, bool isStatic)** – Create new Tile at Runtime.
- **void RemoveTileFromCategory(string categoryName, string tileUniqueName)** – Remove Tile which is created at Runtime only.

Get Data from Tile-Editor

- **List<string> GetListOfCategoryNames()** – Get all the categories from Tile-Editor.
- **void SetCurrentTileInstantly(GameObject go)** – Use this when instantiating Tiles procedurally.
- **Category GetCategoryByCategoryName(string catName)** - Get Category by its name.
- **Tile GetTileFromCategoryByName(string categoryName, string tileName)** - Get Tile from Category

by giving category and tile name.
- **List<Tile> GetTileListByCategoryName(string catName)** – Get list of Tiles from Category.
- **Tile GetTileByID(int id)** – Get Tile by ID.

Procedural

- **void PlaceCurrentTileAtPosition(Vector3 position, Vector3 rotation)** – Place tile at position with rotation (for procedural generation).
- **void PlaceCurrentTileAtPosition(Vector3 position)** – Place tile at position (for procedural generation).

Save / Load Map

- **void SaveMap(string mapName)** - Save map.
- **void LoadMap(string mapName, bool loadAdditive = false)** – Load Map.
- **string[] GetMapNamesList()** – Get saved map list.
- **bool CheckIfMapExists(string mapName)** - Check if map exists.
- **void DeleteMap(string mapName)** – Delete map with given name.

* For more information and explanation see all the examples that are included in the package.