Hey there!

Thank you for your interest in **Maintainer**, your attentive Unity projects housekeeping assistant!

v. 2.1.0
[version history]

This is an editor extension made of three separate modules:
Issues Finder, **Project Cleaner**, **References Finder**.

See more details about each module at the *Modules* section below.
Or watch a quick intro to know some essentials right away!

Contents

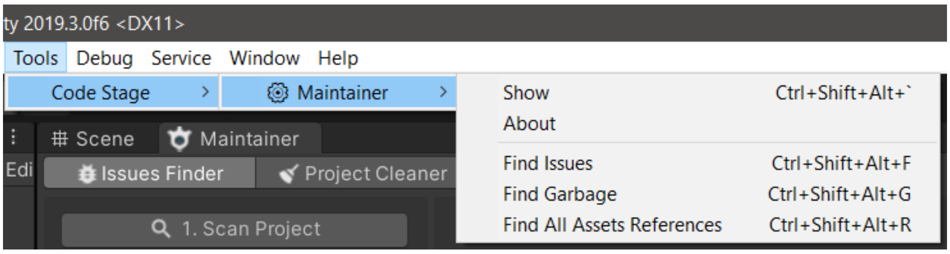
Installation and setup

IMPORTANT:
To avoid different compatibility issues and errors:

- Always close Maintainer window before updating.
- Always remove previous version from project Assets before updating.

Menus

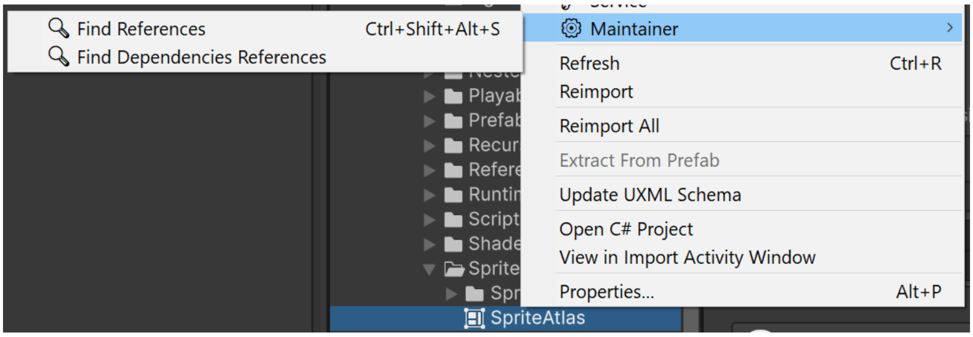
As you import plugin, you will see new menu commands at the top menu:



Tools > Code Stage > Maintainer

- **Show**
  Opens Maintainer window at last opened tab or at the first tab if was not opened before.
- **About**
  Opens Maintainer window at the About tab with helpful links and additional information about extension.
- **Find Issues**
  Starts *Issues Finder* module search and shows Maintainer window with results.
- **Find Garbage**
  Starts *Project Cleaner* module search and shows Maintainer window with results.
- **Find All Assets References**
  Starts *References Finder* module global assets references search and shows Maintainer window with results.

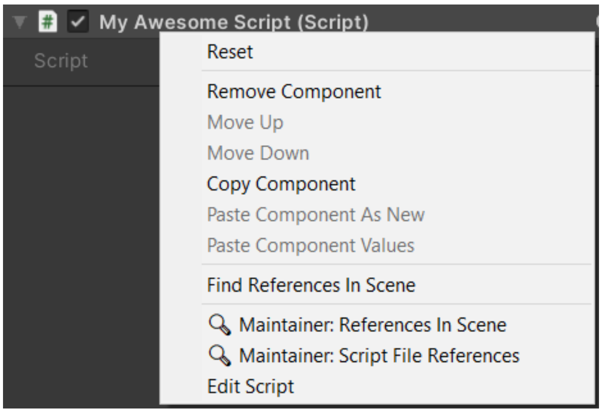
In addition, these menu items appear:



Project View Context Menu (Assets menu) > Maintainer

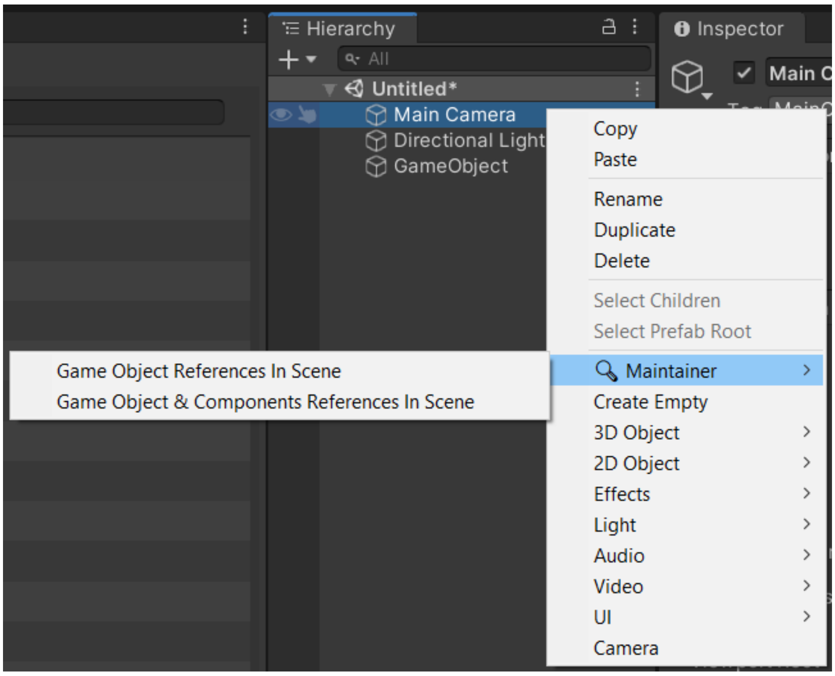
- **Find References**
  Starts *References Finder* module search in Assets Scope mode for the selected item(s) and shows Maintainer window with results.

  When called for folder, it will check all files in that folder recursively.
  Works both with in-project assets (placed in Assets folder) and with package (visible at the Packages Project Browser section) assets.
- **Find Dependencies References**
  Calls **Find References** command described above for selected item(s) direct dependencies and shows Maintainer window with results.



Component Context Menu

- **References In Scene**
  Starts *References Finder* module search in Hierarchy Search Scope mode for the component which context menu was called for. Searches references in current Hierarchy for any Game Object components in Scenes or Prefabs.
- **Script File References**
  Starts *References Finder* module search in Project Assets Search Scope mode for the script file asset which context menu was called for.
  Since it looks for the script file references, it will be active only for the components deriving from the MonoBehaviour or ScriptableObject and having real script asset file in the project (either in Assets or Packages).



Hierarchy Context Menu (Game Object Menu) > Maintainer

- **Game Object References In Scene**
  Starts *References Finder* module search in Hierarchy Search Scope mode for the Game Object which context menu was called for. Searches references in current Hierarchy for any Game Object in Scenes or Prefabs.
- **Game Object & Components References In Scene**
  Starts *References Finder* module search in Hierarchy Search Scope mode for the Game Object which context menu was called for and for all its components as well. Searches references in current Hierarchy for any Game Object and all its components in Scenes or Prefabs.

Files in your project

Here are the files Maintainer may produce in your project during its activity:

- **YourProject/ProjectSettings/MaintainerSettings.asset**
  This is a Maintainer Settings for your project. File is a standard Unity serialized asset (so it may be binary or text depending on your assets serialization settings of the project).
- **YourProject/[Library or UserSettings in 2020.1+]/MaintainerPersonalSettings.dat**

This is a Personal Settings file for your project - such as last open tab, scroll position, sorting orientation, etc. File is a standard Unity serialized asset. Please ignore it in your VCS to reduce chance of merge conflicts (it's a good practice to ignore whole Library \ UserSettings folder actually).

- **YourProject/Library/MaintainerMap.dat**
  This is a Maintainer-optimized Assets Map of your project used by all modules. It's a binary file.
- **YourProject/Temp/Maintainer*Results.bin and MaintainerReferencesLastSearched.bin**
  These are Maintainer search results files, which contain last search results of different modules. The Temp folder is deleted by Unity when it exits, so results are stored only for one session and will be erased on Unity restart. These are binary or json-compatible files, depending on module.

Modules

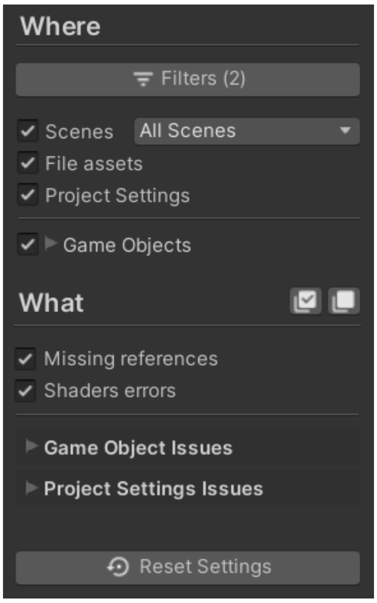
Issues Finder

This module allows finding different issues within your Unity project, like missing scripts, missing references, and much more, including any custom issues of your own (see below). It also able to fix some of these issues in one-by-one or batch mode to let you quickly get rid of them.

You can use this module both from Maintainer window and via menu command / shortcut (see *Installation and setup*). If you will use it via menu command / shortcut, it will perform search with latest (or default) settings and will show results in the Maintainer window after that.

Issues Finder Settings

If you wish to tune settings before search, open Maintainer window, at the **Issues Finder** tab and tune them from there:



Where

☑ Scenes All Scenes
☑ File assets
☑ Project Settings
☑ Game Objects

What

☑ Missing references
☑ Shaders errors

Settings are made of two separate groups:

- **Where** section with different filtering and ignoring settings allowing to specify where do you wish to perform the issues lookup.
- **What** section with settings allowing to specify what specific issues you would like to look for.

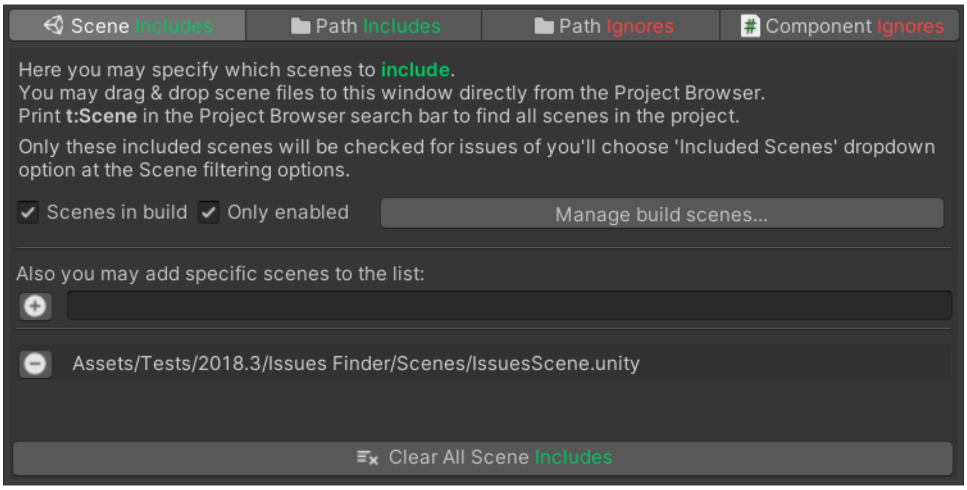
Let's describe them one-by-one, in their exact order.

Filters… button opens **Precise Filters** window with four tabs: **Scene Includes**, **Path Includes**, **Path Ignores** and **Component Ignores**. See detailed description of each tab below.

Number in brackets indicate active filters amount.

Precise Filters tabs

IMPORTANT:
Please keep an eye on your lists of filters to make sure they do not have old and not actual items to avoid extra processing resources waste during the Issues Search.



Here you may specify which scenes to **include**.
You may drag & drop scene files to this window directly from the Project Browser.
Print **t:Scene** in the Project Browser search bar to find all scenes in the project.

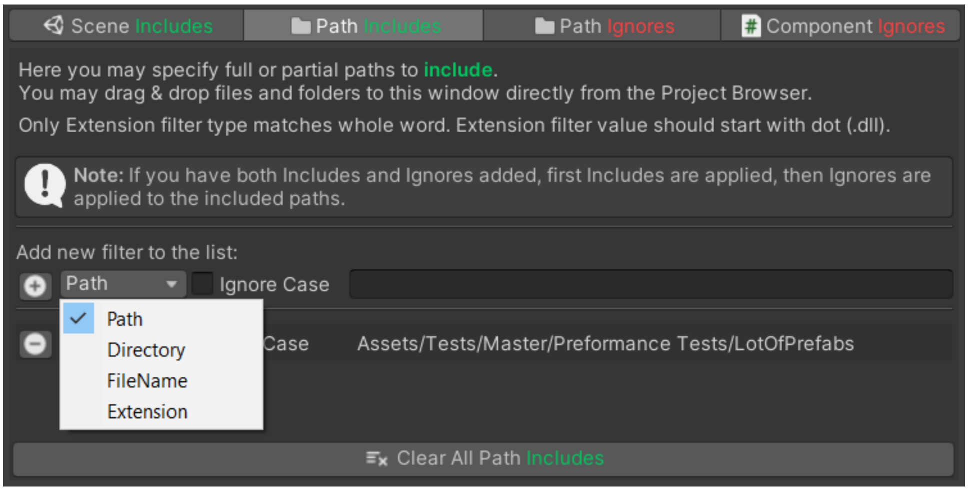
Only these included scenes will be checked for issues of you'll choose 'Included Scenes' dropdown option at the Scene filtering options.

☑ Scenes in build ☑ Only enabled

Also you may add specific scenes to the list:

Assets/Tests/2018.3/Issues Finder/Scenes/IssuesScene.unity

Scene Includes tab allows including specific scenes into the issues search. Here you may easily include all scenes added to the Build Settings and specify if you also wish to include disabled items or not. In addition, you may add specific scenes by dragging them from the Project Browser or by entering their path relative to the project folder.
Use "Included Scenes" option for the Scenes setting to search issues only in these included scenes.



Here you may specify full or partial paths to **include**.
You may drag & drop files and folders to this window directly from the Project Browser.
Only Extension filter type matches whole word. Extension filter value should start with dot (.dll).

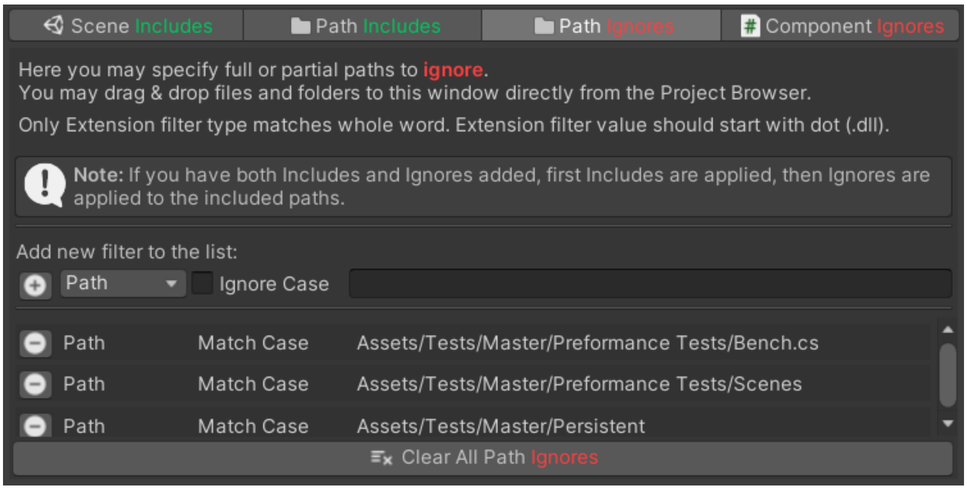
Note: If you have both Includes and Ignores added, first Includes are applied, then Ignores are applied to the included paths.

Add new filter to the list:

Path □ Ignore Case

Case Assets/Tests/Master/Preformance Tests/LotOfPrefabs

Path Includes tab allows including specified files, folders, extensions or partial paths to the Issues Search.
You may add filters manually or just drag & drop items from the Project Browser.
You may include files with specified extension, full or partial paths and names, both with Match or Ignore Case.
If you have both Includes and Ignores added, first Includes are applied, then Ignores are applied to the included items.



Here you may specify full or partial paths to **ignore**.
You may drag & drop files and folders to this window directly from the Project Browser.
Only Extension filter type matches whole word. Extension filter value should start with dot (.dll).

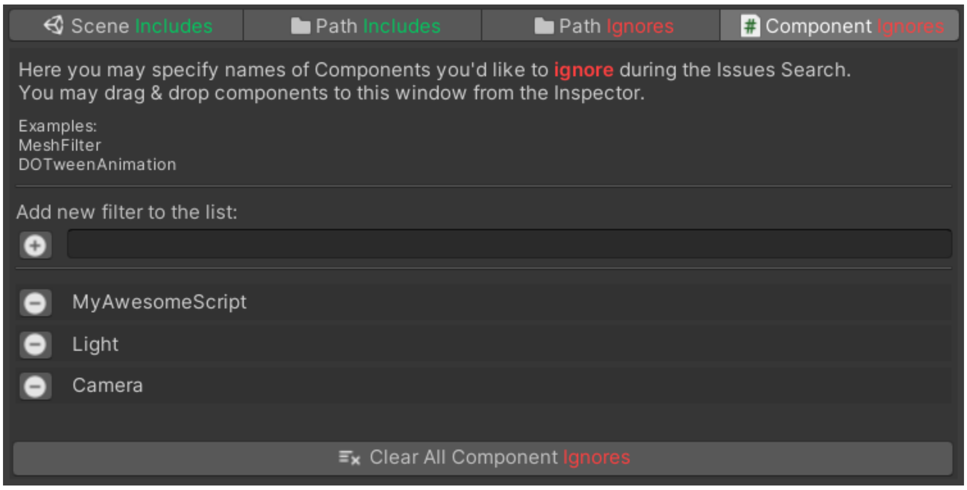
Note: If you have both Includes and Ignores added, first Includes are applied, then Ignores are applied to the included paths.

Add new filter to the list:

Path □ Ignore Case

Path	Match Case	Assets/Tests/Master/Preformance Tests/Bench.cs
Path	Match Case	Assets/Tests/Master/Preformance Tests/Scenes
Path	Match Case	Assets/Tests/Master/Persistent

Path Ignores tab allows excluding specified files, folders, extensions or partial paths from the Issues Search.
You may add filters manually or just drag & drop items from the Project Browser.
You may exclude files with specified extension, full or partial paths and names, both with Match or Ignore Case.
If you have both Includes and Ignores added, first Includes are applied, then Ignores are applied to the included items.



Scene Includes | Path Includes | Path Ignores | Component Ignores

Here you may specify names of Components you'd like to **ignore** during the Issues Search.
You may drag & drop components to this window from the Inspector.

Examples:
MeshFilter
DOTweenAnimation

Add new filter to the list:

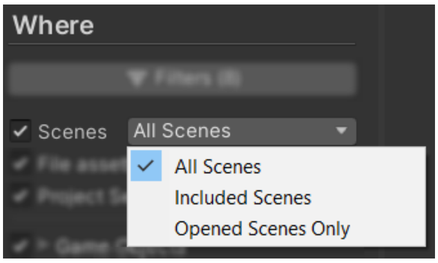
⊕

⊖ MyAwesomeScript
⊖ Light
⊖ Camera

Clear All Component Ignores

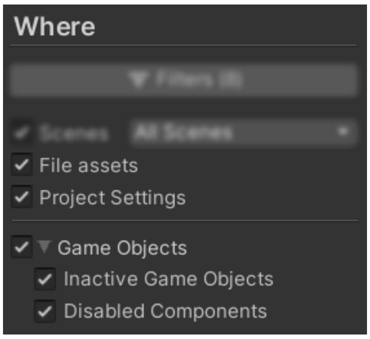
Component Ignores tab allows excluding components with specified names from the Issues Search.
You may add names by hand or just drag & drop desired component from inspector to this window.
In case you'll add component by hands, entered name will be looked in all currently loaded assemblies before saving it to the list to make sure you entered it correctly.

***Where** settings section*



Scenes. Includes scenes. Uncheck to exclude all objects and settings placed in scenes from the Issues lookup.

- **All Scenes**. Performs search in all scenes in the project.
- **Included Scenes.** Checks only those scenes you have included at the Scene Includes filters. *By default, all enabled scenes from the "Scenes in build" are included.*
- **Opened Scenes Only**. Performs search only in currently opened scenes (multi-scene setup is supported). Path Ignores will not affect such search.

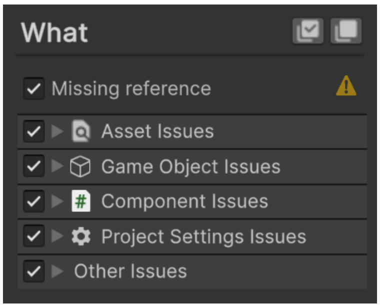


File assets. Includes file assets (like prefabs or shader scripts).

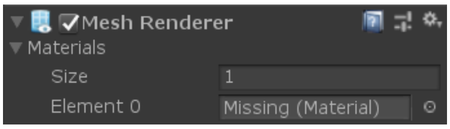
IMPORTANT:
In-scene prefab instances, which have no changed (overriding) properties, will be skipped when this option is checked since all such objects will be scanned as File assets.

Project Settings. Includes Project Settings into the issues search.
Game Objects. Includes Game Objects into the issues search (in scenes and prefab assets if scenes and File assets are included).
Inactive Game Objects. Includes inactive Game Objects.
Disabled Components. Includes disabled Components.

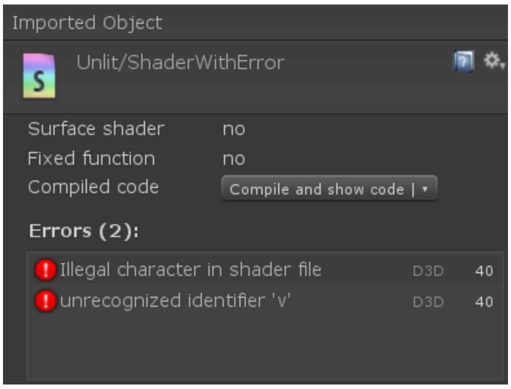
***What** settings section*



Here you can pick desired kinds of issues you would like to check. There are Game Object - specific issues, Project Settings - specific issues, and few common issues out of any groups.



Missing reference. Search for any missing references across whole project, depending on your "*Where?*" settings. Reference can miss when you remove referenced asset from the project.



Imported Object

Unlit/ShaderWithError

Surface shader no

Fixed function no

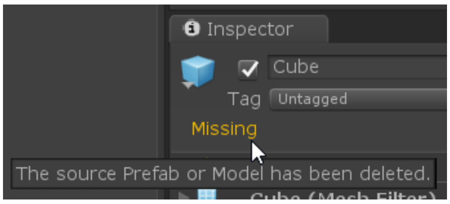
Compiled code Compile and show code | ▾

Errors (2):

Illegal character in shader file D3D 40

unrecognized identifier 'v' D3D 40

Shader with error(s). Search for any shader scripts with compilation errors within project assets. Will not work with disabled **File assets** "**Where**" option.



Inspector

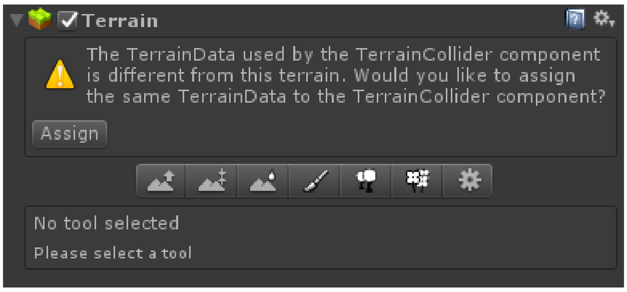
☑ Cube

Tag Untagged

Missing

The source Prefab or Model has been deleted.

Missing Prefab. Search for instances of prefabs which were removed from project.



☑ Terrain

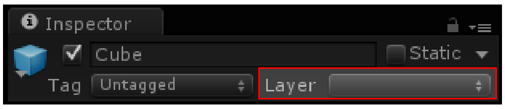
The TerrainData used by the TerrainCollider component is different from this terrain. Would you like to assign the same TerrainData to the TerrainCollider component?

Assign

No tool selected

Please select a tool

Inconsistent Terrain Data. Search for Game Objects where Terrain and TerrainCollider have different Terrain Data. It may lead to the incorrect collisions with your terrain. Though it may be expected in some cases, e.g. when you wish to make separate smoother collision data to reduce resources usage.

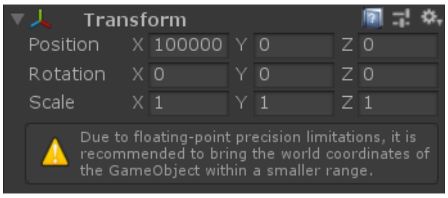


Inspector

☑ Cube □ Static

Tag Untagged Layer

Invalid Layer. Search for GameObjects with unnamed layers. Happens when you use some custom layer and remove it from Tag Manager afterwards.

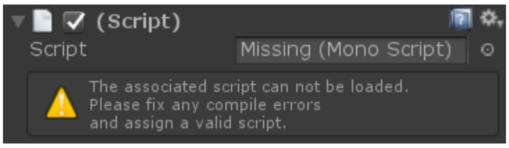


Transform

Position	X	100000	Y	0	Z	0
Rotation	X	0	Y	0	Z	0
Scale	X	1	Y	1	Z	1

Due to floating-point precision limitations, it is recommended to bring the world coordinates of the GameObject within a smaller range.

Huge position. Search for GameObjects with too big world positions (more than 100k by any axis). Will check both regular objects and UI objects as well.

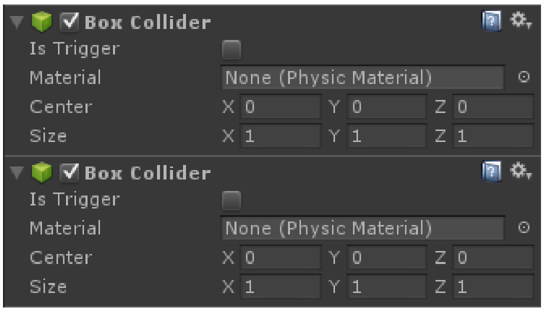


☑ (Script)

Script Missing (Mono Script)

The associated script can not be loaded. Please fix any compile errors and assign a valid script.

Missing Component. Search for the missing components at the Game Objects or Scriptable Objects. Happens when you delete MonoBehaviour or Scriptable Object scripts, sometimes accidentally.



☑ Box Collider

Is Trigger □

Material None (Physic Material)

Center X 0 Y 0 Z 0

Size X 1 Y 1 Z 1

☑ Box Collider

Is Trigger □

Material None (Physic Material)

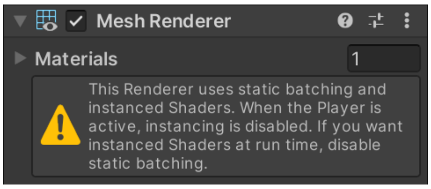
Center X 0 Y 0 Z 0

Size X 1 Y 1 Z 1

Duplicate Component. Search for the multiple instances of the same component on the same object, taking into account values of the components' properties.

IMPORTANT:

Gradient values are not supported for now (so you may get false positives if you have similar components with only differed gradients).

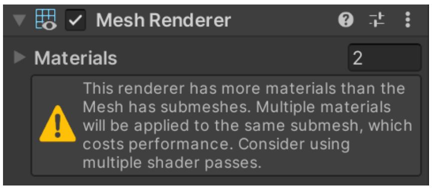


☑ Mesh Renderer

Materials 1

This Renderer uses static batching and instanced Shaders. When the Player is active, instancing is disabled. If you want instanced Shaders at run time, disable static batching.

Invalid Renderer Batching. Look for renderers with incompatible batching setup.

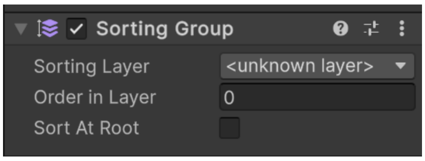


☑ Mesh Renderer

Materials 2

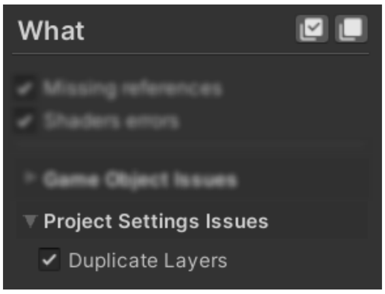
This renderer has more materials than the Mesh has submeshes. Multiple materials will be applied to the same submesh, which costs performance. Consider using multiple shader passes.

Invalid Renderer Materials. Look for renderers with incorrect materials setup.



Sorting Layer <unknown layer>
Order in Layer 0
Sort At Root □

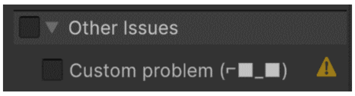
Invalid Sorting Layer. Search for the components where Sorting Layer is set incorrectly.



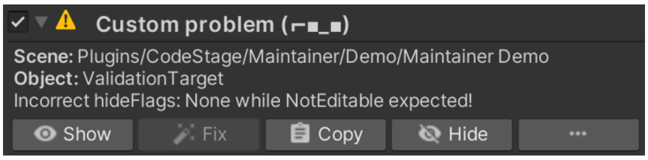
☑ Duplicate Layers

Issues Finder can also check your project settings for possible issues.

Duplicate Layers. Looks for the duplicate entries at the *Tags and Layers* settings of the project.



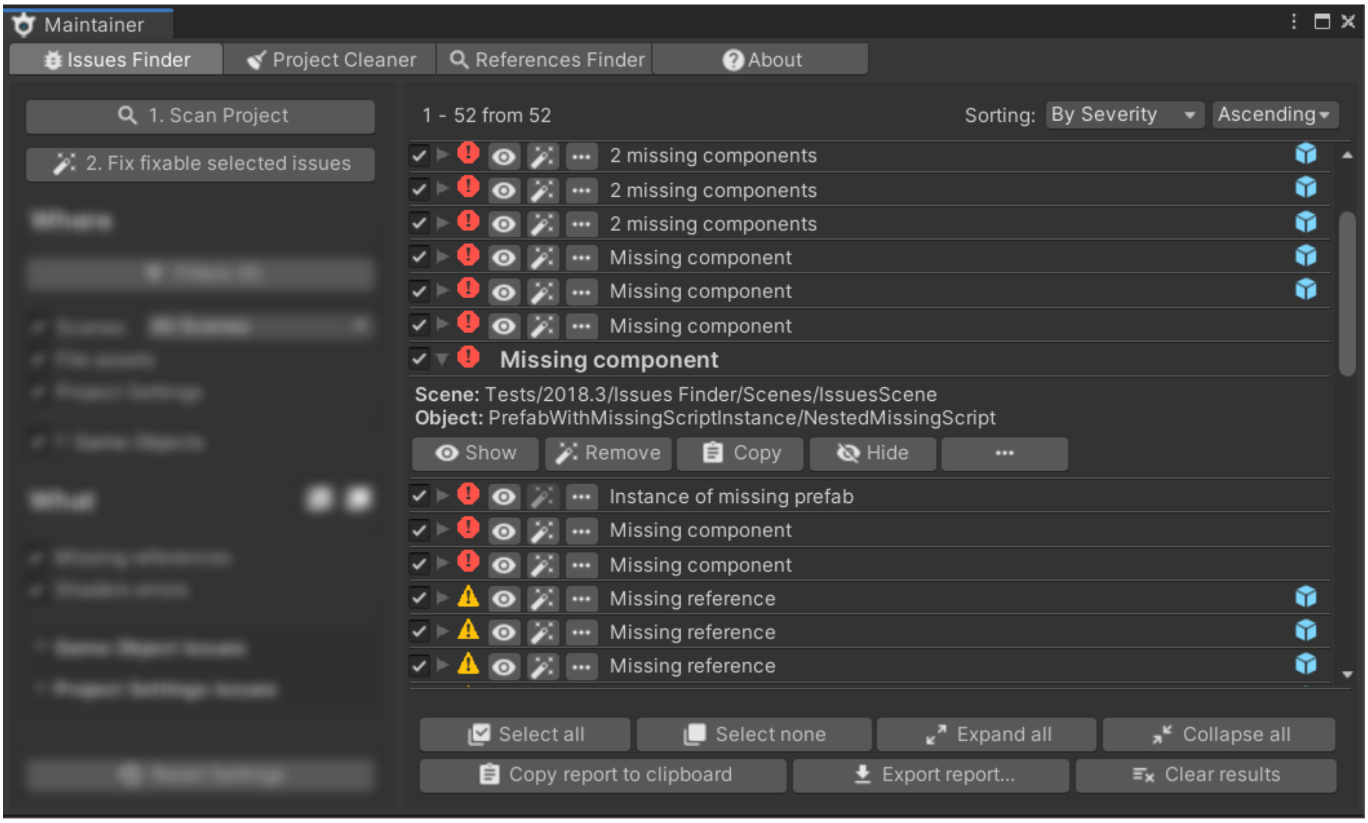
□ Custom problem (⌐■_■)



Scene: Plugins/CodeStage/Maintainer/Demo/Maintainer Demo
Object: ValidationTarget
Incorrect hideFlags: None while NotEditable expected!

It's possible to add any custom issues detection through the API. See IssueDetector API documentation for details.

Issues Finder Usage



1 - 52 from 52

Sorting: By Severity Ascending

2 missing components
2 missing components
2 missing components
Missing component
Missing component
Missing component
Missing component
Scene: Tests/2018.3/Issues Finder/Scenes/IssuesScene
Object: PrefabWithMissingScriptInstance/NestedMissingScript
Instance of missing prefab
Missing component
Missing component
Missing reference
Missing reference
Missing reference

After you have done with settings, just press "**1. Scan Project"** button or **CTRL / CMD + SHIFT + ALT + F** shortcut to start issues search. If you have any unsaved changes in current scene, you will be prompted to save them before search starts in case you did not choose Opened Scenes Only filtering option.

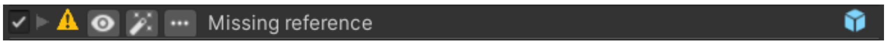
You will see search progress indicating current search phase and giving you some insight on what is currently happening. After search, you will see search results in the Maintainer.

Results list has such elements:

- Current page issues indexes and total issues count.

- Sorting options. You may sort by type, severity and path both ascending and descending.
- Found issues list itself (will be covered in more details below).
- Paging controls if there are more than 100 issues found (one page shows 100 issues).
- Few self-describing helper buttons.

Results list consists of separate issues records. Records are collapsed to the compact view by default and look like this:



Missing reference

Compact record view has such elements:

- Selection box. Allows selecting items for the auto-fix. Use **2. Fix fixable selected issues** button to fix selected items.
- Expand arrow. Allows expanding a record to the detailed view.
- Severity icon. Shows severity (importance) level. Info, Warning or Critical.
- **Show** button (eye icon). Navigates you to the issue as close as possible.
  Works differently for issues from different sections:

  How Show button works:

  If issue was found in Game Object, Component or Component's field: opens containing Scene or Prefab, pings it at the Project Browser, selects target Game Object in Hierarchy view, tries to collapse all components on the Game Object keeping expanded target component if record has it.

  If issue was found somewhere in Project Settings: opens or highlights desired section of the Project Settings.

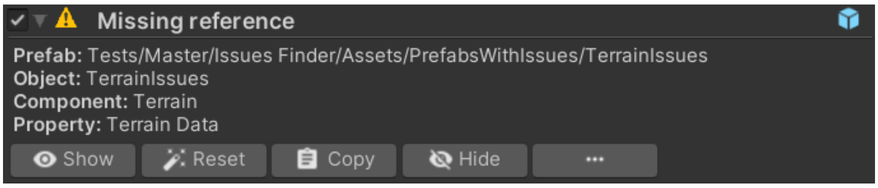
  If issue was found in Scene Lighting settings: opens containing scene, pings it at the Project Browser, opens Lighting settings window.

  In other cases, it works in similar manner - tries to open and / or select target asset / setting / window and pings objects and / or assets with issue.

- **Fix** button (magic wand icon). Applies suitable auto-fix for the issue, if possible. Hover mouse cursor to see the tooltip with description of the applied fix.
- **More** button (dots icon). Shows menu with additional functionality, such as quick add to the ignores.
- Issue caption. Usually briefly describes issue. May have some useful info like index of the unnamed layer to let you easily locate and name that layer in Tags Manager.
- Optional Prefab icon for issues found in prefab file assets at the right-hand corner of the row.

If you wish to see more details about desired issue, just press at the expand arrow, or click the record itself (except other buttons of course) to expand it to the detailed view.

Detailed view looks like this:



Missing reference

Prefab: Tests/Master/Issues Finder/Assets/PrefabsWithIssues/TerrainIssues
Object: TerrainIssues
Component: Terrain
Property: Terrain Data

Detailed record view has same elements as Compact record view, with additional items:

- Additional info and issue location. Depending on issue kind, can include such levels of precision:

  For Game Object Issues:

  - **Scene** (for objects in scene) / **Prefab** (for objects in prefab file assets) – path to the scene or prefab file.
  - **Game Object** - path to the Game Object in the scene.
  - **Component** (optional) - name of the Component, which has an issue.

    IMPORTANT:
    If Game Object has more than one of such Components, you will see the (#*) added to the Component's name. This is a sequence number of the target Component.

  - **Property** (optional) - name of the property, which has an issue.

  For Project Settings Issues:

  - Where issue was found

- What exactly wrong there
- May vary for different kinds of issues

- **Copy** button. Copies record text to the clipboard.
- **Hide** button. Hides record from the results list. Useful when you wish to hide appropriate record from the search results after you fixed an issue.

If you wish to see less details about expanded issue, just press at the collapse arrow, or click the record itself (except other buttons of course) to collapse it to the compact view.

Project Cleaner

This module allows finding unused (unreferenced) assets in your project and lets you clean it up with few clicks.

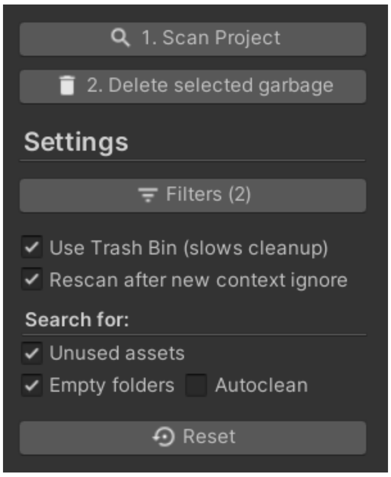
IMPORTANT:

- **Please note, this module does removes files and folders physically from your system. Please always make a backup of your project before using Project Cleaner! Author is not responsible for any data loss due to use of the Project Cleaner!**
- **Keep in mind that Project Cleaner looks for the assets which are not referenced anywhere in the project. However, they still may be used dynamically at runtime. Project Cleaner tries to avoid such cases ignoring special folders, but, for example, some custom build systems still may affect this. So please carefully inspect each found item before removing it.**

You can use this module both from Maintainer window and via menu command / shortcut (see **Installation and setup**). If you will use it via menu command / shortcut, it will perform garbage search with latest (or default) settings and will show results in the Maintainer window after that.

Project Cleaner Settings

If you wish to tune settings before search, open Maintainer window first, go to the **Project Cleaner** tab and tune them from there:



☑ Use Trash Bin (slows cleanup)
☑ Rescan after new context ignore

Search for:
☑ Unused assets
☑ Empty folders □ Autoclean

Here you can see settings available for this module.

Settings are made of two separate groups: different filtering and ignoring settings and "Search for" settings.
Let me just describe them one-by-one, in their exact order.

Filters… button. Opens **Precise Filters** window with three tabs – **Scene Ignores**, **Path Includes** and **Path Ignores**.

Number in brackets indicate active filters amount.

Precise Filters tabs

Tabs are equal to the filters tabs from the Issues Finder module.

Scene Ignores tab allows excluding specific scenes from the garbage search. Ignored scenes will be considered needed and everything used in them will be excluded from the garbage search.

Path Includes tab allows including paths, extensions etc. which should filter out results by including only what's specified on this tab.

Path Ignores tab allows excluding specified files, folders or partial paths from the Garbage Search. Paths from this tab will be ignored from included paths if both Includes and Ignores are specified (includes are applied before ignores).

IMPORTANT:

Please keep an eye on your lists of filters to make sure they do not have old and not actual items to avoid extra waste of the resources and time during the search.

Rest of the settings

Use Trash Bin. Keep this on checked if you wish to move all garbage to the Trash Bin instead of deleting it permanently. Always recommended setting, it will allow you to recover stuff you deleted by mistake. It may not work in some environments, for example, early Unity builds for Linux ignore this option. Please note, this option reduces cleanup performance, when enabled.

Rescan after new context ignore. When checked, garbage scan will be automatically re-started after you add any new ignore filters from the results "More" buttons context menus.

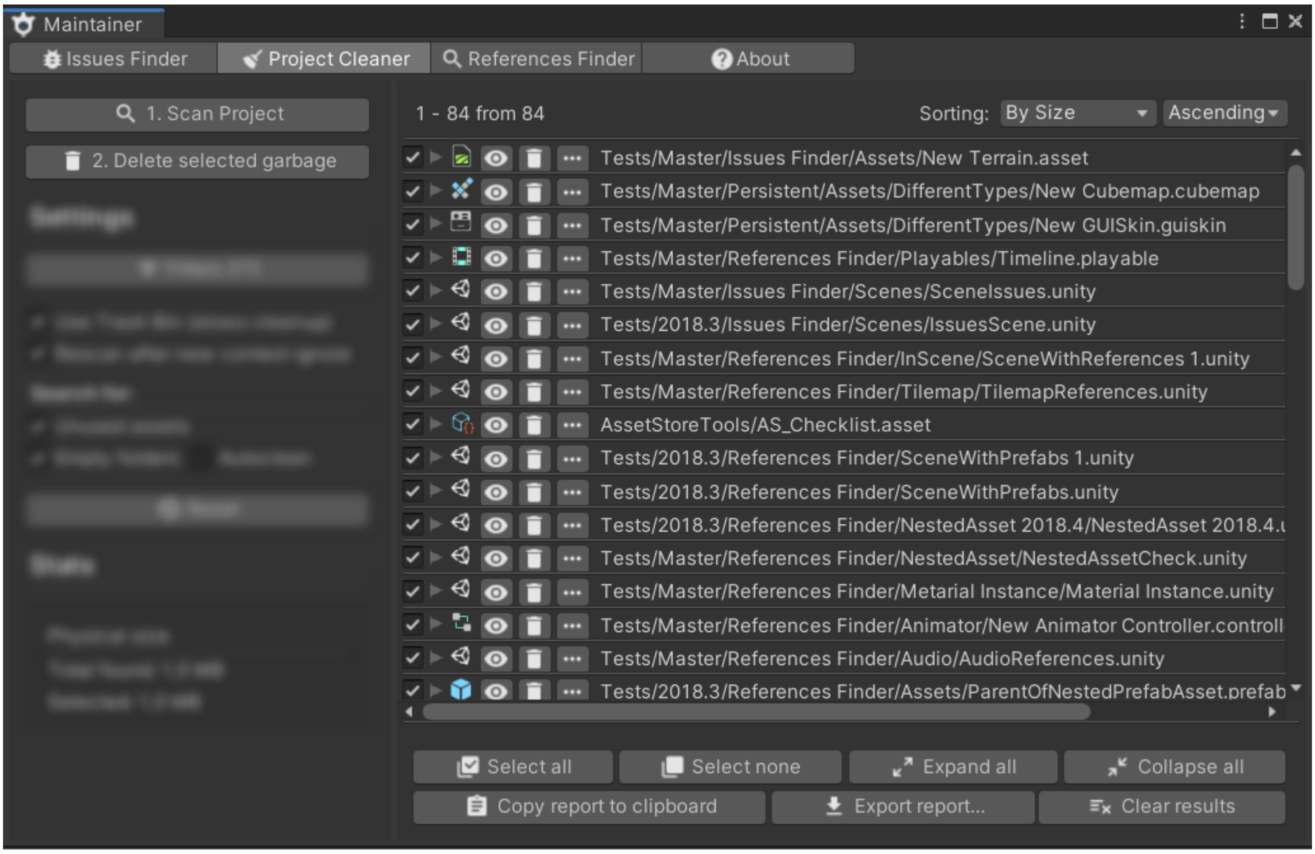
Unused Assets. Check to let Project Cleaner look for unused assets in your project.

IMPORTANT:

- Project Cleaner doesn't look for the unused scripts, since it's nearly impossible to reliably say if some script is used or not because it might be accessed by name or using Reflection at runtime, etc.
- In addition, to avoid false positives, Project Cleaner skips some special folders like Plugins and Editor.
- Unity 2019.3 or newer: Project Cleaner checks your last BuildReport to make search more accurate. Search will be more accurate if you will have build information persisted in your project's Library folder (Unity saves it to the Library/LastBuild.buildreport). You'll see a warning at the Project Cleaner's left panel if last build report will not be found. Just make a build for the target platform to make use of this feature.

Empty folders. Finds empty directories in your project. You also may enable **Autoclean** option to automatically find and delete empty folders on scripts recompile. Useful when you work in big team using GIT or similar VCS that do not synchronize empty folders. It allows to automatically cleaning empty folders for all your colleagues as soon as they get the Maintainer with configured settings from the repository.

Project Cleaner Usage



1 - 84 from 84

Sorting: By Size Ascending

- ✓ Tests/Master/Issues Finder/Assets/New Terrain.asset
- ✓ Tests/Master/Persistent/Assets/DifferentTypes/New Cubemap.cubemap
- ✓ Tests/Master/Persistent/Assets/DifferentTypes/New GUISkin.guiskin
- ✓ Tests/Master/References Finder/Playables/Timeline.playable
- ✓ Tests/Master/Issues Finder/Scenes/SceneIssues.unity
- ✓ Tests/2018.3/Issues Finder/Scenes/IssuesScene.unity
- ✓ Tests/Master/References Finder/InScene/SceneWithReferences 1.unity
- ✓ Tests/Master/References Finder/Tilemap/TilemapReferences.unity
- ✓ AssetStoreTools/AS_Checklist.asset
- ✓ Tests/2018.3/References Finder/SceneWithPrefabs 1.unity
- ✓ Tests/2018.3/References Finder/SceneWithPrefabs.unity
- ✓ Tests/2018.3/References Finder/NestedAsset 2018.4/NestedAsset 2018.4.u
- ✓ Tests/Master/References Finder/NestedAsset/NestedAssetCheck.unity
- ✓ Tests/Master/References Finder/Metarial Instance/Material Instance.unity
- ✓ Tests/Master/References Finder/Animator/New Animator Controller.controll
- ✓ Tests/Master/References Finder/Audio/AudioReferences.unity
- ✓ Tests/2018.3/References Finder/Assets/ParentOfNestedPrefabAsset.prefab

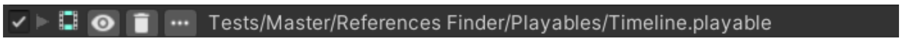
After you have done with settings, just press the "**1. Scan project**" button or **CTRL / CMD + SHIFT + ALT + G** short key to perform project-wide search for unused assets.

You will see search progress indicating current search phase and giving you some insight on what is currently happening. After search, you will see search results in the Maintainer.

Results list has such elements:

- Currently shown items indexes and total items count.
- Sorting options. You may sort by path, size, and type both ascending and descending.
- Found unused (unreferenced) assets list itself (will be covered in more details below).
- Paging controls if there are more than 100 items found (one page shows 100 items).
- Some self-describing helper buttons.

Results list consists of separate records. Records are collapsed to the compact view by default and look like this:

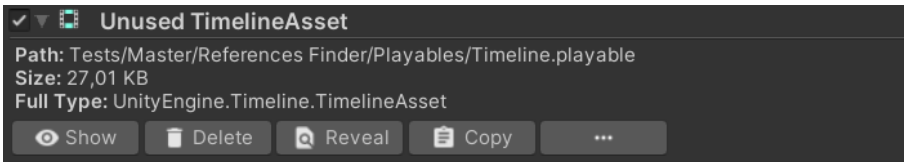


Tests/Master/References Finder/Playables/Timeline.playable

Compact record view has such elements:

- Selection box. Allows selecting items for the cleanup. Only checked items will be deleted on the cleanup phase.
- Expand arrow. Allows expanding a record to the detailed view.
- Asset icon. Shows icon similar to what you see at the Project Browser for each asset.
- **Show** button (eye icon). Selects asset at the Project Browser.
- **Delete** button (trash bin icon). Immediately deletes asset and removes item from the results list.
- **More** button (dots icon). Shows menu with additional functionality, as quick add to the ignores.
- Unused asset path relative to the Assets folder.

If you wish to see more details about desired garbage, just press at the expand arrow, or click the record itself (except other buttons of course) to expand it to the detailed view.
Detailed view looks like this:



Unused TimelineAsset

Path: Tests/Master/References Finder/Playables/Timeline.playable
Size: 27,01 KB
Full Type: UnityEngine.Timeline.TimelineAsset

Detailed record view has same elements as Compact record view, with additional items:

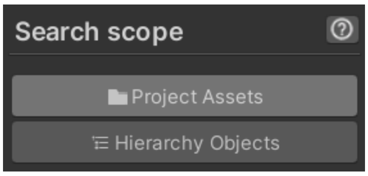
- Asset size (if it is not an empty folder).
- Full type (if it is not an empty folder).
- **Reveal** button. Shows file or folder in the system File Manager (Explorer on Windows, Finder on Mac, etc.).
- **Copy** button. Copies record description to the clipboard.

If you wish to see less details about expanded garbage item, just press at the collapse arrow, or click the record itself (except other buttons of course) to collapse it to the compact view.

After you observed found assets, just check desired items and clean them up using the "**2. Delete selected garbage**" button.

References Finder

This module allows to accurately find references in your project.
There are two references search *scopes* available: **Project Assets** scope and **Hierarchy Objects** scope.



You can switch scope view using Search scope buttons at the left column.

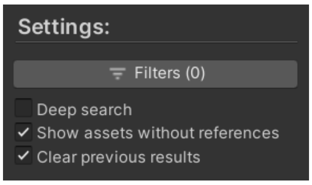
Project Assets scope allows looking for assets references in whole project.

Hierarchy Objects scope allows looking Game Objects and Components references in current Scene or Prefab.

Project Assets scope

In this scope, any asset reference can be found in whole project to let you quickly find out where exactly this or that asset is referenced in your project, if referenced at all.

Project Assets scope Settings


Settings:

Filters (0)

□ Deep search
☑ Show assets without references
☑ Clear previous results

Filters button opens filtering options with **Path Includes** and **Path Ignores** tabs, similar to the precise filters tabs at the *Issues Finder Settings* section covered above.

IMPORTANT:
Please keep an eye on your lists of the filters to make sure they do not have old and not actual items to avoid extra waste of the resources and time during the References search.

Deep search. Includes more items into the Project References search, like fields with [HideInInspector] attribute or hidden system fields. Can reduce search performance and produce confusing results in some scenarios. Changing this setting will trigger Exact References Entries cleanup.

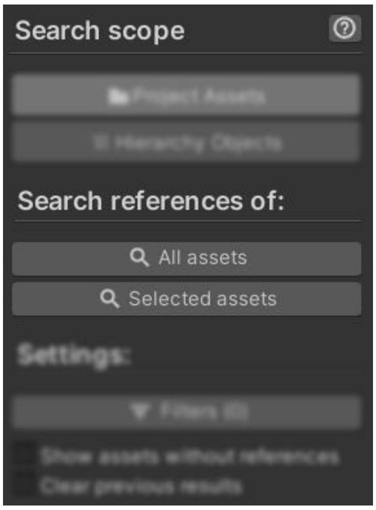
Show assets without references. Enable to keep assets without found references in list. They will be just dimmed a bit to indicate they have no references found.

Clear previous results. Enable to automatically clear previous search results on new search.

PERFORMANCE NOTES:

- First assets references search may run noticeably slower, especially if it is a project-wide search of all assets references.
- Internal assets cross-references map is created on first run.
- **All assets** references search usually runs much slower comparing to the **Selected assets** search.
- **Deep search** includes invisible serialized properties into the search making it slower in some scenarios.

Project Assets scope usage

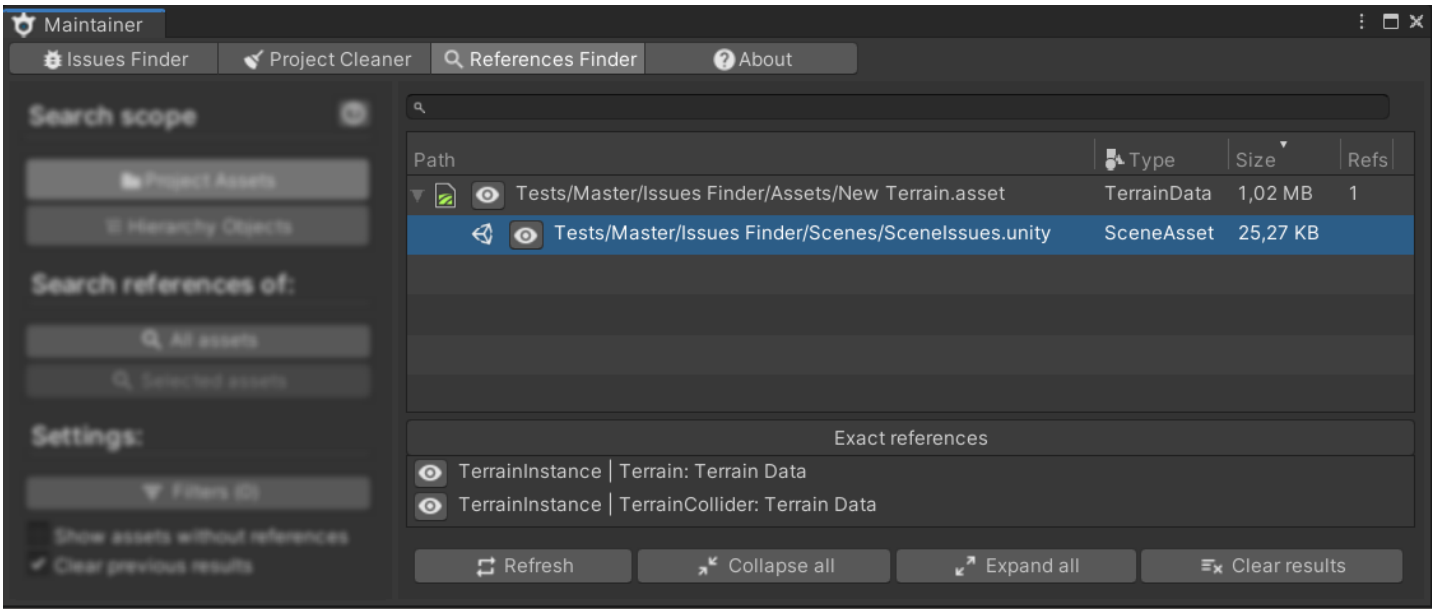


To run asset reference search, use either:

- Main menu command or CTRL / CMD + SHIFT + ALT + R shortcut (see *Installation and setup*).
- Project Browser context menu item Maintainer > Find References in Project or CTRL / CMD + SHIFT + ALT + S shortcut for selected assets.
- Component context menu Maintainer: Script File References for components made from script asset files.
- **All assets** button at the left column (this might take a lot of time!).
- **Selected assets** button at the left column for selected assets.
- Drag & drop asset(s) or components created from asset script files right to the right-hand area with results view.

If you have any unsaved changes in opened scenes, you will be prompted to save them before search starts.

You will see search progress indicating current search phase and giving you some insight on what is currently happening. After search, you will see search results in the Maintainer.



Path	Type	Size	Refs
Tests/Master/Issues Finder/Assets/New Terrain.asset	TerrainData	1,02 MB	1
Tests/Master/Issues Finder/Scenes/SceneIssues.unity	SceneAsset	25,27 KB	

Exact references

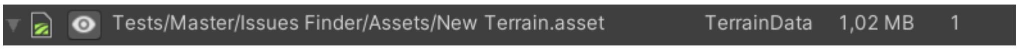
TerrainInstance | Terrain: Terrain Data

TerrainInstance | TerrainCollider: Terrain Data

Here you can see (from top to the bottom):

- **Search bar**. Use it to quickly filter out results by Path. You may enter any part of the path and search is case-insensitive.
- **Results** list. It has header with few columns: Path, Type, Size and Refs (references count). Results may be sorted by any of these columns. Each column may be resized. Header also has a context menu where you may hide or show columns and auto-resize headers to fit the width. More details on the results in this list please read below.
- **Exact references** list. If asset was referenced from scene or prefab and Maintainer could figure out exact reference location (up to the exact component field), it will add such locations to this list. See more details on this below.
- **Refresh** button. This button restarts last search and refreshes contents of the results if something changed.
- **Collapse all / Expand all** buttons. Just collapses or expands all results items.
- **Clear results** button. Clears last search results.

Results list consists of separate ***Source* Asset Rows** for each scanned source asset.



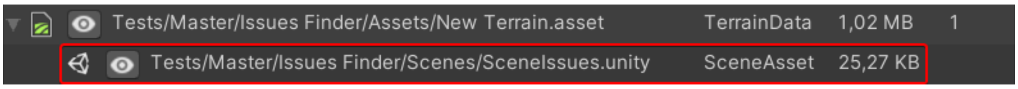
Tests/Master/Issues Finder/Assets/New Terrain.asset TerrainData 1,02 MB 1

Each source Asset Row has:

- **Expand / Collapse arrow**, if any references were found to this asset.

- Asset **icon**.
- **Show button**. Selects this source asset in the Project Browser.
- **Path** to the source asset file in the project we look for the references to.
- **Type**, file **size** and found **references** count (if all columns are enabled in header context menu).

Source Asset Row will have expand / collapse arrow in case there are any references found to the source asset.
When you expand ***Source*** Asset Row, you'll see ***Reference*** Asset Rows – all assets which do reference to the source asset.



Tests/Master/Issues Finder/Assets/New Terrain.asset TerrainData 1,02 MB 1
Tests/Master/Issues Finder/Scenes/SceneIssues.unity SceneAsset 25,27 KB

If reference was found at the project or scene settings, you will see the gear icon next to it.

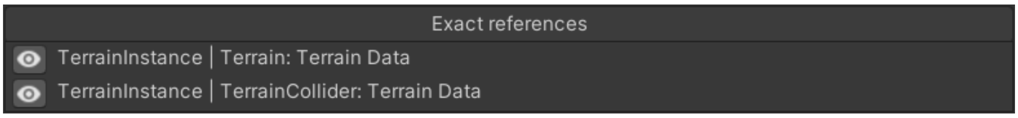
Reference Asset Rows are similar to the Source Asset Rows except they cannot be expanded further and they have few additional features:

Show button. Similar to Source Asset Row's reveal, but also can open Settings windows in case reference was found in settings.

Find references button. Searches references of this Reference Asset to see where it is used. Selects it in the results if it's already presents there. It is optional and shows only for referenced assets.

Reference Asset Row has no Refs column value.

Select Reference Asset Row to see where exactly Source asset is referenced, if exact references were found. **Exact references** are printed at the lower resizable pane:



Exact references

TerrainInstance | Terrain: Terrain Data

TerrainInstance | TerrainCollider: Terrain Data

Generally, exact reference has such mask: [Optional Hint] Transform Path | Component: Field [Array item], but it may vary for different kind of references.

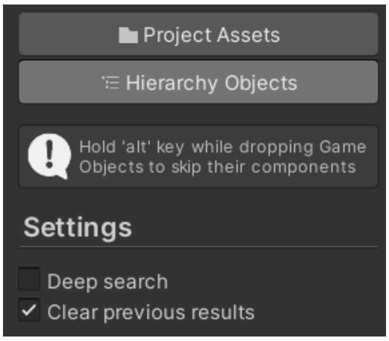
Each exact reference has own **Show** button which navigates to the found reference as closer as possible. It will open settings inspector if reference was found in settings. In case of reference in scene object, it will open the target scene, select target object and fold all components except the target one if necessary, to make it as easy as possible for you to find the reference by the eye.

When navigation needs to open scene, it tries to add it to the current scene setup but uses only 1 extra slot so it will close previously opened scene for navigation before opening new one.

Hierarchy Objects scope

In this scope, any Game Object or Component reference can be found in current Hierarchy (e.g. Scene, Prefab, multi-scene setup).

Hierarchy Objects scope Settings



Hold 'alt' key while dropping Game Objects to skip their components

Settings

□ Deep search
☑ Clear previous results

Deep search. Includes more items into the Hierarchy References search, like fields with [HideInInspector] attribute or hidden system fields. Can reduce search performance and produce confusing results in some scenarios.

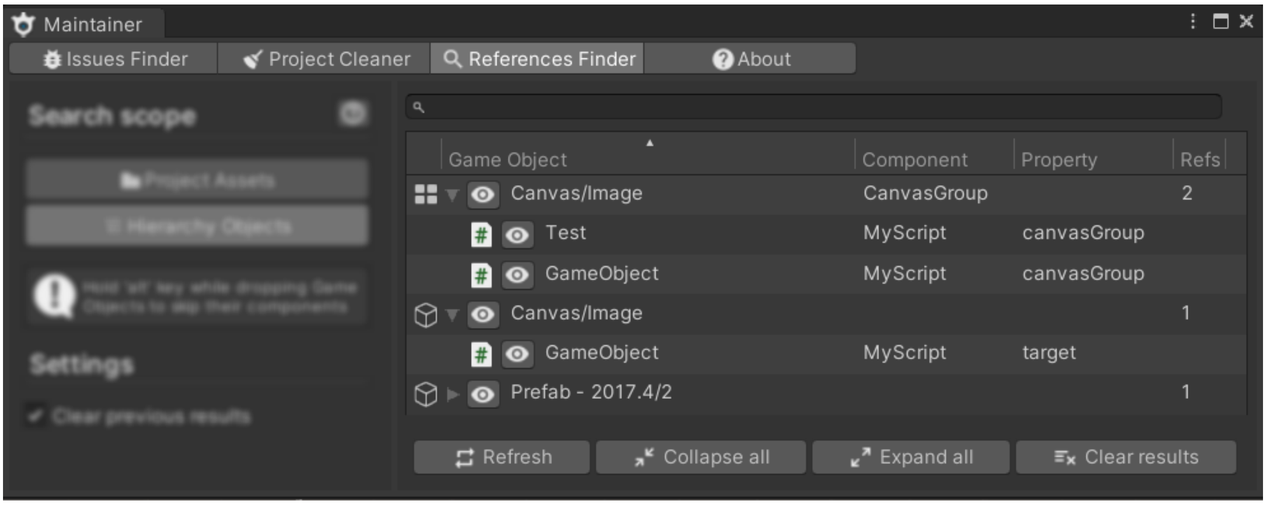
Clear previous results. Enable to automatically clear previous search results on new search.

Hierarchy Objects scope usage

To run Game Object or Component reference search, use either:

- Hierarchy context menu item Maintainer > Game Object References In Scene to look for selected Game Objects references in current Hierarchy.
- Hierarchy context menu item Maintainer > Game Object & Components References In Scene to look for selected Game Objects and all their components references in current Hierarchy (non-recursive, only selected items will be processed).
- Component context menu Maintainer: References In Scene for components.
- Drag & drop Game Objects right to the right-hand area with results view to look for selected Game Objects references in current Hierarchy. Hold ALT key while dropping to include only Game Objects, without their Components.

After search, you will see search results in the Maintainer.

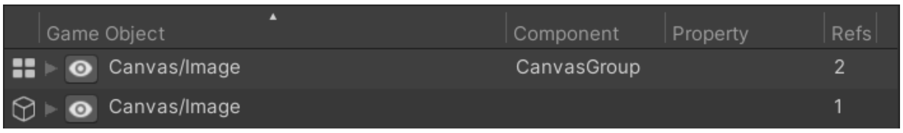


Game Object	Component	Property	Refs
Canvas/Image	CanvasGroup		2
Test	MyScript	canvasGroup	
GameObject	MyScript	canvasGroup	
Canvas/Image			1
GameObject	MyScript	target	
Prefab - 2017.4/2			1

Here you can see (from top to the bottom):

- **Search bar**. Use it to quickly filter out results by Game Object Transform Path. You may enter any part of the path and search is case-insensitive.
- **Results** list. It has header with few columns: Game Object (Transform path), Component, Property and Refs (references count). Results may be sorted by any of these columns. Each column may be resized. Header also has a context menu where you may hide or show columns and auto-resize headers to fit the width. More details on the results in this list please read below.
- **Refresh** button. This button restarts last search and refreshes contents of the results if something changed.
- **Collapse all / Expand all** buttons. Just collapses or expands all results items.
- **Clear results** button. Clears last search results.

Results list consists of separate ***Source* Rows** for each scanned source Game Object or Component.

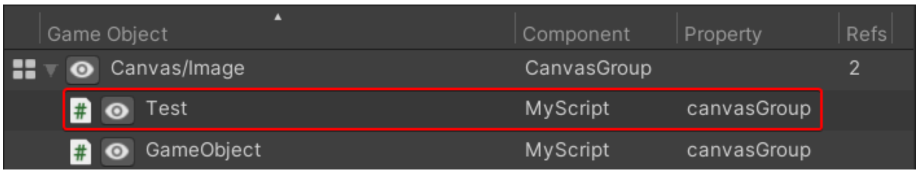


Game Object	Component	Property	Refs
Canvas/Image	CanvasGroup		2
Canvas/Image			1

Each Source Row represents source Game Object or source Components and has:

- Game Object or specific Component **icon**.
- **Expand / Collapse arrow**, if any references were found to this Game Object / Component.
- **Show button**. Selects Game Object and collapses all components except Source one if necessary.
- **Transform path** to the source Game Object (or Component's Game Object).
- **Component name** if source is a component.
- **References** count (if all columns are enabled in header context menu).

Source Row will have expand / collapse arrow in case there are any references found to the source object.
When you expand ***Source*** Row, you'll see ***Reference*** Rows – all Components which do reference to the source Game Object / Component.



Game Object	Component	Property	Refs
Canvas/Image	CanvasGroup		2
Test	MyScript	canvasGroup	
GameObject	MyScript	canvasGroup	

Reference Rows are similar to the Source Rows and represent exact references to the Source Objects.
The only difference is Component's **Property** name where reference to the Source Object was found.
Reference Row has no Refs column value.

Tips section

- Do not forget to configure filters in all modules! It is a great tool to improve modules performance and to reduce amount of not relevant results in all modules.
- You may keep only most important issues for you in Issues Finder module settings to improve search performance and reduce amount of not relevant results.
- All filters windows and References Finder module tab supports drag & drop from the Project view.

Using Maintainer from code

Maintainer has public API for all modules. APIs allow using and extending modules from your which may be useful when you wish to integrate Maintainer's modules into your pipelines.

Extending Assets Map dependencies resolve

In order to let Maintainer "know" about custom source dependencies from your assets which are not tracked by AssetDatabase.GetDependencies(), inherit your Dependencies Parser from the DependenciesParser base class (see example at API docs). That's all you need to do: Maintainer will automatically find and include your Dependencies Parser extension into the dependencies parsing process.

See online API docs for details:
docs.codestage.net/maintainer/api

Troubleshooting

- **I have errors in console after importing plugin.**
  - Consider making clean update: delete whole folder with plugin before importing new version into your project.
  - Make sure you have no namespaceless classes in your project which may interfere with some system classes.
  - If errors still there, please contact me, and I will try to help (see Support contacts below).
- **I cannot see issue at the specified object.**
  - Try to enable Debug Inspector mode, some issues may be not visible in Normal Inspector mode.
- **I cannot find target component with issue at the specified object because it has lots of such components.**
  - Check if issue record has (ID: ***) next to the Component name. In such case it indicates index number of that component in the inspector. Use show button to collapse all components on the Game Object except the target component to quickly distinguish it among others.
- **I cannot find target property with issue at the specified component both in Normal and Debug inspector modes.**
  - In some rare cases, actual property names may be obscured with PropertyDrawers or Editors. It is not an actual bug. However, if you face such cases too often - just let me know and we will think how to deal with it.
- **References Finder cannot find exact references locations for object in .max model (or any other non-native for Unity).**

- In order to be able to look for the exact references locations of non-native to Unity items, you usually should have third-party software installed, like 3ds Max, which Unity may use to parse such formats as .max models to let References Finder look into it.

Secret debug menu (use at own peril!)

You may activate super-secret (hush-hush, don’t tell anyone!) debug menu with few helpful buttons which may help workaround some issues while you’re waiting for a fix for example.
To activate this menu just open the About tab and press CTRL / CMD + D hotkey.

You will see these buttons right there, at the About tab:

- **Remove Assets Map**
  Deletes Assets Map file. Next demand will re-create Assets Map, possibly fixing itself if something went wrong previously.
- **Measure Assets Map build time**
  Deletes and re-creates Assets Map file with creation duration log to the console. Useful for benchmarking or debugging.
- **Remove User Settings and Close**
  Deletes User-specific Maintainer settings file and closes Maintainer window. User settings will reset to defaults on next Maintainer launch.
- **Remove All Settings and Close**
  Deletes all (both User and Project) Maintainer settings files and closes Maintainer window. All settings will reset to defaults on next Maintainer launch.
- **Re-save all scenes in project**
  Iterates all scenes in the project: opens and saves them, one by one. This action has two goals:
  - all scenes in project will be updated to the current Unity format if you have some scenes which weren’t re-saved in current Unity version yet
  - Unity will fix and clean out some garbage and incorrect dependencies in scenes helping References Finder module to be more accurate in some cases
- **Reload settings**
  Forces all (both User and Project) Maintainer settings reload.

Compatibility

Maintainer works with any Unity license and skin.
Maintainer works on all platforms (Windows, OS X, Linux).
Maintainer works fine with new scripting backend (.NET 4+) and Assembly Definition feature.

Plugin takes into account some specific behavior of the third-party plugins and assets to avoid false positives. For example, empty MeshFilters on objects with TextMeshPro or 2DToolkit components will be considered valid.
If you use some third-party assets that make false positives for you because of specific behavior, feel free to report them.

Friendly Assets

Do you like keeping your project in a good shape and have a lot of assets purchased where you can’t easily find and import the exact asset you need?

Then Asset Inventory by Impossible Robert is right for you, a tool to quickly import desired asset into the proper place, reverse find the asset and many more. That kind of tools which should be built into the Unity out of the box.

Final words from author

I hope you'll find **Maintainer** helpful in your daily work and it will save some of your priceless time!

Please, leave your reviews at the Asset Store, it's really important and any user feedback motivates me to continue my work with extra passion.

Also feel free to drop me bug reports, feature suggestions and other thoughts on the forum or via support contacts you'll find below.

Maintainer links and support

Asset Store | Homepage | API Reference | Discussions | YouTube

Support contacts:
discord.gg/FRK5HRzZvq
codestage.net/contacts
support@codestage.net

Subscribe for updates and news:
Discord Announcements Channel
@codestage_net
@codestage

Best wishes,
Dmitry Yuhanov
Asset Store publisher
codestage.net

P.S. #0 I wish to thank my family for supporting me in my Unity Asset Store efforts and making me happy every day!
P.S. #1 I wish to say huge thanks to ***Daniele Giardini*** *(DOTween, HOTools, Goscurry and many other happiness generating things creator) for awesome logos, intensive help and priceless feedback on this plugin!*