Table of contents

Installation

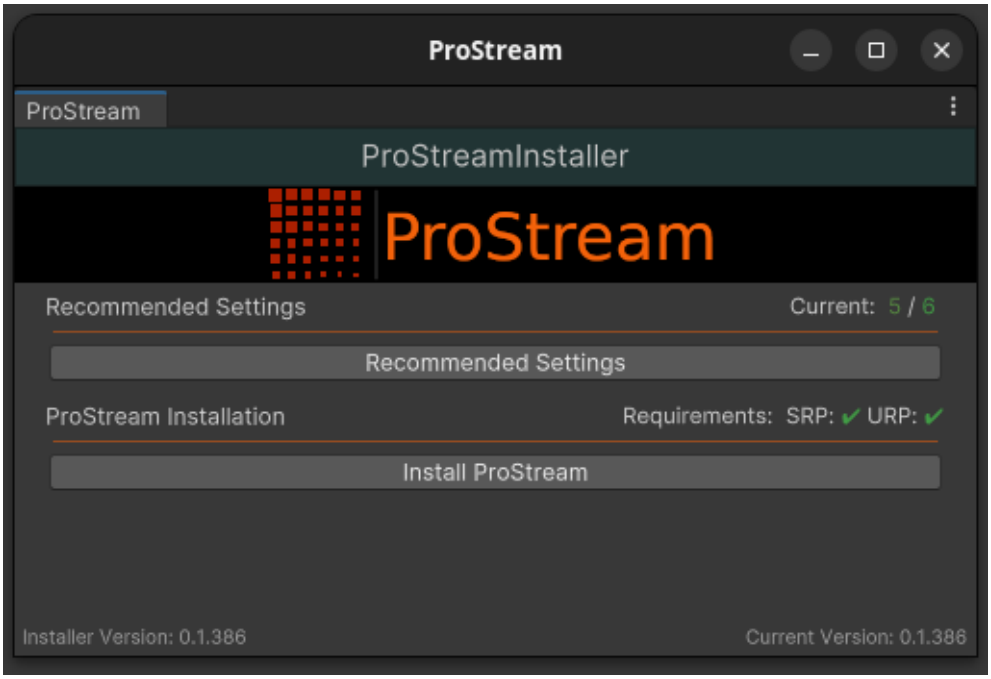
Upon first importing the ProStream package, the ProStream Installer will appear. For optimal performance use the settings described in the Recommended Project Settings ("Recommended Project Settings" in "Prerequisites"). These settings can be automatically applied in the `Recommended Settings` window of the installer.

If an appropriate SRP (URP/HDRP) is installed, you can press the `Install ProStream` button to begin the installation.

Why an Installer?

ProStream is designed to be installed as a package in order to ensure proper dependency installation and management and will not show up in your **Assets/** folder.

The purpose of the installer is to ensure that the package is installed to the and that the necessary dependencies are installed.



ps_installer.png

> ⚠ The installer can also be accessed by navigating to **Tools** > **instance.id** > **ProStream** > **(Install/Update) ProStream**.

Basic Workflow

Prerequisites

Prerequisites Details (Prerequisites)

Before you start, make sure that you have the following:

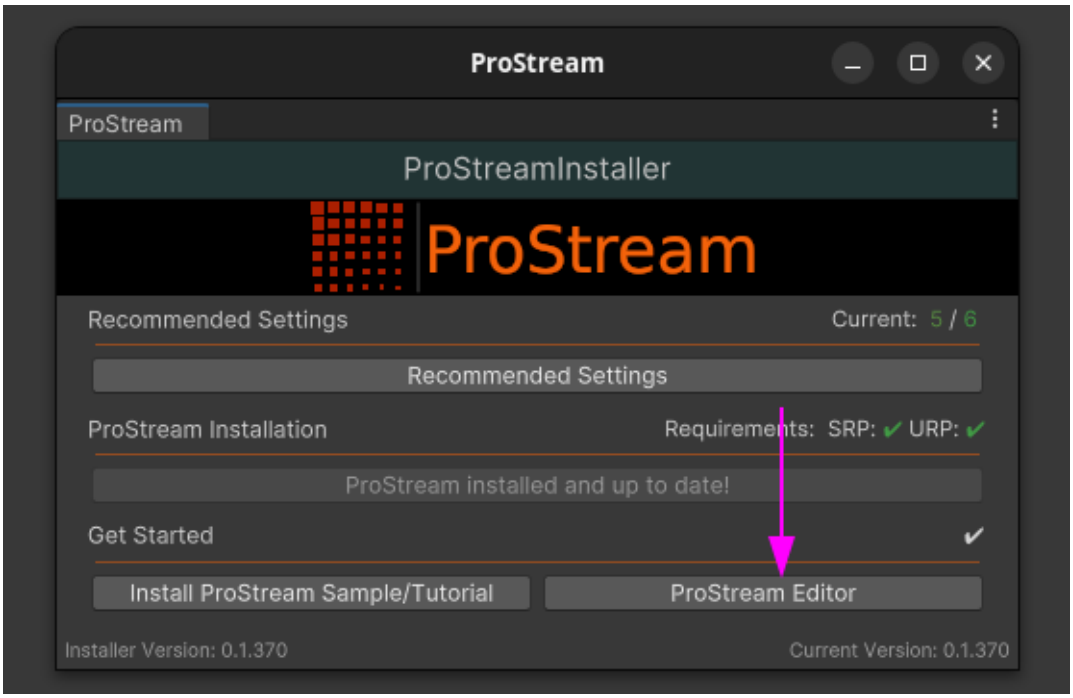
Requirements and Notes
• Unity 2022.3.0f1 or later (2022.3.21f1 is recommended) • Scriptable Render Pipeline (SRP) installed and configured (URP or HDRP) • Scene GameObjects which are not Prefabs (Importance Of Prefabs) are ignored • (Recommended) Scene Prefabs are children of an Empty Root-Level Parent GameObject

ProStream Editor

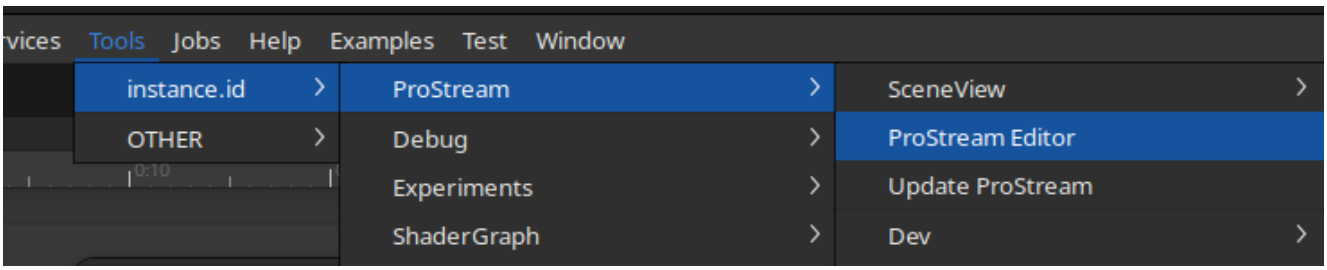
ProStream Editor Details (ProStream Editor)

To access the ProStream Editor, you can use the following methods:

1. Press `Shift+Alt+P` shortcut to open the ProStream Editor
2. Press the "ProStream Editor" button in the installer



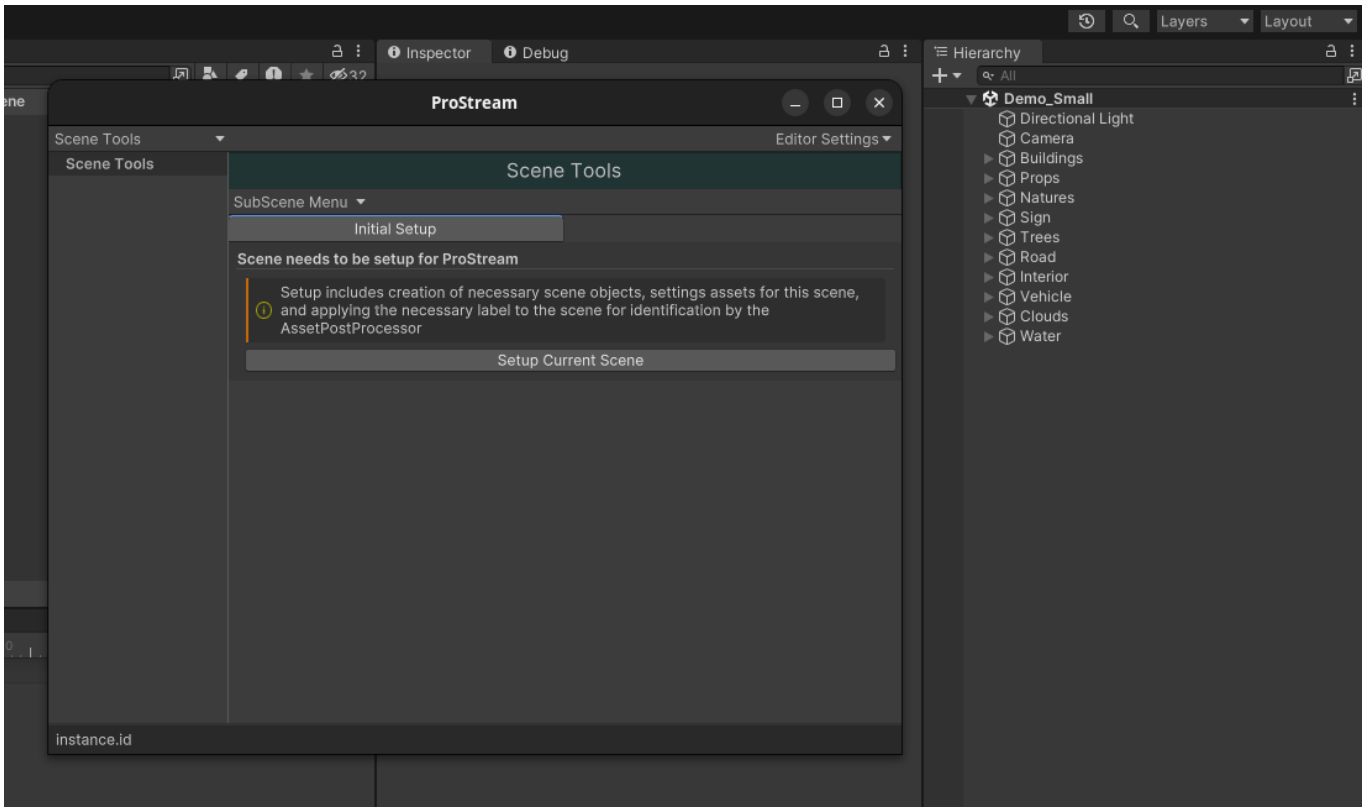
3. Go to the following Menu path `Tools`-> `instance.id`-> `ProStream`



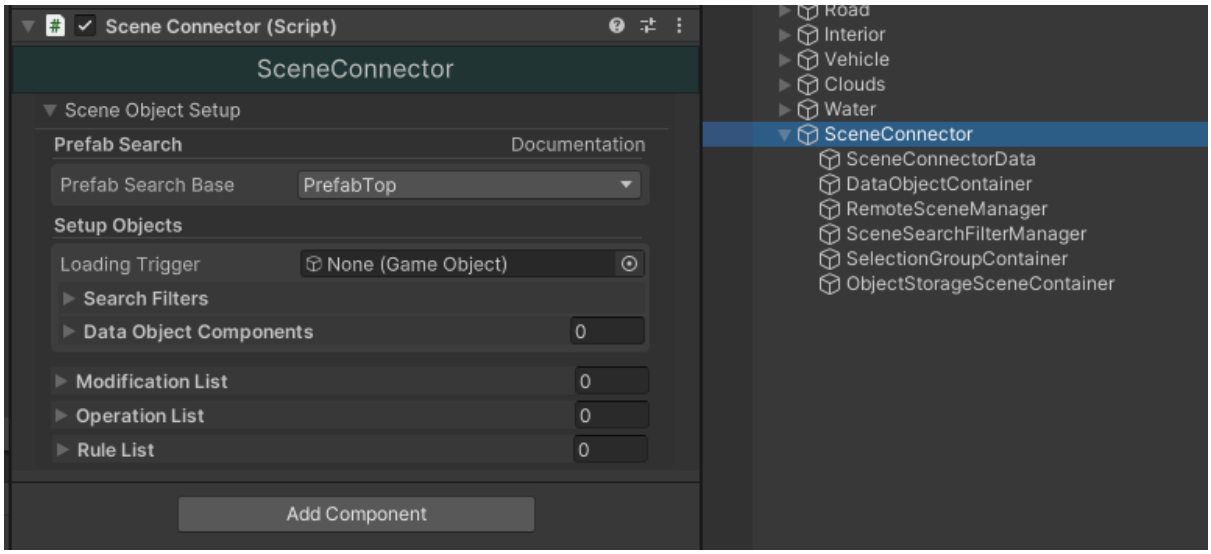
Setup Scene

ProStream Editor Details (ProStream Editor)

Press the "Setup Current Scene" button in the ProStream Editor to add the necessary components to the scene.



This will add SceneConnector (SceneConnector) the following components to the scene:

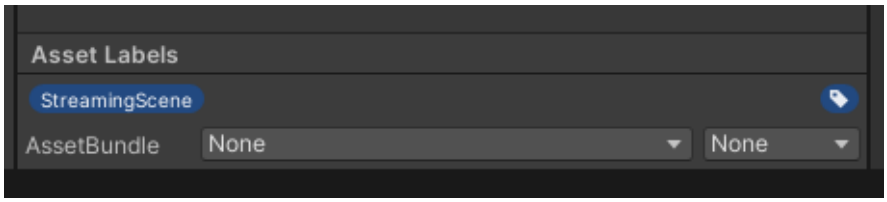


scene_connector_object.png

```
SceneConnector
  SceneConnectorData
  RemoteSceneManager
  DataObjectContainer
  SceneSearchFilterManager
```

```
SelectionGroupContainer
ObjectStorageSceneContainer
```

A new label will be added to the scene to indicate that it is a streaming scene.



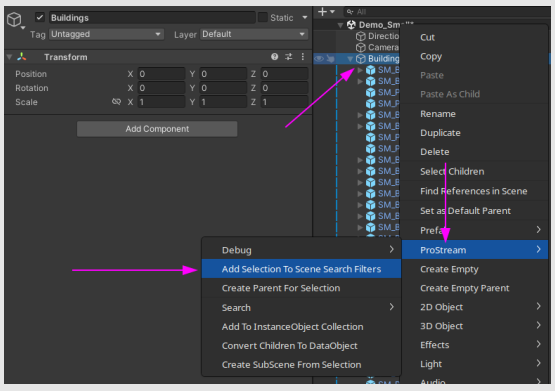
streaming_scene_label.png

Add Scene Search Filters

SceneSearchFilter Details (SceneSearchFilter)

> ⚠ It is highly recommended to use SceneSearchFilters, as they help improve setup performance by designating and limit the search scope of the RuleEngine Match system to the child Objects of the SceneSearchFilter GameObject.

> ℹ All GameObjects must be Prefabs to be compatible with ProStream systems. The Prefab system is used to differentiate one GameObject Hierarchy from another. on-Prefab GameObjects are skipped by the matching systems.

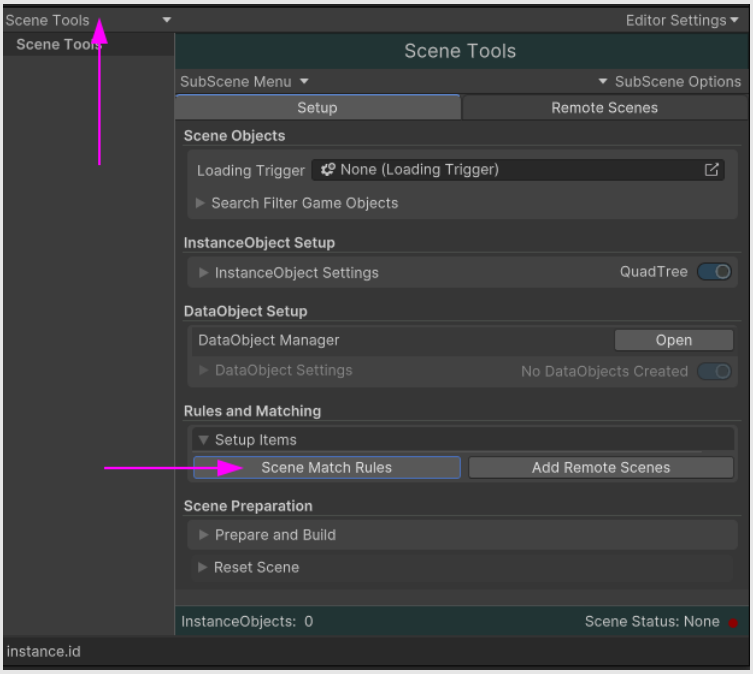
Add Scene Search Filters • **Right-click on (one or many) empty top-level G ameObject in the scene hierarchy** • **Select ProStream-> Add Scene Search Filter**	
This will add a new SceneSearchFilter Component to the GameObject and begin indexing the child P refab GameObjects.	

Configure Matching Rules

RuleEngine Details (RuleEngine)

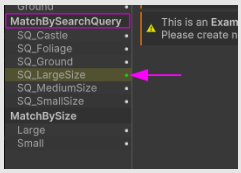
Open Match Rules Menu

- **Press the "Scene Match Rules" button**
- **Select the "Rule Editor" option from dropdown menu**



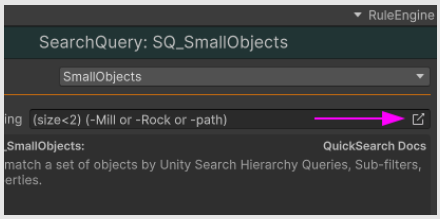
Enabled Example Rules

- **Locate the example items under the MatchBySearchQuery rules category**
- **Double click rule to enable**



You can also verify the currently enabled rules by viewing them directly in the SceneConnector's inspector under the MatchRules `List<RuleEngine>` Object

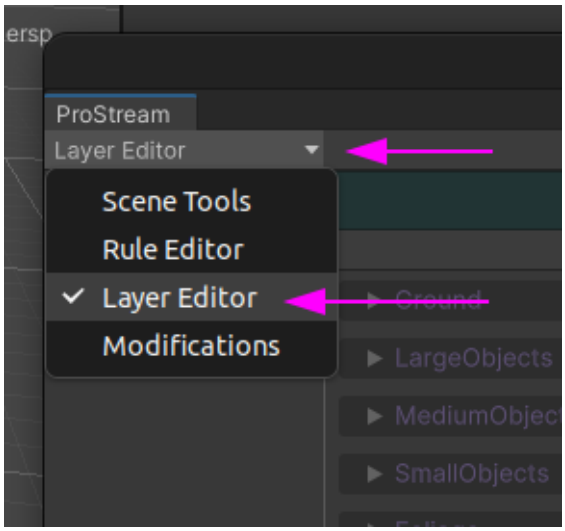
You can edit and preview the results of the query by clicking icon on the right side of the query input box.



Streaming Layers

StreamingLayers Details (Streaming Layers)

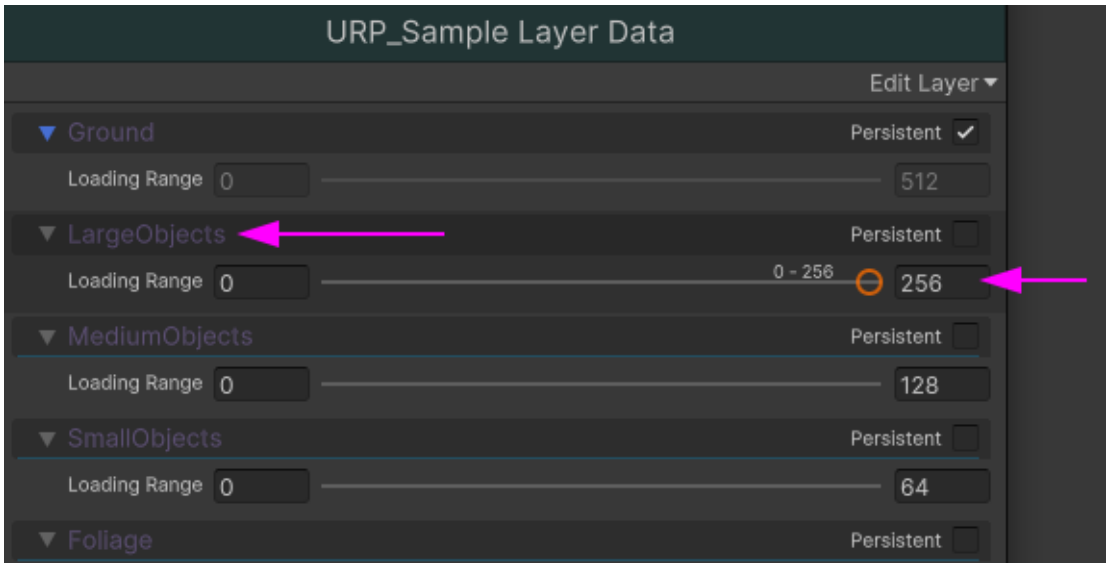
Access Streaming Layers



pst_3_menuLayerEditor_image_psTutorial.png

Streaming Distance Editing

In the **Layer Editor** menu, you can adjust the distance in which each SubScene layer will load into the scene.



pst_3_layerDistanceLoad_image_psTutorial.png

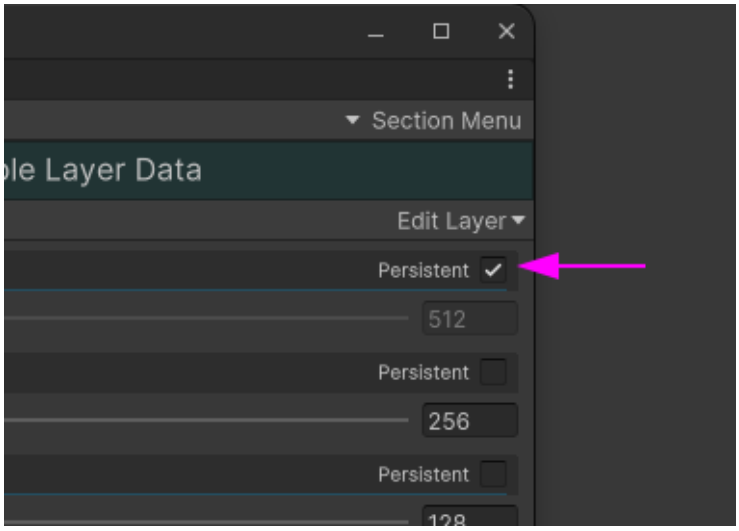
In the image above, you can see the **LargeObject** layer has a loading range of **0-256.** This means that if the **Loading Trigger** (Player, Camera, etc) is 256 units of distance away or less, this SubScene Layer will be loaded.

> The higher the number, the farther away the Loading Trigger can be for the SubScene layer to load. The lower the number, the closer the Loading Trigger must be for the SubScene Layer to load.

The numeric value in the right-side input box is the current maximum value, and the right-side slider is the current value. Hovering over the slider handle shows what the current distance setting is, and can be dragged left/right to be adjusted.

To raise the value above the current maximum, just input a new value into the right-side input box.

Persistent Layers



pst_3_layerEditorPersistent_image_psTutorial.png

Layers marked as **Persistent** are always loaded, and thus, do not need to have a range set.

See also

Additional Info

Importance of Prefabs (Importance Of Prefabs)

Engine Details

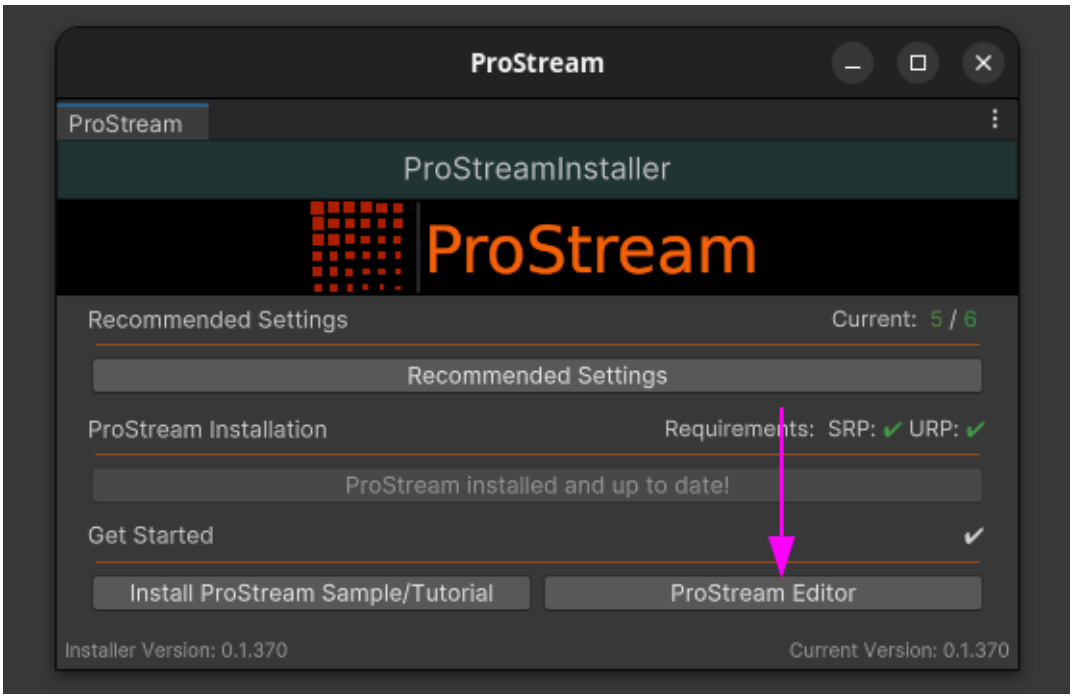
RuleEngine Details (RuleEngine)

ProStream Editor

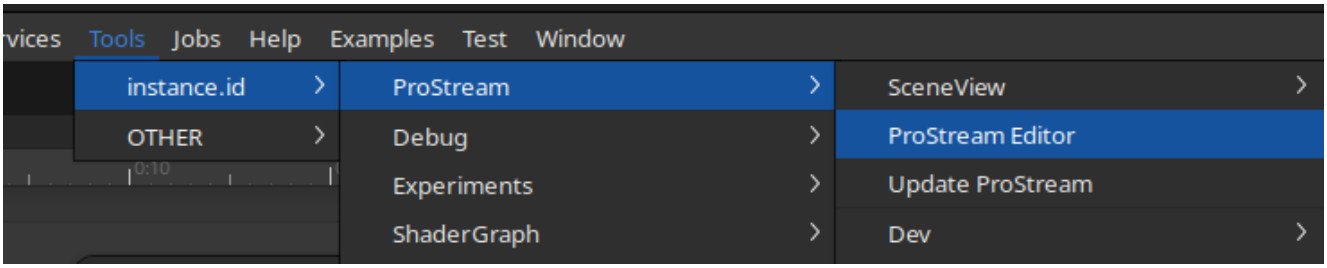
Access ProStream Editor

To access the ProStream Editor, you can use the following methods:

1. Press `Shift+Alt+P` shortcut to open the ProStream Editor
2. Press the "ProStream Editor" button in the installer

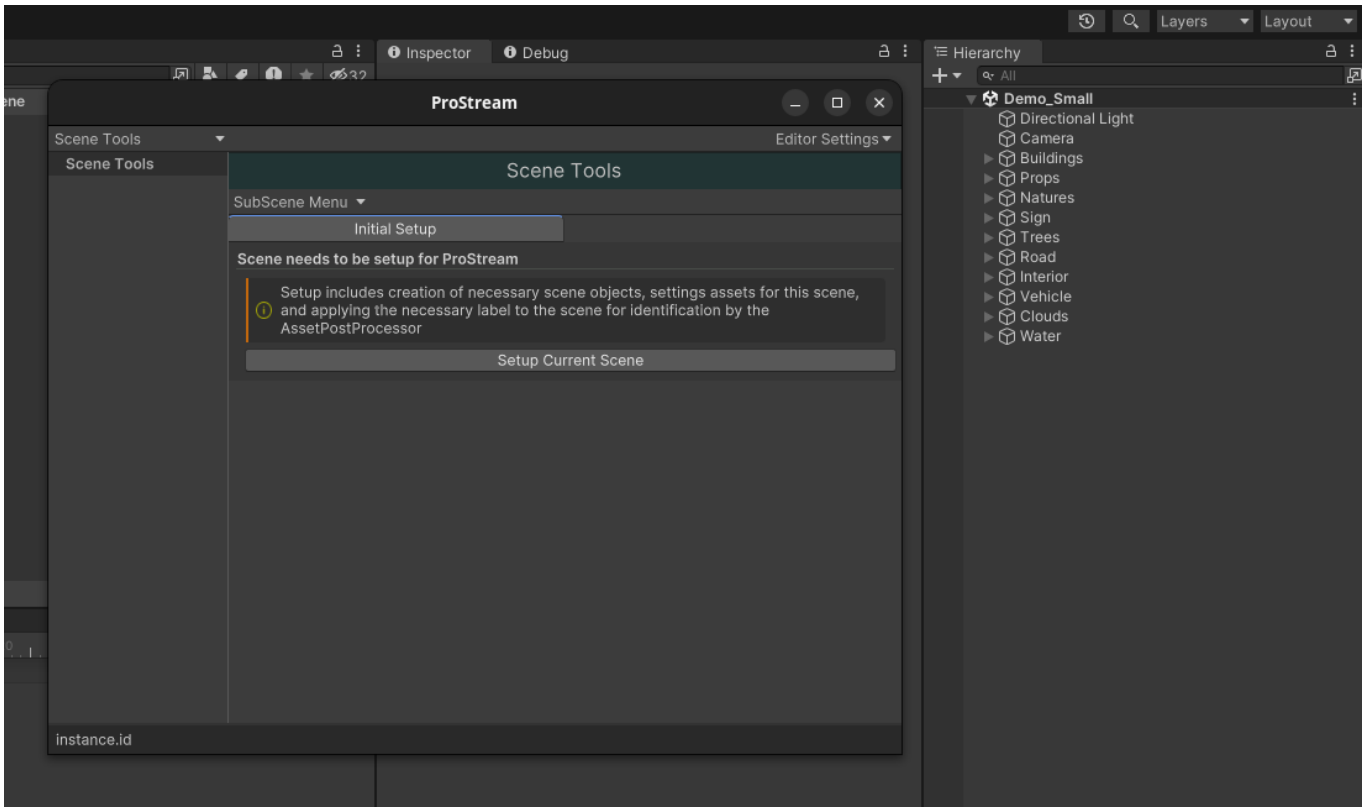


3. Go to the following Menu path `Tools`-> `instance.id`-> `ProStream`

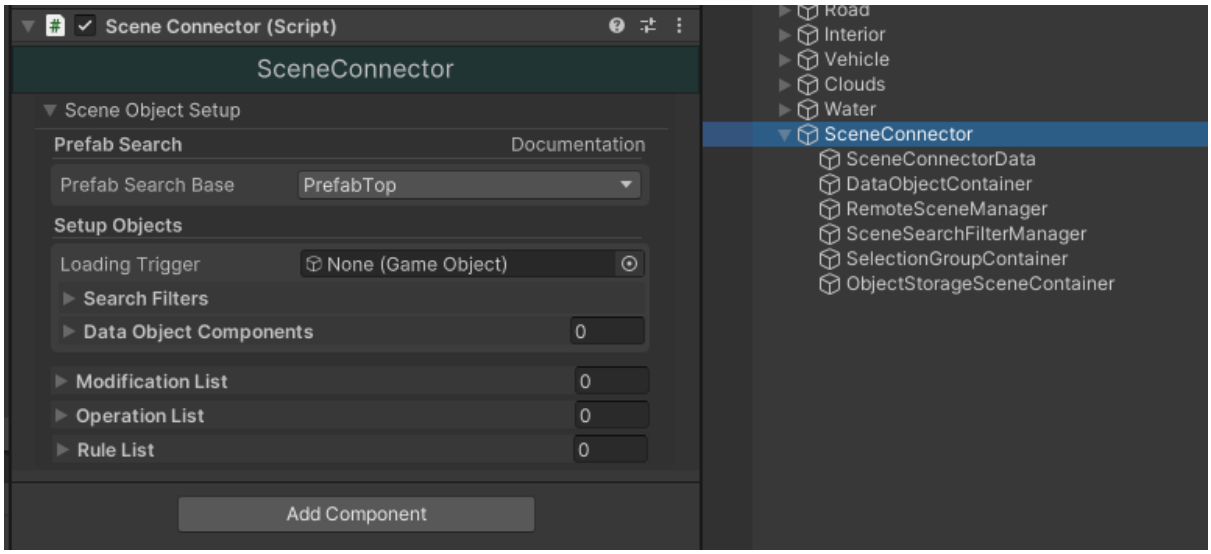


Setup Scene

Press the "Setup Current Scene" button in the ProStream Editor to add the necessary components to the scene.



This will add SceneConnector (SceneConnector) the following components to the scene:

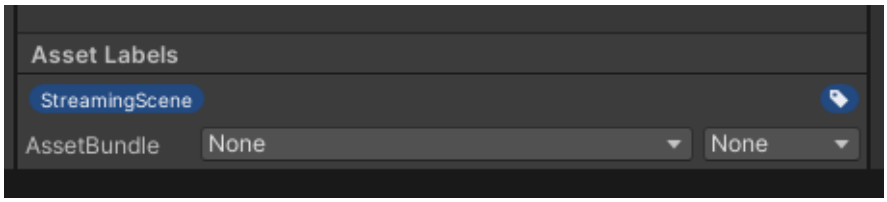


scene_connector_object.png

```
SceneConnector
  SceneConnectorData
  RemoteSceneManager
  DataObjectContainer
  SceneSearchFilterManager
```

```
    SelectionGroupContainer
    ObjectStorageSceneContainer
```

A new label will be added to the scene to indicate that it is a streaming scene.



streaming_scene_label.png

SceneConnector

The SceneConnector is a component that is used as the main connecting point for the ProStream systems.

Loading Trigger

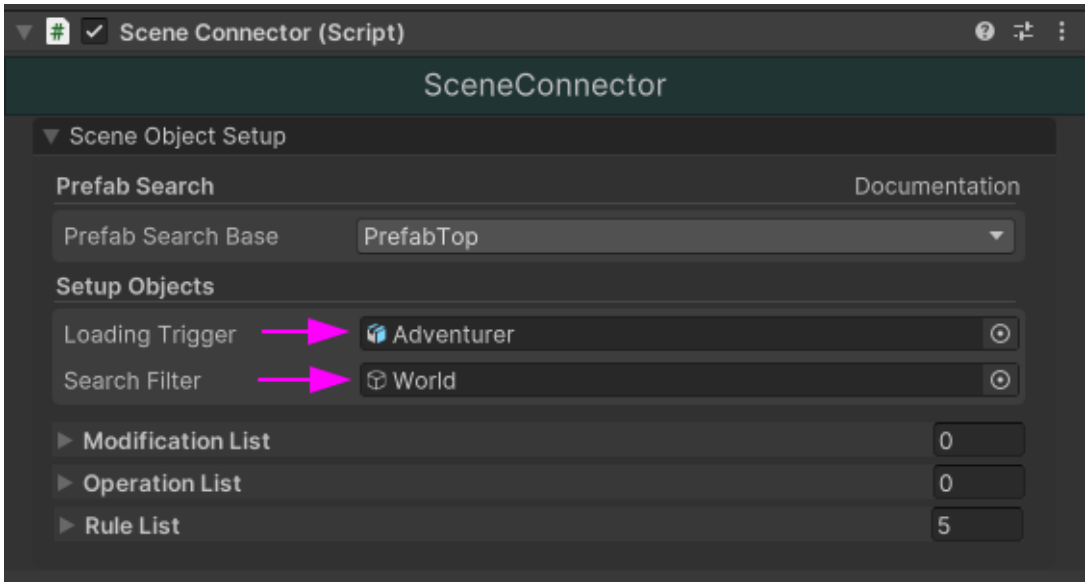
The Loading Trigger position value is used to determine when a SubScene (and layers) loads and unloads from the world. While it is not necessary for the **Loading Trigger** to be a Player Character (it can be a Camera, NPC Object, etc) it is necessary for the loading system to operate properly.

> ⚠ **If this field is left empty, when the system starts, it will attempt to locate and use Camera.main as the Loading Trigger.**

Search Filter

When the Search Filter field is populated, and the RuleEngine Match system runs, matches are restricted to child GameObjects of the Search Filter GameObject.

> ⚠ **If no GameObjects are assigned to the Search Filter list, the system will attempt to match against all GameObjects in the hierarchy.**
>
> This is typically undesired as this could include Cameras, Lighting, Light Probes, etc, and any number of things that you might wish to keep in the main parent scene.


Scene Connector (Script)
SceneConnector
Scene Object Setup
Prefab Search
Documentation
Prefab Search Base
PrefabTop
Setup Objects
Loading Trigger
Adventurer
Search Filter
World
Modification List
0
Operation List
0
Rule List
5

Image

SceneSearchFilter

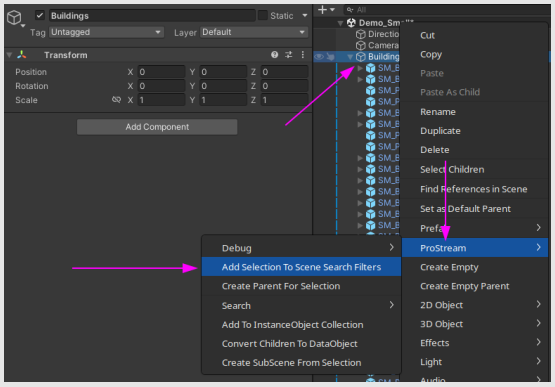
Add Scene Search Filters

> ⚠ It is highly recommended to use SceneSearchFilters, as they help improve setup performance by designating and limit the search scope of the RuleEngine Match system to the child Objects of the SceneSearchFilter GameObject.

> ℹ All GameObjects must be Prefabs to be compatible with ProStream systems. The Prefab system is used to differentiate one GameObject Hierarchy from another. on-Prefab GameObjects are skipped by the matching systems.

Add Scene Search Filters

- **Right-click on (one or many) empty top-level GameObject in the scene hierarchy**
- **Select ProStream-> Add Scene Search Filter**



This will add a new SceneSearchFilter Component to the GameObject and begin indexing the child Prefab GameObjects.

RuleEngine

Overview

The RuleEngine is a system that allows for the creation and use of custom rules to match GameObjects with Streaming Layers.

Rule Providers

The RuleEngine system is designed to be extensible enables the use of any number of underlying providers.

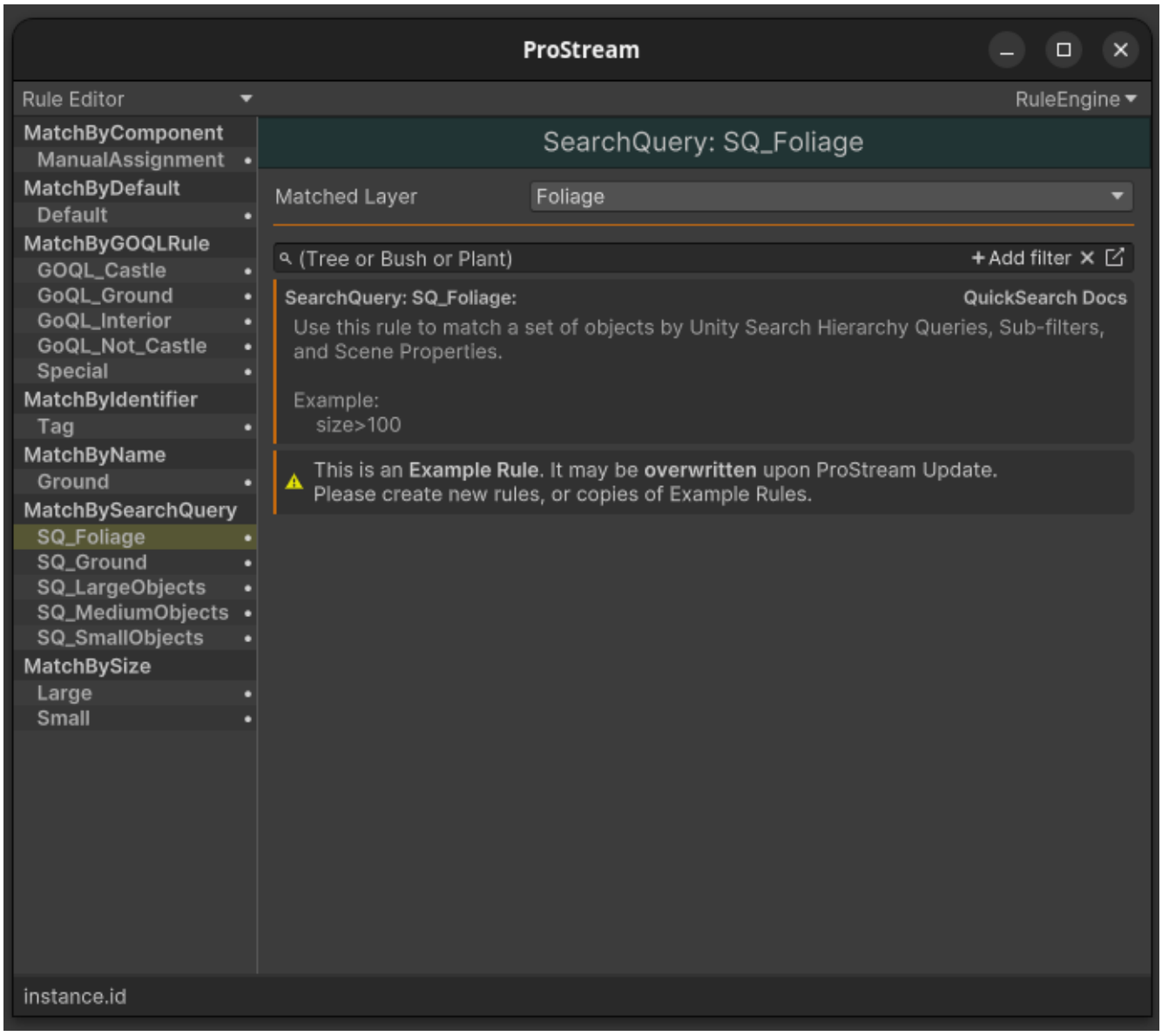
Current Providers	• **Unity Search Query system (MatchBySearchQuery)** • **GameObject Query Language (MatchByGoQL)** • **Unity Identification Systems (Layer, Tag, Label)** • **Custom Matching via C# code**

Example Provider: MatchByName

```
public class MatchByName : RuleEngine
{
    // --| Base Class Overrides -------------
    public override string Title => "Match By Name";
    public override string RuleName => FriendlyName(name,
typeof(MatchByName));

    // --| Rule Members ---------------------
    private enum MatchType { Contains, DoesNotContain }

    [SerializeField] private MatchType matchType;
    [SerializeField] private string nameMatchString;

    public override int CheckRule(Transform obj)
    {
        return matchType switch
        {
```

```
            MatchType.Contains when obj.name.Contains(nameMatchString)
=> Matched,
            MatchType.DoesNotContain when
!obj.name.Contains(nameMatchString) => Matched,

            _
=> Unmatched
        };
    }
}
```

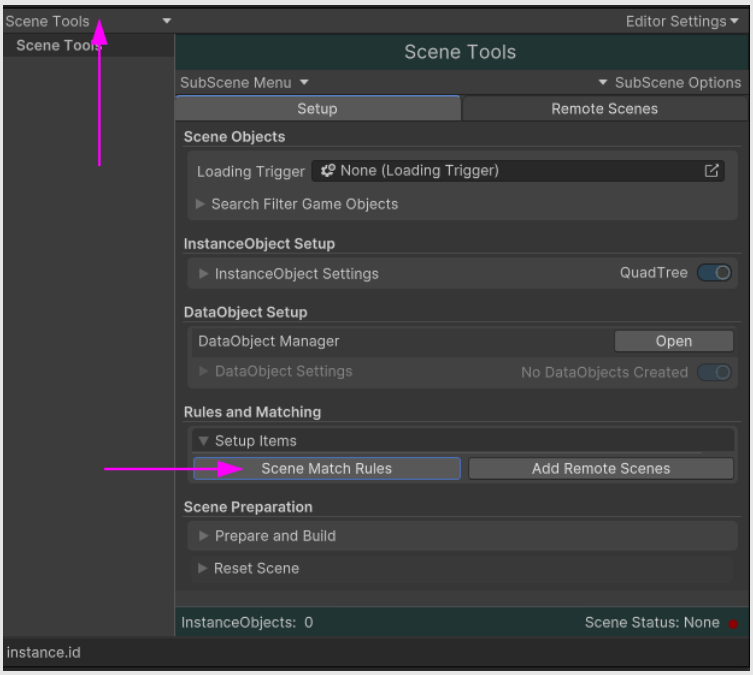
Rule Editor



Configure Matching Rules

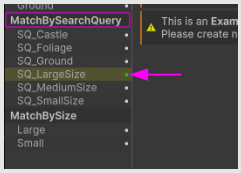
Open Match Rules Menu

- **Press the "Scene Match Rules" button**
- **Select the "Rule Editor" option from dropdown menu**



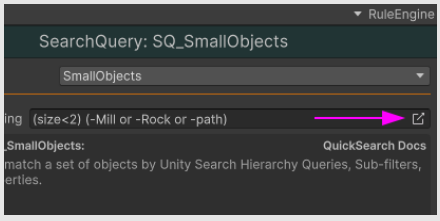
Enabled Example Rules

- **Locate the example items under the MatchBySearchQuery rules category**
- **Double click rule to enable**



You can also verify the currently enabled rules by viewing them directly in the SceneConnector's inspector under the MatchRules `List<RuleEngine>` Object

You can edit and preview the results of the query by clicking icon on the right side of the query input box.



ModificationEngine

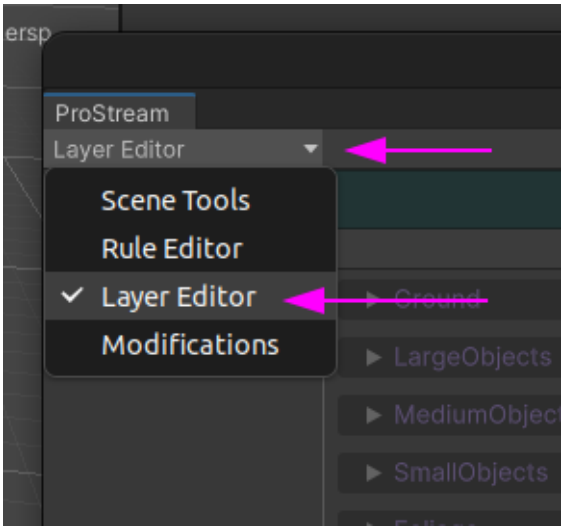
Start typing here...

OperationEngine

Start typing here...

Streaming Layers

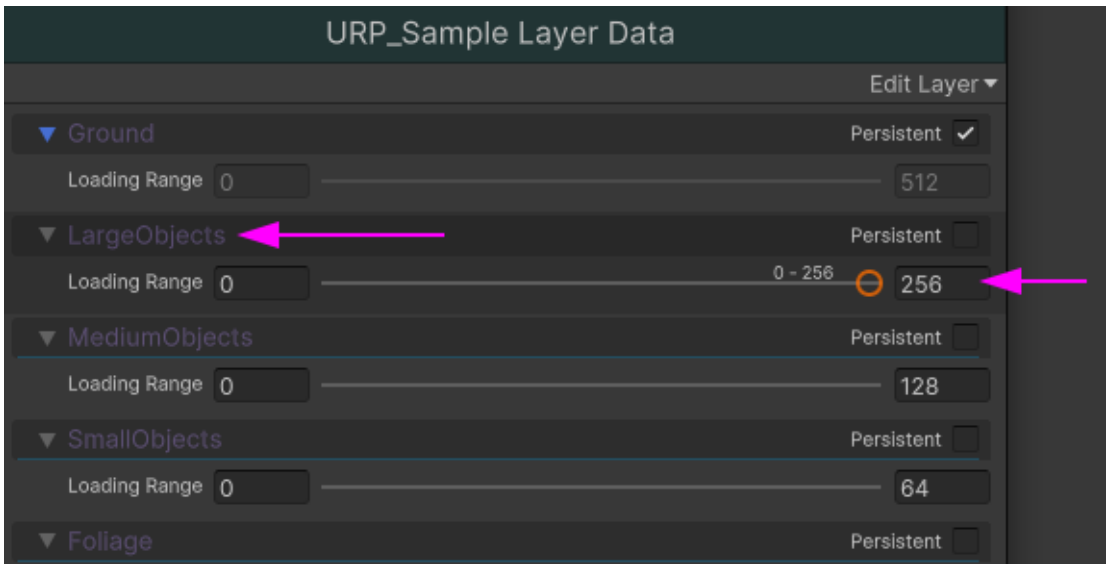
Access Streaming Layers



pst_3_menuLayerEditor_image_psTutorial.png

Streaming Distance Editing

In the **Layer Editor** menu, you can adjust the distance in which each SubScene layer will load into the scene.



pst_3_layerDistanceLoad_image_psTutorial.png

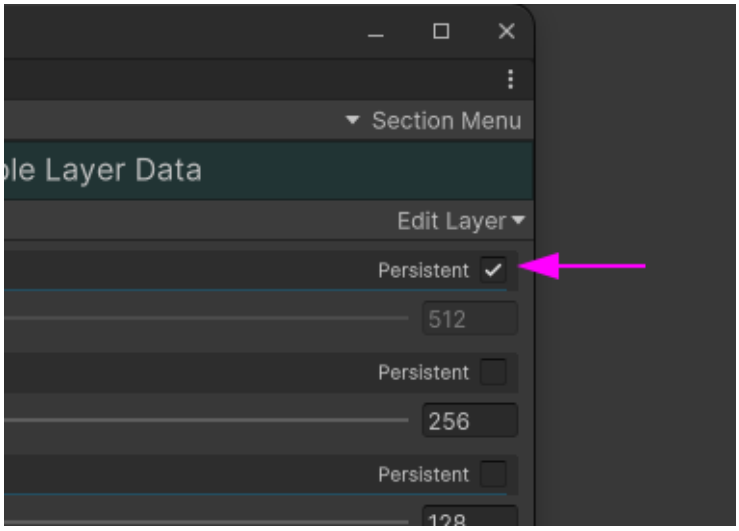
In the image above, you can see the **LargeObject** layer has a loading range of **0-256.** This means that if the **Loading Trigger** (Player, Camera, etc) is 256 units of distance away or less, this SubScene Layer will be loaded.

> The higher the number, the farther away the Loading Trigger can be for the SubScene layer to load. The lower the number, the closer the Loading Trigger must be for the SubScene Layer to load.

The numeric value in the right-side input box is the current maximum value, and the right-side slider is the current value. Hovering over the slider handle shows what the current distance setting is, and can be dragged left/right to be adjusted.

To raise the value above the current maximum, just input a new value into the right-side input box.

Persistent Layers



pst_3_layerEditorPersistent_image_psTutorial.png

Layers marked as **Persistent** are always loaded, and thus, do not need to have a range set.

Process SubScenes

Start typing here...

Position Calculation

Prerequisites

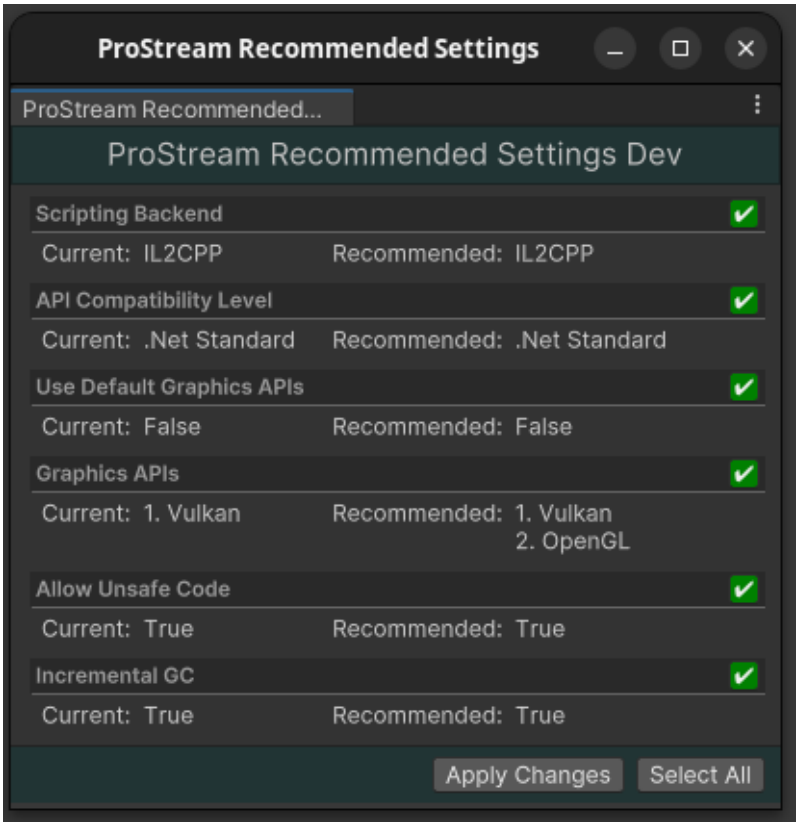
Before you start, make sure that you have the following:

Requirements and Notes
• Unity 2022.3.0f1 or later (2022.3.21f1 is recommended) • Scriptable Render Pipeline (SRP) installed and configured (URP or HDRP) • Scene GameObjects which are not Prefabs (Importance Of Prefabs) are ignored • (Recommended) Scene Prefabs are children of an Empty Root-Level Parent GameObject

Recommended Project Settings

To take full advantage of ProStream and the Data Oriented Technology Stack, it is recommended to use the following settings:

Setting	Value
Scripting Backend	IL2CPP (https://docs.unity3d.com/Manual/IL2CPP.html)
API Compatibility Level	.Net Standard 2.1 (https://docs.unity3d.com/Manual/dotnetProfileSupport.html)
Graphics APIs	Vulkan, Metal, and DX11/12 (https://docs.unity3d.com/Packages/com.unity.entities.graphics@1.2/manual/requirements-and-compatibility.html)
Incremental GC (Garbage Collection) (https://docs.unity3d.com/Manual/performance-incremental-garbage-collection.html)	Enabled



ps_recommended_settings_1.png

Importance Of Prefabs

The Prefab Asset acts as a template from which you can create new Prefab instances in the Scene.

ProStream uses Prefab system to differentiate one GameObject Hierarchy from another.

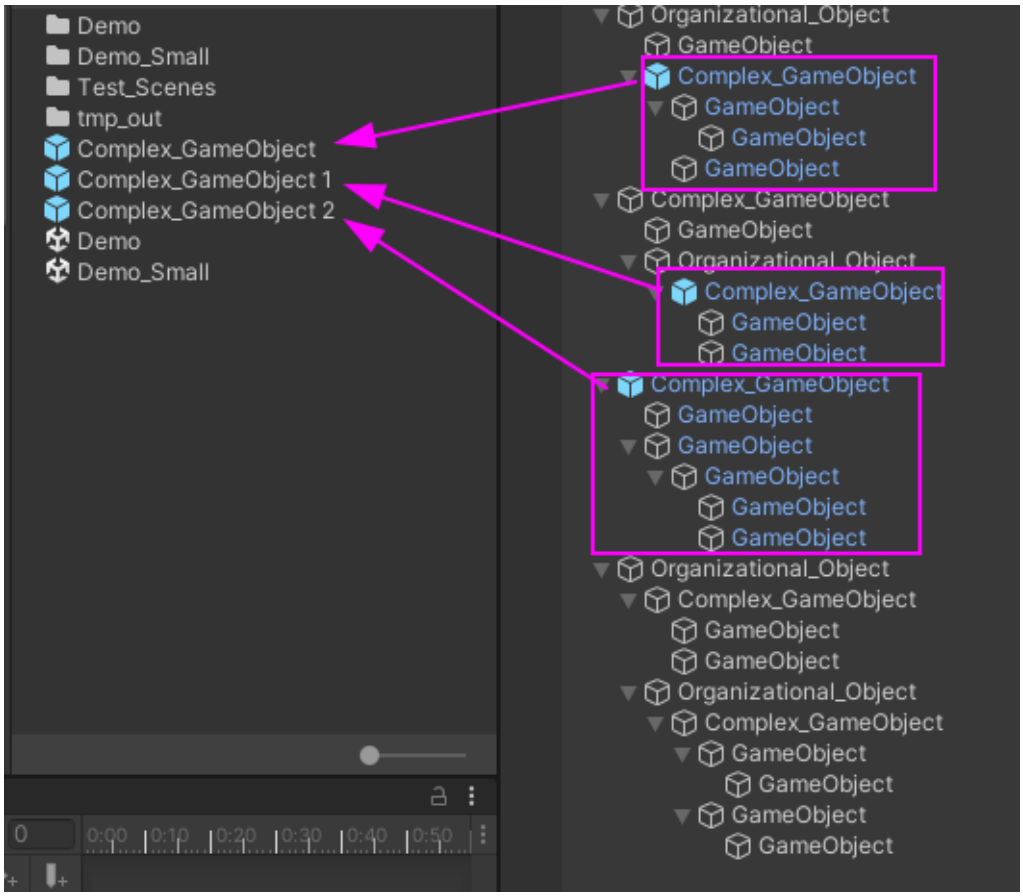
> All GameObjects must be Prefabs to be compatible with ProStream systems. Non-Prefab GameObjects are skipped by the matching systems.

As illustrated below, without Prefabs the only distinction between an `Organizational_Object` and a `Complex_GameObject` is the GameObjects name. While this might be sufficient to visually distinguish between the two, programmatically, they are the same.

Creating a Prefab removes the ambiguity and outlines a concrete blueprint of what a particular GameObject is and what it should contain.

With this distinction, the ProStream systems can accurately identify and match GameObjects

in the scene.



See also

Additional Info

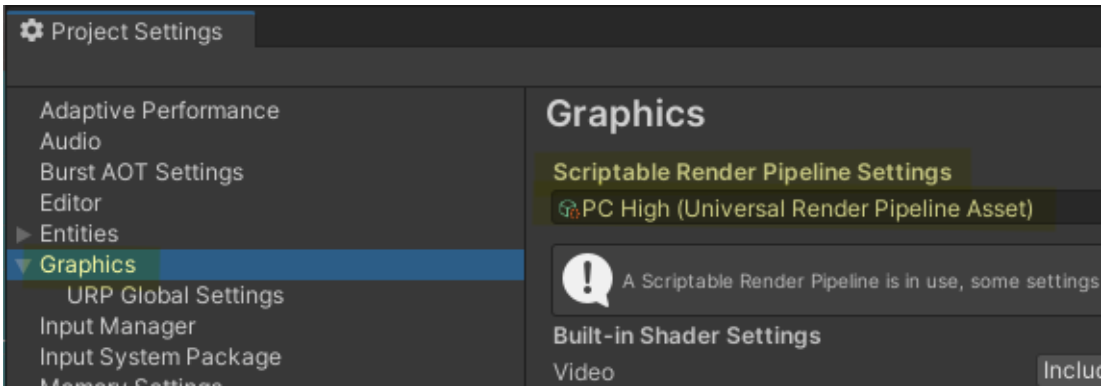
Unity Prefabs (https://docs.unity3d.com/Manual/Prefabs.html)

Build/Runtime issues

Common and possible issues related to the Build process, or Runtime issues.

No Objects In Game

If you perform a build and there are no objects in the game, it is possible that the SRP asset is not assigned . Ensure that a SRP asset is assigned: `Project Settings`-> `Quality`-> `Scriptable Render Pipeline Settings`



No Objects In Game

See also

Change Log

Change Log