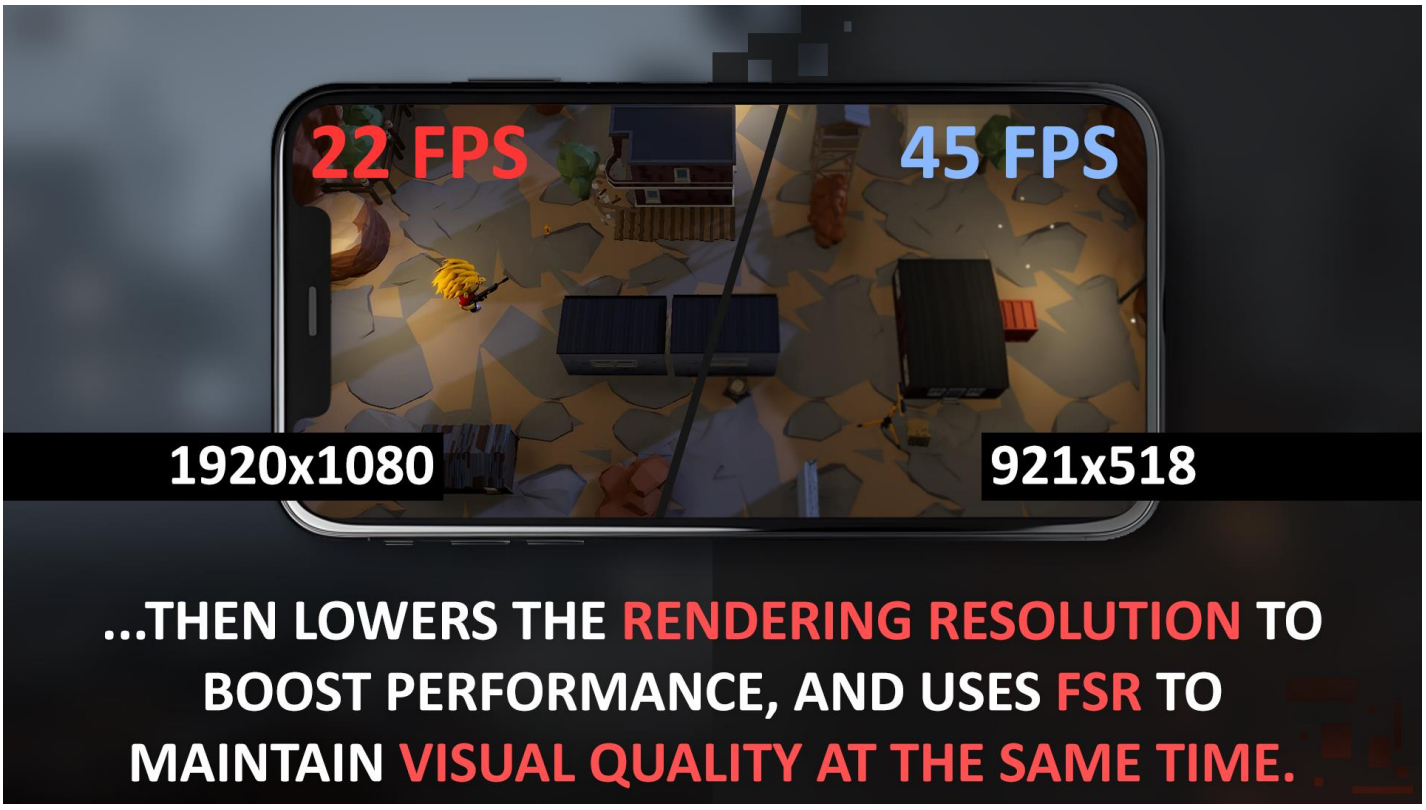
PIXELCRAFT: Dynamic Resolution for URP (Documentation)

- JiRo Entertainment -

SUMMARY

PIXELCRAFT: Dynamic Resolution for URP is a tool that handles dynamic resolution for your URP games, and unlike traditional solutions, this one uses FSR to reconstruct the missing pixels.



DESCRIPTION

🚀 Introducing PIXELCRAFT: Dynamic Resolution for URP 🚀

In the realm of game development, maintaining optimal performance while preserving visual quality is a perpetual challenge. PIXELCRAFT addresses this by seamlessly integrating a powerful *DYNAMIC RESOLUTION* system.

Key Features:

🌐 Performance Monitoring: PIXELCRAFT continually monitors your game's performance in the background. Should the frame rate drop below the developer-specified target, it dynamically adjusts the rendering resolution to restore intended performance levels.

💡 FSR Integration for Visual Restoration: Beyond standard dynamic resolution, PIXELCRAFT incorporates AMD's FidelityFX Super Resolution (FSR). This feature reconstructs missing pixels when the rendering resolution is below the native resolution, ensuring a native frame without compromising performance during demanding scenarios.

🚀 Balanced Performance and Visual Quality: PIXELCRAFT strikes a harmonious balance between performance and visual quality. It eliminates the need for trade-offs, providing developers with a comprehensive solution.

🎮 Developer Control: PIXELCRAFT empowers developers with granular control. Fine-tune dynamic resolution parameters to align with the specific requirements of your game.

🌐 Volume System for Targeted Optimization: PIXELCRAFT introduces a sophisticated Volume system, allowing developers to strategically define specific areas within their game world. Each encapsulated Volume serves as a zone of targeted optimization, offering precise control over performance parameters and ensuring a nuanced gaming experience.

CREDIT

1- One of the Demo scenes used is made with Synty's Western Frontier pack.
2- The scene used to showcase the performance of Smart Dynamic Resolution is a tweaked version of AFS- Seaside Environment.

SUPPORT – QUESTIONS - NEWS

Hop on our Discord server. Follow on Twitter/X.

TECHNICAL DETAILS

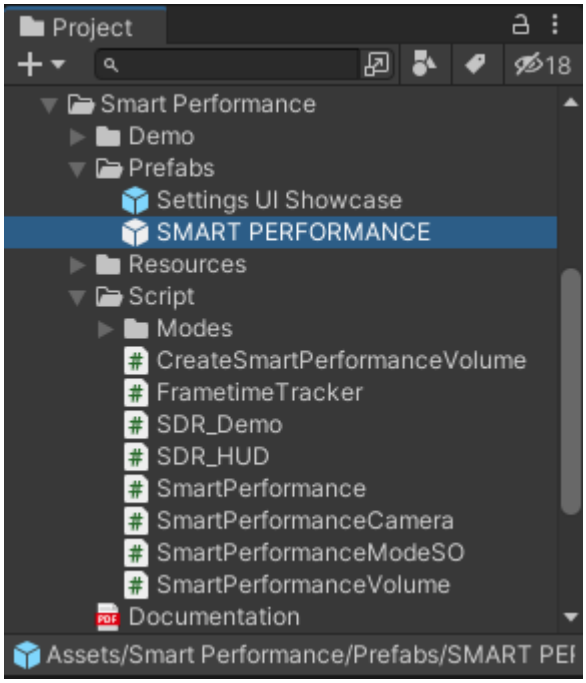
Craft the ultimate URP gameplay experience with intelligent resolution scaling:

- Seamless Frame Rate: Maintain your desired FPS target (30, 45, or 60) with dynamic adjustments to rendering resolution.
- Precision Control: Choose from multiple FPS calculation methods for responsiveness or stability based on your game's needs.
- Quality Presets: Define customizable ranges for scaling resolution within three presets: Quality, Balanced, and Performance.
- Frame Reconstruction: Choose between a simple Bilinear reconstruction or Enhance visuals at lower resolutions using FSR 1.0.
- Volume Overrides: Fine-tune specific areas of your game world with custom overrides for Quality Presets and FPS Targets. When the Camera used for Dynamic Resolution enters a Volume, this tool would then use FPS Target and Quality Mode of that Volume, this is especially helpful to fine tune your game's experience by targeting 30FPS in heavy areas and 45FPS or 60FPS in open areas. (90FPS and 120FPS have been added in Update 1.2.0)
- More Advanced Tweaks: Control MSAA values, adjust FPS tracking update rate, and define resolution step increments for even finer control.

NOTE: Dynamic Resolution is less impactful when the CPU is the bottleneck, it only works best when the GPU is the bottleneck.

SETUP

1. To get start, simply drag “SMART PERFORMANCE” Prefab to your Hierarchy:



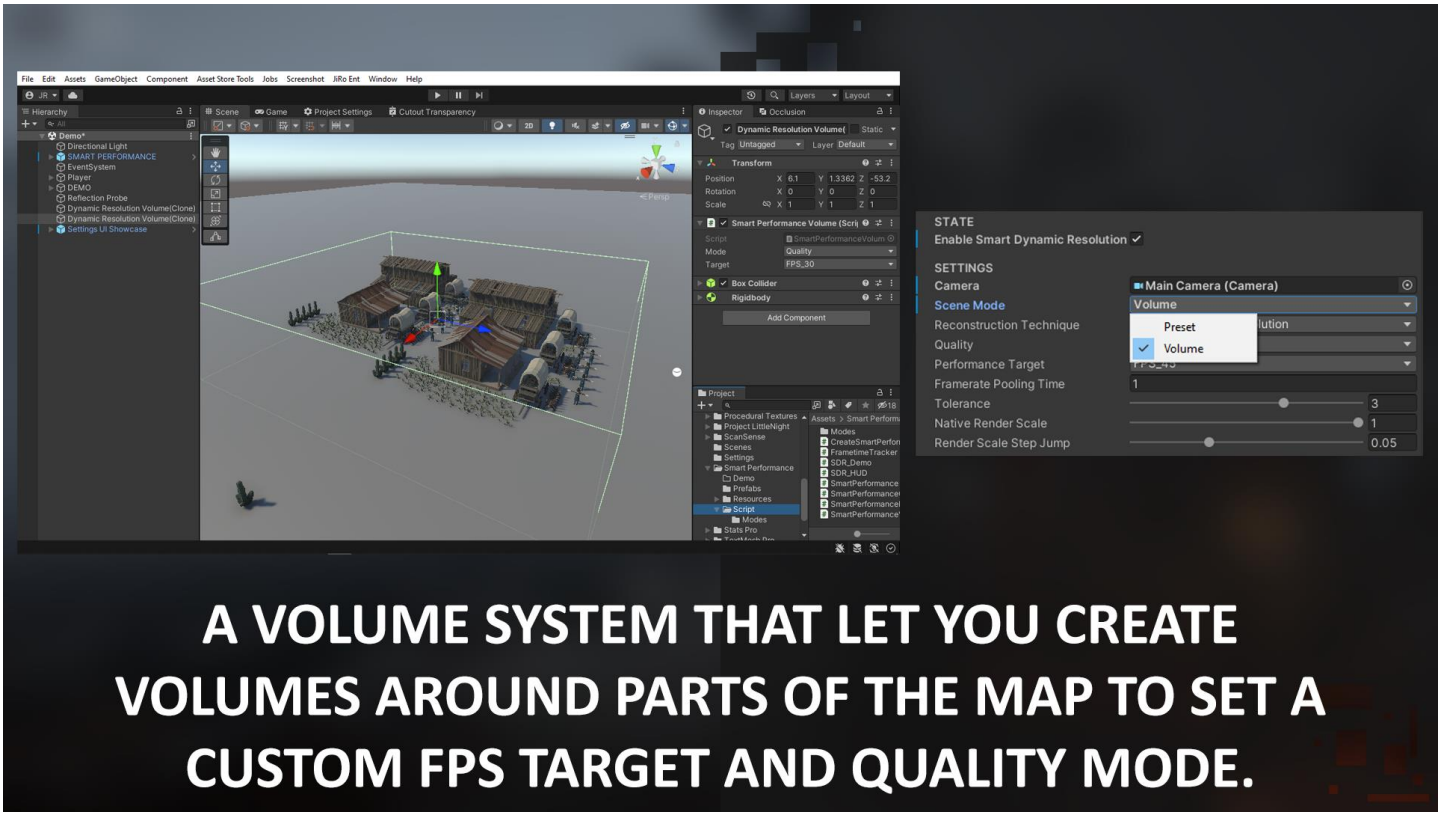
2. And then simply tick "Enable Smart Dynamic Resolution", now when you enter Play mode Dynamic Resolution would be working as expected targeting the set Performance Target.

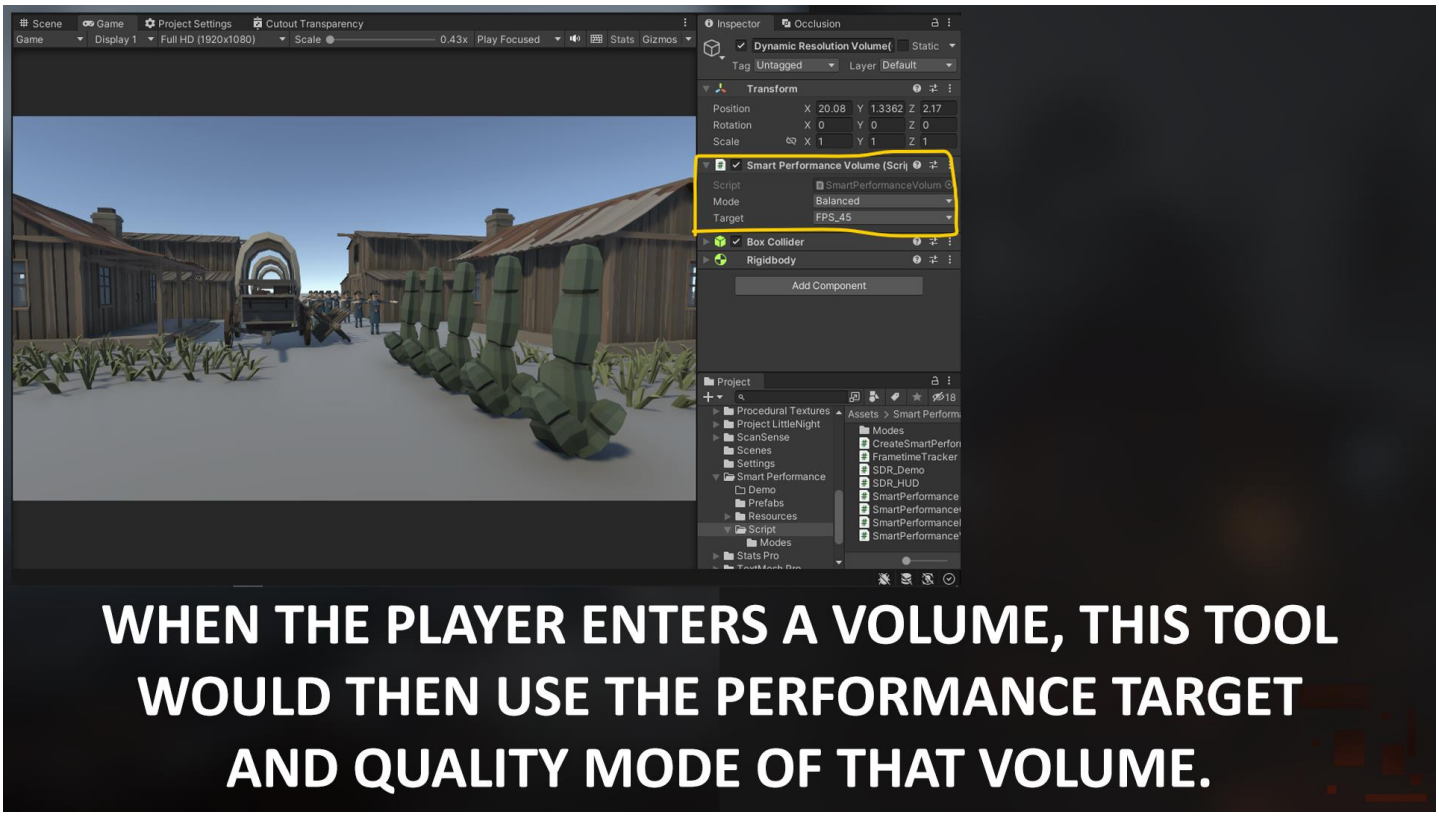
PARAMETERS

VARIABLE	FUNCTION
Volume Camera (Updated in 1.1.0)	This field is reserved for the Camera meant to operate in the Volume mode. You don't need to assign a Camera if you're using the Preset mode.
Scene Mode	This lets you choose between 2 options: - Preset: If "Preset" is picked this tool would then use FPS Target and Quality Mode of the prefab "SMART PERFORMANCE" throughout the whole map. - Volume: If "Volume" is picked this tool would then use FPS Target and Quality Mode of the Volume it is inside of, when the Camera is not inside any Volume it would then defaults to the Preset FPS Target and Quality Mode.
Reconstruction Technique	This lets you choose between 2 options: - Bilinear: This uses a Bilinear algorithm when reconstructing from low resolution frames to native frames. - FidelityFX Super Resolution (FSR): Thise uses FSR tech to reconstruct from low resolution frames to native frames, this option looks better than Bilinear.
Quality	This lets you choose between 3 options: - Performance - Balanced - Quality Each option has a specific range of rendering resolution, so for example when you pick "Performance" this system would then be allowed to render as low as 0.33 of the full resolution frame, if you pick "Quality" the minimum rendering resolution would then be 0.6 of the full resolution frame. Each option also features a different FPS Sensitivity, for each there's a different algorithm of calculation FPS of the game "Low" being more resilient towards FPS fluctuations and "High" being less resilient towards FPS fluctuations.
Performance Target	This lets you choose between 3 options: - FPS_30 - FPS_45 - FPS_60 - FPS_90 (New in Update 1.2.0) - FPS_120 (New in Update 1.2.0) Each option defines what FPS target Dynamic Resolution would aim at maintaining.
Framerate Pooling Time	Time in seconds in which the FPS is being reported to the tool to work with, by default this is set to 1 second, meaning the tool would adjust its calculation every 1 second.
Tolerance	The number of frames per second allowed to be above or under the Target FPS without triggering a change in dynamic resolution. By default, this variable is set to 3, meaning if the target is 60FPS then this tool wouldn't change the dynamic resolution when the FPS of the game is between 60FPS – 3 and 60FPS + 3,

	meaning a range of 57FPS-63FPS. This option is made to ignore small FPS fluctuations and create a gray zone where dynamic resolution can be stable.
Native Render Scale	This is the Render Scale value your game will defaults to when Smart Dynamic Resolution is disabled.
Render Scale Step Jump	This is the smallest amount of adjustment being made to the render scale when the resolution is being dynamically adjusted.

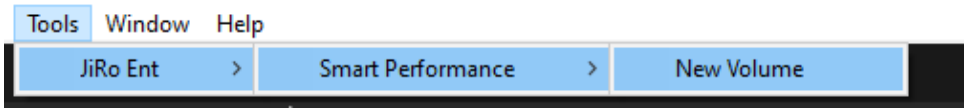
VOLUME SYSTEM



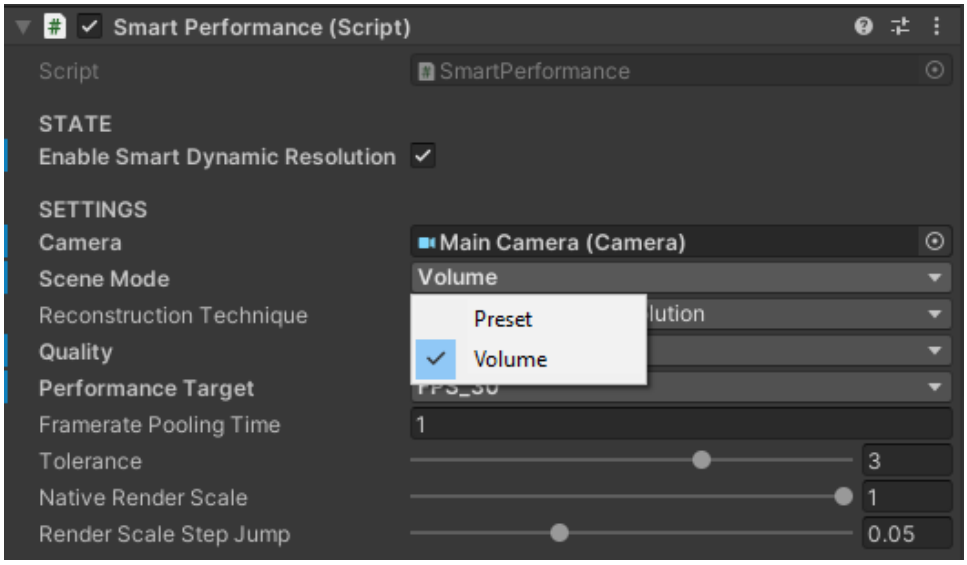


PIXELCRAFT introduces a groundbreaking Volume System, providing developers with a powerful tool to precisely tailor the optimization of their game environments. With this sophisticated feature, developers can strategically define specific areas within their game world, encapsulated by Volumes. These Volumes serve as zones of targeted optimization, enabling fine-grained control over two crucial parameters—Quality Presets and FPS Targets. This granular control empowers developers to curate diverse gaming experiences throughout their virtual landscapes. Imagine designating high-intensity battle zones with a Performance preset and a 60FPS target, while urban centers or intricate towns utilize a Balanced preset with a 30FPS target. The Volume System offers unparalleled flexibility, allowing developers to optimize each segment of their game world individually and control the experience throughout the map.

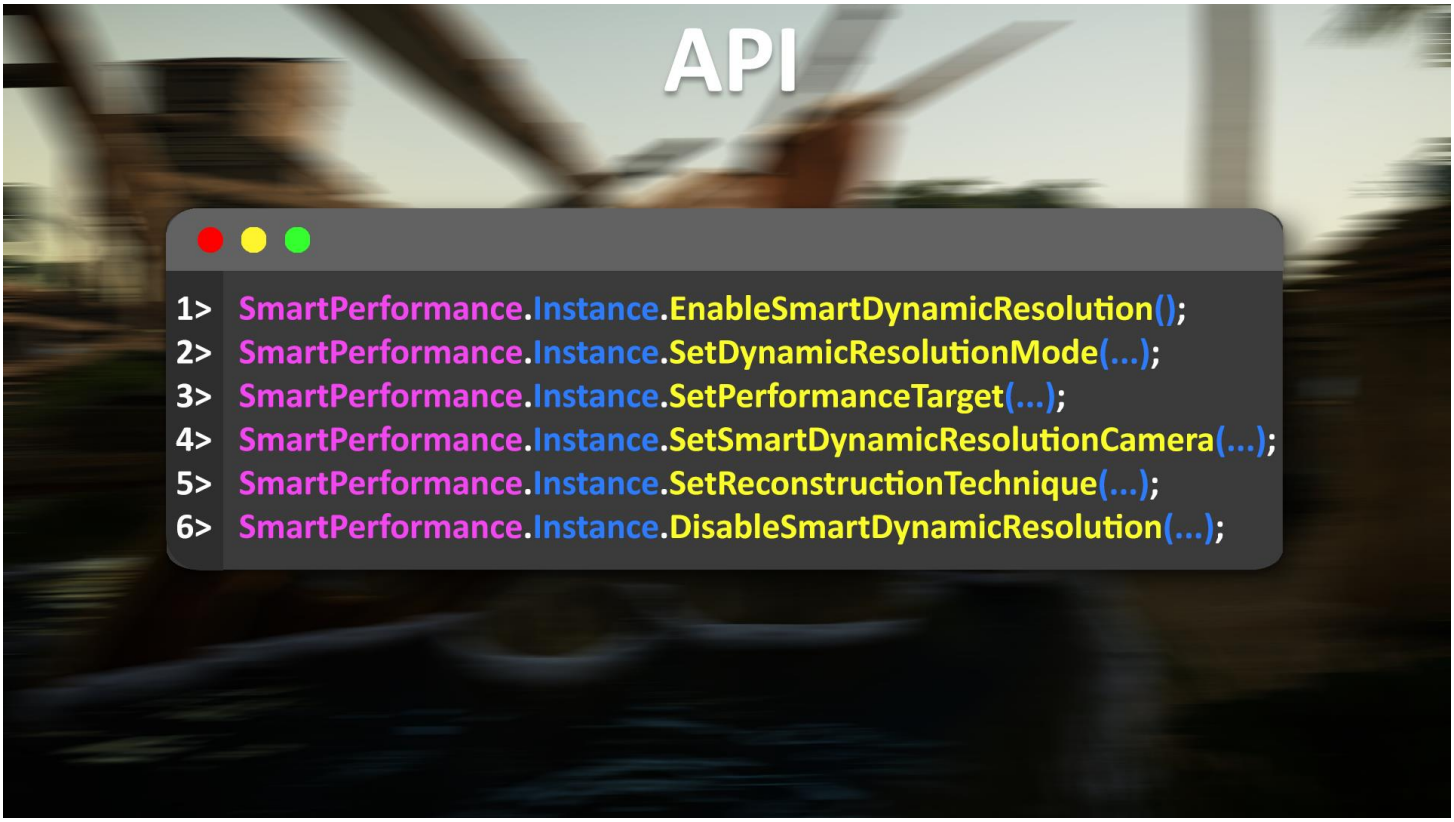
- To Add a new Volume, go to Tools/JiRo Ent/Smart Performance/New Volume:



- The Volume would be created exactly where your Scene view camera is.
- After filling your map with Volumes, go to “SMART PERFORMANCE” prefab that you dragged to your Hierarchy, and set its “Scene Mode” variable to “Volume” instead of “Preset”:



API



- First, make sure to implement the namespace "`JiRoEnt.Utils.SmartDynamicResolution` " using this line:

```
using JiRoEnt.Utils.SmartDynamicResolution;
```
This namespace includes the necessary functions that enable Dynamic Resolution.

- To enable Dynamic Resolution, simply call this function:
```
SmartPerformance.Instance.EnableSmartDynamicResolution();
```

- To set a Dynamic Resolution Mode, call this function:
```
SmartPerformance.Instance.SetSmartDynamicResolutionMode(SmartPerformanceQuality quality, PerformanceTarget fpsTarget);
```

IMPORTANT NOTE: This function should not be called when the Volume mode is selected. It only works when you're using Preset mode.

This function takes 2 parameters:

"SmartPerformanceQuality" which is defined with 3 members:
- Performance
- Balanced
- Quality

"PerformanceTarget" which is defined with 3 members:
- FPS_30
- FPS_45
- FPS_60

Example:
```
SmartPerformance.Instance.SetSmartDynamicResolutionMode(SmartPerformanceQuality.Balanced, PerformanceTarget.FPS_45);
```

This line sets the quality mode to "Balanced" and the Performance target to 45.

- To set a different FPS target call this function:
```
SmartPerformance.Instance.SetPerformanceTarget(PerformanceTarget fpsTarget);
```

This function takes 1 parameter:
"PerformanceTarget" which is defined with 3 members:
- FPS_30
- FPS_45
- FPS_60

- To set a Camera for the Volume System, call this function:
```
SmartPerformance.Instance.SetVolumeCamera(Camera cam); // New in Update 1.1.0
```

Note: This function needs to be called before enabling Dynamic Resolution in Volume mode, if you're in Preset mode you don't need to call this function.

- If you have a Camera that is spawned mid gameplay, make sure to call this function:
```
SmartPerformance.Instance.NotifyOfCameraChange();
```

- (New in Update 1.2.0) Added an overload to "NotifyOfCameraChange" which takes type "Camera" as a parameter:

```
SmartPerformance.Instance.NotifyOfCameraChange(Camera camera);
```

It is best to use this overload over the previous one for performance reasons. The previous overload cycles through all camera meanwhile you can specify the newly spawned Camera with this new overload by simply passing through the new Camera as a parameter.

- To change the reconstruction technique of the frame when the rendering resolution is below native resolution, call this line:

```
SmartPerformance.Instance.SetReconstructionTechnique(Reconstruction reconstruction);
```

This function takes 1 parameter:
"Reconstruction" which is defined with 2 members:
- Bilinear
- FidelityFxSuperResolutution

Bilinear is a simple but blurrier reconstruction technique, while FSR (FidelityFx Super Resolution) is a solution made by AMD to upscale low-resolution frames to native resolution without compromising performance or quality. FSR is what we recommend as the reconstruction technique.

Example:

```
SmartPerformance.Instance.SetReconstructionTechnique(Reconstruction.FidelidyFxSuperResolution);
```

- To disable Smart Dynamic Resolution, call this function:

```
SmartPerformance.Instance.DisableSmartDynamicResolution();
```

This function comes with 2 overloads:

```
1/
SmartPerformance.Instance.DisableSmartDynamicResolution();
2/
SmartPerformance.Instance.DisableSmartDynamicResolution(int msaaSampleCount);
```

The difference being you can specify which MSAA value to default to when Smart Dynamic Resolution is disabled. The 1st overload automatically defaults to the value that was present before Smart Dynamic Resolution was enabled, the 2nd overload is there in case you want to manually control it. This might be necessary because Smart Dynamic Resolution may change the MSAA value depending on which quality preset was active. The low Preset disables MSAA, the Medium preset also disables MSAA, the Quality preset set it to 2x MSAA for better visual quality.

BONUS (UI SETTINGS DEMO)

This Asset comes with a Demo prefab that showcases how Smart Dynamic Resolution can be used with a TMP Dropdown to implement different Dynamic Resolution Presets.