# SHADER CONTROL

**RELEASE NOTES**
**V.2.0 / DEC.2016**

# Index

**Introduction** 3

**Quick Start** 3

**About shader keywords** 6

**Support & Contact info** 6

## Introduction

Thank you for acquiring Shader Control!

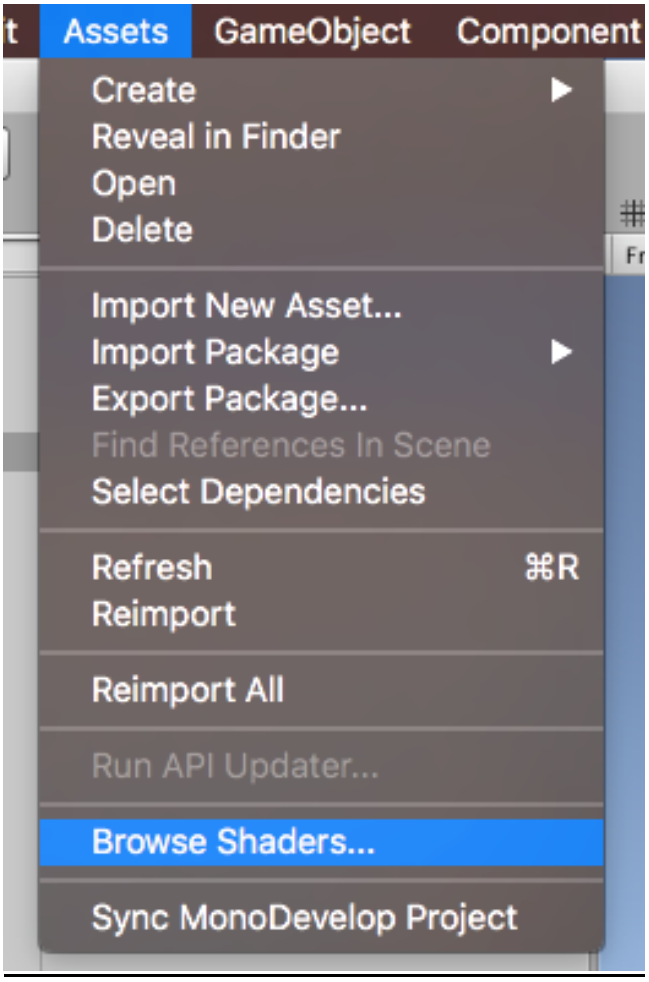
**Shader Control** allows you to:

- Quickly locate and list shaders in your project along keywords used.
- Disable/enable keywords per shader.
- Calculate shader variants based on shader modifications.
- Reduce build size
- Reduce compilation time

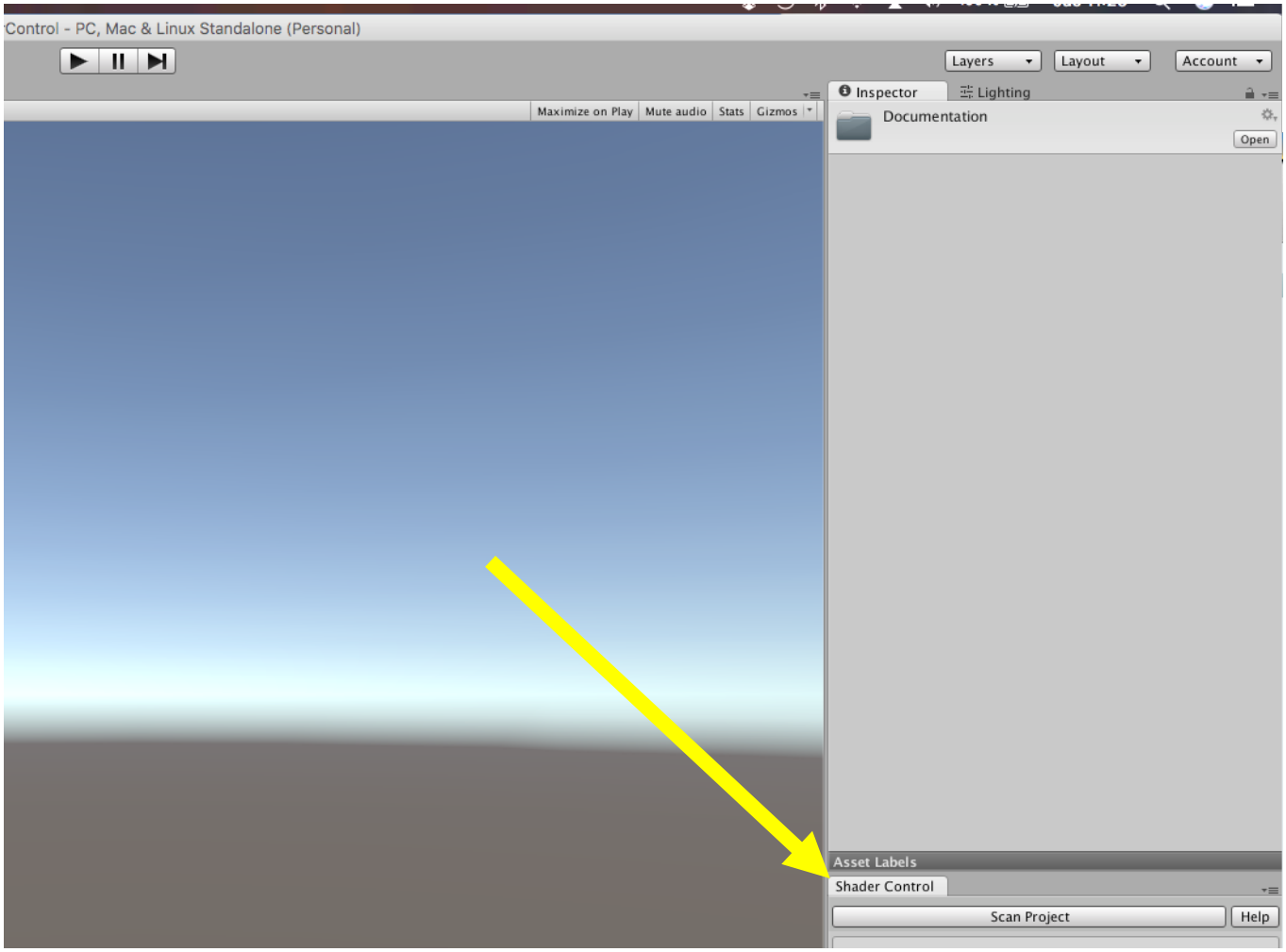
## Quick Start

1. Import the asset into your project

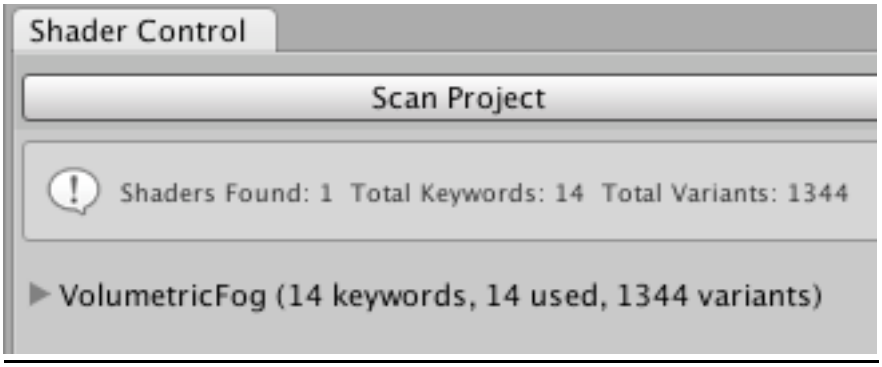
2. Go to menu Assets and select Browse Shaders…



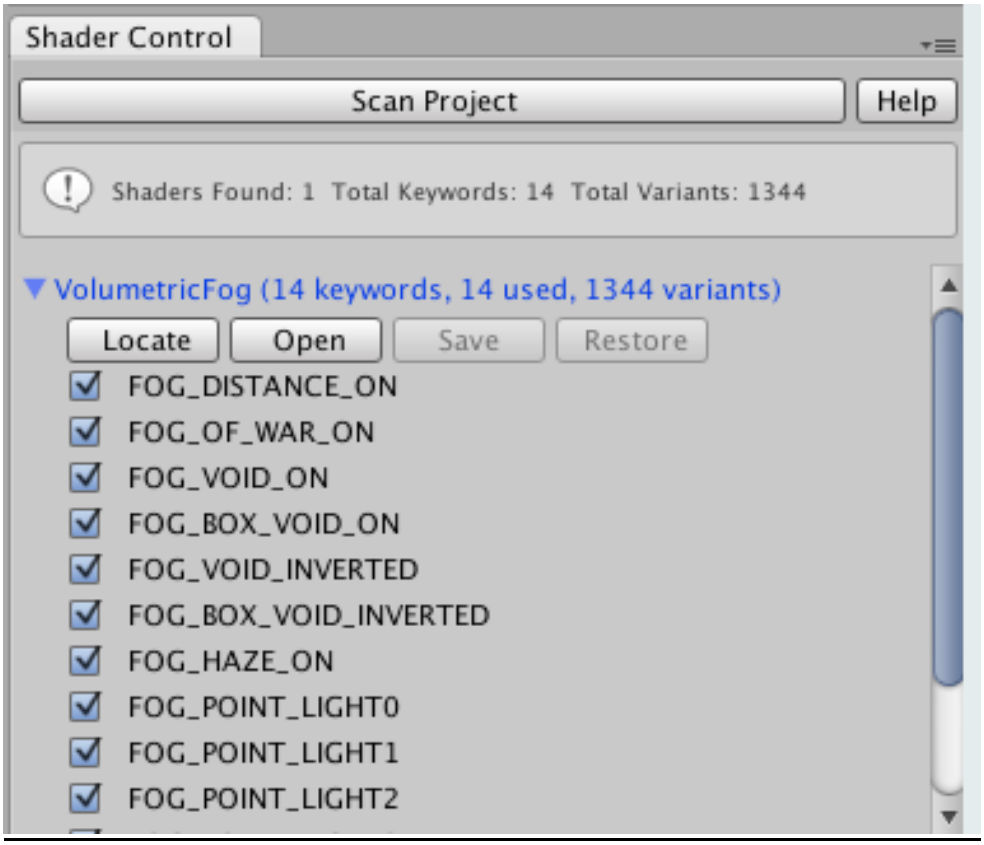

3. Shader Control window will show. It’s useful to drag it and dock it under the Inspector panel:




4. Click “Scan Project” to show a list of shaders along with the keywords used.




5. Expand any shader to list its keywords and options:




Click "**Locate**" to select the shader in the project panel or "**Open**" to open it with default system editor (if it does not open, make sure an application is configured to open files with .shader extension).

Click on the checkbox to disable or enable a keyword. Then click "**Save**" to update the shader file.
Shader Control will create a backup copy with same filename ending with "_backup". Click "**Restore**" to recover the backup copy.

Each keyword you disable will reduce the total shader variants, which will contribute to a reduced build size and compilation time.

"**List Materials**": this option will be shown next to the shader when one or more materials use it in your project. Click to show a list of the materials and have the ability to quickly locate them in the project panel.

"**Prune Keyword**": if a shader source is not available (for instance, the Standard shader), Shader Control will still offer you a view of the materials that use that shader . It will also give you the option to prune any of the keywords used by those materials belonging to that shader. Prune keyword will disable the keyword reference in the materials listed. It won't modify the shader itself, because the source is not available, but the materials won't contribute to the keyword limit in the project.

## About shader keywords

Shaders are extremmely optimized pieces of code that run in the GPU. Because they need to use the fewer number of instructions, shader keywords related to their features or options can be used to create compiled variations of the same code base. This way the code actually executed in the GPU is locally optimized with fewer register usage and less conditional jumps, resulting in faster execution.

For example, in a fog shader, an option could be to blend the fog using 2 colors instead of only one, creating an additional gradient effect. Since you may want to use a single color, a keyword could be used to generate two variants of the same fog code at compilation time, one that uses the second color and another one that just uses one color. This keyword could be named FOG_USE_GRADIENT and is speficied in the shader using a special compiler instruction called "#pragma multi_compile". When the compiler detects a #pragma multi_compile line, it will generate several variations of the shader according to the keywords found.

Following the above example, in the shader file you could find:
#pragma multi_compile __ FOG_USE_GRADIENT

The underscore tells the compiler it must generate a shader variant without any keyword, and then a variant using the keyword FOG_USE_GRADIENT. This keyword can be detected later in the shader code using a compiler conditional that surrounds a piece of code. That piece of code will be include in that compilation if the keyword is enabled.

Many complex shaders, also called "uber shaders", make use of lot of keywords so as much as possible code can be used in the same render pass. This approach creates faster shaders compared to chaining effects on sequential render passes.

But uber shaders usually adds lot of features (shader keywords) that many users won't use, producing unnecesary variants hence bigger build size and compilation time. This is the case where Shader Control tool is very useful. By inspecting the shader keywords you can decide to remove the keywords from the shader file itself, avoiding compiling unused variants.

Note that modifying shaders incorrectly can render them useless or produce undesirable visual effects! Make sure you have a backup copy of your shaders or project before applying any automatic change. Also you may need to contact the asset author regarding some shader keywords to ensure you can safely remove them from the shader file.

## Support & Contact info

We hope you find the asset easy and fun to use. Feel free to contact us for any enquiry.
**Visit our Support Forum for usage tips and access to the latest beta releases**.

***Kronnect***
Email: contact@kronnect.me
Kronnect Support Forum: http://www.kronnect.me