PSO Caches

Record the GPU states for your application in advance to optimize rendering.

Earlier graphics APIs, such as **Direct3D 11**, needed to make dozens of separate calls to configure GPU parameters on the fly before issuing draw calls. More recent graphics APIs, such as **Direct3D 12 (D3D12)**, **Vulkan**, and **Metal**, support using packages of pre-configured GPU state information, called **Pipeline State Objects (PSOs),** to change GPU states more quickly.

Although this greatly improves rendering efficiency, generating a new PSO on-demand can take 100 or more milliseconds, as the application has to configure every possible parameter. This makes it necessary to generate PSOs

long before they are needed for them to be efficient.

In a highly programmable real-time rendering environment such as **Unreal Engine (UE)**, any application with a large amount of content has too many GPU state parameters that can change to make it practical to manually configure PSOs in advance. To work around this complication, UE can collect data about the GPU state from an application build at runtime, then use this cached data to generate new PSOs far in advance of when they are used. This narrows down the possible GPU states to only the ones used in the application. The PSO descriptions gathered from running the application are called **PSO caches**.

The steps to collect PSOs in Unreal are:
1. Play the game.
2. Log what is actually drawn.
3. Include this information in the build.

After that, on subsequent playthroughs the game can create the necessary GPU states earlier than they are needed by the rendering code.

This document describes the available PSO types in UE and the detailed processes for generating PSO caches.

Terminology and Supported PSO Types

This document universally refers to GPU state with the term Pipeline State Object (PSO), a name used in D3D12 API. Other APIs use slightly different names. For example, Vulkan uses *pipeline*, and Metal uses *pipeline state*. However, conceptually they all are similar.

The term *PSO Cache* refers to a file with PSO descriptions that is included in the build, so the game can create these states early. In other words, PSO caches are lists of PSOs to create early.

Unreal Engine supports two types of PSOs:

- **Graphics PSOs**, which represent the state of the application's graphics pipeline and consist of many configurable variables.

- **Compute PSOs**, which usually take the form of **compute shaders**.

> (i) In addition to the above PSO types, **ray tracing PSOs** also exist, but UE 5.0 does not support PSO caching for ray tracing data.

Generate PSO Caches

See the pages below for instructions on how to generate PSO caches:

PSO Precaching

Use PSO precaching to automatically collect and compile PSOs.

Creating a Bundled PSO Cache

Manually create and bundle a PSO cache with your game.