

Developer

/ Documentation

/ Unreal Engine ▾

/ Unreal Engine 5.4 Documentation

/ Testing and Optimizing Your Content

/ Unreal Insights

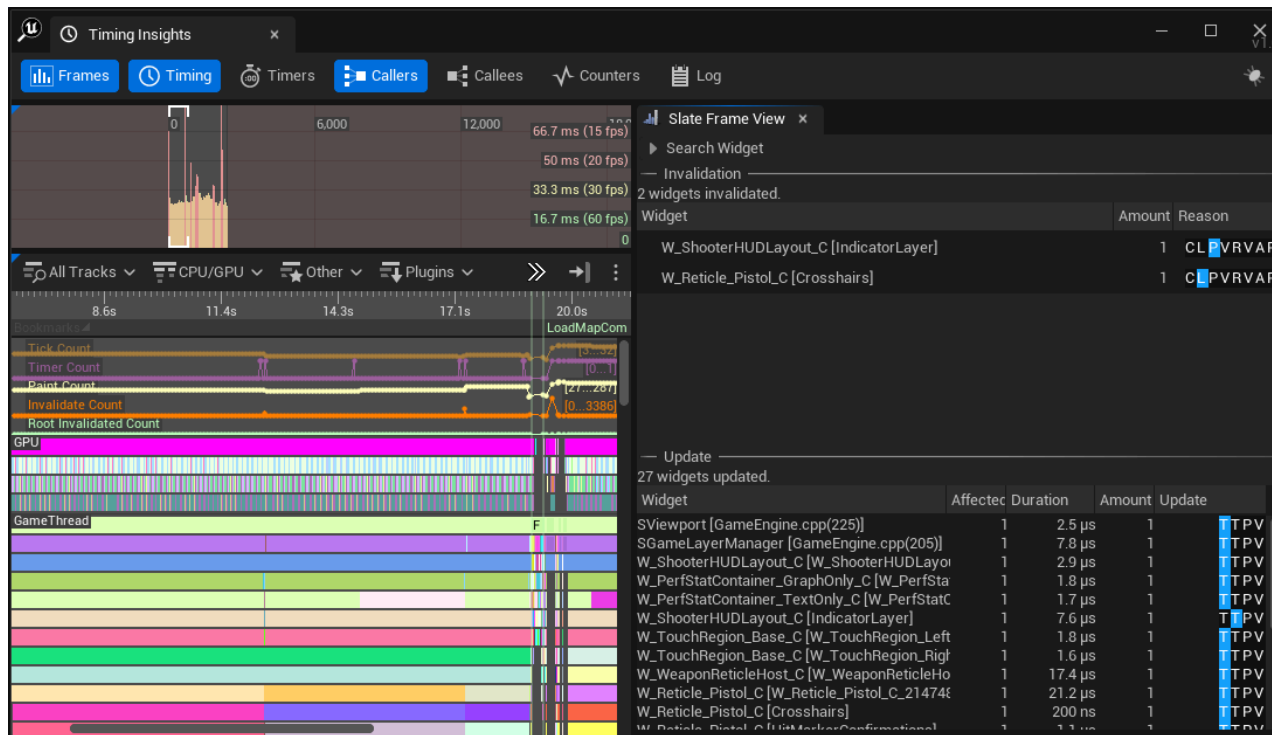
/ Slate Insights

Slate Insights

Overview of Slate Insights, an extension of Unreal Insights that helps users debug Slate and Unreal Motion Graphics (UMG).



While developing the user-interface (UI) for applications, UI programmers can have a difficult time finding the source of changes that trigger UI updates. **Slate Insights** assists developers improve the performance of their UI by providing the tools to identify the root cause of a specific Slate and UMG update.



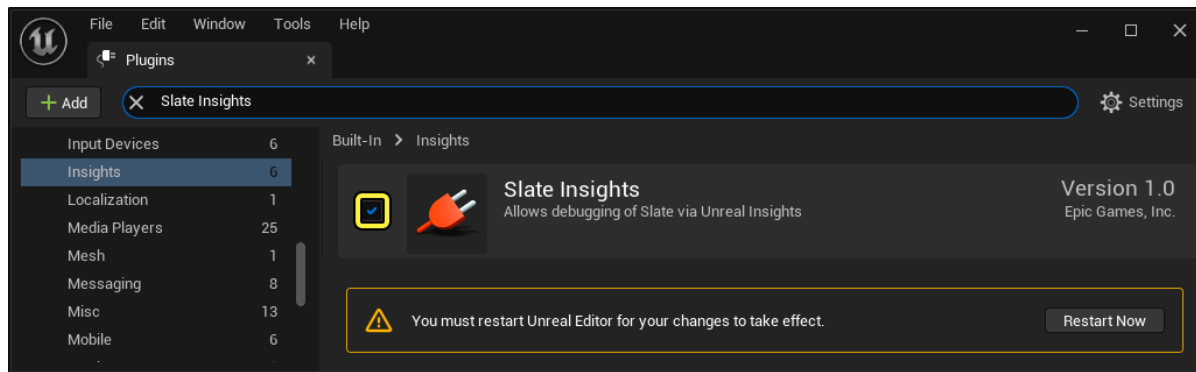
The Slate Frame View running in Slate Insights.

While using the Slate Insights plugin, developers can use **Slate Frame View** to get a list of widgets being painted, invalidated, or updated per frame. Slate Insights enables developers to debug and optimize their UI.

Setup

Using Slate Insights

To begin using Slate Insights in the editor, you can either enable the plugin from inside the editor or from the `*.uproject` file. To enable the Slate Insights plugin from the editor, locate it under **Editor > Plugins > Built-In > Slate Insights**.



Otherwise, enable the plugin from the project's `*.uproject` file.

```
1 {
2   "FileVersion": 3,
3   "EngineAssociation": "{2118FE05-4070-6D80-44FC-17A9E96EAE93}",
4   "Category": "",
5   "Description": "",
6   "Plugins": [
7   {
8     "Name": "SlateInsights",
9     "Enabled": true
10  }
11 ]
12 }
```


 Copy full snippet



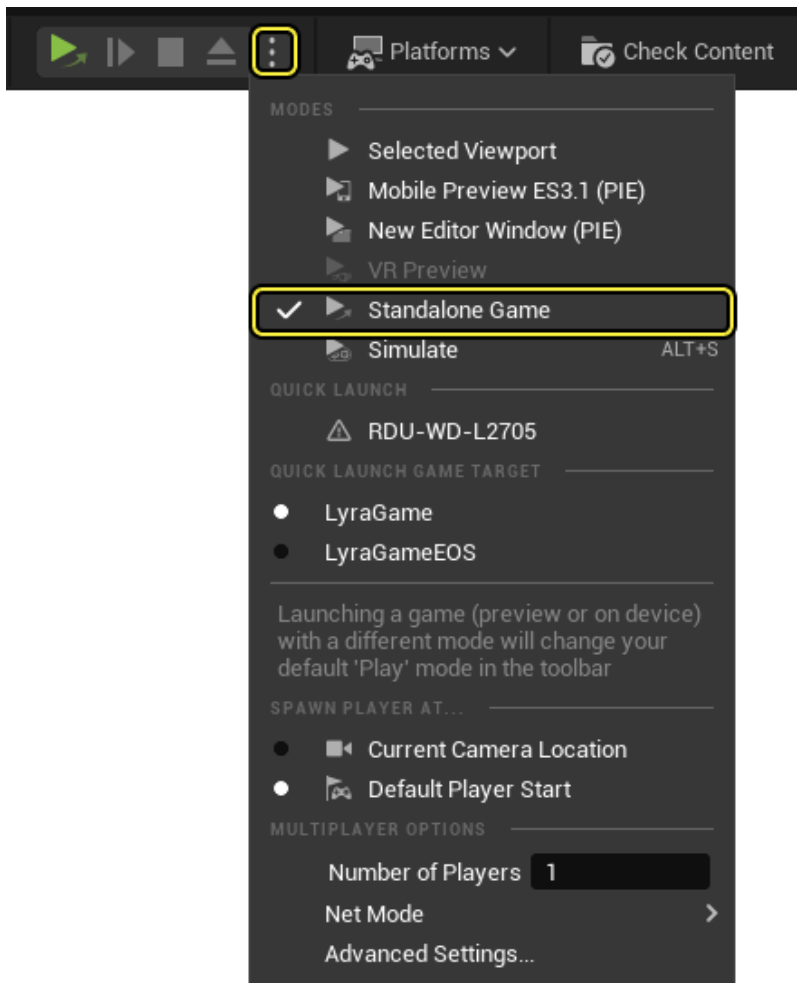
If you built the editor from source, you will need to compile and run the editor for your project after enabling the required plugin.

After enabling the required plugin, restart the editor. follow these steps to trace data for Slate Insights.

1. Open the game project in the editor, and open **Editor Preferences**.
2. Under **Level Editor - Play > Play in Standalone Game**, enter `-trace=slate` into the **Additional Launch Parameters** text box.

 Learn more command-line options in the [Unreal Insights Reference](#).

3. Set the project's Active Play Mode to **Standalone Game**.

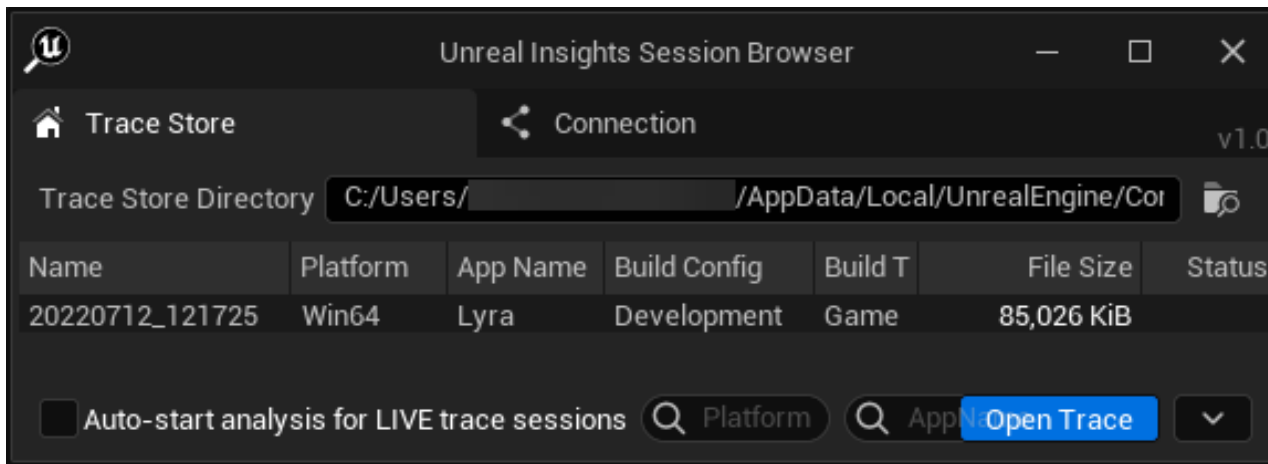


4. If Unreal Insights hasn't been built, find the Unreal Insights project under `Engine\Source\Programs\UnrealInsights`, and build the application.
5. Run the **UnrealInsights** application. Find the application under `Engine\Binaries[Platform]`.
6. With Unreal Insights running, run the project to record the application's trace data.



- Traces automatically write to the following project directory:
`%appdata%/local/UnrealEngine/Common/UnrealTrace/Store/001`.
- Because trace sessions can grow in size over time, make sure to actively monitor and manage the `*.utrace` files.

7. After recording the project's trace data, close the standalone application.
8. To start the analysis in Unreal Insights, select the project's trace session (1) and click **Open** (2).



9. To open the Slate Frame View tab, select **Menu > Slate Frame View**

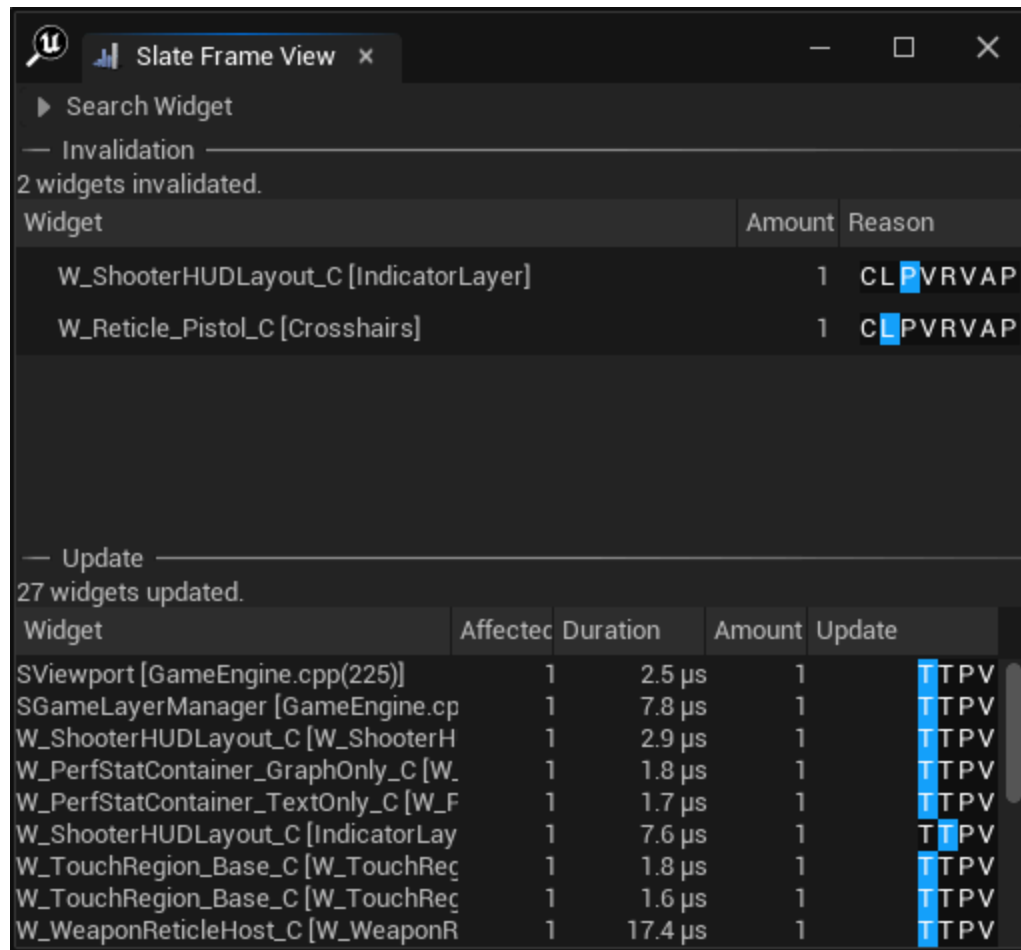


For general information about navigation and using Unreal Insights, read the [Unreal Insights Overview](#).

10. Analyze the UI data with **Slate Frame View**.

Slate Frame View

After capturing the application's trace data with Unreal Insights, open the `*.utrace` file to begin analyzing the UI states with **Slate Frame View**.



Invalidation

2 widgets invalidated.

Widget	Amount	Reason
W_ShooterHUDLayout_C [IndicatorLayer]	1	CLPVRVAP
W_Reticle_Pistol_C [Crosshairs]	1	CLPVRVAP

Update

27 widgets updated.

Widget	Affected	Duration	Amount	Update
SViewport [GameEngine.cpp(225)]	1	2.5 µs	1	TTPV
SGameLayerManager [GameEngine.cp	1	7.8 µs	1	TTPV
W_ShooterHUDLayout_C [W_ShooterH	1	2.9 µs	1	TTPV
W_PerfStatContainer_GraphOnly_C [W_	1	1.8 µs	1	TTPV
W_PerfStatContainer_TextOnly_C [W_F	1	1.7 µs	1	TTPV
W_ShooterHUDLayout_C [IndicatorLay	1	7.6 µs	1	TTPV
W_TouchRegion_Base_C [W_TouchReq	1	1.8 µs	1	TTPV
W_TouchRegion_Base_C [W_TouchReq	1	1.6 µs	1	TTPV
W_WeaponReticleHost_C [W_WeaponR	1	17.4 µs	1	TTPV

Slate Frame View features two report areas, **Invalidation** and **Update**. Each area features its own collection of trace flags.

Invalidation Trace Flags

The Invalidation Trace Flags area reports **Widget** information, including a corresponding **Amount** and one or more **Reason** for the widget invalidation. When expanding widgets to review their invalidation reasons, Slate Frame View highlights one or more flags from the "LPUCRV" flag list, defined as follows.

Flag	Definition
L	Layout flag. True means the widget's desired size changed.
P	Paint flag. True means the widget needs repainting but nothing affects its size.
U	Volatile flag. True means the widget volatility changed.
C	Child order flag. True means a child was added or removed, implying a layout exists.
R	Render transform flag. True means the widget render transform changed.
V	Visibility flag. True means the widget visibility changed, implying a layout exists.

Update Trace Flags

The Update Trace Flags area reports **Widget** information, including a corresponding **Amount** and one or more **Reason** for the widget update. When expanding widgets to review their update reasons, Slate Frame View highlights one or more flags from the "UTPV" flag list, defined as follows.

Flag	Definition
U	Tick flag. True means the widget was updated.
T	Active timer update flag. True means the widget had an active timer.
P	Repaint flag. True means the widget was dirty and repainted.
V	Volatile flag. True means the widget was volatile and repainted.