

F.A.Q:

Q: WHY DO I NEED THIS?

A: This plugin will help you achieve a high level of confidence in the actions to take and changes you want to make to your game. If you ever found yourself in a situation where you knew something doesn't work as intended, something about your design or your layout felt off, or worse, caused a lot of frustration and yet you still couldn't put your finger on what that something was exactly, then this plugin is for you.

Q: HOW DOES THE PLUGIN HELP ME ACHIEVE THIS?

A: By recording an exact replay of what your player or players have been doing and what happened to them and turn it into statistical data, via a heatmap or by importing it into a spreadsheet software. This way you can tell exactly what went wrong and probably why, too. We found this to be extremely helpful already and we have good examples to prove it.

So far, using this plugin for our own game, we were able to:

- Identify hot spots in two maps where one team would have trouble breaking through the other team's defenses, essentially getting the game stuck. This frustrated our players for months until we finally brought this tool online.
- Identify many places where players managed to leave the game's level boundaries
- Determine routes that are most popular and some that are less popular, allowing us to rebalance layouts to get players to use all routes more equally.
- Find places where players managed to hide in large meshes with missing geometry, exploiting this to their advantage
- Find places where players camped frequently
- Find places where players had found a small gap or angle in our layouts, allowing them to spawn camp over long distances
- We also realized that the ShooterGame AI prefers straight paths and dies in just a handful of locations while testing it. In other words, you can use this to improve your AI behaviors, too.

In short, this tool was invaluable in helping us identify the cause for several frustrating design flaws and bugs in our own game. And now we want to share it with you.

Q: WHAT DOES IT DO?

This plugin is a tool to record player and item locations, actions taken, events that happened to them or any entity in the world. Think of it as a Blackbox for airplanes – but for your game.

- Record any event when and where it happened via Blueprints or C++, the choice is yours.
- Every event can be associated with a specific player. Ideal for online matches.
- Load the recorded data into the editor, via the Telemetry Widget
- Display your player movements as lines in the world to track their movement.
- Determine where players died most, who killed them and where the killer has been.
- Display any number of events you recorded as a heatmap. Whether that's player paths, deaths, respawns or skills used.
- Filter what you see per player, per event type or both
- Comes with all its code so you can define and record your own events that match your games needs
- All events are stored as JSON data and can be easily imported into your favorite spreadsheet software to create statistics, provided that that software supports JSON imports.
- Written in C++, and BP, can be used for both the GitHub version of UE as well as the launcher version
- The UI is made in UMG

- As a bonus, we also added a small UI to trigger Replay Recordings to go on top of your telemetry data. You can then literally watch what happened *and* see the relevant parameters behind

Q: HOW DO I EVALUATE THE RECORDED DATA?

A: The Plugin comes with an in-editor UI to load, filter and manage the recorded events and generate a heatmap from all events or a narrow selection of them.

With the recordings you can look at the events in the editor, as they occurred in the game, replay them in their timeline and see where your players went, what they did and what happened to them. If you'd like to know if certain events happened in a specific location more often than in other locations, there's a handy Heatmap tool built-in, too. E.g., you suspect that there's a boss or location where your players got defeated most and need to tweak the game or level layout to keep that from happening? Just look for the big red dot in your map. All done conveniently from within the Unreal Editor.

Q: IS THIS PLUGIN RIGHT FOR ME AND MY GAME?

A: Every game has something worth tracking, to determine if you and your players are on the right path. This plugin was built for an existing multiplayer FPS game and the default set of events reflects this to some degree. However, with just a little more work, you can easily create events to track the gameplay of an economy simulator, if you like.

Q: HOW DO I IMPLEMENT IT?

A: Enable the plugin, restart your editor if required and start hooking up the appropriate record blueprint nodes to whatever you want to record.

Then you can run a round of your game, verify if all events were recorded right – the data is written in human-readable JSON – and load it back into the editor.

Q: DO I NEED TO WORK WITH LYRA/SHOOTERGAME OR CAN I INTEGRATE THIS PLUGIN INTO ANY TYPE OF GAME?

A: You can implement this plugin into any kind of UE4 and 5 based games. Most of the plugin works by calling Blueprint Library functions to record the data and uses an Editor Widget Blueprint as GUI to load and analyze the data.

Q: ARE THERE ANY LIMITATIONS?

A: We found a few limitations while developing the plugin:

- It takes a lot of time to create actors in Unreal Engine and therefore we limited the number of event actors, to keep loading times low. By default, 100 events are displayed, if they are nearby and within your camera angle. You can modify these parameters to your liking though, by increasing or decreasing the pool size and varying the draw distance.
- It can take several seconds to load a large set of data. The largest we recorded was 30mb and takes about 20 seconds to load on our reference machine, most of time is taken to create the event actors. Don't worry if it takes a bit of time, it's just busy – not crashing.
- Drawing hundreds, if not thousands of lines to reflect the paths your players have taken can be quite a performance hit. So, we introduced a line pool, too and a drawing limit for paths.

Q: CAN I CREATE MY OWN EVENTS?

A: Yes, you can. You can go about this in two ways:

1. Define a new event type in the **ETIEventTypes** Enum in *TIGameTelemetryTypes.h*. This method requires a bit of C++ knowledge and requires the plugin to be rebuilt, but allows you to filter your new event type in the easiest way. We recommend you use this way.
2. Record a CustomEvent and use the Data input to give it a tag that describes your new event type. This is easier to setup on one hand, but takes a few extra steps to filter later when you evaluate the data.

Q: COULD YOU NOT USE A BLUEPRINT ENUM SO WE CAN EASILY ADD OUR OWN, INSTEAD OF IT BEING HARDCODED?

A: We have looked into this in the past, however, a lot of the code relies on the enum in C++ and reading a BP enum from C++ and then working with it in code is rather tricky, so we decided against it for the time being.

HOW IS THIS DIFFERENT FROM RECORDING A DEMO REPLAY?

A: Demo replays are a valuable tool that can serve many purposes. In order to get an accurate overview of what happened at a glimpse or get a full set of the parameters involved, you want to record more data than what demo replays will provide you with. In other words, both tools have their purpose and for this reason, this plugin contains a UI (WIP) to record and replay them as a bonus.

FEATURES

EVENT RECORDING

Any event that is called in code can be recorded. Some examples of the existing record nodes:

- Player respawns
- Player location
- Player skill used
- Player weapon used
- Events, such as explosions
- Objective state change
- Objective completed
- Match end

EVENT REPLAY

You have three options to display the events that took place:

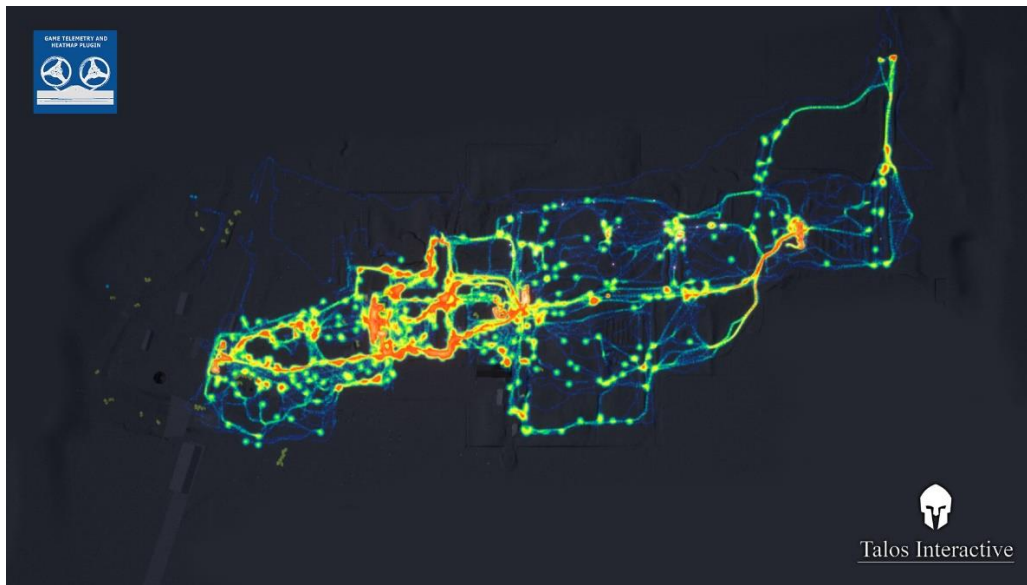
- All events at once over the entire timeline.
- All events from a specific timestamp + a fixed number of seconds past that timestamp. How many seconds that is can be changed in the settings.
- As a replay, to watch the match unfold
- To be added in a future version: As an actual demo replay + event replay

EVENT FILTERING

You can filter all displayed events by:

- A specific player ID
- A team ID
- An event type, like locations, skills used or objectives that got completed
- Their timestamp
- Many more

HEAT MAP



- You can draw a heatmap in the level, based on your recorded data, by hitting a single button. The heatmap will display all currently filtered or simply all events.
- You can change the event size on the heatmap to either see them better, or to differentiate multiple event locations across the map better, in case they have merged into a large event.
- You can change the event magnitude, to differentiate the number of events in their locations better. The color code goes from Blue, over Green to Yellow and ends with Red.