

Developer

/ Documentation

/ Unreal Engine ▾

/ Unreal Engine 5.4 Documentation

/ Making Interactive Experiences

/ Networking and Multiplayer

/ Testing, Debugging, and Optimization

/ Network Profiler

Network Profiler

Analyze network traffic and performance information captured at runtime.



The **Network Profiler** is a standalone tool used to display network traffic and performance information recorded by **Unreal Engine (UE)** as a game runs. This is a good way to identify areas of a multiplayer game that use unusually high bandwidth, since you can see how much bandwidth individual actors, remote procedure calls (RPCs), and properties contribute to the total.

Record a Profiling Session

Before you can use the network profiler, you must record some data for it to analyze. To do this, you need to use a version of the engine with stat tracking enabled. Typically, this means you need to use either a debug build, or for non-debug configurations, an editor build. More specifically, the engine should have been compiled with the `STATS` macro defined to a nonzero value.



For more information on stat tracking, see our [Stat Commands](#) documentation.

With a Command-Line Argument

You can pass the `networkprofiler=true` command-line argument to record as soon as the engine starts up.

With a Console Command

You can also use the following console commands at runtime to control the recording of network profiler data:

- `netprofile`: toggles recording on and off
- `netprofile enable`: start recording if not already recording
- `netprofile disable`: stop recording if currently recording

File Save Location

The data file recorded during a profiling session is saved as

`<PROJECT_DIRECTORY>/Saved/Profiling/<PROJECT_NAME>-<TIMESTAMP>.nprof`. If you see a file named `NetworkProfiling.tmp` in the same directory, this is just the temporary file the engine uses while collecting data. When the current profiling session is stopped, this file will be renamed according to the scheme above and is then ready to be opened in the standalone network profiler tool.

Server and client considerations

You can record profiling data from both clients and servers. However, be aware that since only servers replicate actors and their properties, when viewing a profile recorded on a client, you will only see detailed data for RPCs that were sent from that client.

View a Profiling Session

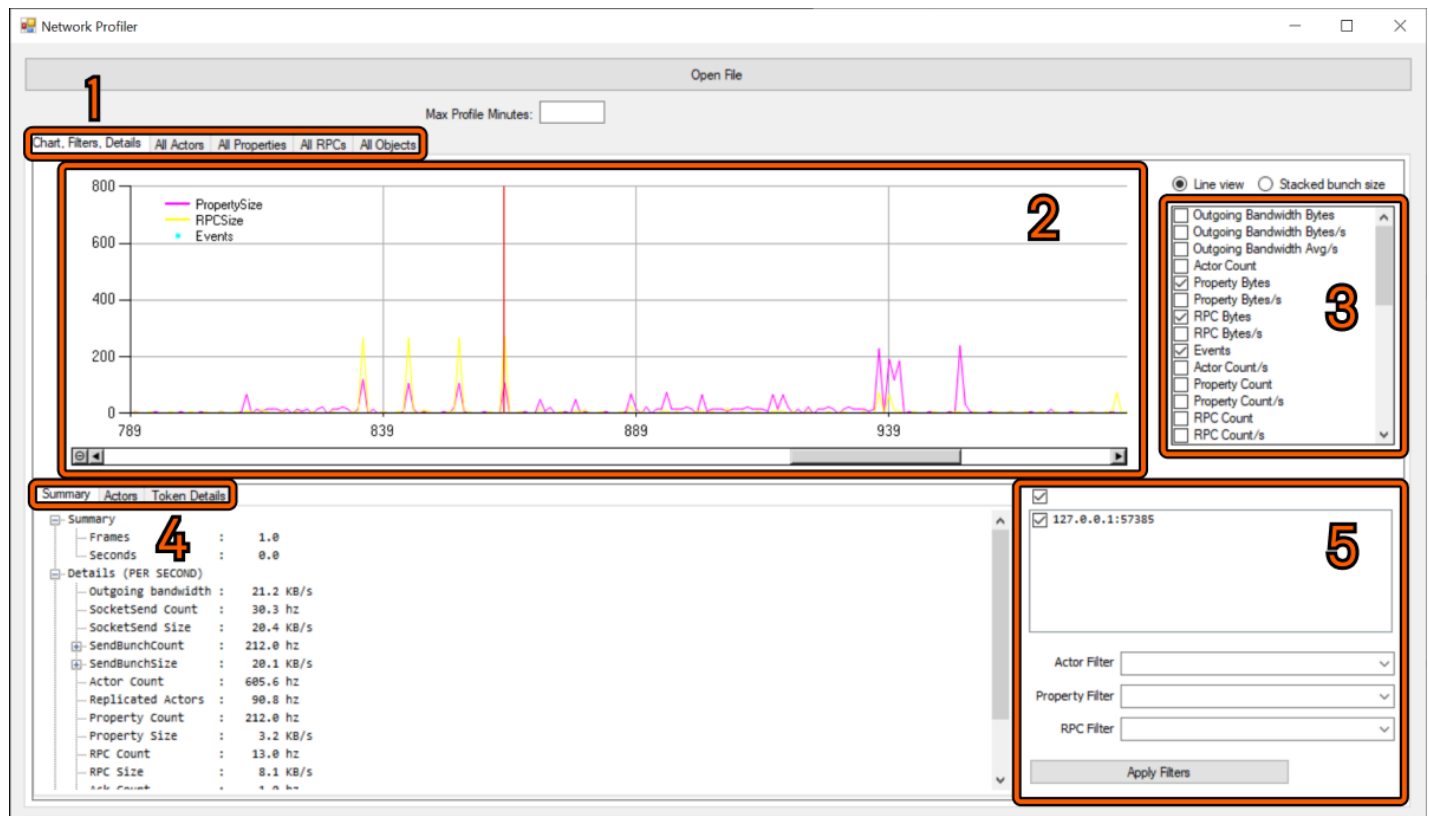
The network profiler is a standalone application located at

`<UNREAL_ENGINE_ROOT>/Engine/Binaries/DotNET/NetworkProfiler.exe`. At the top of the

window, click the **Open File** button to select a `*.nprof` file and view the data contained within.

User Interface

By default, the **Chart, Filters, Details** tab is selected. This section provides an overview of the network profiler user interface.



1. [View Tabs](#)

- [Chart, Filters, Details](#)
- [All Actors](#)
- [All Properties](#)
- [All RPCs](#)
- [All Objects](#)

2. [Chart](#)

3. [Stats List](#)

4. [Details](#)

- [Summary](#)
- [Actors](#)

- [Token Details](#)

5. [Apply Filters](#)

View Tabs

The **View Tabs** are where you select the information that you would like to view.

Chart, Filters, Details

This view is the most detailed. In this view mode, you can:

- View the [Chart](#)
- Refine analysis with the [Stats List](#)
- Get an in-depth look at [Details](#)
- [Apply Filters](#)

All Actors

The **All Actors** view mode displays information about all of the actors that were profiled in this session. This tab has similar functionality and display information as the [All Properties](#), [All RPCs](#), and [All Objects](#) tabs. Each of these tabs show a summary of the respective replication data for the entire profiling session. Note that the data displayed in these tabs is not affected by the current frame or time span selected in the chart on the **Chart, Filters, Details** tab.

The tabs show a list of actors, properties, RPCs, or objects respectively. The **Total Size** and **Average Size** columns show how much bandwidth was needed for a particular item, and the **Count** column represents how many times a particular item was replicated during the profiling session. This information includes:

Header Name	Description
Actor Class	Name of the actor class for this row.
Average Time	Average time to send replicated information in <input type="text" value="ms"/> .
Time	Total time spent sending replicated information in <input type="text" value="ms"/> .

Header Name	Description
Average Size	Average size of replicated information in Bits .
Average Size	Average size of replicated information in Bytes .
Count	Number of times this actor's information was replicated.
Total Size	Total size of replicated information in KBytes .



You can toggle between sorting by a column in ascending or descending order by clicking on the column heading.

All Properties

The **All Properties** view mode displays information about all of the properties that were profiled in this session. This table has the same headers as the [All Actors](#) section except the **Actor Class** header is replaced with the **Property** header representing the name of the property for the specified row.

All RPCs

The **All RPCs** view mode displays information about all of the RPCs that were profiled in this session. This table has the same headers as the [All Actors](#) section except the **Actor Class** header is replaced with the **RPC** header representing the name of the RPC for the specified row.

All Objects

The **All Objects** view mode displays information about all of the objects that were profiled in this session. This information includes:

Header Name	Description
Object Class	Name of the object class for this row.
# Comparisons	Number of comparisons made.
# Replications	Number of replications performed.
Comparison Time	Amount of time spent making comparisons in ms .
Average Time Per Compare	Average amount of time spent making comparisons in ms .

This view mode also includes a separate table for properties. This information includes:

Header Name	Description
Property	Name of the property for this row.
# Comparisons	Number of comparisons made.
# Times Changed	Number of times the property value changed.
# Replications	Number of replications performed.

Chart

This is the main chart view. Similar to charts in other profiling applications, it shows a graph of the enabled stats over time. You can click on a particular frame to populate the other views with the data associated with that frame, or click and drag across the graph to zoom in to that area and view a summary for that time span.



Chart view and all of the following options are only available in the [Chart, Filters, Details](#) view mode.

Stats List

This is a list of all the stats tracked by the network profiler. Toggling the checkboxes will show or hide that particular stat in the graph. For most stats, you can choose to show the raw count, the count per second, the number of bytes, or the number of bytes per second.

Details

The details tabs display more in-depth network profiling information.

Summary

This is a summary of the data for the time span that has been selected in the chart. It shows raw totals as well as per-second information. Note that the "Outgoing bandwidth" will be greater than the "Game socket send size" because "Outgoing bandwidth" includes an approximation of the IP and UDP header sizes as well.

Actors

While a single frame is selected in the graph, this view will show a list of actor types that were replicated during that frame, sorted by the CPU time it took to replicate them. If an actor has associated replicated properties, expanding the tree view for the actor will show each property. For actors, the first column of numbers is the CPU replication time for that actor, in milliseconds. The second column is the number of bytes used to represent the actor in the bunch, and the third column is the number of actors of the given type that were replicated during the current frame. For property rows, the first column is the number of bytes used for the property, and the second column shows how many properties with the given name were replicated for the actor.

Token Details

This view breaks down the data that was sent on the currently selected frame. You can see which actors, properties, and remote functions were sent, as well as how many bytes each one contributed. Note that replicated actors and RPCs are both included in bunches, so the value in the `SendBunchSize` sections includes the bytes from sending RPCs and replicating

actors. Ultimately, the `SocketSend` entries will tell you exactly how many bytes the engine sent to the outgoing socket. Also note that the standard UDP packet overhead is not included here.

Apply Filters

You can enter a filter for actors, properties, and RPCs, and after clicking the **Apply Filters** button, the graph will update to only show data that is relevant for actors, properties, and RPCs that contain the text entered in the filter fields. Selecting a frame or range of frames in the graph after a filter has been applied will also filter the data in the summary and frame details views accordingly.