# Remote Inspector

## What Remote Inspector is

A toolset designed for fast iteration of Unity remote debugging and optimization.

Wouldn't it be great to be able to browse GameObjects in the Hierarchy window and change their properties in the Inspector window just like you would in the editor when running the game remotely? What if you could change the quality settings or change the time scale? Furthermore, if you could call a script, how many possibilities would you have?

**The Remote Inspector was built to make all of this possible.**

## Update

- Method Calling and Static in Remote Inspector 1.5
- Huge Performance Boost in Remote Inspector 1.4

## Features

For large projects, building an APK or IPA takes minutes to hours. So if you have to re-build to add a single log line, you're wasting a lot of your work time. But with this asset, many things are now possible without a build. You can spend your time on things that matter.

These are possible when running the game on a remote, including Android or iOS. Even when running in the editor.

- Exploring GameObjects in the Hierarchy window
- Viewing and editing game object and component information in the inspector window (private and public members supported. fields and properties supported)
- Calling script methods in the Inspector window (ref and out types are supported)
- Incremental search in the Hierarchy window and the Inspector window
- Various view options provided
- [PRO] Viewing and editing static fields and properties
- [PRO] Calling static methods
- [PRO] Viewing and editing material information in the Inspector window
- [PRO] Adding/deleting shader keywords for materials in the Inspector window
- [PRO] Incremental search for material properties and static members

## Upgrade Info

You have to delete the Remote Inspector free version before you install the pro version.

## Installation

Just add the RemoteInspectorServer prefab to your first scene and you're done. This prefab has a small web server that communicates with the Unity Editor over the network.

If your PC and your mobile device are on the same network, enter the IP address and port of your mobile device and click the Refresh button in the Hierarchy window. Default port is 8080. Or you can change the default port in the RemoteInspectorServer prefab.

Alternatively, if your Android device is connected to your PC via a USB cable, you can use ADB's port forwarding function. The following command will add port forwarding settings. If successful, type 'localhost:8080' and click the refresh button in the Hierarchy window.
See Unity document's ADB profiling section for more information.

```
    adb forward tcp:8080 tcp:8080
```

# IL2CPP

If you're using IL2CPP, you may not see the information you need. This is because IL2CPP deletes unused classes and members through static inspection. To avoid this, you need to set the link.xml file in your Unity project. See Bytecode Stripping page for more information.

The Remote Inspector has a file example.link.xml that preserves all UnityEngine core classes. When creating a development build, change the name to link.xml. We don't recommend using it in builds for your product, as it may contain unnecessary classes.

# Limitations

Because Remote Inspector does not support custom editors, it displays members in raw form. This is similar to the Debug mode of the Unity Inspector. It can be confusing because it is different from what the Unity Inspector shows, but it has the advantage of being able to view and edit information that was not previously possible.

Supports most Unity primitive types, but does not yet support asset types except textures. We plan to continuously increase the supported types, including asset types.

# Connection Troubleshooting

## Net Error:Cannot connect to the destination host

First make sure you have built it with the RemoteInspectorServer prefab. If there is no problem with the build, check the network connection from the PC to the mobile device. Enter http://[IP Address]:[Port] in the address bar of a web browser such as Chrome. Replace [IP Address] and [Port] with the values entered in the Hierarchy window. If there is no problem with the network connection, the "Welcome to Remote Inspector!" in the web browser text appears.

If you want to connect your device via WIFI, make sure your PC and device are on the same network.

If you are trying to connect your Android device via USB/ADB, check the ADB forward settings. Enter the following on the command line.

```
adb forward --list
```

If the ADB forward setting is normal, the following output is displayed.

```
1515fa33f65c3282 tcp:8080 tcp:8080
```