


Setup Guide

To get setup with GONet v1.3+, follow the steps below. This is only required once per project.

1. Prerequisites
    a. Ensure you are using Unity3D 2022.3 or later
    b. (optional) Watch a complete sample project tutorial video: https://www.youtube.com/watch?v=fs1flIi35JM
2. Instructions (pre-import)
    a. Ensure unsafe code allowed: Edit => Project Settings => Player => Allow 'unsafe' Code
    b. Ensure .NET API version: Edit => Project Settings => Player => Api Compatibility Level* => .NET Framework
    c. (optional) If you want GONet logging features, ensure logging configured: Edit => Project Settings => Player => Scripting Define Symbols (contains at least: "LOG_DEBUG;LOG_INFO;LOG_WARNING;LOG_ERROR;LOG_FATAL")
    d. (optional) It is recommended that you check the box next to "Use incremental GC" in Edit => Project Settings => Player since the introduction of MemoryPack to replace MessagePack requires use of class instead of struct for events (on list of things to add memory pooling for in later releases)
    e. Import GONet unity package into your project. NOTE: Most likely based on how you acquired this document, you already imported GONet prior to reading this. That is OK. You will just notice some compilation errors prior to executing the above steps.
3. Instructions (post-import)
    a. Compile (this should happen automatically after import), or the subsequent steps are not going to work
    b. Drag *Assets/GONet/Resources/GONet/GONet_GlobalContext* into your start-up scene (optionally => open *Assets/GONet/Sample/GONetSampleScene.unity* that already has it instead of using your scene)
    c. Click Run/Play in Unity editor to play the scene (code generation will occur and scene should play….no errors/exceptions…if all is well)
    d. With scene running and Game windows focused, press the following keys simultaneously on the keyboard to spawn a server: left ALT + S (*GONetServer(Clone)* instance appears in the scene and server is now listening for connections from clients)
    e. Click Run/Play in Unity editor to stop playing the currently playing scene

f. If all that went well, you should be set to test some stuff in builds or just develop some and then test

4. Instructions (with build)
    a. Create a build (e.g., gonet_sample.exe on Windows)
    b. If on Windows:
        i. Open project folder => */Assets/StreamingAssets/GONet* and copy Start_CLIENT.bat and Start_SERVER.bat
        ii. Paste files into build folder where gonet_sample.exe exists
        iii. Open both pasted files, changing GONetSandbox.exe to gonet_sample.exe, save files
        iv. Run *Start_SERVER.bat* (server needs to start first, running this bat file does that)
        v. Run *Start_CLIENT.bat* (client will connect to local server)
    c. If not on Windows:
        i. Run first instance of build, focus mouse there, press left ALT + S (server needs to start first…BEWARE: there is really no indication the server is started, but if the window had focus and you pressed the key combo it started)
        ii. Run second instance of build, focus mouse there, press Left ALT + C (client will connect to local server…BEWARE: before pressing the key combo in the client window, the game might appear to be out of "data sync" and will correct itself once the client is connected to server….the client thinks he owns stuff that he does not until it is forced into submission!)
    d. Yay! You are all setup to use GONet. Go add some GameObjects to the scene with *GONetParticipant* component added to them and do something interesting with it, create another build, test again, rinse, repeat!
5. Support
    a. If any issues arise, please reach out to support via email: contactus@galoreinteractive.com

Visit the product website: https://galoreinteractive.com/gonet