

GoToTag - Unity Project Version Switcher

This set of scripts allows you to quickly switch between different Git tags in your Unity project and automatically launch the correct Unity Editor version associated with that tag.

Files Provided:

- `gototag.sh` : For macOS/Linux users.
- `gototag.ps1` : The core PowerShell script for Windows.
- `gototag.bat` : A simple batch file wrapper for Windows to make `gototag.ps1` easier to run.

Installation and Usage:

Step 1: Place the Scripts

Place all three files (`gototag.sh` , `gototag.ps1` , `gototag.bat`) directly into the **root folder of your Unity project** (the same folder where your `Assets` , `ProjectSettings` , and `Library` folders are located).

Step 2: Make `gototag.sh` Executable (macOS/Linux Only)

On macOS or Linux, you need to make the `gototag.sh` script executable. Open your Terminal, navigate to your Unity project's root folder, and run the following command:

```
Bash
```

```
chmod +x gototag.sh
```

Step 3: Add to System PATH (Recommended for Convenience)

To call `gototag` from any directory within your Unity project, it's highly recommended to add your Unity project's root folder to your system's PATH environment variable. This step is

optional but greatly improves usability.

For Windows:

1. **Search for "Environment Variables":** In the Windows search bar, type "environment variables" and select "Edit the system environment variables."
2. **Open Environment Variables:** Click the "Environment Variables..." button.
3. **Edit Path Variable:** Under "User variables for <Your Username>" (or "System variables" if you want it available for all users), find the `Path` variable and click "Edit...".
4. **Add New Entry:** Click "New" and add the **absolute path to your Unity project's root folder** (e.g., `C:\Users\YourUser\Documents\UnityProjects\MyGame`).
5. **Confirm:** Click "OK" on all open windows to save the changes.
6. **Restart Terminal/PowerShell:** Close and reopen any Command Prompt or PowerShell windows for the changes to take effect.

For macOS/Linux:

1. **Open your shell configuration file:** This is typically `~/.bashrc` , `~/.zshrc` , or `~/.profile` , depending on your shell.
2. **Add the following line** to the end of the file, replacing `/path/to/your/unity/project` with the actual absolute path to your Unity project's root folder:
3. **Save and Exit:** Save the file (Ctrl+O, Enter, Ctrl+X in nano).
4. **Source the file:** Apply the changes by running:

Step 4: Usage

Once the scripts are in place and (optionally) your PATH is configured, you can use `gototag` from your Unity project's root directory (or any subdirectory if PATH is configured):

```
Bash
```

```
# On macOS/Linux
```

```
gototag <tag-name> [--clean]
```

Plain Text

```
# On Windows (PowerShell)
gototag <tag-name> [--clean]
```

Plain Text

```
# On Windows (Command Prompt)
gototag <tag-name> [--clean]
```

Replace `<tag-name>` with the actual Git tag you wish to check out. The optional `--clean` flag will perform a `git clean -x` operation.

If you run `gototag` without any arguments, it will display usage instructions.

The script will:

1. Close any running Unity Editor instances.
2. Perform a `git reset --hard`.
3. **Optionally**, if `--clean` is specified, perform a `git clean -x` to remove untracked files and directories.
4. Check out the specified Git tag.
5. Perform another `git reset --hard` and **optionally** `git clean -x` after switching.
6. Read the `ProjectSettings/ProjectVersion.txt` file to detect the required Unity Editor version.
7. Launch the detected Unity Editor version with your project.

Important Notes:

- **Git Repository:** Your Unity project must be a Git repository with tags defined.

- **Unity Hub Installation:** Ensure the Unity Editor versions specified in your `ProjectSettings/ProjectVersion.txt` are installed via Unity Hub on your system. The scripts look for Unity installations in standard Unity Hub paths.
- **Windows PowerShell Execution Policy:** The `gototag.bat` file uses `powershell - ExecutionPolicy Bypass` to allow the script to run. If you encounter issues, you might need to adjust your system's PowerShell execution policy (though `Bypass` is generally sufficient for single-script execution).
- **Linux Unity Path:** The `gototag.sh` script assumes Unity is installed under `/opt/Unity/Hub/Editor` on Linux. If your installation path differs, you may need to edit the `UNITY_HUB_EDITORS_PATH` variable within `gototag.sh`.