



<http://www.chadjiantoniou.com>

# Easy Console Vars Documentation

Unreal Engine 4



## Contents

Introduction .....	3
Interface.....	3
Header section (1).....	4
ECVar List (2) .....	4
ECVar List Actions (3) .....	5
Add new ECVar form (4) .....	5
Notifications area.....	6
ECVar File definition.....	6

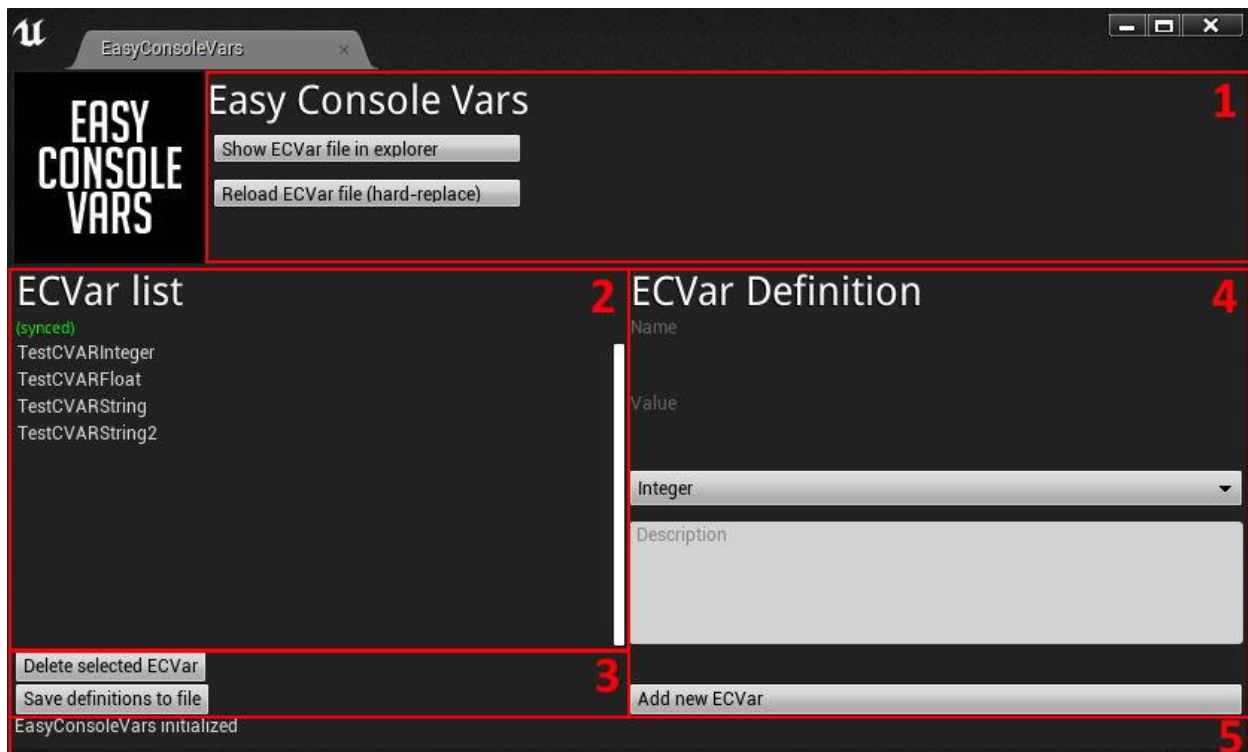
## Introduction

The following will describe the functionality of the Easy Console Vars (“**ECVar**”) editor plugin for Unreal Engine 4 and how you can use it to easily register and save console variables to use for development. The console variables created through this plugin will persist editor close.

Once purchased, the plugin can be found under the “CHadjiantoniou” category in the editor’s Plugin window.

## Interface

The plugin provides an easy way to register console variables via a simple interface. The below section lists and explains the various sections of the interface. Images below represent the state of the interface at the time of this writing and may vary slightly from the final product.



1. Header section with general options
2. List of ECVars defined
3. Actions regarding saving definitions to file and deleting ECVars
4. Form to add new ECVar
5. Notifications area

## Header section (1)

The header section provides the following actions:

- **Show ECVar file in explorer:** Opens Windows Explorer to the location where the ECVar file exists.
- **Reload ECVar file (hard-replace):** Reloads ECVar definitions which are included in the file into the ECVar list. This will replace all existing ECVars which are presented in the ECVar list.

## ECVar List (2)

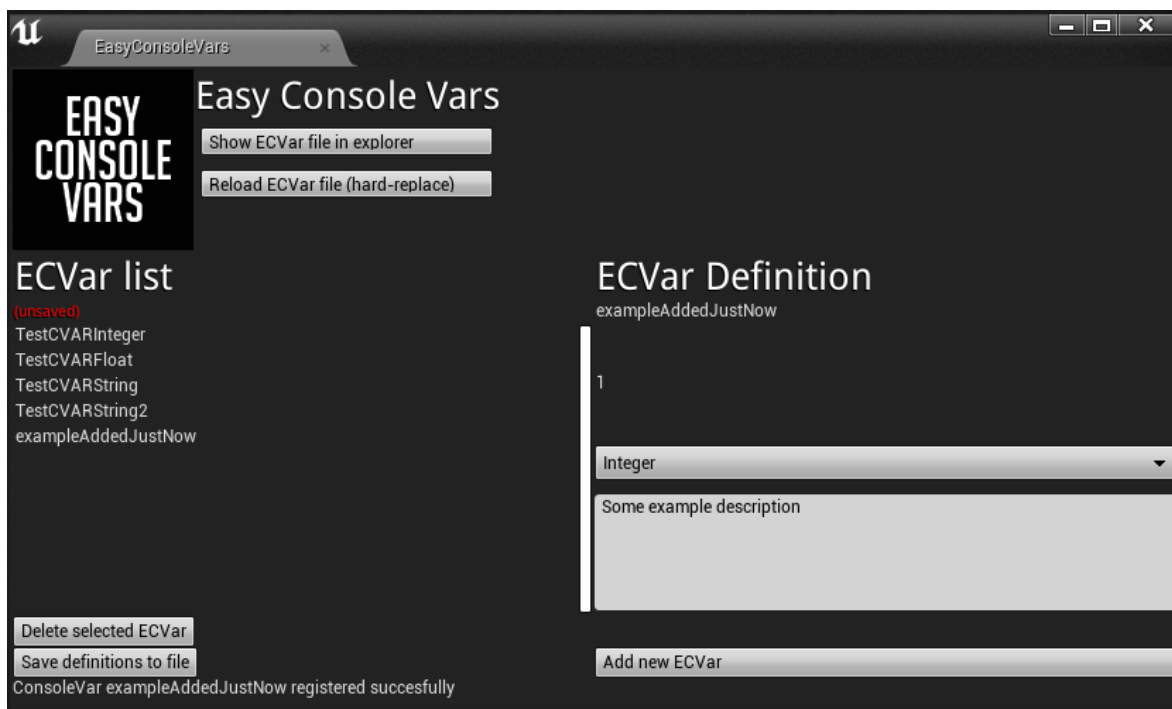
Presents a list of ECVars which are available to use in the engine. The definitions included in this list may be different than the definitions included in the ECVar file. It is important to understand the difference between what is presented in the ECVar list, and what is included in the ECVar file.

- **ECVar List:** Includes all ECVars which are registered in the engine at the present time
- **ECVar file:** Includes ECVars that have been explicitly saved to the file, in order to be made available after editor close.

An indication exists below the title on whether the ECVar list and ECVar file are synced. The indicator may have 2 states as described below:

- **(synced):** The ECVar list and ECVar file contain the same definitions.
- **(unsaved):** The ECVar list and ECVar file contain different definitions. (some of which might be identical)

An example of how the indication appears immediately after adding a new ECVar is shown in the image below:



### ECVar List Actions (3)

The actions that can be taken on the ECVar list are described below:

- **Save definitions to file:** Permanently saves the listed ECVars to the ECVar file. Essentially replaces the definitions in the ECVar file with the ones contained in the ECVar list.
- **Delete selected ECVar:** Removes the selected ECVar from the ECVar list and unregisters the console variable with the engine. This action cannot be reverted.

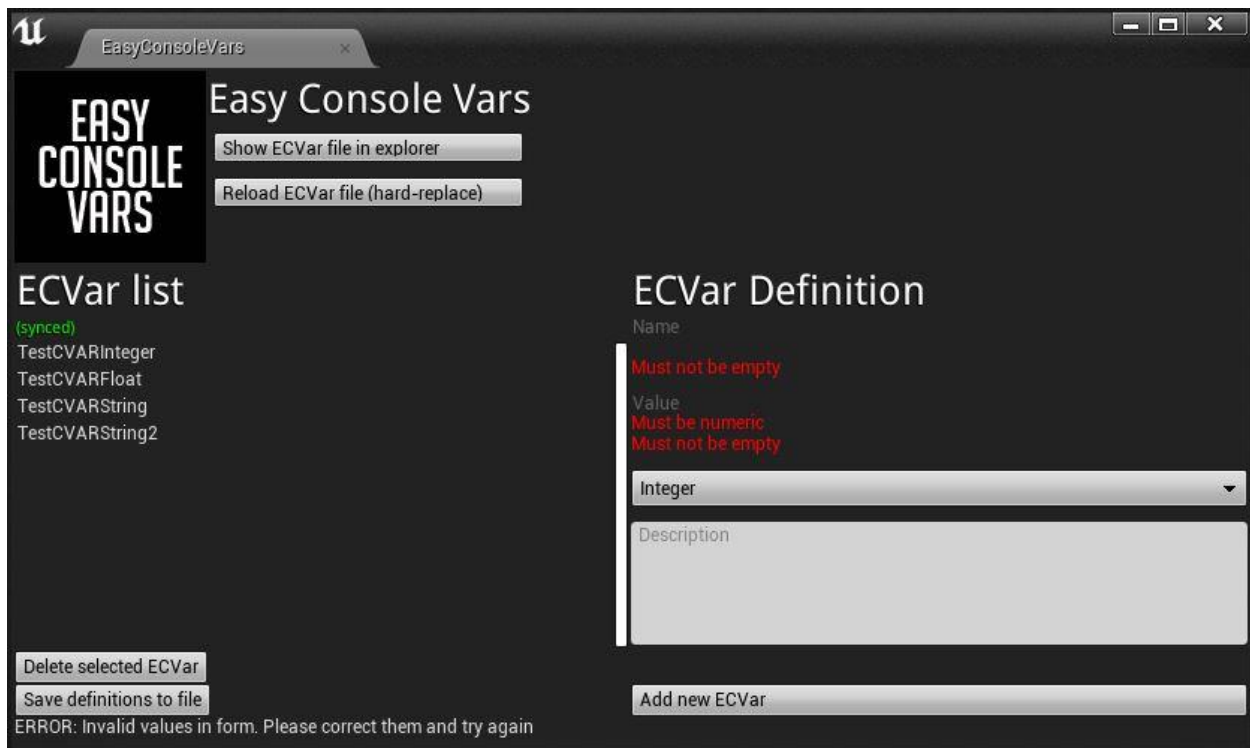
### Add new ECVar form (4)

Presents a form to create a new ECVar and register the console variable with the engine. The form fields are described below:

- Name: The name of the console variable
- Value: The initial value of the console variable
- Type: Combobox that sets the type of the console variable (Integer, String or Float)
- Description: The console variable's help description

The **"Add new ECVar"** button will create a new ECVar with the information entered in the form and will register the console variable with the engine.

The form fields are validated upon clicking the **"Add new ECVar"**, such that no illegal values are entered. For example if "Value" field contains non-numeric data and the "Type" is set to "Integer". If any errors exist, an indication will appear on each erroneous field, prompting the user to corrective action. An example of such error messages is shown in the image below:





## Notifications area

Presents various messages in response to actions taken with the interface.

## ECVar File definition

The plugin uses a custom file to permanently save any console variables created via the EasyConsoleVars interface.

The file lists variable definitions that were created through the plugin in separate lines. Each field in the line is delimited by “|”. The structure of a single ECVar definition is described below:

Field name	Description
Name	The console variable name (ex. “cvar_Speed”) <b>Note: Must not be 0</b>
Type	The console variable type: <ul style="list-style-type: none"><li>• Integer = 0</li><li>• String = 1</li><li>• Float = 2</li></ul>
Value	The console variable’s starting value
Help Description	The console variable’s help section <b>Note: Must not be 0</b>
Flags	Bitmask flags

It is strongly advised **not** to manually edit the file, as this may lead to unexpected results and potential loss of console variable information.