

# LINK

*Version 1.4 by Maskar @ 29<sup>th</sup> June 2020*

## **Table of contents**

### **1. About**

*1.1 Introduction*

*1.2 Compatibility and requirements*

*1.3 Installation and configuration*

### **2. LINK for users**

*2.1 Introduction*

*2.2 Mods menu*

*2.3 Configuration menu*

### **3. LINK for mod authors**

*3.1 Introduction*

*3.2 Cfg files*

*3.3 Adding support*

*3.4 Optional configuration*

**Download this mod at [TESNexus](#)**

# 1. About

## 1.1 Introduction

LINK adds the ability to configure mods in-game, through a configuration menu, making configuring mods faster and more user friendly.

Any mod with LINK support automatically gains the ability to be configured with the configuration menu, when LINK is installed. Adding LINK support to new and existing mods is relatively easy. Have a look at chapter 3 for further details.

**Note that mods with LINK support still work as normal when LINK is not installed.**

## 1.2 Compatibility and requirements

This mod requires [OBSE](#) (version 21 or higher) and [MenuQue](#) to function properly. [ConScribe](#) is recommended to easily export/import settings.

Currently [DarNified UI](#), [DarkUI'd DarN](#), [Dark Transparent UI](#) and [NorthernUI](#) are automatically supported. Any other UIs, like [Btmod](#) and [Major Jim's UI](#) will use the new vanilla UI menus.

## 1.3 Installation and configuration

### Installation

1. Extract the files to a temporary location
2. Copy files to `(install folder)\Oblivion\Data\`
3. Start Oblivion Launcher, click *Data Files*, place a checkmark beside the .esp file
4. Optionally edit the .ini file to customize it to your preferences

### Uninstallation

1. Delete the LINK files in the `Data` folder.

### Upgrading

1. Follow the uninstallation instructions and then the installation instructions.

### Configuration

Either edit the ini file, or use the configuration menu to configure this mod. The ini file is considered the default when pressing the “defaults” button in the configuration menu.

When having [ConScribe](#) installed you can export/import settings from all configurable mods, including LINK, through the mods menu.

## 2. LINK for users

### 2.1 Introduction

LINK adds the ability to configure mods in-game, through a configuration menu, making configuring mods faster and more user friendly.

The configuration menu is accessible through both the main menu (before starting/loading a game) and the pause menu when in-game. Selecting “Options” on either the main menu, or the pause menu (press escape key in gamemode) will, depending on your UI, show a menu similar to the image below.



Press the “Mods” button to open the Mods menu.

## 2.2 Mods menu

The mods menu shows a list of all active mods, if they can be configured with the configuration menu and their mod index.

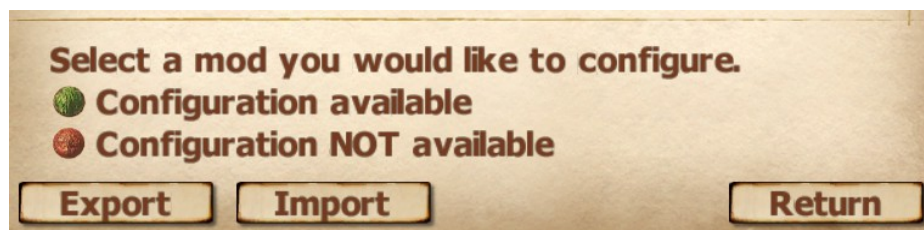


Select a mod that you would like to configure. Only mods with a green image have LINK support and can therefore be configured.

Note that mods can be sorted by name or mod index, by clicking the “Mods” title.

When [ConScribe](#) is installed the mods menu will have an export button added, which can be used to export all settings from all configurable mods to the LINK.log file. This file is located in the `Data\ConScribe Logs\Per-Mod` folder.

When a log file is created (either manually or through exporting) an import button will be available, as shown below.

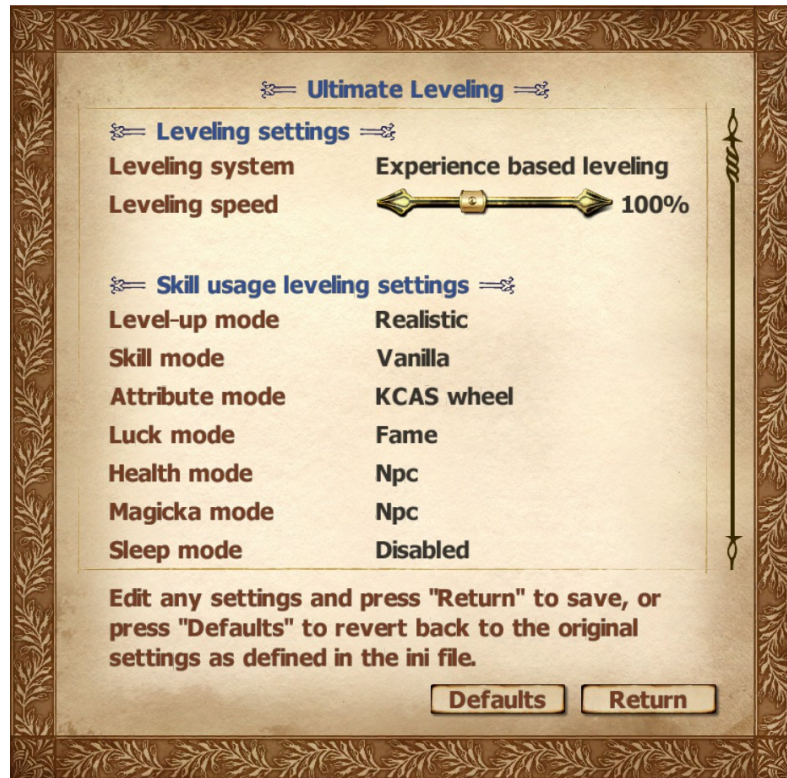


Import settings to quickly configure your mods when starting a new game, load an old savegame, or when a mod was updated and has reset its ini settings (useful for MOO, etc.).

### 2.3 Configuration menu

The configuration menu shows the settings that are available to be configured. There are currently 3 types of settings. These are toggles, sliders and keybindings.

Simply edit the settings and press the “Return” button to save your changes, or press “Defaults” to load the default settings, as defined in the ini file.



Note that clicking the mod name hides or reveals all settings, while clicking a category name toggles the specific settings of that category.



## 3. LINK for mod authors

### 3.1 Introduction

Adding LINK support to new and existing mods is relatively easy. It requires creating a config (cfg) file and making a small change to the code to only load the ini file when needed. For more complex mods a custom event handler is available for any additional changes that might be necessary after the settings have been applied by the user.

**Note that LINK.esp should not be used as master file and that your mod will work like normal when LINK is not installed.**

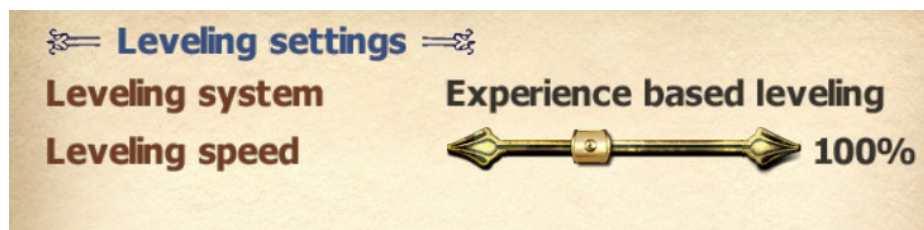
### 3.2 Cfg files

A config file is much like an ini file, but with the extension “cfg”. It contains information required to make ini settings configurable in the configuration menu.

There are 4 types of settings that can be configured, which are:

- headers;
- toggles;
- sliders;
- keybindings.

A header is a visual separator, between a group of settings. A toggle can switch between values, simply by selecting it. A slider provides an easy method to select a value from a range, like between 0 and 300, as shown below.



An example of a header in a config file would be as follows:

```
; TITLE
set LINK.cfg_setting_title to sv_Construct "Leveling settings"
set LINK.cfg_setting_type to 0
SetStage LINK 10
```

Setting the type to 0 defines it as an header, as shown in the image above.

The toggle is defined as follows:

```
; LEVELING SYSTEM
set LINK.cfg_setting_name to sv_Construct "ULVL.ini_leveling_system"
set LINK.cfg_setting_description to sv_Construct "Either pick an
experience based leveling system (like Oblivion XP), or a skill usage
leveling system (like Realistic Leveling)."
set LINK.cfg_setting_title to sv_Construct "Leveling system"
set LINK.cfg_setting_type to 1
set LINK.cfg_setting_options to 2
set LINK.cfg_setting_option1_value to 1
set LINK.cfg_setting_option2_value to 2
set LINK.cfg_setting_option1_description to sv_Construct "Skill usage
leveling"
set LINK.cfg_setting_option2_description to sv_Construct "Experience based
leveling"
SetStage LINK 10
```

The setting should be easy enough to understand, but for clarity, `LINK.cfg_setting_name` is the actual name of the variable in the ini file and `LINK.cfg_setting_description` is the text shown when hovering over the setting with a mouse. The above setting has 2 options, which can have the value of 1 or 2, as expected in the ini file.

A slider is a slightly more complex setting and could be configured as follows:

```
; LEVELING SPEED
set LINK.cfg_setting_name to sv_Construct "ULVL.ini_leveling_speed"
set LINK.cfg_setting_description to sv_Construct "By default leveling
speed is at normal speed. Set to any percentage, where higher is faster."
set LINK.cfg_setting_title to sv_Construct "Leveling speed"
set LINK.cfg_setting_type to 2
set LINK.cfg_setting_decimals to 0
set LINK.cfg_setting_percent to 1
set LINK.cfg_setting_min to 0
set LINK.cfg_setting_max to 300
set LINK.cfg_setting_step to 1
SetStage LINK 10
```

Sliders are of type 2. In the above example the slider has 0 digits after the decimal point and a “%” sign follows the value, as shown in chapter 3.2. In this case the range is between 0 and 300, with steps of no smaller than 1.

A keybinding can be configured as follows:

```
; TAKE
set LINK.cfg_setting_name to sv_Construct "LootMenu.TakeKey"
set LINK.cfg_setting_description to sv_Construct "Key used to take items."
set LINK.cfg_setting_title to sv_Construct "Take"
set LINK.cfg_setting_type to 3
SetStage LINK 10
```



Keybindings are of type 3. In the above example it defines the take key for the Loot Menu mod.

### 3.3 Adding support

Where normally the ini file is loaded, a small alteration is required. This because it should only be loaded when the ini file changed (because of an update, first time loaded, etc.), or when LINK is not loaded. An example:

```
if ( IsModLoaded "LINK.esp" == 0 ) || ( SKILL1.ini_version < 110 )
    RunBatchScript "Data\New skill - Unarmored.ini"
endif
```

In the above bit of code you can see that the ini file is loaded when either LINK is not loaded, or if SKILL1.ini\_version is less than 110 (v1.1). **This setting is set to the correct value in the ini file.** In this case the ini file contains the following bit of code at the end of the file:

```
;=====;
; Read-only settings ;
;=====;
; DO NOT CHANGE THE FOLLOWING SETTINGS.
set SKILL1.ini_version to 110
```

Note that the ini file is loaded in the main menu, when looking at the mods menu, which is why this setting in the ini file is necessary. It can (optionally) also be used to check if the user is indeed using the correct ini file and not trying to use an older version.



### 3.4 Optional configuration

After the user makes changes in the configuration menu and presses the “Return” button, the ini settings are updated. For more complex mods it might be necessary to respond to these changes, by editing leveled lists, updating the UI, etc. This can be done by using the `OnSettingsUpdated` event handler. For example:

```
if IsModLoaded "LINK.esp"
    SetEventHandler "OnSettingsUpdated" MyScript
endif
```

In this case the event handler calls the script called “MyScript”. This script should look something like this:

```
Scn MyScript

array_var args

Begin Function { args }

if eval ( GetModIndex "mymod.esp" ) != ( args->modindex )
    return
endif

; YOUR CODE HERE

End
```

The above code simply checks if the settings are indeed related to “mymod.esp” and if so, executes the code that might follow.

Note that when a mod consists of both an esp and esm file, both files should be checked. The script should then look like this:

```
if eval ( GetModIndex "mymod.esp" ) != ( args->modindex )
if eval ( GetModIndex "mymod.esm" ) != ( args->modindex )
    return
endif
endif
```