

Introduction

This is a guide how to convert a SimSettlements 1 plot addon to SimSettlements 2 using the script “SS2_PlotConverter.pas”.

This script can convert the following content:

- Regular building plans
- Entire branching building plans
- Building skins, if the corresponding plan is part of your addon
- Furniture for the Furniture Store
- Foundations, both regular and terraformers
- Settlement Flags
- City Planner’s Desks
- City Leaders

Requirements

Tools needed

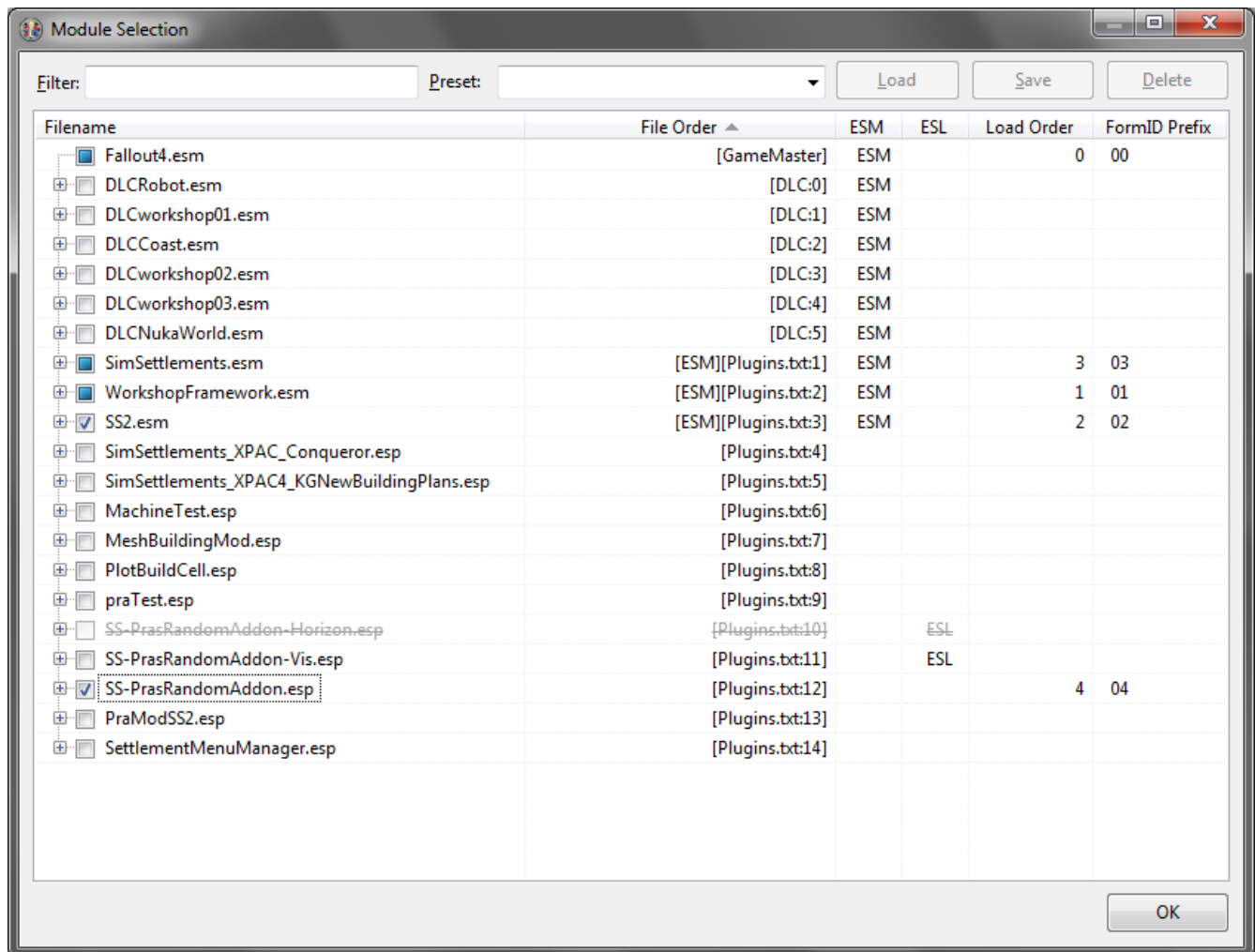
- Your Addon
- SimSettlements 1 and 2
- FO4Edit
- FO4Edit Scripts included in the SimSettlements 2 toolkit
- Creation Kit

The following scripts are relevant for the conversion process:

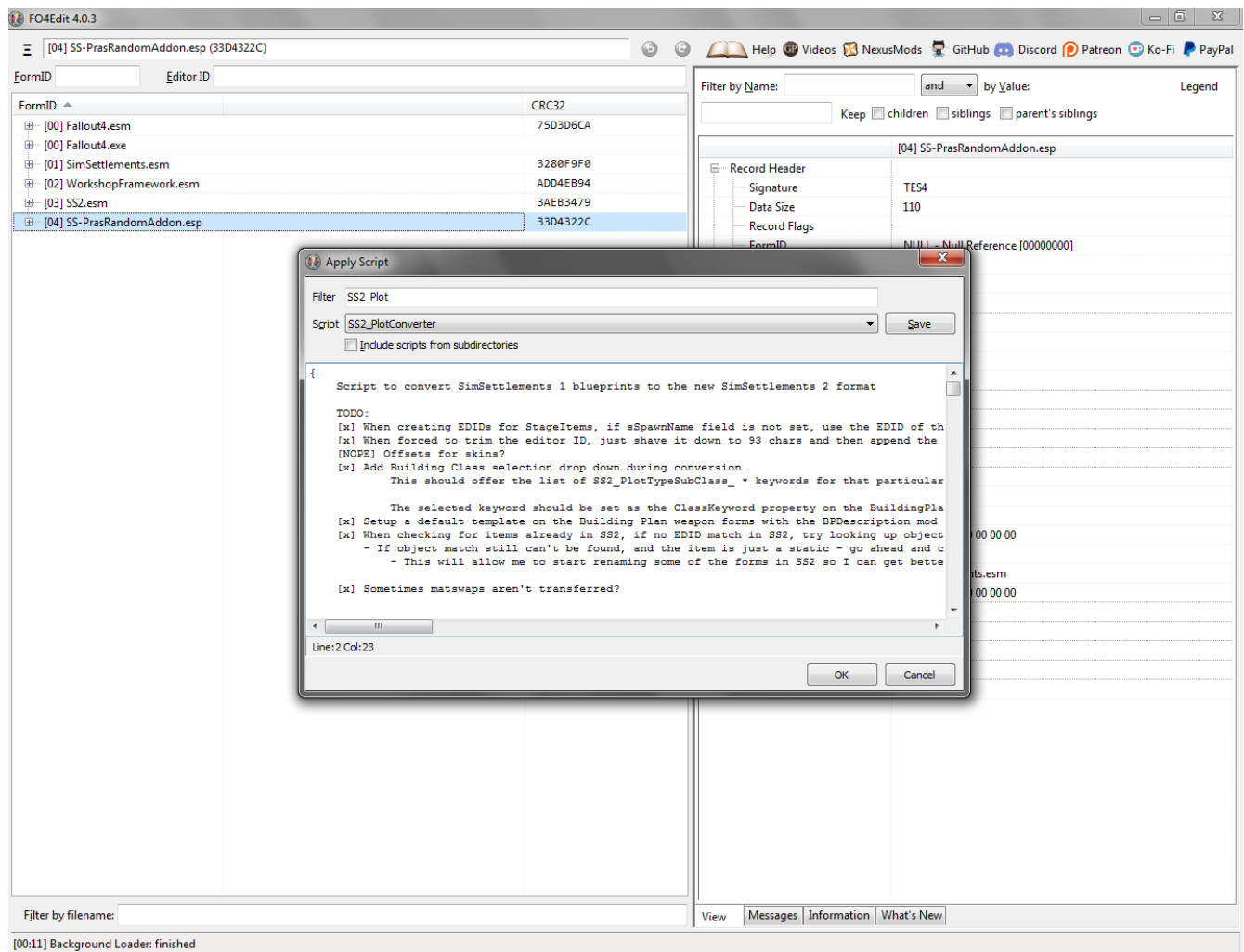
- SS2_PlotConverter.pas: This is the main script which you have to run on your addon.
- SS2Lib.pas: Contains code shared between various SimSettlements 2 scripts
- praUtil.pas: Contains code shared between various SimSettlements 1 and 2 scripts
- mteFunctions.pas: Contains code shared between various SimSettlements 1 and 2 scripts
- (optional) Search And Replace Form.pas: can be used to replace references to a specific form in a file
- (optional) SS2_TranslateForm.pas: can be used to force the conversion of a specific form to the new addon.

Converting

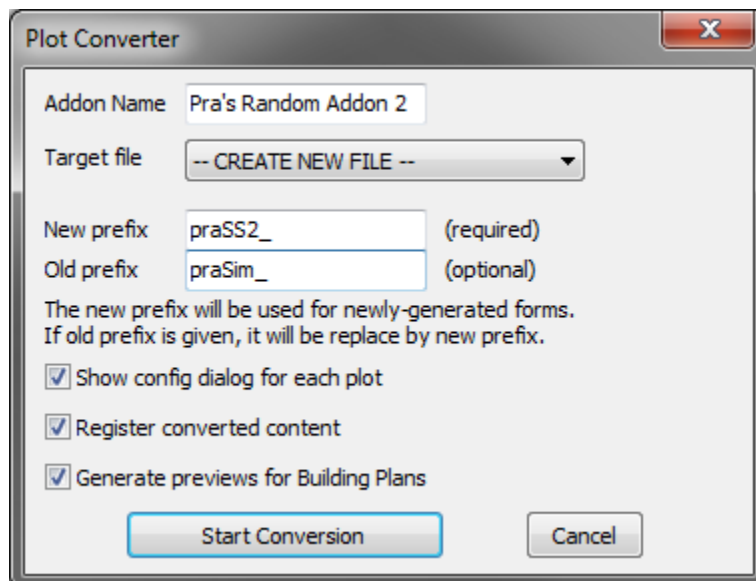
Start FO4Edit. Load your addon, and SS2.esm. Your addon should also load SimSettlements.esm.



Once everything loaded, right-click your addon, select “Apply Script...”. In the following dialog, select “SS2_PlotConverter” and click “OK”.



A new dialog will appear, asking you for various information:



“Addon Name” will be used to name the new addon quest and configuration form, but this isn’t too important.

“Target File” can be left at “– CREATE NEW FILE –”, in that case, you will be asked for the filename after starting the process. You can also select an existing file, in that case, plots you converted already will be updated.

“New Prefix” must contain something. It will be used as the prefix for EditorIDs of forms generated by the script.

“Old Prefix” can be filled out, if you want to change your mod prefix.

Note: I strongly recommend changing your prefix for Sim Settlements 2. If the script fails to do a full conversion right away and you have to use the Creation Kit (see below), you will have to load both your old and new addon at the same time. Records having the same prefix will be very confusing in this case.

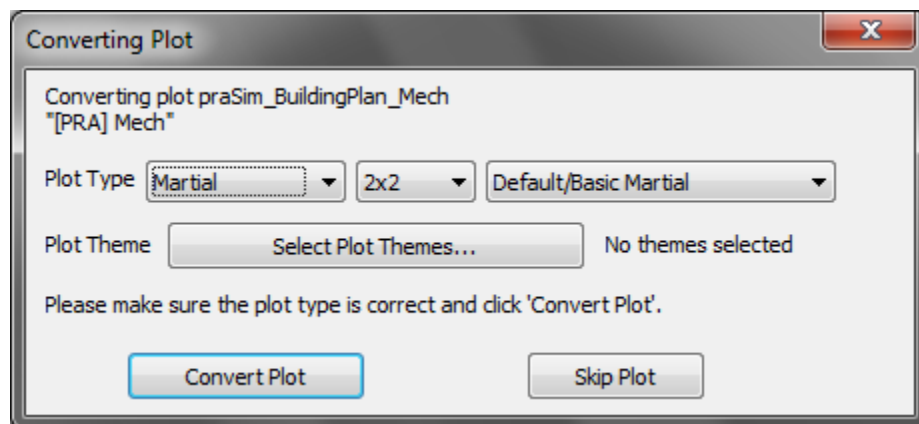
“Show config dialog for each plot” means, that for each plot, a dialog will be shown for each plot. See below for more information.

“Register converted content” will make the script register everything within the addon quest (and generating this quest automatically). You want to keep this enabled, unless you are doing in-game plot unlocking, or similar.

“Generate previews for Building Plans” will assign the building plan’s first level model as preview in the building plan selection. This might, maybe, cause issues on low-end systems, like the Xbox.

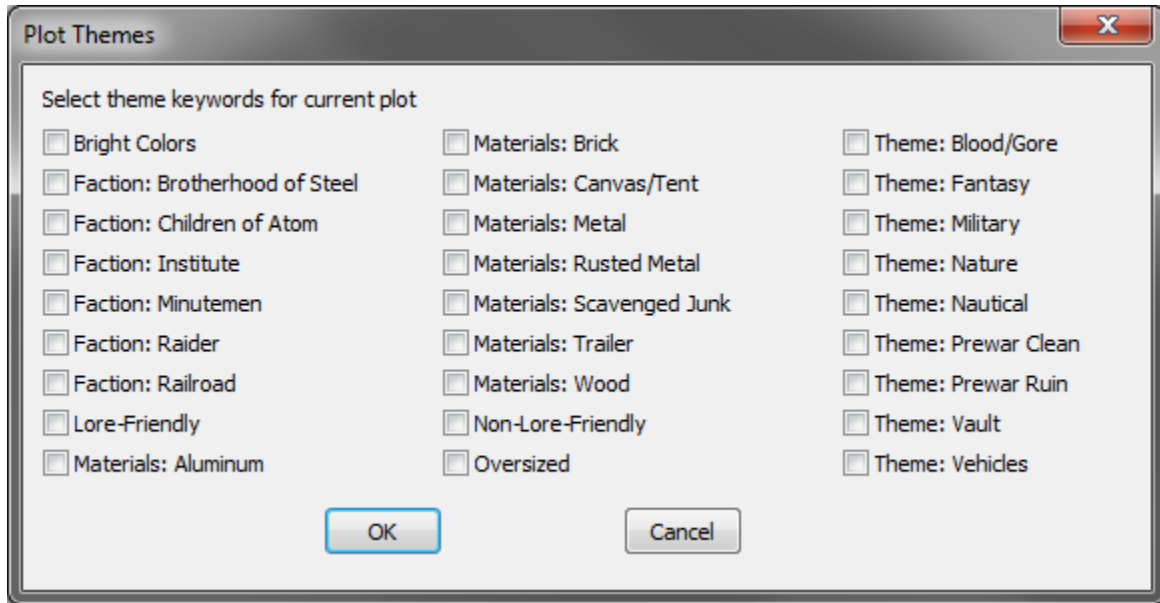
Converting plots

If you checked “Show config dialog for each plot”, or if the script has trouble determining the plot’s type automatically, a special dialog will appear for each plot. Here you can configure it’s type, size, subtype, and optionally theme keyword. Otherwise, the script will use the same type and size as in SimSettlements 1, and the default subtype.



In most cases, you probably don’t want to change these settings. However, you might consider classifying some of your Industrial or Recreational plots as Municipal.

The Plot Theme selection is optional, and can be used to select any number of theme keywords:



Click “Convert Plot” to begin the conversion process, or “Skip Plot” to leave this one out.

After Conversion

If everything worked right, you will now have a new file, with the masters “Fallout4.esm”, “SS2.esm”, and potentially any DLCs. This should work right out of the box.

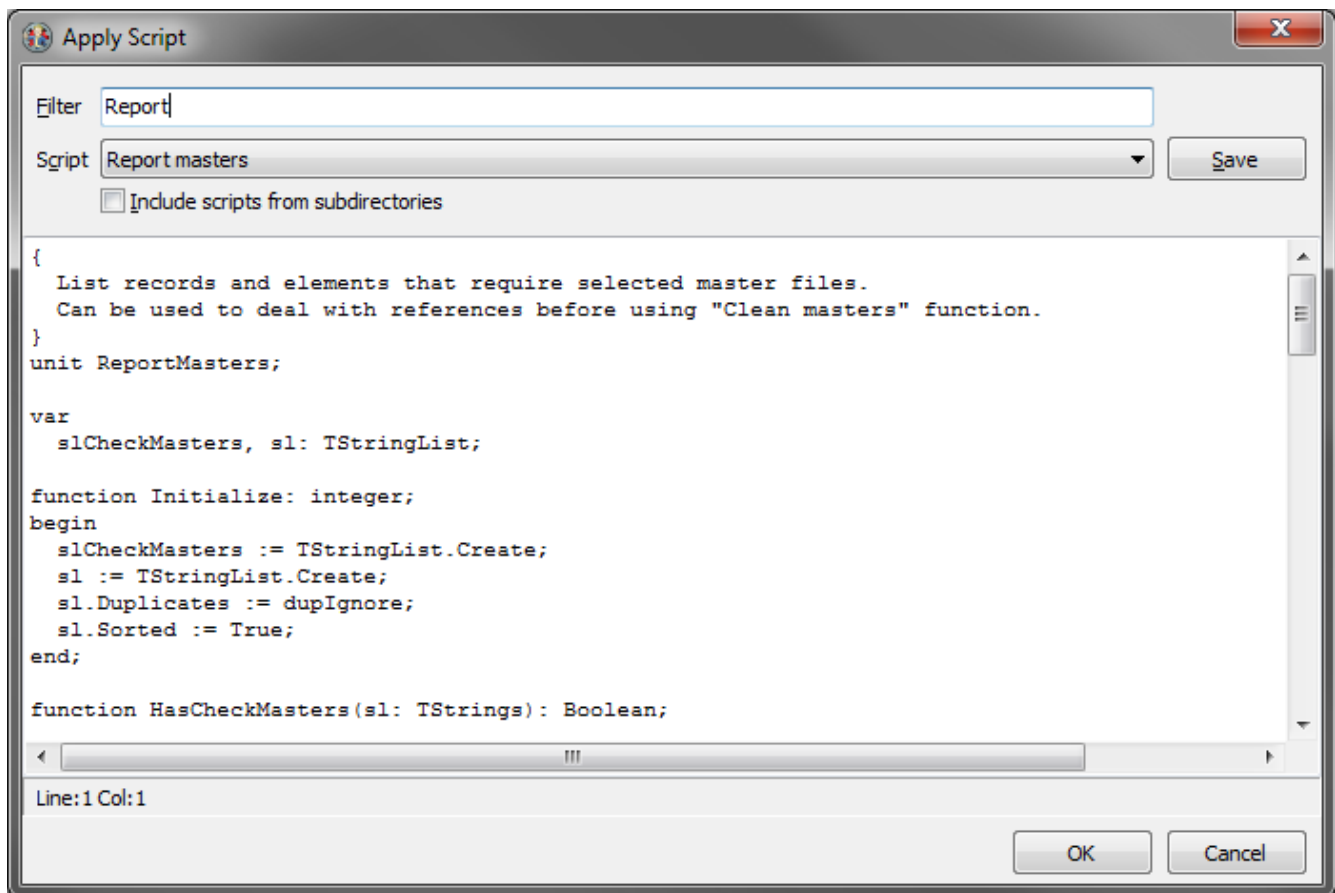
However, if the target file has “SimSettlements.esm” and your old AddOn as a master, your work isn’t done.

There are limitations to what a FO4Edit can do. Among the things it can’t do, is copying entire quests, or cells. Therefore, references to quests, cells, or placed objects, are not converted, but left as is.

Cases where manual conversion is necessary:

- “Bigger on the inside” plots, where an interior cell is referenced
- Custom vendors
- A custom management quest with custom code
- An actual quest which is used in a plot

If you are not sure which parts of your old AddOn or of SimSettlements 1 the new AddOn is using, you can use the “Report Masters” script (part of the default FO4Edit installation):



The output will look somewhat like this:

```

[00:00] Start: Applying script "Report masters"
Selected masters are required by the following records and elements:
praSS2_MarkerSouvenirVendorCounter [FURN:05000FC8] \ VMAD - Virtual Machine Adapter
praSS2_WorkbenchPowerArmor_AutoLink "Power Armor Station" [FURN:05000E65] \ VMAD - Virtual Machine Adapter
praSS2_WorkbenchWeaponsMicro_AutoLink "Weapons Workbench" [FURN:05000E63] \ VMAD - Virtual Machine Adapter
praSS2_WorkbenchUtilityMicro_AutoLink "Hardware Workbench" [FURN:05000E60] \ VMAD - Virtual Machine Adapter
praSS2_workbench_Electronics_TerminalTextScrolling "Electronics Workbench" [FURN:05000E5D] \ VMAD - Virtual Machine Adapter

```

“VMAD – Virtual Machine Adapter” means, there is a script property which needs updating. This should be the usual case.

“NAVI – Navigation Mesh Info Map” is something the Creation Kit keeps adding. It will have the Form ID 00000FF1 and be the only entry in it’s group. You can delete it.

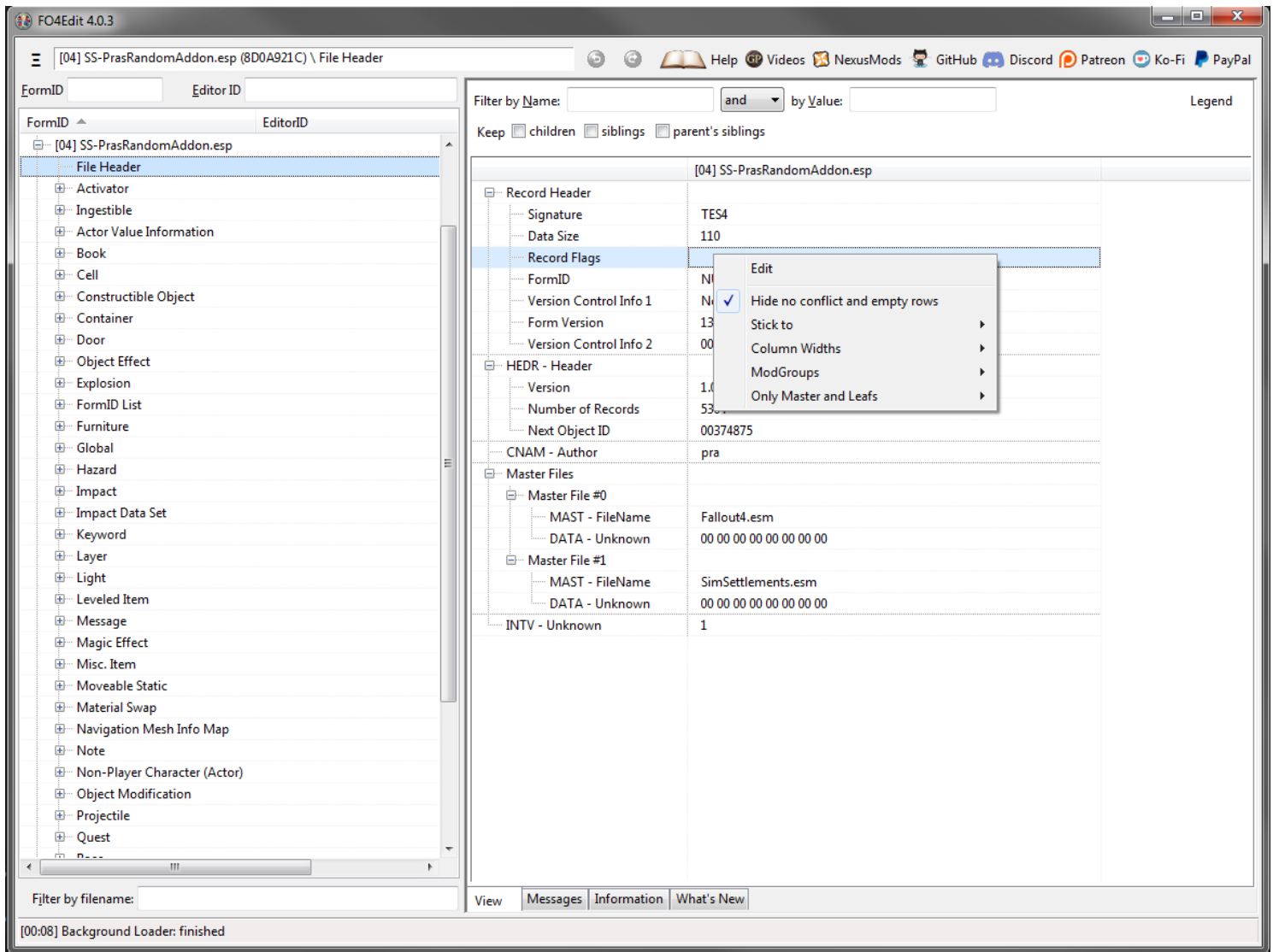
WARNING: If you save your new addon in the Creation Kit while it still refers to your old addon, it will remove it as a master, invalidating all references to it!

You must either do as much work in FO4Edit as possible first, or flag your old mod as an ESM, before starting the CK.

To flag a file as ESM, select it’s File Header, right-click the Record Flags, and select Edit:

In the following dialog, simply check “ESM”. Or, uncheck it, if you want to undo it.

After this, it should be safe to open your converted AddOn in the CK, while it is still referencing the old version.

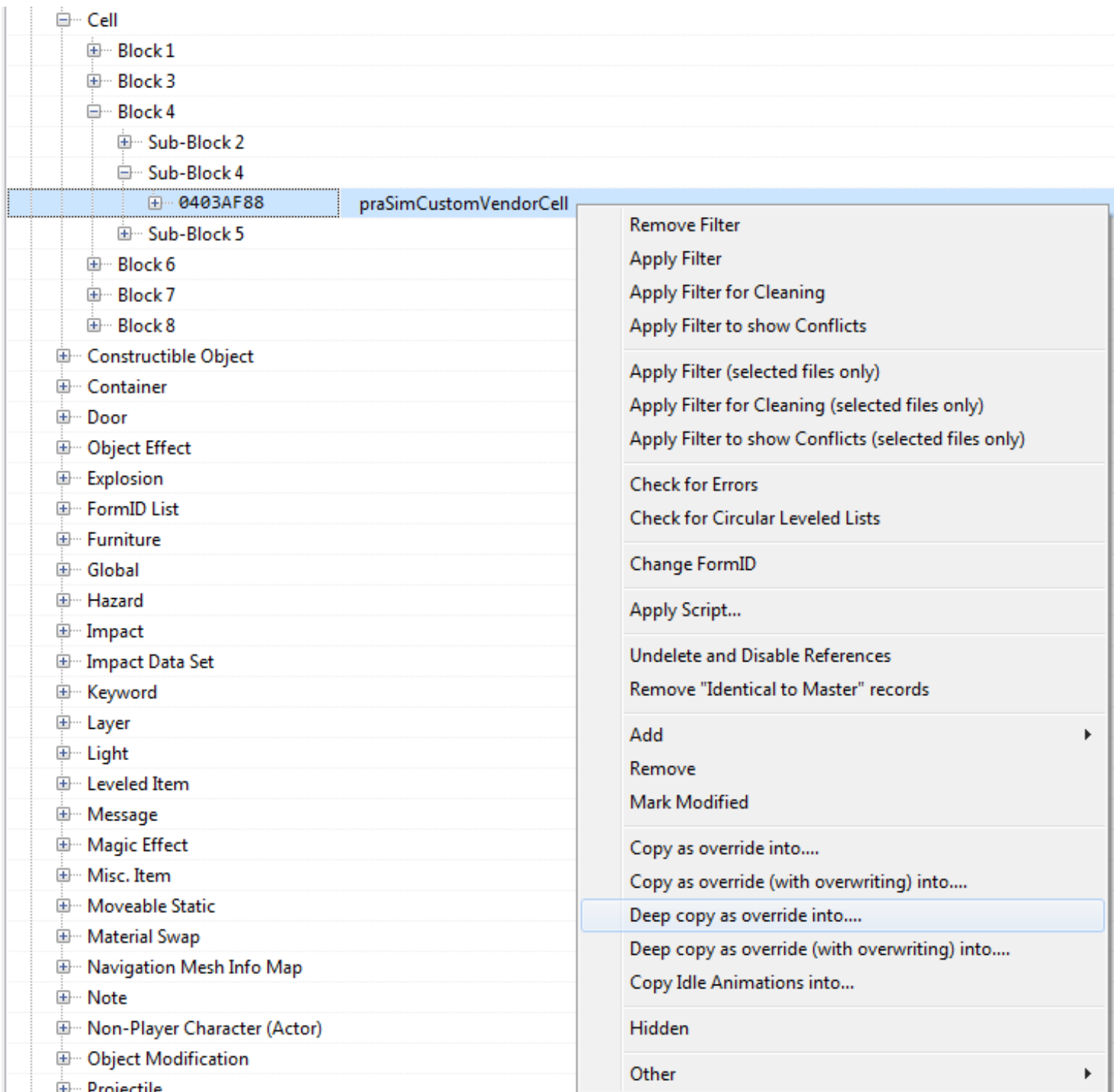


Copying Cells

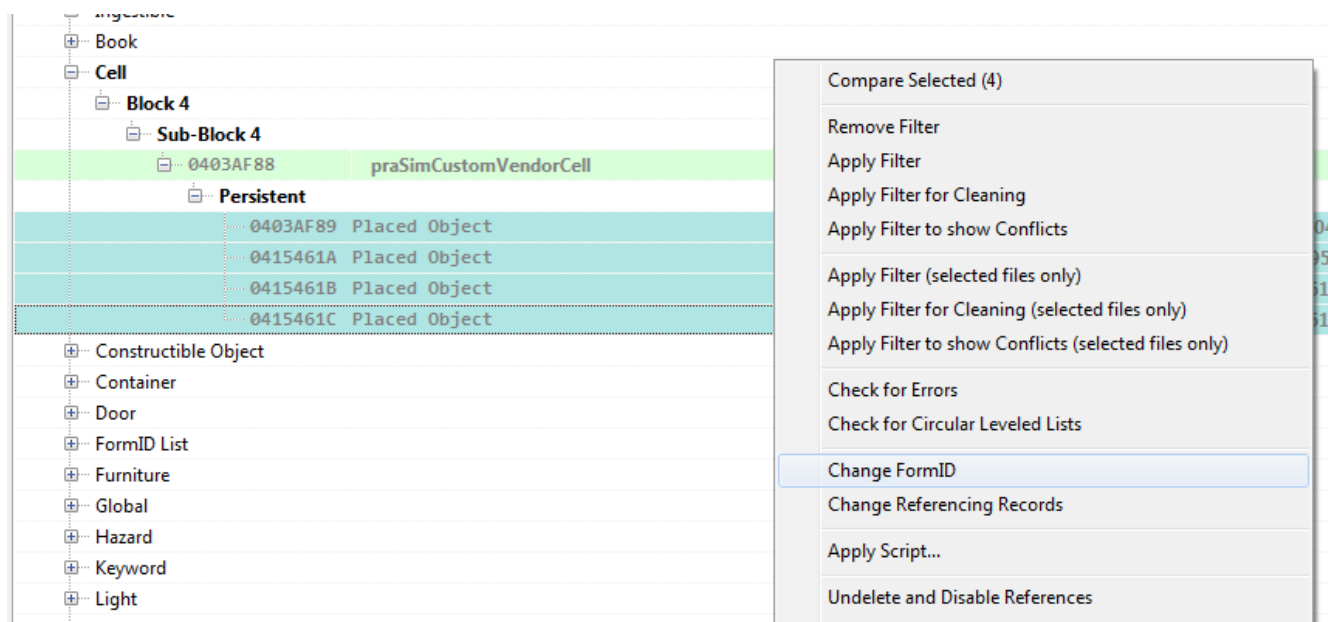
There are two possibilities to copy a cell, using either FO4Edit or the CreationKit. Both of them have their own pitfalls.

Using FO4Edit

Right-click the cell in question, select “Deep Copy as override into...”



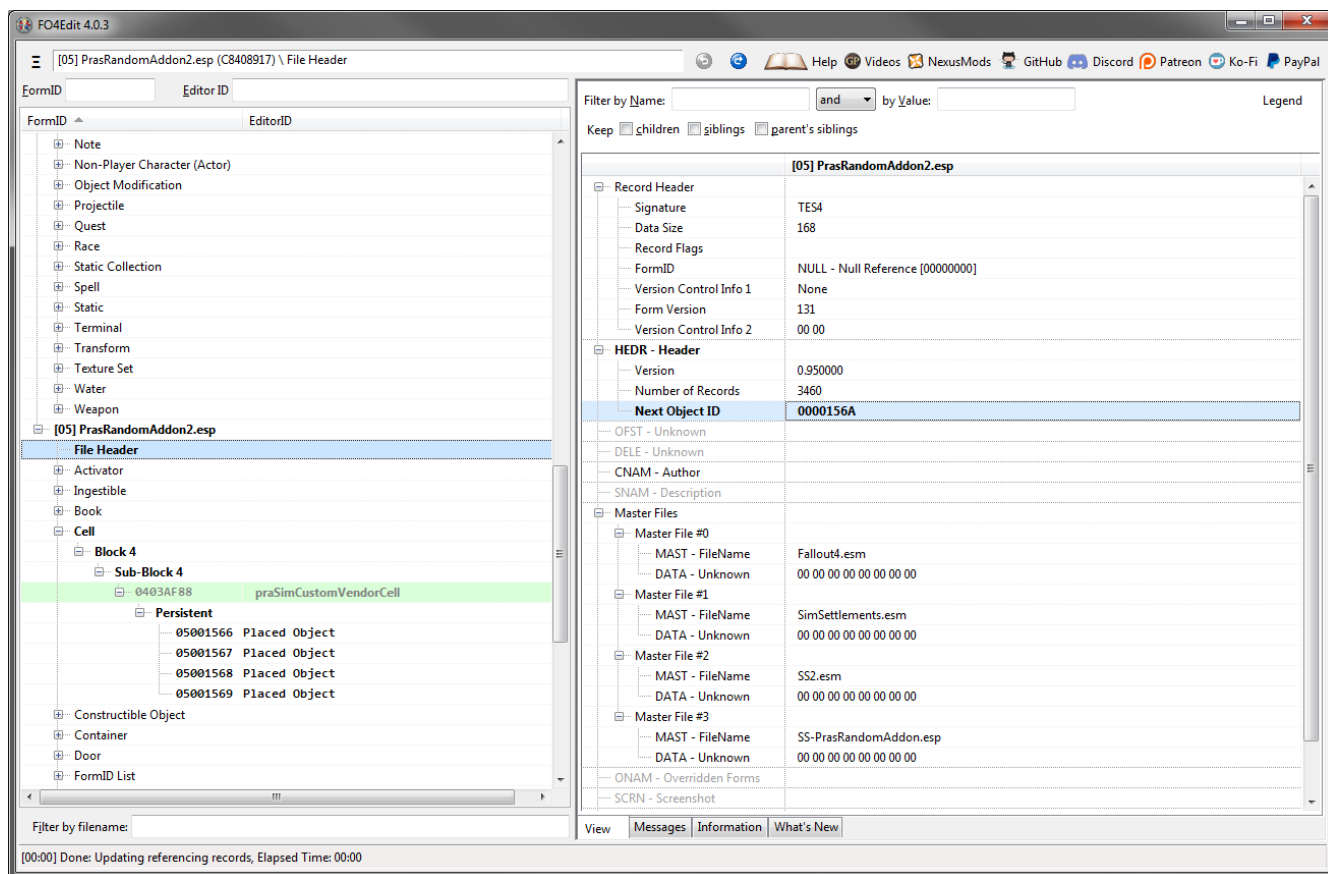
After that, go to the override it just created, select all references in it, right-click, and select “Change Form ID”:



Keep in mind, you must do this for both Persistent and Temporary, if both are present within the cell.

You will be asked to select a target file. Select your addon again. This will automatically turn these overrides into new records, and update references within your addon.

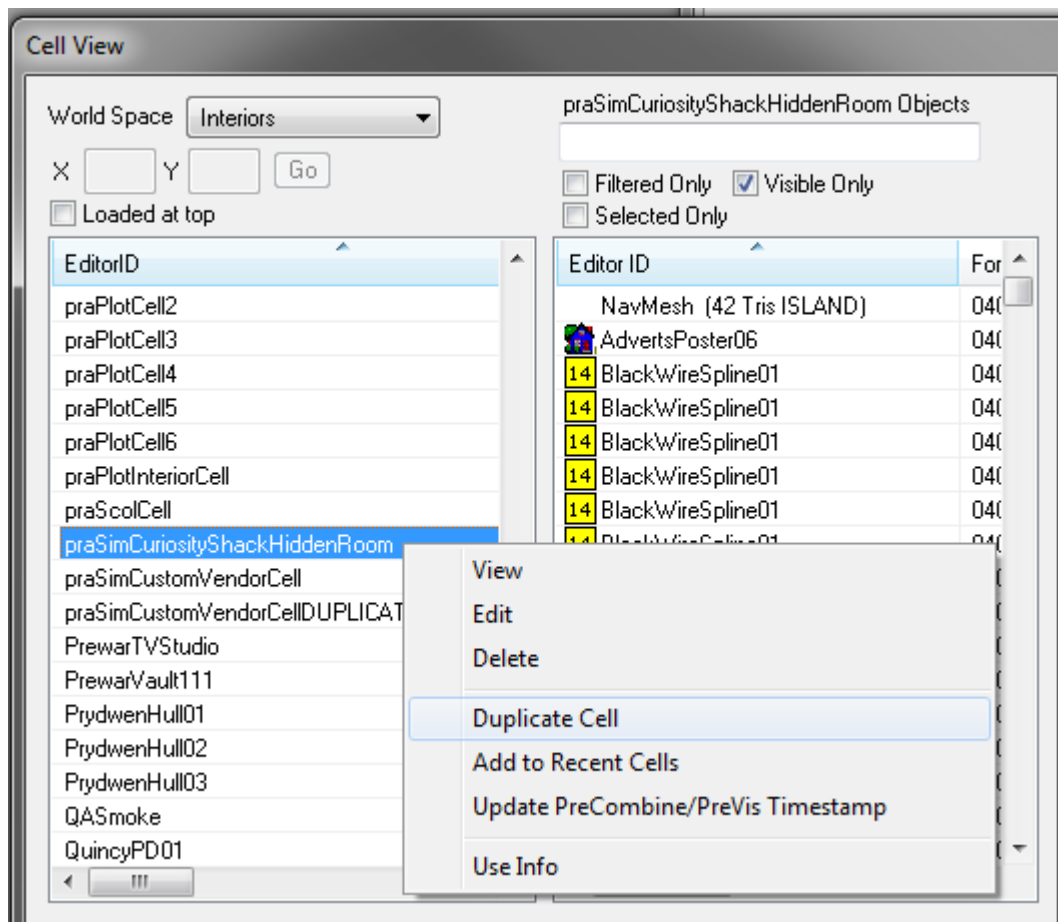
After that, you need to change the cell's Form ID as well. This cannot be done automatically, you must put in a new one manually. Fortunately, the next available one can be looked up in your file's header:



The important part here is to use the same Load Order prefix, 05 in this case. This turns an override into a new record. Therefore, here, the new Form ID must be 0500156A.

Using the Creation Kit

Load your AddOn in the CK, find your cell, select “Duplicate Cell”:



you can then press F2 to change the new cell’s EditorID.

In this case, you must know yourself what exactly is referencing that cell, and change it manually.

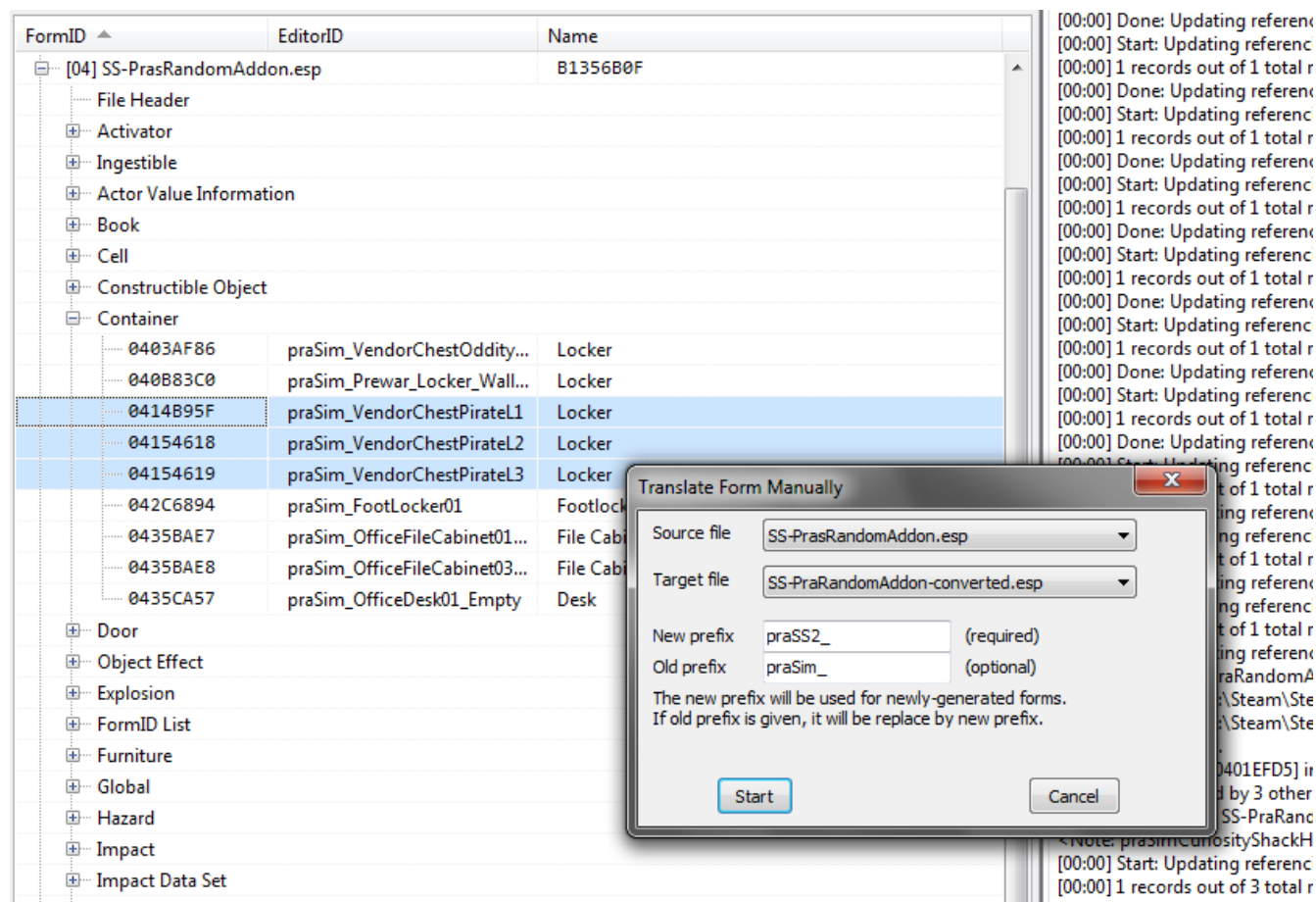
Also, keep in mind that “Duplicate Cell” is inherently dirty, as it is always creating an override of the source cell in your mod. You must clean it in FO4Edit afterwards.

If you have custom objects placed in the cell, these must be replaced manually. See “Converting References” or “Replacing Forms” below.

Converting References Manually

The conversion script tries to find all forms required by a building plan, and to migrate them to the new file. However, in cases where the form in question isn’t referenced by any building plan, this mechanism will fail.

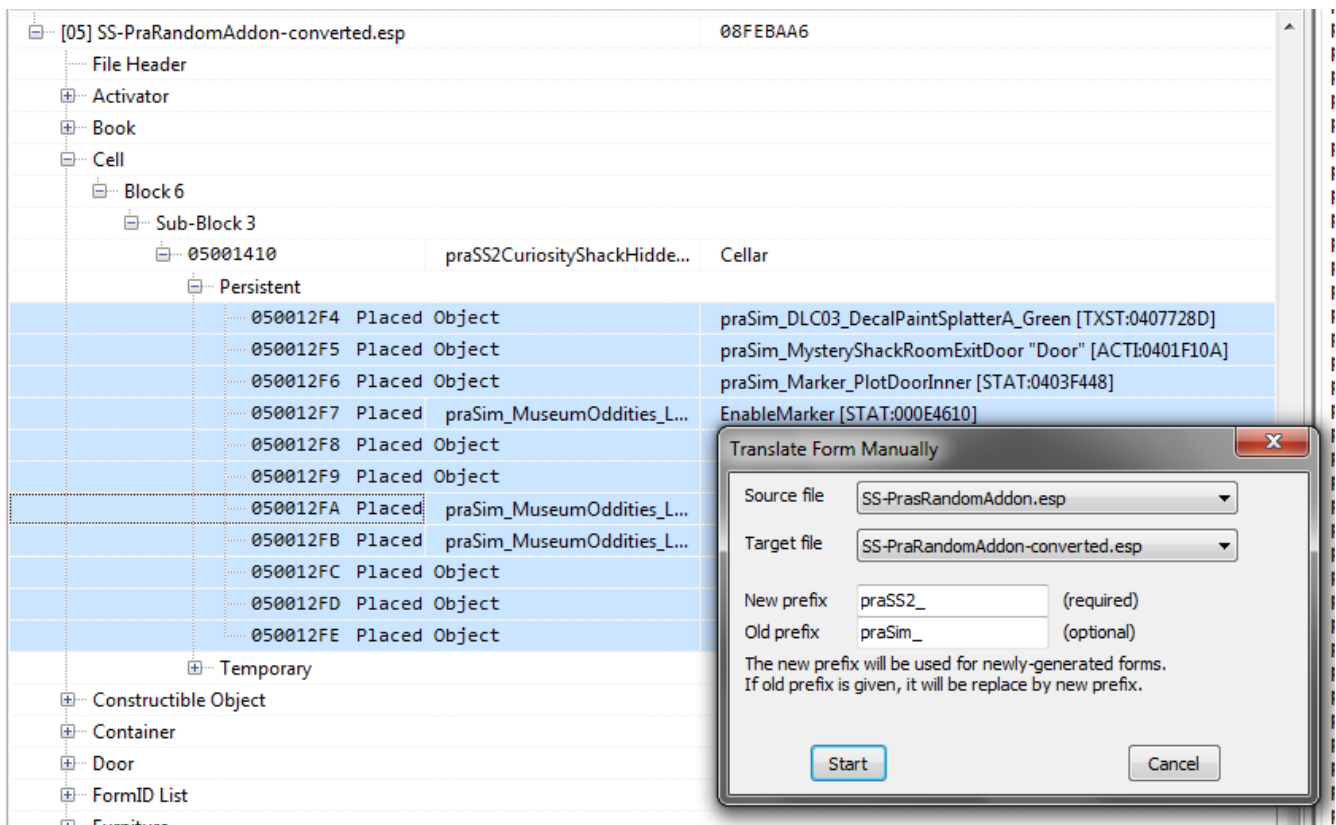
If you know that you need a specific form in your new addon, you can use the script “SS2_TranslateForm.pas” to copy it over manually. Simply run it on any number of forms in your old addon:



It's important to select your old addon under “Source file” and the new version under “Target file”. If you ran the plot converter script before, this dialog should be pre-filled with the last settings from there.

Similar to the plot converter, not only will this copy the selected forms into your new addon, but also any forms required by them.

The script can also be run on forms within your new addon, which are referencing forms from the old one. For example, Object References in a cell after copying it:



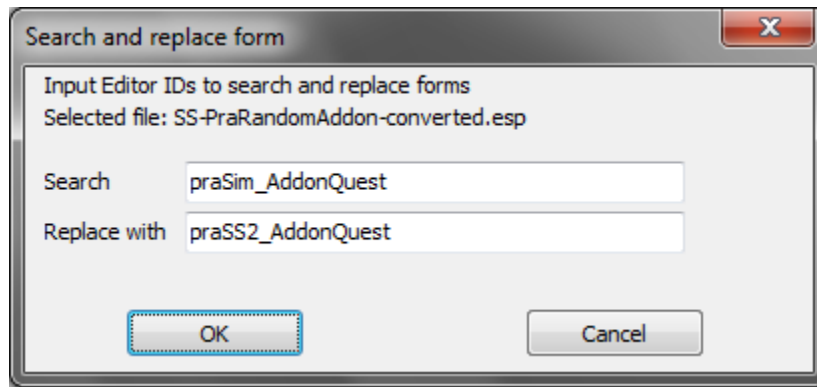
In fact, the script can be safely run on the entire cell itself, as long as it and it's contents exist within the target file, and are not overrides.

Note: this script can **not** convert quests, cells, or references. However, it should be able to update these types of records, once they exist within a target file.

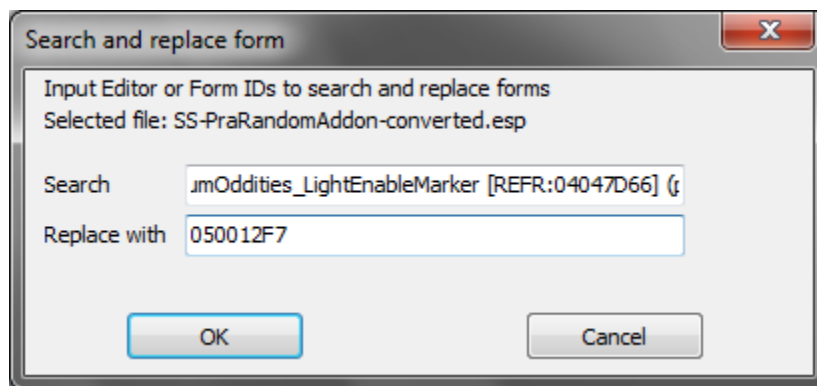
Replacing Forms

The script "Search And Replace Form.pas" can be used to replace all uses of a form within a file with a different one. This can be useful if you have migrated content like a quest or cell manually, and want to update script properties or similar accordingly.

Run the script on your new addon's file (or anything within it), and put in the Editor or Form ID of the old form under "Search", and of the new form under "Replace with".



Valid values for the two fields are either the Editor ID, the hexadecimal Form ID, or FO4Edit strings like “Toilet01 [ACTI:000248C2]”:



Note: for cells or object references, you must use the FormID or the FO4Edit string. It will not work with just an Editor ID.

Custom Vendors

Thanks to WorkshopFramework, SimSettlements 2 now has native support for custom vendors. This is best done using the CreationKit. Using FO4Edit is generally possible, but far from trivial.

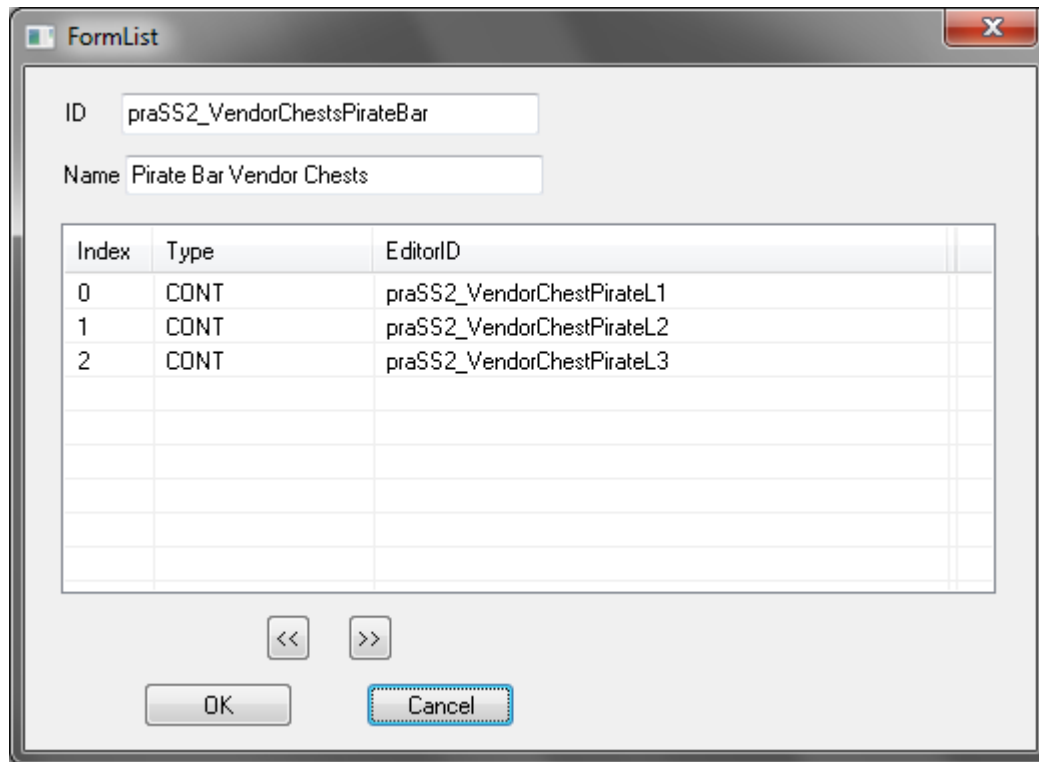
There is also a video tutorial for using WorkshopFramework’s custom vendors, which can be helpful: <https://www.youtube.com/watch?v=RJJzIZjRoLY>

An important difference to that tutorial is that you don’t need to create a new quest. The Sim Settlements 2 Addon Quest has a property for custom vendor registration already.

As opposed to Sim Settlements 1, you no longer need a holding cell with your vendor containers actually being placed. They will be spawned in on demand automatically.

1. Formlist

First thing a custom vendor needs is a formlist. It must have 3 entries, for the 3 levels. Each entry must correspond to a Container record:



Index	Type	EditorID
0	CONT	praSS2_VendorChestPirateL1
1	CONT	praSS2_VendorChestPirateL2
2	CONT	praSS2_VendorChestPirateL3

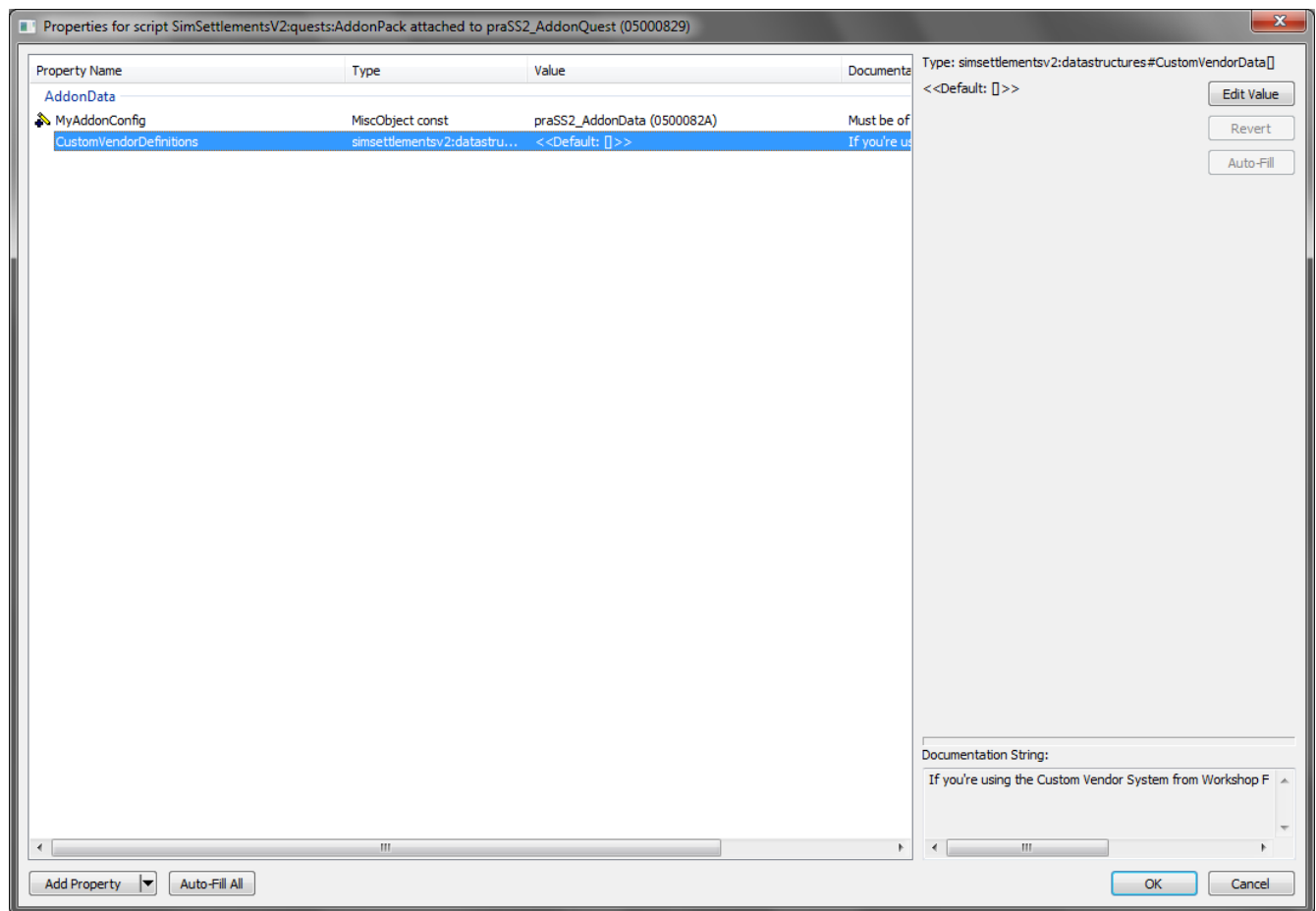
You do not need to fill out the name. It's just a note for yourself.

Note: The script will not copy the containers to your new AddOn. You have to copy them yourself.

Note: This formlist **must** have 3 entries. If you have less containers than that (for example, a plot which only gains a vendor at level 3), you must create and add empty dummy containers.

2. AddOn Quest

Go to your addon quest's scripts tab, and open the AddonPack properties:

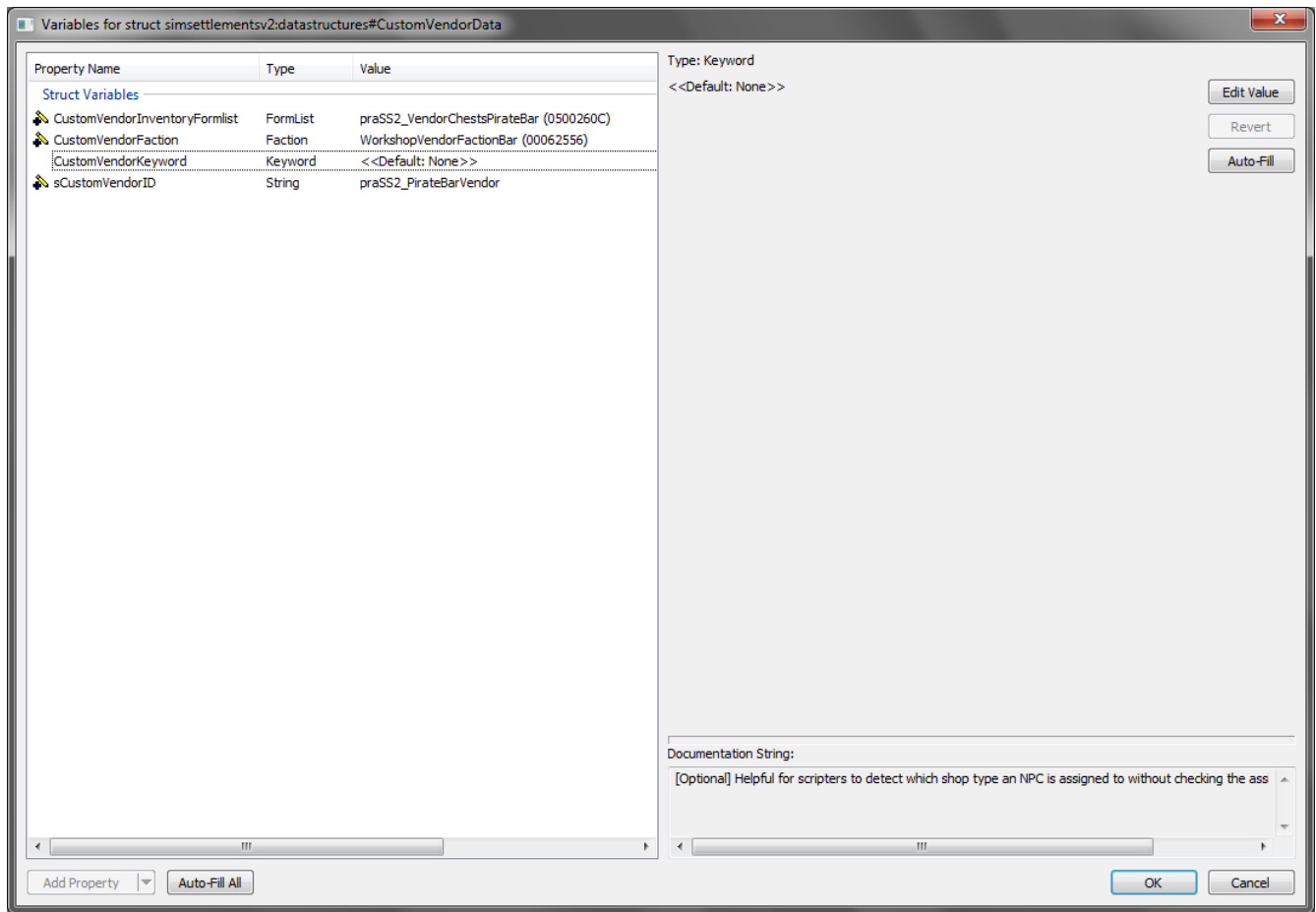


For each custom vendor you have, you must add a new entry in CustomVendorDefinitions and fill at least these two properties:

- CustomVendorInventoryFormlist: the formlist you created before
- sCustomVendorID: a string to identify your vendor. **This must be unique across all installed mods.** For this reason, it is best to add your prefix to this ID.

Additionally, you can fill out CustomVendorFaction. This will change how the vendor talks to you. For example, if the custom vendor is a bar, use WorkshopVendorFactionBar. You can find the rest in the CK under Factions, if you filter by “WorkshopVendorFaction”. If left blank, it will default to WorkshopVendorFactionMisc (General Store).

In the end, the entry should look similar to this:



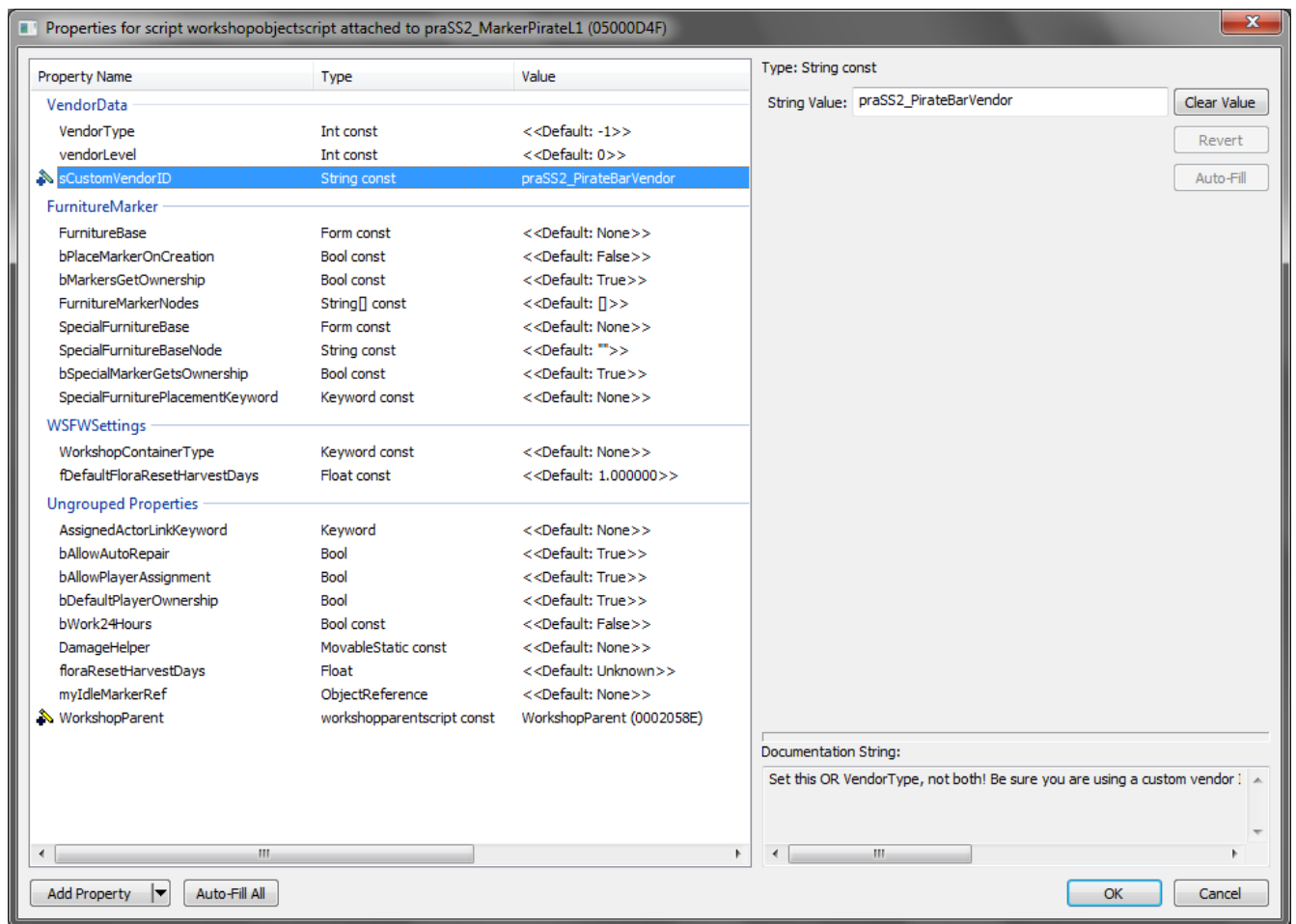
3. Furniture Object(s)

On your furniture objects, remove the custom vendor script you were using, add the generic workshopobjectscript to it. As usual, click auto-fill all to fill WorkshopParent.

The important parts here are:

- sCustomVendorID: same as you used in the quest
- VendorType: this **must not** set. If it has any value, select it and click on Clear Value.
- VendorLevel: this works as in vanilla. 0 corresponds to Level 1, 1 to Level 2, 2 to Level 3.

A Level 1 vendor marker's script should look like this:



Custom Quests

For custom quests, there is no simple solution. You will pretty much have to recreate it manually. Good luck.

Custom Scripts

In general, what goes for quests also goes for scripts. In most cases, you will have to tweak/recreate them by hand. Again, I recommend using a different namespace/prefix for the new versions.

However, here, the Papyrus Script Manager (under “Gameplay”) can help you a lot. Put in your old addon prefix/script namespace, and sort the list by Use Info:

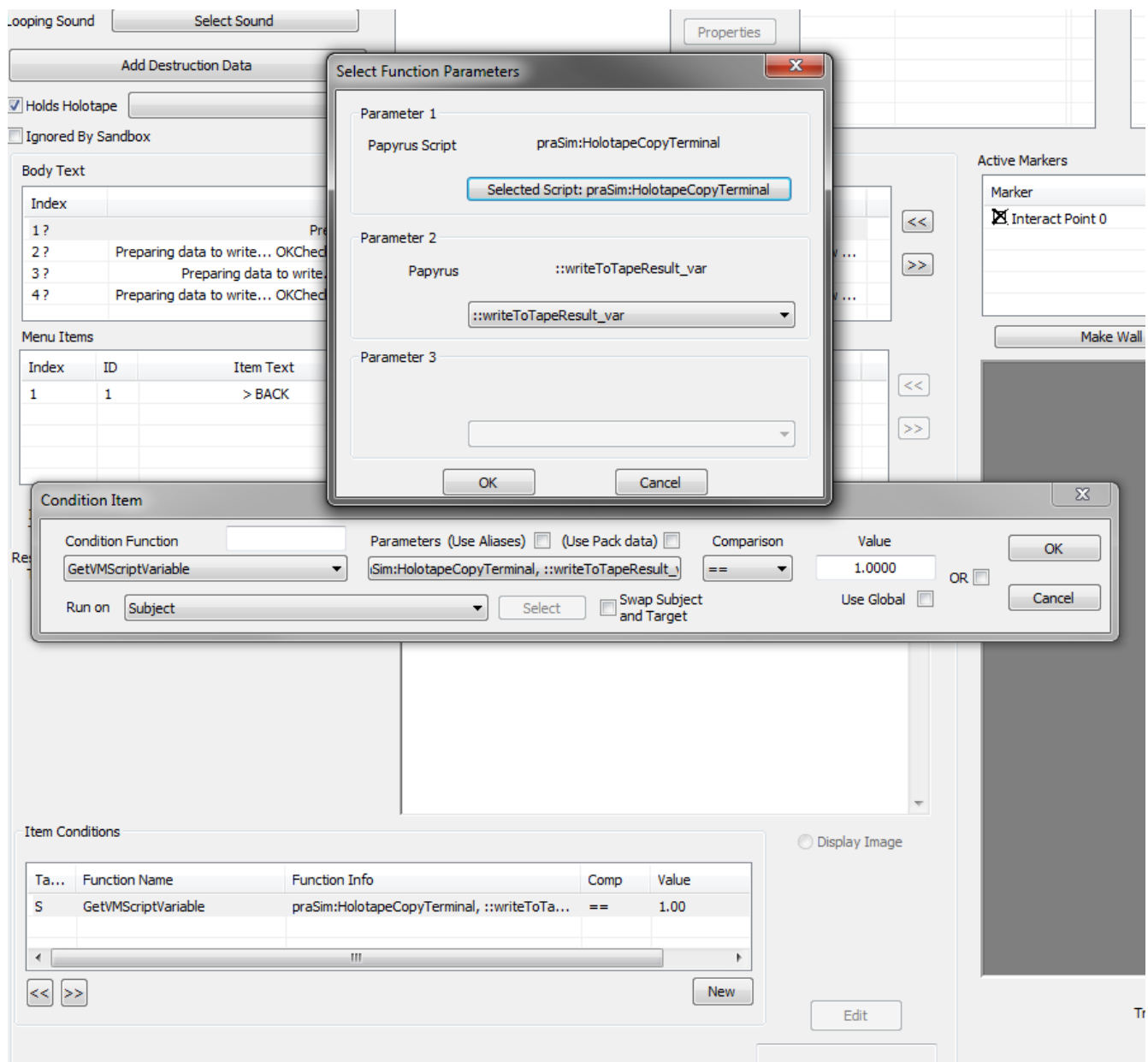
Papyrus Manager

praSim

Script Name	Compiled	Has Source	Use Info	
praSim:PlotSubSpawnerMultistage	No	Yes	2	
praSim:HolotapeCopyTerminal	Yes	Yes	1	
praSim:ConditionSpawnerMultistage	Yes	Yes	1	
praSim:ConditionSpawner	Yes	Yes	1	
praSim:PlotCellExitDoor	Yes	Yes	1	
praSim:RandomPlaySound	Yes	Yes	1	
Fragments:Terminals:TERM_praSim_CityPlannersLapt_02371860	Yes	Yes	0	
pra: Fragments:Terminals:TERM_praSim_HolotapeCopyTerm_0513E807	Yes	Yes	0	
praSim:BasicLadderExitSideways	Yes	Yes	0	
praSim:PlotCellEntranceDoor	Yes	Yes	0	
praSim:MuseumOdditiesExitDoor	Yes	Yes	0	
praSim:PlotSubSpawner	Yes	Yes	0	
praSim:WallTurretScript	Yes	Yes	0	
praSim:PiratedArcadeTerminal	Yes	Yes	0	
praSim:DiscoveryNote	Yes	Yes	0	
praSim:DeployableCityPlannersDesk	Yes	Yes	0	
praSim:VaultExerciseFurniture	Yes	Yes	0	
praSim:AutoLinkingWorkbenchScript	Yes	Yes	0	
praSim:CityPlannerTerminal	Yes	Yes	0	
praSim:HolotapeDummyScript	Yes	Yes	0	
praSim:AddonHelperScript	No	Yes	0	
praSim:PirateCompatUpdater	Yes	Yes	0	
praSim:CustomMenderScript	Yes	Yes	0	

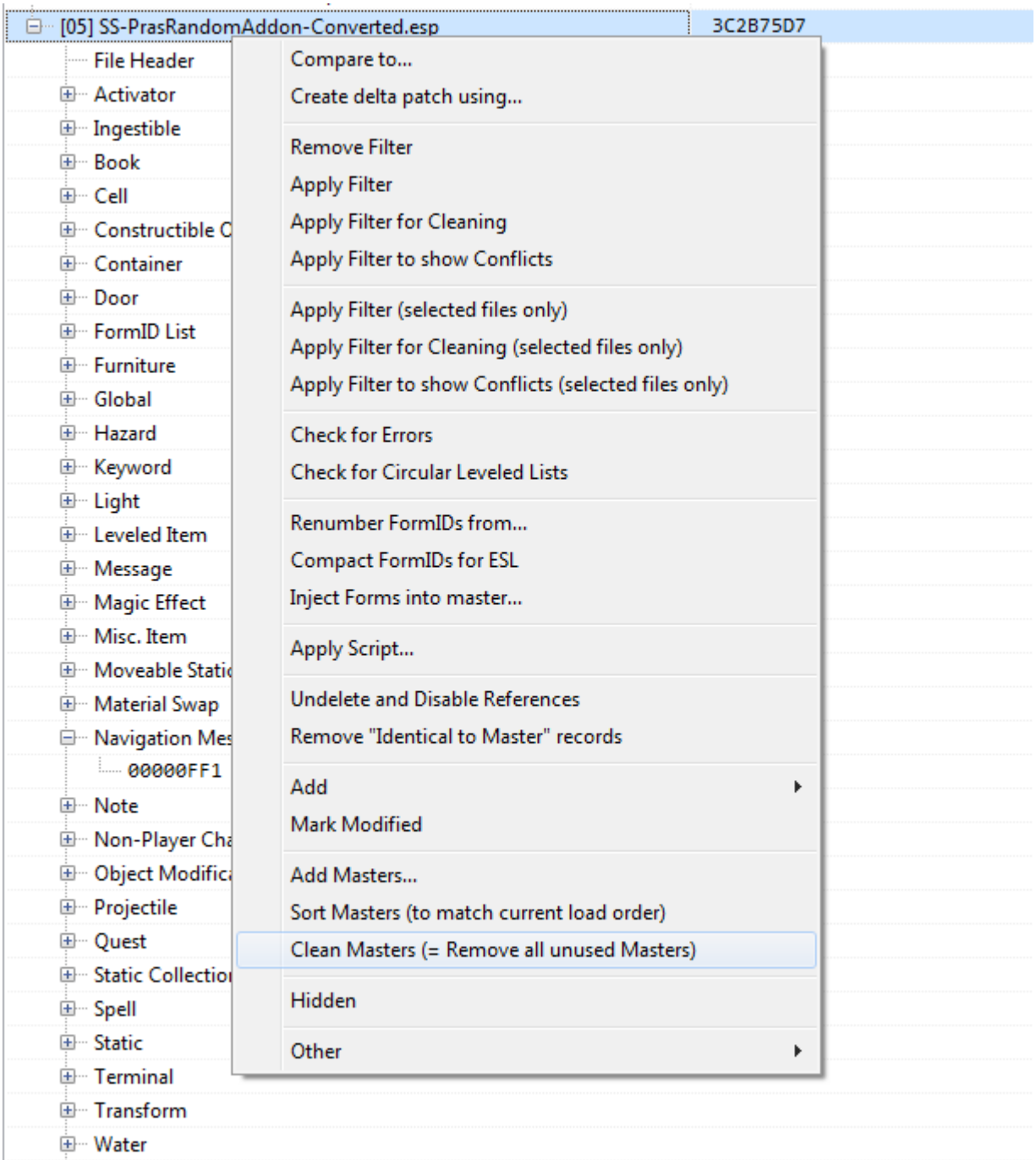
Now, you can right-click any script in this list, and select “Use Info”. It will show you some of the instances where the script in question is still being used.

Warning: unfortunately, this feature is not as reliable as it should be. It is not able to detect if a script is used in a Condition which uses GetVMScriptVariable. Below is an example of a condition in a terminal, where the script manager was not able to detect usage:



Cleaning Masters

After you copied or removed everything, load the file in FO4Edit once again. Right-click your file, select "Clean Masters (= Remove all unused Masters)".



If you did everything correctly, SimSettlements.esm and your old AddOn should have disappeared from the header.

Post-Conversion Tweaks

There are some new features in SimSettlements 2 which you might find useful. There are also some changes to existing systems which you might want to tweak.

Powered Objects

New Power System

In Sim Settlements 2, no special treatment for plot-powered objects is required anymore. Any object spawned by a plot with the WorkshopCanBePowered should work automatically at engine level. This means, scripting functions like isPowered() or power-related events should work as with regular workshop-built objects.

However, objects in converted plots will still be using the old, animation-based mechanism of being plot-powered, for maximum compatibility.

If you want to use the new system, simply remove “kgSIM_PlotPowered” from the object, and replace it with “WorkshopCanBePowered”.

Day/Nighttime lighting system

Another new feature is the ability to make objects switch on or off depending whenever it's day or night.

To use it, the object must be using the new power system, and then have either the “SS2_Tag_NighttimeLight” or “SS2_Tag_DaytimeLight” keyword.

Building Plan Description

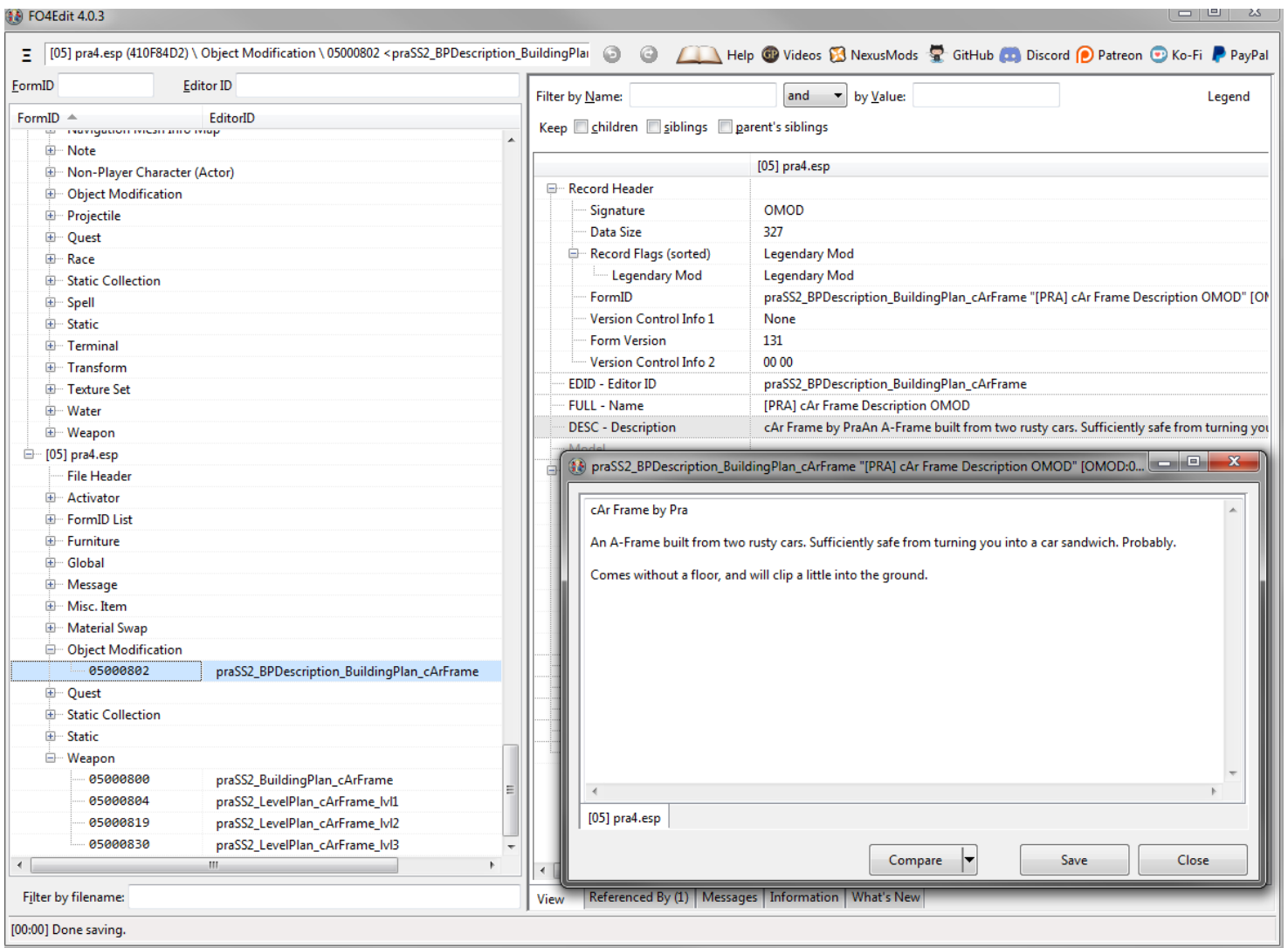
The description is now shown in the building plan selection. If it's too long for the screen, or you want to modify it for another reason, you can find them under Object Modification. The field you need to modify is the description.

Constructible Objects

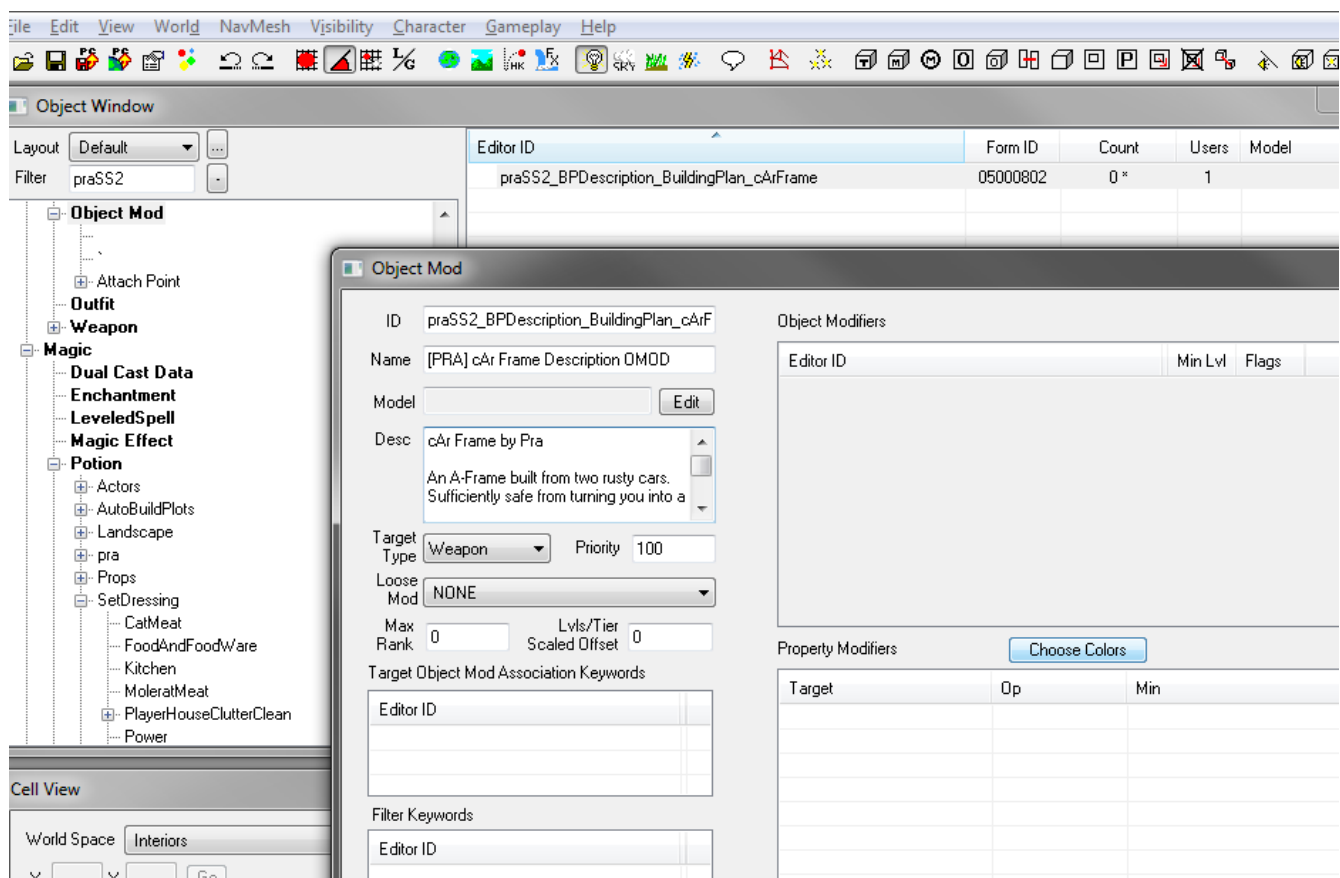
If you have custom constructible objects in your mod, in most cases, you can convert them using SS2_TranslateForm.pas. The script should be able to pick previously-translated keywords, globals, crafting results etc automatically, as long as you use the same prefixes in the dialog box.

Keep in mind that you do not have to manually convert constructible object records for foundations, furniture store items, or city planner's desks. These should have been generated automatically.

Furthermore: You no longer need to take care of Settlement Menu Manager. Since SimSettlements2 uses WorkshopFramework, menu management or repair happens automatically.



Using Fo4Edit



Using the Creation Kit

In-Game Testing

If your Building Plan is registered with SimSettlements2, you can just select it via the ASAM sensor, click the plot in the console, and use

```
cf unlocklevel
```

This works on unassigned and unbuilt plots as well. Each call will make the plot go up one level, going through the construction stages.

If it is not registered, you can still test it using it's form ID. To find it, open the console and type:

```
help "my plot name" 4
```

This should generate an output like this:

```
help "bunker elevator" 4
----VIEWER STRINGS NOT AVAILABLE-----
----OTHER FORMS-----
OMOD: (0A0010BD) '[PRA] Bunker Elevator Description OMOD'
MSG: (0A0010BE) '[PRA] Bunker Elevator Confirmation MSG'
WEAP: (0A0010BB) '[PRA] Bunker Elevator'
WEAP: (0A0010BF) '[PRA] Bunker Elevator Level 1'
WEAP: (0A0010DC) '[PRA] Bunker Elevator Level 2'
WEAP: (0A0010FC) '[PRA] Bunker Elevator Level 3'
```

The relevant one is the WEAP without the Level X suffix, in this case, 0A0010BB.

Afterwards, click an existing plot of the correct size, and use the command

```
cf TestBuildingPlan xxx N
```

Where “xxx” is the FormID, and N the level you want to test. In this case, it would be

```
cf TestBuildingPlan 0A0010BB 1
```

for level 1, for example.

Note: when clicking the plot using the console, keep in mind that the ASAM sensor and the power pole are not part of the plot. You must click the invisible rectangular area.