

Merge Plugins Standalone

by Mator

aka matortheternal

Table of Contents

<u>Introduction.....</u>	1
<u>What is Merge Plugins?.....</u>	1
<u>Why should I use Merge Plugins?.....</u>	1
<u>What are the benefits of merging plugins?.....</u>	1
<u>What are the alternatives?.....</u>	1
<u>Documentation Overview.....</u>	2
<u>Comparison.....</u>	3
<u>Removed Features.....</u>	3
<u>New Features.....</u>	3
<u>Guide to the GUI.....</u>	5
<u>Select Profile Window.....</u>	5
<u>Automatic Game Detection.....</u>	5
<u>New Profile Panel.....</u>	5
<u>Profile Panel.....</u>	6
<u>Profile States.....</u>	6
<u>Profile Popup Menu.....</u>	6
<u>Plugin Selection Window.....</u>	7
<u>Plugins Selection Actions.....</u>	7
<u>Plugin Selection Popup Menu.....</u>	8
<u>Plugin Selection Styles.....</u>	8
<u>Splash Window.....</u>	9
<u>Automatic Updates.....</u>	9
<u>Main Application Window.....</u>	10
<u>Quick Bar.....</u>	10
<u>Plugins Tab.....</u>	12
<u>Merges Tab.....</u>	16
<u>Log Tab.....</u>	19

Status Bar	21
View Dictionary Window	22
Filtering and Sorting	22
Entry Fields	23
Dictionary Information	23
Entry Notes	23
Report Window	24
Edit Merge Window	25
Options Window	26
General Tab	26
Merging Tab	28
Advanced Tab	30
Integrations Tab	32
Changelog Window	34
Progress Window	35
General Reference	36
Merge Statuses	36
Rating Colors	37
Plugin Ratings	38
File-specific Assets	39
Merging and Renumbering Methods	40
Renumbering	40
Record Copying	40
Limitations	41
Doesn't Do Conflict Resolution	41
255 Plugins Limit	41
Conflicting Navmeshes	41
Plugins Must be Loaded	41
Master Dependencies	42

Merged Plugins and Saves	42
Mods that Use DLLs	42
GetFormFromFile isn't Handled	42
Arguments	43
Arguments with Mod Organizer	43
Arguments with Shortcuts	44
Tutorials	45
Merging with a Mod Manager	45
Mod Organizer Advantages	45
Mod Organizer Integration Set Up	46
Mod Organizer Post-Merge Steps	48
Nexus Mod Manager Integration Set Up	49
Nexus Mod Manager Post-Merge Steps	50
Integration Set Up	51
Papyrus	51
BSAOpt	52
Rebuilding Merges	53
When to Rebuild a Merge	53
Effects of Rebuilding a Merge	53
Rebuild Status Detection	53
Updating	54
Updating the Program	54
Updating the Dictionary	54
Automatic Updating	54
Submitting Reports	55
When to Submit Reports	55
How to Submit Reports	55
Debugging	56
Logs	56
Debug Options	56

Reading the Documentation	56
Community	57
Making a Translation	57
Latest english.lang	57
Notepad++ Environment	57
UTF8	57
Syntax	58
Development	59
GitHub, Hipchat, and Forums	59
Reporting Issues	59
Feature Requests	59
Joining the Development Team	60
Service	61
Service Agreement	61
Service Transference	61
Privacy	62
License	63
Plain English	63

Introduction

What is Merge Plugins?

Merge Plugins is a Windows application coded in Delphi (object-oriented Pascal). Merge Plugins automates the process of combining multiple Bethesda Plugin/Master Files (**.esp** and **.esm**) into a single file. The program interfaces with plugins through the xEdit framework, and supports plugin files from Skyrim, Oblivion, Fallout: New Vegas, and Fallout 3.

Merge Plugins doesn't just combine the contents of plugin files, it also handles asset files associated with those plugin files, automatically copying, renaming, and rebuilding assets as necessary to create a mod that functions as the original mods did prior to merging. It also provides a dictionary file built from reports submitted by users as a guide to the tool's effectiveness at merging various plugin files. Overall, Merge Plugins provides a powerful interface by which merges can be created, tracked, and rebuilt.

Why should I use Merge Plugins?

What are the benefits of merging plugins?

- You can (indirectly) pass the 255 plugin file limit
- Increased game stability
- A better organized, easier to manage load order

What are the alternatives?

- **[Merge Plugins xEdit Script](#)**
The precursor to Merge Plugins Standalone. It's somewhat dated and doesn't provide the power, speed, flexibility, or features of Merge Plugins Standalone.
- **[Skyrim Mod Combiner \(SMC\)](#)**
Only works for texture mods, and doesn't combine ESP files. Fulfills a very different use case compared to Merge Plugins Standalone.
- **[Manual merging](#)**
You can use TES5Edit to manually merge mods. This will not handle asset files for you, so you'll have to do all of that by hand after merging. While this can be educational, it will limit you to merging only certain types of mods (e.g. armor and weapon mods) before requiring an unfeasible amount of manual file copying, renaming, and modification.

Documentation Overview

This document is divided into 5 major sections:

1. [Comparison](#)
Covers the differences between Merge Plugins Standalone and the Merge Plugins xEdit Script.
2. [Guide to the GUI](#)
Explains the graphical user interface of Merge Plugins Standalone.
3. [General Reference](#)
Provides some general information on coloring or rating that is reflected in multiple parts of the Merge Plugins GUI.
4. [Tutorials](#)
Teaches you how to use different parts of the application with step-by-step instructions.
5. [Community](#)
Specifies what you can do to contribute to the Merge Plugins community, and your relationship with Merge Plugins as a user.

Comparison

Removed Features

Several features were present in the Merge Plugins script, but are no longer present in Merge Plugins Standalone. These features are listed below:

Feature	Reason
Removing plugins from a merged plugin	This can't be done properly due to issues with the Rule of One and conflicting asset files. Rather than providing a feature that will randomly not work, I've removed this feature. If you want to remove a plugin from a merge, just rebuild the merge.
Merging into existing merges and combining merges	Due to complexities in overrides and conflicting asset files, using an existing merged plugin in a merge can't be done properly in most circumstances. Additionally, allowing it creates issues with managed merges and creates a large increase in complexity. To merge a merged plugin you will have to make a secondary merge profile.
Renumber none	There's no reason for you to use this with the standalone version of Merge Plugins. The standalone program reloads the plugins so this feature has no use case and would only create conflicting FormIDs and broken merged plugins.
Two-pass copying	This used to be necessary a long time ago due to NAVI/NAVMM copying issues, but is no longer necessary. It provides no quantifiable benefits, and has thus been removed.

New Features

Merge Plugins Standalone provides a large number of new features that greatly enhance the experience of merging plugins and managing merges. New features are listed below:

Feature	Description
Script fragment handling	Script fragments are small scripts that are created by the game to assist with Quests, Scenes, and Dialog Topics. These script fragments have FormID specific filenames, and thus need to be renamed and recompiled when the FormID of the record they're associated with changes. Merge Plugins Standalone does this if you set up the Papyrus Compiler and Champollion Decompiler integrations.
INI file handling	Some ESP files have INI files associated with them. These INI files are loaded by the game based on their filename. In Merge Plugins Standalone, INI files are renamed and combined so their functionality is retained after merging.
Merged BSA creation	Merge Plugins Standalone can create a merged BSA file for your merges. This is done through the BSAApt integration.
Toggleable asset handling	In Merge Plugins Standalone you can toggle what asset copying features you want to make use of. This allows users to debug merges more effectively.
Speed increase	Because Merge Plugins Standalone is a compiled program, rather than an interpreted script, it runs a lot faster. You will now be able to build merges faster than ever before.
Merge tracking	Merges created with Merge Plugins Standalone will be tracked from within the program. When a plugin that is in a merge changes (e.g. is updated), the merge it is associated with will flag itself as needing to be rebuilt.

Easy merge rebuilding	You can now rebuild a merge with the click of a button. Combined with merge tracking, this makes for a far more efficient workflow around merging plugins.
Integrated dictionary updates	With the script, you had to update the dictionary through the Merge Plugins Report Client Java application. Now, you will be notified of dictionary updates directly from the program and will be able to update the dictionary without the hassle of having to start another application.
Integrated program updates	Merge Plugins Standalone can be updated easily. You won't need to re-download it from the Nexus to update it, you will be informed of updates from within the program itself, and can update the program with a click of a button.
Integrated reporting	With the script, reports had to be generated as text files that were then read by a separate Java application which sent them to a server to be combined into a dictionary file. With Merge Plugins Standalone this functionality has been integrated into the main program, so you don't need to start a separate program to submit reports anymore.
Plugin error tracking	Once you check a plugin for errors in Merge Plugins Standalone, its errors will be saved, and then reloaded the next time you use the program. These saved errors are linked to the CRC32 hash of the plugin, so when the plugin is updated you'll have to check it for errors again (it may have more/fewer errors).
Multiple languages	With an easily extensible language system, the program can now be easily translated. Several translations have already been made thanks to some volunteer translators.
Statistics tracking	Merge Plugins Standalone will locally track statistics relating to merging. Now you can keep track of how many merges you've build, how many plugins you've checked for errors, how many times you've started the program, etc.

Guide to the GUI

Select Profile Window



Illustration 1: The select profile window.

The Select Profile Window displays when the program first starts. From here you can create and manage your merge profiles. To start the application, select a valid profile and click OK. You can bypass the Select Profile window by using the `-profile` target line parameter.

Automatic Game Detection

The profile window will automatically detect the games you have installed from the registry. Assuming you performed a standard or steam game installation, the program should detect any supported games you have installed and create profiles for them when you first run it. Automatic detection will only occur if you have no profiles saved in the *profiles* folder.

New Profile Panel

This is a special panel which will always be below your other profiles. You can click it at any time to create a new profile.

Profile Panel

Component	Description
Game icon	An icon corresponding to the currently selected game mode. If the profile is valid, you can click the game icon or anywhere else on the panel to select the profile.
Profile name text box	Here you can enter a profile name, which will be used to identify the profile.
Game mode drop-down	A drop down where you can specify the game mode you want the profile to be for. The options are: Skyrim, Oblivion, Fallout New Vegas, and Fallout 3.
Game path text box	The path to the folder where the game you have selected is installed. You can use the browse button to browse for this folder.
Browse button	Click this button to browse for your game path.
Delete button	Click this button to delete the profile. You will be asked to confirm this action to make sure you didn't accidentally click delete.

Profile States

State	Color	Description
Invalid	Light red	If the profile name or game path is invalid, the background color of the profile panel will turn light red. You can't select an invalid profile.
Valid	Light green	If the profile name and game path are valid, the background color of the profile panel will turn light green. You can select valid profiles.
Selected	Light blue	If you click a valid profile, its background color will turn light blue. This indicates the profile is selected. Once you've selected a profile, you can click OK to start Merge Plugins with that profile.

Profile Popup Menu

Right-click in the scrollbar to access the profile popup menu.

Name	Action	Enabled
New profile	Creates a new merge profile.	Always.
Delete profile	Deletes the profile you right clicked on.	If you right clicked on a merge profile.

Plugin Selection Window

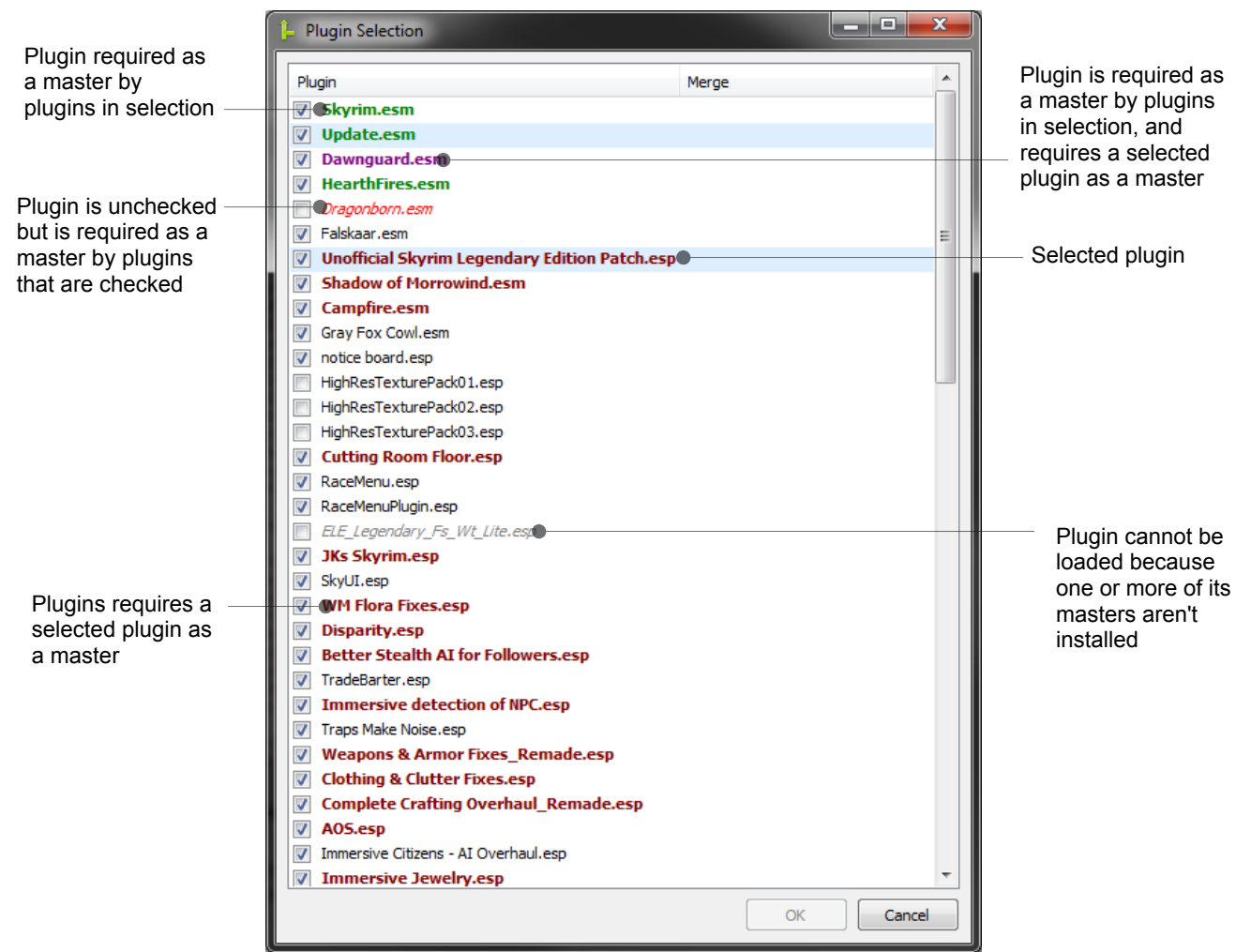


Illustration 2: The plugin selection window.

The plugin selection window appears after you've selected a profile, and allows you to specify what plugins you want to load.

Plugins Selection Actions

Action	Description	Availability
OK	Starts the application and loads the checked plugins.	If no unchecked plugin is required by checked plugins as a master, as indicated by <i>red italic</i> font.
Enter	Equivalent to clicking OK.	Same as OK.
Space	Toggles the check box of the selected plugins.	If one or more plugins are selected.
Cancel	Terminates the application.	Always.
Close	Terminates the application.	Always.

Plugin Selection Popup Menu

Right click in the list view to access the “Plugin Selection Popup Menu”.

Name	Action
Check all	Checks all plugins in the list except ones that cannot be loaded.
Uncheck all	Unchecks all plugins in the list.
Toggle all	Toggles the selection of all plugins in the list except ones that cannot be loaded.

Plugin Selection Styles

Style	Meaning
Light blue background	The plugin is selected.
Bold green font	The plugin is required as a master by a selected plugin.
Bold maroon font	The plugin requires a selected plugin as a master.
Bold magenta font	The plugin is required as a master by a selected plugin, and requires a selected plugin as a master.
<i>Gray italic font</i>	The plugin cannot be loaded because one or more of its masters aren't installed. That is, its masters were not found in the data directory.
<i>Red italic font</i>	The plugin is unchecked but is required as a master by one or plugins that are checked. You will need to check this plugin or uncheck the plugins that depend on it to continue.

Splash Window



Illustration 3: The splash window.

The splash will be displayed after a merge profile has been chosen. The splash shows the progress of automatic updates as well as the loading of plugins, merges, and plugin information. It also displays the version of the application. The splash should close automatically and open the main application window after 10-30 seconds, depending on the size of your load order and your system specs.

Automatic Updates

If you have enabled automatic updates in your settings, the application will attempt to connect to the server and get any dictionary or program updates that are available. If there is a program update, the application will restart, taking you back to the profile screen.

Main Application Window

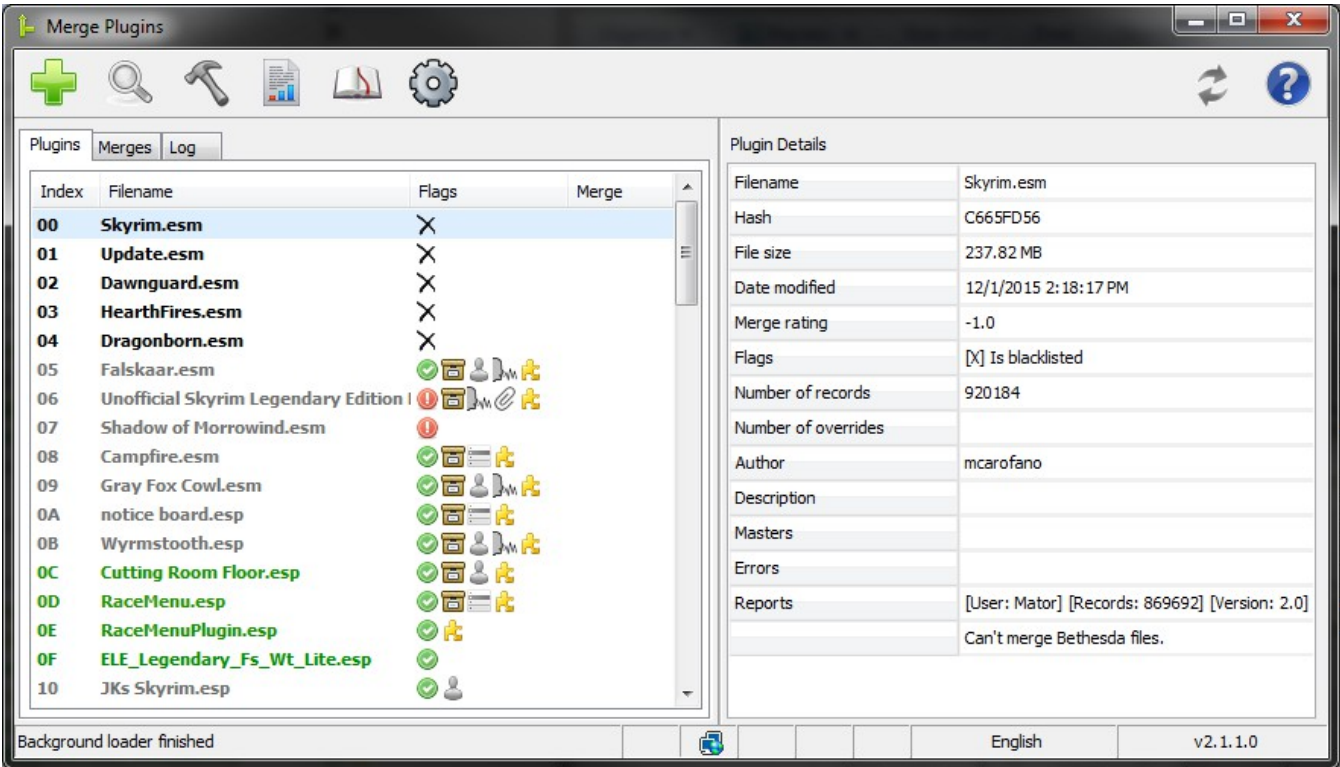


Illustration 4: The main application window (at small window size so you can see it).

The Main Application Window is the location from which you can interface with all of the functionality offered by Merge Plugins, and is the window that is shown after you've selected a profile and the plugins you want to load.

Quick Bar



Illustration 5: The quick bar.

The Quick Bar provides multiple buttons to quickly access various functionalities in the application.

Create New Merge

You can click the Create New Merge button at any time to create a new merge. The button will

automatically bring up the [Edit Merge Window](#) for the new merge so you can specify the merge's name, filename, and options.

Find Errors

Clicking the Find Errors button will check for errors in all loaded, non-blacklisted plugins that have not yet been checked for errors.

Build Merges

Clicking the Build Merges button will build any merges that are ready to be built or rebuilt. The button will be disabled if the background loader is not completed or if there are no merges to build. See [Merge Statuses](#) if you want to force a rebuild of your merges.

Report on Merges

Clicking the Report on Merges button will bring up the [Report Window](#). From there, the user can create and submit reports on all plugins in merges that have been built successfully, excluding plugins the user has already submitted reports for.

View Dictionary

Clicking the View Dictionary button will bring up the [View Dictionary Window](#) with no filters set.

Options

Clicking the Options button will bring up the [Options Window](#).

Update

Clicking the Update button will update the dictionary if any update is available. If an application update is available the [Changelog Window](#) will be shown, where you can see the changes between your current version and the most up-to-date version and choose to download and install the new version. Hovering over the “Update” button will display a hint specifying what updates are available.

Help

Clicking the Help button will change the mouse cursor to a pointer with a question mark. You can then click on any component on the [Main Application Window](#) to access the help files on it. You can also press F1 from any window that's a part of the application to access the same functionality.

Plugins Tab

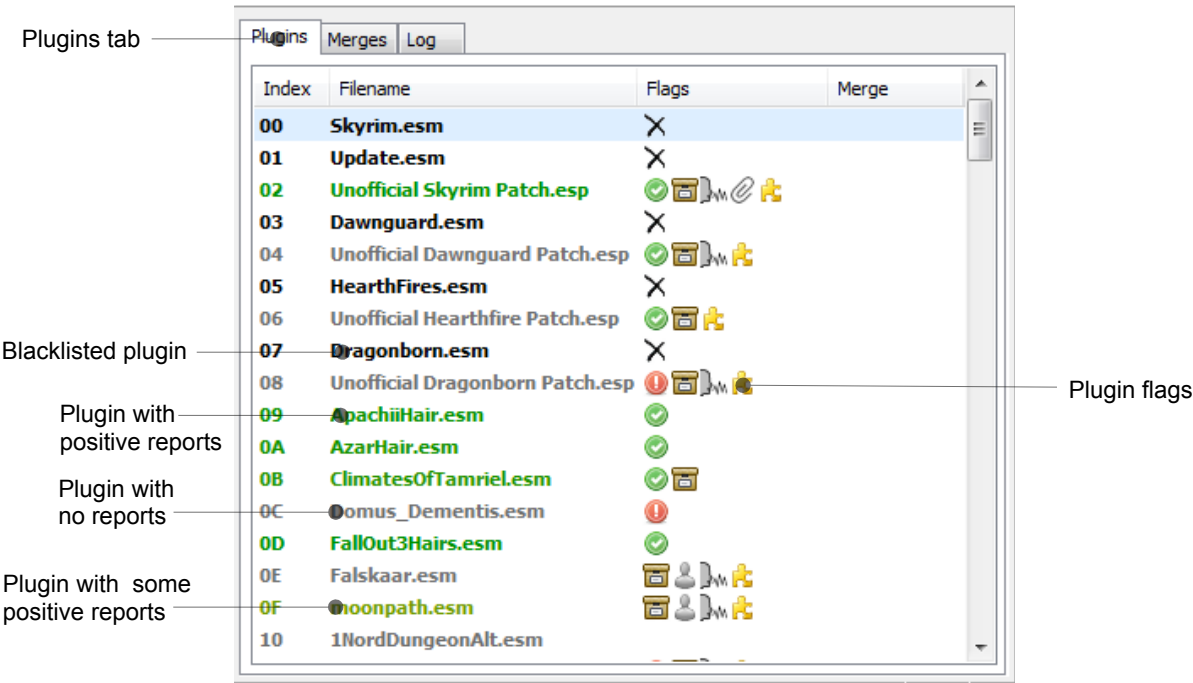














Illustration 6: The Plugins Tab and Plugins List View.

The Plugins Tab displays an ordered list of loaded plugins. Unless simple plugins list is enabled, the plugins will be colored based on average rating reported by users. See [Rating Colors](#) for a table of colors and their meaning.

Plugin Fields

Name	Description
Index	The hexadecimal load-order index of the plugin.
Filename	The filename of the plugin.
Flags	The flags associated with the plugin. If you have simple plugins list enabled, flags will be represented by a single character, else flags will be displayed as icons.
Merge	The merge that the plugin is associated with.

Plugin Flags

<i>Icon</i>	<i>Character</i>	<i>Description</i>
	X	Is blacklisted
	E	Has errors
	R	Errors ignored
	N	Has no errors
	A	Has a BSA file
	G	Has FaceGenData
	V	Has voice data
	T	Has MCM translations
	I	Has an INI file
	F	Has script fragments
	D	Disallow merging
	S	Scripts reference self

Plugins Popup Menu

Right-clicking on the plugins list will bring up a popup menu with various options. Below is a listing of these options, what they do, and under what conditions they're enabled.

Name	Action	Enabled
Add to merge → <New Merge>	Creates a new merge and adds the selected plugins to it.	If no selected plugin is blacklisted, marked as do not merge, or in a merge.
Add to merge → <Merge>	Adds the selected plugins to the specified merge.	If no selected plugin is blacklisted, marked as do not merge, or in a merge.
Remove from merge	Removes the selected plugins from any merge they're in.	If all selected plugins are in a merge.
Disallow merging	Marks selected plugins with the Do Not Merge flag.	If all selected plugins are not blacklisted or in a merge.
Allow merging	Unsets the Do Not Merge flag on selected plugins.	If all selected plugins have the do not merge flag set.
Report on plugin	Brings up the Report Window for the selected plugins.	If no selected plugin is blacklisted.
Errors → Check for errors	Starts an error-checking task for the selected plugins.	If the background loader is done, no selected plugin is blacklisted, and no selected plugin has been checked for errors.
Errors → Fix errors	Starts an error fixing task for the selected plugins.	If the background loader is done, no selected plugin is blacklisted, and all selected plugins have errors.
Errors → Ignore errors	Marks selected plugins with the Errors Ignored flag if they have errors.	If no selected plugin is blacklisted, no selected plugin hasn't been checked for errors, and all selected plugins have errors.
Errors → Unignore errors	Unsets the errors ignored flag on selected plugins.	If all selected plugins have the ignore errors flags set.
Open plugin location	Opens the location of each selected plugin file in explorer.	Always.

Plugin Details

When you select a plugin, its details will be populated in the Details Pane. Below is a listing of these details, with descriptions.

Name	Description
Filename	The filename of the plugin.
Hash	The CRC32 hash of the plugin file. This is used to track when the plugin is updated.
File size	The file size of the plugin.
Date modified	The date the plugin was last modified.
Merge rating	The plugin's average rating as reported by users, between 0.0 and 4.0. See Merge Ratings for more information.
Flags	A list of the flag characters associated with the flags on the plugin. See Plugin Flags for more information.
Number of records	The number of records in the plugin.
Number of overrides	The number of override records in the plugin.
Author	The author of the plugin.
Description	The description of the plugin.
Masters	The master files of the plugin.
Errors	The errors in the plugin. Will be "None" if no errors were found in the plugin.
Reports	A list of the user reports submitted on the plugin.

Merges Tab

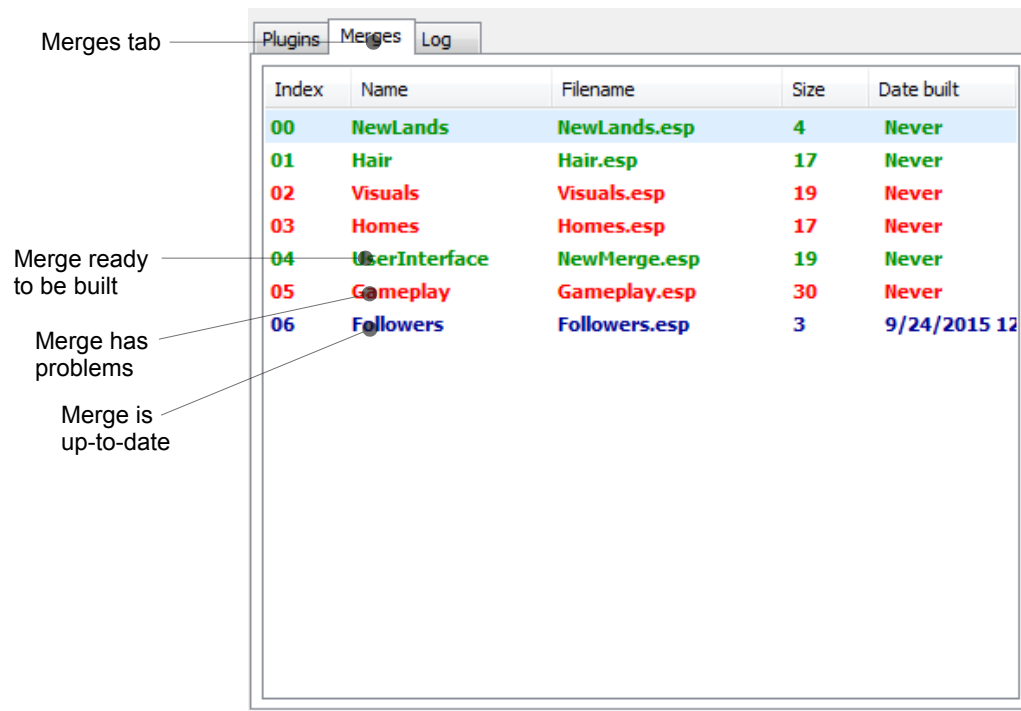


Illustration 7: The merges tab.

The Merges Tab displays a list of merges managed with the program. Merges are colored based on their status. Also see [Merge Statueses](#).

Merge Fields

Name	Description
Index	The build order of the merge.
Name	The name of the merge.
Filename	The filename of the plugin file created for the merge.
Size	The size of the merge in number of plugins.
Date built	The date when the merge was last built.

Merges Popup Menu

Right-clicking on the merges list will bring up a popup menu with various options. Below is a listing of these options, what they do, and under what conditions they're enabled.

Name	Action	Enabled
Create new merge	Creates a new merge and opens the Edit Merge Window for it.	Always.
Edit merge	Opens the Edit Merge Window for each selected merge.	If a merge is selected.
Delete merge	Deletes the selected merges.	If a merge is selected.
Plugins → Remove plugins	Removes plugins that have errors or that aren't loaded from the merge.	If the merge has unloaded plugins or plugins with errors.
Plugins → Check plugins for errors	Creates an error-checking task for the plugins in selected merges that haven't been checked for errors.	If the background loader is done, a merge is selected, and one or more plugins in the selected merges need to be checked for errors.
Plugins → Fix errors in plugins	Creates an error-fixing task for the plugins in selected merges that have errors.	If one or more selected merges has a plugin with errors in it.
Plugins → Report on plugins	Opens the Report Window with all plugins in selected merges.	If one or more selected merges is up to date.
Build merge	Builds the selected merges.	If a selected merge is ready to be built/rebuilt.
Open in explorer	Opens the folder associated with the merge in explorer.	If a merge is selected.
Toggle rebuild status	Switches merges from the "Up to date" status to "Ready to be rebuilt [forced]" and from the "Ready to be rebuilt" status to "Up to date [forced]".	If one or more selected merges is up to date or ready to be rebuilt.

Merge Details

When you select a merge, its details will be populated in the Details Panel. Below is a listing of these details, with descriptions.

Name	Description
Status	The status of the merge. See Merge Statuses.
Merge name	The name of the merge.
Filename	The filename of the merge.
Plugin count	The number of plugins in the merge.
Date built	The date the merge was last built.
Plugins	A list of the plugins in the merge.
Merge method	The merge method that will be used when building the merge.
Renumbering	The renumbering method that will be used when building the merge.
Files	A list of the files associated with the merge. If there are over 250 files the list will not be displayed.
Fails	A list of the failures that occurred when last building the merge. If there are over 250 failures the list will not be displayed.

Log Tab

The Log Tab displays enabled log messages that have occurred during program execution. You can change the colors of log messages from the [Options Window](#). You can filter what log messages are displayed from the Log Popup Menu.

Note: Log messages are displayed as a single string. The log message template string can be specified from the [Options Window](#).

Log Fields

Name	Description
Time	The time when the log message was created.
AppTime	The application time when the log message was created. (00:00:00 is when the application is started).
Group	The group the log message is associated with. See Log Groups.
Label	The label the log message is associated with. See Log Labels.
Text	The text of the log message.

Log Popup Menu

Right-clicking on the log will bring up a popup menu with various options. Below is a listing of these options, what they do, and under what conditions they're enabled.

Name	Action	Enabled
Filter group → <Toggle> <Group>	Toggles visibility of log messages from the specified group.	Always
Filter label → <Toggle> <Group>, <Label>	Toggles visibility of log messages from the specified group-label combination.	Always
Copy to clipboard	Copies selected log messages to the clipboard.	If one or more log messages are selected.
<Toggle> auto scroll	Toggles automatic scrolling of the log.	Always
Save and clear	Saves the log to a file and then clears it.	Always

Application Details

Application details appear in the Details Pane when you switch to the Log Tab. Below is a listing of these details, with descriptions.

Name	Description
Application	The name of the running application, "Merge Plugins".
Author	The author of the running application, "matortheeternal".
Version	The version of the application you're using.
Date built	The date when the version of the application you're using was built.
Game mode	The game mode you're currently using the application in.
Language	The current language mode you're using the application in.
Times run	The number of times you've run the application.
Merges built	The number of merges you've built.
Plugins checked for errors	The number of plugins you've checked for errors.
Plugins merged	The number of unique plugins you've merged.
Reports submitted	The number of reports you've submitted.
GitHub Repo	The GitHub repository for the Merge Plugins project. You can go here to submit issues, track development, and download dev builds.
Nexus Mods Page	The Nexus Mods page for Merge Plugins.
API Credits	Credits for APIs used in Merge Plugins.
xEdit Version	The version of xEdit that Merge Plugins is using.
xEdit Credits	Credits for the creators of xEdit.
Testers	List of beta testers who contributed to the development of Merge Plugins.
Translators	List of translators who have translated Merge Plugins into different languages.

Status Bar

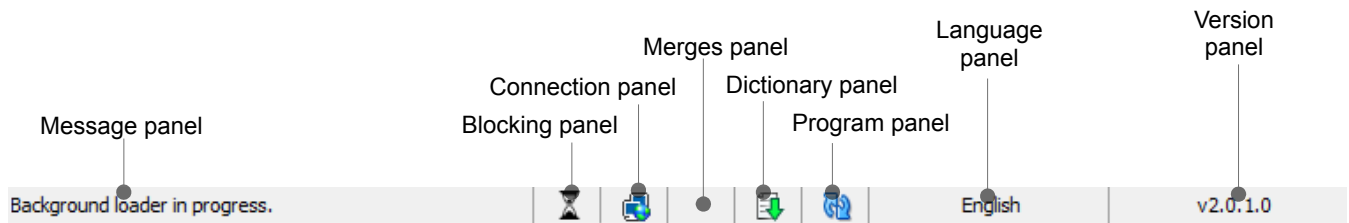







Illustration 8: The status bar.

The Status Bar provides quick reference to the status of various aspects of the Merge Plugins program.

Status Panels

Name	Icons	Description
Message panel	N/A	Displays a message regarding overall program status.
Blocking panel		Displays its icon if a blocking background operation is in progress (e.g. background loader, checking for errors, or merging plugins).
Connection panel		Displays an icon indicating whether or not the program is connected to the Merge Plugins server.
Merges panel		Displays its icon if there are merges that are ready to be rebuilt.
Dictionary panel		Displays its icon if a dictionary update is available.
Program panel		Displays its icon if a program update is available.
Language panel	N/A	Displays the current language mode.
Version panel	N/A	Displays the program's version.

View Dictionary Window

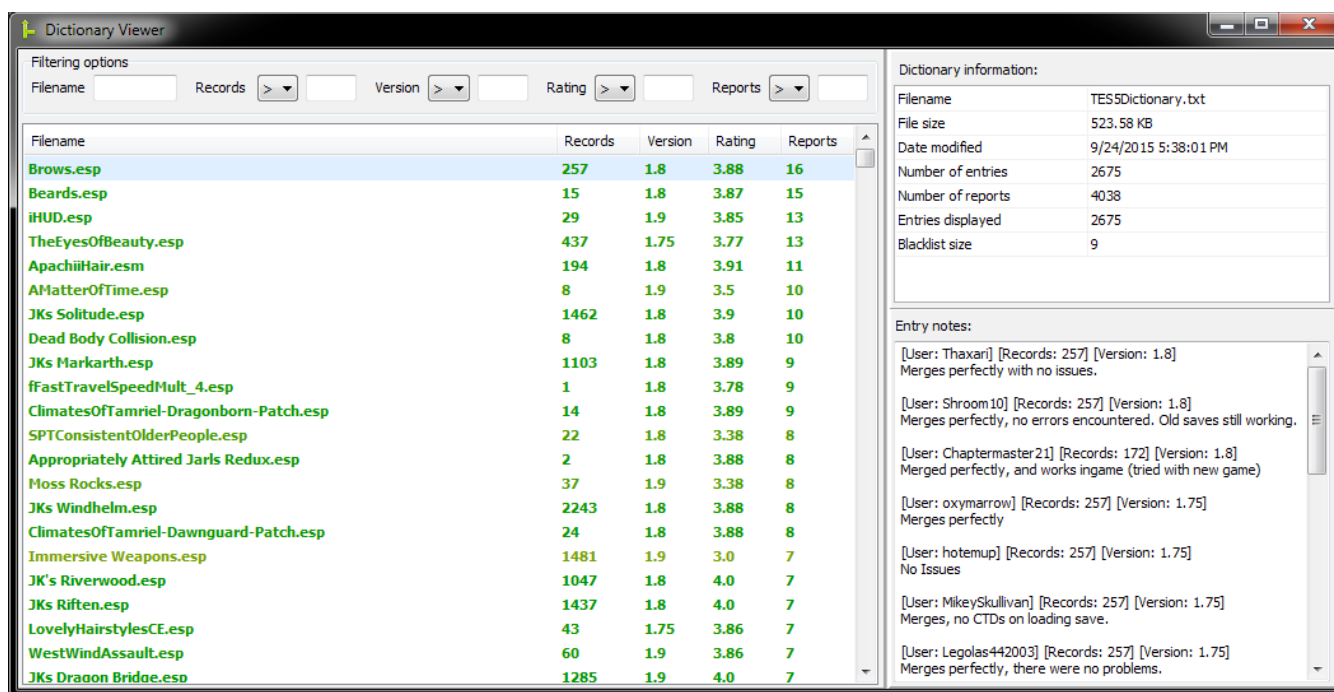


Illustration 9: The view dictionary window.

The View Dictionary Window displays a list of all dictionary entries in the dictionary file for the current game mode. The window supports full filtering and sorting functionality.

Filtering and Sorting

You can sort by any single column by clicking on its header. You can reverse the order of sorting from ascending to descending and vice versa by clicking on the column header a second time.

Filtering options are provided in a group above the entries list. After entering your filters, press Enter in any of the filtering text fields to filter the entries list.

Field	Modes	Filtering functionality
Filename	Contains	Selects entries which have filenames that contain the entered text.
Records	Greater than, less than, equal to	Selects entries which have a number of records <mode> the entered integer number.
Version	Greater than, less than, equal to	Selects entries which have a version <mode> the entered floating point number.
Rating	Greater than, less than, equal to	Selects entries which have a rating <mode> the entered floating point number.
Reports	Greater than, less than, equal to	Selects entries which have a number of reports <mode> the entered integer number.

Entry Fields

Name	Description
Filename	The filename of the plugin to which the dictionary entry corresponds.
Records	The largest reported number of records in the plugin the dictionary entry corresponds to.
Version	The latest version of Merge Plugins that has had a report submitted on the plugin to which the dictionary entry corresponds.
Rating	The average rating of all reports submitted on the plugin to which the dictionary entry corresponds.
Reports	The number of reports associated with the dictionary entry.

Dictionary Information

The Dictionary Information panel lists various stats relating to the dictionary file being viewed.

Name	Description
Filename	The filename of the dictionary file being viewed.
File size	The file size of the dictionary file being viewed.
Date modified	The date when the dictionary was last modified (updated).
Number of entries	The number of entries in the dictionary file.
Number of reports	The number of reports in the dictionary file.
Entries displayed	The number of entries currently being displayed due to filtering.
Blacklist size	The number of plugins in the dictionary file that have been marked as blacklisted.

Entry Notes

The Entry Notes panel displays the user notes associated with the selected entry. User notes have a header associated with them which displays information relating to the particular report the notes are associated with. Depending on whether or not the report was submitted before before Merge Plugins Standalone's public release, the header will differ.

Reports submitted with Merge Plugins Standalone:

[User: {{user}}] [Hash: {{hash}}] [Version: {{version}}]

Reports submitted with the Merge Plugins Report Client:

[User: {{user}}] [Records: {{records}}] [Version: {{version}}]

Report Window

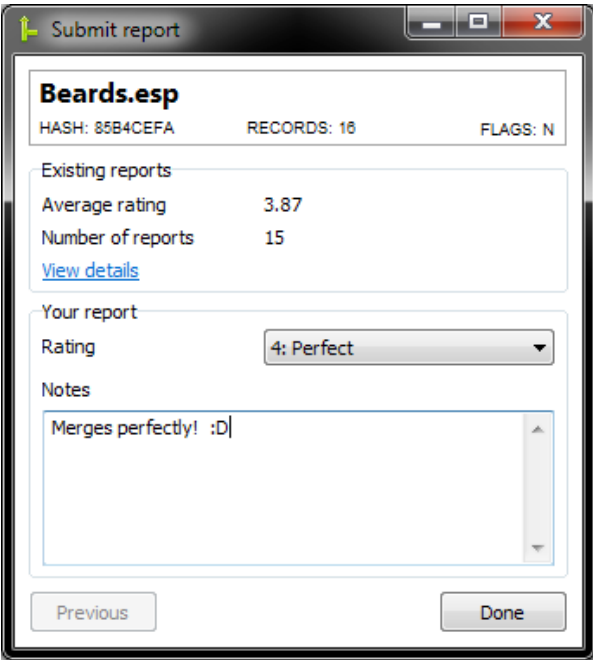


Illustration 10: The report window.

The Report Window is used to create reports on the merge-ability of plugins to be submitted to the Merge Plugins server, where it is later built into the dictionary file.

Element	Description
Filename	The filename of the plugin you're reporting on.
Hash	The CRC32 hash of the plugin you're reporting on.
Records	The number of records in the plugin you're reporting on.
Flags	The flags associated with the plugin you're reporting on, as a string.
Average rating	The average rating from existing user reports on the plugin you're reporting on.
Number of reports	The number of reports that have already been submitted on the plugin you're reporting on.
View details	Opens the View Dictionary Window with a filename filter for the plugin you're reporting on.
Rating	This is a drop-down where you can specify the rating you want to submit for the plugin.
Notes	Notes on the merge-ability of the plugin for your report. You are required to enter at least 16 characters in the notes field to create a report.
Next button	Proceed to reporting on the next plugin in the reporting task you created. Changes to "Done" when you reach the last plugin.
Previous button	Go back to the report on the previous plugin in the reporting task you created.

Edit Merge Window

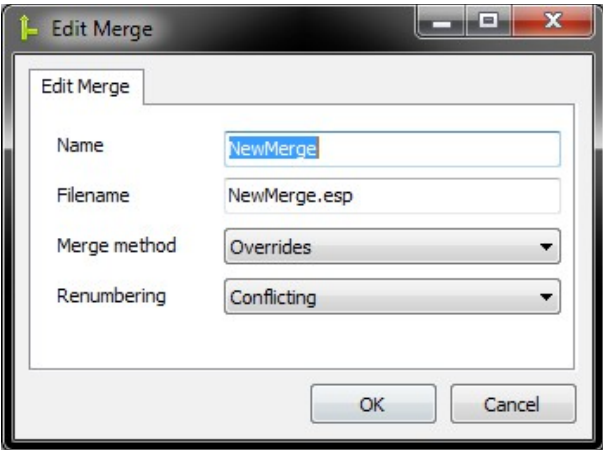


Illustration 11: The edit merge window.

The Edit Merge Window is used to edit the details of a merge. Also see [Merging and Renumbering Methods](#).

Element	Description
Name	The name of the merge you're editing. This is used for naming the folder the merge will be stored in, and for identifying the merge in the application. The text will turn red and will display a hint if a folder matching the specified name already exists in the merge destination directory.
Filename	The filename of the ESP file that will be produced by the merge. Also used for determining the path of various asset files. The text will turn red and the OK button will be disabled if an ESP of the specified filename is currently loaded in Merge Plugins.
Merge method	The method that will be used when merging the plugins in the merge. Two methods are available: Overrides and New Records.
Renumbering	The method that will be used when renumbering FormIDs for the merge. Two methods are available: Conflicting and All.
OK button	Click to save your changes to the merge. Will be disabled if a conflicting filename is entered.
Cancel button	Click to cancel your changes to the merge.

Options Window

The Options Window is where you can set options that dictate how the program runs and how merges are executed. You won't need to go here much after initial set up.

General Tab

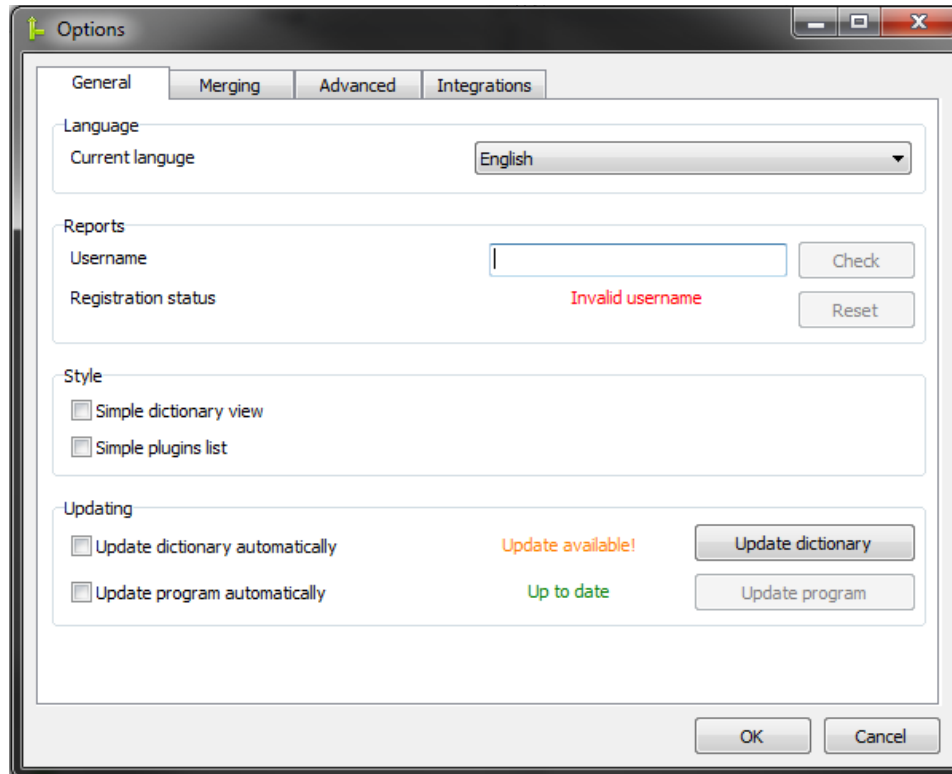


Illustration 12: The General Tab of the Options Window.

Language

Here you can set the language to be used by the program. For information on creating translations, see Making a Translation.

Reports

This group has controls that pertain to the submission of user reports to the backend. If you want to submit reports on plugins you've merged to the backend, you need to register a username.

1. Enter a username in the text box. The username must be between 4 and 32 characters long.
2. Click the “Check” button to see if the username is available.
3. If the username is available, the “Check” button will change to say “Register”. Click it to register the username.

Technical

Whenever you start the program, the program generates a random 32-character long Base64 key. When you register, your key at the time of registering is stored in your settings.ini file and in the registry. This key serves to authenticate you to the server. If you somehow lose this key, you can attempt to reset it by clicking the Reset button. This will only work if your IP address hasn't changed since the key was lost.

If you're planning on reformatting your drive and want to maintain access to your old username, you should save your settings.ini file somewhere where you can access it later (e.g. email, upload to Dropbox, external storage medium, etc.).

Style

Element	Description
Simple dictionary view	Affects the View Dictionary Window. Disables the coloring of dictionary entries.
Simple plugins list	Affects the Plugins Tab. Disables the coloring of plugins, and uses flag characters instead of flag icons.

Updating

Element	Description
Update dictionary automatically	If checked, Merge Plugins will check for a dictionary update once initial plugin loading is complete. If an update is available, it will be downloaded automatically.
Update program automatically	If checked, Merge Plugins will check for a program update as soon as the program is started. If an update is found, it will be downloaded and installed automatically, and the program will restart itself.
Dictionary status label	Indicates whether or not a dictionary update is available.
Program status label	Indicates whether or not a program update is available.
Update dictionary	If a dictionary update is available, you can click this button to download and install it.
Update program	If a program update is available, you can click this button to download and install it. Note: The program will close and restart itself when updated.

Merging Tab

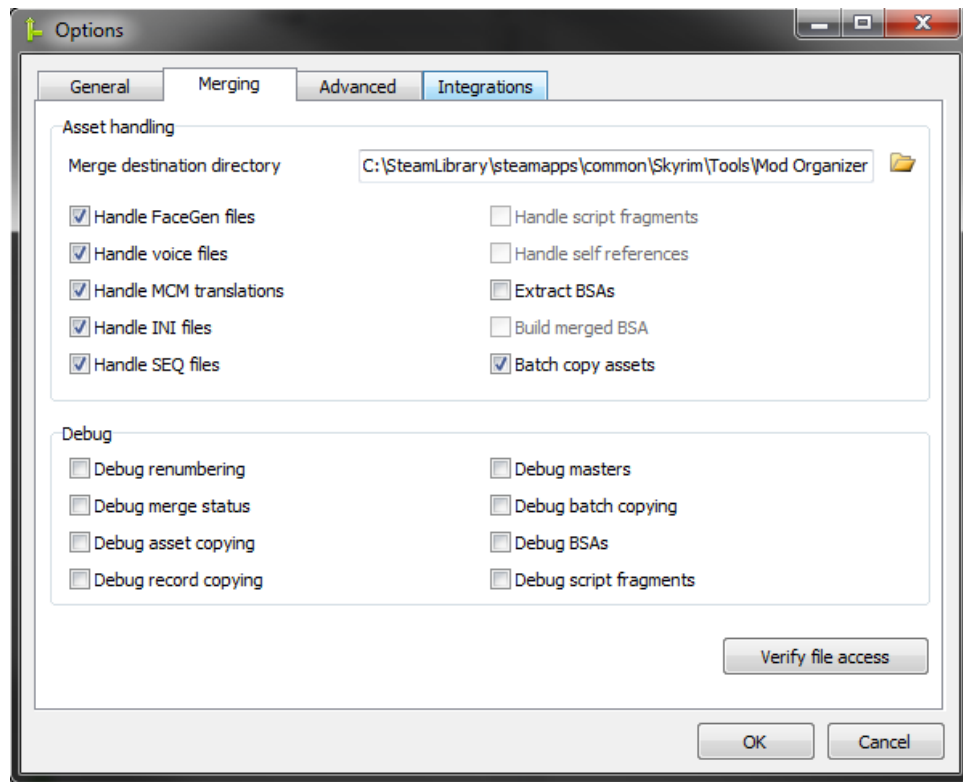


Illustration 13: The Merging Tab of the Options Window.

Verify File Access

You can click the “Verify file access” button to verify that Merge Plugins has the necessary filesystem permissions to function properly. I recommend you verify file access when you first install Merge Plugins and again whenever you change the directories used by Merge Plugins.

Asset handling

Note: The copy general assets option is available on the [Integrations Tab](#) as a part of the Mod Manager integration.

<i>Element</i>	<i>Description</i>
Merge destination directory	The directory that merges will be created in. Merges will be created in separate folders corresponding to their name. You can click the browse button to specify a folder to use, or enter the path manually.
Handle FaceGenData	If checked, file-specific FaceGenData will be handled when merging. This means that facegeom and facetint data files that were associated with the plugins you were merging will be copied to a new directory for the merged plugin and renamed if the FormID of the NPC record they are associated with was changed. If not checked, the gray face bug will occur for new NPCs in the merged plugin.
Handle voice assets	If checked, file-specific voice assets will be handled when merging. This means that sound\voice data files that were associated with the plugins you were merging will be copied to a new directory for the merged plugin and renamed if the FormID of the record they are associated with was changed. If not checked, NPCs will lose any custom voicing.
Handle MCM translations	If checked, MCM translation files for the plugins being merged will be combined into a single MCM translation for the merged plugin. If not checked, some MCMs may become garbled to variable names instead of proper labels.
Handle INI files	If checked, INI files that were associated with the plugins being merged will be combined into a single INI file for the merged plugin. If not checked, INI files will be ignored during merging.
Handle script fragments	If checked, Script Fragments that were associated with the plugins being merged that were associated with a record that was renumbered will be recompiled. This option requires setting the papyrus integration up correctly, and will be disabled until you have done so. If not checked, Quests, Scenes, and Dialog Topics that relied on new script fragments may not work correctly.
Extract BSAs	If checked, BSAs associated with the plugins being merged will be extracted in the merge's data folder. This does not require setting up the BSAOpt integration. This is only necessary if you are not using Mod Organizer, which supports the loading of arbitrarily named BSAs.
Build merged BSA	If checked, a merged BSA will be created from the contents of all the BSAs associated with plugins that were merged. This option requires setting up the BSAOpt integration, and will be disabled until you have done so. This is a nice alternative to Extract BSAs if you're using NMM. Note: BSAs have a maximum file size of 2gb.
Batch copy assets	If checked, all file-specific and general asset copying operations will be performed through a batch (.bat) script. This is a recommended option, as it is more efficient than copying files through Delphi methods.

Advanced Tab

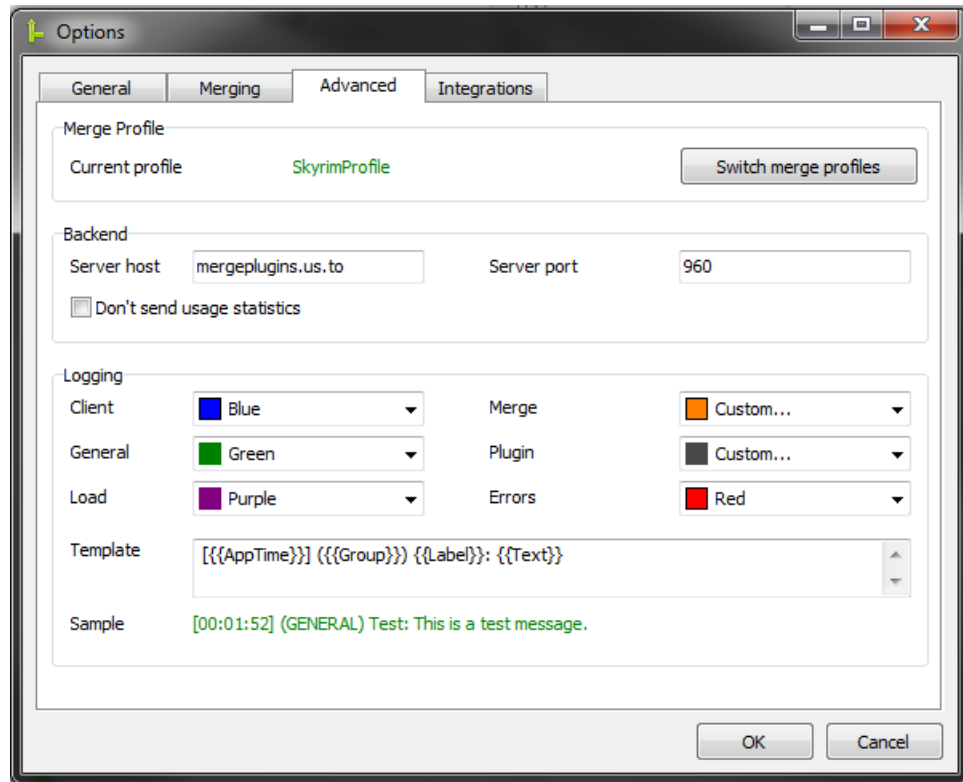


Illustration 14: The Advanced Tab of the Options Window.

Merge Profile

Provides the name of your currently active profile. You can click the “Switch merge profiles” button to restart the application and go to the [Select Profile Window](#).

Backend

Element	Description
Server host	The hostname that will be connected to for backend services.
Server port	The server port that will be connected to for backend services.
Don't send usage statistics (functional as of v2.0.2.3)	If checked, user statistics such as how many merges you've made and how many plugins you've checked for errors will not be sent to the backend. Other server-side statistics will still be tracked (e.g. bandwidth usage, date last seen, etc.). For a full listing of client-side and server-side statistics, please see the Privacy section .

Logging

<i>Element</i>	<i>Description</i>
Client	Specifies the color of Client messages in the log.
General	Specifies the color of General messages in the log.
Load	Specifies the color of Load messages in the log.
Merge	Specifies the color of Merge messages in the log.
Plugin	Specifies the color of Plugin messages in the log.
Errors	Specifies the color of Errors messages in the log.
Template	Specifies the template to be used by messages in the log. The accessible variables are: {{Time}}, {{AppTime}}, {{Group}}, {{Label}}, and {{Text}}.
Sample	A sample log message using the template you've specified.

Integrations Tab

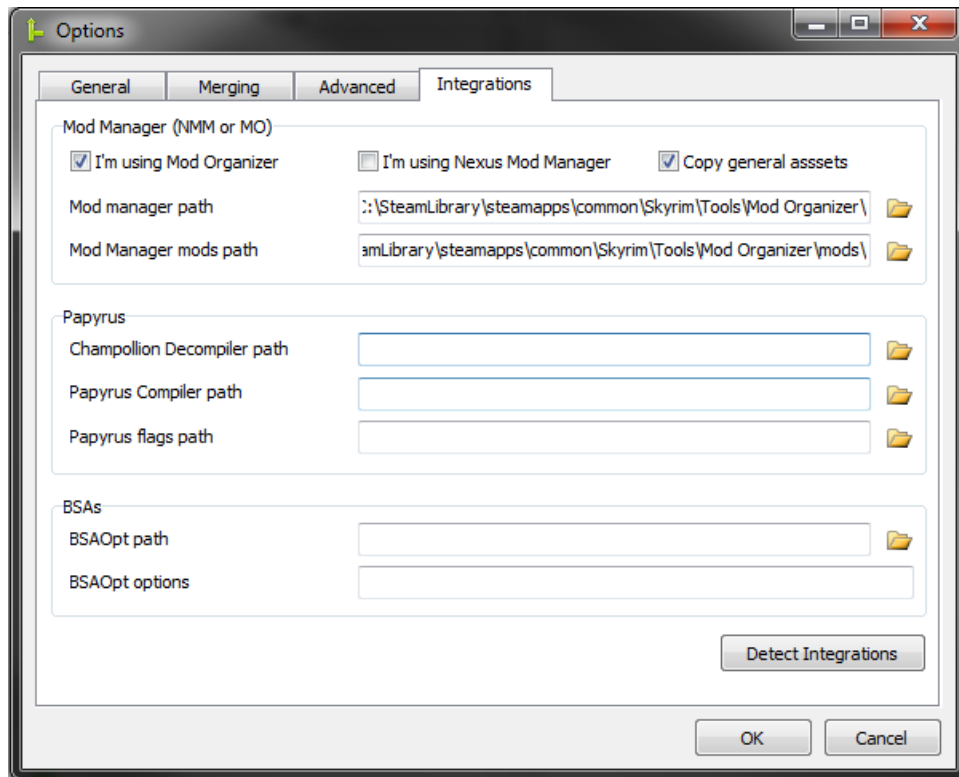


Illustration 15: The Integrations Tab of the Options Window.

Detect Integrations

You can have Merge Plugins search for installed integrations by clicking the Detect Integrations button. Make sure you verify the integration paths found by Merge Plugins, especially in regards to the Mod Manager integration. Sometimes Merge Plugins detects the wrong installation of Mod Organizer.

Mod Manager

<i>Element</i>	<i>Description</i>
I'm using Mod Organizer	Check this if you're using Mod Organizer.
I'm using Nexus Mod Manager	Check this if you're using Nexus Mod Manager 0.60a or later. 0.60a and later versions of NMM use a virtual directory structure similar to Mod Organizer.
Mod manager path	Set this path to the folder holder the Mod Manager's executable file. E.g. ModOrganizer.exe or NexusModManager.exe.
Mod manager mods path	Set this to the path to the folder that holds the mods associated with the mod manager. For Mod Organizer the default location is the mods folder alongside ModOrganizer.exe.
Copy general assets	If checked, general assets (ones that aren't file-specific) will be copied. This is only possible with the Mod Manager integration.

Papyrus

<i>Element</i>	<i>Description</i>
Champollion Decompiler path	The path to the Champollion Decompiler executable. This must be set to make use of the handle script fragments option.
Papyrus Compiler path	The path to the Papyrus Compiler executable. This must be set to make use of the handle script fragments option.
Papyrus flags path	The path to the papyrus flags file (TESV_Papyrus_Flags.flg). This must be set to make use of the handle script fragments option.

BSAs

<i>Element</i>	<i>Description</i>
BSAOpt path	The path to the BSAOpt executable. This must be set to make use of the build merged BSA option.
BSAOpt options	The target line options you want to use with BSAOpt.

Changelog Window

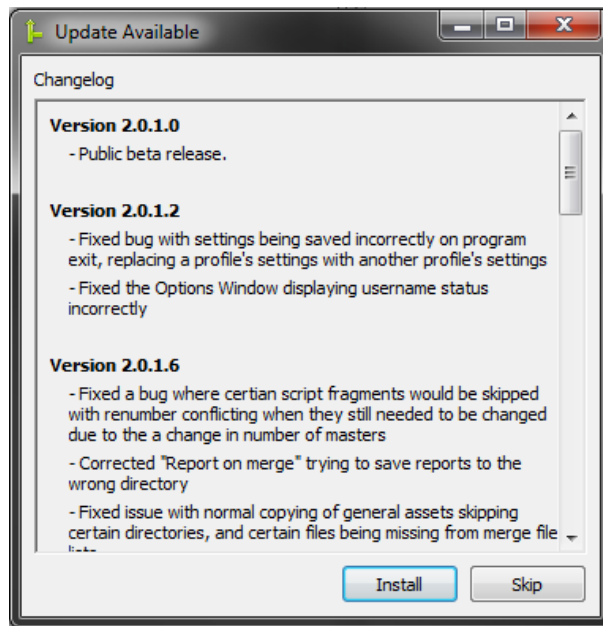


Illustration 16: The changelog window.

The Changelog Window shows you the changes between the version you're running and the newest program update before installing it. Once you've reviewed the changes you can click "Install" to have the program download and install the update or "Skip" to not do so.

Progress Window

The Progress Window is a window that comes up when Merge Plugins is performing a large task, such as merging plugins or checking plugins for errors. The window displays the progress of the task being performed through an embedded log and a progress bar.

<i>Element</i>	<i>Description</i>
Progress label	Displays text corresponding to the operation currently being performed.
Progress bar	Displays the progress of the task currently being performed. Allows you to track how close the operation is to completion.
Log	Displays log messages corresponding to the task currently being performed, including errors. The log is only visible if you click the show details button.
Show/Hide details	Clicking this button will toggle the log's visibility.
Cancel/Close	Clicking this button while the task is being performed will cancel it. Once the task is completed, it can be clicked to close the progress window.

General Reference

Merge Statuses

<i>Color</i>	<i>Title</i>	<i>Description</i>
Grey	Unknown	The status for this merge has not yet been determined.
Red	Plugins not loaded	Plugins from the merge are not loaded in the application.
Red	No plugins to merge	The merge has no plugins in it and can't be built.
Red	Directories invalid	Your directories are invalid. Set them in the Options Window.
Red	Errors in plugins	Plugins in the merge have errors in them.
Red	Merge failed	Building the merge failed.
Orange	Check for errors required	You need to check all plugins in the merge for errors before the merge can be built.
Blue	Up to date	The merge has been built and doesn't need to be rebuilt.
Blue	Up to date [Forced]	The merge has been marked as up to date by the user.
Green	Ready to be built	The merge has never been built before, and is ready.
Green	Ready to be rebuilt	The merge has been built, but changes in source plugin files have been detected.
Green	Ready to be rebuilt [Forced]	The merge has been built, but marked as needing to be rebuilt by the user.

Rating Colors

<i>Color</i>	<i>Meaning</i>
Grey	No user reports on this plugin. Merge rating unknown.
Black	Plugin blacklisted. This plugin should never be merged under any circumstances.
Red	Average rating ~0.0, failure to merge
Orange	Average rating ~1.0, dysfunctional
Yellow	Average rating ~2.0, partially functional
Olive	Average rating ~3.0, tweaking required
Green	Average rating ~4.0, merges perfectly

Also see Plugin Ratings.

Plugin Ratings

Rating	Notes
-1: Blacklist	Only recommend a plugin be blacklisted if you're certain there are no circumstances in which it will ever be mergeable. This is for plugins that absolutely cannot be merged. Please explain why you feel the plugin should be blacklisted in the notes section.
0: Failure	Failure to merge means merging the plugin fails completely. E.g. the merged plugin causes CTDs, major stability problems, or merging fails to create a plugin at all.
1: Dysfunctional	Dysfunctional means the merge doesn't work ingame, with major aspects of the plugins that were merged not working or being entirely missing.
2: Partially functional	Partially functional means the majority of the plugin content works but some of it doesn't. This would be if the plugin achieves its goals but has bugs that were created in merging.
3: Tweaking required	Tweaking required means you got the plugin to work perfectly in a merge, but had to tweak some things to get it to work. E.g. you modified the plugin or its asset files.
4: Perfect	Perfect means all aspects of the plugin work perfectly right away and you didn't have to do any tweaking. Nothing missing, nothing broken.

File-specific Assets

The following table lists the paths of all file specific assets that Merge Plugins can handle. Anything that is put into <angled brackets> is a placeholder.

Path	Description
<filename>.bsa	A Bethesda archive file.
<filename>.ini	An INI file, which can hold the same values as SkyrimProfile.ini. Loaded by the game after SkyrimProfile.ini.
seq\<filename>.seq	Helps the game boot up game-start-enabled quests.
interface\translations\<filename>-<language>.txt	An MCM translation for the specified language.
meshes\actors\character\facegendata\facegeom\ <filename>\<FormID>.nif	FaceGenData for the shape of an NPC's face. Associated with the record at <FormID>.
textures\actors\character\facegendata\facetint\ <filename>\<FormID>(.dds or .tga)	FaceGenData for the tinting of an NPC's face. Associated with the record at <FormID>.
sound\voice\<filename>\<VoiceType>\ <Name>__<FormID>_<#>(.xwm, .lip, or .fuz)	Voice data for how an NPC speaks.
Scripts\ <Name>_(TIF, QF, or SF)_<name>_<FormID>.pex	Script fragment for a dialogue topic-info record (TIF), quest record (QF), or scene record (SF).

Merging and Renumbering Methods

Renumbering

Renumbering methods correspond to how the application rennumbers FormIDs when merging to prevent collisions. The method you want to use by default is Renumber Conflicting, which only rennumbers FormIDs that conflict with each other, and, by definition, does not affect any FormIDs in the first plugin in a merge (the one at a lowest index).

Renumber All rennumbers every FormID in the mods being merged, and is less useful. It is still valuable for testing and is the only valid option for use with the New Records record copying method.

Record Copying

Record Copying methods correspond to how the application copies records to a merged plugin. The Overrides method is the method that has been used since 2013, and copies records to the merged plugin as overrides. This is the method you should usually use.

The New Records copying method has the program copy non-override records from the plugins being merged into the merged plugin as new records. This method has an advantage over the Overrides method in that it doesn't require the clamping of FormIDs after merging. FormID clamping is in an inexact process and can, in extremely rare cases, cause issues after merging. You should use the New Records merging method if the Overrides method produces a plugin with errors.

Limitations

Doesn't Do Conflict Resolution

A common misconception is that Merge Plugins resolves conflicts like a compatibility patch, similar to Wrye Bash or the TES5Edit Merged Patch. Merge plugins does not do this. Merge Plugins combines plugins following the rule-of-one so you can disable them from your load order, freeing up space for more mods. This is very different from a bashed patch or merged patch in that it completely replaces the plugins that are merged.

You can (and should) use compatibility patches with your Merges. Merge Plugins takes a set of plugins and combines them in a way that corresponds to how those plugins would operate in your game if they were separate, but in a single contiguous block in your load order. Some information is lost when combining plugins, so building a bashed/merged patch after combining plugins will not yield the same results as when the plugins were separate. If you want to have a merged plugin with conflicts resolved you need to include a compatibility patch created only for the plugins you're merging in the merge.

255 Plugins Limit

You cannot load more than 255 plugins into Merge Plugins. This is due to an engine limitation with how the game (and xEdit) represent FormIDs – the IDs associated with records. There is no way to easily bypass this, so you're just going to have to deal with it. Load only the plugins you plan on merging if your load order is extremely large, uninstalling/disabling them after merging.

Note: Each merge you build will also use up another of the slots going towards the 255 plugin limit. So, if you have 250 plugins loaded in Merge Plugins you can build a maximum of 5 merges before running out of plugin slots.

Conflicting Navmeshes

The game merges NAVI records at run-time. Merge Plugins isn't capable of merging NAVI records, so it follows the rule-of-one. This means that mods which add navmeshes in the same region will need to be manually handled in the CK after merging. All you need to do to fix this issue is open the merged plugin in the CK and save it.

The consequence of not handling conflicting navmeshes after merging is NPCs will be unable to move in areas the mods you merged modified. This often isn't too big of an issue, but can be serious in certain cases.

Plugins Must be Loaded

You cannot build a merge unless all of the source plugins for it are loaded. In the past you could merge a merged plugin or merge into an existing plugin, but that is no longer supported due to issues with asset priority and override record priority. You'll have to load all plugins associated with a merge in order to rebuild it, even if you're only removing or adding a single plugin.

Master Dependencies

Once you merge a plugin, any plugins that were dependent on that plugin (used it as a master) will no longer be able to function. This is a consequence of how master dependencies work, and cannot be resolved by simply changing the master dependency to the merged plugin because of how FormIDs are renumbered during merging. Merge Plugins will notify you and not allow you to proceed with a merge if it breaks a master dependency with a loaded plugin.

You have three options when dealing with master dependencies and merges:

1. Include plugins that depend on plugins in your merge to the merge.
2. Remove the plugin which other plugins depend on from your merge.

Note: If you generate a patch for your entire load order (e.g. a SkyProc patch, or a Bashed Patch) that should not be included in your merge.

Merged Plugins and Saves

Merging plugins is equivalent to uninstalling and reinstalling those plugins. That means that any content associated with those plugins in a current play-through will disappear, including hair, items, quest progress, NPCs, etc. It is highly recommended to merge plugins before making a lot of progress in your game.

Mods that Use DLLs

There aren't many mods that use dynamic link library files (.dll files), but those that do cannot be merged. Whether or not this can be resolved in the future has not yet been determined.

GetFormFromFile isn't Handled

Some papyrus scripts use `GetFormFromFile` to get a record reference in the context of the script. This call includes both the filename and the FormID to be targeted. If a plugin is targeted by this sort of script call and is then merged, the `GetFormFromFile` calls will stop working. Fixing this requires a very large and complex feature. This sort of feature is on the horizon, but isn't the current priority. For now all you can do is avoid merging mods that are targeted by other scripts using `GetFormFromFile`. There is currently no easy way to recognize whether or not a mod is targeted by `GetFormFromFile`, so the community will have to use user reports to identify and share information on which mods are targeted in this way.

Arguments

The following table lists arguments you can use to adjust how Merge Plugins starts. You can apply these arguments to shortcuts to the Merge Plugins executable or in Mod Organizer.

Argument	Description
-profile {ProfileName}	Use this argument to start the program using the profile specified by {ProfileName}.
-offline	Use this argument to start the program in offline mode. In this mode the program will not attempt to connect to the backend server under any circumstances.

Arguments with Mod Organizer

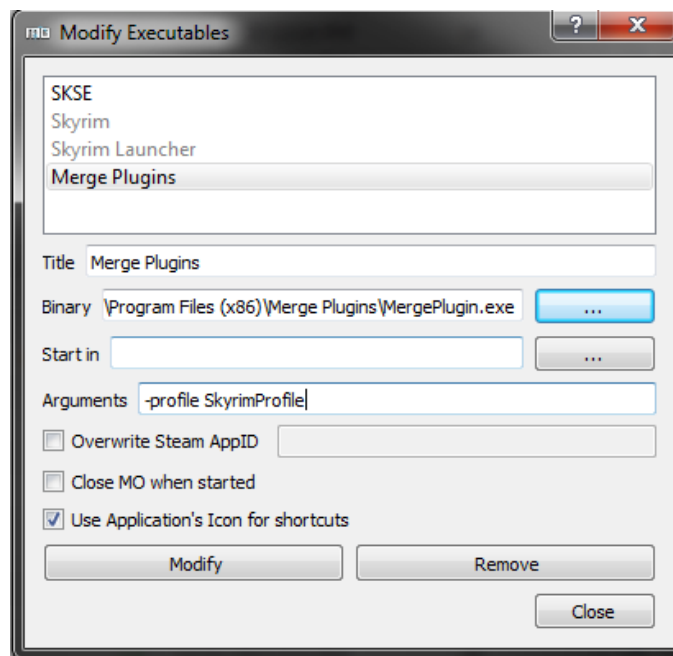


Illustration 17: Adding arguments through Mod Organizer.

If you use Mod Organizer you can configure the Merge Plugins executable to have the arguments you want from the Arguments text box in the Modify Executables Window. See [Starting Merge Plugins through Mod Organizer](#) for more information.

Arguments with Shortcuts

If you don't use Mod Organizer you can apply a target line parameter to a shortcut to Merge Plugins.

1. Go to the directory where you installed Merge Plugins, right click the Merge Plugins executable, and choose Create Shortcut.
2. Move the shortcut wherever you want, and rename it as you see fit.
3. Right-click the shortcut and choose properties.

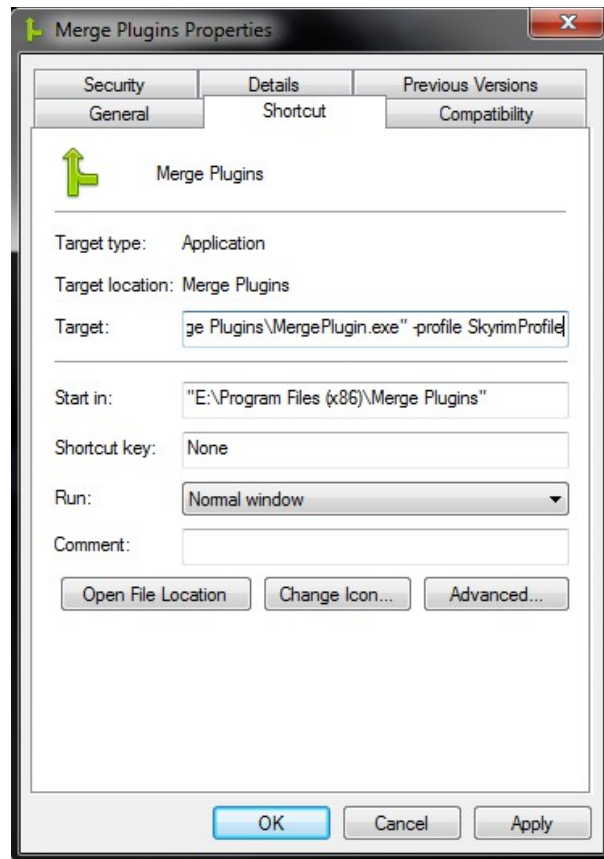


Illustration 18: Adding a target line parameter to a shortcut.

4. At the end of the Target text box type a space and then the target line parameter you want to use. E.g. -profile SkyrimProfile

Tutorials

Merging with a Mod Manager

Using Mod Organizer or Nexus Mod Manager v0.60a+ will allow you to set up the Mod Manager integration, which helps Merge Plugins to better interface with your mods.

Mod Organizer and Nexus Mod Manager's virtual mod installations allow Merge Plugins to associate assets with specific ESP files. As a result, you can easily create entirely independent, encapsulated merged mods where you'd normally have to do manual asset copying.

Mod Organizer Advantages

Mod Organizer's arbitrary BSA loading allows you to create merges without having to extract the BSAs of the ESPs merged. This will save on hard drive space, keep your merged mods cleaner, and allow you to specify asset priority between the assets in the BSAs from the merged plugins after merging. Using the Mod Organizer integration also requires fewer post-merge steps, with merged mods made available in Mod Organizer's mods list immediately after merging.

Mod Organizer Integration Set Up

Starting Merge Plugins through Mod Organizer

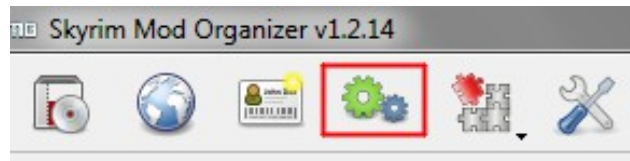


Illustration 19: The Modify Executables button in Mod Organizer.

1. Click on the Modify Executables button.

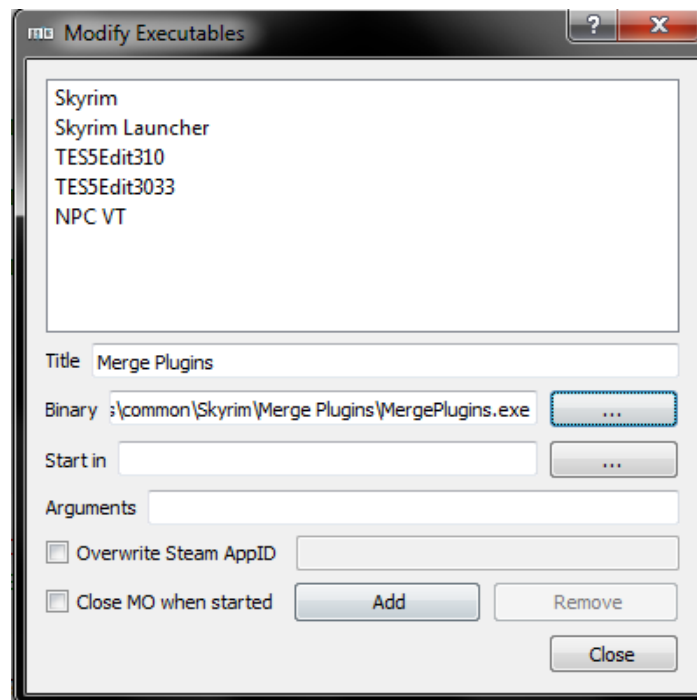


Illustration 20: Adding the Merge Plugins executable.

2. Enter the title “Merge Plugins” and click the first [...] button to specify the location of the Merge Plugins binary.
3. Click Add and close the window.

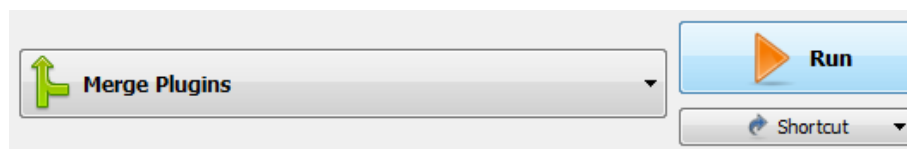


Illustration 21: Running Merge Plugins through Mod Organizer.

4. When you want to start Merge Plugins, select it from the drop-down menu and click Run.

Configuring the Mod Organizer Integration

1. Run Merge Plugins through Mod Organizer.
2. Open the [Options Window](#) and go to the Integrations Tab.

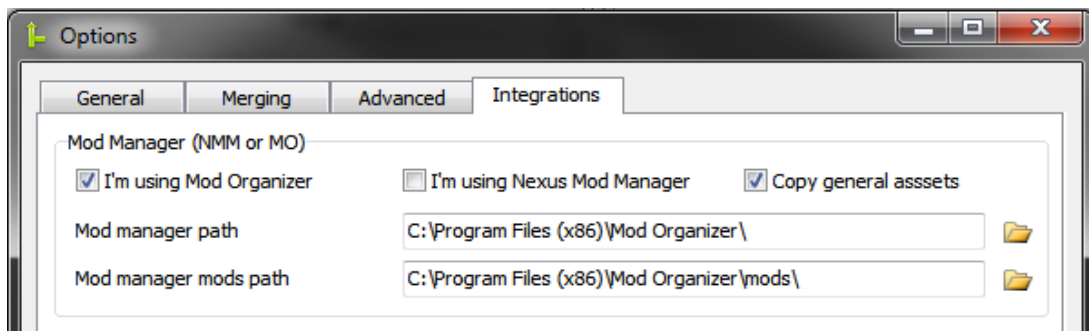


Illustration 22: Setting up the Mod Organizer Integration.

3. Check **I'm using Mod Organizer** and click **Detect integrations**. If your Mod Organizer installation is not found automatically, click on the folder icon by the **Mod manager path** text box and browse to its location on your file system. Do the same for the **Mod manager mods path**, browsing to the location of your mods folder if it is not the default *Mod Organizer\mods* folder.
4. Verify the paths are correct.
5. Proceed to the [Merging Tab](#) of the Options Window and specify the **Merge destination directory**. It is best to set this to Mod Organizer's mods folder because merges will then be created as mods directly available through Mod Organizer's interface, without requiring any post-merge steps.
6. Restart Merge Plugins to correctly load your Mod Organizer load order.

Mod Organizer Post-Merge Steps

With Copy General Assets

1. If you don't see the merged mod in Mod Organizer, right-click → All → Refresh.
2. Enable the merged mod in Mod Organizer. It should be the last mod in your list, before Overwrite.
3. Disable the mods you merged in Mod Organizer. For mods that have multiple ESPs in them, if you didn't include all of those ESPs in your merge you'll have to leave the mod enabled and disable the ESPs you merged from your load order instead (in the plugins tab). It's not recommended to merge only a few ESPs from a mod because it will create asset duplication.
4. Go to the Archives tab in Mod Organizer and enable all BSA archives associated with the merged mod you just created. Be sure to order them properly as well by dragging and dropping them.

Without Copy General Assets

1. If you don't see the merged mod in Mod Organizer, right-click → All → Refresh.
2. Enable the merged mod in Mod Organizer. It should be the last mod in your list, before Overwrite.
3. Disable the ESPs you merged from your load order.

Nexus Mod Manager Integration Set Up

Configuring the Nexus Mod Manager Integration

1. Open the [Options Window](#) and go to the Integrations Tab.

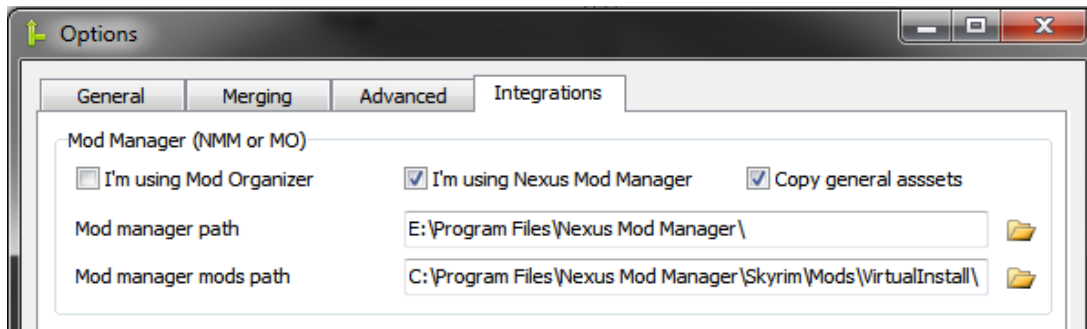


Illustration 23: Setting up the Nexus Mod Manager Integration.

2. Check **I'm using Nexus Mod Manager** and click **Detect integrations**. If your Nexus Mod Manager installation is not found automatically, click on the folder icon by the **Mod manager path** text box and browse to its location on your file system. Do the same for the **Mod manager mods path**, browsing to the location of your Virtual Install folder.
3. Verify the paths are correct.
4. Proceed to the [Merging Tab](#) of the Options Window and specify the **Merge destination directory**. You can set this to whatever you want, but I'd recommend something like "C:\Games\Nexus Mod Manager\Skyrim\Mods\Merges".
5. Nexus Mod Manager doesn't do any special handling of BSA files. Because of this, you need to use either the Extract BSAs or the Build merged BSA setting. You can enable these in the [Merging Tab](#) of the Options Window. You should only enable one of these options.

Nexus Mod Manager Post-Merge Steps

With Copy General Assets

1. Go to your **Merge destination directory**, navigate into the folder named after your merge, and add all of the files and folders to an archive. Install the archive into Nexus Mod Manager.
2. Enable the merged mod in Nexus Mod Manager. It should be the last mod in the list.
3. Uninstall the mods you merged in Nexus Mod Manager. For mods that have multiple ESPs in them, if you didn't include all of those ESPs in your merge you'll have to leave the mod enabled and disable the ESPs you merged from your load order instead (in the plugins tab). It's not recommended to merge only a few ESPs from a mod because it will create asset duplication.

Without Copy General Assets

1. Go to your **Merge destination directory**, navigate into the folder named after your merge, and add all of the files to an archive. Install the archive into Nexus Mod Manager.
2. Enable the merged mod in Nexus Mod Manager. It should be the last mod in the list.
3. Disable the ESPs you merged from your load order.

Integration Set Up

All integrations can be automatically detected by Merge Plugins. Merge Plugins searches for the Decompiler, Compiler, and BSAOpt integrations in its folder and in your game path up to two folders deep. Merge Plugins searches for the papyrus flags file at *scripts\source\TESV_Papyrus_Flags.flg*. Merge Plugins searches for Mod Organizer in your game path (up to two folders deep) and in both **:\Program Files* and **:\Program Files (x86)*.

Papyrus

The papyrus integrations are only used for handling Skyrim script fragments. If you don't want to handle Skyrim script fragments, you don't need to set them up.

Champollion Decompiler

1. [Download Champollion](#).
2. Install it somewhere by extracting the archive into a folder. I recommend installing in a folder in your Skyrim directory or in the Merge Plugins directory.
3. Click the Detect Integrations button, the program should find Champollion.exe automatically. If it doesn't, click the browse button and go to the location of Champollion.exe, select it and click OK.
4. Don't forget to endorse Champollion on Nexus Mods!

Papyrus Compiler

1. [Download the Creation Kit from Steam](#).
2. Click the Detect Integrations button, the program should find PapyrusCompiler.exe automatically. IF it doesn't click the browse button and go to the location of the Papyrus Compiler executable, select it, and click OK.
3. If you're merging mods that use SKSE, you should also install the SKSE source files.

Papyrus Flags

1. [Download the Creation Kit from Steam](#). If you already have it, make sure it is up-to-date.
2. Find the scripts.rar archive in your Skyrim data directory.
3. Extract the archive. It should create a file *Data\scripts\source\TESV_Papyrus_Flags.flg*
4. Click the Detect Integrations button, the program should find TESV_Papyrus_Flags.flg automatically. If it doesn't, click the browse button and browse to the location of the TESV_Papyrus_Flags.flg file, select it, and click OK.

BSAOpt

You only need the BSAOpt integration if you plan on using the Build merged BSA option.

1. [Download BSAOpt v2.0 beta.](#)
2. Install it somewhere by extracting the archive into a folder. I recommend installing in a folder in your game directory or in the Merge Plugins directory.
3. Click the Detect Integrations button, the program should find BSAOpt automatically, regardless of the version you installed. If it doesn't, click the browse button and go to the location of the BSAOpt executable, select it and click OK.
4. You may want to configure the BSAOpt command line options. Refer to the BSAOpt help for help configuring these. (you can also just leave them at the defaults)
5. Don't forget to endorse BSAOpt!

Rebuilding Merges

When to Rebuild a Merge

You should rebuild a merge when:

- a) You download a new version of one or more mods in the merge.
- b) You add a new plugin to the merge.
- c) You remove a plugin from the merge.
- d) The merge has issues ingame and you want to try building it with different options set.

It's not necessary to rebuild merges when the program updates unless your merge didn't work and you want to try it again with the new version.

Effects of Rebuilding a Merge

Rebuilding a merge will delete all files associated with that merge. Nothing from the previous build of the merge will be preserved. In addition, all properties of the merge will be updated (e.g. plugin hashes, and record copying failures).

Rebuild Status Detection

The application will detect that a merge needs to be rebuilt if one or more plugins associated with it have been modified (detected through their CRC32 hash) or if the merged plugin has been deleted.

Note: you can ignore or force rebuild statuses by right-clicking a merge and choosing [Toggle rebuild status](#).

Updating

Updating the Program

1. Click the [Update button](#) in the Quickbar, or go to the [Options Window](#) and click Update Program.
2. Look over the changes in the [Changelog Window](#). If you're cool with the changes, click Install.
3. The program will download the new version of the program from the backend, install it, and restart itself. **NOTE:** If you're using a recent version of Mod Organizer, the program restarting may cause Mod Organizer to stop applying its virtual directory structure. If this happens you'll have to close and restart the program manually.

Updating the Dictionary

1. Click the [Update button](#) in the Quickbar, or go to the [Options Window](#) and click Update Dictionary.
2. The program should update the plugins list view if necessary, using the new dictionary definitions.

Automatic Updating

If you want to have the program automatically update itself or the dictionary file, you can check the appropriate options in the [General Tab](#) of the Options Window. If these options are enabled, automatic updating will occur when the [Splash Window](#) is displayed during application startup.

Submitting Reports

When to Submit Reports

When you've built a merged and tested it ingame thoroughly, it is a good idea to submit reports on the plugins in that merge, even if reports have already been submitted on them. Submitting reports in this way helps other users to know what they can and cannot merge. Don't submit reports unless you tested thoroughly, incorrect reports aren't very nice.

Please make certain to note if the plugin you're reporting on was the first plugin in your merge, whether you used the [Handle Script Fragments](#) option, and any bugs you encountered ingame.

How to Submit Reports

You can submit reports on all of the plugins in the merges you've built by clicking the [Report on Merges Button](#) in the Quickbar. You can also use the [Merges Popup Menu](#) to report on individual merges, and the [Plugins Popup Menu](#) to report on individual plugins.

You can update a report on a plugin by submitting a new report on the same plugin.

Debugging

Logs

The program saves logs into the logs folder, which is located alongside the executable. You will be asked to provide a log whenever you submit a help request on Nexus Mods, GitHub, or elsewhere. Logs provide a means to understand what the program is doing and what may be going wrong. If you take the time you can learn to spot issues in your own logs instead of having someone else evaluate them for you.

Folder	Description
<i>logs\merge</i>	Logs from merges.
<i>logs\errors</i>	Logs from error-checking and error-fixing tasks.
<i>logs\main</i>	Logs from the main program. This includes saving logs and the log that is displayed in the Log Tab of the Main Application Window.

Debug Options

In the [Merging Tab](#) of the Options Window you can find several debug options. These debug options make Merge Plugins print more messages to the log. Enabling these debug options can help you and others to identify issues with Merge Plugins.

Reading the Documentation

Issues can occur because you don't know what you're doing. For this reason, I recommend you read this documentation thoroughly *before* asking for help. I know, it's long but it's an extremely valuable source of information. Your ability to resolve issues and create merges successfully will be greatly improved if you read this documentation thoroughly. Help me to help you – RTFM.

Community

Making a Translation

Merge Plugins Standalone offers a highly extensible, simple translation system. If you want to make a translation, you need only translate the strings in the *lang\english.lang* file to your chosen language and save it as a new file. From there, you can switch to your new language from within the program from the language drop-down.

Latest english.lang

If you want to make a translation, I recommend you get the [latest english.lang file from GitHub](#). This will make certain your translation doesn't need to be updated anytime soon.

Notepad++ Environment

If you use Notepad++, you can use my user-defined language and custom theme when editing language files to make your experience of making a translation easier and more efficient.

Custom Style

1. Download the [inverted.xml file](#) from GitHub
2. Install it in the themes folder in your Notepad++ program directory, e.g.
C:\Program Files (x86)\Notepad++\themes
3. Select the theme from Settings → Style Configurator in Notepad++

User-defined Language

1. Download the [lang-markup.xml file](#) from GitHub
2. Start Notepad++ and go to Language → Define your language, then click the Import button and navigate to the lang-markup.xml file you downloaded
3. Once you language is imported, close the User defined language window and click on Language → Lang

UTF8

If you're translating into a language that has special non-alphanumeric characters (e.g. any character not in standard [ASCII](#)), you'll need to save your file with UTF8 encoding. How you do this will depend on the text editor you're using. I recommend using Notepad++. If you're using Notepad++ you simply need to go to the Encoding tab, choose “Encode in UTF8”, and then save the file.

Syntax

There is some basic syntax to language files. Below are some symbols and their meanings.

Symbol	Meaning
=	Used to separate name value pairs. You should never translate names, only values.
%s, %d	Used to substitute values. Don't translate these. You can view them in context to adjust their positioning in your translated string to fit your language's grammar.
{ }	Used for comments. Don't translate these.
	Used to separate titles from descriptions for certain hints. Some strings have trailing pipes because they're used in constructing hints.
#13#10	Replaced with a new line in the program. You'll want to use these with long strings.

Development

GitHub, Hipchat, and Forums

There are several places where you can participate in the development of Merge Plugins.

- **GitHub** provides a means by which you can follow Merge Plugins [development](#) and [issues](#). This is a great way to get an idea of how development is progressing.
- **Hipchat** is a program that provides a network of chatrooms. The xEdit Hipchat has a room for Merge Plugins as well as general Chit-chat. If you want to get help with the program or get involved in the development process, you can join as a [guest](#).
- **Forums** are used by most users. Merge Plugins discussion takes place on the Nexus Mods forums, STEP forums, AFKMods forums, and Bethesda forums. Feel free to post in these topics whenever you need help, or to keep up-to-date with the latest news and developments.

Reporting Issues

If you feel you may have found an issue in Merge Plugins, I ask that you submit it in a certain fashion. This is to help me to resolve issues in a timely manner and keep my sanity intact. Developing a large application is a lot of work, and there is nothing quite so frustrating as an unhelpful bug report. Please, help me to help you to help us all by following these simple guidelines when submitting an issue:

1. **Clearly state what the bug is.** If I don't know what the bug is, there's no way for me to fix it.
2. **Clearly state when the bug happens.** I cannot find a tree in a forest if you do not tell me where the tree is. When a bug occurs during program execution can be roughly translated into an area in the codebase where it may be located.
3. **Provide logs.** Merge Plugins logs things extremely well. Go to the logs folder and get the log where the bug occurred and include it with your bug report.
4. **Provide screenshots.** Depending on the bug, screenshots may or may not be useful. If your bug has anything to do with the GUI, a screenshot can be valuable.

Please submit issues to the [GitHub issue tracker](#).

Feature Requests

Merge Plugins is a massive project. I really appreciate community input on features, but it's important to understand that not every feature that is presented to me will be made. Some features aren't feasible, and others are simply not useful enough to be worth the time required to build them. Please exercise the understanding that your feature requests may be accepted or denied.

Joining the Development Team

Merge Plugins Standalone is open source, and I would love the opportunity to work with other developers to improve it. If you have experience programming, and like what Merge Plugins does, please feel free to contribute to the repository on GitHub. I will gladly review and approve pull requests that include valuable additions to the application without compromising established style or architecture.

There are other ways to can help as well. E.g. testing features, proofreading documentation, submitting issues, making feature requests, etc. I appreciate any help I can get.

Service

Service Agreement

Merge Plugins Standalone is both a backend and a frontend. The frontend runs on your computer, and is what you interface, where the backend is run on a server that I manage with. The backend is what distributes program and dictionary updates, accepts your reports, and tracks user statistics so I can improve the software in the future. Your usage of the frontend is of no concern to me so long as you remain within the legal bounds of the license (the MPL). However, your usage of the backend service is of concern to me, and may be terminated at any time should you break from the following agreement.

1. You will not, under any circumstances, act to damage or slow down the server. E.g. by sending false data, or overusing bandwidth by requesting files repeatedly.
2. You will not send false or maliciously incorrect reports on plugins.
3. You will not interface with the server through any means other than the client service offered through the Merge Plugins application.

Should you be found in violation of any of these agreements, you will be blacklisted, and will no longer be able to make use of the backend service. I do not expect to encounter a large amount of abuse, but I have the infrastructure in place to deal with any abuse that does occur. Please be aware that your access to the backend service is not irrevocable, and malicious actions will not be tolerated.

Service Transference

Should I ever stop running the server that hosts the Merge Plugins backend, it is completely possible for another user to create a new server hosting the backend. The source code for the Merge Plugins backend is available on GitHub in the same repository as the Merge Plugins frontend, and there are also several resources for set up that should make it extremely easy.

Backend Set Up

1. Compile the backend using Delphi XE or newer.
2. Install MySQL 5.6, and set up a database and tables for the backend to use. You can run the [backend setup script](#) for setting up tables to the application's specifications.
3. Start the backend and enter the information necessary to access your MySQL database.
4. Make sure your server is visible to the outside world, and that port 960 is open.
5. Get a public domain or subdomain address, and bind it to the IP address of your server. You can do this through any of many services, such as [DynDNS](#) or [freedns.afraid.org](#). **NOTE:** this is optional, you can use an IP address instead, but using a DNS is highly recommended.
6. Instruct users to modify their *profiles\<ProfileName>\settings.ini* file, changing the serverHost value to the address of your server.

Privacy

Your privacy is important to me, and so is transparency. I understand that you may not be interested in allowing Merge Plugins to gain access to statistics about your usage of the program. Below is a listing of all the information that Merge Plugins gathers on users, and how you can opt out of them.

Statistic	Description	How to opt out
Times run	The number times you've run the application.	[1]
Merges built	The number of merges you've built.	[1]
Plugins checked	The number of plugins you've checked for errors.	[1]
Plugins merged	The number of unique plugins you've merged.	[1]
Plugins fixed	The number of plugins you've fixed errors in.	[1]
Reports submitted	The number of reports you've submitted to the backend.	[2]
IP Address	Your IP address.	[2]
Username	Your username.	[3]
First seen	The date and time you first connected to the backend.	[2]
Last seen	The date and time you last connected to the backend.	[2]
Times seen	The number of times you've connected to the backend.	[2]
Download	The number of bytes you've downloaded from the backend.	[2]
Upload	The number of bytes you've uploaded to the backend.	[2]

[1] Enable the [Do not send usage statistics](#) option.

[2] Use [the -offline argument](#). Doing this will make it so you cannot get dictionary or program updates through the program.

[3] Don't enter a username on the options window. If you do this, you will not be able to submit user reports.

License

Merge Plugins is [open source](#), and licensed under the MPL. Please refer to [the license on GitHub](#) for more information. Merge Plugins cannot be licensed under any license other than the MPL because xEdit is licensed under the MPL, and Merge Plugins relies on xEdit as an API.

Plain English

Check out [the tl;drLegal page on the MPL 1.1](#).