Inhalt

[First things first…. 2](#_Toc503698985)

[First steps 2](#_Toc503698986)

[Extracting the current story as a file 2](#_Toc503698987)

[Open the file with the story mod tool 3](#_Toc503698988)

[What can I actually do with the tool ? 4](#_Toc503698989)

[The GUI of the tool 4](#_Toc503698990)

[The Commands I can use for a VN frame 5](#_Toc503698991)

[How to create new events 10](#_Toc503698992)

[One more operator 11](#_Toc503698993)

[Import my modfile into AATOFL 12](#_Toc503698994)

[How to use the AATOFL mod manager 13](#_Toc503698995)

[How to upload a mod 13](#_Toc503698996)

[How to download a story mod 14](#_Toc503698997)

[We create a little story mod for AATOFL , a little example on how to do it. 15](#_Toc503698998)

Little modding tutorial for AATOFL

# First things first….

! Important notice !

I am not a professional game developer (important information) if you expect a full blown modding program like skyrim has i have to disappoint you. My story mod program is only able to change the story a little bit. It was developered with spellchecking in mind but it can do other things as well.

Currently only one mod file will be supported , it must be named \*StoryMod.js\* and must be located in the same directory as the AATOFL.exe. It will overwrite all found Events and will also inject new events into the story (if you created some.. but more on that later)

IMPORTANT : if you want to transfer all the changes from the old spellchecking tool then you just have to start the game with the old spellchecking function activated and then extract the mod file.

Also important… the tool is not foolproof. If you do stupid things then you will get stupid results ☺

# First steps

## Extracting the current story as a file

The very first thing you want to do is to extract the current storyMod file to do this go into the game and start ANY vn event (maintext must be visible) andthen press F3 to call this menu…



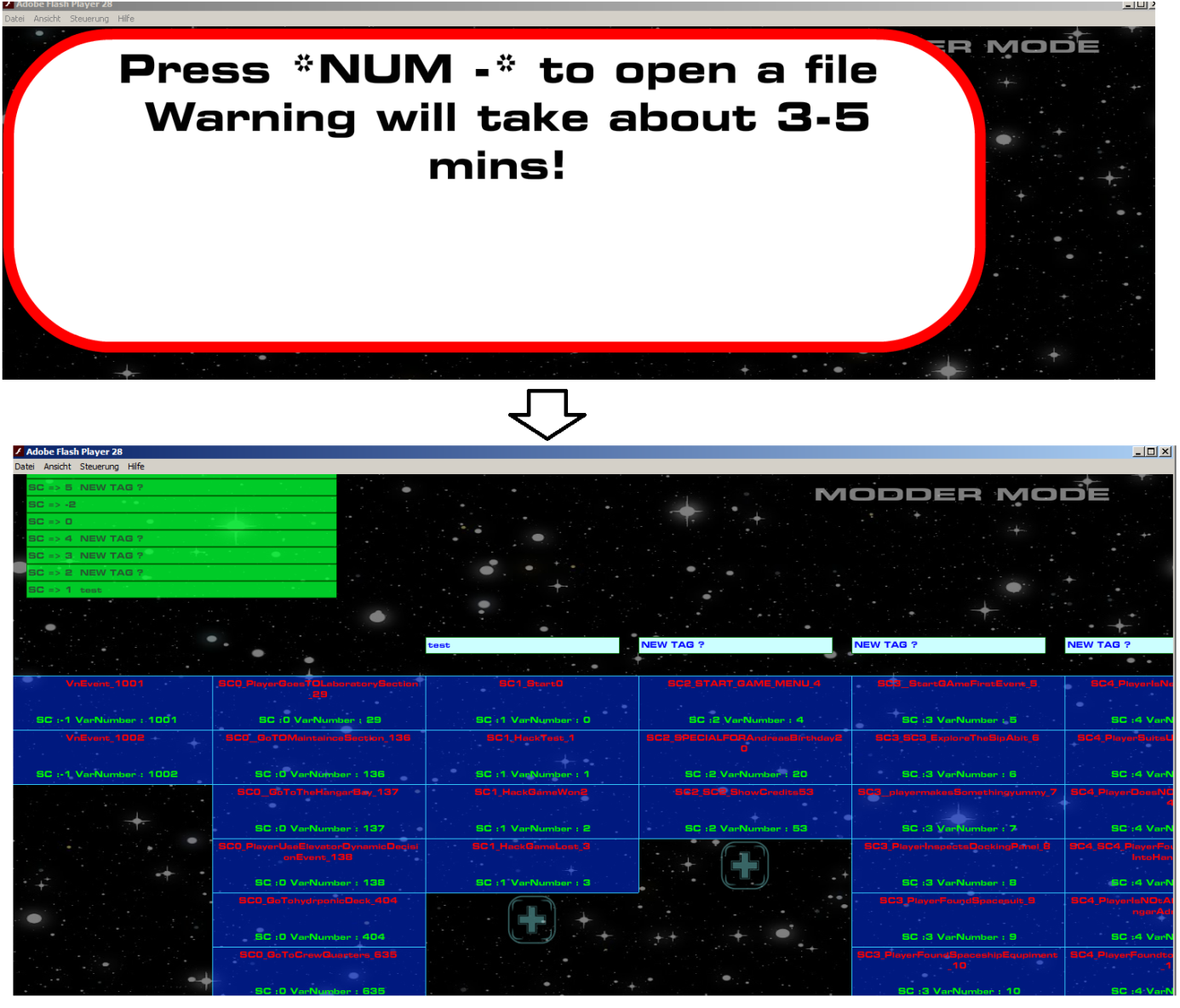
Now Press \*extract story mod\* and a new file will be created named \*storyEdit.js\* this file can be read with the modding tool.

## Open the file with the story mod tool

Important : ONLY OPEN THE MOD TOOL WITH A DEDICATED FLASH PLAYER… a browser will likely crash! (A decicated flashplayer should also be in the folder)

Now open the file with the story mod tool. TO do this start \*\*CreateStoryWithFlash0xx.swf\* and press - on the numpad to open this file.

You will see something like this…



Now you can start to work with the tool , just click on one of the events (the blue buttons) and then you can change a VN frame. A VN event consist of VN frames… simple as that.

If you want to save your changes you can press + on the numpad and then save into a file.

## What can I actually do with the tool ?

The tool has been invented for spellchecking since my English.. is… >\_<

Anyway , currently you are able to do anything I can do with the tool (I use the tool myself in developer mode) except you can NOT erase my VN frames since the base frames must stay intact to avoid null reference frames.

Unfortunately my framework is very… stupid. If you want to do more than just spellchecking then please use the extracted file as a example how to do it properly. Some commands are only very limited usuable since it depends on positions I create with my unity 3d program. (please keep in mind I created this framework for spellchecking)

IMPORTANT : the order is from bottom to top.. so the top entry is the last frame! (scroll with mouse wheel)

## The GUI of the tool



As mentioned you can not delete my frames you can only make them be skipped by the framework and you can NOT delete my events. You can only Delete your own events.

## The Commands I can use for a VN frame

There are several commands you can use for a frame. Some are not implemented and some are obsolete. Important is to know that EVERYTHING you do will be saved , that means if you use a function for Aemi to use a specific cloth style then function will trigger even after you saved and loaded the game.

The little plus icon on the line is your very best friend (2), it will show you the commands I used and you can also search with it if you just write a key word into the textfield and then press the little plus button.. for example :



..and then just select what you want to have from the command list.

SetParentForGameobject (not implemented yet)

StoryQuestText (this is the text shown if you press F1 in the game)

MakeSaveGameFile ( creates a savegamefile (will work but you can not use it))

Load level (not used)

Repos in other level (not used , obsolete)

MUSIC (set music , this is an old function I rarely use please use the little plus icon to see what music themes are available , the themses are not named but in numbers)

Sound\_Pos index etc… Plays a sound on current position (not music , also in numbers!)

Here is a current overview of the sounds I used:



FADE OUT SLOW (fades the screen to black slow)

FADE OUT (faster fade out to black)

FADE\_IN\_SLOW (Fades slow to the scrren from black)

FADE\_IN ()

SETSC (sets the current story counter , only important for triggers I set in unity 3d , I would not use this function to avoid trigger releated erros)

SC++ (same a above but it will increment the story counter)

IF\_SMART

(uff… okay this was my try to make something more dynamic , you can actually compare hashtable keys and if the compareision failes the frame will be skipped)

I only use it to see if a boolean value is false or true for example :



That means that the frame will be only played if Aemis hands are bound

You can do a little bit more like compare hashkeys but.. I never used it.

IF\_GO\_ACTIVE\_SMART

Checks if a certain gameobject is active , you can also use the ! operator in front of the gameobject you want to check.

Example :



Checks if the generic chastitzy trail level is currently NOT active (will skip the frame if it is)



And.. the other way around , it will skip the frame is the level is NOT active

IF\_NORMAL and IF\_INVERSE (you can delete one of them if you do not need the other)

THis was my early attempt to skip frames , it just checks if a hashtablekey is set (IF\_NORMAL)

..of if a hashtablekey is not set (IF\_INVERSE) -> you can not use the ! operator.

If name and if name text (just exchange the nametxt if a hashtablekey is set to true)

If story and if story text (just exchange the maintext to this sentence if a hashtable key is set)

Jump to story array

important function , you can jump to ANY event and with the release of the modding tool the current version and above can use the name of the event (just copy the name of the event and implement if in here)

SEt\_Main\_hashtablekey (obsolete please do not use anymore , because it can only be used once on a frame)

Set Main hashtablekey\_1 (you can set up to 10 hashtablekeys to any value , I mostly use Boolean values like false , true)

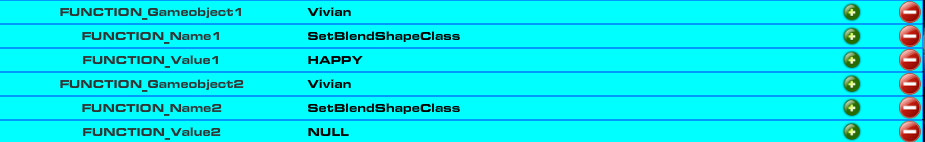
Jump to story array 1 and Jump to story array if key 1

Will jump to the story array ONLY of the hashtable key is set to true!

Function Gameobject 1..etc..

Uff… okay calls a specific function for a specific gameobject. You can call 10 functions each frame and please keep in mind that EVERY function and the value will be saved once you call it. They will be called again if the savegame loads. That means , if you set the blendshape key of Vivian to HAPPY she will be happy until this function is called again (I used it sometimes)

If you do not want it to be saved then please do this (example)



NULL is (mostly) the keyword for \*return\* since many of my function looks like this…



There are some exceptions but.. not many >\_< (just use the extracted story file as a big example)

Decision , Decision if key and Decision 11

Creates a simple decision screen like this one… (you can create up to ahm.. 8 buttons before it gets ugly xD)



Every entry will create one button to press and you can delet the \*DecisionIfKey1\* if you do not want to use it.

In the Decision11 textfield you can just paste the name of the event you want to call once the player pressed this button.

Activate gameobject (just activates/deactivates gameobject by name , true = activate , false is deactivate)

NPC turn to (makes the NPC turn to any named gameobject in the scene)

ANIM\_QUEUED (calles a animation sequence, I mostly use it for orgasm animations since the orgasm is a \*play it once\* thing and then I play a looped animation)

ANIM1 (plays a specific looped animation for a specific gameobject)

POSLERP (smoothly float from on position to another , barely/not usable for modders since you do not know where I placed the position markers)

POS1 (changes the position of gameobjects , also not really usable since you do not know where I placed all the markers in the scene , maybe I will create a function to place markers yourself if many of it want to)

LookTo (player looks to a specific gameobject)

SKIP\_THIS\_FRAME (skips this frame , please do not use this command and just press the \*delete current frame button\*

## How to create new events

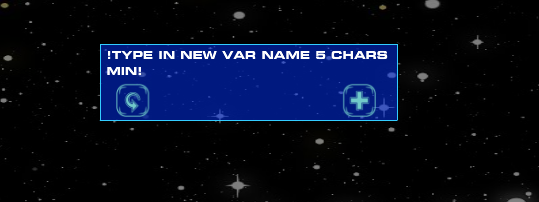
Simple , click on a PLUS icon in the main view of the modding tool and give the new event a name.

IMPORTANT : ONLY USE NORMAL CHARACTERS AND DO NOT USE THE SPACEBAR… the spacebar is evil.. ~\_~

Click..



Give the event a name (no spacebar..do not even touch it!)



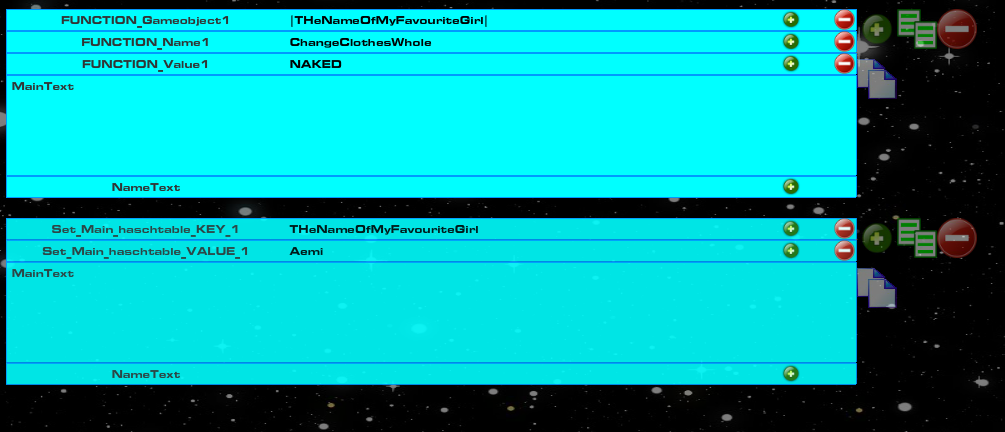
And you have something like this



## One more operator

To make it even a little bit more dynamic you can assign hashkeys a name of something (gameobject or a animation) and then call it by encapsuling the hashkey with |nameofhashkey| to call it by the value.

A little example : ( a stupid one btw…)

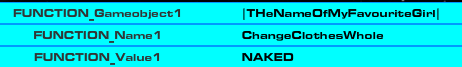


Okay.. some explanation

First I set the variable (hashkey) to a specific value , in this case \*Aemi\*



Then I encapsule the hashkey with |…|



What will happen is that the function will not read |THeNameOfMyFavouriteGirl| but instead it will ready \*Aemi\*.

This works everywhere!

You can use it with EVERY command not just functions. It is a general function which will read all things encapsuled with |…| and try to cast it to a string variable.

# Import my modfile into AATOFL

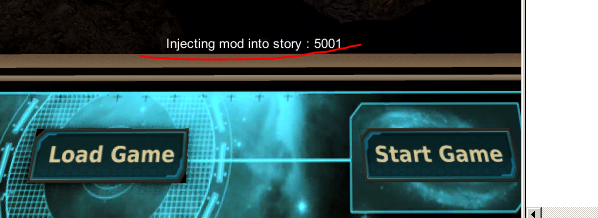
To import your modfile into AATOFL you need to save it first , so.. press numpad + and then save it into the folder where your AATOFL.exe is located. The filename MUST be \*StoryMod.js\*

After that you need to activate the inject story-mod function in AATOFL , just start AATOFL and then enable the story-mod injection in the game.



Then you need to restart the game and the story mod file will be read out on startup.. it will look something like this..





# How to use the AATOFL mod manager

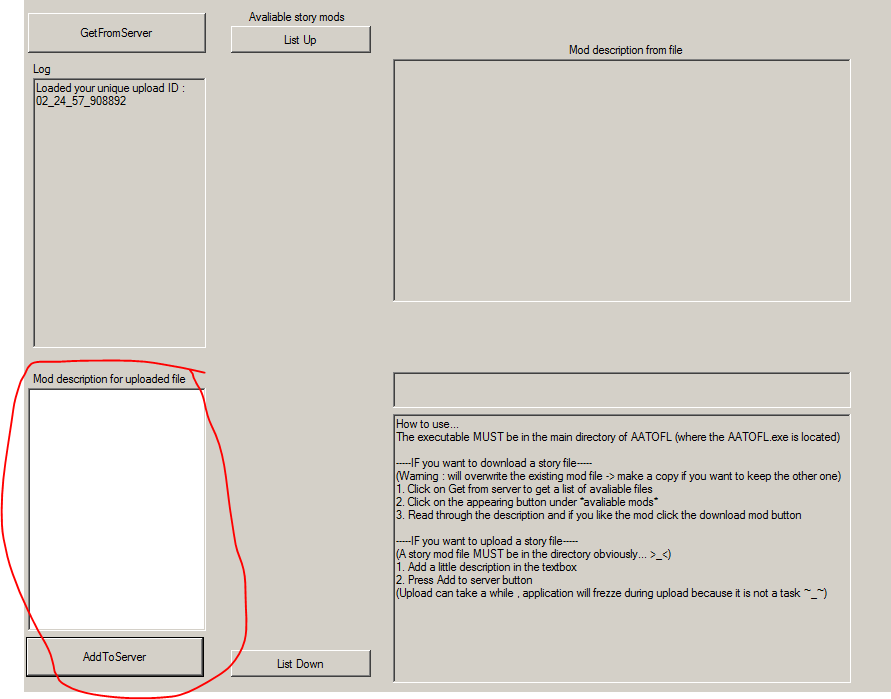
The AATOFL mod manager is a application to upload or download AATOFL story mods , it is connected to a mysql database and a FTP fileserver. The application must be in the same folder where your StoryMod.js file is so it can upload something onto the server.

If you start the application the first time you will get a unique id which will be used to identify you for the FTP fileserver. Furthermore there is a upload file size limit of 30 MB!

## How to upload a mod

You can only upload ONE file and if you upload another after that the file you uploaded will be overwritten!

To upload a file just write some usefull sentences into the description box and press AddTOserver

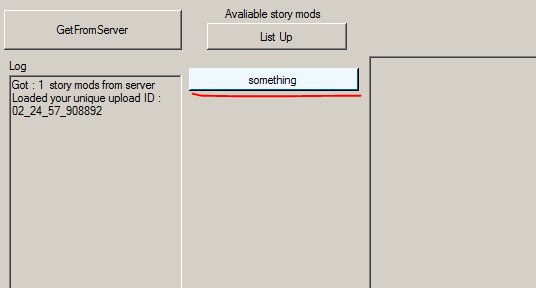


During upload the application will likely freeze (it will upload/download nevertheless , just wait) because I did not stuffed the FTP upload and download into a task. And I guess this application will not be used much because the game is not known by many people.

## How to download a story mod

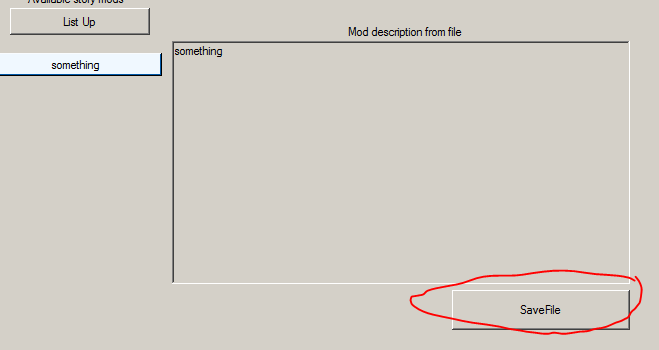
TO download a story mod please press the \*GetFromServer\* button to get a list of available story mods from the mysql server.

It will look something like that…



Press the buttons (Every button represents a mod file) who appeared after a couple of second and than read the description to see if the story mod is okay for you. If you want to download the file than press the \*download mod\* button , the application will also likely freeze during download because of my bad programming skills and laziness :D

After that a new button should appear where you can save the downloaded mod file.



Save it as StoryMod.js in the AATOFL main folder and please make sure you do not accidently overwrite your own mod file.

# We create a little story mod for AATOFL , a little example on how to do it.

Okay , this is just an example if you really… really want to use new events (even if it not really supported) and how you can make them appear for the player.

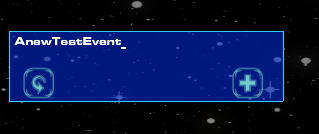
First extract the current story file , so start the game , start any VN event , press F3 and use the extract story file button. (game will freeze because big file)

After that we have the StoryEdit.js file in our main directory , we open it with the story mod program CreateStoryWithFlash0xx.swf (remember to use a dedicated player not your browser) and wait until the commands are available. ( a little message will appear in the background and after that you can use the plus icon to autofill commands)

Now we Create a new event.. just for fun..



Then..



Then we create some frames in it..



Then we copy the name if the event (SC64\_AnewTestEvent\_941)

I also created a second event to just go to the dialogue options please remember that the copy function..



..Is a good friend :D

Leave the frame screen



Then we go into the event where you speak with Aemi in the free roaming mode. (I know which event because… I just know Hihihi… :D)



In there we use the second last frame and insert our decisions in there. We will have two decisions one is leading to the event we just created and one will lead to the usual dialogues.

Here is the reworked frame…

BEFORE



AFTER



For your information , I deleted the decision\_if\_key since I do not need to check any hashkeys.

The CONTINUE Keyword is just to continue with the next VN frame.

Then I want to try it out Save with numpad + and name it StoryMod.js

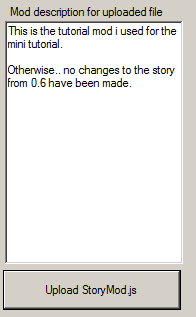


German GUi… >\_<

Then we try it out in game… (restart AATOFL , wait for injection and speak with Aemi in the free roaming mode)



And after that I upload this file so everyone can just download it as a little tutorial :D



And.. everyone can download it afterwards.. I hope >\_<

