Contents

[1 XML2 Console Build Tutorial 2](#_Toc110875356)

[1.0 Before Getting Started 2](#_Toc110875357)

[1.1 Getting the Project 2](#_Toc110875358)

[1.2 Folder Layout 3](#_Toc110875359)

[1.3 Importing a Default Character 5](#_Toc110875360)

[1.3.1 Creating the folder 5](#_Toc110875361)

[1.3.2 Importing initial assets 8](#_Toc110875362)

[1.3.3 Cleaning up the CFG files 12](#_Toc110875363)

[1.3.4 Moving the effects textures 14](#_Toc110875364)

[1.3.5 Setting up data files 20](#_Toc110875365)

[1.3.6 Importing X2UP Assets – Replacing existing files. 26](#_Toc110875366)

[1.3.7 Importing X2UP Assets – Adding new files 33](#_Toc110875367)

[1.3.8 Importing X2UP Assets – Bonus and console-specific files 39](#_Toc110875368)

[1.3.9 Importing PSP Assets 46](#_Toc110875369)

[1.3.10 Setting Up the Update Script 54](#_Toc110875370)

[1.3.11 Adding the character to data files 59](#_Toc110875371)

[1.3.12 Setting up the compile script 64](#_Toc110875372)

# XML2 Console Build Tutorial

## Before Getting Started

You will need the following programs:

* fbExtractor and fbBuilder (both installed to C:\Windows)
  + <https://www.mediafire.com/file/2eq99x8jazuudhj/NR_fb_pack.rar/file>
* Python v 3.8 or later (current is 3.10.6 as of 06Aug22), 64-bit version
  + When installing, be sure to add Python to PATH. **If you don’t know how, ask me**
  + <https://www.python.org/downloads/>
* Raven-Formats
  + Open command window after installing Python and type “pip install raven-formats”
* A GitHub account
  + <https://github.com/>
* The GitHub desktop app
  + <https://desktop.github.com/>

You will also need to get the various console-specific assets. You can download these as needed. The total file size is pretty big, so it may take some time to get them all. The XML2 assets are the ones you will certainly need, but you may need the MUA1 PS2/PSP assets possibly. Let me know if I need to upload the XML1 assets.

* <https://www.mediafire.com/folder/07oq0gku1ywjz/Console+Assets>

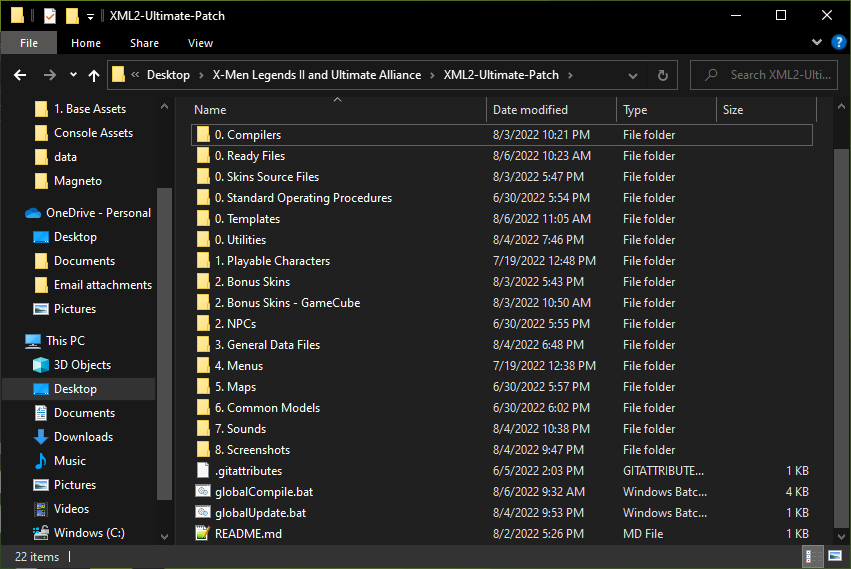
And lastly, you’ll need the assetsfb files from consoles to remove files as needed:

* <https://www.mediafire.com/folder/m8390kz9hi5ko/Console+assetsfb+files>

## Getting the Project

Will update these steps as we go. I’ll need to walk someone through it, so I remember how to do it.

## Folder Layout



The folder layout is pretty straightforward. Empty folders will not show up on GitHub currently, but this is the full planned layout. Here’s what’s in every folder:

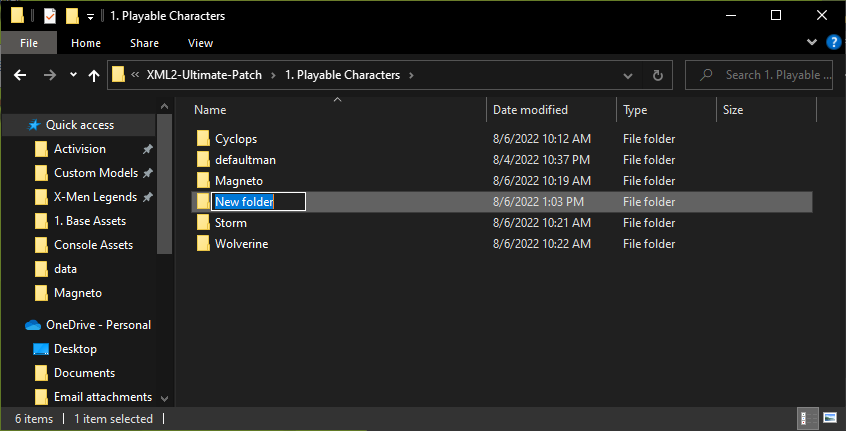
* 0. Compilers: has the necessary stuff to compile files. Don’t mess with this folder
* 0. Ready Files: the staging area for files that have been compiled. Once you run the compiler, you can take files from here and add them to the game.
* 0. Skins Source Files: The source files for the skins I’ve made for the X2UP.
* 0. Standard Operating Procedures: helpful documentation, including this tutorial
* 0. Templates: Where you can find templates for compiling different asset types (mainly characters right now)
* 0. Utilities: contains some additional utilities to help you set things up. Certain files used by the compiler process are stored here.
* 1. Playable characters: files for playable characters (as well as their NPC versions) get added here.
  + If a playable character (or there NPC version) has files that appear in a map or other place, they have a script that updates the files in the relevant places. More on that later.
* 2. NPCs: Files for NPCs that aren’t playable characters get added here. More on that later
  + NPCs will have scripts that updates their files in the maps where they appear. More on that later.
* 3. General data files: Permanent, permanent\_fightstyles, and any loose common data files can be found in this folder. More on this later.
* 4. Menus: any menu packages (as well as the loose textures for comic covers, loading screens, and concept art) get added here
  + Comic covers and loading screens don’t get added here manually. They are uploaded with their characters/maps and then sent to this place through an update script. More on that later.
* 5. Maps: map packages are found here, sorted by act and then map
  + 17Jul22: Currently nothing here
  + Any files from playable characters and NPCs don’t get uploaded here directly. They are uploaded in the character’s folder and then sent to the relevant map through an update script. More on that later.
* 6. Common Models: any models that are found in multiple places will be uploaded here
  + 17Jul22: Currently nothing here, don’t know if it will be used
* 7. Sounds: any sounds go here.
  + Nothing gets added here manually. Update scripts from the maps and playable characters send the files here.

## Importing a Default Character

To demonstrate how this works, I’ll be uploading Nightcrawler’s files.

### Creating the folder

* Go into the “1. Playable Characters” folder and add a folder with your character’s name.



A screenshot of a computer

Description automatically generated

* Copy “createFolders.bat” from the “0. Utilities” folder into your newly created folder.

A screenshot of a computer

Description automatically generated

* Double click it to run it. It will ask you questions about what folders you want to create. Type “Y” or “N” to answer the prompts.
  + All characters (X2UP, AXE, BHE, MUE) will have PC folders.
  + All X2UP characters except for Cable and Cannonball will have GameCube folders.
  + All X2UP characters will have PS2 folders.
  + All X2UP characters except for Cable, Cannonball, Dark Phoenix, Jubilee, Magma, and X-Man will have PSP folders.
    - Cable, Cannonball, Dark Phoenix, and X-Man will use tweaked versions of their default PSP mods for the PSP version
  + All X2UP characters will have Xbox folders.

A picture containing graphical user interface

Description automatically generated

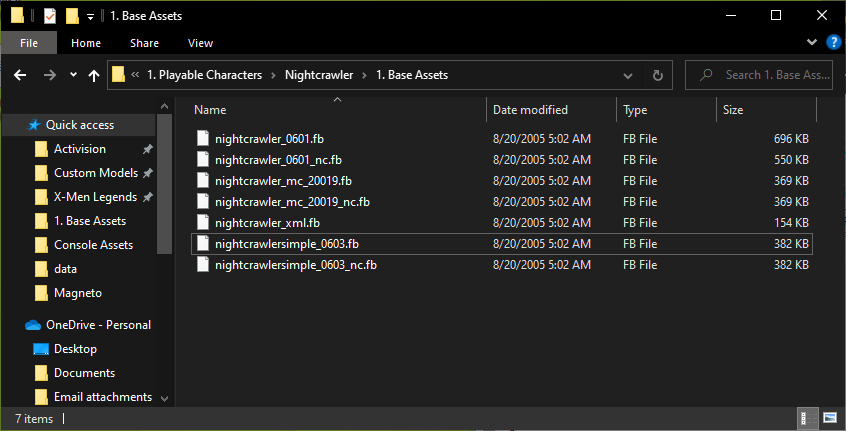
* You’ll now have up to 6 folders in your character’s folders. You can delete “createFolders.bat” now.

A screenshot of a computer

Description automatically generated with medium confidence

### Importing initial assets

* I prefer to start with console assets whenever importing characters. Obviously, non-default characters don’t have console assets (and some won’t be getting any), but for default characters it’s much easier to start with that. For non-default characters, I can create a separate tutorial that covers the principles, but it’s essentially the same process.
* So first, get the character’s FB packages from the PS2 version of the game (XML2 PS2 Assets.rar). I provided the MediaFire link for the console assets at the beginning of this tutorial. Get any NPC packages, their name\_xml package, and the packages from their first skin, and add them to the “1. Base Assets” folder.

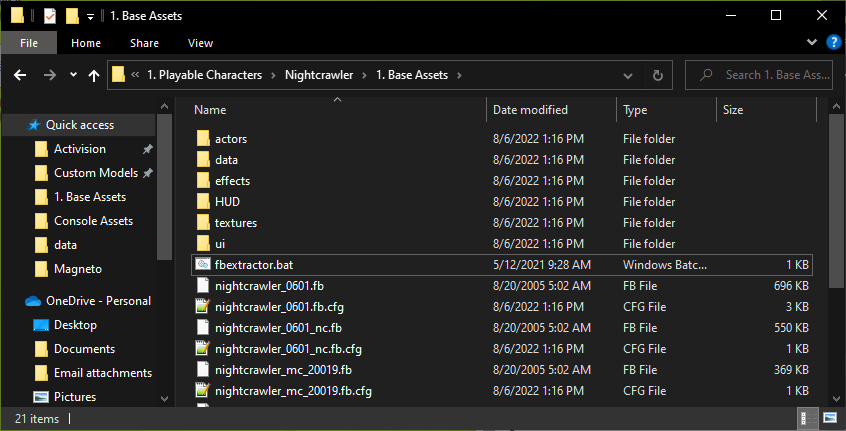


* In this case, “NightcrawlerSimple” uses skin 0603, but we’re changing him to use skin 0601. So I’ll go ahead and rename those packages.

A screenshot of a computer

Description automatically generated with medium confidence

* Now you can copy in “fbextractor.bat” from the “0. Utilities” folder into the “1. Base Assets” folder. Double click it to run it. It will ask if you want to create pkg files. Type N for no. It will then dash through several lines of code, after which a bunch of new files and folders will be created.



* You can now delete “fbExtractor.bat” and the fb files. Keep the cfg files and the folders that were created.

A screenshot of a computer

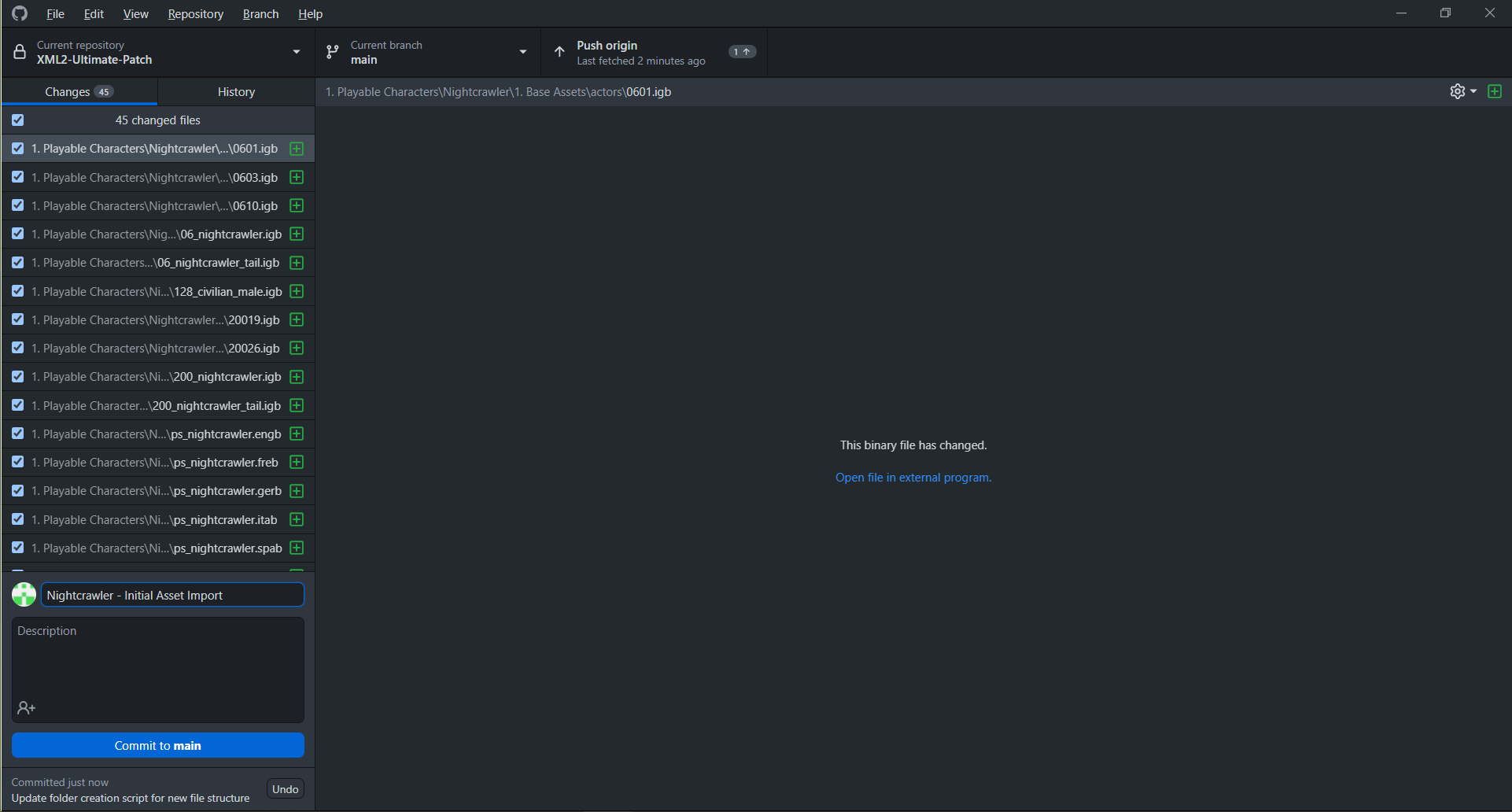
Description automatically generated with medium confidence

* Next, you’ll want to do some reorganization of the files. For starters, change the name of the “HUD” folder to “hud”, and do the same for “ui/HUD” (changing it to “ui/hud”)

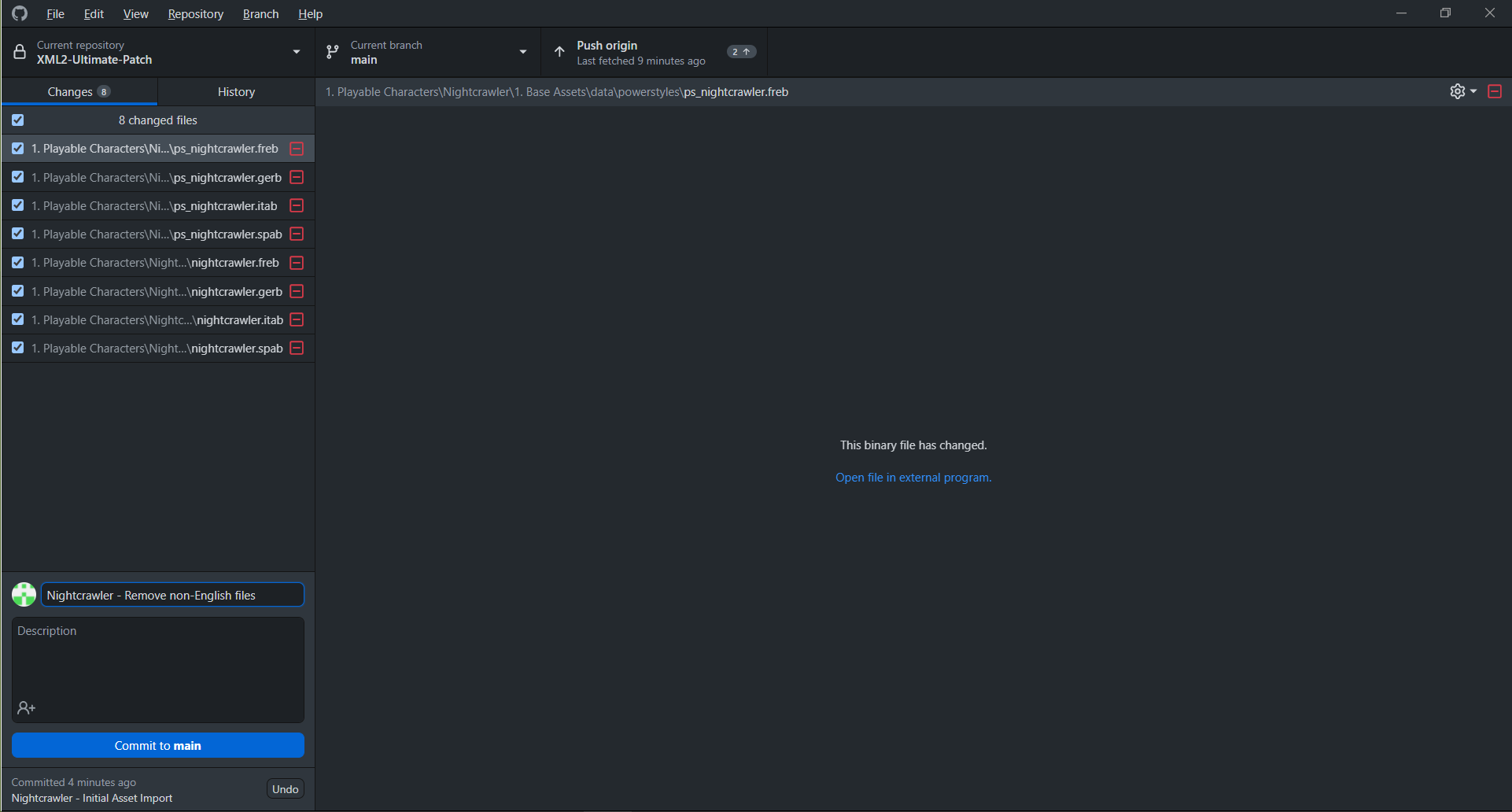
A screenshot of a computer

Description automatically generated with medium confidence

* Now is a good time to do your first commit to GitHub on the desktop app. For the short description, you can name it something like “(character name) – Initial Asset Import”, and then click the “Commit to main” button



* Next, you’ll want to remove any non-english data files. These are .freb, .gerb, .itab, and .spab files. In this case, Nightcrawler only has talents and powerstyles, but check your character for other files, including an NPC powerstyle. You can now make another commit to GitHub indicating that you removed non-English files.



### Cleaning up the CFG files

* Next, you’ll want to do some cleanup of the cfg files. Open them up and change any instance of “HUD” to “hud”. If you changed the number of an NPC package (such as how I changed NightcrawlerSimple from 0603 to 0601), you’ll want to replace any instances of the old number with the new one. Then, you’ll want to alphabetize them. An easy way to do this is to go to <https://alphabetizer.flap.tv/>. Once you’ve done that, you can remove any non-english data files from the cfg files.

Text

Description automatically generated

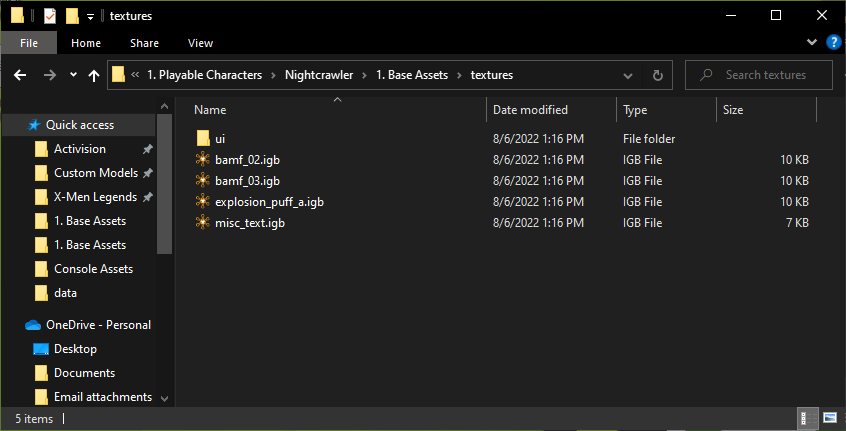
* Once again, do a GitHub commit.

A screenshot of a computer

Description automatically generated

### Moving the effects textures

* Next, you’ll want to see if the character’s effects textures are already in the Permanent package. Open the “3. General Data Files” folder in another window of File Explorer and go to the “permanent” folder. Compare the “2. Default Assets – PS2\textures” folder to the “textures” folder in your character’s “1. Base Assets” folder. In this case, Nightcrawler has 4 textures that aren’t in the permanent file.



A screenshot of a computer

Description automatically generated with medium confidence

* You’ll want to move the textures from your character’s “1. Base Assets\textures” folder to the “2. Default Assets – PS2\textures” folder for permanent. You can copy and paste for now so you can remember which textures they were.

A screenshot of a computer

Description automatically generated

* Track down these textures in the “XML2 GC Extracted Character Packages.rar”, “XML2 PSP Extracted Character Packages.rar”, and “XML2 Xbox Extracted Character Packages.rar” and add them to the “textures” folder for their respective consoles. Images for the GameCube version are included here, but do it for all applicable consoles. These textures don’t need to be added to the “2. Default Assets – PC” folder unless they’ve been modified or don’t exist in the PC version.

Graphical user interface, application

Description automatically generated

A screenshot of a computer

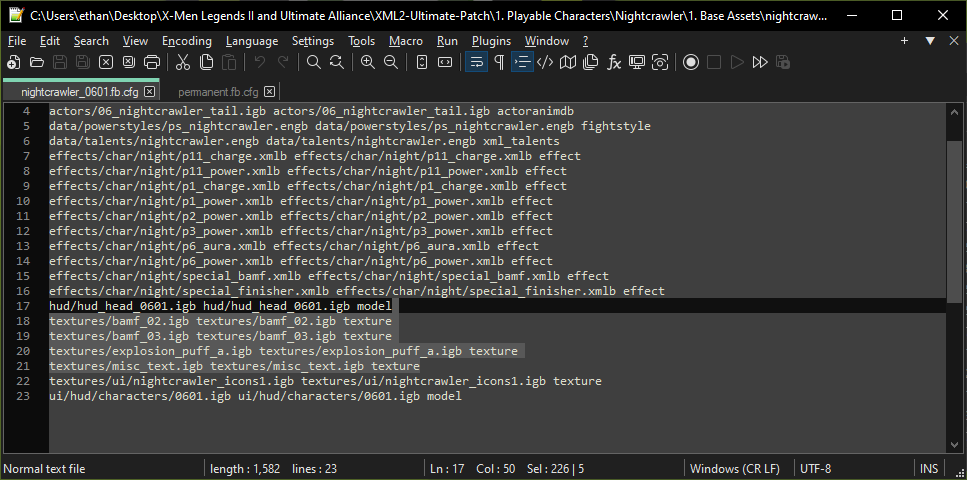
Description automatically generated

* Delete the textures from your character’s packages if they haven’t been deleted already.

A screenshot of a computer

Description automatically generated

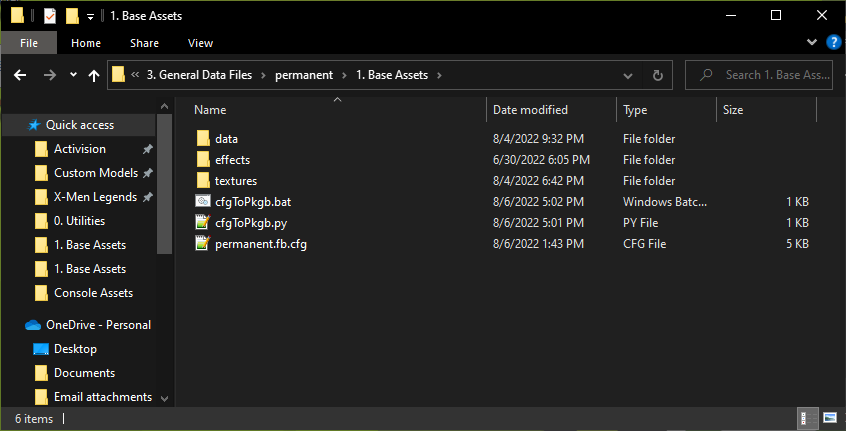
* Go to the “1. Base Assets” folder for permanent and open “permanent.fb.cfg”. Also open the cfg file for your character’s primary skin in their “1. Base Assets” folder. Cut and paste the lines relating to the textures from your character’s cfg file to permanent.fb.cfg.



Text

Description automatically generated

* You can alphabetize permanent.fb.cfg manually or use <https://alphabetizer.flap.tv/>. Also, make sure that these textures are not referenced in any of the cfg files for your character.
* Lastly, you’ll want to copy “cfgToPkgb.py” and “cfgToPkgb.bat” from “0. Utilities” to “3. General Data Files\permanent\1. Base Assets”. Double click the .bat file to run it. After that, you can delete the .bat and .py files. Make sure that there are no empty lines in your cfg file; if there are, the python file will error out.



* A new file will be created called “permanent.pkgb.json”. Move this from the “1. Base Assets” folder for permanent to the “2. Default Assets – PC\packages\generated\maps\package” folder for permanent.

A screenshot of a computer

Description automatically generated

* So in review: You character should not have any effect textures in their folder or in their cfg files. All the effects textures should be moved to the files for the permanent package. You should add these files for all the console versions. You should also update the cfg file and the decompiled .pkgb file for permanent to add the appropriate references. You can now make another commit to GitHub.

Graphical user interface, text

Description automatically generated

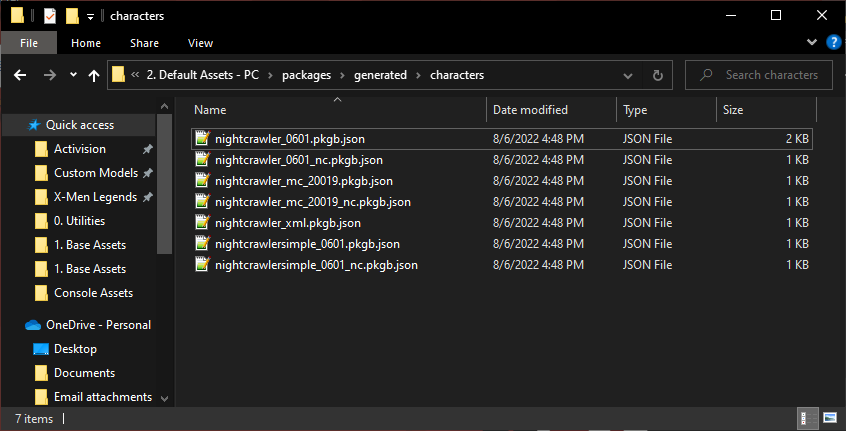
### Setting up data files

* Next up, it’s time to get the data files ready. The first thing to do is to decompile them. In the “0. Utilities” folder, you can find “(RF)DecompileJSON.bat”. If you drag and drop the data files onto this, they’ll get decompiled. Decompile the character’s powerstyle, talents, entities (if they have them), and effects. You can delete the original data files and keep the decompiled json data files. The powerstyle is shown as an example here.

A screenshot of a computer

Description automatically generated

* Next, go into the “2. Default Assets – PC” folder for your character and create folders for “packages/generated/characters”. There, add the same pkgb files (from your XML2 install) as the fb files you added earlier: first skin packages, name\_xml.pkgb, and any npc packages. Rename/renumber any that need it, and then decompile them using “(RF)DecompileJSON.bat”.

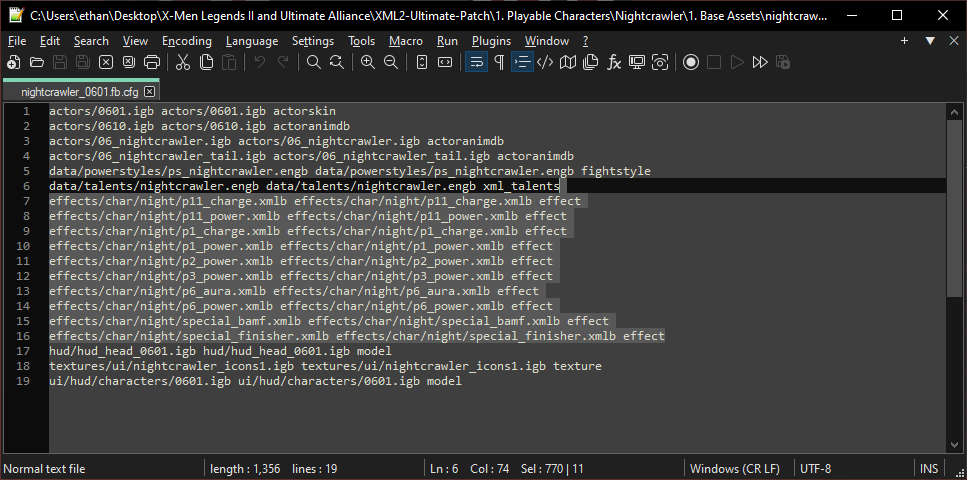


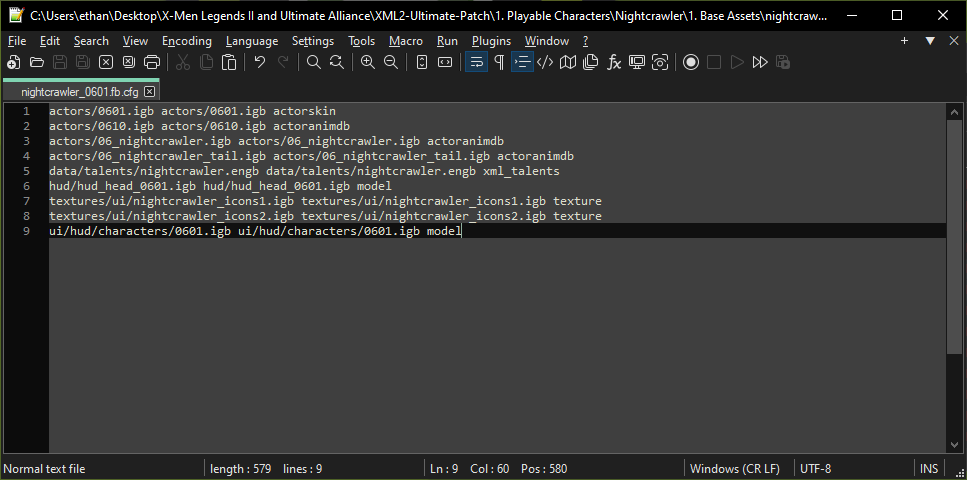
* Go ahead and do a GitHub commit showing that you decompiled the data files.

Graphical user interface, text

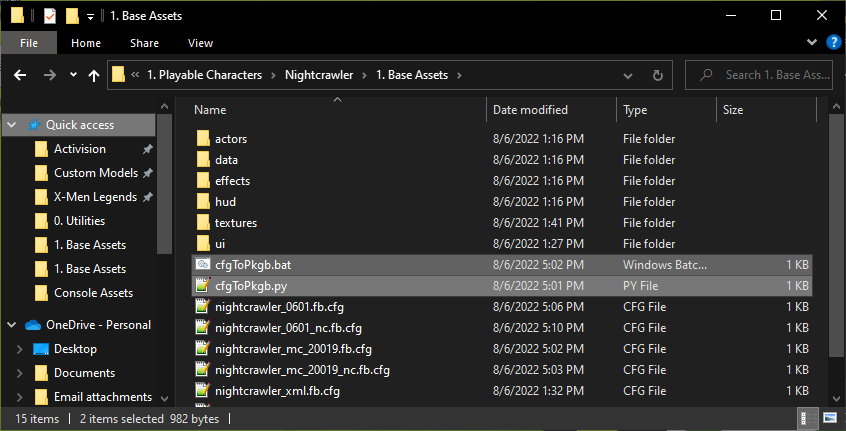
Description automatically generated

* There are a couple other things you’ll want to do with the data files before adding the X2UP content. First, you’ll want to clean up the cfg files.
  + Delete any effects referenced in the cfg files. Don’t delete the effects themselves; just delete their references in the cfg files. Generally, only the main package for the character will have effects by default, but you should still check the others. An NPC version may have effects if they’re ever in combat.
  + Delete the powerstyle referenced in the cfg file. Generally, the powerstyle is referenced in the main package for the character. NPCs may have powerstyles if they’re ever in combat. The powerstyle is also referenced in the (name)\_xml package.
  + Add the (name)\_icons2.igb files to the character’s primary and (name)\_XX01\_nc cfg file.
  + **Make sure there are no empty lines at the end of any cfg files!**
* Feel free to reference the cfg files of other characters to see how they look.

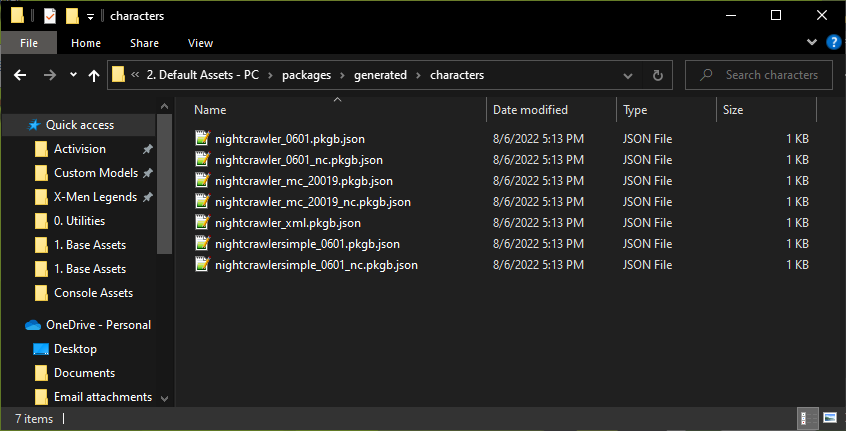




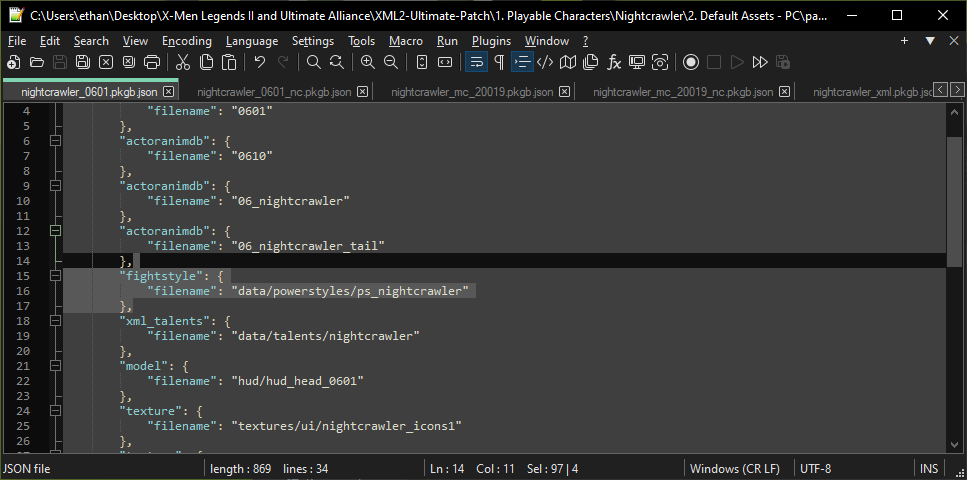
* Now, copy the “cfgToPkgb” files from the “0. Utilities” folder and double click the .bat file to run it. You can delete the .bat and .py file when you’re done.



* There will now be a bunch of .pkgb.json files. Move those from “1. Base Assets” to “2. Default Assets – PC\packages\generated\characters”.



* It’s generally fine to leave the .pkgb.json files exactly as they are (with the content being identical to the .cfg files), but you’ll want to add two things:
  + Add the reference back for the powerstyle. It’s technically not necessary, but it think it’s best to have it for the PC packages. You’ll only need to add this to the main package and to the (name)\_xml.pkgb.json package.
  + Add references to any deathsounds from any entities, if applicable. Check Storm’s packages for an example of this.



A screenshot of a computer

Description automatically generated

* Now you can make another GitHub commit.

Graphical user interface, text, application

Description automatically generated

### Importing X2UP Assets – Replacing existing files.

* Now it’s time to import the character’s X2UP assets! Make sure you have the latest files downloaded from the Google Drive before doing this, so you don’t miss anything.
* Start in the “1. Base Assets\actors” folder and copy in all skins for that character from the X2UP. Currently, Nightcrawler doesn’t have new mission briefing models, so I’ll just leave them alone for now.

A screenshot of a computer

Description automatically generated

Graphical user interface, text

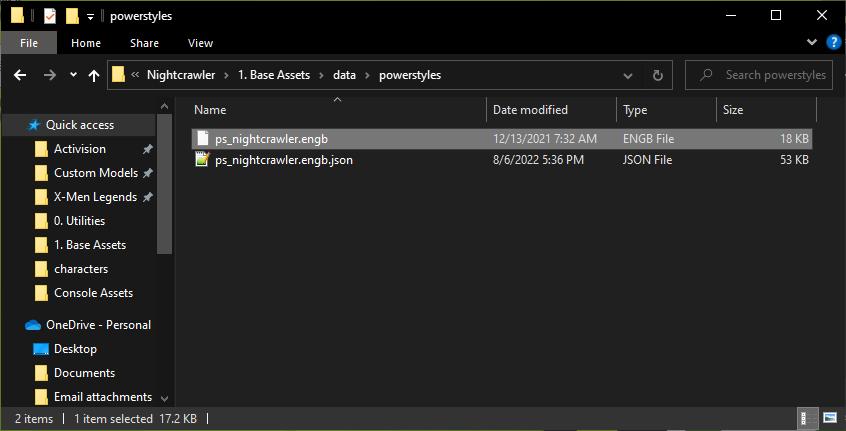
Description automatically generated

* Nightcrawler only has 8 skins in the X2UP, but we want to be able to create a full 9 packages for the console versions. So duplicate his primary skin and rename it to “0609.igb” (the missing skin number).

A screenshot of a computer

Description automatically generated

* Next, copy the data files into “1. Base Assets\data”. Decompile them with “(RF)DecompileJSON.bat”, and then delete the .engb/.xmlb file. This will overwrite the existing .engb.json/.xmlb.json file.



A screenshot of a computer

Description automatically generated

* Nightcrawler is unique in having his own moveset, which I’ll add in as well

A screenshot of a computer

Description automatically generated

* Next, do the same thing with effects (if they’re in the X2UP): copy in the X2UP effects files, decompile them, and then delete the .xmlb files. The .xmlb.json files will get updated.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

* Next, add the conversation portraits to the “1. Base Assets\hud” folder

A screenshot of a computer

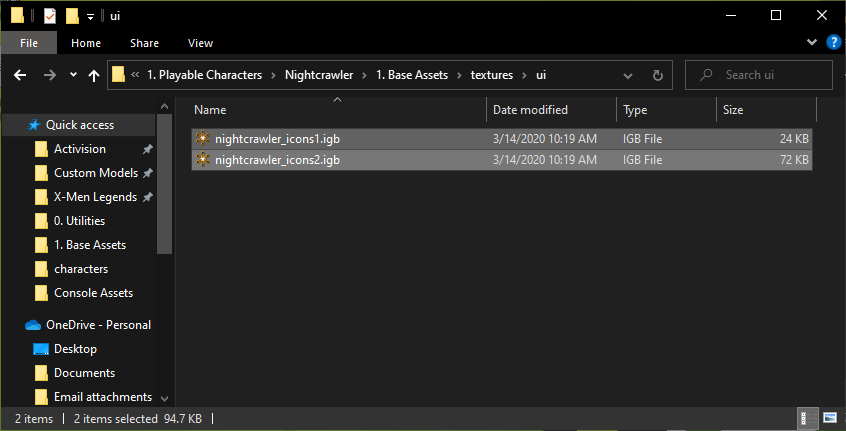
Description automatically generated with medium confidence

* Again, since he doesn’t have a full 9 skins, duplicate the first portrait and rename it to the missing skin number.

A screenshot of a computer

Description automatically generated with medium confidence

* Nightcrawler doesn’t have anything in the models folder, but if your character does, you could now add it there.
* Next, go to “1. Base Assets\textures\ui” and copy in the new icons.



* Next, it’s on to ui/hud/characters to add in the 3D heads

Graphical user interface

Description automatically generated

* Just like before, duplicate the first one to fill in any remaining skin numbers.

A screenshot of a computer

Description automatically generated with medium confidence

* You’ve now added all the assets that need to be replaced. You can now make a GitHub commit.

A screenshot of a computer

Description automatically generated

### Importing X2UP Assets – Adding new files

* Now that you’ve replaced existing assets with their X2UP versions, you can start adding in the new files. First will be the character select portrait. In “1. Base Assets\ui”, create a “models\characters” folder and add in the character select portrait from the X2UP.

A screenshot of a computer

Description automatically generated

* Next, go into “1. Base Assets\textures”, create a “comic” folder and add the character’s comic cover.

A screenshot of a computer

Description automatically generated

* Do the same for loading screens

A screenshot of a computer

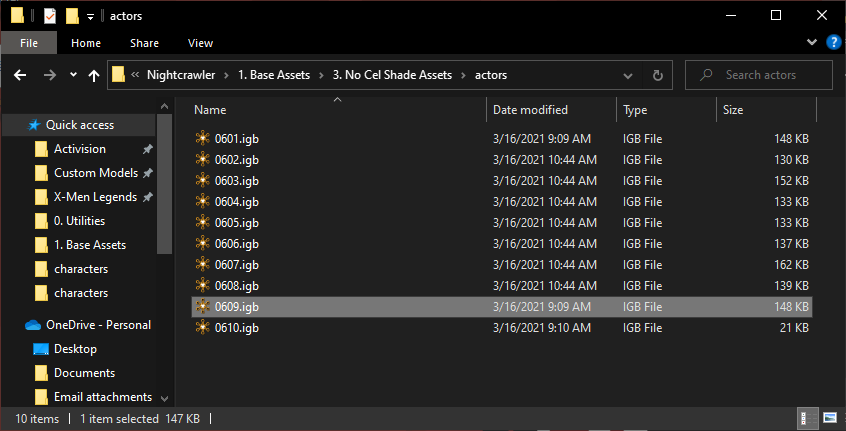
Description automatically generated with medium confidence

* Now, back out the “1. Base Assets” folder and within it, create “3. No Cel Shade Assets\actors.” Copy in the skins without cel shading. If the character has other files without cel shading, add those as well here.

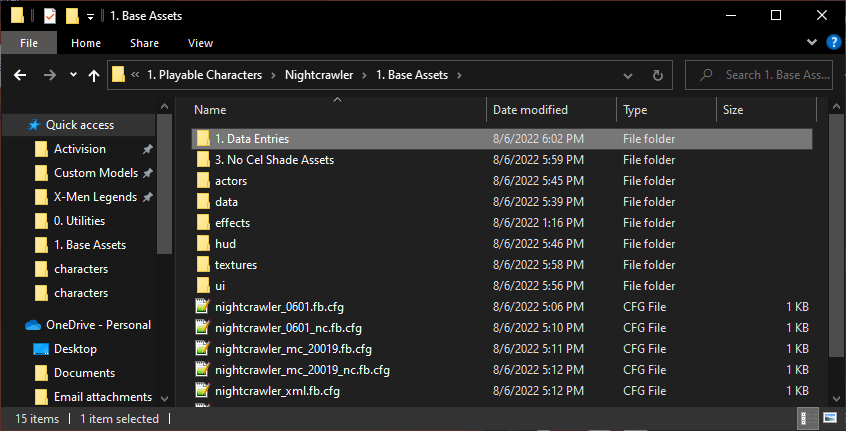
A screenshot of a computer

Description automatically generated with medium confidence

* Once again, create a duplicate skin to fill any empty slots.



* Back in the “1. Base Assets” folder, create a folder called “1. Data Entries.” This is where you’ll be adding the data file entries for the character.



* First, get the character’s herostat entry from the X2UP Google Drive’s “1. Herostats” folder. This is the *only* file here that will still be in XMLBCUI format, since these will be used in OpenHeroSelect.

A screenshot of a computer

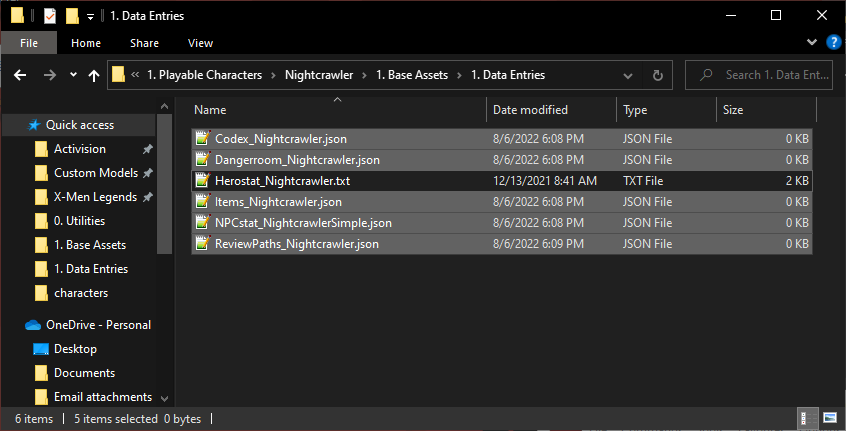
Description automatically generated

* After importing the file, rename it to “Herostat\_(name).txt”

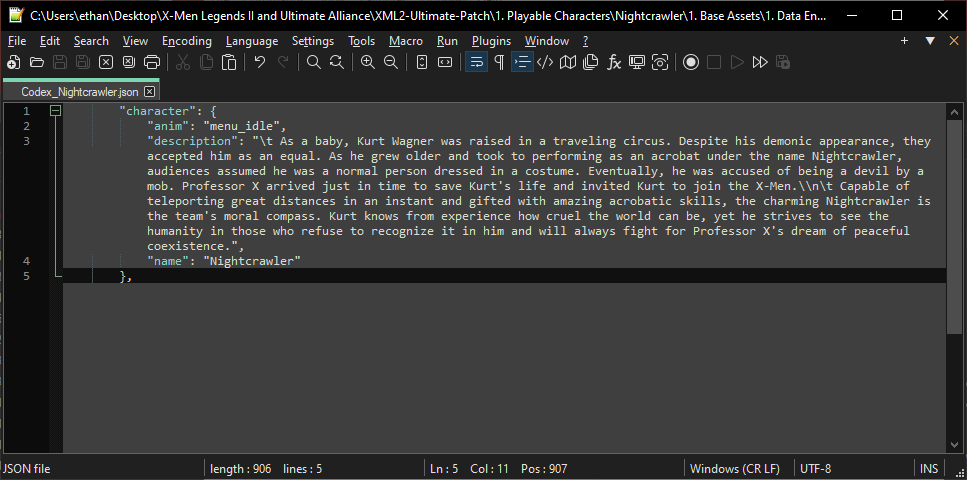
A screenshot of a computer

Description automatically generated

* Next, create empty text documents with .json extensions and the following names:
  + Codex\_(name).json
  + Dangerroom\_(name).json
  + Items\_(name).json
  + NPCstat\_(npc name).json
    - Create NPCstat files for all the character’s NPCs
  + ReviewPaths\_(name).json
* In this image, I forgot to add an NPCstat entry for Nightcrawler\_MC (NPCstat\_Nightcrawler\_mc.json), but it should be there.



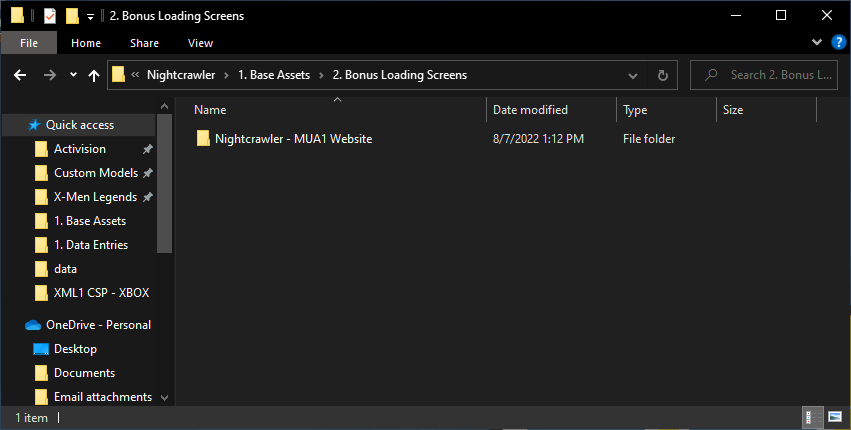
* Now, you can go through the X2UP and decompile the relevant data files and add the entries for the character. At time of writing, the X2UP does not have a Dangerroom or Items file, so you can use the default one for that. The codex is shown here as an example.
  + Also at time of writing, review\_paths is not updated with the latest loading screens for the characters. Just copy what’s there for now, and this can be updated later. For review\_paths, don’t add the concept art for the character.



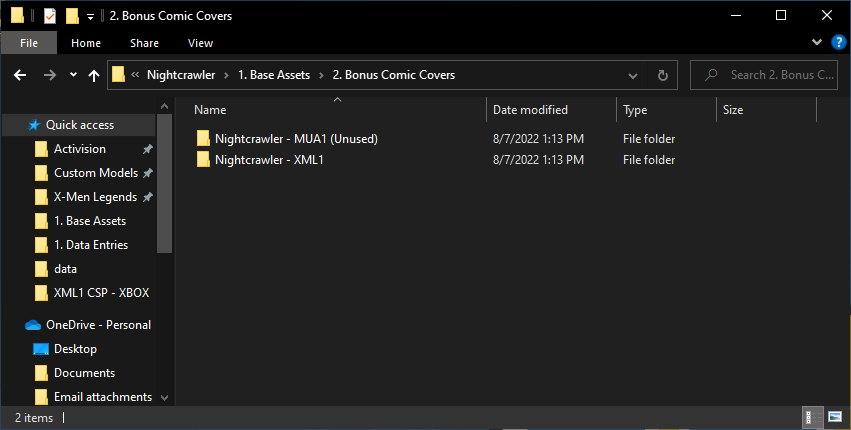
* Now that all the main assets are here, you can make another GitHub commit (I forgot to get a screenshot, but you know how at this point)

### Importing X2UP Assets – Bonus and console-specific files

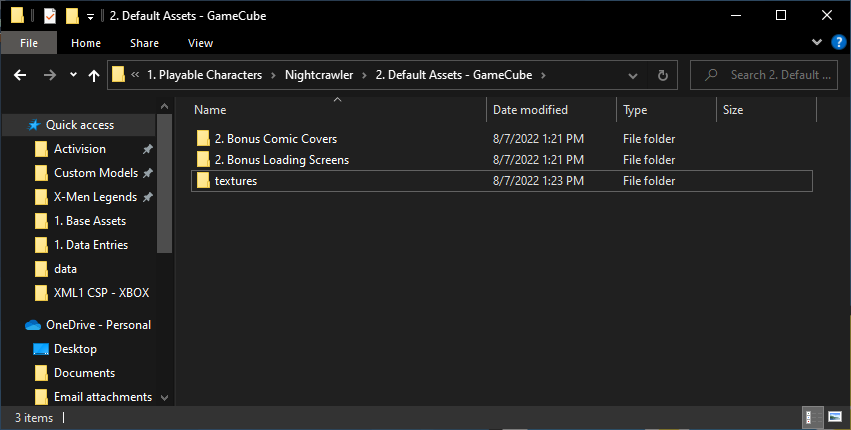
* The next thing to upload for the character is their bonus content from the X2UP. This includes extra assets and console-specific assets.
* We’ll start with the bonus content that’s applicable to the “1. Base Assets” folder. Generally, this will be bonus loading screens and comic covers. Nightcrawler has both.
* For loading screens, create a folder in the character’s “1. Base Assets” folder called “2. Bonus Loading Screens.” You can drag and drop the folders from the X2UP in here.



* Comic covers are the same, but they go in the “1. Base Assets\2. Bonus Comic Covers” folder.



* That’s everything that the “1. Base Assets” folder has, so now it’s time for console-specific assets. First, head over to the “2. Default Assets – GameCube” folder for your character. It should be empty currently. The GameCube has several files that are exclusive to it:
  + Visual assets with reduced textures (GameCube only):
    - Skins
    - 3D heads
    - Briefing models
    - Boltons/power models
    - Please get with Lags/Generic Unit to see if these assets have been made yet. I’ll be skipping over them for now.
  + Visual assets with reduced textures (Shared with PS2):
    - Comic covers (default and bonus)
    - Loading screens (default and bonus)
* Since I’m skipping over the skins, 3D heads, and other models for the GameCube, I’ll be focusing on the loading screens and comics. In the X2UP Google Drive, there’s a folder called “4. GameCube PS2 Common Assets.” Get the appropriate loading screens and comic covers (both the main and the bonus) and add them here.

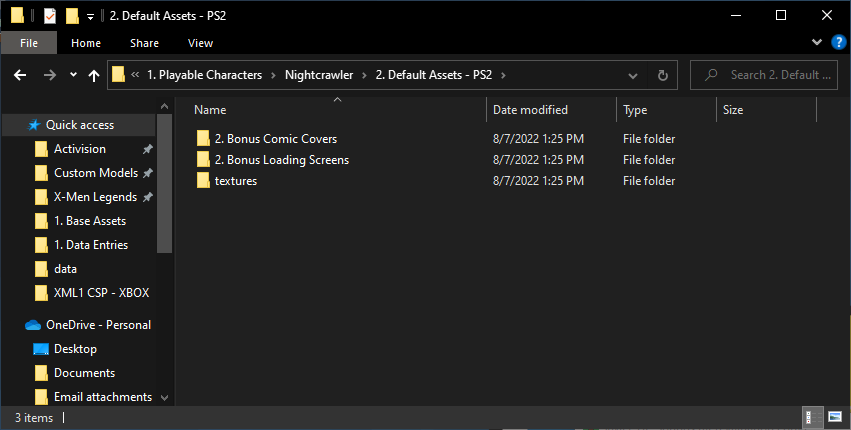


* You can now also add the sounds for the character, which you can get from “XML2 GC Sounds.rar”. The sounds aren’t being changed as of right now, but I think it’s beneficial to have them uploaded.

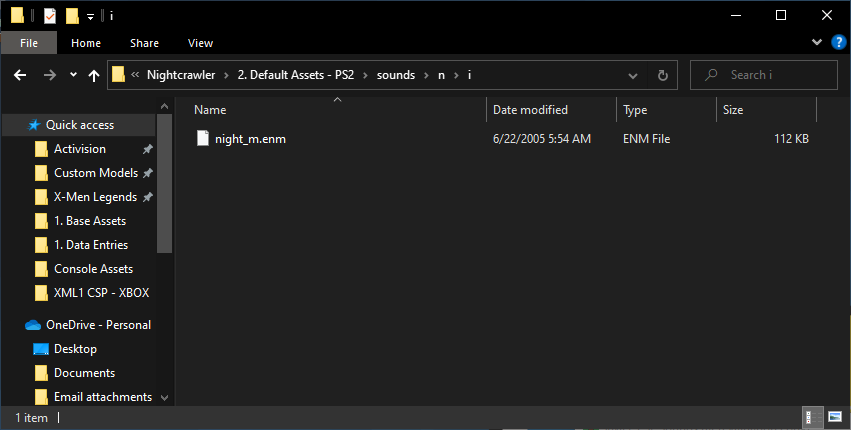
A screenshot of a computer

Description automatically generated

* Since the PS2 and GameCube use the same loading screens and comic covers, you can copy and paste the “2. Bonus Comic Covers”, “2. Bonus Loading Screens”, and “textures” folders from the GameCube folder to the PS2 folder.



* The PS2 version will have lower resolution briefing models than the Xbox and PC versions. If those have been made, add them here. Currently, Nightcrawler’s hasn’t been made, so I won’t add anything.
* You can also add the PS2 sounds at this point. ***Note that the PS2 version uses a different sound folder structure and doesn’t have the eng folder***.



* That’s it for the PS2 version, so you can move on the Xbox version. The only thing that the Xbox folder will have is the sounds, since everything else is the same as the PC.

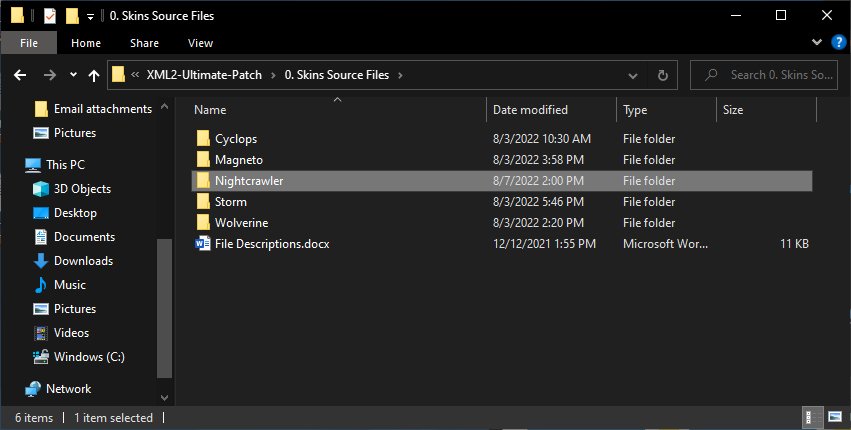
A screenshot of a computer

Description automatically generated

* You can also add the sounds to the PC folder.



* That’s it for the X2UP assets in the main folder, but there are also assets that can be found in other places. First are the skin source files. Each character has a folder in “0. Skins Source Files” for their skin source files. On the Google Drive, these can be found in the “1. Skins Source Files” folder.



* Next is the bonus skins. Each bonus skin is added to its own folder in the “2. Bonus Skins” folder in the main folder. On the google drive, these can be found in the “2. Bonus Skins” folder as well.

A screenshot of a computer

Description automatically generated

* Lastly, each character has a screenshot of their skins (and NPC versions if applicable). These were originally found in the “1. Screenshots” folder on the X2UP Google Drive and can be added to the “8. Screenshots\1. Base Assets” folder on the X2UP GitHub.

A screenshot of a video game

Description automatically generated

* Now that you’ve added all the X2UP content for the character, you can make another GitHub commit.

A screenshot of a computer

Description automatically generated

### Importing PSP Assets

* The character now has all their X2UP assets added, but the PSP-specific assets still need to be added. Since the PSP version doesn’t support custom models currently, only default assets can be used. For this, you’ll make heavy use of the “XML2 PSP Extracted Character Packages.rar” file, and maybe even the “MUA1 PSP Extracted Character Packages.rar” file.
* In the character’s “2. Default Assets – PSP” folder, make the following folders:
  + actors
  + hud
  + models (if applicable)
  + textures\loading
  + textures\comic
  + textures\ui
  + ui\hud\characters
  + ui\models\characters

A screenshot of a computer

Description automatically generated with medium confidence

* To the actors folder, add all of the character’s skins and animations (including those used by their NPC versions). When you first add them, the skin numbering won’t be correct.

A screenshot of a computer

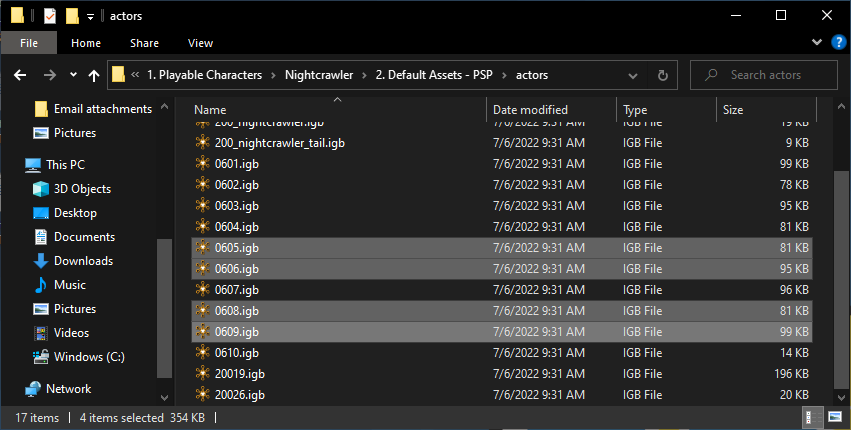
Description automatically generated with medium confidence

* Next, you’ll want to renumber the skins to the numbering style that the X2UP uses. For Nightcrawler, I’ll do:
  + 0601 -> 0602
  + 0602 -> 0603
  + 0603 -> 0601
  + 0610 -> 0610 (unchanged)
  + 0621 -> 0604
  + 0623 -> 0607

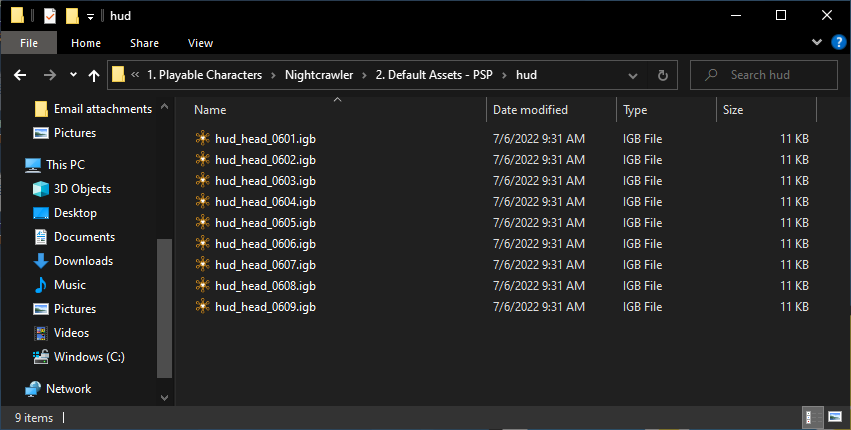
A screenshot of a computer

Description automatically generated with medium confidence

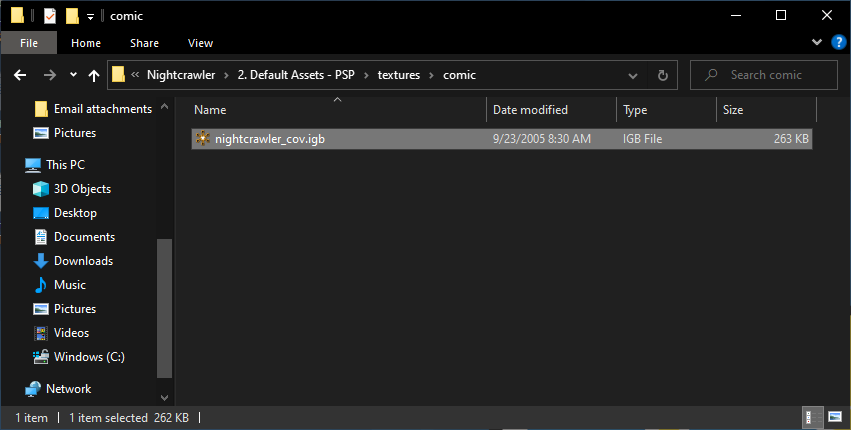
* You’ll notice that that’s only 5 skins, and Nightcrawler has 8 skins in the X2UP and 9 skin slots. If Nightcrawler had any unique assets in MUA1 PSP, I would add them in to fill additional skin slots, but he doesn’t, so I can’t do that. (For reference, the only characters with unique assets in MUA1 PSP are Colossus, Deadpool, Iceman, Iron Man, Psylocke, Storm, and Wolverine). For Nightcrawler, I’ll just duplicate the skins that are close and use those. For 0605 (Excalibur), I’ll duplicate 0604. For 0606 (2000s), I’ll duplicate 0603. For 0608 (Ultimate XML1), I’ll duplicate 0604. For 0609 (empty slot), I’ll duplicate 0601.



* Now that you’ve renumbered the skins, it’s a good idea to hex edit them so that the internal number matches the name of the file.
* Next, you can copy the character’s conversation portrait into the hud folder. Each character only has one portrait in XML2 PSP, but you can duplicate the portrait for the remaining skin numbers.



* Next, it’s on to the textures folder. First, go into the “comic” folder. The character’s comic cover can be found in “XML2 PSP Assets.wad”. Add it to the “comic” folder.



* You can do the same with textures\loading

A screenshot of a computer

Description automatically generated

* However, you’ll notice that the character has fewer loading screens in the PSP version than the X2UP. To fix this, just duplicate the loading screen.

A screenshot of a computer

Description automatically generated

* In textures\ui, add the character’s power icons from “XML2 PSP Extracted Character Packages.rar”

A screenshot of a computer

Description automatically generated

* Next, you can add the 3D heads from the extracted character packages to “ui\hud\characters”

A screenshot of a computer

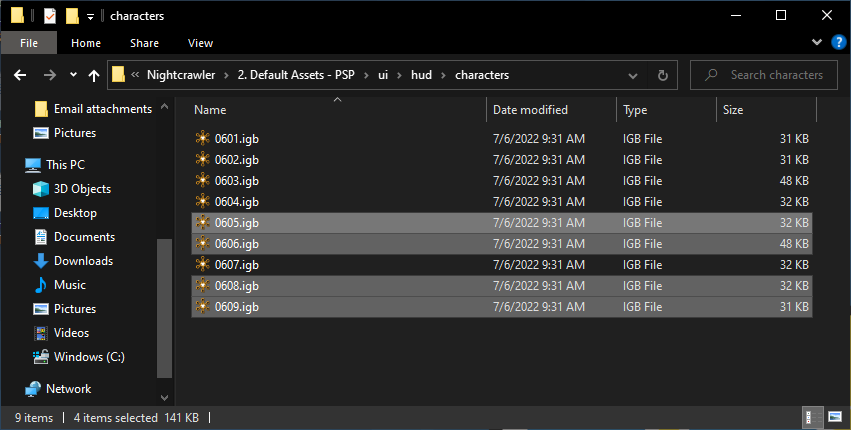
Description automatically generated

* Like with the skins, you can renumber them to the new numbering.

A screenshot of a computer

Description automatically generated

* Then, make duplicates as needed to fill the remaining slots, like you did with the skins. If you used a skin from MUA1, you can use the closest XML2 equivalent for the 3D head.



* Lastly, you can add the character select portrait from the extracted character packages

A screenshot of a computer

Description automatically generated

* The PSP version won’t feature any bonus visuals (loading screens, comic covers, bonus skins, etc), so the only thing left is to add the sounds from “XML2 PSP Sounds.rar”

A screenshot of a computer

Description automatically generated

* And now, all that’s left is to commit to GitHub.

A screenshot of a computer

Description automatically generated with medium confidence

### Setting Up the Update Script

* Now that all the assets for the character are imported, you need a way to update them to the appropriate places. Go into the “0. Templates” folder in the main folder and copy “update.bat” to your character’s main folder.

A screenshot of a computer

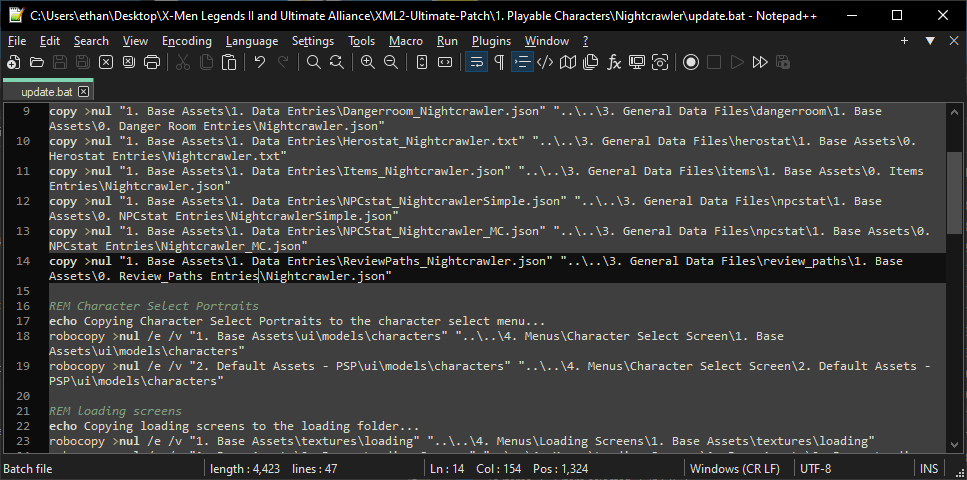
Description automatically generated

* Right click on update.bat and choose “Open with Notepad++” (if you have it, which hopefully you do). You’ll see all the code for the update script.

Text

Description automatically generated

* All the lines that start with “REM” are comments. The first section before any comments (starting on line 3), is where NPC assets are moved. Currently, this will be left blank. I want to wait until the maps are uploaded before doing this.
* First, go to the section for data entries, which starts with the comment on line 6. You’ll want to add all the character’s data entries to the list. Examples are set up already, and you can add more as needed. The structure of the code is as follows:
  + copy (the command to copy the file)
  + >nul (makes it so that the copy command does not display any messages in the command prompt)
  + “1. Base Assets\1. Data Entries\(file name)” (the relative path to the file. You replace the (file name) portion with whatever is applicable. Like I mentioned, some are already set up for you)
  + “..\..\3. General Data Files\(the data file’s folder)\1. Base Assets\(folder for the entries)\(new file name)” (the relative file path to the destination folder. The “..” means go to the parent folder. You’ll need to fill in the folder of the data file and the folder for its entries as needed. The new file name will just be the name of the character without the prefix specifying the file type. Like I mentioned, some are already set up for you)
* Here’s how it looks with Nightcrawler’s entries:



* There are a few other things that you may need to change, depending on your character
  + Removing anything pertaining to bonus loading screens (there’s a line for the “2. Bonus Loading Screens” folders for “1. Base Assets,” “2. Default Assets – GameCube”, and “2. Default Assets – PS2”. They’re the 2nd, 4th, and 6th lines after the comment).

A screenshot of a computer

Description automatically generated

* + Removing anything pertaining to bonus comic covers (there’s a line for the “2. Bonus Loading Screens” folders for “1. Base Assets,” “2. Default Assets – GameCube”, and “2. Default Assets – PS2”. They’re the 2nd, 4th, and 6th lines after the comment).

Text

Description automatically generated

* + Removing anything pertaining to consoles that the character does not appear on.
    - The Character Select Portrait section has a line for the PSP version.

Text

Description automatically generated

* + - The loading screens and comic covers sections have lines for GameCube, PS2, and PSP

Text

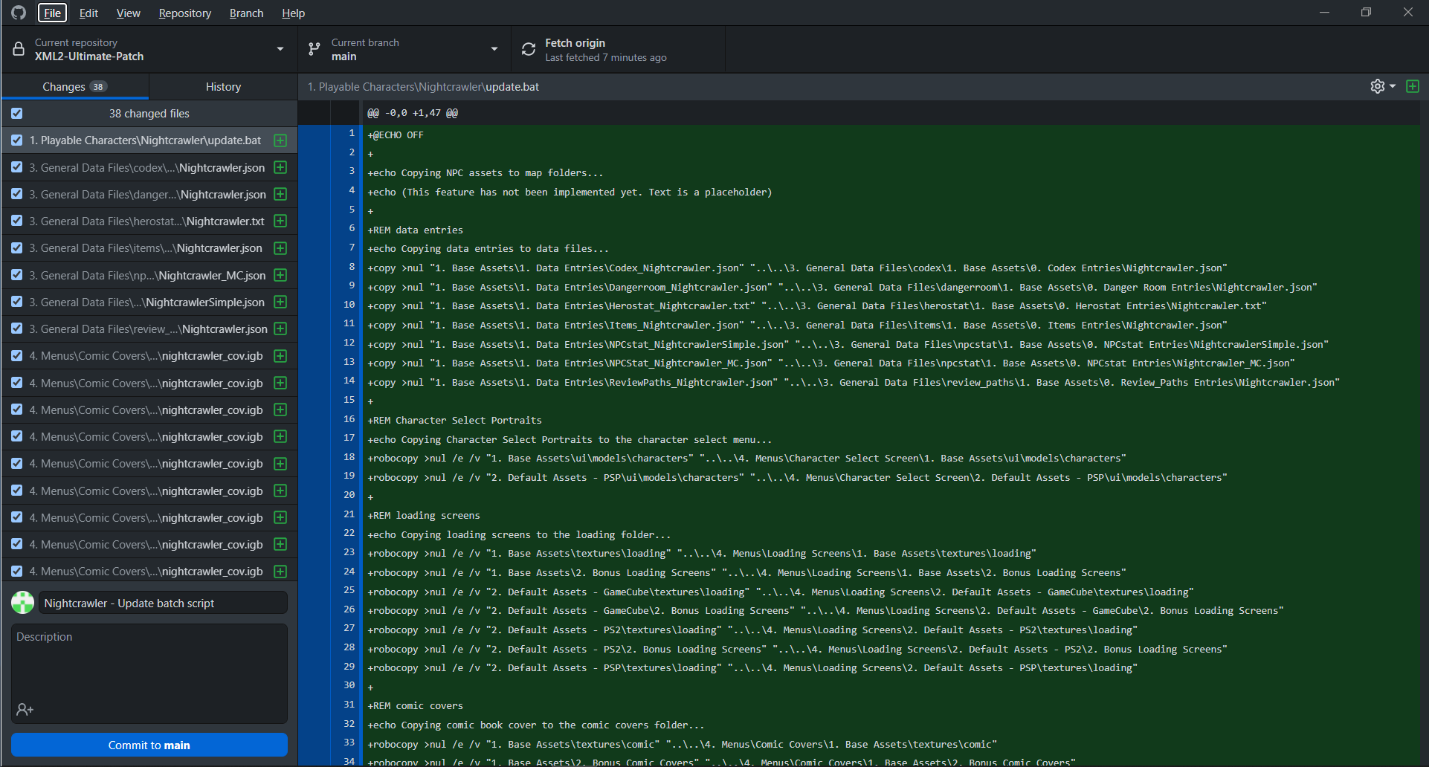
Description automatically generated

* + - The sounds section has lines for each console

Text

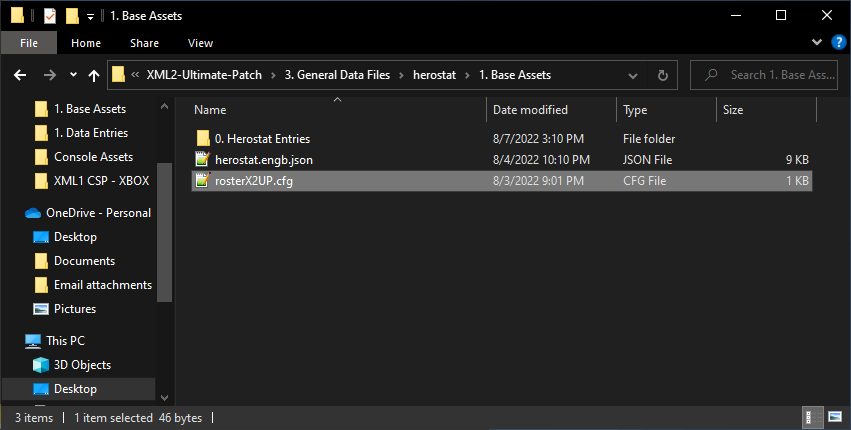
Description automatically generated with medium confidence

* Since Nightcrawler is playable on all consoles and has both bonus loading screens and bonus comic covers, I won’t be removing anything. That’s all for update.bat for now! First, double click it to run it. You can now make a GitHub commit for it.



### Adding the character to data files

* Now that your character has entries for various data files, they can be added to the lists for the various data files. I’ll do the herostat as an example. Go to “3. General Data Files\herostat\1. Base Assets” and open “rosterX2UP.cfg”.



* Add the file name of your character’s herostat entry without the file extension to this list if it’s applicable to do so (I’ll explain later who should be where).

A screenshot of a computer

Description automatically generated

* The “2. Default Assets – PC” folder has its own “rosterX2UP.cfg” as well as a “rosterAXE.cfg” and “rosterBHE.cfg”. Add your character’s name to these if applicable.

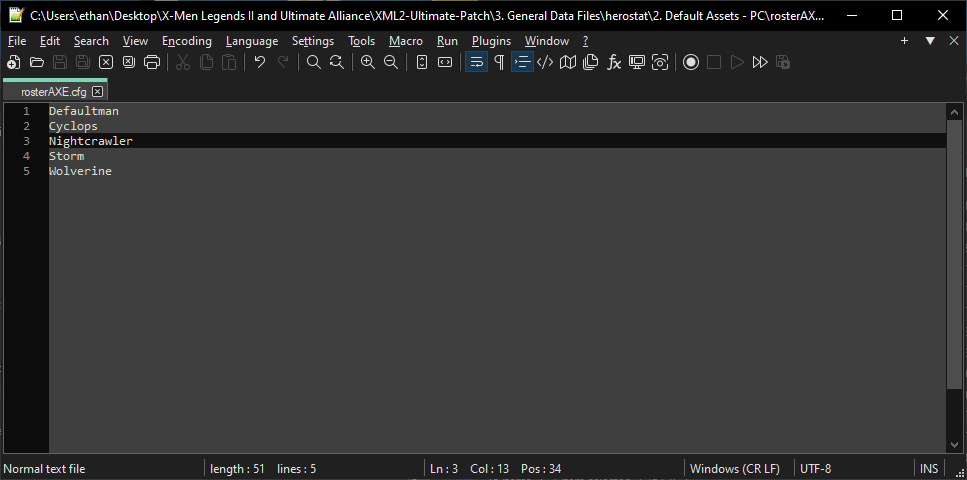
A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* The PSP also has its own version of “rosterX2UP.cfg”, which you can add your character to if it’s applicable.



* If the data file has an “update.bat” script, double click that to run it. At time of writing, only the npcstat has an update.bat script.

A screenshot of a computer

Description automatically generated with medium confidence

* Do the same for the other data files using the following lists:
  + Codex
    - Only one rosterX2UP.cfg for all versions, which will contain all 29 X2UP characters
    - PC has rosterAXE.cfg and rosterBHE.cfg, which will contain the names of the 21 characters from those roster mods. The Marvel Universe Edition will get its own later.
  + Dangerroom
    - RosterX2UP.cfg in the “1. Base Assets” folder will contain the 19 characters from every console (18 base characters and Psylocke).
    - PC has its own RosterX2UP.cfg which will be the same but with the addition of Pyro and Sabretooth.
    - PC has rosterAXE.cfg and rosterBHE.cfg, which will contain the names of the 21 characters from those roster mods. The Marvel Universe Edition will get its own later.
    - The organization of these files will change later to account for the base danger room courses and the grades.
  + Herostat
    - All files will contain Defaultman as the first entry.
    - RosterX2UP.cfg in the “1. Base Assets” folder will contain the 19 characters from every console (18 base characters and Psylocke).
    - PC has its own RosterX2UP.cfg which will be the same but with the addition of Pyro and Sabretooth.
    - PC has rosterAXE.cfg and rosterBHE.cfg, which will contain the names of the 21 characters from those roster mods. The Marvel Universe Edition will get its own later.
    - PSP has its own RosterX2UP.cfg, which will have the 19 characters from every console (18 base characters and Psylocke) plus the 4 psp-exclusive characters
  + Items
    - RosterX2UP.cfg in the “1. Base Assets” folder will contain the 19 characters from every console (18 base characters and Psylocke).
    - PC has its own RosterX2UP.cfg which will be the same but with the addition of Pyro and Sabretooth.
    - PC has rosterAXE.cfg and rosterBHE.cfg, which will contain the names of the 21 characters from those roster mods. The Marvel Universe Edition will get its own later.
    - The organization of these files will change later to account for the basic items
  + NPCstat
    - RosterX2UP is the only roster file for the NPCstat. It will contain the names of all the NPCs in it. Organization will probably change to account for default characters. PSP will probably have its own version for characters that appear in its comic book missions.
  + ReviewPaths
    - RosterX2UP.cfg in the “1. Base Assets” folder will contain the 19 characters from every console (18 base characters and Psylocke).
    - PC has its own RosterX2UP.cfg which will be the same but with the addition of Pyro and Sabretooth.
    - PC has rosterAXE.cfg and rosterBHE.cfg, which will contain the names of the 21 characters from those roster mods. The Marvel Universe Edition will get its own later.
    - The organization of these files will change later to account for the generic things
* Once you’ve added the character’s name to the necessary data file cfg files and run any update scripts, you can make a commit to GitHub

A screenshot of a computer

Description automatically generated

### Setting up the compile script

* Now that everything else is set up for the character, the compiler script is the only thing left!
* Go to the “0. Templates” folder and copy “compile.ini” and “characterCompile.bat” to your character’s folder.

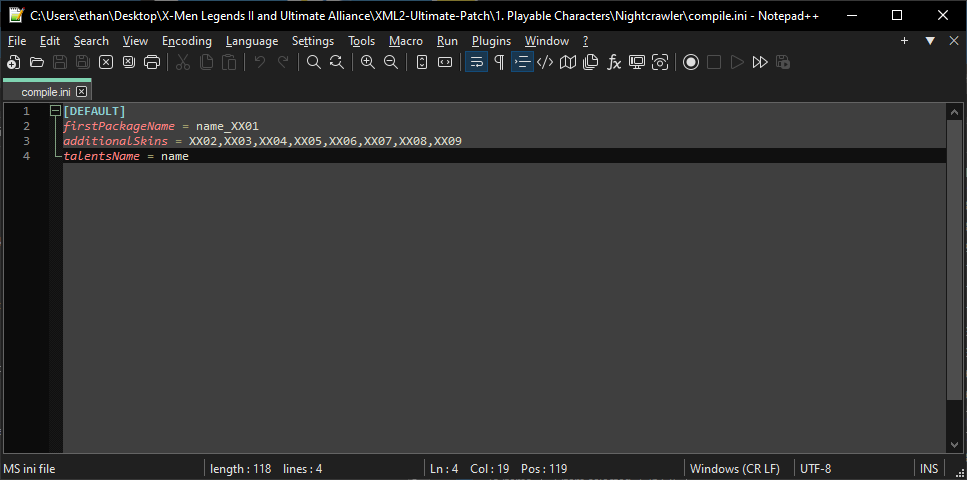


* Rename “characterCompile.bat” to just “compile.bat”

A screenshot of a computer

Description automatically generated with medium confidence

* First, open compile.ini. You’ll see a list of options. The first line is for the name of the character’s first package. The next line is a list of the other skin numbers the character uses. Lastly is the name of the character’s talent file.



* If your character uses normal skin numbering, you can just hit Ctrl+H and change “XX” to your character’s number. If they have different skin numbering (skipping a number or starting at a different number), make the edits as needed. Make sure that the “additionalSkins” list has no spaces in it. The additional skins list should have 8 skins (for a total of 9) regardless of how many skins your character actually has; all characters should have 9 packages.

A screenshot of a computer

Description automatically generated with medium confidence

A screenshot of a computer

Description automatically generated

* Also replace the “name” portion to the correct thing.

A screenshot of a computer

Description automatically generated

* Save the file. Then, right click “compile.bat” and choose “Open with Notepad++”. You’ll see all of the character’s compiler code.

A screenshot of a computer

Description automatically generated

* There are a few things you’ll need to change here. First, in line 2, replace “NAME” with your character’s name.

A screenshot of a computer

Description automatically generated

* The other changes will happen in “Section 0 – User Input”. In lines 9 and 10, set up the names of your character’s packages.

A screenshot of a computer

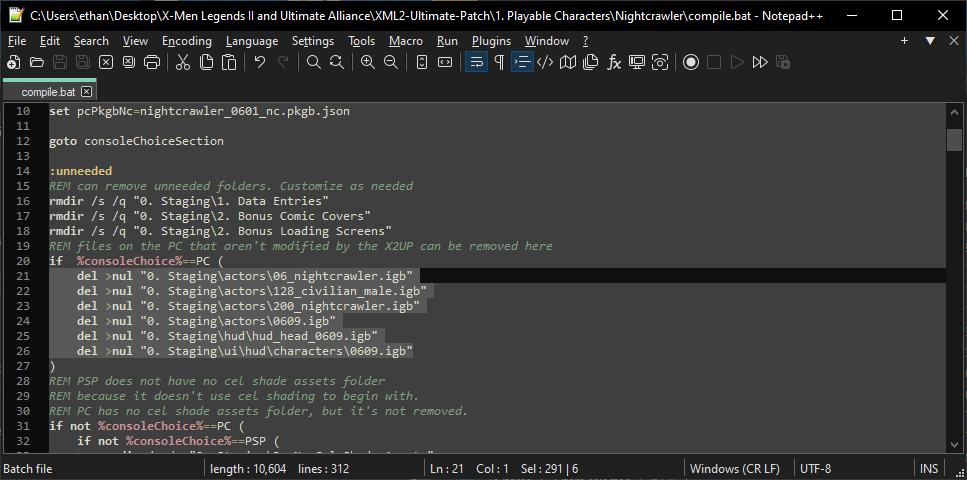
Description automatically generated

* In the “:unneeded” section (line 14), remove line 17 or 18 if your character does not have bonus loading screens or bonus comic covers. Nightcrawler has both, so I’ll leave those lines.

A screenshot of a computer

Description automatically generated

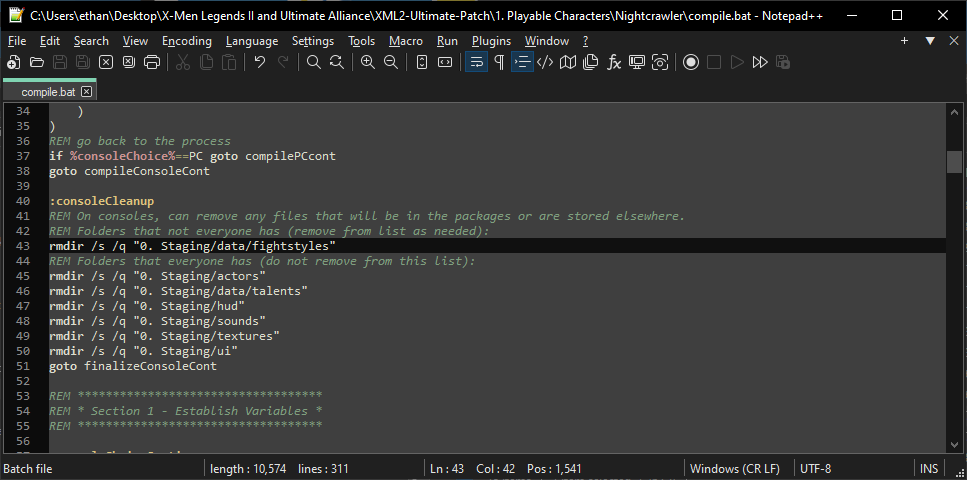
* If there are any files that the character has but aren’t modified by the PC version, you can add them to the list starting in line 21. Be sure to add “0. Staging\ before each file name and path. In Nightcrawler’s case, his character animations and NPC animations aren’t modified. His 0609 skin also isn’t used on PC, so everything related to it can be removed. This list has the “models” folder on it, but since Nightcrawler doesn’t have a “models” folder, it can be removed.



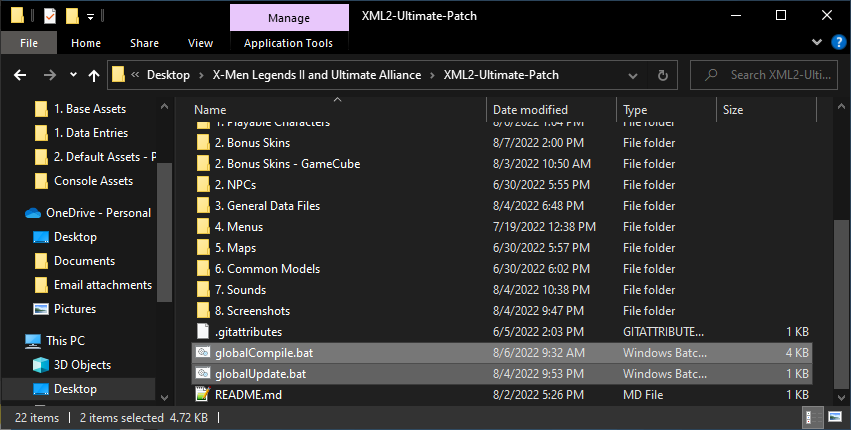
* Down a few lines, you’ll find a section titled “:consoleCleanup”. There, you can delete the folders of any files that get added to the console files. The ones that generally vary by character are the “data\entities” folder and the “models” folder. Nightcrawler doesn’t have entities or models, so these folders don’t need to be removed and thus don’t need to be on this list. Nightcrawler is unique in having a custom moveset, though, so I’ll add that to be removed from this list.

A screenshot of a computer

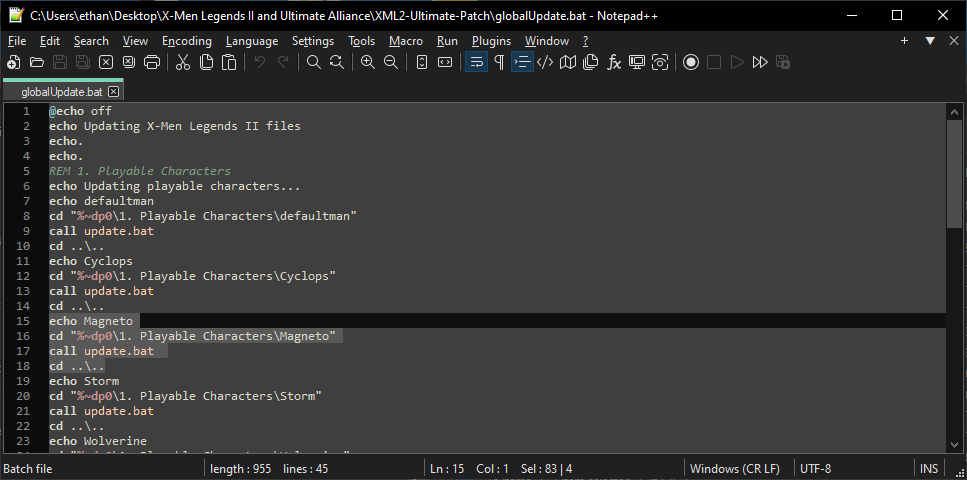
Description automatically generated



* Everything else in the compiler can be left alone. You can now save it and close it.
* There’s one last step now, which is to add the character to the global scripts. Back out to the main folder, where you’ll find “globalCompile.bat” and “globalUpdate.bat”.



* Right click>edit with Notepad++ for globalUpdate.bat first. Every character has 4 lines associated with them in this folder:
  + echo (name of character) (show the character’s name in command prompt. Edit this to show the character’s name)
  + cd "%~dp0\1. Playable Characters\(character’s folder)" (change the folder to the character’s folder. Edit this to be the character’s folder name)
  + call update.bat (run the character’s update batch file)
  + cd ..\.. (move back out to the main folder. “cd” means “change directory”, and “..” is the parent folder of the current folder.
* For the sake of simplicity, you can just copy these 4 lines from another character’s entry and paste and edit them. Keep it in alphabetical order please.



A screenshot of a computer

Description automatically generated

* Save globalUpdate.bat and then right click>edit with Notepad++ globalCompile.bat

Text

Description automatically generated

* Scroll down to section 2, where you’ll see a list of character compilers. The coding is almost identical to the update batch file. The only difference is the “call” line:
  + call compile.bat %consoleChoice% %skinPackChoice% %celChoice%
* It calls the “compile.bat” script for the character instead of the “update.bat” script and passes in the values of 3 variables: the console, the skin pack (something I set up for my personal uses and isn’t enabled normally since it’s not part of the X2UP), and the choice of cel shading.
* Like before, you can copy and paste it from another character and modify it.

Text

Description automatically generated

A screenshot of a computer

Description automatically generated

* You can now save this file, and then make a commit to GitHub!

A screenshot of a computer

Description automatically generated

* And that’s everything for importing a character!