

Ys VIII Switch Modding Archive Notes — Full Start-to-Finish Guide

This note documents the full working process we used to port Ys VIII PC `.it3` mods to Nintendo Switch by modifying the game's giant `data.dat` archive. This is meant for future reference so the whole project can be resumed without relearning everything from scratch.

Final Result / Core Discovery

The winning method was **not** cooking/rebuilding PC `.it3` files.

The breakthrough was:

```
PC `.it3` mod files can work on the Switch as-is.
```

The real obstacle was not the `.it3` format itself. The real obstacle was that the Switch version stores the character files inside one huge `data.dat` archive, so normal loose LayeredFS paths like `romfs/chr/pc/...` did not work.

The final working path was:

```
Extract Switch data.dat
→ replace target `.it3` files inside extracted `chr/pc/`
→ reimport modified files into data_MODDED.dat using QuickBMS reimport2
→ rename/copy modified archive as data.dat
→ place it at Atmosphere LayeredFS path:
    atmosphere/contents/01007F200B0C0000/romfs/data.dat
```

Huge Lesson Learned

Before trying to “fix” or “cook” PC files, always test the simplest possible direct-port method first:

```
Try PC `.it3` as-is over the matching Switch `.it3` filename.
```

We spent hours attempting hybrid fixes like:

```
Switch base + PC mesh + Switch textures
Switch base + PC mesh + PC material + texture aliases
```

Those failed visually. The untouched PC `.it3` worked.

So the new rule is:

Do not overengineer a mod until the direct PC-as-is port has been tested.

Core Modding Principle Going Forward

For all future expensive test batches:

1. Inspect first.
2. Read internal identifiers.
3. Confirm character / slot / family.
4. Confirm PC file and Switch file are matching pairs.
5. Compare structures.
6. Identify risky chunks.
7. Preserve proven wins.
8. Build only high-value tests.
9. Avoid random unknown-slot testing.
10. Never waste a 15GB repack/transfer on guesses.

Both user and assistant are learning collaboratively, but archive-based tests must be disciplined because one failed test can cost 1–2 hours.

Project Folder Layout

The workspace was eventually moved from `D:` to SSD on `C:` for faster copying/repacking.

Final SSD project root:

`C:\Switch Mods\Ys8_Mod`

Important files/folders:

```
C:\Switch Mods\Ys8_Mod\data.dat
C:\Switch Mods\Ys8_Mod\data_MODDED.dat
C:\Switch Mods\Ys8_Mod\COPY_TEMP
C:\Switch Mods\Ys8_Mod\data_faullfs
```

```
C:\Switch Mods\Ys8_Mod\data_unpacked
C:\Switch Mods\Ys8_Mod\quickbms_win
```

QuickBMS folder contents:

```
C:\Switch Mods\Ys8_Mod\quickbms_win\quickbms.exe
C:\Switch Mods\Ys8_Mod\quickbms_win\quickbms.txt
C:\Switch Mods\Ys8_Mod\quickbms_win\quickbms_4gb_files.exe
C:\Switch Mods\Ys8_Mod\quickbms_win\reimport.bat
C:\Switch Mods\Ys8_Mod\quickbms_win\reimport_4gb_files.bat
C:\Switch Mods\Ys8_Mod\quickbms_win\reimport2.bat
C:\Switch Mods\Ys8_Mod\quickbms_win\reimport2_4gb_files.bat
C:\Switch Mods\Ys8_Mod\quickbms_win\reimport3_localizations.bat
C:\Switch Mods\Ys8_Mod\quickbms_win\changelog.txt
C:\Switch Mods\Ys8_Mod\quickbms_win\fafullfs.bms
```

What Each Folder/File Is For

`data.dat`

This is the clean original Switch archive dump.

Treat this as sacred.

```
DO NOT directly modify data.dat.
```

It is the clean backup source used to create/reset `data_MODDED.dat` if something goes wrong.

`data_MODDED.dat`

This is the working modified archive.

QuickBMS reimport modifies this file directly.

Once the workflow is stable, there is no need to recreate it every time. You can keep reimporting new `.it3` replacements into the existing `data_MODDED.dat`.

Recreate it from clean `data.dat` only if:

- the game crashes on boot
- QuickBMS errors badly
- the archive seems corrupted

- bad experimental files were injected
- you want a clean baseline again

`COPY_TEMP`

Temporary folder used by `robocopy` when creating a fresh `data_MODDED.dat` from `data.dat`.

This folder can be deleted after the copy finishes.

`data_faullfs`

This is the extracted contents of `data.dat`.

This is the main working folder where `.it3` files are replaced before reimporting.

Important path inside it:

```
C:\Switch Mods\Ys8_Mod\data_faullfs\chr\pc
```

Character `.it3` files live here.

`data_unpacked`

This was from an earlier attempt using YsVIII-tools `undat.exe`.

That tool did not work for this Switch `data.dat`, so this folder is not important for the final method.

`quickbms_win`

Contains QuickBMS and the required `faullfs.bms` script.

For this archive, use:

```
quickbms_4gb_files.exe
```

Do not use normal `quickbms.exe` for the main archive because `data.dat` is larger than 4GB.

Tools Tried

YsVIII-tools — failed for this Switch `data.dat`

We tried the YsVIII-tools set:

```
dat.exe  
plt.exe  
undat.exe  
unplt.exe  
unxai.exe  
xai.exe  
xaiPatch.exe
```

Original old path:

```
D:\Switch Mods\New Projects\Ys8_Mod_1.1_IT3_version\YsVIII-Win64-1.0.3\YsVIII-Win64-1.0.3\undat.exe
```

Command tried:

```
"D:\Switch Mods\New Projects\Ys8_Mod_1.1_IT3_version\YsVIII-Win64-1.0.3\YsVIII-Win64-1.0.3\undat.exe" "D:\Switch Mods\New Projects\Ys8_Mod_1.1_IT3_version\data.dat" "D:\Switch Mods\New Projects\Ys8_Mod_1.1_IT3_version\data_unpacked"
```

Result:

```
Number of declared entries      : 0  
Number of extracted entries     : 0
```

Conclusion:

YsVIII-tools `undat.exe` does not handle this Switch `data.dat`.

We also tried `unxai.exe` :

```
"D:\Switch Mods\New Projects\Ys8_Mod_1.1_IT3_version\YsVIII-Win64-1.0.3\YsVIII-Win64-1.0.3\unxai.exe" "D:\Switch Mods\New Projects\Ys8_Mod_1.1_IT3_version\data.dat"
```

```
Projects\Ys8_Mod_1.1_IT3_version\data.dat" "D:\Switch Mods\New  
Projects\Ys8_Mod_1.1_IT3_version\data_unxai"
```

It also did not extract anything useful.

Conclusion:

This is not the simple DAT/XAST format handled by YsVIII-tools.

QuickBMS + fafullfs.bms — worked

Correct extractor/reimporter:

```
quickbms_4gb_files.exe + fafullfs.bms
```

This successfully extracted the huge Switch archive.

Extracted folders included:

```
actev  
bgm  
chr  
debug  
dlc  
efx  
flash  
inc  
map  
movie  
nxshader  
script  
system  
text  
visual  
voice
```

This confirmed the archive contained actual internal paths like:

```
chr\pc\c002_p.it3  
chr\pc\c005_p.it3
```

Important Failed LayeredFS Paths

Before the archive method worked, these loose-file paths were tested and did not work:

```
01007F200B0C0000\romfs\chr\pc
01007F200B0C0800\romfs\chr\pc
01007F200B0C0000\romfs\data\chr\pc
01007F200B0C0800\romfs\data\chr\pc
01007F200B0C0000\romfs\data.dat\chr\pc
01007F200B0C0800\romfs\data.dat\chr\pc
```

Why they failed:

```
The game does not open `romfs:/chr/pc/c002_p.it3` directly.
The game opens `romfs:/data.dat`, then reads `chr/pc/c002_p.it3` internally from
the archive.
```

Atmosphere LayeredFS can override files that the game asks the filesystem for, but it cannot magically override entries inside a custom archive.

So this does not work:

```
romfs\data.dat\chr\pc\...
```

because `data.dat` is a file, not a folder.

Final working LayeredFS path:

```
atmosphere/contents/01007F200B0C0000/romfs/data.dat
```

How To Extract `data.dat`

Open Command Prompt.

Go to project folder:

```
cd /d "C:\Switch Mods\Ys8_Mod"
```

Extract the clean archive:

```
quickbms_win\quickbms_4gb_files.exe quickbms_win\fafullfs.bms data.dat  
data_fafullfs
```

Only do this once unless you delete the extracted folder or need to start over.

After extraction, character files are located at:

```
C:\Switch Mods\Ys8_Mod\data_fafullfs\chr\pc
```

Character Slot Notes

Confirmed/used mapping:

```
c000 = Adol  
c001 = Sahad  
c002 = Laxia  
c003 = Hummel  
c004 = Ricotta  
c005 = Dana
```

Important files encountered:

```
c002_p.it3 = Laxia main/default visible slot  
c002b.it3  = Laxia alternate/variant  
c005_p.it3 = Dana main/default visible slot  
c005b.it3  = Dana alternate/variant  
c007.it3   = Dana-family / special Dana-related form  
c008.it3   = Dana-family / special Dana-related form
```

We confirmed `c002_p.it3` and `c005_p.it3` were visible/default slots because replacing them changed Laxia and Dana in-game.

C007 and C008 were not random, but they should have been identified earlier. Internal strings suggested Dana-family/special forms because they included identifiers like:

```
c005_emb_mesh  
Null_dana_light
```

```
007_skin / 007_face  
008_skin / 008_face
```

Future rule:

```
Always inspect internal identifiers before including a file in a test batch.
```

How To Replace `.it3` Files

Put the PC mod `.it3` files directly into:

```
C:\Switch Mods\Ys8_Mod\data_fafullfs\chr\pc
```

Overwrite matching names.

Example:

```
PC mod c002_p.it3 → data_fafullfs\chr\pc\c002_p.it3  
PC mod c005_p.it3 → data_fafullfs\chr\pc\c005_p.it3
```

Important:

```
Use the PC .it3 as-is first.  
Do not cook, convert, or hybrid patch unless direct-as-is fails.
```

Also:

```
Do not keep backup folders inside data_fafullfs\chr\pc\.
```

Keep backups outside, for example:

```
C:\Switch Mods\Ys8_Mod\BACKUPS
```

QuickBMS reimport usually ignores extra files/folders not present in the original archive table, but keeping the workspace clean avoids confusion.

Creating a Fresh data_MODDED.dat

This is only needed when starting fresh or recovering from corruption.

Go to project folder:

```
cd /d "C:\Switch Mods\Ys8_Mod"
```

Delete old modified archive:

```
del "data_MODDED.dat"
```

Clear/create temp folder:

```
rmdir /s /q "COPY_TEMP"  
mkdir "COPY_TEMP"
```

Use robocopy to copy the giant archive:

```
robocopy "." "COPY_TEMP" "data.dat" /J /ETA /R:1 /W:1
```

Rename copied file:

```
ren "COPY_TEMP\data.dat" "data_MODDED.dat"
```

Move it back to project root:

```
move "COPY_TEMP\data_MODDED.dat" ".\data_MODDED.dat"
```

Remove temp folder:

```
rmdir "COPY_TEMP"
```

Confirm sizes:

```
dir data.dat
dir data_MODDED.dat
```

Before reimport, `data.dat` and `data_MODDED.dat` should be the same size.

Important Note About Copying

Normal Windows `copy` can appear frozen with no progress when copying a 15GB file. It may show the destination file instantly in Explorer with the same size even while the copy is not truly finished.

Use `robocopy` because it gives more reliable behavior.

Normal Fast Repack Workflow

Once `data_MODDED.dat` already exists and is healthy, do not recreate it every time.

The fast loop is:

```
replace/add `.it3` files in data_fafullfs\chr\pc
→ QuickBMS reimport2 into existing data_MODDED.dat
→ copy data_MODDED.dat as romfs\data.dat
→ upload to Switch
→ test
```

Command:

```
cd /d "C:\Switch Mods\Ys8_Mod"
quickbms_win\quickbms_4gb_files.exe -w -r -r quickbms_win\fafullfs.bms
data_MODDED.dat data_fafullfs
```

Meaning:

```
-w      = write mode
-r      = reimport
-r -r   = reimport2 mode, needed when replacement files are larger than originals
```

Use `-r -r` because modded `.it3` files can be bigger than the originals.

If QuickBMS asks about a larger file:

DO NOT type force.

`force` can corrupt the archive.

Good result looks like:

```
- 28635 files reimported in ____ seconds
```

Example successful output:

```
- 28635 files reimported in 735 seconds
coverage file 0      0%   2042192   15746164543 . offset 00000003a845f110
C:\Switch Mods\Ys8_Mod>
```

The `coverage file 0 0%` line looked weird but did not mean failure. The game booted afterward.

Staging the Final Atmosphere File

After QuickBMS finishes, stage the modified archive as `data.dat` for LayeredFS.

Create the staging folder:

```
mkdir "01007F200B0C0000\romfs"
```

Copy modified archive to final staging path:

```
copy /Y "data_MODDED.dat" "01007F200B0C0000\romfs\data.dat"
```

Final local staging path:

```
C:\Switch Mods\Ys8_Mod\01007F200B0C0000\romfs\data.dat
```

Final Switch path:

```
sd:/atmosphere/contents/01007F200B0C0000/romfs/data.dat
```

Uploading to Switch

FAT32 Problem

The modified `data.dat` is around 15GB.

If the Switch SD card is FAT32, normal file transfer usually fails for files over 4GB.

DBI MTP responder / Windows Explorer may drop or fail when copying arbitrary 15GB files.

Working Transfer Method

The method that worked was:

```
DBI FTP server on Switch + FileZilla on PC
```

Upload local:

```
C:\Switch Mods\Ys8_Mod\01007F200B0C0000\romfs\data.dat
```

to remote:

```
/atmosphere/contents/01007F200B0C0000/romfs/data.dat
```

Overwrite the previous test archive.

After upload:

```
Reboot Switch fully before testing.
```

Transfer speed over FTP was slow and could take around an hour, but it worked.

Test Result Meanings

```
Game boots + character mod appears:  
SUCCESS. Archive replacement and `.it3` worked.
```

```
Game boots + no visible change:
```

Wrong character slot, wrong costume slot, or the changed file is not the active one.

Game crashes when loading a character:

The archive injection worked, but the `.it3` file is incompatible or bad.

Game crashes on boot:

The modified `data.dat` or reimport process likely broke something.

Restore from clean data.dat and retry.

What Failed During Testing

Hybrid Test 02

A test was built using:

```
Switch base/container
+ PC visual mesh chunks
+ Switch materials/textures
```

It booted and proved:

```
- data.dat override worked
- QuickBMS reimport worked
- game read the modified c002_p.it3
- PC mesh chunks could load
```

But textures were imperfect.

6-Slot Hybrid Fix Batch

A later batch tried:

```
Switch base
+ PC mesh chunks
+ PC material chunks
+ Switch/aliased textures
```

This failed visually. Laxia became much worse with rainbow/noisy textures, and Dana was not good.

Conclusion:

Do not manually mix PC materials with Switch textures unless absolutely necessary.

The untouched PC `.it3` files worked better because PC mesh/material/texture relationships stayed internally consistent.

Wrong Assumption

We assumed PC texture chunks like:

ITP / IHDR / IDAT

would not work on Switch because Switch files used:

DDS / YKCOMP_V1

That difference was real, but it was not a hard blocker. The Switch build was tolerant enough to load the PC `.it3` files as-is.

Final Working Formula

1. Dump Switch data.dat.
2. Extract data.dat using QuickBMS 4GB + fafullfs.bms.
3. Put PC `.it3` mod files as-is into data_fafullfs\chr\pc\.
4. Match filenames exactly where possible.
5. Reimport using QuickBMS reimport2 into data_MODDED.dat.
6. Copy data_MODDED.dat as romfs\data.dat.
7. Upload to Switch via DBI FTP + FileZilla.
8. Reboot and test.

Most important short version:

Do not cook the `.it3` unless direct-as-is fails.

DLC Costume Mod Notes

The newer amorrow28 Ys VIII costume/nude mod uses equippable DLC costumes on PC.

His post explained:

For the DLC, if using the NISA Steam release, replace ys8.exe or patch it.
The patch enables the DLC costume system.

On Switch, there is no directly patchable `ys8.exe`. The Switch equivalent would likely require patching the game executable / main NSO / exefs, which is a much harder reverse-engineering project.

Switch DLC folder structure was different from the PC mod DLC structure.

Example Switch dumped DLC structure:

```
dlc\ys08jpdlc00n0044\data\en
dlc\ys08jpdlc00n0044\data\fr
dlc\ys08jpdlc00n0044\data\ja
dlc\ys08jpdlc00n0044\data\dlc.xai
dlc\ys08jpdlc00n0044\data\init.txt
dlc\ys08jpdlc00n0044\data\asset_original_data
```

Inside `asset_original_data`:

```
asset_original_data\chr
asset_original_data\script
asset_original_data\text
asset_original_data\dlc.xas
asset_original_data\dlc_xai_create.bat
```

PC mod DLC structure looked more like:

```
dlc\ys08jpdlc00n0037\en
dlc\ys08jpdlc00n0037\fr
dlc\ys08jpdlc00n0037\ja
dlc\ys08jpdlc00n0037\dlc.xai
```

Attempted adapted structure:

```
dlc\ys08jpdlc00n0037\data\en  
dlc\ys08jpdlc00n0037\data\fr  
dlc\ys08jpdlc00n0037\data\ja  
dlc\ys08jpdlc00n0037\data\dlc.xai
```

The game booted, but the costumes did not appear. The DLC menu said everything was already claimed.

Likely conclusion:

```
The files may load, but the DLC costume IDs/entries are not registered/unlocked  
on Switch.  
The PC `.exe` patch probably enables those entries internally.
```

Recommended DLC Workaround

Do not fight DLC unlocks for now.

Instead:

```
Extract the newer DLC mod `.it3` files  
rename/use them as direct replacements for existing default slots  
inject them into data.dat via the proven method
```

For example:

```
newer Laxia nude `.it3` → replace c002_p.it3  
newer Dana nude `.it3` → replace c005_p.it3
```

This avoids the DLC unlock system completely.

Recommended Future Workflow for More Mods

When collecting more `.it3` mods:

1. Prefer exact filename matches first.
2. If PC mod has c002_p.it3, replace Switch c002_p.it3.
3. If PC mod has c005_p.it3, replace Switch c005_p.it3.

4. Do not rename random files unless internals confirm the character/slot.
5. If filenames do not match, inspect internal identifiers before using them.

Do not manually test one by one if avoidable. Instead:

- inspect filenames/internal strings
- batch only high-confidence matches
- repack once
- test multiple confirmed slots together

Quick Command Reference

Go to folder

```
cd /d "C:\Switch Mods\Ys8_Mod"
```

Extract archive once

```
quickbms_win\quickbms_4gb_files.exe quickbms_win\fafullfs.bms data.dat  
data_fafullfs
```

Create fresh data_MODDED.dat

```
del "data_MODDED.dat"  
rmdir /s /q "COPY_TEMP"  
mkdir "COPY_TEMP"  
robocopy "." "COPY_TEMP" "data.dat" /J /ETA /R:1 /W:1  
ren "COPY_TEMP\data.dat" "data_MODDED.dat"  
move "COPY_TEMP\data_MODDED.dat" ".\data_MODDED.dat"  
rmdir "COPY_TEMP"
```

Normal reimport2 repack

```
quickbms_win\quickbms_4gb_files.exe -w -r -r quickbms_win\fafullfs.bms  
data_MODDED.dat data_fafullfs
```

Stage final data.dat

```
mkdir "01007F200B0C0000\romfs"  
copy /Y "data_MODDED.dat" "01007F200B0C0000\romfs\data.dat"
```

Final Switch path

```
sd:/atmosphere/contents/01007F200B0C0000/romfs/data.dat
```

Final Emotional Summary / Project Outcome

This project was painful because Ys VIII Switch wraps basically the entire game into one giant ~15GB `data.dat`. Every failed test meant:

```
replace files  
→ reimport/repack  
→ copy/transfer massive archive  
→ reboot/test
```

It could take around 1.5 hours per serious test including transfer.

But the final result was successful:

```
Ys VIII PC `.it3` mods can work on Switch as-is when injected into data.dat.
```

This was the major win.

The biggest technical lesson:

```
The archive was the problem, not necessarily the PC `.it3` format.
```

The biggest workflow lesson:

```
Always test the simplest direct port before trying to fix/cook/rebuild anything.
```

The partner rule:

We learn together, but we respect the cost of each test.
Inspect first. Confirm first. Then build.