Page contents

- Inventory the files before packing
- Choose the correct archive type
- Separate development, package, and preservation copies
- Troubleshoot the packaged copy
- Sources

Offline copy of Assets, BA2 archives and release testing .

Fallout 4 assets and BA2 release workflow

Documentation status: In progress. **Sources checked:** 2026-11-11. **Technical validation:** the packaging workflow has not been run against a local Fallout 4 build for this publication. This chapter covers PC mod assets and distribution, not copyrighted game-asset redistribution.

A Fallout 4 plugin stores records, but custom meshes, textures, materials, sounds, and compiled scripts are separate files. A working development profile can conceal a broken package because loose files elsewhere in `Data` satisfy paths that the release never included. The release test must therefore start from an explicit file manifest and end in a clean profile.

Inventory the files before packing

Create a staging directory containing only files you made or have permission to redistribute. Preserve the relative paths that the plugin refers to. A small example release might look like this:

```
ModDocsExample.esp
Meshes/ModDocsExample/example.nif
Materials/ModDocsExample/example.bgsm
```

```
Textures/ModDocsExample/example_d.dds
Scripts/ModDocsActivationProbe.pex
```

This is an illustrative file tree, not a claim that these example assets exist. The compiled `.pex` is a runtime file; keep the editable `.psc` in your source archive, but include only files needed by the installed mod in the runtime package. Check every path referenced by a record or script property against the staging tree. Record the source of each file and its license. Do not repack Bethesda's game archives or include extracted base-game files in a release without rights to do so.

Choose the correct archive type

Fallout 4 uses BA2 archives for mod assets. The publicly readable STEP Archive2 guide documents separate **DDS** texture and **General** asset modes. Its installation section is outdated, so use a current Creation Kit source for installation. A texture archive is commonly named `<Plugin> - Textures.ba2`; a general archive is commonly named `<Plugin> - Main.ba2`. Keep the archive base name aligned with the plugin name and verify that the intended game build loads both archives. The names shown here are a convention to test, not a substitute for a load test.

For a texture archive, stage only the texture tree, choose Archive2's DDS format, import the staged files, save the archive, then reopen it and verify paths and file counts. For meshes, materials, scripts, and other non-texture files, create a separate General archive. A Nexus Archive2 guide warns that sounds and precombined meshes may require uncompressed General packing, but that URL returned HTTP 403 in our link audit. Treat this as an unverified compatibility warning: identify such files in your release and test their actual behavior rather than applying one compression setting to every asset.

Do not assume an extracted DDS is bit-for-bit identical to its source. The ByroRedux archive reader documentation explains that a texture BA2 stores mip chunks and requires its reader to reconstruct a DDS header. Keep your original editable texture sources and verify the archive in game. An older Nexus technical note, which returned HTTP 403 in our link audit, reports compatibility differences between archive/tool builds; use a matching Creation Kit and game build for the release you intend to support. Do not change archive version bytes by hand as a routine compatibility fix.

Separate development, package, and preservation copies

Keep three distinct artifacts:

Artifact	Purpose	Contents
Editable source	Rebuild later	Plugin project notes, `.psc`, original art and export settings
Installable release	Player use	Plugin, required BA2 archives or allowed loose files, dependency list
Verification record	Reproduce the release	Game/editor versions, archive-tool version, test profile, file manifest, checksums, results

After creating BA2 files, remove or disable the staged loose files from the game's load path. Install only the release package in a clean profile. Confirm that the plugin loads, the custom item or scene appears, its model and material resolve, and any scripted behavior still runs. Repeat on a fresh save and a disposable existing save where relevant. If the packaged copy fails but the development profile works, compare the archive's internal paths with the record paths before changing scripts or load order.

For preservation, store the source project and final release together with a readable build note. A version number in a filename is useful but insufficient: include the actual Fallout 4 executable build and Creation Kit build used to create and test the archive. Bethesda's August 2026 update notes changed the supported BA2 archive count and Creation Kit archiving behavior, so earlier packaging assumptions need renewed tests after an update.

Troubleshoot the packaged copy

Symptom	First comparison	Follow-up
Plugin works but model is invisible	Record model path versus archive internal path	Check the mesh and material paths, then retest with staging files disabled

Symptom	First comparison	Follow-up
Texture is black, missing, or wrong	Texture record/material path and DDS archive mode	Reopen the archive, inspect the source DDS, and test with the matching tool build
Script stops running after packaging	Plugin attachment and `.pex` location	Confirm the compiled file is present, then use the Papyrus diagnostics chapter
Game rejects or ignores the archive	Game build, Archive2 build, archive type, and name	Rebuild with a compatible tool and test without other mods
Release behaves differently from development	Extra loose files or other enabled mods	Compare a clean install against the manifest, one dependency at a time

Do not infer success from a successful Archive2 save alone. The archive must be reopened, installed, and exercised in the game. This chapter does not yet document binary BA2 headers, mesh export settings, material authoring, animation pipelines, or a named asset built and tested end to end; those remain explicit coverage gaps.

Sources

- Fallout 4 update notes for archive and Creation Kit changes
- STEP Archive2 guide, publicly readable format and packing instructions; its installation section is historical.
- ByroRedux BA2 reader documentation, publicly readable implementation notes on General and DX10 texture archives.
- Nexus Archive2 packing procedure and BA2 technical note, preserved for provenance; both returned HTTP 403 in the 2026-11-11 link audit.