Converting idStudio Mods to Atlan

Atlan Mod Packager can convert idStudio mods into Atlan mods. The contents of `base.pk5` are parsed and placed into an Atlan zipfile. You do not need to be the mod author to use this conversion tool. However, if any edits are required to ensure compatibility with Atlan, you'll probably need to contact the mod author.

Usage

To run the idStudio Mod Converter:

1. Download **Atlan Mod Packager** and unzip it into your DOOM Eternal game folder.
2. Drag and drop the idStudio mod's `modinfo.json` or `modinfo.dev.json` onto `AtlanModPackager.exe`
3. Output: `AtlanPackage.zip` will be created inside the idStudio mod folder.
4. Take note of any errors in the console output. All console output is written to `packager_log.txt`

Advanced users may wish to run the idStudio mod converter from the command line. You may do this as follows:

```
AtlanModPackager.exe "path/to/modinfo.json"
```

Only one idStudio mod can be converted per execution of the packager.

Supported File Types

Not all file types encountered in a PK5 can be built into an Atlan mod zip. The following file types are supported. Adding support for more file types is possible, if there is sufficient demand.

Type	idStudio Path	Atlan Zip Path
Decls	generated/decls/	decls/
Strings	strings/<...>.blang	binaryFile/
Audio Files	sound/<...>.wem	audio/
Entities	generated/compfile/	compfile/

SWFs	generated/cswf/	cswf/
Images	generated/image/	image/
Static Models	generated/model/	model/
Rigged Models	generated/basemodel/	baseModel/
Animations	generated/anim/	anim/
Skeletons	generated/skeleton	skeleton/
Binarymd6defs	generated/binarymd6def	binarymd6def/

Atlan Config File

You may place an `atlanmod.txt` config file in your idStudio mod folder. Place it in the same folder as the modinfo JSON, and it will be packaged into the zip file. See **this page** to read about the Atlan config. file format.

idStudio Level Packager

> This is an advanced feature that requires careful preparation and attention to detail. Please read the wiki pages on **Packaging idStudio Mods** and **Known Fixable Issues** before proceeding.

Atlan Mod Packager can package idStudio level mods into a form that's usable with Atlan Mod Loader.

Atlan Config. File

To package an idStudio level, you **must** have an `atlanmod.txt` in your idStudio mod folder. Your config file **must** have the `load_level` property set. This tells the packager what to name the folder your level files get placed in.

This name must only contain lowercase letters and underscores. To prevent conflicts with other Atlan level mods, **this name should be unique to your level mod.** Some examples:

```
// BAD: Not unique names, may be used by other level mods and cause conflicts
load_level {
	mod
}
load_level {
	customlevel
}
```

```
load_level {
	hell_level
}

// GOOD: Sufficiently unique names that nobody else may accidentally use
load_level {
	velser_hell_crusade
}
load_level {
	svs_pvp_hurricane
}
```

PK5 Contents

Whether your level mod correctly works, is determined by whether your `base.pk5` is packaged correctly. To understand why, you must know how the level packager works:

Packaging a level in idStudio produces several archives:

- One `gameresources_mod.resources` and one `warehouse_mod.resources` archive, which get loaded globally.
  - These archives are typically very small, and mostly contain vanilla game assets. However, some files unique to your mod that need to be loaded globally may be packaged into these archives.
- One `game/shell/shell.resources` archive
  - This archive should always be empty, unless you're modifying something related to the main menu.
- One level-specific `.resources` that only loads when playing your custom level.
  - This is typically quite large, and will contain all your level's assets.
- One `.streamdb` file.
  - This is typically quite large, and is necessary for your mod to work properly.

Atlan does not package the gameresources or warehouse archives. Only the level-specific archives and the .streamdb file are packaged. Atlan relies on the assumption that any files packaged in these global archives are also packaged in the `base.pk5`. Atlan will analyze your level mod's global archives, and alert you if any unique files aren't included in the PK5. **Pay close attention to the console output, which gets written to the packager_log.txt. You may receive some false-negatives about files your mod doesn't edit. These can likely be ignored.**

Packager Output

When packaging an idStudio level, Atlan Mod Packager will generate 2 zip files:

`AtlanPackage.zip` - This is the zip file normally generated when packaging any idStudio mod. It contains your mod's Atlan config file, and the contents of `base.pk5`. There is nothing different about it.

`AtlanPackage_InstallOnce.zip` - A zip file unique to the level packager. It will contain:

- Your level's .resources and .streamdb archives, stored in `modarchives/<load_level>`
- Your level's baked lightmaps, stored in `virtualtextures/`
- A generated config. file with asset lists for your level mod's global assets.
  - **Do not modify this config. file if you don't know what you're doing.**
  - Found at `modarchives/<load_level_property>/atlanmod.txt`
- Another generated config. file with the `install_once` property set
  - This property tells Atlan Mod Loader to perform a level mod installation with this zip. This zip file will be deleted after installation is complete, keeping it's contents permanently unzipped on the user's disk.
  - The archives are unzipped and stored in `base/modarchives/<load_level>` while the lightmaps are unzipped into `virtualtextures`
  - This config file is named `atlanmod.txt` and is stored at the root of the zip file.

Loading Level Mods

Installing a level mod does not mean it's archives always get loaded by the game. This is where your `AtlanPackage.zip` (the Atlan mod with the PK5 data) matters. The config file's `load_level` property instructs Atlan to load your level's archives. This also means that multiple level mods can be loaded simultaneously, if several mods have a `load_level` property in their config. files.

Created 2026-10-13 07:02:19 PDT by FlavorfulGecko5
Updated 2026-11-13 07:18:16 PST by FlavorfulGecko5