

Eternal Atlan Strings Modding

Atlan Mod Loader supports injecting data into Doom Eternal's binary language (blang) files.

Extracting Vanilla Strings

To extract Doom Eternal's vanilla string data, use **Atlan Resource Extractor** to extract resources of type `binaryFile`. This will dump the game's vanilla strings to CSV files. Each language has its own blang file.

Creating String Mods

To setup a string mod for a particular language, add a new CSV file to your mod that matches the original .blang path.

For Example: to inject data into `binaryFile/strings/english.blang`

your CSV path should be `binaryFile/strings/english.csv`

The file extension doesn't matter. It can be .csv, .txt, or anything else you prefer.

CSV Format

Atlan Mod Loader requires your CSV obey the following format: `#STRING_NAME "STRING TEXT"`

To summarize:

1. Your string name MUST start with a `#`
2. Your string text MUST be wrapped in quotes `"Like this"`
3. The following escape sequences are supported: `\n \r \\ \" \t`
4. OPTIONAL: You may have a comma between your string's name and text.
5. HIGHLY RECOMMENDED: Your string names should only use lowercase letters, underscores, and numbers

You may refer to the vanilla string CSV files as examples. These are all valid string CSVs that Atlan can load.

You can replace the content of existing strings or add new strings.

The following is a basic example CSV that replaces several strings found in the main menu.

```
#str_code_mainmenu_dlc1_name "THE ANCIENT CHUNGUS 1"
#str_code_mainmenu_dlc2_name "THE ANCIENT CHUNGUS 2"
#str_code_mainmenu_extras_name "SUPER COOL EXTRA STUFF"
#str_code_mainmenu_horde_name "HORSE"
```

Default CSV

Doom Eternal supports 13 different languages, each with its own blang file. If you're only adding strings to your native language's blang, your custom strings will not exist when the game is played in other languages. To ensure parity between all languages, you may wish to add your untranslated strings to every blang file.

Atlan Mod Loader supports this with a special CSV: `binaryFile/strings/default.csv`

Strings in this CSV get added to all blang files that don't have their own CSV in the same mod.

For example: Let's say you have a mod with `binaryFile/strings/default.txt`

and `binaryFile/strings/french.txt`

1. The data inside `french.txt` gets injected into the French blang
2. The data inside `default.txt` gets injected into every blang **except** French

This way, your mod can have a default/fallback CSV while being localized in specific languages.

Additional Information

To learn how to add colored text and other customization to your strings: see [String](#)

Customization

Revision #2

Created 11 October 2026 18:13:50 by FlavorfulGecko5

Updated 11 October 2026 18:59:35 by FlavorfulGecko5