

How to create own package

Creating *.a3 package is pretty easy. A3 packages are basically zip files with renamed extension. So if you want to create one, for example *MyPackage.zip* rename to *MyPackage.a3*. That's it.

Structure of *.a3 package

Every package must contains *info.xml* or *info_replace.xml* (more later). This files define what the package will change in the game.

Here is example of info.xml:

```
<Info>
  <DefaultInclude>true</DefaultInclude>
  <Description>Bunker Opener
  You are able to go inside all bunkers / silos before mission start and even after regions are liberated
.</Description>
  <HeaderPicture1>bun1.jpg</HeaderPicture1>
  <HeaderPicture2>bun2.jpg</HeaderPicture2>
  <Category>3</Category>

  <Dependencies>
    <Dependency Action="and">
      <Package>Script Loader.a3</Package>
    </Dependency>
  </Dependencies>

  <Pairs>
    <Pair>
      <FromDatFile></FromDatFile>
      <Source><![CDATA[fc5_bunkersopener.lua]]></Source>
      <Target><![CDATA[armaniii_stuff/fc5_bunkersopener.lua]]></Target>
    </Pair>
  </Pairs>
</Info>
```

This is info.xml of Bunker Opener package. Now what all these tags mean:

DefaultInclude

If this is set to true, then this will be selected after first run of Mod Installer. Also, if an user click to „CLEAR ALL MODS“, then this package will be selected.

Description

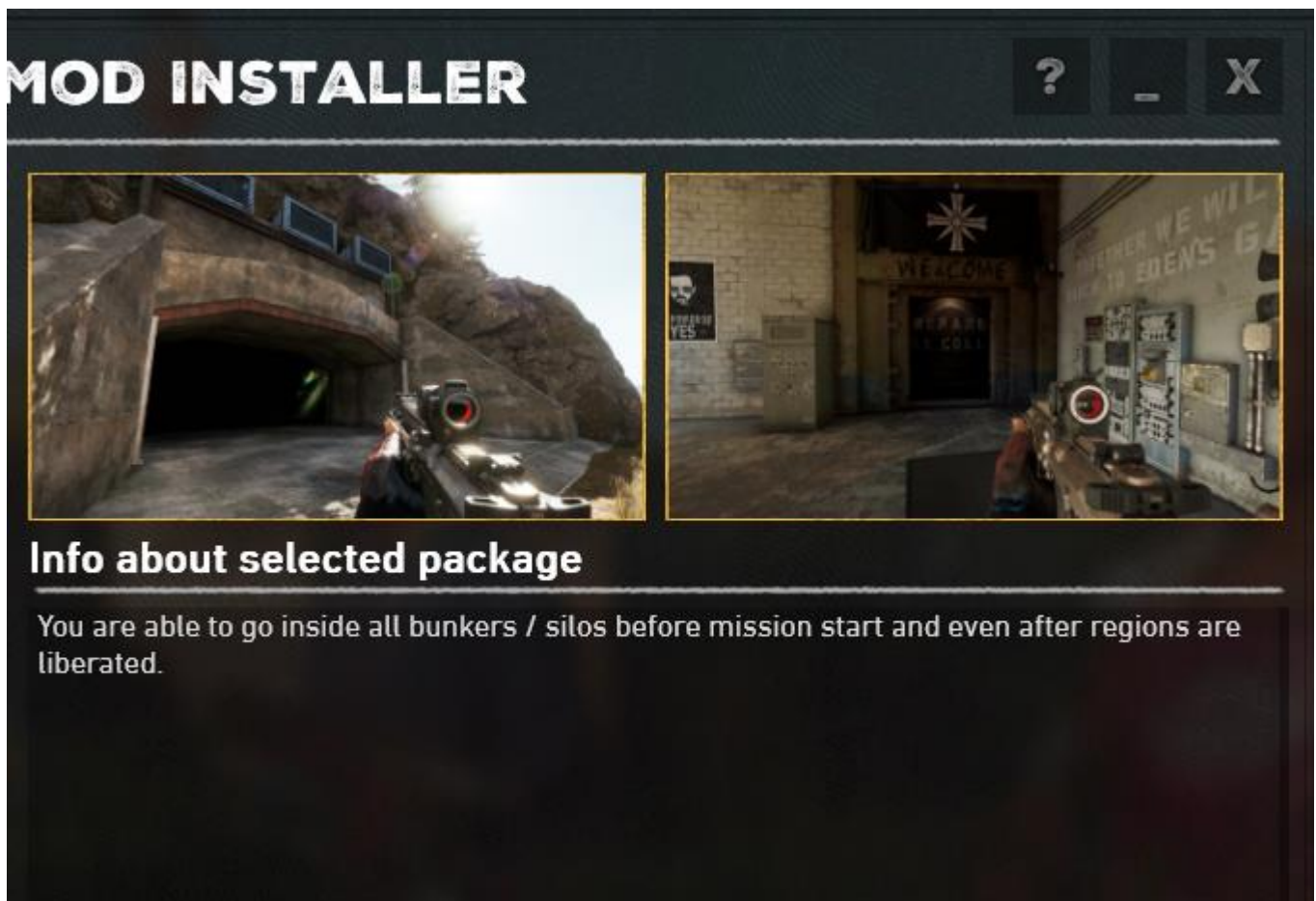
Defines name and description of the package. First line is name which is shown in list of packages. Second and all other lines are description of the package. Description also can contains BBcode tags (more later).

HeaderPicture1

Defines first (left one) header picture of the package.

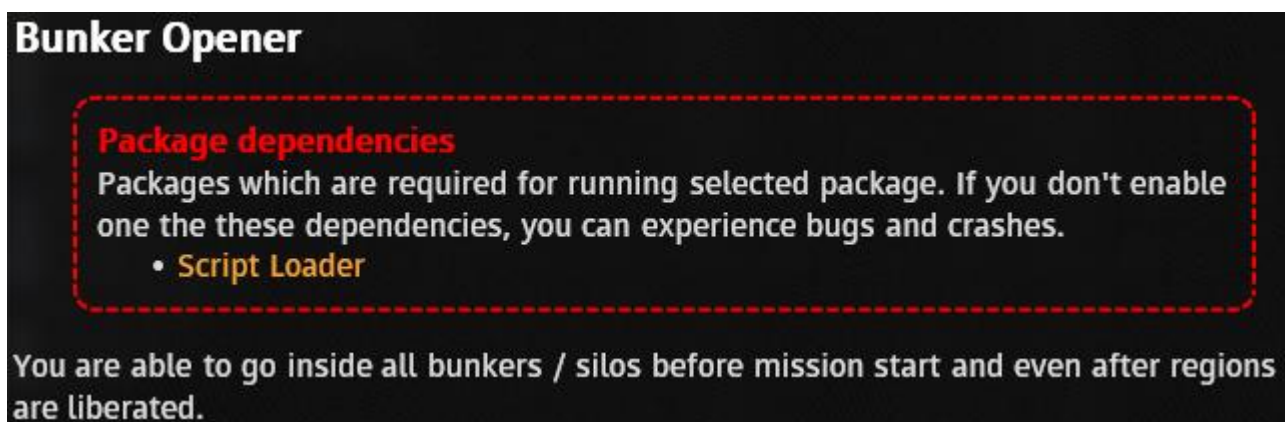
HeaderPicture2

Defines second (right one) header picture of the package.



Dependencies

Defines what packages are required by this one. Look below how it works. This tag is optional.



You can add as many *Dependency* tags as you want. Same as *Package* tag – it's also unlimited.

The *Action* attribute is required – it defines action of the dependency:

- **and** – this means that all defined packages in *Package* tags are required

- **or** – only one of package from *Package* tags is required
- **optional** – all defined packages in *Package* tags are optional

The *Dependency* tag has one optional attribute – *Name*. It can be used if you don't want to show list of required packages names, but you want to show your own text.

The *Package* tag defines required package. It must contain exactly the file name of required package including extension.

Category

Defines which category will this package have. You can use categories which are defined by Mod Installer. Their IDs and names (names are set by language ID, so you must search in languages) are defined in Categories.xml file.

OwnCategory

Basically same as *Category* but with this you can define your own category. Just write some text. Also, if you make more packages with same *OwnCategory* tag value, all these packages will be in one category.

This category is always on top in list of packages.

OnlyOne

When you have two replace packages which modifying same things, example can be Random NPC Weapons, then Mod Installer will not mark those packages as conflicting. If you don't want to allow to use those packages at the same time, you define this tag. Usage is simple: create an unique name and put it to this tag, like:

```
<OnlyOne>random_npc_weapons</OnlyOne>
```

Then put this tag to another package. And then, those two packages will not be allowed to be selected at the same time.

This tag can be used also for info_replace.xml

Plus18

This tag can be used if a package contains NSFW content. It shows a warn message box when an user select the package.

This tag can be used also for info_replace.xml

Pairs

There are defined files which will be packed if this package is selected. You can add as many as you want, it is not limited.

FromDatFile

If you want just replace a file, you don't have to include it, you can use this tag where you define path to another fat file. Defined file will be extracted from this fat file and inserted to defined target file. **This tag is not required and can be omitted.**

Example: worlds\installpkg.fat

Source

Name of a file in *.a3 package. If you're using *FromDatFile*, you must define whole file path inside dat file.

Target

Target file name. A file will be inserted to this path.

UseHD

If set to **true**, then the file defined in Target will be packed into installpkg_hd.dat, because if HD pack is installed, all textures in patch.dat are overridden with HD. If somebody doesn't have HD pack installed, then all files marked with **UseHD** will **NOT** be packed, if they should be packed, use the tag below. **This tag is not required and can be omitted.**

UseHDAndPatch

If set to **true**, then the file defined in Target will be packed into installpkg_hd.dat and into patch.dat.

Overwrite

If set to **true**, it can overwrite same file which is set in another package. Example: you have ported clothes with textures in one package, and now you want a small replacement, so you create new package with one replacement texture, and when defining *Pair*, you set this tag. **This tag is not required and can be omitted.**

Prepacked

It can be used only with prepacked files. It requires only *Source* tag.

Alternative structure of *.a3 package

If your package has a lot of files, then adding all of them to *Pairs* tag could be very time consuming. So you can use alternative structure.

You will not use *Pairs* tag in info.xml, just do not write it here. Info.xml then will look like this:

```
<Info>
  <DefaultInclude>true</DefaultInclude>
  <Description>Bunker Opener
  You are able to go inside all bunkers / silos before mission start and even after regions are liberated
.</Description>
  <HeaderPicture1>bun1.jpg</HeaderPicture1>
  <HeaderPicture2>bun2.jpg</HeaderPicture2>
  <Dependencies>Script Loader</Dependencies>
  <Category>3</Category>
</Info>
```

Now you must create same folder structure as it will be in patch.dat file. So our file *armaniii_stuff/fc5_bunkersopener.lua* will be placed in *armaniii_stuff* folder. Structure of the package will then look like this:

```
MyPackage.a3 / armaniii_stuff / fc5_bunkersopener.lua
MyPackage.a3 / info.xml
MyPackage.a3 / bun1.jpg
MyPackage.a3 / bun2.jpg
```

Note: all files with extension jpg, png, gif will not be packed.

How to use folder structure for packing to HD?

As for Pair there is *UseHD* tag, in alternative structure there is any *Pair* tags, so you must create a directory named **UseHD** (case sensitive) and put all files you want to pack to HD there.

So, it will look like this:

```
MyPackage.a3 / armaniii_stuff / fc5_bunkersopener.lua
MyPackage.a3 / UseHD / armaniii_stuff / fc5_bunkersopener.lua
```

```
MyPackage.a3 / graphics / <any files and directories will be packed to patch.dat>
MyPackage.a3 / UseHD / graphics / <any files and directories will be packed to installpkg_hd.dat>
```

Note: HD means the file *installpkg_hd.dat* is used for packing. If somebody doesn't have HD pack installed, then all files in **UseHD** will be packed into *patch.dat*

BBcode in description

You can use styling of text in descriptions.

Due to bad UI control which displaying description text, some tags are a bit different than they should be.

If you make a mistake in BBcode, Mod Installer show error when you move over package item in list of packages.

Basic tags

```
[b]bold text[/b]
some text
[i]italic text[/i]
some text
[u]underline text[/u]
[s]strikethrough text[/s]
[color=#fba536]colored text[/color]
[color=green]green text[/color]

[b]Some[/b] [i]long[/i] [u]text[/u] [s]where[/s] [color=green]are[/color] placed multiple styles on one
line.
```

```
[color=red][b]Please do not use underline and strikethrough at the same time. The program can't display it.[/b][/color]
```

As stated in example, you cannot use underline and strikethrough at the same time.

In color tags you can use text names of colors – green, black, red, and more.. or you can use hex color - #fba536.

Dotted list

```
Dotted list
[list]
[*]List item 1
[*]List item 2
[*][color=green]List item 3[/color]
[/list]
```

Pictures

You can of course insert pictures. Picture have max width of description area and this cannot be changed. You can of course make a half of picture transparent, then the picture will be only half size of description area. Also, if the picture is smaller than description area, it will be resized to max width.

```
[img]pic.jpg[/img]
```

Img tag must be alone on whole line, otherwise it will not work.

Tables

You can use two types of table – bordered and non-border.

This example will make table without borders.

```
Simple table without borders
[table]
[td]
  [tr][b]table cell 1[/b][tr]
  [tr][i]table cell 2[/i][tr]
[/td]
[td]
  [tr][u]table cell 3[/u][tr]
  [tr][s]table cell 4[/s][tr]
[/td]
[/table]
```

And this is bordered table:

```
Simple table with borders
[table style=1]
[td]
  [tr]table cell 1[/tr]
  [tr]table cell 2[/tr]
[/td]
[td]
  [tr]table cell 3[/tr]
  [tr]table cell 4[/tr]
[/td]
[/table]
```

You just add parameter to *table* tag.

You can also set width of column in pixels. This example set first column to be 400 pixels width.

```
Simple table with borders
[table style=1]
[cols]
  [col=400]
  [col]
[/cols]
[td]
  [tr]table cell 1[/tr]
  [tr]table cell 2[/tr]
[/td]
[td]
  [tr]table cell 3[/tr]
  [tr]table cell 4[/tr]
[/td]
[/table]
```

You add after *table* tag *cols*. Tags *col* must match count of columns of a table otherwise Mod Installer will not load description. Width is in pixels.

Text size (headers)

There are three sizes of headers.

```
[size=1]Font size 1 more than normal[/size]
[size=2]Font size 2 middle[/size]
[size=3]Font size 3 big[/size]
```

Info_replace.xml

This info is for replacing values in fcb files. It is due to many small changes in big files like **nomadobjecttemplates_rt.fcb**.

Also imagine that we want four changes in the file. If the changes are alone the it is OK – we have only four combinations. But what if somebody want Change 1 to be merged with Change 3? And what about Change 1 and Change 2 and Change 4? I can continue, but then we get $4 * 4$ combinations, and that's 16 different packages with same file. Also, for mod file size it is too much, the file in package has 4MB, so $4 * 16$ and that's 64MB!

So that's why this type of info.xml. You just define what you need to change in the file and Mod Installer replace it when packing. Only one negative is that packing process can take 1 minute instead of few seconds.

Anyway, if you want to know how to use this info_replace.xml, see another tutorial „How to use new replace package“.

That's all for now. I also make more tutorials.

As always, if you want ask something, do not hesitate to ask me :D