

# Additional technical information to CP 1.70

## Table of contents

1	Modding.....	2
1.1	Handling of .mod and .nod files implemented.....	2
1.1.1	Introduction: Basic knowledge about data files.....	2
1.1.2	First extension: file ending ".cpt" .....	2
1.1.3	Modding extensions: file endings ".mod" and ".nod" .....	2
1.1.4	Bonus information: What to consider when modding dialogs .....	3
1.2	A few new parameters in .info files .....	4
1.2.1	CondItemAmounts=0.....	4
1.2.2	CondHasSkill.....	4
1.2.3	CondPlayerKnowsNot .....	4
1.2.4	CondWearsItem .....	5
1.2.5	Erase.....	5
2	New parameters in ge3.ini .....	6
2.1	DisableDEP .....	6
2.2	Threads.Priority .....	6

# 1 Modding

For those of you who want to develop modifications for Gothic 3, we added a small amount of new features.

## 1.1 Handling of .mod and .nod files implemented

### 1.1.1 Introduction: Basic knowledge about data files

You perhaps already know the data organization of Gothic 3: All game data is "stored" in files with the ending ".pak". Example: "Music.pak".

If you want to change (edit) content of this file, you have two options:

1) replace the entire .pak file (this could be impractical if the file is very big, and you change only very few of it), or

2) keep the old .pak file and just add a new "generation" of the .pak file.

The next "generation" of the file Music.pak is: Music.p00, followed by Music.p01, then Music.p02 and so on.

Always make sure that there is no gap in the consecutiveness of the generations, because the engine will stop reading if it can't find a file with the subsequent ending.

Example: If there are three Music files in the data directory, named Music.pak, Music.p00 and Music.p02, the game will only read until Music.p00 and ignore Music.p02, because ".p01" is missing.

### 1.1.2 First extension: file ending ".cpt"

With CP 1.70 we introduced the feature "Alternative Balancing" ("AB"). For this, we needed to separate the modified data of the AB from the original data of the regular game version. This became possible by establishing the new file ending ".cpt".

All files with this ending will only be loaded into the game if the switch "Alternative Balancing" is set to "on". If AB is deactivated, .cpt files will be ignored by the game.

If AB is switched on, the engine will first load the contents of the .pak files and their generations, and then all .cpt files. So the contents of the .cpt files will overwrite the corresponding data of the .pak files.

Further generations of .cpt files, if there should be any some day, will have the endings ".c00", ".c01" etc.

### 1.1.3 Modding extensions: file endings ".mod" and ".nod"

Our request to all modders of Gothic 3 is to **keep all .pak and .cpt files as they are now**. Please do not exchange them or add further generations to them!

To enable a strict separation of the files of the "original game" and the files of modifications, we invented two other new file endings. They should make it easier for the players/users to keep an overview over the contents of their Gothic 3 data folder and to identify modifications, and we hope it's helpful for the modders themselves, too. If there should actually be new real patches some day in the future, these patches will be able to

continue using .pak and .cpt files without risking to delete or overwrite files of modifications.

There are just three rules you have to understand and keep in mind when deciding how to name your data files:

- **.mod** files continue the sequence of .pak files and will be loaded by the engine *in any case*.
- **.nod** files are the modding version of .cpt files, meaning that their contents will only be loaded by the game *if Alternative Balancing is switched on*.
- Every file with a "modding" file ending will "win" over every "regular" file ending.

Here's a complete overview of the data loading order of Gothic 3:

- (1) .pak files with all generations
- (2) .cpt files with all generations (only loaded if "Alternative Balancing" is active)
- (3) .mod files with all generations
- (4) .nod files with all generations (only loaded if "Alternative Balancing" is active)

Of course, the generations of .mod files must have the ending ".m00", ".m01", etc., and the generations of .nod files are called ".n00", ".n01" and so on.

#### 1.1.4 Bonus information: What to consider when modding dialogs

To change or add dialogs, it's necessary to edit/add two groups of files: Infos and Strings (or the stringtable.ini, to be more specific).

In both cases, your modded files will only take effect in the game if you **create binary versions** of those files and copy them into the data folder, too. (Or if you delete or rename the files Projects\_compiled.p00 and Projects\_compiled.cpt, but this would noticeably slow down the loading process of the game!)

Here's how it's done:

- 1) Create a new subfolder "Infos" in the "Data" directory of your Gothic 3 installation.
- 2) Create a new subfolder "Strings" in the "Data" directory of your Gothic 3 installation.
- 3) Make sure that your modded stringtable.ini and **all** info files (the ones you changed/added, and all unchanged info files!) are in the correct places in the data folder.
- 4) Launch Gothic 3, then start a new game
- 5) Close Gothic 3
- 6) Open the folder Gothic III\Data\Infos
- 7) There's a file "\_\_compiledinfos\_G3\_World\_01.bin" now - rename it to "compiledinfos\_G3\_World\_01.bin" (without the "\_")
- 8) Open the folder Gothic III\Data\Strings
- 9) There's a file "\_\_stringtable.bin" now - rename it to "stringtable.bin" (without the "\_")
- 10) Create a new folder "Projects\_compiled" somewhere else on your hard drive
- 11) Copy or move both .bin files of steps 7 and 9 into this new folder
- 12) Create a .pak file from the folder with the tool G3PakDir.exe. The result should be a file with the name " Projects\_compiled.pak"
- 13) Rename the file to Projects\_compiled.mod or Projects\_compiled.nod - depending on your intentions with your modification
- 14) Copy this file to your Data folder, and don't forget to publish it in your mod packet together with your Infos.mod and Strings.mod (or \*.nod...)
- 15) Delete the subfolders " Gothic III\Data\Infos " and " Gothic III\Data\Strings" again!

## 1.2 A few new parameters in .info files

A few modders asked us to expand the possibilities of dialog design. We could only program a few due to lack of time, but we hope they will come in handy.

### 1.2.1 CondItemAmounts=0

The well-known condition "CondItemAmounts" is related to the other parameters "CondItem" and "CondItemContainer". Example: The entries

```
CondItemContainer=Markus  
CondItems=It_FiremageCup  
CondItemAmounts=1
```

mean that the given dialog would be available if the NPC "Markus" had at least one fire chalice (It\_FiremageCup) in his inventory.

But until now, it wasn't possible to trigger a dialog if a NPC **didn't** have a particular item somewhere in his inventory.

With CP 1.70, you'll achieve this by using "CondItemAmounts=0".

Attention: Be careful with traders! They might not have the item of interest in their "loot inventory", but nevertheless might have it somewhere in their trade inventory. This may lead to unexpected dialog constellations.

### 1.2.2 CondHasSkill

The meaning of this new condition parameter should be obvious.

You can concatenate several skills - and spells, by the way -, separated by semicolons.

Syntax:

```
CondHasSkill=Perk_Barter;Spl_IceBomb  
(all perks and spells without the prefix "It_")
```

In this example, the dialog will only be available if the hero learned the skill "barter" and the spell "ice explosion".

Attention: This new condition is always related to the hero, not to any NPC! (NPCs don't have skills or know spells anyway.)

And: The condition is only fulfilled if the hero has actually learned the skills and spells from a teacher. Bonuses of armor, weapons etc. don't count!

### 1.2.3 CondPlayerKnowsNot

This new instruction is the exact opposite of the well-known instruction "CondPlayerKnows". This means that the condition is fulfilled if a certain GameEvent has not been set before, or has already been cleared again.

Syntax:

```
CondPlayerKnowsNot=TheHeroDoesntKnowThis
```

## 1.2.4 CondWearsItem

This new parameter is always related to the parameter "CondItemContainer". And in every info file, there can only be a maximum of one "ItemContainer".

The meaning of this condition should be clear: If a particular NPC is wearing a particular item, e.g. a weapon, ring, amulet, etc., the dialog will be available.

Example:

```
CondItemContainer=PC_Hero  
CondWearsItem=Body_Druid
```

Result: The dialog of this info file can only be activated when the hero is wearing a druid's robe.

Please make sure that there will be no situations where you need more than one CondItemContainer in one .info file! And please stay aware of the fact that the CondItemContainer is related to the parameters "CondItems" and "CondItemAmounts" on one hand and "CondWearsItem" on the other hand now. There's a slight risk of confusing these two things.

## 1.2.5 Erase

In contrast to the other four .info file innovations mentioned before, this is not a condition parameter, but a new command for the InfoScript lines. Just like "Say", "Give", "SetGameEvent", etc.

With "Erase", you can silently delete a specified amount of items from the inventory of a specified NPC. (This command was initially created for the feature "milk cows", to delete one empty bottle from the player's inventory each time he milks a cow, without a message being displayed.)

Syntax:

```
InfoScript_Commands=Erase  
InfoScript_Entities1=player  
InfoScript_Entities2  
InfoScript_IDs1=It_Gold  
InfoScript_IDs2=100  
InfoScript_Texts
```

We don't know if this command makes sense in other contexts. Please don't misuse it - stay fair to the players. :-)

## 2 New parameters in ge3.ini

There has been a moderate amount of changes done to the ge3.ini. Two new entries might require some further explanation.

### 2.1 DisableDEP

Some players, especially Vista users, can't actually play the game, but instead receive a message box saying "Genome Gaming System has stopped working". The main reason for this is a security mechanism of Vista (and some other operating systems) called "Data execution prevention", or in short: DEP.

This protection mechanism can be switched off manually in the system control panel of your operating system. For those of you who don't want to switch it off generally, or who don't know how to do it, we included the parameter "DisableDEP" in the ge3.ini.

If this parameter is set to "true", hopefully the mentioned error message won't show up again.

If you don't have any trouble with this message anyway, please keep using the default value, "false".

You use this parameter at your own risk!

There still might occur problems while playing the game.

### 2.2 Threads.Priority

With this switch, you can give Gothic 3 a higher priority compared to other applications currently running on your computer, e.g. Acrobat Reader. The main intention of increasing the priority of Gothic 3 is to provide more CPU time to the game. This should result in a better performance, but that's not generally guaranteed.

The values are:

0=Normal (default),

1=Above Normal,

2=Highest

Try it out if you like, but if you notice any negative side effects, undo your changes again.

We know from our tests that a priority value of 1 (above normal) can evoke problems with the game control on weak/slow computers, resulting in an increasing delay of every keyboard or mouse action.

So - all changes of the Thread.Priority are at your own risk.