

Gontc manpage

August 12, 2002

1 NAME

gontc - compile and link Gont files

2 SYNOPSIS

gontc [*OPTIONS*] *files...*

3 DESCRIPTION

gontc is used to compile source Gont into binaries, and then linking them together. It uses concept of module in managing compilation. Module consists of interface (in *.gi source file), and implementation (in *.g source file). In order to compile implementation compiler has to be able to find compiled interface for module being compiled, and all modules it depends on.

Typical compilation involves running, for each module in project, `gontc -c module.gi` to compile interfaces, then `gontc -c module.g` to compile implementation, and finally `gontc -o prog mod1.go mod2.go...`

4 OPTIONS

gontc recognizes following options:

`-c`

just compile, don't try to link

`-save-temps`

save temporary files. In normal circumstances temporary *.ksi files produced during compilation, and auxiliary linker files are removed.

`-ksi OPT`

pass option to Ksi compiler. Example usage might be: `-ksi -fomit-frame-pointer`

`-ksicc PROG`

set Ksi compiler, for example `-ksicc ppc-pld-linux-cc`

`-O0`

don't optimize. Same as `-ksi -O0`.

`-O1`

optimize. Same as `-ksi -O1`.

-O2

optimize more. Same as **-ksi -O2**.

-O3

optimize even more. Same as **-ksi -O3**.

-Werror

treat warnings as fatal.

-g

produce debug info. Same as **-ksi -g**.

-S

just generate *.ksi file and stop.

-o *OUTFILE*

set linker output file name. This doesn't work with **-c**

-B *PREFIX*

override prefix. Prefix is directory in which **gontc** looks for standard libraries, and from which configuration is read.

-v

be verbose, display what is being done.

-dump *MODULE*

dump named module and exit.

-gia

operate in make interface archive mode. Usage is then: **gontc -gia *archive.gia mod1.gio...***

-ga

operate in make compiled implementation archive mode. Usage is then: **gontc -ga *archive.ga mod1.go...***

-ugly

generate ugly Ksi code. This might be somewhat faster.

-pretty

generate pretty Ksi code. Slower, useful, when you have to look at the generated code.

-L *DIR*

add directory to library search path. This affects only looking for *.go and *.ga files. Use **-ksi -L** if you want to change path for *.a files.

-I *DIR*

add directory to interface search path. **gontc** looks for compiled interfaces (*.gio) there. You can also specify *.gia archive instead of directory.

-link-aux *NAME*

set name of auxiliary Ksi file generated during link ('gont-link-aux.ksi' by default).

-link-init *NAME*

set name of generated initialization function ('gont_init' by default).

-link- ni *NAME*

set name of generated initialization function ('gont_ ni' by default).

-link-main *NAME*

set name of generated toplevel function ('main' by default).

-just-make-aux

just make auxiliary linking file and stop.

-only-needed *LIBRARY*

add only needed files from named library. This is default behavior for standard libraries.

-lib *LIBRARY*

same as -only-needed

-greedy *LIBRARY*

add all files from named library.

-ciface

this option causes gontc to output C header files for named modules, i.e. all subsequent non-option arguments are treated as names of modules to dump. Example usage might be:
gontc -ciface FooBar > foobar.h.

-help

display list of options.

{ help

same as -help

5 FILES

gontc recognizes type of input file based on extension. Following rules apply:

***.g** *Gont sources*

are compiled into *.ksi and *.go. *.ksi file is passed to Ksi compiler, that produces *.o object file with machine code.

***.gi** *Gont interface sources*

are compiled into *.gio.

***.go** *Gont compiled implementation file*

are used to resolve linking dependencies. You should never supply *.o file resulting from *.g compilation for linking. Always use *.go file. Otherwise linking error will occur (unresolved reference to Module\$\$\$_go_ le_linked_\$ or something similar).

***.gio** *Gont compiled interface files*

gontc doesn't know what to do with this kind of files. I.e. they cannot be passed as command line options. They are open implicitly during compilation. -I option can be used to help gontc find this kind of files.

****.ksi Ksi source les***

are passed to Ksi compiler. It should produce *.o les from them.

****.o Object les with machine code***

are just passed to linker. Never use *.o les, produced from *.g, directly. Always use *.go les.

****.a Libraries of object les with machine code***

are just passed to linker. Never use *.a les, produced from linking results of *.g compilation together, directly. Always use *.ga les.

****.ga Archives of Gont implementation les***

are unpacked to memory, and then treated as if each of *.go les inside was speci ed in command line.

****.gia Archives of Gont interface les***

similarly to *.gio les, gontc doesn't know what to do with these beasts. They can however be passed as arguments to -I option, instead of directories to make gontc look in archive when looking for interfaces.

6 AUTHOR

This manpage was written by Michal Moskal <malekith/at/pld-linux.org>.