

Information about PManager

COLLABORATORS

	<i>TITLE :</i> Information about PManager		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		December 18, 2022	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	Information about PManager	1
1.1	PManager	1
1.2	Program Introduction	2
1.3	System requirements	2
1.4	Installation	3
1.5	Configuration	3
1.6	Copyrights	4
1.7	Disclaimer	4
1.8	History	5
1.9	Known bugs	5
1.10	Fixed bugs	5
1.11	How to use	5
1.12	To do	8

Chapter 1

Information about PManager

1.1 PManager

PManager v1.2

Written by
Andrei Chereshnev (Russia)

*** FREeware ***

Introduction

System requirements

Installation
- Important!

Configuration
- Important!

How to use

History

Known bugs

Fixed bugs

Todo

Copyrights

Disclaimer

Please, send me your opinion and suggestions about PManager:
acheresh@ic.ru (Andrei Chereshnev).

1.2 Program Introduction

PManager is a program which lets you manage C projects. It uses Listtree custom class for hierarhical appearance, like in StormC. You needn't any project file, just run PManager with path name and it will scan directory for files. Edit config file to use your C environment, SAS/C default.

Features:

1. Managment of multiple projects at once.
2. Use mknk utility for auto smakefile creation (SAS/C).
3. Arexx script for use PManager with GoldEd v3 and v4 (load file even when GoldEd's port is not open). *NEW*
4. You can manage GST creation from check menu option (SAS/C).
5. Edit, Compile, Make, Build, Run, Debug, Options, Generate makefile options.
6. Config output for compiler tools and Run option.
7. Quick access to projects windows (hotkey).
8. Quick access to Message Browser window (SAS/C). *NEW*
9. Snapshot/UnSnapshot current open projects. *NEW*
10. And others...

1.3 System requirements

Required:

- MUI 3.7 and C compiler (for example SAS/C 6.xx, gcc, vbcc).
I have only tested with SAS/C!
- Minimum 3MB free memory to use MUI and C compiler.

Recommended:

SAS/C 6.xx and GoldEd(v3.xx or v4.xx).

This application uses

MUI - MagicUserInterface

(c) Copyright 1992-97 by Stefan Stuntz

MUI is a system to generate and maintain graphical user interfaces. With the aid of a preferences program, the user of an application has the ability to customize the outfit according to his personal taste.

MUI is distributed as shareware. To obtain a complete package containing lots of examples and more information about registration please look for a file called "muiXXusr.lha" (XX means the latest version number) on your local bulletin boards or on public domain disks.

If you want to register directly, feel free to send

DM 30.- or US\$ 20.-

to

Stefan Stuntz
Eduard-Spranger-Straße 7
80935 München
GERMANY

Support and online registration is available at

<http://www.sasg.com/>

1.4 Installation

If you not use PManager before:

Copy the program directory into one of your choice. Then if you are not using SAS/C and(or) GoldEd go to the
Configuration

.

If you are using GoldEd v4.xx: copy "edit_file.ged"(in PManager/arexx) to GOLDED:arexx/ directory, if you have GoldEd v3.xx rename "edit_file.ged.v3" to "edit_file.ged", then copy "edit_file.ged" to GOLDED:arexx/.

If you already have PManager version:

If you are not using GoldEd just replace pmanager.config file in v1.2 by yours AND INSERT "run " BEFORE EDITOR NAME! .

If you are using GoldEd v4.xx: copy "edit_file.ged"(in PManager/arexx) to GOLDED:arexx/ directory, if you have GoldEd v3.xx rename "edit_file.ged.v3" to "edit_file.ged", then copy to GOLDED:arexx/.Then replace "edit= editor_name" by "edit= rx GOLDED:arexx/edit_file.ged".
The End:).

1.5 Configuration

PManager.config file:

```
----- Default -----  
edit=    rx GOLDED:arexx/edit_file.ged  
compile= sc  
make=    smake  
build=   smake -u  
debug=   cpr -w  
makefile= mkmk force target  
options= scopts  
tool_out=CON:0/50//130/PManager/CLOSE/AUTO/NOWAIT  
run_out=CON:0/50//130/PManager/CLOSE/AUTO/WAIT  
----- Default -----
```

Edit PManager.config file to use your editor, compiler etc.. By default it is SAS/C environment and GoldEd editor. When need it, file name from the project will be added by PManager to the end of the lines: edit=, compile=, debug=, makefile= . For example "run ed test.c", "sc test.c", "mkmk target test", "cpr test". "make=", "build=" and "options=" are tools that have no args(from PManager).

"edit=" - if you are not using GoldEd then replace default by "edit= RUN your_editor" (example: "edit= run ed").

"compile=" - usually frontend to compiler.
"make=", - "make" utility (make, cmake, ...)
"build=" - "make" utility but with "build key".
"debug=" - debugger you want to use.

"makefle=" = makefile = mkmk:) is a sas/c tool which makes developing one target projects easy(no need to edit makefile). I hope that in future there will be "makefile generators" for other compilers too.

"options=" - utilities like "scopts" help to edit many config options for cli compiler(include directories, defines, data & code models, debug info, cpu, optimizer and others) and linker(libs & objects to add) and save them to locale options file.

"tool_out=" - out from make, build and makefile.
"run_out=" - out from the Run option, compile and debug.

The main idea of "tool_out=" is that when tool ends its work, the console window will close automaticaly (.../NOWAIT). In such case you need message browser to watch compiler errors and messages. If you use sas/c then choose "Options..." from PManager menu, in open window choose "Message Options..." and then set "ErrorRexx" and "NoErrorConsole", click OK and Save Default.

1.6 Copyrights

The program PManager is copyright © 1998 by Andrei Chereshnev. All rights reserved.

PManager is FREeware. This program may be freely distributed as long as:

- the executable and documentation remain unchanged and are included in the distribution.
- no other charge is made than to cover time and copying costs.

1.7 Disclaimer

I CANNOT BEEN MADE RESPONSIBLE FOR ANY PROBLEM WHICH OCCURES USING THIS PROGRAM. THIS AFFECTS ALL KINDS OF PROBLEMS. YOU ARE USING THIS PROGRAM AT YOUR OWN RISK.

1.8 History

v1.2:

- New "Run" and "Debug" buttons in Run window.
- Asynch Run windows.
- New "Message Browser" option (sas/c).
- Requesters for "New File" and "Delete File" options.
- Arexx script (edit_file.ged) for use PManager with GoldEd v3 and v4.(it will run GOLDED:ED when GoldEd's port is not open)
- Snapshot/UnSnapshot current open projects (PManager.prefs file).
- If compiling is ok and "Make GST file" check on, then "turn off make GST?" will pop up (sas/c).
- If contents of Run window not empty(you have target name) then "Generate makefile" will be without requester.
- Contents of Run window will now saving.
- "Compile" will now with busy pointer.
- Recognized: read.me, #?.cxx, #?.cc, #?.exe .
- Improved docs.
- Misc minor improvements.
- Misc minor bugfixes.

v1.01b:

- Bugfixes in unlocking code.
- Misc minor bugfixes.

v1.00b:

- Initial release.

If you find any bugs, have a suggestion or comments please e-mail me.

1.9 Known bugs

If you find any bugs, have a suggestion or comments please e-mail me:
acheresh@ic.ru (Andrei Chereshnev).

1.10 Fixed bugs

If you find any bugs, have a suggestion or comments please e-mail me:
acheresh@ic.ru (Andrei Chereshnev).

1.11 How to use

Starting a program

```
PManager [dir] ...
```

You can run PManager, for example, from cli or (in my case) ToolManager like program(Exec Type - Shell !!!). If you supply PManager with dir name(s), it will open window(s) with contents of that dir(s).(example: "pmanager work:hello_world!" if this project directory contains source files, makefile, etc. the window will list them in correspondent sections, if "hello_world!" is empty you can start

creating this project). Running PManager without directory name(s) will popup requester telling you to select project directory.

I think, that the best place from to run PManager is a ToolManager (freeware utility on the aminet, Copyright (C) 1990-1998 Stefan Becker):

1. Quick start (from dock, menu and hotkey).
2. By restarting PManager, it will move its project windows to front.
3. You can Drag&Drop dir(s) on PManager.

Starting new project

Lets create "hello_world!" project:

. Start PManager, in popup requester choose directory where you want to be "hello_world!" dir, OK. Then select New Dir... from Project Menu in open requester enter "hello_world!". The project window name will change to "Project: hello_world!". Select New File... and enter "hello_world.c". The editor window will be open with "hello_world.c" file. Enter:

```
#include <stdio.h>
#include <stdlib.h>

void main()
{
    puts("Hello World!");
}
```

Save file. Now you have section - "Sources" with "hello_world!.c" file. Then if you have SAS/C, just select Generate makefile... in Compile Menu and in open requester enter target - "hello_world!".

Later, in many sources projects, you will need to (Re)Generate makefile... every time you add the new file to a project(?.c, ?.cpp, ?.h etc. files) and please, keep in mind, that the file without extension and .exe file(demo coders) recognized by PManager as executable.

If you havent SAS/C, you need to create "makefile", by selecting New File....

Now, you are ready to "make" project: select Make All.... If making - successful, the "Run" window will be open with "hello_world!". You can press Enter to execute, edit string to add args, "debug" program.

If you want to work with currently open project(s) next time, you can Snapshot all of them. When you start PManager, all snapshot(ed) projects will be open.

GST - Global Symbol Table (SAS/C)

GST - is a precompiled headers. In PManager you can "make" and "use" .gst file from option - Make GST file. First, you need to enter the name of .gst file: choose Options..., "Compiler Options", and in "GST" enter: "hello_world!.gst ←
",

click OK and Save. The Make GST file option is now enable, telling you that "hello_world!.gst" will now in use. Now you need to create "hello_world!.gst" file - choose Make GST file to check it on.

At first time, choose Build All... to build project and create .gst file, next time you add new #include <.h> or #include ".h" to the sources or edit one of the headers, you only need to check on Make GST file and then Make All....

After successful compiling, making or building, the "turn off make GST file?"

requester popup - OK. Now you have new (re)created .gst file.

Menus

Project Menu

Create new project window
Select New... and enter project directory.

Open new project
Select Open... and enter new project directory.

Create new file
Select New File... and enter new file name.

Create & Open new project directory
Select New Dir... and enter new directory name.

Delete active project file
Select Delete File

Edit active project file
Select Edit... or double click on the filename.

Snapshot
Save all currently open projects dirs to PManager.prefs.

UnSnapshot
Delete PManager.prefs file.

Close all open windows
Select Quit All.

Compile Menu

Compile... - compile active file.
Make All... - if successful, Run window will be open (see below).
Build All... - if successful, Run window will be open (see below).

Run... - Open window with filename from Others section
To run another file clear Run window(A+x) and press Enter,
then activate new file and choose Run....
You can also run by double clicking in Others section.

Debug... - debug last Run window contents.
Options... - compiler options.
Generate makefile... - enter target filename (SAS/C).
Make GST file - You need to enter in scopts GST filename for
your project, and then you may control creation GST
from this option (SAS/C).

Window Menu

Message Browser - activate scmsg window(SAS/C).

1.12 To do

If you have a suggestion or comments please e-mail me:
acheresh@ic.ru (Andrei Chereshnev).