

gsm

COLLABORATORS

	<i>TITLE :</i> gsm		
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>	<i>SIGNATURE</i>
WRITTEN BY		January 31, 2023	

REVISION HISTORY

NUMBER	DATE	DESCRIPTION	NAME

Contents

1	gsm	1
1.1	Contents	1
1.2	Copyright and Disclaimer	2
1.3	Introduction to GSM	3
1.4	System Requirements	3
1.5	Installing GSM	3
1.6	Use GSM from Workbench	4
1.7	Use GSM from the Shell	4
1.8	GSM's ToolTypes	5
1.9	GSM Features	7
1.10	GSM Menus	8
1.11	Flush Memory	9
1.12	Flush Memory and Remember	9
1.13	Save Settings	10
1.14	About GSM	10
1.15	Quit GSM	10
1.16	Display free Chip memory	11
1.17	Display free Fast memory	12
1.18	Display memory fragmentation	13
1.19	Display CPU idle rate	13
1.20	Show a grid in the displays	14
1.21	Control the measurement intervals	15
1.22	Changing the window dimensions	16
1.23	Public Screen selection	16
1.24	Font Selection	17
1.25	Graph colors	17
1.26	Window without title bar	18
1.27	How memory fragmentation is computed	18
1.28	How CPU idle time is measured	20
1.29	Known bugs	20

1.30 Future developments	20
1.31 Version history	21
1.32 The author	21
1.33 GSM Info Page	22

Chapter 1

gsm

1.1 Contents

GSM - Graphic System Monitor V 1.3

GSM shows a graphical display of memory usage, memory fragmentation and CPU usage.

Table of Contents:

Introduction

Copyright and Disclaimer

System Requirements

Installation

Use from Workbench

Use from Shell

Tool Types

Features

Menus

Bugs

Future

History

Author

GSM Info Page

Quick Start:

Follow the Quick Start links if you want to use GSM quickly without reading all of the guide.

Quick Start Link:

Installation

1.2 Copyright and Disclaimer

GSM - Graphic System Monitor V 1.3

GSM is Copyright (c) 1995
Yvon Rozijn

All rights reserved.

Copyright notice:

This program is freeware.

You may use this program without charge.

Copying, inclusion in public domain collections (including CD-ROM collections), redistribution (also via the Internet) is allowed provided that the following conditions are met:

1. Redistributions must include the unmodified program and all its documentation and other files.
2. Redistributions must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation or other materials provided with the distribution.
3. The redistributor is not allowed to charge more than a reasonable small fee for collecting the material, the actual copying and/or the information carrier used for the redistribution.
4. No part of this program or the documentation may be included in any commercial product without previous written permission from the author.

DISCLAIMER:

This software is provided "as is". You use it completely on your own risk. Yvon Rozijn shall not be liable for any direct or indirect damage arising in any way out of the use of this software, even if advised of the possibility of such damage.

1.3 Introduction to GSM

Introduction to GSM

The Graphic System Monitor (GSM) is a system monitoring tool that shows graphically:

- o Memory usage (both CHIP and FAST)
- o Memory fragmentation
- o CPU idle rate

I built GSM because I felt that there was no tool available that had all of these options. There are many free memory gauges, but these are just like fuel gauges, and don't produce the nice "heartbeat" like graph I wanted. Some tools do make such a graph, but they don't show all options at the same time.

Some tools measure CPU load or idle rate, but many do so in a terrible inaccurate way (counting the number of READY tasks).

Very few tools show memory fragmentation, and those who do, present the information in textual form, not graphically.

That's why I decided to build my own tool that shows everything I want to see the way I want it to see.

GSM was inspired on the very old PerfMon tool by Dale Luck, but it has some extra options. Besides, the PerfMon tool was developed for WB 1.x, and looks ugly on a 2.x/3.x Workbench screen.

1.4 System Requirements

System Requirements

To successfully use GSM, you need:

- o An Amiga with Kickstart/Workbench 2.0 or later
- o A CPU and some memory ;-)

1.5 Installing GSM

Installing GSM

Well, there is not much to say about this.

Just copy the executable and this guide (and the icons) to where you want them to be.

That's it.

GSM doesn't need third-party libraries, and doesn't need any Assigns.

Quick Start Links:

Use from Workbench

Use from Shell

1.6 Use GSM from Workbench

Use GSM from Workbench

GSM can be started from the Workbench like any other program: just double-click on its icon.

You can set several
 Tool Types
 using the Workbench information
 menu choice.

Alternatively, GSM can be started from the
 Shell

.

Quick Start Link:
 Features

1.7 Use GSM from the Shell

Use GSM from the Shell

GSM can be started from the Shell (or CLI).

Format: GSM [CHIP] [FAST] [CPU] [FRAG] [NOFRAG] [GRID] [NOGRID]
 [DELAY delay] [WINDOW window_spec]
 [PUBSCREEN screen_name]
 [FONT font_name] [FONTH font_height]
 [BGPEN pen] [FG1PEN pen] [FG2PEN pen]
 [GRIDPEN pen] [TITLEBAR yes_or_no]

Template: GSM CHIP/S,FAST/S,CPU/S,FRAG/S,NOFRAG/S,GRID/S,NOGRID/S,
 DELAY/N/K,WINDOW/K,PUBSCREEN/K,FONT/K,FONTH/N/K,
 BGPEN/N/K,FG1PEN/N/K,FG2PEN/N/K,GRIDPEN/N/K,TITLEBAR/K

When started from the shell, GSM tries to read its Workbench icon file. It then uses the

 Tool Types
 in that file for its default configuration.

The command line arguments can be used to override the Tool Type settings.

GSM supports the following command line arguments:

 CHIP

FAST

CPU

If any of these three are present, they override all three of the corresponding Tool Types

.

NOFRAG

NOGRID

DELAY number_of_milliseconds

WINDOW left/top/width/height

PUBSCREEN Public_Screen_Name

FONT=fontname.font

FONTH=font_height

BGPEN=pen_number

FG1PEN=pen_number

FG2PEN=pen_number

GRIDPEN=pen_number

TITLEBAR=yes_or_no

The meaning of these arguments are the same as the corresponding

Tool Types

FRAG

Can be used to override a NOFRAG Tool Type setting.

GRID

Can be used to override a NOGRID Tool Type setting.

Alternatively, GSM can be started from the

Workbench

.

Quick Start Link:

Features

1.8 GSM's ToolTypes

GSM's ToolTypes

GSM can be configured using the Tool Types in its Workbench icon file. Use the Workbench "Information" menu item to change them, or select the GSM

Save Settings

menu item to save the current configuration.

The Workbench icon file is also read for the configuration when GSM was started from the shell.

GSM supports the following ToolTypes:

CHIP

Display

```
        Chip memory usage
        .

FAST
    Display
        Fast memory usage
        .

CPU
    Display
        CPU idle rate
        .
```

If neither of the CHIP, FAST and CPU tooltypes are present, all three are displayed. This is because it is rather silly to start GSM if you don't want to see any of its displays.

```
NOFRAG
    Do not display memory
        fragmentation
        .
```

```
NOGRID
    Do not display a
        grid
        .
```

```
DELAY=number_of_milliseconds
    Set the
        delay
        between display updates. The minimum delay
        is 250 ms, the maximum is 5000 ms (5 seconds).
        Default delay is 1 second.
```

```
WINDOW=left/top/width/height
    Set the initial
        window dimensions
        .
    Default window dimensions are 0/0/260/120.
```

```
PUBSCREEN=Public Screen Name
    The name of the
        screen
        GSM should open its window on.
    Default screen is the current default public screen (usually
    the Workbench screen).
```

```
FONT=fontname.font
FONTH=font height
    The
        font
        name and height to use for displaying text.
    Default is the system's default font.
```

```
BGPEN=pen number
FG1PEN=pen number
FG2PEN=pen number
```

GRIDPEN=pen number

The

pen numbers

to use for drawing the graphs.

Default are the screen's background, text, highlight and fill pens, respectively.

TITLEBAR=yes_or_no

Open the window with or without

title bar

.

Default is to open with title bar.

GSM's tooltypes are not case-sensitive.

1.9 GSM Features

GSM Features

System information:

- o Amount of free
 - chip memory
 - and optional
 - fragmentation
 - rate
- o Amount of free
 - fast memory
 - and optional
 - fragmentation
 - rate
- o
 - CPU
 - idle rate
- * o
 - Difference
 - in free memory since some settable starting point

Other features:

- o User control of the
 - speed
 - at which the graphs proceed
- o Optional
 - grid
 - in the graph displays
- o Fully resizable
 - window
 - o Memory
 - flush
 - option
- o Non-modal
 - About
 - requester
- * o User settable
 - colors

- for the graphs
 - * o User settable
 - font
 - to use for text
 - * o Window opens on any public screen
 - * o Window opens with or without title bar
 - * o Settings can be saved
 - in the Workbench icon file
- * These features are new for version 1.3
-

Quick Start:

There are no more Quick Start links.
Click on any of the above to see more about that feature of GSM,
or go back to
Installation
.

1.10 GSM Menus

GSM Menus

The GSM Menus are layed out in the following way:

Project

- Flush
- Flush & Remember
-
- Save Settings
-
- About...
- Quit
- Display
- Chip Memory
- Fast Memory
- Fragmentation
- CPU Idle

```
-----  
Grid  
-----  
Title Bar  
Click on an item to see more.
```

1.11 Flush Memory

Flush Memory

DESCRIPTION

Flush memory from within GSM.

All disk-based libraries, devices, fonts, etc that are no longer in use are removed from memory.

MENU SELECTION

Project / Flush

KEYBOARD SELECTION

Right-Amiga F

1.12 Flush Memory and Remember

Flush Memory and Remember

DESCRIPTION

Memory is flushed like the
flush
function.

The resulting amount of free memory is remembered, and the difference between the remembered and actual amounts of free memory is displayed in the

Chip
and
Fast
memory graphs.

Note that Intuition may allocate memory for its IDCMP messages while a window is open. Just clicking in a window may result in allocation of some memory, as is any other Intuition event. This memory will not be freed by Intuition until the window is closed.

MENU SELECTION

Project / Flush & Remember

KEYBOARD SELECTION

Right-Amiga R

1.13 Save Settings

Save Settings

DESCRIPTION

The current configuration of GSM is written to its Workbench icon file as

Tool Type
values.

If GSM cannot find its icon file, nothing is saved.

MENU SELECTION

Project / Save Settings

1.14 About GSM

About GSM

DESCRIPTION

Shows a requester with version number and copyright information.

Unlike the About requesters of virtually all other programs, this is a non-modal system requester. Non-modal means, it does not suspend the operation of the program.

If there is already a GSM about requester hanging around on the screen, that window will pop up to front and is made active.

MENU SELECTION

Project / About...

1.15 Quit GSM

Quit GSM

DESCRIPTION

Terminates GSM.

If an

about
requester is still open, that window is

closed.

MENU SELECTION

Project / Quit

KEYBOARD SELECTION

Right-Amiga Q

SHELL SELECTION

GSM listens to Control-C signals and the BREAK shell command if started from a

shell

.

1.16 Display free Chip memory

Display free Chip memory

DESCRIPTION

Shows a graph of free chip memory in your system. The amount of free chip memory is also displayed as text, as is the difference from a previously

remembered
amount.

Optionally the

fragmentation
rate of chip memory is displayed

in the same graph using a second color.

Beneath the graph is a comment line reading:

CHIP (aaaaaK) bbbbbbK free (+cccccc) Frag: ddd%

where:

aaaaa = total amount of chip memory in the system in Kbytes.

bbbbbb = current amount of free chip memory in Kbytes

cccccc = difference between the current amount of free chip memory and the remembered amount in bytes. Positive values mean there is more free memory, negative values mean there is less free memory.

ddd = chip memory fragmentation rate.

This line might appear slightly different depending on available room in the window and options selected.

The

speed
at which the graph proceeds is user controllable,

as are the

colors
in which it is drawn.

MENU SELECTION

Display / Chip Memory

This menu item toggles the chip memory display.

WORKBENCH TOOLTYPE

Use the CHIP

tooltype
to get the chip memory display.

SHELL ARGUMENT

Use the CHIP
 argument
 to get the chip memory display.

1.17 Display free Fast memory

Display free Fast memory

DESCRIPTION

Shows a graph of free fast memory in your system. The amount of free fast memory is also displayed as text, as is the difference from a previously

 remembered
 amount.

Optionally the
 fragmentation
 rate of fast memory is displayed
 in the same graph using a second color.

Beneath the graph is a comment line reading:

FAST (aaaaaK) bbbbbbK free (+ccccc) Frag: ddd%

where:

aaaaa = total amount of fast memory in the system in Kbytes.
 bbbbbb = current amount of free fast memory in Kbytes
 cccccc = difference between the current amount of free fast
 memory and the remembered amount in bytes. Positive values mean
 there is more free memory, negative values mean there is less free
 memory.

ddd = fast memory fragmentation rate.

This line might appear slightly different depending on available room in the window and options selected.

The
 speed
 at which the graph proceeds is user controllable,
 as are the
 colors
 in which it is drawn.

MENU SELECTION

Display / Fast Memory
 This menu item toggles the fast memory display.

WORKBENCH TOOLTYPE

Use the FAST
 tooltype
 to get the fast memory display.

SHELL ARGUMENT

Use the FAST
 argument
 to get the fast memory display.

1.18 Display memory fragmentation

Display memory fragmentation

DESCRIPTION

Shows a graph of chip and fast memory fragmentation rate.

The graph is displayed in the free

chip

and

fast

memory displays,

using a second color to make it distinguishable from the free memory graph.

Fragmentation is not displayed if the corresponding memory display is not selected.

Fragmentation is computed as a percentage:

0% means that all free memory is in one (large) block

100% means that all free memory is scattered around in many little pieces

MENU SELECTION

Display / Fragmentation

This menu item toggles the fragmentation graph in the free memory displays.

WORKBENCH TOOLTYPE

Use the NOFRAG

tooltype

to prevent the fragmentation display.

By default the fragmentation rate is displayed.

SHELL ARGUMENT

Use the NOFRAG

argument

to prevent the fragmentation display, or the

FRAG

argument

to override a NOFRAG

tooltype

By default the fragmentation rate is displayed.

SEE ALSO

How fragmentation is computed

1.19 Display CPU idle rate

Display CPU idle rate

DESCRIPTION

Shows a graph of the average CPU idle rate.

CPU idle rate is computed as a percentage of the (real) time that the CPU is not busy with any task, averaged over the measurement time interval.

0% means the CPU was busy all the time

100% means the CPU was not busy at all.

In practice, 100% will never be reached, because there are always several background system tasks that consume some CPU time, including GSM itself.

Beneath the graph is a comment line reading:

```
CPU Idle aaa%
```

where:

aaa = current CPU idle rate.

This line might appear slightly different depending on available room in the window.

The

speed

at which the graph proceeds is user controllable,

as are the

colors

in which it is drawn.

MENU SELECTION

Display / CPU Idle

This menu item toggles the CPU idle rate display.

WORKBENCH TOOLTYPE

Use the CPU

tooltype

to get the CPU idle rate display.

SHELL ARGUMENT

Use the CPU

argument

to get the CPU idle rate display.

SEE ALSO

How CPU idle time is measured

1.20 Show a grid in the displays

Show a grid in the displays

DESCRIPTION

Shows a grid in the

free chip memory

,
 free fast memory
 and
 CPU idle rate
 displays.

The grid is displayed in the selected
 grid color

The grid consists of horizontal lines at 25%, 50% and 75%, and vertical
 lines after every 15 seconds.

After every minute, the vertical line is drawn as a solid line.

If the
 window height
 is too small, the 25% and 75% horizontal
 lines are not drawn.

MENU SELECTION

Display / Grid
 This menu item toggles the grid in the displays on and off.

WORKBENCH TOOLTYPE

Use the NOGRID
 tooltype
 to turn the grid off.
 By default the grid is turned on.

SHELL ARGUMENT

Use the NOGRID
 argument
 to turn the grid off, or the GRID
 argument
 to override a NOGRID
 tooltype

By default the grid is turned on.

1.21 Control the measurement intervals

Control the measurement intervals

DESCRIPTION

Controls the time to elapse between successive system analyses. This
 effects the speed at which the graphs are drawn.

The intervals can vary between 250 ms (0.25 seconds) and 5000 ms
 (5 seconds).

MOUSE SELECTION

Use the slider in the right window border to change the interval
 delay.

WORKBENCH TOOLTYPE

Use the DELAY=milliseconds
tooltype
to set the initial interval delay.

SHELL ARGUMENT

Use the DELAY milliseconds
argument
to set the initial interval delay.

1.22 Changing the window dimensions

Changing the window dimensions

DESCRIPTION

The GSM window is fully resizable. If resized, the displays are resized too, to reflect the new window size.

MOUSE SELECTION

Use the standard window drag bar, sizing gadget and zoom gadget to move or resize the window.

WORKBENCH TOOLTYPE

Use the WINDOW=left/top/width/height
tooltype
to set the initial
window dimensions.

SHELL ARGUMENT

Use the WINDOW left/top/width/height
argument
to set the initial
window dimensions.

1.23 Public Screen selection

Public Screen selection

DESCRIPTION

Allows the GSM window to open on any public screen.
If the selected screen cannot be found, the default public screen (usually the Workbench) is used.

WORKBENCH TOOLTYPE

Use the PUBSCREEN=Public Screen Name
tooltype
to select the screen.

SHELL ARGUMENT

Use the PUBSCREEN Public_Screen_Name

argument
to select the screen.

1.24 Font Selection

Font Selection

DESCRIPTION

Select the font to be used for text displays. Both fixed and proportional fonts may be used.
If the font cannot be opened, the system's default font is used.

I haven't included an ASL font requester in the program. Why not? Because the ASL requesters are modal requesters, they stop the program if they are opened. And a program like GSM should continue to operate, even if requesters are open.

WORKBENCH TOOLTYPE

Use the FONT=fontname.font and FONTH=font height
tooltypes
to set the
font to be used. The font name should include the ".font" extension.
Both a FONT and a FONTH must be present to use the font.

SHELL ARGUMENT

Use the FONT=fontname.font and FONTH=font height
arguments
to set the
font to be used. See the notes in the previous paragraph.

1.25 Graph colors

Graph colors

DESCRIPTION

Pick the pen numbers to use when drawing graphs.

Four colors are recognized:

BGPEN = background of graph.

FG1PEN = foreground 1, used for the first function line in a graph.

FG2PEN = foreground 2, used for the second function line in a graph.

GRIDPEN = grid color.

Sharable pens can be used, but they will not be properly obtained by GSM.

KEYBOARD SELECTION

Use function keys F1 to F4 to cycle through the available pens. Combine with the SHIFT key to cycle backwards.

F1 = cycle BGPEN

```
F2 = cycle FG1PEN
F3 = cycle FG2PEN
F4 = cycle GRIDPEN
```

WORKBENCH TOOLTYPE

Use the BGPEN=pen number, FG1PEN=pen number, FG2PEN=pen number and GRIDPEN=pen number
tooltypes
to set the colors.

SHELL ARGUMENT

Use the BGPEN pen_number, FG1PEN pen_number, FG2PEN pen_number and GRIDPEN pen_number
arguments
to set the colors.

1.26 Window without title bar

Window without title bar

DESCRIPTION

By default, GSM opens a normal window with title bar.
As an option, the window can be without title bar, and close, zoom and depth gadgets.

MENU SELECTION

Display / Title Bar
This menu item toggles the title bar.

WORKBENCH TOOLTYPE

Use the TITLEBAR=yes_or_no
tooltype
to control the title bar.

Valid tool type values are:

NO or 0: open window without title bar
any other value: open window with title bar.

SHELL ARGUMENT

Use the TITLEBAR yes_or_no
argument
to control the title bar.
See the previous paragraph for valid values.

1.27 How memory fragmentation is computed

How memory fragmentation is computed

What is memory fragmentation?

If free memory is not available as one large chunk, but merely as many smaller chunks, memory is fragmented.

Memory fragmentation can occur when programs have allocated many separate pieces of memory, and have not yet returned all these pieces to the system.

When memory is returned, the Exec will mend adjacent pieces of free memory together into a larger block, thus decreasing memory fragmentation.

If memory is fragmented, a memory allocation of a large block could fail, although the total amount of free memory could be sufficient. Allocation fails because the free memory is subdivided into several parts, each not sufficiently large to honour the allocation request.

For example: if there are 2 chunks of 512 kB free memory (total 1MB), and a program requests 513 kB, the allocation will fail.

It is obvious that memory fragmentation is a bad thing.

How is the percentage that GSM shows, computed?

The most important factor is the ratio

$$\frac{\text{size of largest chunk}}{\text{total free memory size}} \quad [1]$$

Fragmentation increases if this ratio is smaller. To scale it into [0,1], the factor

$$\left(1 - \frac{\text{size of largest chunk}}{\text{total free memory size}} \right) \quad [2]$$

is computed. This factor will be zero if all free memory is in one large chunk, and approaches 1 if free memory is scattered into many small chunks.

The second important factor is the number of free memory chunks. This is intuitively a measure of fragmentation.

Moreover, if there are more free memory chunks, there is a bigger chance that any freed memory will mend up with a chunk that is not the largest chunk, thereby decreasing factor [1].

This leads to a second factor, scaled into [0,1]:

$$\left(1 - \frac{1}{\text{number of free chunks}} \right) \quad [3]$$

The fragmentation percentage as displayed by GSM is the product of [2] and [3].

This way of expressing memory fragmentation as a percentage is undoubtedly arbitrary, as is every other way.

If you think you have a better way of computing this percentage, please send

me
an E-mail.

1.28 How CPU idle time is measured

How CPU idle time is measured

When the CPU is idle, there is currently no task to be executed. So how can you measure this time if nothing is running?

GSM takes advantage of a property of the operating system, namely, that a task with lower priority will never be executed as long as there are tasks with higher priority ready to run.

To measure the CPU idle time, GSM starts a dummy task at the lowest possible priority, -128. This task will get the CPU only if nothing else wants to be run.

Using special Switch and Launch routines, this dummy task reads the system clock every time it is launched (execution starts) and switched (execution is suspended).

The total time for which this dummy task has run in a specific interval, is the elapsed time for which no other task was ready to run, i.e. the CPU idle time.

GSM uses E-clock values for maximum accuracy. In fact, the idle time measurement is so accurate you can notice the CPU usage when you move the mouse, for instance.

It is obvious that GSM's strategy doesn't work if there are other tasks in the system running at -128 priority. In particular, if you run GSM more than once at the same time, the CPU idle rate displayed is not accurate. Also, some screen savers (like the standard Workbench screensaver) operate on this priority.

1.29 Known bugs

Known bugs

Currently, there are no known bugs :-)

If you think you found a bug, please let
me
know.

An actualized list of known bugs in the current version is kept on the

GSM Info Page
on the World Wide Web.

1.30 Future developments

Future developments

Possible enhancements of this program are:

- "Memory map" type display visualizing where the free memory is left.
- Turn GSM into a commodity.

If you feel something else is missing, let
me
know and I might build it
in.

An actualized list of possible future enhancements is kept on the

GSM Info Page
on the World Wide Web.

1.31 Version history

Version history

Date	Version	Comment
------	---------	---------

16-11-1995	1.3	Several features added: <ul style="list-style-type: none">o Read WB icon file if started from shello Save Settings in WB icon fileo Open on any public screeno Optionally open without title baro Font and colors can be seto Remember function addedo Listens to Ctrl-C break
13-11-1995	1.2	Bug fix: Crashed if no fast memory was present.
24-10-1995	1.1	Bug fix: Due to scaling factors, wrong results were displayed if total memory was > 4 MB. Fixed.
12-10-1995	1.0	Initial release.

1.32 The author

The author

This program was written by:

Yvon Rozijn
Zuideinde 9
7941 GA Meppel
The Netherlands

If you have any comments, questions, bug reports, ideas for improvement,
please send me an E-mail. My address is:

yrozijn@xs4all.nl

You can also visit my home page:

<http://www.xs4all.nl/~yrozijn>

1.33 GSM Info Page

GSM Info Page

There is a GSM Info Page on the World Wide Web:

<http://www.xs4all.nl/~yrozijn/gsm.html>

On this page you will find the latest info on GSM, like:

- o Version number of the latest version
 - o Actualized list of known bugs
 - o Actualized list of possible future enhancements
-