

ResGauge v2.1

January 27, 1994

Richard Franklin Albury

Songdog Software

1124 Toppe Ridge Court

Raleigh, NC 27615-6039

CompuServe ID: 76477,534

Internet: 76477,534@compuserve.com

America Online: Developer

What is ResGauge?

ResGauge is a simple, nonintrusive little utility which monitors free system resources in Windows 3.1. **ResGauge** can easily be configured to alert you when free system resources drop too low, permitting you to save your work before problems occur. Best of all, **ResGauge** is free, relieving you of the annoyance of nag screens and the guilt of unregistered shareware.

What are free system resources anyway?

I'm glad you asked that. The core of Windows is implemented in three distinct modules: the system kernel, the GDI (Graphics Device Interface) module, and the User (interface) module. The GDI and User modules each have a local heap, an area of memory in which they allocate memory for their own use. Since a local heap can't exceed 64 KB, each module is limited in the amount of memory it can allocate for its own use. This gives rise to the notion of free system resources, usually expressed as the unallocated percentage of the GDI or User modules' local heap. The number you see in the **About Program Manager...** dialog is the smaller of these two percentages, and this number will change as programs load and free menus, icons, and other interface resources. Not surprisingly, **Program Manager** in Windows 3.0 was one of the worst offenders in exhausting free system resources: every time you opened a group window, **Program Manager** would load the program icons but it wouldn't free them when the window was subsequently closed. This behavior has been fixed in Windows 3.1.

How does it work?

ResGauge has been designed to work like **Clock** in Windows 3.1. You can iconize the program and still get information, you can hide the title bar for a cleaner look, you can set the colors, etc. Unlike **Clock**, however, you can run more than one copy of **ResGauge**. For this reason, **ResGauge** doesn't save its window location. Otherwise, each new copy you started would come up over any existing copies.

What's on the system menu?

- **Always on top** allows you to keep **ResGauge** "floating" above other windows, much like **Clock** does. This is most useful when **ResGauge** is iconized.

What's on the Settings menu?

- **Monitor System/GDI/User heap** allows you select which heap you want **ResGauge** to monitor.

The **Monitor System heap** setting reports the smaller of the two values reported for the GDI and User heaps. This is also the same number reported by **Program Manager**.

- **No Title** removes the menu bar and the title bar from the window. Double-clicking on the window or pressing the Escape key allows you to toggle between states (this behavior was slavishly copied from **Clock**). With the title bar gone, mouse users can still size the window, and the window can be moved by dragging the window contents. Keyboard users, however, will have to hit Escape to get the title bar back and use the system menu to size or move the window.
- **Set alarm...** brings up a dialog for setting alarm options. The threshold value - the value at or below which the alarm goes off - can be set as high as 99% or as low as 1%, but the default value of 10% seems a good number. **ResGauge** can be set to beep and/or flash when the alarm is triggered, but if this annoys you, you can disable it at your own risk.
- **Set color...** brings up a dialog for setting the gauge color for the heap you're currently monitoring. If the default gauge colors - pure red for System, pure green for GDI, and pure blue for User - don't appeal to you, you can set them to whatever colors you like.

What's on the Help menu?

- **About...** is the usual shameless blurb.

How (and why) do I run more than one copy of ResGauge?

ResGauge will either read its configuration file when it starts or use default settings if the file doesn't exist. This scheme obviously doesn't lend itself to running different configurations simultaneously, so version 1.3 of **ResGauge** added support for command line arguments. Starting with version 2.0 of **ResGauge**, the command line support was simplified. **ResGauge** monitors the system heap by default, and command line arguments of **GDI** and **User** can be used to have **ResGauge** monitor the GDI heap or the User heap respectively. The case of the argument doesn't matter, but the spelling does, and an invalid argument will result in a warning, as will more than one argument.

How do I install ResGauge?

The installation procedure is extremely simple: copy the executable to a convenient location, create a program item - an icon, in English - for **ResGauge** in **Program Manager** (or whatever shell you use), and you're done. In my case, I have two icons of **ResGauge** in my Startup group, one for the GDI heap and one for the User heap. In most cases, however, a single icon in the Startup group is sufficient.

How do I get rid of it?

If you're really sure you don't like **ResGauge** - although I can't imagine why! ☺ - the "deinstallation" procedure is also extremely simple: delete the **Program Manager** icon, delete the **ResGauge** executable `resgauge.exe`, and delete the **ResGauge** initialization file `resgauge.ini` in your Windows directory.

Who helped you?

I wrote this program by myself. Honestly! I did have some helpful suggestions from the following folks, though:

- **Anthony W. Rairden** for the **USER/GDI/Both** options and the alarm threshold idea
- **Greg Saddler** for the **Always on top** option
- **Edward Bauman** for the color configuration idea
- **David Hoos** for the no-float/flash fix, some coding suggestions, and the correct spelling of "threshold" ☺
- **Larry LaBella** for displaying both USER and GDI data
- **Charlie Wathen** for putting the code back in to bring the window up when the alarm goes off
- **Juan M. Aguirregabiria** and **Julio Pons** in the January 1994 **Windows/DOS Developer's Journal** for their letter about how to drag a window by its contents

Aren't you done yet?

Almost. I'd like to say in closing that if you have any comments, kudos, complaints, or suggestions, I'd like to hear them. If you have any ideas for any other utilities you'd like to see, please let me know: I'd love to see them.

Version history

- 1.0** initial release
- 1.1** added configuration code, changed display
- 1.2** converted to medium model - still under 10 KB!
- 1.3** converted to Windows 3.1 STRICT compliance, added Windows-3.1 compliant "floating," made gauge configuration more informative, added color configuration, removed the restriction to a single instance, added command line support, and fixed the no-float/flash situation - all this in less than 12 KB!
- 2.0** made **ResGauge** behave like **Clock**, made the dialogs less fancy and more portable, and added version-checking code - all this in an additional 3 KB!
- 2.1** rewrote in C++ (no MFC, that's why it's so small!), cleaned up the painting of the "gauge," stripped out unnecessary code, put code back in to bring window to top when alarm goes off - now as small as version 1.3!