

## SAW Help Contents

### Commands



[The SAW Program Interface](#)  
[Keyboard shortcuts](#)  
[SAW Help Index](#)

### Glossary



[Foreword](#)  
[General Information](#)  
[System Summary](#)  
[SAW Errors/Events](#)

### Extra Information



[Bugs](#)  
[Extra Utilities](#)  
[Books](#)  
[Trademarks](#)  
[List of many acronyms](#)  
[Freeware Licence](#)

### Return to SAW



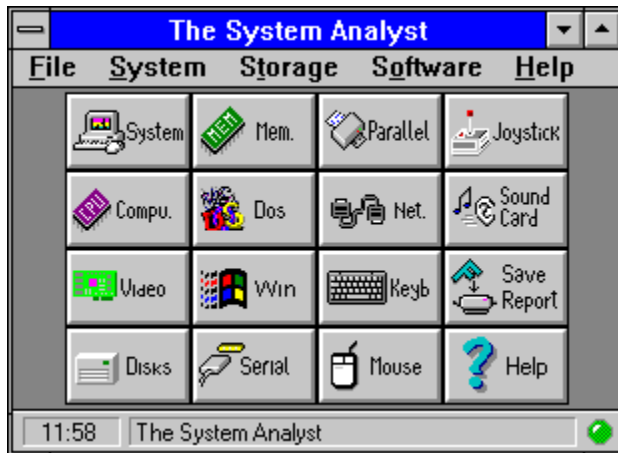
Click this button *on the Help toolbar* (or here) to return to (or start) SAW.

The **System Analyst for Windows ("SAW")**, Copyright © Catalin-Adrian Silasi, 1992-1995, **ALL RIGHTS RESERVED**. Freeware program. Freely distributable - subject to the rules in stated in the documentation (**Licence** file).

Please see the [Foreword](#) topic for more information.

***Enjoy SAW ! :)***

## The System Analyst for Windows (SAW) Interface



- Click on **any** area of the program (graphic above) for more information.
- Press the **TAB** key to highlight the part of the SAW window you want information on, and then press the **ENTER** key.

As you can see, SAW looks and behaves like any other Windows application - nothing different. If you don't recognise most of the elements featured you should read your Windows manual.

**Control Box**

Used to maximise / minimise / close the application. Also, there are a few other commands which may come in handy.

## **Maximise Box**

Used to make the application full screen. Do note that SAW can be made to look in any way - check ***File-Options-Button Options***.

**Minimise Box**

Used to minimise the application. SAW will continue monitoring your system even *minimised* - so you should keep it that way for monitoring. SAW will also inform you of Events / Errors / Warnings by visual and acoustic means.

## **Title Bar**

Displays the program name if nothing interesting is going on. However, if an Event occurs (error / system warning / some changes / problems) the error will be displayed and the title bar will flash to get your attention. Press **F1** to get on-line-help on the error / warning encountered.

## **Program Menu**

All the commands are available from the menu. Shortcut keys are also listed. Note that to prevent clutter some commands are only available from the menu !

## Hint Bar

As you move the cursor across the program button rows, the hint bar will display a summary of what will happen if you press that button. If the mouse is outside program window, the program name will be displayed. Use **Shift-F1** to get on-line help.

If **N/A** is displayed then you have made the program window too small for the text to fit the *Hint Bar*. Resize the horizontal length of the SAW window.



## **Toolbar (Button Rows)**

All main commands are available from buttons. Each has a picture on it and a name to give you a clue about what it does - check the Hint Bar as well. Just click (*left button*) them and see...

*Right button* brings on-line help.

Note that to prevent clutter some commands are available only from the Program Menu.

## The Clock

At the left hand side of the Hint Bar there is a little window which displays the current time or date. Click on it (*left button*) and it changes from one to the other. You can customise it to show the date or time by default. See ***File-Options***.

## **Help Button**

Press it to see the main contents / index. When an event / error occurs (the Title Bar is flashing) it will give you on-line information about the error encountered.

## Status Light

This resembles a *LED* which you can see on most electronic apparatuses today. It shows the system status in three colours.

- **Green** - All is well (as far as SAW knows) and no special events.
- **Yellow** - A event has taken place or a normal warning was encountered.
- **Red** - An error has occurred or a serious warning was encountered. Take action now !

**Note:** The **Yellow** and **Red** LEDs will flash. See Title Bar, SAW Errors/Events.

## Foreword

There are tens of million PC Compatibles world-wide, all due to the enormous popularity of the standard IBM PC. All these are supposed to be *100% compatible* to the "real thing". However, nowadays, as IBM is no longer setting the standard, there is no such thing as **true** IBM compatible; this term might mean, today, the most popular specification, i.e. a certain processor, graphics card, sound card, printer, etc. which is widely used - therefore supported by most software. A true PC compatible might mean the one (if available) that the software companies use - if it works on theirs it should work on yours, too. :)

Worse, there were and still are many types / versions of DOS or DOS emulators and now quite a few versions of Windows and Windows emulators. Also, there are 100's of thousand of software utilities- each doing its own thing. I'm not saying that everything should be the same - of course not - just want to put you in the picture. As you can deduce, it is very hard to make a program work on most systems. Just take a look at DOS games for example, and the problems of supporting different sound cards, joysticks, mice, video cards, memory managers, caches, etc.

Now Windows ! Windows has brought us the key to all this - your program is supposed to work on every machine if you stick to the operating system calls (API). (Other operating systems like Windows NT, OS/2, UNIX, Linux, etc. all do much the same thing or have done long before Windows; I will stick to Windows as SAW is targeted at this operating system) In a way this is true, most programs now work on systems under Windows that they would not have under DOS.

As far as SAW is concerned, or other diagnostic programs, the situations has not become that much better... Yes, it is easier to detect the characteristics of the system components (if the respective drivers are installed) but not always - the drivers have bugs themselves and sometimes tell lies for whatever reason. There are still holes in Windows, in the way that you still have to do much of the detection yourself. Worse, Windows is far cleverer than DOS and you can't get away with the things that some DOS programs do. As you still need DOS for many things, it gets a bit complicating to juggle virtual & protected mode, ring 0 & ring 3, VxDs & callgates and all other things...

Windows 95, as you may know, aims to replace DOS rather like Windows NT. Where will this lead is still unknown, but one thing is for sure - there will be fewer people who will know what is exactly going on.

What that all this mean ? Well, basically we all know that Windows is a complex environment - maybe more complex that it needs to be - and will become even more so. Can we help it ? In a way yes, if we keep it simple. That's exactly what I'm going to do - you won't find me writing any more detection programs... I'll stick to boring numerical computing :) :)...

## Enjoy SAW...

### Extra Information about SAW

Well, there is some information in this help file so maybe you can read the bits and pieces that interest you. Firstly, though, there are the **Licence & ReadMe** files which were supposed to be both installed by the **Install** Program.

#### Freeware

If you want a brief explanation about the Freeware concept and the SAW Licence, please read this topic.

In case you deleted the *ReadMe* file, here are some topics that are not covered somewhere else and might be important to you in case you have difficulties.

## Technical Support

Well, this is not exactly what is usually available - the program is free - but should you have problems with SAW and the bug / incompatibility is not listed, then you can contact me via **E-Mail**. I cannot offer you help and / or advice in other subjects though.

If the problem is known and fixed in the new version, then I might inform you about it (only during the discussion). I will not be able to get you a beta version of SAW - you will have to wait for the release version...

## Updates & Fixes

There are (there have been) updates every 3-4 months. Therefore, if you feel you want more information than SAW can provide, maybe there is help.

Please, **do not** nag me for releases every day ! When something arrives, you will know.

What should you expect ? Well, if this is not the first version of SAW, then you know what to expect. If this is your first, then...

## Where to contact Me...

I have tried to put as much information as I could in this file and the help file. I cannot (I'm not even trying to) explain many terms or results - I do presume you are familiar with what I'm talking about (Drivers, Utilities, etc.). Get some books and read some magazines if you want to know more...

If you **do** find bugs or incompatibilities (read the Bugs paragraph to see if it's a documented one) I would be very, very, very happy if you could E-Mail me with a brief description **and** SAW's report (complete if you can). If you have another diagnostic program (MSD, SI, Q&A) you could add it's side of story (a saved report) so I know what it should be... **Thanks...**

If a specific *Information Paragraph* causes the problem, uncheck it in the "*Include in the Report*" window and create a report without it. I don't need the full report if unavailable - send me what you can. As I said a MSD, SI, Check-It Pro, Q&A report is just as useful... **Thanks...**

Please do note that I may not be able to correct it - I may not know how/what to do !

You can contact me at the **E-Mail** address below:

---

**Internet (October to July mostly): ce146@city.ac.uk**  
(: *that's CITY University, London, UK by the way... :*)

---

**Note:** The subject of the message **should** begin with the word "**SAW**" followed by the version, e.g. "**SAW 1.10 : Detection Errors**"; otherwise I may not know what you talking about ! SAW is **not** always on my mind...

**Warning:** This is a *UNIX Sun Sparc* system (we use Pine), which may not understand "**fancy**" MS-Windows attachments, so please do not send any - include the reports (if any) in the main message ! If in doubt, ask ! No OLE objects in any case :)... A **full** SAW report may be over 100KB, so you may wish to compress it using ZIP or similar.

Sorry, if you need to contact me during other periods I may be very slow to answer as **I am away** (on holiday - lucky me)... I will still get your message... Also, I **do not** have Internet access during weekends... Finger me to find out what I'm doing...

Don't forget to say if I should reply or not ! If yes, then include the your correct *E-Mail* address to which I should reply to (if different from the one used to send the message). And **don't** flame me if SAW displays garbage instead of what it should...

#### **Extra Information (under construction):**

- The **latest** version of SAW can be found on **SimTel** (**/win3/sysutil**) and all its mirrors. This is the main site. As **CICA** (**/utils**) is always busy, I may not even upload it there. If someone can do it for me, I will be very grateful. Different versions may be available in different places. Use whichever you like.

- If you want to "**finger**" me, try **finger ce146@city.ac.uk -l | more** - hopefully you should find the latest news about SAW and all the other projects I'm working on...

- A **WWW** (World Wide Web - Mosaic, Netscape, Cello, etc.) page (a bit small) is available at **<http://www.city.ac.uk/~ce146/saw.html>**, which contains the latest information about SAW. It contains links to download the latest version of SAW from a few sites, and you can view the *finger* information as well. This will be updated regularly, but it still has a lot to go...

Beta versions may be available as well, although you are advised not to use them unless you really want to...

- If you do not have access to E-Mail but **cannot sleep** until you told me just how *bad* SAW is, you can send me a letter at the address below:

**C. A. Silasi (EEE Dept)  
City University  
Northampton Square  
London EC1V 0HB  
UK**

Heck, do not expect a reply, unless you enclose a SAE (self-addressed-envelope) too. I don't recommend it, though, as it does take a very, very, long time *to talk* using this link :), but if you really don't have a Internet connection this may be the only way. **Get connected soon !!!**

#### **To SAW's Regular Users**

If you use SAW on a regular basis you *should* send me E-Mail to inform me about it. No, you do not have to pay or register or something else. I just need to know what are you use it for, what features you use, which ones you like, etc. A SAW report would be your *donation* if you are kind enough...

No, this is not because I want SAW to become shareware. It *will* remain freeware. However, your message will make me continue its development (should you want it). Otherwise, I will abandon SAW in the future to focus on developing other programs much more interesting...

If you have sent me a SAW report previously, using an older version, then you could send me a new report so that I can see what has been fixed and what it has not. If I don't know then I may take a wrong turn and modify good code for no reason. I know its a pain, but what can I do ???

### **1.20 is it ! There will be no more !**

Yes, there is one more scheduled release. Therefore, if you have anything to add, now is your last

chance...





## Keyboard shortcuts in SAW


You can see the shortcuts listed in the program menu. Don't forget - clicking the buttons is easier. See the [SAW Program Interface](#) topic.

<b>F1, H</b>	-	Help Contents (or <i>Help on Error</i> ).
<b>F2</b>	-	System Summary Information.
<b>F3</b>	-	Computer Information.
<b>F4</b>	-	Video System Information.
<b>F5</b>	-	Windows Memory Information.
<b>F6</b>	-	Logical Drives Information.
<b>F7</b>	-	Windows Operating System Information.
<b>F8</b>	-	Network Information.
<b>F12</b>	-	Save Report.
<b>Alt-F4</b>	-	Exit (Standard Quit).
<b>Ctrl-Esc</b>	-	Switch To (Standard Switch).

## SAW Help Index

 • If you want to find information on a SAW menu option / topic press **F1** or click the **Help** button in the dialog box.

 • If you want to browse through all the SAW information screens, go to the [General Information](#) topic and press the **Forward Browse** button which will take you through all the main information screens.

 • If you are searching for a term, click the **Search** button and enter the required word in the box. If you've seen the term you have been looking for, click the **Show Topics** button, select a topic and click the **GoTo** button.

**The List** • If you are looking for an acronym or specific term and all the above has failed, try the [List of many acronyms](#) topic. Scroll the window until you find the term you are looking for. Should this fail too, a recent technical dictionary could help.

## General Information

To obtain information on a topic it is far easier to press the **Help** button in any window and the required topic will be shown. SAW has on-line Help like most other Windows applications (**Shift-F1**). You are advised to follow that route.

### Information Topics:

<a href="#"><u>System Information</u></a>	<a href="#"><u>Computer Info</u></a>	<a href="#"><u>Video Info</u></a>
<a href="#"><u>Logical Drives Info</u></a>	<a href="#"><u>Physical Drives Info</u></a>	<a href="#"><u>Windows Mem Info</u></a>
<a href="#"><u>DOS Memory Info</u></a>	<a href="#"><u>DOS Info</u></a>	<a href="#"><u>Windows Info</u></a>
<a href="#"><u>Serial Ports Info</u></a>	<a href="#"><u>Parallel Ports Info</u></a>	<a href="#"><u>Keyboard Info</u></a>
<a href="#"><u>Mouse Info</u></a>	<a href="#"><u>Network Info</u></a>	

### Program Topics:

<a href="#"><u>Uninstalling SAW</u></a>	<a href="#"><u>Integrity Check</u></a>	<a href="#"><u>Options</u></a>
---	--	--------------------------------

Use the **Browse** buttons - in SAW or Help - to browse through the various topics.

## System Summary

This window shows the computer's specification. If you don't want to get into details, this could be all you need...

All the information given here is presented into detail elsewhere - see the window that interests you. The name and icon of each button should give you a clue about the information that will be given if you click it - and don't forget the hint bar. Click the right mouse button for on-line Help. See the [Help Contents](#) topic for full details.

### Topics:

[Computer Type](#)  
[Main Processor](#)  
[Total Memory](#)  
[Video Adapter](#)  
[Serial Ports](#)

[Parallel Ports](#)  
[Sound Card](#)  
[Keyboard Type](#)  
[Mouse](#)  
[Floppy Disks](#)

[Hard Disks](#)  
[DOS System](#)  
[Windows System](#)  
[Network](#)

### Related Topics:

[Computer Info](#)  
[Physical Drives Info](#)  
[DOS Info](#)  
[Parallel Ports Info](#)  
[Network Info](#)

[Video Info](#)  
[Win Memory Info](#)  
[Windows Info](#)  
[Keyboard Info](#)  
[Joystick Info](#)

[Logical Drives Info](#)  
[DOS Memory Info](#)  
[Serial Ports Info](#)  
[Mouse Info](#)

Use the **Browse** buttons - in SAW or Help - to browse through the various topics.

## Computer Information

This window gives you information about the main part of your computer - the motherboard.

### Related topics:

[AMI Info](#)  
[Plug & Play](#)

[APM Info](#)  
[CMOS](#)

[The BIOS](#)

### Main Topics:

[Computer Type](#)  
[Computer Name](#)  
[Manufacturer](#)  
[IBM Class Type](#)  
[System BUS Type](#)

[BIOS Version](#)  
[BIOS Date](#)  
[BIOS ID. Number](#)  
[BIOS Ext. Area](#)  
[Main Processor](#)

[DMA Controller](#)  
[DMA Ch 3 Used](#)  
[Cascaded Irq. 2](#)  
[Real Time Clock](#)  
[Co-Processor](#)

## Computer Type

There are two main types:

- **PC/XT & Compatibles** (the first computers with an 8088/8086 processor).
- **PC/AT & Compatibles** - all new IBM compatible computers.

Also:

- **PS/2** - the "new" range of IBM computers.
- **PC Standard** - all PC/AT clone computers.

Nowadays, this information is somewhat superfluous as most computers are *PC/AT & Compatibles* especially as Windows, starting with version 3.1, *does not* run on XTs.

## Computer Name

This is the "name" of your computer if it has one. Usually only brand name computers (IBM, Compaq) have one. For clones you will see the name of the BIOS (or motherboard) manufacturer.

If a name is not found - **Unknown** will be displayed.

**Note:** The name might not be always found, depending on the machine.

## Computer Manufacturer

This should be the name of the computer manufacturer for brand name computers. For clones this will be the name of the system BIOS (OEM) manufacturer (usually the same as the computer name).

If a name is not found - **Unknown** will be displayed.

## IBM Class Type

This is the way computers were classified until a few years ago by IBM, but nowadays this is not so important. This is based on the BIOS ID. Number (see this topic) which was used to tell the machines apart using simple means.

Most modern clones will be classed as **IBM PC/AT or AT/386+** (BIOS ID No. **FC**) even if they have a 486 or even Pentium processor.

However, original IBM PCs should be correctly classified (as of 1995), but even IBM does not strictly follow this rule now...



## System BUS Type

The bus is a common connector made of *sockets* in which you plug the cards your computer needs. A computer usually has a graphics card, an input / output card, a disk controller card, etc. and the processor is interfaced to them by the bus.

There are now four main types of system buses detected by SAW:

- **ISA Bus** - Industry Standard Architecture - the original bus.
  - **XT Bus**, for XT computers 8088/8086
    - 8 data / 20 address bits (1MB addressable)
    - ~4.77MHz, asynchronous
    - 1.7MB/s burst thruput
    - Found in old computers now; many XT cards will work with the AT bus (see below)
  - **AT Bus**, for 286+ computers (ATs)
    - 8/16 data / 20 address bits (16MB addressable)
    - ~8MHz (usually overdriven up to 20MHz), asynchronous
    - 5.6MB/s burst thruput (max. 16MB/s if overdriven to 16MHz, 0WS)
    - bus master support
    - Until 1993 it was the mostly used bus, and we will still use ISA cards for a long, long time. The name says it all...
- **MCA Bus** - Micro Channel Architecture - used in expensive IBM PS/2
  - 16/32 data / 32 address bits (4GB addressable)
  - 80MB/s burst thruput, synchronous
  - full bus master support, bus arbitration, auto configurable
  - Runs faster, should have been the replacement of ISA for a long time but... Needs special MCA cards.
- **EISA Bus** - Enhanced ISA bus
  - 32 data/address bits (4GB addressable)
  - ~8MHz, synchronous
  - 32MB/s burst thruput (sustained)
  - full bus master support, bus arbitration, auto configurable
  - Designed by competition to beat MCA. It is also **compatible** with ISA, but needs special cards if you want its features. Used only in a few, expensive computers and network servers.

There are also *local* buses which connect the processor directly to only 2-4 cards to give much higher performance. In the past most were proprietary to the computer manufacturer and used custom designed cards; now there are two standards. The system bus is usually the ISA bus.

- **PCI Bus** - Peripheral Component Interconnect local bus
  - 32/64 data/address bits (4GB/16,777,216TB addressable - Wow!)
  - 33MHz, synchronous (1994), 50MHz+ (1995)
  - 132MB/s 32bit / 264MB/s 64bit burst thruput (sustained)
  - full bus master support, bus arbitration, auto configurable
  - limited to 3/4 cards
  - New (1993) and designed for *Pentium* processors although it has appeared in some 486s as well. 32bit / 64bit, fast (at last)...
- **VESA Bus** (VLB)
  - 32 data/address bits (4GB addressable)
  - 25-40MHz, asynchronous
  - 130MB/s burst thruput
  - bus master support
  - limited to 2/3 cards

- Typically connects graphics / disk controllers directly to the processor and it is used a lot nowadays (1993/94). You again need special VESA cards to take advantage of it, but an ISA card will work.

**Note:** Some computers have multiple busses, e.g. ISA/MCA, and some new ones even ISA/VESA/PCI busses. Therefore, they would be listed as: *ISA (16)+VESA (32)+PCI (32) bit*

## Main Processor

This lists the generic name of your computer's processor (the *brain* of your computer) as it is identified by Windows & SAW. This test does not determine the minor variations between processors.

Most new processors can "*tell*" their manufacturer. On this type of processors their manufacturer name will be displayed. The rest are classified as **Generic**.

As Windows 3.1 (and later) does not run on 8088, 8086, 80188, 80186 processors - support for them has been dropped.

As we go into the 32bit era, anything less than a 386 will be obsolete - both Windows NT and Windows 95 require at least a 386, preferably a 486DX2-66 or higher.

In brackets you can see the **Model** and **Stepping** number of your processor is known.

**Note:** If you run SAW under Windows NT or other Windows Emulator, your processor could be wrongly diagnosed. Emulators will show the processor they emulate - not the true processor.

## **Co-Processor**

This lists the generic name of your mathematical co-processor if it exists (is recognised by Windows). A co-processor executes floating point operations much faster than the processor and it is usually required for CAD / Mathematical work. Windows itself does not yet use it but it is good to have it.

The name is usually derived from your processor name, e.g. a 386 will have a 387 co-processor, etc., as it is usually the case. The Intel / AMD 486DX/DX2 and Intel Pentium™ have it built in; the equivalent co-processor will be shown.

## The BIOS

A BIOS or **Basic Input Output System**, is a software program built into your computer (in a ROM/EEPROM chip) which does the following:

- Is executed every time you start your computer. It checks the hardware (memory ,processor, disks, etc.) to make sure that everything is OK and tells you if it is not.
- Loads the operating system.
- Has built-in functions (routines - interrupts) the programs (even DOS and Windows) use to perform various operations.

The BIOS is therefore very important, as without it nothing will work. As it is built into a ROM chip you cannot change it as easily as upgrading DOS; you have to take out the old chip from the motherboard and put a new one (compatible !) - but, usually, this is never done as it is not really needed.

Some new computers have it in EEPROM and the BIOS can be updated via a floppy disk just like any other program - although these are still rare (1994). This is called a **Flash BIOS**. All new computers in 1995 should have one.

## Memory Chips

Memory chips are of two main types:

- **Volatile** - they lose their contents when the power is switched off.
- **Non-Volatile** - they keep their contents.

Your computer uses *volatile* memory or **RAM** (Random Access Memory) for programs. There are three types: DRAM, VRAM and SRAM.

The *BIOS* is kept in *non-volatile* memory. The type of memory can be: ROM, PROM, EPROM, EEPROM or something else.

**DRAM** (Dynamic RAM) Used by the computer and some graphics cards. Needs refresh otherwise it loses its contents. Cheap, big capacity but not very fast.

**VRAM** (Video RAM) Used by graphics cards to gain speed. Not faster than DRAM, also needs refresh, but can be accessed by the RAMDAC & Graphics Processor at the same time (i.e. read & written at the same time). Expensive.



**SRAM** (Static RAM) Very fast, usually used as a cache. Does not need refresh, but is very expensive, so only small (kilobytes) capacities are currently used. PCMCIA cards with MB sizes are available - but are no faster than DRAM.

**ROM** (Read Only Memory) - Is programmed by the manufacturer only and it dies with its program (Like Compact Disks). The first type of memory.

**PROM** (Programmable ROM) - Can be programmed with a device by a programmer but only once (Like recordable CDs nowadays).

**EPROM** (Erasable PROM) - Can be erased using ultra violet light - but you have to erase everything !  
Can be re-used again and again.

**EEPROM** (Electrical EPROM) - Can be erased electrically and you can erase only what you want. The best, but expensive.

## **System BIOS Version**

See [The BIOS](#) topic if you don't know what the BIOS is.

The version number of your system BIOS. This is good to know as some software have problems with older versions of certain BIOSes. You should find out what the newest version of your BIOS is if you have bought the computer for some time...

## System BIOS Date

See [The BIOS](#) topic if you don't know what the BIOS is.

The date when the system BIOS was manufactured or released by the manufacturer. This is important as you could buy a computer with a BIOS made in 1992 (in 1994) which may not support certain functions. Some software require a BIOS made after 1990/91 (in 1994).

Newer BIOSes have many more advanced functions which maximise performance or fix certain bugs (hopefully rare).

The date is given in mm/dd/yy (Month/Day/Year) - the American way - regardless of your Windows date set-up to avoid confusion. As most BIOSes are made in the USA, you'll have to use it this way...

## System BIOS ID. Number

See [The BIOS](#) topic if you don't know what the BIOS is.

This consists of 3 numbers, shown in hexadecimal format which are supposed to be a code which identifies your machine: Name, Type, Model even processor. See [IBM Class Type](#) topic.

Now, usually it is **FC -- --** ("- = don't care) for most clones, but IBM machines usually have unique numbers assigned to them. You can get a list of IDs and check it for yourself if you have an IBM computer and your machine is **Unknown**.

Read a book for more information - see [Books](#).



## System BIOS Extended Area

See [The BIOS](#) topic if you don't know what the BIOS is.

The Extended areas are add-ons for your system BIOS. Most computers have a graphics card which has its own BIOS (called **Video BIOS**), a network card which again may have its own BIOS, a [SCSI](#) Disk Controller or Cached Disk Controller with its own BIOS. All these BIOSes each have a unique address.

The Start Addresses (in hexadecimal) and the respective sizes (in kilo-bytes) are listed (e.g. **C000h 32KB**, **C800h 16KB**). It is not specified what each extension is, e.g. at C000 we might have a graphics card and at C800 a cached disk controller. See other main topics for details on this.

## DMA Controller

The DMA (Direct Memory Access) controller allows the peripherals to access main memory directly without CPU involvement. The CPU can do other tasks in the meantime. All modern computers have one.

The number of DMA channels varies between PC types:

- **PC/XT** - 4 8bit DMA channels.
- **PC/AT** - 4 8bit, 3 16bit DMA channels. (4th is cascaded)
- **PS/2 and modern ATs** - 8 8/16bit DMA channels.

Usually, they may be used by the following peripherals (0 - highest priority):

- **Ch 0** - Usually free for use by any peripheral. (Memory refresh on XTs)
- **Ch 1** - Usually *Sound Cards* use it and if you have one you should stick to it if you want to keep out of trouble with DOS games. Should other peripheral use it - configure the other peripheral if you play a lot of games...
- **Ch 2** - Floppy Disk (diskette) controllers use this, so don't touch it !
- **Ch 3** - Some SCSI hard disk controllers use it - See the DMA Channel 3 Used topic for further information. IDE drives do not use it. (Hard Disk on XTs). Also, some *Sound Cards* use it as default. Some PS/2 ESDI controllers do use it.
- **Ch 5** - Some SCSI hard disk controllers use it, and 16bit sound cards.

**Note:** Some variation may arise between different PC designs.

## DMA Channel 3 Used

See the [DMA Controller](#) topic first.

This tells you if some peripheral (usually a hard disk) uses the DMA Channel 3, so you know not to tell another peripheral to use it...

**Note:** *IBM PS/2s* with [ESDI](#) controllers are known to use DMA channel 3.

## Cascaded Interrupt No. 2

See [Hardware Interrupts](#) topic first.

PC/XTs have only 8 hardware interrupts while PC/ATs have 16. To make further interrupts available on ATs, 2 interrupt controllers were connected - one as master and the other as slave. Irq 0-7 are readily accessible while Irq 2 serves as *gateway* to access Irq 8-15. Therefore, only 15 interrupts are available - Irq 2 and Irq 9 being linked. You cannot use *both* Irq 2 and 9.

On ATs this should always be **Yes** ( Irq 2 Cascaded - Irq 9 Redirected).

## **Real Time Clock**

Early PCs did not have a real time clock - you had to enter the time every time you started your machine, and the clock was imprecise...

All new computers have a real time clock, though not as exact as you might think (depends how lucky you are...)

## AMI BIOS Information

If your computer has an **AMI BIOS** (made by **American Megatrends Inc.**) SAW will give you extra information based on the ID. string and the BIOS CMOS values. The button will be greyed if the information is not available.

### Topics:

<a href="#"><u>Required Processor</u></a>	<a href="#"><u>Chipset ID</u></a>	<a href="#"><u>Reference Number</u></a>
<a href="#"><u>Internal Version</u></a>	<a href="#"><u>Release Date</u></a>	<a href="#"><u>Data Size</u></a>
<a href="#"><u>Parity Check</u></a>	<a href="#"><u>HD 47 RAM Area</u></a>	<a href="#"><u>Weitek Processor</u></a>
<a href="#"><u>System Boot Seq.</u></a>	<a href="#"><u>External Cache</u></a>	<a href="#"><u>Internal Cache</u></a>
<a href="#"><u>Fast Gate A20</u></a>	<a href="#"><u>IDE Block Mode</u></a>	<a href="#"><u>IDE Standby Mode</u></a>
<a href="#"><u>Flash Ver</u></a>	<a href="#"><u>Flash Interface Ver</u></a>	

**Note:** SAW **is not** an AMI Configuration / Set-up program ! It only tells you about what I think are the most important BIOS settings. Refer to your AMI BIOS / Motherboard manual for more information.

**Note:** The ID. string is like a recommendation, and sometimes the information given might not be correct. This is because the motherboard manufacturer does not always take it into account.

## **AMI BIOS - Required Processor**

The processor for which the BIOS has been designed (and compatibles). Sometimes, a BIOS can be used with another processor - especially when a new processor family comes out before a specific BIOS is released or in other circumstances.

This may differ. See the Main Processor entry to find out what processor you really have.

### **AMI BIOS - Internal Version**

This lists the internal version of the BIOS. This should be the same as the normal version. It may differ if this is a (minor) revision or a small update of a major version. If no main version is found, then this might be your only indication of the BIOS Version. See the BIOS Version topic.

In brackets you can see the BIOS Modified Flag, incremented each time AMI BIOS is modified by the OEM manufacturer, from 1 to 9, then from A to Z, and then reset to 1. This shows how many motherboard revisions have been.

0 = AMI BIOS has not been modified.



### **AMI BIOS - Reference Number**

This is the reference number for your specific AMI BIOS. It is supposed to be sort of code or serial number as most things today have. Can identify your particular motherboard type & revision number.

### **AMI BIOS - Data Size**

The size of the System BIOS in kilobytes. Most are 64KB today (1994) but in future the BIOS may become bigger as our PC becomes more complicated. Watch this line...

**Note:** EISA systems have a 128KB BIOS or even 512KB.  
VL-BUS machines have a 32KB BIOS extension for VESA local bus.

## **AMI BIOS - Release Date**

The date when the System BIOS was released. Should be the same as the main date. If it is different, this may be the date at which the revision was released. If no main date is found (or garbage), then use this - is all you've got...

See BIOS Date topic.

The date is given in mm/dd/yy (Month/Day/Year) - the American way - regardless of your Windows date set-up to avoid confusion. As most BIOSes are made in USA, you'll have to use it this way...

### **AMI BIOS - Chipset ID.**

This is the ID of your motherboard Chipset. The processor is usually included in the name (e.g. **SYM\_486** - 486 compatible) and this should identify the motherboard maker / type (a bit obscure if you are not from the electronics industry).

Ex:     • **SYMXXXX**                 **Symphony**  
         • **OPTIXXX**                **OPTI**

You can find this the easy way by reading the documentation (if any) or looking inside at the two main chips beside your CPU - don't try it if you are unsure about it...

### **AMI BIOS - Memory Parity Error Checking**

This option enables error checking for all system RAM. If *disabled*, only the first 1MB of RAM is parity checked.

You should always have it *enabled*, unless you are using 8bit (Apple Macintosh) SIMMs which do not support parity checking (are 8bit instead of 9).

### **AMI BIOS - Hard Disk Type 47 RAM Area**

If your hard disk(s) is / are not one of the pre-defined (type 1-46), you will have to store the details in RAM (as well as CMOS of course). This area is either located at *0:300h* in low memory (preferred) or the top *1KB* of main memory (511 or 639KB).

You should set it to 0:300 as some bad behaved DOS programs may crash if they assume you have the top of the base memory free.

### **AMI BIOS - Weitek Co-Processor**

This option specifies that a Weitek co-processor (WTL3167 or WTL4147 for 386/486 motherboards which have a socket for it) is installed in the system. This uses some memory blocks (64KB - usually EFEF-EFFF) which must not be used by something else.

For the ones that don't know, this is a **very** fast math co-processor, but which is not compatible with 387/487 processors. Programs must be specifically written for it. Not much use now as the 486DX/2/4 have a not too bad one, and if fast processing is required a Pentium will do. Actually, you can use both a 387 and a WTL3167 at the same time...

### **AMI BIOS - System Boot Sequence**

This option specifies if the computer should first try to boot from, and if not available the second drive to check. For fast booting it should be **C:, A:**

You can use it to make sure no one tries to hack into your system by inserting a boot disk into the floppy and by-passing your fancy password log-in. Put a *set-up* password as well !



### **AMI BIOS - External Cache Memory**

This specifies if the second level cache should be enabled. It should always be enabled unless you blew it up or you don't have any !

If you don't have any, get some really fast...

### **AMI BIOS - Internal Cache Memory**

This specifies if the internal cache (for 486 and higher processors only !) should be enabled or not. This should always be enabled, otherwise you will get a lot less performance...

### **AMI BIOS - Fast Gate A20 Option**

This option enables the system's Fast Gate A20 circuitry. When running Windows (1994) (or any protected mode program) there is a lot of switching from *protected* to *real mode* (to DOS) and vice-versa; this was (is) accomplished by the keyboard controller which slows down processing.

Fast Gate A20 is another method of handling this, and is found in many chipsets. It is much faster, so it should always be *enabled* unless you've got a duff BIOS / motherboard.

### **AMI BIOS - IDE Block Transfer Mode**

This option appears in BIOS dated 08/08/93 (mm/dd/yy) or later. This option enables multiple sector reads / writes for (E)IDE drives. Also, the maximum number of sectors that can be transferred at a time may be specified.

The (E)IDE drive **must** support this function in order to be used ! *Enabling* it may increase data transfer rate up to +25% depending on benchmark :-)

### **AMI BIOS - IDE Standby Mode Support**

This option appears in BIOS dated 08/08/93 (mm/dd/yy) or later. This function spins down the (E)IDE hard disk drive (not *all* drives support this) after a specified amount of time (which you can set). This reduces the power consumption.

It should be *enabled* if you want power badly on a laptop, and the drive supports it. For desktops, its better to have the **Green** functions disabled until they work as expected (1994). In 1995, most BIOSes have *useable* Green functions.

## **AMI BIOS - Flash BIOS Version**

A **Flash BIOS** is not stored in ROM but rather in EEPROM so it can be modified (updated) using a special program. Older BIOSes can be updated only by replacing the chip itself - a fairly difficult operation. Also, the chips are very hard to find.

This lists the internal version of the flash BIOS. This should be the same as the normal version. It may differ if this is a (minor) revision or a small update of a major version. This may be a more accurate BIOS version. See the BIOS Version topic.

**Note:** This applies only to new AMI BIOSes which can be updated by using the BIOS update disk.

### **AMI BIOS - Flash Interface Version**

A **Flash BIOS** is not stored in ROM but rather in EEPROM so it can be modified (updated) using a special program. Older BIOSes can be updated only by replacing the chip itself - a fairly difficult operation. Also, the chips are very hard to find.

This lists the internal version of the flash interface the BIOS supports. This is required in order to interface with the set-up / update program. The latest version (start 1995) is **2.00**.

**Note:** This applies only to new AMI BIOSes which can be updated by using the BIOS update disk.

## Video Information

This window presents information about your Video System: the Graphics Card and the Windows Video Driver. Do not forget that the video driver is as important as the card itself ! Please choose one of the topics below.

### Topics:

<u>Card Name</u>	<u>Video BIOS Version</u>	<u>VESA VBE</u>
<u>Card Type</u>	<u>Video BIOS Date</u>	<u>Maximum Mode</u>
<u>Card Manufacturer</u>	<u>VESA Support</u>	<u>Current Mode</u>
<u>OEM Manufacturer</u>	<u>VESA OEM Name</u>	<u>Card Memory</u>
<u>Driver Name</u>	<u>Driver ID</u>	<u>Driver Version</u>

**Note:** The use of virtual screen programs may create differences between the true state of the Video System and the apparent one. The information refers (of course) to the true state.



## Video Card Name / Model

The name of your video card or model depending on the manufacturer. Some cards do not have a name, but a number as model (e.g. **2145**). This information is **very** model specific and the name might not be found in some cases.

Useful in identifying your card. Do note that the name of the driver itself, if not generic, lists the name of the card as well (e.g. **Cirrus Logic 2145 1024x768x256**). See [Driver ID](#) topic.

## Video Card Type

The generic type of your graphics card. Up to date there are a few general modes which in the past specified the resolution and colours as well. These are:

- **Monochrome** - Various resolutions - usually for portables / servers.
- **CGA** - Max. 640x200x16 colours. Not really useful in Windows nowadays.
- **EGA** - Max. 640x350x16 colours. Not really useful in Windows nowadays.
- **MCGA** - Max. 640x480x256 colours.
- **VGA** - Max. 640x480x16 million colours. The old standard Windows resolution.
- **SVGA** - 800x600x16 colours and above. Preferred resolutions 800x600x64K or 1024x768x256 for 1MB cards or 1024x768x64K for 2MB cards. The Standard resolution.
- **XGA** - IBM's high resolution naming - 800x600x16 and above. This is different from SVGA but offers much the same thing.

**Note:** If your card supports more than VGA (SVGA or XGA card), but you use the VGA driver, it will be classified as VGA and not otherwise.

## Video Card Manufacturer

The name of the Video BIOS Manufacturer - the one that actually made the card / BIOS (if correctly *signed*). Sometimes the **OEM Video Card Manufacturer** might be listed if the names are not in correct order. See the [Card OEM Manufacturer](#) topic.

This could be useful if the Video BIOS is *off-the-shelf* and not custom as you could gather additional information about it.

## Video Card OEM Manufacturer

The name of the true manufacturer of the video card - the one with the design. For example the *Video 7 Spea* cards have Video BIOSes made by *Phoenix Technologies*. Therefore, *Video 7 Spea* is the OEM Manufacturer and *Phoenix Technologies* the Card Manufacturer (Video BIOS).

(: In fact, the card mentioned above is designed / built by Intel ! :)

See the [Card Manufacturer](#) topic.

**Note:** Anyway, you can open up the computer and take a look at the card to see what it says on it...

## Video BIOS Version

See [The BIOS](#) topic if you don't know what a BIOS is.

The version of your Video BIOS. Also shown at the top of the screen when you start your computer (with the name, date, copyright notice etc. of your Video Card).

Useful to know. Some drivers require specific Video BIOS versions in order to run.

**Note:** If multiple versions are found (it happens many times) the OEM Manufacturer's version is shown. If no version is found then *Unknown* is displayed.

## Video BIOS Date

See [The BIOS](#) topic if you don't know what a BIOS is.

The date of your Video BIOS. Newer BIOSes have more advanced functions which maximise performance or fix certain bugs (hopefully rare). However, replacing the Video BIOS is very difficult compared to replacing your System BIOS - much harder to get...

The date is given in mm/dd/yy (Month/Day/Year) - the American way - regardless of your Windows date set-up to avoid confusion. As most BIOSes are made in USA, you'll have to use it this way...

## VESA Support

See [VESA BIOS](#) if you don't know what it is.

If your video card supports the VESA interface, this will be indicated with the version supported. Nowadays, this should be **1.02** or the new **2.00**.

**Note:** Your card does not have to have VESA support to run Windows - but if you want to use generic drivers, some require it.

## VESA OEM Name

See VESA BIOS if you don't know what it is.

The name of the OEM Manufacturer of the VESA BIOS (if your card supports the VESA interface) - usually the manufacturer of the card itself. Sometimes, the card model / name is also included or some other piece of information. Other manufacturers (software companies) may be listed as well.

If you have a TSR VESA emulator, then it may be generic - this information will relate to the manufacturer which wrote the emulator - not necessarily the card manufacturer.



## The VESA BIOS Interface

The **VESA BIOS** interface is a software interface for video detection capabilities. It can be placed in the Video BIOS or as an add-on TSR.

As above VGA (640x480x256) there is no real standard, every video card manufacturer of SVGA cards can do it in a different way - therefore you need a specific driver to access the card at higher resolutions. As this is not good at all, a specification has been introduced to enable SVGA standardisation - this is the VESA interface.

The VESA interface enables programs to get the video card details (modes supported, memory installed, etc.) in a much simpler way without using complicated detection routines, and also to access the card in the right way (which can be quite different for different cards).

Not all the video cards support it (even newer cards), but the ones that don't, have a TSR driver supplied that emulates it (see your documentation). If the Windows driver does not need it you can uninstall it thus saving memory. However, you will not be able to use Microsoft's generic SVGA drivers if your card does not emulate the VESA standard.

There are now three VESA versions:

- **VESA 1.00** VESA Standard #VS
- **VESA 1.01** VESA Standard #VS900602
- **VESA 1.02** VESA Standard #VS911022
- **VESA 2.00** VESA Standard #VS

**TSR** (Terminate and stay resident) is a program (or device driver) which *sits* in the background (you are not aware that it is running until you call it) to provide some services. In our context it is a BIOS extension, i.e. adds some functions your BIOS does not support.

## VESA VBE/PM

VESA VBE/PM is the VESA interface for Power Management for displays (monitors, laptop displays, etc.) and video cards. This enables the video cards to put the display circuitry in stand-by mode when only a fraction of the normal power is consumed (~**20-35%**).

To enable this, both the card and display (monitor, etc.) must support the VESA VBE/PM interface. The computer BIOS must support it too (Green BIOS).

The latest (start 1995) version is **1.00**.

**Note:** This concerns mostly new desktop computers which until recently lacked power management features. Usually portables use APM (Advanced Power Management) interfaces to minimise power consumption.

## Video Card Maximum Mode

The maximum mode your Video Card supports - i.e. the maximum resolution. This does not take into account that you might not be able to run it in this mode (no high resolution monitor, no driver, etc.).

For example you might run Windows in 640x480x16 when your card can go up to 1280x1024x16 or 1024x768x256 (depending on memory, BIOS version). SAW may display 1280x1024x16, but this does not mean you can do it - your monitor might not take it or your card does not support this mode even if it could. SAW does **not** do any **real** checking.

**Note:** This is a estimation only and might give erroneous readings, especially if you have a *fancy* card (very new or very advanced). You should consult your manual for exact information

## Video Card Current Mode

This gives two pieces of information:

- **Card Mode** (in hexadecimal). This has no real meaning to you, but the manual should list them so you can check the resolution. Two or more modes may have the same resolution, but different attributes, so this can tell them apart.
- **Card Resolution** (in brackets). This gives the actual resolution (e.g. 800x600) and the numbers of colours that your computer can display without dithering. These can be:
  - **Mono** - Monochrome display. (1bit ?)
  - **16** - Windows basic colours. (4bit)
  - **256** - Enhanced colour palette. (8bit)
  - **32K/64K** (32,768/65,536 colours - 15/16bit) - **High Colour**.
  - **16M** (16,777,216 colours - 24bit) - **True Colour**

**Note:** If you use a **Virtual Desktop**, SAW will show the apparent resolution, not the real resolution, but the current mode. For example, if the card is in 4Bh (800x600) mode, but your virtual desktop is 1024x768, you will see 4Bh (1024x768) - the resolution you are using.

**Note:** If you have a VGA monochrome laptop with 256 shades of grey (example) it may be classified as colour - depending on what the driver says...

## Video Card Memory

The amount of memory that your card is using now and the actual memory installed on the card. For example in 640x480x16 you use 150KB of memory only, but your card might have 1MB - the rest being unused (it may be used but for other purposes - depending on the driver). SAW will show: *150KB (1MB)*.

**Note:** If your card supports the VESA interface, then the memory shown to be on the card is as reported by the card itself (See VESA BIOS). Otherwise, SAW will round it up to what it thinks you might have - sometimes less than you actually have.

## Video Driver Name

The expanded (full) name of your Windows video driver. This is the expanded name which usually includes the resolution and, sometimes, the name of the card manufacturer. See [Driver ID.](#)

Nowadays, some manufacturers use a **generic** driver for a family of their cards (e.g. for TS859/ TS857 / TS456) if they use the same / compatible chipsets. Even the Video BIOS may be generic (in the same manner) so its harder to tell exactly which graphics processor you really have...

## Video Driver ID. & Version

This is the DOS filename of the Windows video driver (without extension). If you need to replace it - you will need this name. This is the actual name of the **.DRV** file in your **\windows\system** directory (if your Windows directory is *windows* - See Windows Information) being used.

Also, the version of your Windows video driver is shown.

This is usually the same as the Windows version itself if it a standard one (i.e. has come with Windows itself). If it came with your video card then it may be the specific file version of the driver.

If you can't get a updated version for your card from the manufacturer, check out the Microsoft generic drivers (available for most popular cards) from:

- The SVGA drivers that come with Windows itself or the other drivers included in the package (see manual and browse the drivers in Windows Set-up)
- *Bulletin Boards* - sorry, you have to find out for yourself. If you connect to USA from Europe it may cost you dearly if you don't have a fast modem...
- *FTP* access on *Internet* at **ftp.microsoft.com (198.105.232.1)** in directory **/SoftLib/MSLFiles**. If you live in the UK, connect to **src.doc.ic.ac.uk** instead (**/packages/microsoft** directory). You can also get fixes / answers to common questions for Windows in **/peropsys/...** (windows, wfw, mouse).

For all types of drivers (video, sound & printer) also try:

- FTP to **ftp.cica.indiana.edu (129.76.26.27)** in **/pub/pc/win3/drivers/video** [UK: **src.doc.ic.ac.uk** in **/packages/windows3/drivers/video**], **oak.oakland.edu**, **wuarchive.wustl.edu**, **SimTel** and other popular archive sites.

Keep those old drivers handy, just in case...



## **Video Driver Supported Version**

The version of Windows for which the driver was designed.

This should be the same as the Windows version itself. If it is smaller, you should get the new drivers (if available) as they should have a few improvements and usually they are more compatible (fewer crashes) - but this is not always the case.

In any case, check if you have the latest driver. See [Video Driver ID & Version](#) topic.

## Bugs (Argghh....)

Welcome to the BUGS !

Because BUGS appear and are fixed all the time I cannot list them here: I would have had to re-build the help file every day ! On a more serious note, I hope that there are not that many bugs in this version, but you may know better so I'll leave it to you...

The bugs are listed in the **ReadMe** file which should be in your SAW directory. If you deleted it, reinstall it (either from scratch or manually). This file is in Windows Write format.

One more thing. If you really find a **BUG** please **drop me a line** at the address listed in the **ReadMe** file and the Foreword topic. As I said, I may not be able to solve it but I'll sure try...

Thank you for using SAW. 'Hope you like it, otherwise I've worked for nothing...

## The Cache

A cache is memory used as a temporary store from slower memory or disk.

Newer processors have a small internal SRAM cache (like the 486 - 8KB; 486DX4 or Pentium 16KB) which speeds up performance and most motherboards also have a bigger external cache (64-512KB) to further boost performance.

All modern hard disks have internal cache (64-256KB). Cached hard disk controllers have their own processor and cache RAM (1-16MB) to speed up disk access. Also cache software (like *Smart Drive*) can use some of your memory (depending how generous you are) to work in the same way.

## How to UnInstall SAW

This procedure will uninstall the **System Analyst for Windows** from your system. The uninstall will work only if:

- **Local Uninstall:** The drive where SAW is installed is not **write protected** (especially if it is a floppy). Some virus checkers have an option of *locking* your programs drive (to prevent program infections) - turn it off.
- **Network Uninstall:** You **must** have **scan/delete/remove** access to the drive / directory where SAW is installed.
- The **install.exe** program, is in the directory from which the install was made. For example, if you installed from **A:\SAW**, you **must** insert that disk into the **A:** drive with the *install.exe* file in *a:\saw* ! You may also specify a path.
- You have to run the install program through SAW to enter uninstall mode.
- You have not renamed the program icons in Program Manager (or your current shell).

The uninstall program **will**:

- Delete the SAW program files, accessories and configuration files.
- Ask you if you want any reports created deleted as well.
- Delete the SAW directory and other (lower) empty directories.
- Delete the group and/or Saw's program icons from Program Manager (the shell).
- Ask you if you want to delete empty Program Manager groups (the shell).

The uninstall program **will not**:

- Delete non-SAW files & directories (unless you want to).
- Delete the icons if they have been renamed.

**Note:** Even though I am referring to **Program Manager** as the shell, SAW will work with **any** other compatible program shell (*Explorer* in Windows 95, *Dashboard*, *PC Tools Office*, *Norton Desktop*, etc.) or other programs **if** they emulate the shell's functions as they should.

See Manual UnInstall if you want more information about this procedure.

## How to Manually UnInstall SAW

To uninstall SAW is quite easy. You must follow the following steps (this assumes that you have installed SAW in its own directory):

- Determine the directory where you have installed SAW. Use the *File Manager* to do a search from the root directory of all drives for **SAW.\*** if you forgot where it is. If the icons are installed in Program Manager click on SAW icon (to highlight it) and then press **Alt-Enter** to see its proprieties. The *Command Line* should tell you the directory.
- Use a file manager or DOS to log to that directory. You should see following files:
  - **saw.exe, saw.hlp** - Main executable & help files - *delete*.
  - **saw.ini** - Configuration file - *delete*.
  - **licence.wri, readme.wri** - Information Files - *delete*.
  - **ctl3d.dll** - (optional, *only* if Windows directory is protected) - Dialog extension file - *delete*.
  - **saw.txt** or other **.txt** files - Reports generated - *delete* them if you don't need them any more or *move* them somewhere else.
  - SAW does not include **any** other files. If you still have files left then you might have another program in this directory ! Skip next step.
- Delete the empty directory and any other empty directories left.
- Delete the icons from your program shell. If you have let the **Install** program do it then follow the steps below. If you did it yourself then...
- If you have **Program Manager** as your current shell:
  - Look in the groups for an icon named **SAW** (default **Accessories** group). Make the group which contains it active (click on it or double-click on it's icon).
  - Click on the icon named **SAW** to make it active. Press **DEL**. Answer **Yes** to the pop-up window. The icon should disappear.
  - Repeat the above step for icons named **Licence** and **ReadMe**.
  - If the group is empty, delete it altogether.
- If you have an alternative shell, delete the icons **SAW**, **Licence** and **ReadMe** icons from the group - your manual should tell you how to do this.

SAW is uninstalled now ! If you did not like it, would you be so kind to try the next version when available ?!?

**Note:** If you have trouble uninstalling programs, you should look for some *uninstallers* which will delete any program from your system with minimum of fuss. There are both shareware (unfortunately no freeware :( or commercial programs available at decent prices. Check them out !

## Integrity Check

I designed SAW to be used mainly for snooping on your friend's computer to see why it is faster than yours (as always... :). True, there are no benchmarks, but you must have done that a long time ago. To find out what exactly he / she has in that machine is a bit harder.

This could mean that SAW will be kept on a floppy, send through a modem or by any other means. It may have had a long way until it reached your machine. The chances of infection are small today, but to give you piece of mind this checking is made. It is based on a custom CRC algorithm and checks *every* byte of the program. This means that most changes will be detected. Also, it should detect hacked copies or modem CRC failed check.

This will **not** work reliably if:

- You have a stealth virus which will show a correct copy of the file, so the program will think it is OK. Small chance, though. You need a good TSR scanner to protect against this.
- The program was hacked by a '*good*' hacker. Well, it was not made to be 100% *hacker proof* - after all SAW is not an encrypted program.

**Note:** This checks that the **saw.exe** file is OK. It does **not** check **any** other files.

## Program Options - Customising SAW

Enables you to customise SAW to your requirements. This is a main menu for program options. Select the one you wish to customise by pressing the corresponding button.

Alarm Options - Change the way SAW warns you of system events.

Button Options - You can change SAW's looks: horizontal, vertical, square & change button alignment.

General Options - Enables detail program customisation, should you wish to do it.

Customise Toolbar - Enables you to specify which buttons are to be included, i.e. to have only the information you want readily available and avoid clutter.

Save Settings Check box - Enables you to save your customised selections.

**Note:** If you messed up the configuration and want to return to the default, delete the **SAW.ini** file in the directory where SAW is installed. SAW will load the default values next time it is loaded.

## Alarm Options

SAW monitors your system regularly. See [SAW Errors / Events](#). SAW does all this even when it is minimised, and will warn you when something takes place by the following means:

- **Audio** - A beep will be heard or, if you have a sound card, the .wav file assigned to the *default beep* will be played.
- **Video** - If SAW is minimised, the icon will flash and the icon name will be changed to the error / event encountered. If it is not minimised the title bar will flash and the error / event will be displayed on the title bar.

The *Alarm Options* dialog box offers you the following choices:

- Audio Alarm**
  - **Beep** - SAW will use a beep to warn you.
  - **No Beep** - SAW will not use audio warnings.
- Video Alarm**
  - **On Top** - SAW will pop up over the active application.
  - **Normal** - SAW will remain normal.
- Alarm Mode**
  - **Enabled** - You will be told of all system events.
  - **Severe Only** - Only severe warnings will be issued.
  - **Disabled** - SAW will not monitor your system.

### Sound Alarm

- **System** - The minimum percentage of free system memory your system should have that you consider *safe*. If the free system memory percentage falls below this number, SAW will warn you. This number is an integer between **1** and **99**. Default **15%**.
- **GDI** - The minimum free GDI resources that should be free on your system. If the free GDI resources fall below this number, SAW will warn you. This number is an integer between **1** and **99**. Default **15%**.
- **USER** - The minimum free USER resources that should be free on your system. If the free USER resources fall below this number, SAW will warn you. This number is an integer between **1** and **99**. Default **15%**.
- **Base Mem** - The minimum contiguous space of base memory available to Windows applications. If you run out of this, even 32bit Windows applications halt. SAW will warn you if this falls below the this number. This should be an integer between **1** and **640**. **0** means that the checking is turned off ! Default **5KB**.



## Button Options

You *can* change the way SAW looks ! See the options below:

**Window Form** - changes the way the program window looks. If you have sized the window and you want to be the default select *Current* and check the Save Settings Check box.

- **Horizontal** - All buttons in a single row.
- **Default** - All buttons grouped in square root (of number of buttons) rows.
- **Current** - Keep the current custom form. (to save it)
- **Vertical** - All buttons in a single column.

**Align Buttons** - When you maximise SAW you see that the buttons align from left to right in rows. You can select vertical or make rows equal to columns.

- **Horizontal** - Buttons align in rows.
- **Vertical** - Buttons align in columns.
- **Symmetrical** - Same number of rows / columns.

**Note:** All this arrangements are made depending on the resolution, system font and other various Windows parameters. SAW tries to look at its best with all system configurations.

## General Options

These specify a few configuration options which instruct SAW to skip some detection procedures which may cause problems when running on some systems. These change quite a lot so expect some changes in future versions.

Read the **ReadMe** file for details.

The **Topics do** vary with program versions and are not guaranteed to be present in future versions. Therefore, a list is not available. See above.

## Customise Toolbar

As SAW is just a big toolbar and nothing much, there had to be a way to customise it in much the same way all modern Windows applications allow you to. However, I decided against a direct approach (i.e. directly moving the buttons in the toolbar) as this would probably create a mish-mash out of the interface.



The dialog presents you with a picture of a **Trash Can** (seen on the left) in the middle and two groups of buttons:

- **Active Buttons** - the ones that are currently in the toolbar.
- **Inactive Buttons** - the other buttons which you may include in the toolbar.
  - To remove a button, drag it (the cursor will change shape but the button will remain visible) from the *Active* group to the *trash can* or to the other group.
  - If you want to add a button, drag it from the *Inactive* group to the *Active* group.
  - To move a button, drag it from its current position to its new position.



The icon on the left denotes a **Free Button**, i.e. a place where is available for you to place a button in (on the toolbar).

The groups will change if the operation was successful. Press **OK** to accept the changes you have made or **Cancel** to abandon them.

**Note:** The *Button Description* field gives brief information about the purpose of the last button clicked.

## Save Options Check box

If you leave this check box unchecked (default), then all the selections you have made will be active only for the current session. Next time, SAW will default to its own values. This is good when you want to experiment with the program choices.

If you check it, SAW will save them in a file called *saw.ini* in its own directory. SAW does **not** touch **any** Windows **.ini** files. Next time you start SAW, it will look as you wanted and will retain this look until you change it.

If you keep it on floppy you should copy this file as well. It is very small, but will make SAW look more normal to you... (on another system)

## Total Memory Information

This is the total physical memory that your computer has in **MB** (Megabytes). This is the number quoted when you see computers advertised - you don't buy a 640KB Base Memory and 7,424KB Extended Memory machine but a **8MB** one. This number is far more meaningful especially for beginners.

However, the two numbers mentioned are still important for a couple of reasons:

- Some older computers had only **512KB** of base memory. Even today, some have only **639KB** available as they use *1KB* for *disk information*. If you are in this situation, try to put the information somewhere else as some programs may crash (badly designed ones **only**) as they assume 640KB.
- Extended memory is also important. For example a **8MB** machine has the first MB made of base memory (**640KB**) and UMB (**384KB**). The UMB contain the System BIOS (usually **64KB**) the Video BIOS (usually **32KB**, **48KB** for **8514/A**, **16KB** for **EGA**) which are rounded off to 128KB. This 128KB (in total) are usually Shadowed to give improved performance. (The UMB also contain 128KB or Video Display Memory which should **not** be shadowed). You have three choices:
  - Shadow all UMBs to give you all the performance you can get. But, if you don't have any network or fancy disk controllers (some cached controllers do not work if you shadow their addresses) you shadow *256KB* for nothing. Also, Windows bypasses the BIOS sometimes. You end up with **7168KB** this way (**7MB**).
  - Shadow only those *128KB* which gives you good performance and also get 256KB of Extended Memory (by relocation) that gives a total of **7424KB** (**3.6%** extra). This is not a lot, but Windows will appreciate it. This mode is **Recommended**.
  - Now, as Windows by-passes the BIOS many times you may not use BIOS shadowing at all and gain all the 384KB which gives you **7552KB** Extended Memory. Now, that's different isn't it ? A **5.3%** increase of memory for free over a conventional system. Your system may start very slowly though...

**Note:** This Shadowing / Memory Relocation should be explained in your Motherboard / BIOS **manual**. It is not very difficult to adjust it. For AMI BIOS this options are in the **Advanced Options** menu. Don't forget, to relocate the un-shadowed memory otherwise you don't gain much !

## Shadowing

Shadowing is a technique that copies the code from slow ROM to much faster RAM). Usually the BIOSes are shadowed to achieve greater performance. Your Motherboard / BIOS manual should tell you about this function.

## **UMB**

UMB are Upper Memory Blocks - Memory between 640KB and 1024KB (A0000-FFFFF). They contain the BIOSes extra memory etc. See [Books](#).

## Floppy Disks

The list of all the physical floppy disks found in your system (as recognised by the BIOS).

They are listed as: **Drive Name** (e.g. **A:**) and **Capacity / Format** (e.g. **1.44MB 3½ inch**). The drives in the list are separated by two spaces.

**Note:** Removable devices like optical & floptical disks may also be listed as fixed disks depending on device driver.



## Hard Disks

The list of all local hard disk systems (fixed) present in your system. This is the logical drives list (e.g. how DOS/Windows sees it) not the physical list.

They are listed as: **Drive Name** (e.g. **C:**) and **Capacity** (e.g. **80MB**). The drives in the list are separated by two spaces.

**Note:** Removable devices like optical & floptical disks may also be listed as fixed disks depending on device driver.

**Note:** CD-ROM and Memory Drives will not be included in this list. Host drives are not included either.

## Freeware Licence - Short Summary

The program and associated files may be copied, used and posted on any BBS, FTP site, etc. without charge or permission. It may be included in a companion disk with any book or magazine, and included in any shareware library, as long as the program and associated documentation are **not** modified. It should be specified that the program itself is free.

The program and associated documentation may not otherwise be modified, sold, or used as part of a commercial package without the express permission of the author.

The **System Analyst for Windows ("SAW")** and its **Install Program** are **Copyrighted** by me, **Catalin-Adrian Silasi**, © 1992-1995, **ALL RIGHTS RESERVED**. SiSoft Software © (p) 1995.

**WARNING:** This is only a very short summary. It is by no means a full explanation of its concept nor it intends to be one. For a full description see the **Licence** file in your SAW directory.

## Network Information

This window provides general information about the network your computer is connected to. The first table shows the generic information which is the same no matter which network you are connected to. The other tables list the topics specific to the network used.

### General Topics:

<u>Network</u>	<u>Network Name</u>	<u>Second Network</u>
<u>NetBIOS Installed</u>	<u>IPX/SPX Installed</u>	<u>TCP/IP Installed</u>

### Novell Netware Topics:

<u>Default Server</u>	<u>User Login</u>	<u>Real Name</u>
<u>Shell Info</u>	<u>Internet Address</u>	<u>Comp. Node</u>

### Windows for Workgroups Topics:

<u>User Name</u>	<u>Workgroup</u>	<u>Computer Name</u>
<u>File/Print Share</u>	<u>Validated Login</u>	<u>Start-up Reconnect</u>

**Note:** For more information see the relevant manuals. The Novell Netware manuals are quite hefty, but there are a number of documents/books that explain the basics without getting into much detail. The Windows for Workgroups manuals are not that big so you can browse through them easier.

## **Network: General Type**

This is the general type of network as detected by Windows not by SAW. If Windows does not detect your network SAW will not either ! It does this specially to prevent crashes, etc.

There are two types of network set-up:

- **Single** - one network running only.
- **Multiple** - two or more networks running.

Networks detected:

- **Novell Netware**
- **Microsoft's Windows for Workgroups (WFWG)**
- **Microsoft's Windows 95**

## Network Name

This lists the name of the network your computer is connected to (if you are connected to a network in Windows). Usually the name of the manufacturer of the network software and its version is also shown.

**Note:** This information refers to the Windows network device driver and **not** the network software running on the server(s) ! This is because different file servers may use different network operating system versions (e.g. 3.11 and 3.12).

For example the network may be running *Novell Netware 4.00*, but your Windows network driver has version *3.11* (the workstation's drivers were not upgraded, etc.). You should try to use the same version of drivers even if older drivers will still work. You may be missing some new features / improvements.

## Network: Second Network

Similar information as Network Name, but about the **Second Network** (if any) are installed and running. See Network for more information.

Note that **no** further information is given on this network, all further information refers to the **Main Network** as specified by the Network Name !

## Network: NetBIOS Installed

This is another major low-level network layer, among IPX/SPX and TCP/IP. NetBIOS was developed in 1984 as a high-level programming interface to the IBM PC Network. The first implementation was a BIOS extension on a LAN adapter (network) card, but many current networks support NetBIOS as the session layer.

As I mentioned, this can either be a device driver (like the others) or built in the BIOS of a network card (optional). This BIOS can also enable remote booting for diskless workstations.

## Network: IPX / SPX Installed

Specifies if the IPX / SPX (InterNetwork Packet Exchange) layer is installed. IPX is a low-layer of the Novell Network operating system.

Some network-aware programs do not need full-blown Netware of WFWG if you just want to send a few files back and forth. IPX may be the cheapest (in terms of software) network to have, but as we are running Windows, WFWG could be the **best** around...



## Network: TCP/IP Installed

Specifies if a **TCP/IP** (*Transmission Control Protocol / Internet Protocol*) stack (protocol) is running at the moment. This enables you to connect to the Internet, although other *forms* of connection exist but merely as a gateway to a service provider.

By the way, this packet mode protocol was designed to survive a nuclear bomb disaster, so don't worry: at least your data will be safe... :)

**Note:** Not all TCP/IP protocol implementations may be detected...

**WFWG Network: User Name**

The *default* Login name of the current user (you?). Several login names may exist for different servers / rights.

**WFWG Network: Workgroup**

The workgroup (group) name that the current user belongs to (if any). This could uniquely identify a user from a another with the same login (to be avoided !).

**WFWG: Computer Name**

The name of the current workstation as specified at install time. All workstations on a network should have a unique name - to enable better network management.

## **WFWG: File / Print Sharing**

This shows if you have enabled:

- **File Sharing** - some / all / no directories of your local hard drives are accessible from the network depending on set-up.
- **Print Sharing** - the local printer(s) attached to your computer is / are accessible from the network or not.

**Note:** This resources can be password protected if required (recommended).

**WFWG: Validated Login**

This specifies if the user has to do a *validated login* (probably on a Windows NT server) in order to use the current workstation. This method is more secure than a usual login.

**WFWG: Start-up Reconnect**

This specifies if all the current connections and mappings are to be enabled at start-up. Use this if you want to keep the current network set-up as default.

## Netware Network: Default File Server

This shows:

- The name of the *Default (Preferred)* File Server that you are logged onto.
- The version of the *Netware Operating System* that the server is running.
- The maximum number of users that can be attached at the same time.

**Note:** The default server may change in some circumstances.



## **Netware Network: User Login/Station**

This shows:

- The login name on the default server. (See [Default File Server](#)).
- The station number that is connected to the server.

**Netware Network: User Real Name**

This shows the real name (if specified) of the user with respect to the login (See User Login) on the default server. If the login name is a shared (transfer) one - no such name exists.

## **Netware Network: Shell Information**

The shell is the front-end to the network driver that enables you to connect and use the *Netware Network Operating System*. You can see:

- The version of the shell and its revision. This may not be the version of the network device driver or the version of Netware running on any of the servers you are connected to.
- The type of shell and the memory type it uses.

**Netware Network: Internet Address**

This is the Internet IP Address (4 bytes) of the machine that you are using. This is not unique, and may be shared by many workstation on the same node. However, each machine has a different *physical address* (6 bytes); See Computer Node.

### **Netware Network: Computer Node / Socket**

This shows:

- The physical address (node) of the workstation on the network. This number is unique.
- The socket that this workstation is using on the network.

## Advanced Power Management (APM) Information

APM is a special facility offered mainly by portables and the new *Green* PCs. It powers down parts of the system to conserve power when idle. It must be implemented in both hardware and software .

To take advantage of it the BIOS must implement APM. A **SL** processor is recommended (386SL, 486SL, etc. - use less power) but the new processors will be all low power (like the 486DX4). For portables, a software driver is usually the only other thing required (DOS & Windows) which comes with the computer, but there are also generic ones.

For desktops it's more complicated. To power down the monitor the video card and monitor must support APM as well. Also some peripherals will give you problems (end 1994) so you may be forced to ignore it *for now*.

### Topics:

[APM Version](#)

[APM Status](#)

[Battery Charged At](#)

[P. Mode Support](#)

[AC Line Status](#)

[Time Left](#)

[Clock Slowdown](#)

[Battery Status](#)

[System Listbox](#)

## **APM Version**

The version of the APM specification your BIOS supports. Newer versions are compatible with the old ones and provide more functions (and more information).

The newest version (end 1994) is **1.10**.

**Note:** Of course, this is **not** the version of your BIOS. See [BIOS Version](#).

## **APM: Protected Mode Support**

This information is mainly for programmers.

It specifies if the BIOS can be accessed in protected mode (on 286, preferably 386 processors and higher). This may include support for **16** and / or **32bit** programs.



## APM: Clock Slowdown

Specifies if the BIOS will slow down the processor's clock when idle. By slowing the clock to 1/8 of its nominal frequency power saving can be achieved in addition to the usual power down of hard disks, displays, etc. This is useful mainly on computers which use fast processors (clock doubled or tripled, etc.) but are under a *light* workload.

Some put the processor in **0MHz** *idle* mode, when it consumes even less power when processing is not required.

**Note:** Even if this facility is not supported, your APM BIOS may use other methods to conserve energy. Read your user manual.

## APM Status

The status of your APM BIOS. This can be:

- **Disabled** - i.e. off. No power conservation facilities are used - like not having APM at all.
- **Disengaged** - APM is on stand-by. Power saving facilities will be used when needed.
- **Enabled** - APM is enabled, power saving facilities are operational.

## **APM: AC Line Status**

The status of the AC line (mains power). This can be:

- **On Line** - the computer is connected to the mains.
- **Off Line** - the computer is *not* connected to the mains.
- **Backup Power** - the system is on backup (auxiliary) power.

## APM: Battery Status

The status of the battery in the system. This may be:

- **High** - battery is OK, charged; no worries.
- **Low** - you should consider recharging it or swapping it with a charged one. If you plan on going away keep a spare set handy - you may need it...
- **Critical** - **save** your work immediately ! Imminent power failure ! Swap batteries or plug the computer into mains.
- **Charging** - the battery is charging (the computer is on mains power). The status of the battery does not matter for the time being, but you should charge it until *high*.

**Note:** If no system battery is found or the battery is damaged, *No System Battery* will be displayed.

## **APM: Battery Charged At**

The **estimated** percentage of power (charge) left in battery (0-100%). You should aim to keep it high (85+) when you intend *to go out* (i.e. where no mains is available).

If the battery cannot be charged to 90+ no matter what you do, it has probably acquired *memory* (i.e. cannot be charged fully). See your manual about how to fix this.

See [Battery Status](#) for more direct information.

## APM: Time Left

This is an **estimation** of the time the computer will continue to function at the current consumption rate. If the workload varies, the reading may be inaccurate. In time you should get an idea about what this numbers really mean to your work-style.

**Note:** Usually this is the *best time*; unfortunately the machine will function a much shorter time. You should check Battery Status instead.

## APM: System Listbox

This shows a detailed list of APM status for all the components in your system. The system components are grouped in the following classes:

- **All Devices**
- **Display** (Graphic Adapter + Screen)
- **Secondary Storage** (Hard Disks, Floppy, CD-ROM, etc.)
- **Parallel Ports**
- **Serial Ports**
- **Network Adapters** (Network Card, etc.)
- **PCMCIA Sockets** (Flash Disks, Hard Disks, etc.)
- **Other** (Any other devices)

The APM status for every device can be one of the following:

- **Enabled** - APM functions enabled.
- **Stand-By** - APM functions will be used when device is idle.
- **Suspend** - APM suspended for this device(s).
- **Off** - APM has been turned off for now.
- **Reserved** - Hardware manufacturer or OS special mode.
- **Disabled** - APM completely disabled.

**Note:** If a device class contains only one device then only class information will be shown. If more than two, the numbering starts from **0**.

**Note:** If all system devices are in the same state then no information will be shown. See [APM Status](#) for this kind of information.

## Hard Disk Interfaces

This little topic tries to explain the differences between the types of hard disk controllers.

**ST506(/412)** - is the first controller used in the PC/XT. This consists of a *Host Adapter* (translates computer's requirements into disk controller's language) and the actual *Disk Controller* on an expansion card. The data was transmitted serially, and the controller looked after everything: moving the heads, selecting the heads, etc. This basic system can still achieve maximum 640KB/s.

**ESDI** (*Enhanced Small Device Interface*) - is a development of the ST506 controller which is more intelligent, requiring less PC involvement. Also, it can hide its true configuration in order to get over many limitations of the ST506. Maximum data transfer rate 2MB/s.

**SCSI** (*Small Computer System Interface* - pronounced "scuzzy") - has the Host Adapter on an expansion card while the Disk Controller is embedded in the Hard Disk. SCSI can also connect up to 8 devices (in 1994) including optical drives, tape backup, scanners, etc. Data is transferred in parallel. SCSI devices can also communicate with each other without involving the PC at all; great for (true :) multitasking systems.

The original SCSI had a 8bit data bus giving 5MB/s data transfer rate; new ones have 16bit or 32bit (*wide bus*) and also *fast bus* - 10MB/s and *fast and wide* - 20MB/s. The **SCSI-2** devices is an enhanced SCSI with *fast / wide bus* and also includes new commands. You should buy SCSI-2 devices if you can.

**IDE** (*Integrated Drive Electronics*) - is a SCSI derivative which removes much of the translation procedures. The Disk Controller remains on the Hard Disk while the PC talks directly to it (actually through a pretty dumb interface). This makes it cheaper, but you can only control 2 hard disks. A 16bit parallel data link gives 4.5MB/s data transfer rate.

**EIDE** (*Enhanced IDE*) is a developed IDE controller which removes many limitations and also gives faster transfer rate. You can also connect CD-ROM drives or other EIDE peripherals (up to 4). Go for EIDE if you can.

---

## Q & A

*Why doesn't IDE use DMA ?* IDE uses programmed I/O which operates more efficiently than DMA because the fixed disk data is buffered. This is accessed using 16bit 286 repeat I/O instructions (32bit access on 386+) which operate at memory bandwidth and require no instructions fetches once inside the CPU instruction queue.

*So, is IDE faster than SCSI ?* Well, is not as easy as that, in fact depends on the hard drive quality more than interface. Do a test for yourself and find out. Also EIDE does support DMA...

*That's all !?!* Yeah, I'm afraid so...



## List of many acronyms

<b>AC</b>	Alternating Current
<b>AE</b>	Applications Engineer
<b>AFS</b>	Andrew File System
<b>AI</b>	Artificial Intelligence
<b>ALU</b>	Arithmetic-Logic Unit
<b>AM</b>	Amplitude Modulated
<b>AMD</b>	Advanced Micro Devices, Inc.
<b>AMI</b>	American Megatrends Inc.
<b>ANSI</b>	American National Standards Institute
<b>ASCII</b>	American Standard Code for Information Interchange
<b>ASIC</b>	Application Specific Integrated Circuit
<b>ASPI</b>	Advanced SCSI Programming Interface
<b>AT</b>	Advanced Technology
<b>ATA</b>	AT bus Attachment
<b>ATDM</b>	Asynchronous Time Division Multiplexing
<b>ATM</b>	Asynchronous Transfer Mode
<b>AUI</b>	Attached Unit Interface
<b>BBS</b>	Bulletin Board System
<b>BCC</b>	Block Check Character
<b>BCD</b>	Binary Coded Decimal
<b>BE</b>	Back End
<b>BiCMOS</b>	Bipolar Complementary Metal-Oxide Semiconductor
<b>BIOS</b>	Basic Input / Output System
<b>BNC</b>	Bayonet Nut Connector
<b>BPS/bps</b>	Bytes/bits Per Second
<b>BSC</b>	Binary Synchronous Communications
<b>BSD</b>	Berkeley Standard Distribution
<b>BTU</b>	British Thermal Units
<b>CAD</b>	Computer Aided Design
<b>CAM</b>	Computer Aided Manufacturing
<b>CAM</b>	Content Addressable Memory
<b>CAM</b>	Common Access Method
<b>CAS</b>	Column Address Strobe
<b>CBX</b>	Common Branch eXtender
<b>CCD</b>	Charge Coupled Device
<b>CCITT</b>	Consultative Committee of International Telephony and Telegraphy
<b>CD</b>	Carrier Detect
<b>CDDI</b>	Copper Distributed Data Interface
<b>CDROM</b>	Compact Disk Read Only Memory
<b>CGA</b>	Color Graphics Adaptor
<b>CHS</b>	Cylinder Head Sector
<b>CISC</b>	Complex Instruction-Set Computer
<b>CLA</b>	Carry Lookahead Adder
<b>CMOS</b>	Complementary Metal-Oxide Semiconductor
<b>CP/M</b>	Control Program / Monitor
<b>CPI</b>	Clocks Per Instruction
<b>CQFP</b>	Ceramic Quad-FlatPack
<b>CPU</b>	Central Processing Unit
<b>CR</b>	Carriage Return
<b>CRC</b>	Cyclical Redundancy Check
<b>CRQ</b>	Command Response Queue

<b>CRT</b>	Cathode Ray Tube
<b>CS</b>	Chip Select
<b>CSMA/CD</b>	Carrier Sense Multiple-Access /with Collision Detect
<b>CSR</b>	Command Status Register
<b>CTS</b>	Clear To Send
<b>DAT</b>	Digital Audio Tape
<b>DC</b>	Direct Current
<b>DCD</b>	Data Carrier Detect
<b>DCE</b>	Data Circuit-terminating Equipment
<b>DD</b>	Double Density
<b>DEC</b>	Digital Equipment Corporation
<b>DES</b>	Data Encryption Standard
<b>DID</b>	Direct Inward Dial
<b>DIN</b>	Deutsche Industrie Norm (German)
<b>DIP</b>	Dual-In-line Package
<b>DIS</b>	Draft International Standard
<b>DMA</b>	Direct Memory Access
<b>DOS</b>	Disk Operating System
<b>DPE</b>	Data Parity Error
<b>DPSK</b>	Differential Phase Shift Keying
<b>DRAM</b>	Dynamic Random Access Memory
<b>DRDOS</b>	Digital Research Disk Operating System
<b>DS</b>	Double Sided
<b>DSP</b>	Digital Signal Processor
<b>DSR</b>	Data Set Ready
<b>DTC</b>	Data Terminal Controller
<b>DTE</b>	Data Terminating Equipment
<b>DTMF</b>	Dual-Tone MultiFrequency
<b>DTR</b>	Data Terminal Ready
<b>EBCDIC</b>	Extended Binary Coded Decimal Interchange Code
<b>ECC</b>	Error Correction Code
<b>ECL</b>	Emitter-Coupled Logic
<b>ECO</b>	Engineering Change Order
<b>ECP</b>	Enhanced Communication Port
<b>ECU</b>	EISA Configuration Utility
<b>EEPROM</b>	Electrically Erasable Programmable Read-Only Memory
<b>EGA</b>	Enhanced Graphics Adaptor
<b>EIA</b>	Electronic Industries Association
<b>EIDE</b>	Enhanced Integrated Device Electronics
<b>EISA</b>	Enhanced Industry Standard Architecture
<b>EMI</b>	Electro-Magnetic Interference
<b>EMF</b>	Electro-Magnetic Force
<b>EMS</b>	Expanded Memory Specification
<b>EOF</b>	End Of File
<b>EOL</b>	End Of Line
<b>EOS</b>	Electrical OverStress
<b>EPP</b>	Enhanced Parallel Port
<b>EPROM</b>	Erasable Programmable Read-Only Memory
<b>ESD</b>	Electro-Static Discharge
<b>ESDI</b>	Enhanced Small Devices Interface
<b>FAT</b>	File Allocation Table
<b>FCC</b>	Federal Communications Commission

<b>FDD</b>	Fixed / Floppy Disk Drive
<b>FDDI</b>	Fiber Distributed Data Interface
<b>FDM</b>	Frequency Division Multiplexing
<b>FDX</b>	Full-Duplex Transmission
<b>FE</b>	Front End
<b>FEP</b>	Front End Processor
<b>FF</b>	Form Feed
<b>FIFO</b>	First-In First-Out
<b>FILO</b>	First-In Last-Out
<b>FM</b>	Frequency Modulation
<b>FPGA</b>	Field Programmable Gate Array
<b>FPU</b>	Floating Point Unit
<b>FRU</b>	Field-Replaceable Unit
<b>FSF</b>	Free Software Foundation
<b>FSK</b>	Frequency Shifty Keying
<b>FTP</b>	File Transfer Program
<b>GAS</b>	Gallium Arsenide
<b>GFLOPS</b>	Billions of FLOating Point Operations Per Second ("GigaFlops")
<b>GNU</b>	Gnu's Not Unix
<b>GUI</b>	Graphical User Interface
<b>HD</b>	High Density / Hard Disk
<b>HDD</b>	Hard Disk Drive
<b>HDX</b>	Half-Duplex Transmission
<b>HFS</b>	Hierarchical File System
<b>HPFS</b>	High Performance File System
<b>HS</b>	Helical Scan
<b>I/O</b>	Input / Output
<b>IBM</b>	International Business Machines Corporation
<b>IC</b>	Integrated Circuit
<b>IDC</b>	Insulation Displacement Connector
<b>IDE</b>	Integrated Device Electronics
<b>IEEE</b>	Institute of Electrical and Electronic Engineers
<b>IMP</b>	Interface Message Processor
<b>IPC</b>	InterProcess Communication
<b>IPX</b>	Internetwork Packet eXchange
<b>IRQ</b>	Interrupt ReQuest
<b>ISA</b>	Industry Standard Architecture
<b>ISDN</b>	Integrated Services Digital Network
<b>ISO</b>	International Standards Organization
<b>JFS</b>	Journalled File System
<b>kVA</b>	KiloVolt-Amps
<b>LAN</b>	Local Area Network
<b>LBA</b>	Linear Block Array
<b>LCD</b>	Liquid Crystal Display
<b>LED</b>	Light Emitting Diode
<b>LF</b>	Line Feed
<b>LIM</b>	Lotus/Intel/Microsoft's Expanded Memory Manager (EMS)
<b>LRU</b>	Least-Recently Used
<b>LSB/lb</b>	Least Significant Byte/bit
<b>LSI</b>	Large Scale Integration
<b>LUN</b>	Logical Unit Number
<b>MAN</b>	Metropolitan Area Network

<b>MB/Mb</b>	Mega Bytes/bits
<b>MBR</b>	Master Boot Record
<b>MCA</b>	Micro Channel Architecture
<b>MCGA</b>	Multi-Color Graphics Array
<b>MCM</b>	Multi-Chip Module
<b>MFLOPS</b>	Millions of FLOating Point Operations per Second ("MegaFlops")
<b>MFM</b>	Modified Frequency Modulated
<b>MHz</b>	MegaHertz
<b>MICR</b>	Magnetic Ink Character Recognition
<b>MIDI</b>	Musical Instrument Data Interface
<b>MIMD</b>	Multiple-Instruction Multiple-Data
<b>MIPS</b>	Millions of Instructions per Second
<b>MISD</b>	Multiple-Instruction Single Data
<b>MMU</b>	Memory Management Unit
<b>MNP</b>	Microcom Network Protocol
<b>MODEM</b>	MODulator / DEModulator
<b>MOPS</b>	Millions of Operations Per Second
<b>MOS</b>	Metal-Oxide Semiconductor
<b>MP</b>	MultiProcessor
<b>MPP</b>	Massively Parallel Processor
<b>MSB/msb</b>	Most Significant Byte/bit
<b>MSDOS</b>	Microsoft's Disk Operating System
<b>MSI</b>	Medium Scale Integration
<b>MTBF</b>	Mean Time Between Failure
<b>N/C</b>	No-Connect
<b>NBS</b>	National Bureau of Standards
<b>NEMA</b>	National Electrical Manufacturers Association
<b>NFS</b>	Network File System
<b>NFU</b>	Not-Frequently Used
<b>NMI</b>	Non-Maskable Interrupt
<b>NMOS</b>	Negatively doped Metal-Oxide Semiconductor
<b>NOP</b>	No OPeration
<b>NRU</b>	Not-Recently Used
<b>NSF</b>	National Science Foundation
<b>NVRAM</b>	NonVolatile Random Access Memory
<b>OCR</b>	Optical Character Recognition
<b>ODI</b>	Open Datalink Interface
<b>OEM</b>	Original Equipment Manufacturer
<b>OS</b>	Operating System
<b>OSF</b>	Open Software Foundation
<b>OSI</b>	Open Systems Interconnect
<b>PAL/PLA</b>	Programmable Array Logic / Logic Array
<b>PB</b>	Push Button
<b>PBX</b>	Private Branch eXtender
<b>PC</b>	Personal Computer, Program Counter
<b>PCB</b>	Printed Circuit Board
<b>PCI</b>	Peripheral Component Interconnect
<b>PCM</b>	Pulse Code Modulation
<b>PCMCIA</b>	Personal Computer Memory Card International Association
<b>PE</b>	Processor Element
<b>PFF</b>	Page Fault Frequency
<b>PGA</b>	Proffesional Graphics Array

<b>PGA</b>	Pin Grid Array
<b>PIC</b>	Programmable Interrupt Controller
<b>PIO</b>	Programmed Input/Output
<b>PLCC</b>	Plastic Leaded Chip Carrier
<b>PLL</b>	Phase Locked Loop
<b>PM</b>	Preventive Maintenance
<b>PMOS</b>	Positively doped Metal-Oxide Semiconductor
<b>POST</b>	Power On Self Test
<b>PPP</b>	Point-to-Point Protocol
<b>PQFP</b>	Plastic Quad-FlatPack
<b>PROM</b>	Programmable Read-Only Memory
<b>PSTN</b>	Public Switched Telephone Network
<b>PTE</b>	Page Table Entry
<b>QAM</b>	Quadrature Amplitude Modulation
<b>QFP</b>	Quad-FlatPack
<b>QIC</b>	Quarter Inch Cartridge
<b>RAID</b>	Redundant Arrays of Inexpensive Disks
<b>RAM</b>	Random Access Memory
<b>RAMDAC</b>	Random Access Memory Digital to Analog Converter
<b>RAS</b>	Row Address Strobe
<b>RCA</b>	Radio Corporation of America
<b>RCC</b>	Routing Control Center
<b>RFC</b>	Request For Comments
<b>RFI</b>	Radio Frequency Interference
<b>RI</b>	Ring Indicator
<b>RISC</b>	Reduced Instruction-Set Computer
<b>RLL</b>	Run Length Limited
<b>RMS</b>	Root Mean Squared
<b>RMW</b>	Read Modify Write
<b>ROM</b>	Read-Only Memory
<b>RPC</b>	Remote Procedure Call
<b>RPM</b>	Rotations Per Minute
<b>RTC</b>	Real Time Clock
<b>RTS</b>	Request To Send
<b>SAM</b>	Sequential Access Memory
<b>SASI</b>	Shugart Associates Standard Interface
<b>SCSI</b>	Small Computer Systems Interface
<b>SD</b>	Single Density
<b>SDLC</b>	Synchronous Data Link Control
<b>SE</b>	Systems Engineer
<b>SIMD</b>	Single-Instruction Multiple-Data
<b>SIMM</b>	Single Inline Memory Module
<b>SIPP</b>	Single Inline Pinned Package
<b>SISD</b>	Single-Instruction Single-Data
<b>SLIP</b>	Serial Line Internet Protocol
<b>SMD</b>	Surface Mount Device
<b>SMT</b>	Surface Mount Technology
<b>SNA</b>	System Network Architecture
<b>SNR</b>	Signal to Noise Ratio
<b>SO/SOL</b>	Small OutLine
<b>SOIC</b>	Small Outline Integrated Circuit
<b>SPOOL</b>	Simultaneous Peripheral Operation On Line
<b>SPT</b>	Sectors Per Track
<b>SPU</b>	Single Processor Unit

<b>SRAM</b>	Static Random Access Memory
<b>SS</b>	Single Sided
<b>STDM</b>	Synchronous Time Division Multiplexing
<b>STN</b>	Super Twisted Nematic
<b>STU</b>	Streaming Tape Unit
<b>SVGA</b>	Super Video Graphics Array
<b>TCM</b>	Trellis Code Modulation
<b>TCP/IP</b>	Transmission Control Protocol / Internet Protocol
<b>TDM</b>	Time Division Multiplexing
<b>TI</b>	Texas Instruments
<b>TIA</b>	Telecomm. Industry Association
<b>TLB</b>	Translation-Lookaside Buffer
<b>TPI</b>	Tracks Per Inch
<b>TRANSISTOR</b>	TRANSformer resISTOR
<b>TSR</b>	Terminate and Stay Resident
<b>TTL</b>	Transistor-Transistor Logic
<b>TUV</b>	Technischer Ueberwachuags Verein (German)
<b>UAE</b>	Unrecoverable Application Error
<b>UART</b>	Universal Asynchronous Receiver/Transmitter
<b>UDP</b>	User Datagram Protocol
<b>UMB</b>	Upper Memory Block
<b>UPS</b>	Uninterruptible Power Supply
<b>USL</b>	Unix System Labs
<b>UUCP</b>	Unix to Unix Copy Program
<b>VCR</b>	Video Cassette Recorder
<b>VESA</b>	Video Enhanced Standards Association
<b>VGA</b>	Video Graphics Array
<b>VLB</b>	VESA Local Bus
<b>VLIW</b>	Very Long Instruction Word
<b>VLSI</b>	Very Large Scale Integration
<b>VM</b>	Virtual Memory
<b>VME</b>	Versa Module Eurocard
<b>VRAM</b>	Video Random Access Memory
<b>VTR</b>	Video Tape Recorder
<b>WAN</b>	Wide Area Network
<b>WATS</b>	Wide Area Telephone Service
<b>WD</b>	Western Digital
<b>WORM</b>	Write Once - Read-Many
<b>WS</b>	Wait State
<b>XGA</b>	eXtended Graphics Array
<b>XMS</b>	Extended Memory Specification
<b>XOR</b>	Exclusive-OR
<b>XT</b>	eXtended Technology
<b>ZIF</b>	Zero Insertion Force

**Note:** Some of these acronyms are explained elsewhere...



## Windows Memory Information

Your system's memory as seen by Windows. As you know, Windows treats it in a totally different way. If you want to see it from DOS's view, see [DOS Memory Information](#). As you are running Windows however, this side of the story is more important. You should be familiar with the various types of memory the complex today's environment provides...

See [SAW Events](#). For detailed explanation see the topics below:

### Topics:

[System Memory](#)  
[Actual Free Space](#)  
[Base Memory](#)

[GDI Heap](#)  
[Largest Free Block](#)  
[Extended Memory](#)

[User Heap](#)  
[Virtual Memory](#)

[The Update Button](#)

**Note:** Slight anomalies may appear on systems running Windows 95, NT or other 32bit operating systems.



## System Memory

This is the total memory available to Windows: **base, extended, virtual** (swap file). It is used by the programs, data, resources, etc. Windows does not use expanded (EMS) memory.

You can see the Total / Free amount of system memory (in Megabytes) and the percentage of free memory. A graphical representation (pie chart) gives a graphical representation of how much memory is used / free. When the percentage falls below the set level (**15%** default) SAW will notify you to close down some applications. (See ***File-Options-Alarm Options***)

The Free System Memory includes the discardable (currently used) blocks.

## GDI Heap

See [The Heap](#) topic if you need more information.

The GDI Heap is memory used to store the graphical elements in Windows, e.g. icons, pens, bitmaps. If you run programs which use a lot of bitmaps, icons, etc., you could run out of GDI Heap. In Windows 3.XX this is **64KB**, so no matter if you have 4, 8, 16, 100MB of RAM you will still run out of heap ! Therefore, you should keep the percentage free above **15%** - otherwise the system may crash. To free GDI heap, close down some applications - but beware: some bad behaved programs (even leading, well known ones !) do not clean up, so you have to restart Windows...

You can see the Total / Free amount of GDI heap (in Kilo-bytes) and the percentage of free heap. A graphical representation (pie chart) gives a graphical representation of how much memory is used / free. SAW monitors continuously your GDI Heap and will inform you of any problems.

**Note:** Program Manager icons do not take GDI Heap.  
Windows NT and Windows 95 do not have this 64KB limitation.

## User Heap

See [The Heap](#) topic if you need more information.

The **User Heap** is memory used to hold information on active windows and other related information. In Windows 3.0 the *Menu* and *String Heaps* are included in the User heap. This heap is also only **64KB**.

Therefore, under Windows 3.0, running big applications with multiple windows and resources may leave you with insufficient User Heap; under 3.1 and later the chance of this happening is reduced. You should keep the percentage free above **15%** - otherwise you may crash. To free User heap close down some applications - but beware: some bad behaved programs (even leading, well known ones !) do not clean up, so you have to restart Windows...

You can see the Total / Free amount of User heap (in Kilo-bytes) and the percentage of free heap. A graphical representation (pie chart) gives a graphical representation of how much memory is used / free. SAW monitors continuously your User Heap and will inform you of any problems.

**Note:** Windows NT and Windows 95 do not have this 64KB limitation.

## The Heap

There are four Windows memory resource heaps (in Windows 3.XX): GDI Heap, User Heap, Menu Heap and String Heap. All have only **64KB**.

The resource heaps (*GDI*, *User*, *Menu* and *String*) are Local Heaps belonging to two system DLLs: *GDI.EXE* and *USER.EXE*. These store objects for Windows applications. In Windows 3.0 Menu and String heaps do not exist - they are included in the User Heap.

In fact there are more heaps in Windows 3.1 but we do not care about them. The following is a summary of the resource heaps found in Windows 3.0 and 3.1X:

Windows **3.0** heaps

- **GDI Heap**                      Graphic resources (pens, brushes device contexts etc.)
- **User Heap**                      Class, Menu, and other Window resources.
- **User Atom Heap**              Global atoms and text.

Additional Windows **3.1 & 3.11** heaps

- **GDI Atom Heap**              GDI atom table (very small - removed from GDI heap)
- **User Menu Heap**              Application Menus.
- **User String Heap**              Application String Resources.

**Note:** Windows NT is a bit different.  
Windows 95 is a bit different too.

## Actual Free Space

A Windows program does not usually load completely into memory - it loads only the part it currently needs. When another part is required, that part is loaded from disk. This process is used because otherwise huge memory would be needed (tens of MB).

When some memory is no longer used, it should be discarded (freed), but if there is plenty of memory free, this operation is postponed until there is a shortage of memory. This is because that part could be required again - being in memory it does not need to be reloaded so performance is improved. Also, to discard it, the memory *records* (kept by Windows) must be amended - this takes time.

The *Actual Free Space* is the actual free memory - as it is now (in kilo-bytes). Therefore, the free system memory reported by SAW (see System Memory) or *Program Manager* is actually the memory which **would** become free if the unused blocks were discarded.

## **Largest Free Block**

As programs allocate and then free memory, the memory becomes *segmented* i.e. used blocks alternate with free blocks. If you need contiguous space - for example for a big bitmap - you need some contiguous space as big as the bitmap (uncompressed). Although you may have plenty of memory, if the memory is segmented you might not be able to load it !

Windows does move memory blocks to create contiguous space if needed, but that takes some time and your machine might become very slow. Some programs do manage with segmented memory, but they will run slower. (analogous to hard disk fragmentation)

SAW tells you the largest free block (in kilo-bytes). To get a feeling of this, run SAW when you start Windows, check this value, run some applications, close them down and check the value again. You will see a much smaller value, even though the total free memory is the same. (assuming all programs cleared up)

## Base Memory

This is the memory used by DOS and its programs to run in (*below 1MB*). They might use extended or expanded memory, but this is used only for data - the programs run only in base memory. Windows programs are different - they do not have this disadvantage.

SAW displays the total / free memory (kilo-bytes) and percentage free. The total memory is usually 640KB (but now always).

### Windows Base Memory Free:

The largest base memory block available to a *Windows* application. It is usually used for data transfer buffers to real mode programs which cannot use memory above 1MB. If you run out of this (bad behaved Windows) you cannot load any applications and the system will slow down. Usually, you will have to restart Windows to correct this.

Most Windows programs keep a DCB in there as well, so, this value will decrease with the number of programs loaded.

This is not the same as you would see in a DOS box or just plain DOS (as reported by the **mem** command) ! (See below)

### DOS Base Memory Free:

This is the memory left free after all TSR drivers plus *win.com* (Windows loader) and *KrnI286/386* have been installed, and is the maximum free memory you could have in a DOS window. Currently (Windows 3.XX), the DOS shell takes a few kilobytes as well, so you end up with something less than this.

## **Extended Memory**

This is the memory *above 1MB*. For example in a 8MB system you would have 640KB base memory and 7MB of extended memory. See your DOS / Windows manual if you are unclear about this.

SAW displays the total / free memory (kilo-bytes) and percentage free.



## Virtual Memory

This is hard disk space used by the swap file to look as real memory. You can use it only in **Enhanced mode**, in *Standard mode* it is inactive. Virtual memory is good but very slow (the computer swaps data to / from disk), but usable if you can't get more physical memory. See your Windows manual for details. You should not set a swap file much bigger than your extended memory as you will not really like it when you get into it.

SAW displays the following information:

<b>Virtual Memory</b>	The size of your swap file
<b>Free Memory</b>	How much of the swap file is unused
<b>On Drive</b>	The drive on which the swap file exists
<b>Active</b>	If the swap file is currently used or not
<b>Swapping</b>	How hard your system is swapping

As long as swapping is *Insignificant / Rare* your system will be quite fast. If it is on *Medium* the system is becoming relatively slow. If it is on *Significant / Frantic* you should consider buying some RAM if you have to use it this way - otherwise close down some applications.

## Windows Memory Information Update Button

If you keep the *Windows Memory Information* window open while you start other applications or have applications running in the background then you may wish to click the **Update** button to get the new readings. You may have to press it three times to get a stable GDI reading.

**Note:** When you make SAW active then the update is done automatically for you.

## DOS Information

This window provides you with information about the DOS layer of your Windows subsystem. As Windows 3.XX still runs under DOS, it is important to know something about it. Even if Windows does not use DOS (Like Windows NT and the *hidden* DOS in *Windows 95*) there is a DOS emulator and you need to know its characteristics if you intend to run DOS programs - which you will still do for some time...

Please choose the topics that interest you and a brief explanation will be shown.

### Topics:

<u>Manufacturer</u>	<u>Name &amp; Version</u>	<u>OEM No.</u>
<u>User No.</u>	<u>Memory Location</u>	<u>Boot Drive</u>
<u>Comm Interpreter</u>	<u>DOS Prompt</u>	<u>Temp Directory</u>
<u>Programs Path</u>	<u>Allocation Strategy</u>	<u>Code Page</u>
<u>Disk Compressor</u>		

## DOS Manufacturer

The name of your DOS manufacturer. As there are a few types of DOS (*MS-DOS*, *IBM-DOS*, *Novell DOS*, *OS/2*, etc.) which are a bit different from one another it is good to know which is the one you have.

See [Name & Version](#) for more information. See [Trademarks](#).

**Note:** See the ***ReadMe*** file for more information about this.

## DOS Name & Version

The actual full name and version of your DOS as you bought it, for example *MS-DOS Version 5.00*. This is important to know as some upgrades (for example from 6.00 to 6.20) will work only if you have the same product and not a compatible one. You have to check before upgrading.

There are a few versions of DOS right now:

- **MS-DOS** - the standard one.
- **IBM-DOS** or **PC-DOS** - MS-DOS modified for / by IBM.
- **DR-DOS** or **Novell-DOS** - the *rebel* DOS which dared to be different.
- **OS/2** - The IBM's giant 32bit operating system.

### Notes

- The version is the true DOS version and not the emulated one (by SETVER).
- For OS/2 the emulated DOS version is shown or the true OS/2 version.
- While this will usually work, there are some problems with this (due to historical reasons, compatibility, etc.) so by typing **ver** or **ver /r** you can see your true DOS name & Version plus some other goodies...
- See the **ReadMe** file for more information about this.

## DOS OEM Serial No.

The identification serial number of your DOS copy in hexadecimal. This determines your DOS Manufacturer. For *MS-DOS* this is **FFh**. There are a lot of older DOS versions which may have all kinds of values.

If you have some peculiar DOS emulator then the manufacturer might not be detected correctly - but you can check using this number.

## **DOS User Serial No.**

The serial number of your particular copy of DOS. Usually this is **000000h** for OEM copies, but it could well be something else. Some use it as a batch or fix identification especially when updated versions are introduced (e.g. 4.00 to 4.01).

**Note:** There is no telling about what this number can be. Check for yourself and maybe drop me a line if a nice number shows up... This piece of information has been sidelined since DOS 5.00.

## DOS Memory Location

The type of memory DOS resides. In newer versions this can be:

- **Low** - Base (conventional) memory (low 640KB). If you have DOS 5.00 or higher you should consider loading it into HMA if you need more base memory for DOS programs. Even if you don't, it is advisable to put it in HMA as it makes '*good*' use of it.
- **HMA** - High memory just above the 1MB border (64KB). This is in fact extended memory, but due to the Intel architecture it can be used by DOS programs if they are loaded there. Usually this is the preferred choice.
- **ROM** - On portables DOS might even reside in ROM and not on disk. This is fine as it does not take time to load or get corrupted / infected, but a bit more difficult to upgrade. Some portables even have Windows in ROM ! (a smaller version). In fact they use PROM, EPROM or EEPROM as this is more economical.



## **Current Boot Drive**

The drive from which the computer was started.

This may be useful to know if you have a bunch of IDE / SCSI disks or many partitions. Some software (like old Windows 3.0) insist on having some files on this drive. Some new Windows programs may create temporary files there even if there is a TEMP directory !

## Command Interpreter

The name and path of your command interpreter (***Shell***). This is not so important on ordinary DOS, but on emulators is it good to know what is going on.

## DOS Prompt

Your DOS prompt command (the **PROMPT** variable). Do note that in Windows the variable **WINPMT** defines another prompt (see [Windows Prompt](#)), so in a DOS Shell you could have a different prompt to tell you that you are not in *straight* DOS.

You can also check out how your friend has such an interesting prompt (I've seen even flags, pictures) so you can have one too. Check out ANSI.SYS in your DOS manual to find out more.

## Temporary Directory

The directory that Windows will use to store temporary files (**TEMP** variable). You should have it set, otherwise it might litter your **\\windows** directory with temporary files (especially if you crash). You should also set it to the fastest / biggest free space disk. If you set it to a memory disk some programs may complain about insufficient space.

**Note:** There are only a few programs that use the **TMP** variable so you don't really need it.

## Programs Path

The path to your programs on disk. Most programs add their path when they install, but, if you launch them under Windows or first get into that directory there is no point to add the directories to the path. Your *\windows* directory is not required either as you can run only **win.com** from there.

Usually the path should include the **\dos** directory, your **\utils** (where you keep utilities like PC Tools or Norton Utilities), **\drivers** (network drivers or CD-ROM / scanner drivers) and / or **\menu** or **\batch** (where you start your programs from). It is advisable to keep the path short as DOS will be faster. See your DOS manual for more details.

## DOS Allocation Strategy

The strategy allocation used by DOS when loading programs into memory. This depends on your memory manager (EMM386, QEMM386, etc.). Usually if you do not use a memory manager this is **First Fit** - DOS loads them as they come. This may not result in efficient memory usage.

If you do use a memory manager this may be **Best Fit** or **Best Fit HMA (Only)**. This depends on the memory manager type / version. The full explanation is beyond SAW's scope. Do read a book though. See [Books](#)

Programmers should be familiar with this.

## DOS Active Code Page

The Code Page of your DOS subsystem. This may be different from your Windows Code Page which is shown by Windows Information. See your manual for a full explanation.

The Code Page is a set of characters available to your computer. For **USA** it is **437**. For **International** this is **850**. Other countries that have special letters in their alphabet may have a different one.

SAW shows you the current active code page and, in brackets, the code page active when you started your computer.

## Disk Compressor Information

A disk compressor is a device driver which intercepts disk requests from programs / OS to simulate a normal drive layout. However, in fact, the **compressed drive** is a big file on a different drive called **host drive** (uncompressed). All the files & directories are archived and compressed in that file. All the compression / decompression is done in real time.

The host drive is the real DOS drive, but the letters are changed at start-up, so you'll see the compressed drive as your usual drive, while the host drive can be hidden from you.

Usually, this compression increases your drive's capacity between 70-100% (or more if you're lucky) and is almost transparent. Works great with almost no performance penalty, but as your set-up is more complicated, there are more things to go wrong. Also, you need to keep the drivers on your start-up disk - should you need it - otherwise your whole drive is just a big file !

The good news is that disk compression will be integrated in newer operating systems like Windows 95, so you will not have to bother with all this.

If you use a disk compressor such as **Stack's Stacker** or **Microsoft's DoubleSpace** or **DriveSpace** the full name and version of your disk compressor will be shown. Depending on the type / version of the compressor, additional information may also be shown.

**Note:** SAW does not detect all disks compressors - only the few widely used.



## Windows Information

This window provides you with information about your Windows System. As Windows becomes much bigger than DOS, in future will not require DOS at all, (DOS will remain as a discardable loader, but it will be hidden from you) your Windows System is **very** important. If you do not have it configured correctly, the performance will be unsatisfactory. **Do** spend some time optimising it...

See [Utilities](#) for extra information.

Please choose the topics that interest you most and a brief explanation will be shown.

### Topics:

<a href="#">Manufacturer</a>	<a href="#">Name &amp; Version</a>	<a href="#">Mode</a>
<a href="#">Extra Info</a>	<a href="#">Build Version</a>	<a href="#">Win Prompt</a>
<a href="#">Windows Directory</a>	<a href="#">WinG Compatible</a>	<a href="#">32bit Compatible</a>
<a href="#">32bit Disk/File</a>	<a href="#">Code Page</a>	<a href="#">Country/Language</a>
<a href="#">Program Shell</a>		

## Windows Manufacturer

The Windows manufacturer. If you are running original **Windows** the manufacturer is, normally, *Microsoft* and if you have **OS/2** the manufacturer is *IBM*.

If you run other emulation operation system like **WABI**, **SoftPC** or something else then its name will be shown (if recognised). See [Windows Name & Version](#). See [Trademarks](#).

## Windows Name & Version

The name of your Windows System and its version. These are the types of Windows now:

- **Windows** (Plain -vanilla) - the older standard.
- **Windows for Workgroups** - Windows with network built-in. The standard.
- **Windows NT** - the 32bit giant Operating System. (*Windows on Windows*)
- **OS/2** - IBM's 32bit Operating System (*OS/2 for Windows* sub-system).
- **Windows 95** - the 32/16bit hybrid - kind of NT lite. Maybe the future standard.

Also, you might have **WABI**, **SoftPC**, **Wine** or some other emulation operating system. If SAW cannot determine its exact name, if it was made after SAW was released, then it will display **Emulation Windows**.

The Windows Version may not be the version on the package box, but the version of Windows that your *Windows System* emulates or the version it wants to be reported to programs.

If you have an *Emulation Windows*, the name of the Emulator will be followed by the true *Emulator* version (if known) in brackets, e.g. **OS/2 (2.10) Ver 3.10**, i.e. *IBM OS/2 Version 2.10* which emulates *Microsoft Windows Version 3.10*.

## Windows Mode

The mode Windows is now in. The three modes are:

- **Real** (all x86 compatible computers, not available in Windows 3.1 and higher)
- **Standard** (286+ computers, ATs, not available in Windows 95 and higher)
- **386 Enhanced** (386+ computers)

Also, you may see various mode information:

- **Paging Memory** - Windows uses paging memory (virtual) (Enhanced Mode).

## Windows Extra Information

This shows extra information about your copy of Windows. This can be one or more of the following:

- **Retail** - if your Windows is the usual type.
- **Debug** - if your Windows is the debugging version . The debugging version of Windows is usually used by *Beta Testers* and *Software Developers* as it offers more debugging facilities. It also comes with the Microsoft's SDK (Software Development Kit) and MSDN.
- **PenWin** - if your Windows uses the **Windows for Pens** extensions (mainly portables). Pen Windows contains special DLLs which take care of input recognition and require special hardware (for pen input).

## Windows Build Version

This is the build number/version of your copy of Windows. If your copy is just a normal retail copy this is the same as the Windows version. If you updated some of Windows modules (e.g. from 3.1 to 3.11) - this is the true Windows Version.

If the Windows is a *Beta* (pre-release) version (or *Debug*) this is the Version + Build number.

If the Windows is a *User Build* one (you are from Microsoft or other Windows developing companies :) then you may set it to whatever you want it to be - but (I hope) not somewhere else...

**Notes:**

- *Windows for Workgroups 3.11* may be detected as *Windows 3.10* but *Build* version 3.11.
- *Windows 95* (Ver 4.0), may be detected as *Windows 3.95*, but build version 4.0.
- This is the way Microsoft wants it to be - and SAW complies. There's a good reason behind all this.

## Windows Prompt

Your Windows prompt command (the **WINPMT** variable). Do note that in Windows the variable **WINPMT** defines another prompt (see [DOS Prompt](#)), so in a DOS Shell you could have a different prompt to tell you that you are not in *straight* DOS. This is useful as some DOS programs (like **chkdsk** and **defragmenters**) could wreck your Windows system if you run them in a DOS Shell - you may just forget about it !

So, you can have in *AutoExec.bat*: **set winpmt=Windows \$p\$g**

You can also check out how your friend has such an interesting prompt (I've seen even flags, pictures) so you can have one too. Check out ANSI.SYS in your DOS manual to find out more.

## 32bit Disk / File

This shows whether Windows is using 32bit software components bypassing DOS & BIOS. This is available only on 386+ computers. This results in a considerable speed increase. They should both be enabled. 32bit File is available only on Windows 3.11 and higher.

Please ignore this topic if you are running *Windows NT*. NT is using 32bit components always.

**32bit Disk** - if enabled, Windows is using the **FastDisk** devices (VxD) to talk to the hard drive directly bypassing the BIOS. By default Windows 3.1 supports the Western Digital standard for all except ESDI and SCSI. FastDisk enables you to get a little more performance boost and run more DOS applications.

**32bit File** - if enabled, Windows uses the **VFAT** device to manage the DOS file system without calling DOS. This results in a considerable increase of speed in file handling (databases, etc.). Also, program loads / saves, etc. will benefit appreciably.

## Windows Directory

The name of your Windows directory (default **lwindows**). This is not useful for a stand alone machine, but for a machine running a network Windows install , it may be somewhere else then you might think !

This directory should not be write-protected, as most Windows programs keep their configuration files (.INI) in there.



## WinG Compatible

WinG (pronounced *Win Gee*) is an optimised library designed to enable high-performance graphics techniques under Windows 3.1X, Win32s, Windows NT 3.5, Windows 95, and future Windows releases.

This enables fast graphic routines to be designed and run under Windows while, until now they were used only on DOS. Maybe a new generation of games will be created to ensure universal compatibility and remain as fast as in DOS while increasing the resolution and number of colours.

To run games as **WinDOOM** you need to get the WinG libraries, which are free, from your favourite archive site. Currently (end 1994) the latest version is **1.00**.

SAW will display the version as well if available.

## 32bit Windows Compatible

32bit compatibility in Windows 3.1X is not exactly what you might think. Windows 4.0 (*Windows 95*) should have full support for this, although it is not entirely 32bit. OS/2 and *Windows NT* are 32bit - see below:

### Preliminary Notes:

- If you have *Windows NT* or *Windows 95*, then of course your Windows is truly 32bit compatible ! You should ignore the information given below; this refers to ordinary Windows only.
- OS/2 is 32bit, but as it cannot at present run 32bit Windows NT applications, it is not considered to be 32bit compatible, although you can install Win32s or WinMem32.

Nowadays 32bit compatible is one of the following extensions:

- You have **WinMem32** which enables some programs to allocate memory using 32bits. This is the lowest of 32bit compatibility.
- You have **Win32s** which are a collection of DLLs which enable specially written programs to use 32bit operation / functions. This enables you to run *most* 32bit Windows (NT) applications. Windows 95 should be able to run Windows NT applications directly, but you will just have to wait...
- Some programs have used custom mechanisms to use 32bit things. This is specific, and not taken into account.

SAW will display the name of your 32bit extension and its version.

To run games like **WinDOOM** or Win32 applications you need to get the Win32s files (free) from your favourite archive site. Latest version (start 1995) is **1.25**.

**Note:** You cannot run OS/2 applications in Windows. Windows NT runs only OS/2 Version 1.XX applications (end 1994).

## Windows Code Page

The Code Page of your Windows system. This may be different from your DOS Code Page. See your manual for a full explanation. If you do not use special characters in DOS you might leave it to the default (**437**) and save some memory.

The Code Page is a set of characters available to your computer. For **USA** it is **437**. The **International** setting is **850**. For other countries, that have special letters in their alphabet, this can be different.

SAW shows you the current active code page and, in brackets the country it corresponds to. This should be the same as Country & Language setting. For **International** any country will do.

## Windows Country & Language

The country Windows was told it is running in (the one your keyboard was made for, of course) and the language if there are more choices.

For example, in the **UK**, the language may be *English American* or *English International*, whichever you prefer.

## Windows Programs Shell

The ID. (the DOS file name) of the programs shell. The programs shell is the program where you start the other programs for, default being **Program Manager (ProgMan)**. You might have another shell like *Dashboard* or *PC-Tools Office*, *Norton Desktop*, etc.

In Windows 95 the default is **Explorer**, though you can use Program Manager.

If you don't use *Program Manager* then you should keep it handy as some programs need to talk to it when they first install and may complain if your shell is not 100% compatible.

## Utilities

You can use other programs to get information / benchmarks. I will list some of them:

### **For DOS:**

- **MSD (Microsoft)** - utility provided with Windows 3.1 or higher or MS-DOS or IBM-DOS 6.0 or higher. Quite good and it is free (if you have MS-DOS of course...).
- **SI (Symantec)** - utility in PC-TOOLS or Norton Utilities for DOS. Both are quite similar and also provide benchmarking to check your system response time. PC-TOOLS Professional has exhaustive testing facilities as well. (Commercial)
- **CompTest** - a public domain utility that checks your processor / memory / disk performance. It also determines cache performance. Best for low-level benchmarking. Also includes FULL Turbo Pascal code, although you have to use DOS not Windows. Search for *CTEST259.ZIP* (ver 2.59). Freeware.
- **Cache** - Very like CompTest - public domain. Same thing but in German ! Also FULL Turbo Pascal code in German ! Would someone make an English version please ! Search for *CACHE.ZIP*. Freeware.
- **Config** - Very good Shareware detection program. Examines many low-level things and has a benchmark suite as well. Tests CD-ROMS too, motherboard. Search for *CONF727E.ZIP* (ver 7.27, English - German version too, is this a trend ?? :)

### **For Windows:**

- **WindSock** - Freeware utility. Benchmarks your system and provides neat information although only a little. Great to have. Ver 3.10 (August '94). Search for *WINDSOCK.ZIP*. Hey, no update - what's going on ???
- **WindSock SRM** - A system resource monitor, but one that has graph and VCR controls like facilities. Good for debugging. A bit hard to find *WINSRM.ZIP*. Shareware but more like Freeware.
- **WinTach** - Public Domain utility from Texas Instruments - benchmarks your Video System and it is often quoted in magazines. Good for benchmarking. Ver 3.10. *WINTACH.ZIP*.
- **UnixCorn** - kind of shell, but can do many things as well like system information, program startup (from DOS), etc. Has DOS&Windows interfaces, good help file and if you live outside USA is costs only a postcard ! Shareware. Search for *UNIXCN20.ZIP* (ver 2.0).
- **Windows System Information** - Sinfo is designed to help Windows programmers by providing information on many aspects of Windows. It can also be used as an advanced substitute for the Task Manager. Quite good, especially for debugging. Freeware. Search for *SINFO104.ZIP* (ver 1.04).
- **Norton Desktop for Windows, PC Tools for Windows** both have Windows information systems although I don't have them so I don't know. Anyone has a spare copy ? (Commercial)
- **Q/A Windows for IBM** - this is a special Windows diagnostic / information utility bundled with IBM PCs (ValuePoint). It is very good. It is free, but it only runs on new IBM computers (not old PS/2s) and you can get updates from IBM's ftp site. Commercial version available.
- Don't forget the **Control Panel and Windows Setup** utilities !
- **The System Analyst for Windows (SAW)** - Very funny, isn't it ? :)

**Note:** There are many very good shareware utilities that I know of, but I tend to concentrate on freeware

ones. The commercial ones mentioned are part of utility packages that may solve all your problems - but you have to pay for it. Good news is that *DOS 6.0+*, *Windows NT*, *Windows 95* comes with a much bigger array of utilities that in the past for free - considering that you need to buy the operating system anyway...

## About SAW

This window tells you information about the copy of SAW that you have. This is a standard feature that all Windows programs have.

You should see there:

- The program icon (top left hand corner).
- Full program name and version (top row).
- The copyright holder of this software (2nd and 3rd rows).
- Various information (6th and 7th rows).
- The name of the licence holder (8th & 9th row).
- The serial number of the software licence (10th row).
- Copyright notice (next rows).

**Windows** button - shows you information about the Windows operating system that you are running. This is the generic *About* window that is used by operating system utilities.

**Note:** SAW is a freeware program. The licenser place holder will be used in the future to give the program a *commercial* look. I don't intend to ask money for it as I think this kind of utility should be free. However, this **test** place holder will be used for other software in the usual manner.

See [Freeware Licence](#). Read the **ReadMe** and **Licence** files in the SAW directory for more information.



## Save Report

This window enables you to save a report about the system. You should see the following controls:

**Check boxes** that end in ...**Information** - These correspond to information windows about your system. To exclude specific information, uncheck the corresponding check box. All are checked by default.

For example if you want to include the *Summary Information* in the report then leave its check box checked. If you don't want it, uncheck it.

SAW remembers your selections. If you select the "*Information's*" required, press *OK* and save / print a report or just press *Cancel*, they will become the default for this session. If you want to make them the default for next time - goto "**File-Options**" and select "**Save Settings to disk**" and press *OK*.

Include Comments check box.

Include System Files check box.

**Send Report To** radio button. Specify if you want to send the report to a *text file* or to *print it* on one of the printers your computer has access to. Default is *Disk File*.

- **Disk File** - Save report to a text file. When you click *OK*, the normal **Save As** window will appear and you will be asked to give a file name (**saw.txt** by default). You can change the drive / directory where you want the file to be placed or the filename / extension. Click *OK* to save report or *Cancel* to abandon the operation.

The file can be read by any text editor (*Notepad*, *Edit*) or any word processor. Use a **fixed spaced font** (*Courier*, *System* or *FixedSys* in Windows) to keep the alignment correct.

Please note that a **full** report (including the initialisation files) might be quite large, **60-80KB** - and you will be unable to load it in *Notepad* (up to 64KB only).

- **Printer** - Print the report. When you click *OK* the **Printer Select** window will appear. Click *OK* to accept the default printer or select the one you want. Click **Setup** to configure the chosen printer. Click *Cancel* to abandon the operation.

Please note that the size of a **full** report might be very large, **5-6MB** (make sure you have enough free space if you spool files using *Print Manager*) and take up to **5-6** times longer to complete compared to a straight file save !

Using the *Generic/Text Only* printer driver will speed up the process considerably and reduce the size to that of a save, but some report lines may be longer than 80 characters, so the alignment might suffer. Nevertheless, it works as a *quick-and-dirty* solution.

You might also need **20-30** pages, so you should include only the information you consider necessary !

**Cancel** button - Abandon operation.

## Include Comments

As you may be preparing reports on many computers, you may need to add a few comments of your own. You can do just that in the window provided. You can type up to 255 characters. Comments are optional.

If you no longer want comments, uncheck the check box. Should you wish to include them again, your previous comments will not be lost, so you can edit them.

**Note:** For new paragraph type **Ctrl-Enter** not **Enter** alone. *Enter* will *close* the window. To bring it back click twice (don't double-click) on the check box.

## Include System Files

As computers rely on system configuration files you may want to include the system files in your report. This window enables you to do just that.

You can include the **autoexec.bat**, **config.sys** (DOS configuration) and **win.ini**, **system.ini**, **control.ini** (Windows configuration). To include a file check its check box (all are checked by default). To exclude one uncheck its check box.

The **Registration Database** is very important and may be regarded as a system initialisation file. In future versions of Windows, it should replace the current initialisation files.

### Notes:

- If you include all of them, the report file might be bigger than **64KB** so you will not be able to view it in *Notepad*. Use another text editor like DOS *Edit* (edit.com) or a word processor.
- Windows NT or Windows 95 Notepads do not have this disadvantage.
- In a word-processor, the text might appear badly aligned - you must use a **fixed spaced font** like **System** or **FixedSys** no *True Type* or *ATM*, etc. (except *MS-LineDraw*)

## DOS Device Drivers Information

This window gives you information about the DOS device drivers active in your system. A *device driver* is a program loaded into memory which allows DOS / Windows to communicate with a device (usually a peripheral).

DOS provides several of its own device drivers in the file IO.SYS (or IBMIO.SYS, etc.) but more device drivers are usually loaded in *Config.sys* (*DEVICE=* entry).

Windows has its own device drivers (VxDs). See the [VxD Info](#) topic for information about Windows drivers. **Only** the Real Mode (Virtual V86 in fact) Windows device drivers, like the EMS (EMM) memory driver, are listed.

### Topics:

<a href="#">Address</a>	<a href="#">Strategy Offset</a>	<a href="#">Interrupt Offset</a>
<a href="#">Device Type</a>	<a href="#">Device Name/Units</a>	<a href="#">Device File Name</a>
<a href="#">Device Attributes</a>		

**Note:** This information is mainly for programmers. All that could interest you are the name of the device drivers which should be familiar (See Device Name / Units). You should read a good (programming) book. See [Books](#).

## DOS Device Driver Address

The address at which the driver begins (device driver header). It is given as **Selector:Offset** (16:16bit). Note that this is the **Protected Mode address** not the *Real Mode* (*Segment:Offset*) address !

To obtain the real mode address get the *selector information* use a DOS (Real Mode) system utility program. See Run (MSD) and Utilities topics.

## DOS Device Driver Strategy Offset

A device driver has two interface parts: the *strategy routine* and the *interrupt routine*. The Strategy Offset gives the offset address of the driver's strategy routine. When DOS needs a device driver to perform a task it sends a *request packet* to the strategy routine and then invokes the interrupt routine to perform the actual operation.

The Strategy Routine address is therefore the **Selector:Strategy Offset** (16:16bit) (the selector is given in the Address). Remember, this is the protected mode address.

## DOS Device Driver Interrupt Offset

A device driver has two interface parts: the *strategy routine* and the *interrupt routine*. The Interrupt Offset gives the offset address of the driver's interrupt routine. When DOS needs a device driver to perform a task it sends a *request packet* to the strategy routine and then invokes the interrupt routine to perform the actual operation.

The Interrupt Routine address is therefore the **Selector:Interrupt Offset** (16:16bit) (the selector is given in the Address). Remember, this is the protected mode address.

## Device Driver Type

Device drivers are of two types:

- **Character** (char) - perform input / output *a character at a time*. Example: keyboard, screen, printers, etc.
- **Block** - perform input / output *a group of characters at a time* (a block). Example: disk and tape drives.

Each type of device driver has its own Attributes.

**Note:** The CD-ROM drivers nowadays (1994) have both *char* and *block* attributes; they are a bit of a hybrid.



## DOS Device Driver Name / Units

If a device driver is a *character device* then its **name** is shown. *Block devices* do not have names, but they have **number of units** which specifies the number of units the driver supports. See [Device Driver Type](#).

DOS Loads the following devices by default:

Device Driver	Device Information
AUX	Auxiliary (First serial port)
Block Device	Disks Driver
CLOCK\$	Clock (Timer)
COM 1	First serial port
COM 2	Second serial port
COM 3	Third serial port
COM 4	Fourth serial port
CON	Console (Display / Keyboard)
CONFIG\$	Windows 95 configuration
LPT 1	First parallel port
LPT 2	Second parallel port
LPT 3	Third parallel port
NUL	Non-existent Device
PRN	Printer (First parallel port)

Additional Devices loaded in Config.sys (optional):

Device Driver	Device Information
\$MMXXX0	UMB Driver (EMM386.EXE noems)
CON (ANSI)	Console Driver (ANSI.SYS)
EMMXXX0	Extended Memory Driver (EMM386.EXE)
IFS\$HLP\$	Installable File System (Win 3.11+)
MSCD\$\$\$	CD-ROM Extensions (MSCDEX.EXE)
SETVERXX	DOS Version Program (SETVER.EXE)
XMSXXX0	Expanded Memory Driver (HIMEM.SYS)

and many other various device drivers...

## DOS Device Driver File Name

The ID. (the DOS file name) of the device driver installed. The device drivers are loaded by the system as standard (See Device Name/Units) or by a program specified in *Config.sys*. Windows also loads extra device drivers, or replaces the ones loaded by DOS.

Standard device drivers loaded by DOS / Windows are located in the system files (IO.SYS, IBMIO.SYS and WIN386.EXE) so, therefore they do not have a file name of their own.

Other like XMS drivers (HIMEM.SYS) or SETVER (SETVER.EXE) do have a file name and this will be displayed without extension (usually .SYS or .EXE).

**Note:** The EMM driver can either be loaded by DOS by EMM386.EXE in which case EMM386 will be displayed as filename, or by Windows itself (VxD) in which case **no** file name will be shown.

## DOS Device Driver Attributes

Because the names of the attributes are too long, I concatenated them to fit in the information window. Here are the full names and their description:

<b>Input</b>	This character device is the standard input device.
<b>Output</b>	This character device is the standard output device.
<b>32bit-Sec</b>	This block device uses 32bit sector addressing.
<b>Nul</b>	This device is the NUL device.
<b>Clock</b>	This device is the CLOCK\$ device.
<b>FChar</b>	This character device supports Fast Character output.
<b>Gen-IO</b>	This block device supports logical drive mapping, generic IOCTL functions or both.
<b>Query</b>	This device supports IOCTL queries.
<b>O/C/RM</b>	This device driver supports functions: Open device, Close device, Removable Media.
<b>FAT-Req.</b>	This block device needs DOS to supply the first FAT sector for Build DCB.
<b>OTB</b>	This character device supports the Output Until Busy.
<b>R/W</b>	This driver supports IOCTL Read and Write.

## Books

If you want to know more about what the hell I'm talking about, you should try read some of the following books or **any** other you fancy. Yes, the list is a bit scarce I'm afraid - as a student I cannot afford too many !

For **DOS**:

- **Peter Norton: A guide to the IBM PC and PS/2.**
- **Peter Norton: Inside the IBM PC and PS/2.**
- **Kris Jamsa: DOS: The Complete Reference.**
- **Kris Jamsa: DOS Programming: The Complete Reference.**

For **Windows**:

- **Microsoft Windows Resource Kit** (3.1 or later).- available by **FTP** or from Microsoft.
- **Microsoft Windows 3.1 (or later) Programmers Reference (Vol. 1 - Overview).**
- **Microsoft Win32s Programmers Reference (Vol. 1).**
- **Microsoft Windows Journal.**
- **Windows Programming Journal.**
- **Inside Microsoft Windows.** (Windows 95)

Don't forget your DOS / Windows Manuals: come on, they're not that bad !...

## SAW System Errors/Warnings/Events

SAW monitors your system memory and Windows local heaps at regular intervals (about 10 seconds), see [Windows Memory Info](#). When these are low SAW will tell you - this is done while SAW is running (even when minimised).

SAW will also warn you of many other errors / warnings / events that may take place while it is active. They are classified in terms of their gravity in levels. Each level has certain attributes. Press **F1** to get help when an event has occurred (*the title bar or icon are **flashing***), you will be taken directly to the certain error level.

Lower level errors (the gravest) will always take precedence over higher (less important) errors no matter in which order they occur. Newer events will take (sometimes) precedence over older events if they have the same level.

### Topics:

[Level 1](#)

[Level 2](#)

[Level 3](#)

[Level 4](#)

[Level 5](#)

## **Level 1 : Errors/Warnings**

These are warnings or errors which must be taken into account immediately. You are advised to take the recommended measures to ensure that your system will remain stable.

Low GDI Resources

System Error

See SAW Errors/Warnings for the full list.

### **Low GDI Resources**

You have less than the programmed value (Alarm Options) GDI Heap free. You may experience re-drawing problems and *True Type / ATM* fonts may become unavailable. You may not be able to start new programs.

**Fix:** Close down Windows applications, especially word processors, spreadsheets and drawing programs. If the problem persists, save your work, exit and restart Windows.

**System Error**

A system error has occurred, but Windows cannot display the message itself. This may happen when you run buggy or bad behaved programs.

**Fix:** Save your work, exit and restart Windows. Approach program in question with caution.



## **Level 2 : Errors/Warnings**

These are warnings or errors which must be taken into account immediately. They are as grave as the level 1 ones, but occur rarely. You are advised to take the recommended measures to ensure that your system will remain stable.

Low System Memory

Low USER Resources

Low Base Memory

See SAW Errors/Warnings for the full list.

## **Low System Memory**

Free System Memory is lower than the programmed value (Alarm Options). Your system may be very slow and application switching will be very slow, too. You might not be able to start new applications or execute certain commands in a application.

**Fix:** Close DOS applications first and check. If you still need more memory, close down graphics editors or databases.

## **Low Base Memory**

This occurs when the free base memory is less than the set value (**5KB** default). You will not be able to start new applications and network / sound drivers may function abnormally. See [Base Memory Heap Info](#) to see which module eats up your memory and [Windows Memory](#) for overall information.

**Fix:** Close DOS applications first and check. If you still need more memory, you will have to remove some Windows drivers (*Control Panel*).

### **Low USER Resources**

You have less than the programmed value (Alarm Options) free USER Heap. You might not be able to execute certain commands in a application or open new files. The system may stall.

**Fix:** Close down open files or big Windows applications, especially word processors and spreadsheets. If the problem persists, save your work, exit and restart Windows.

### **Level 3 : Warnings/Events**

These are warnings which should be noted and maybe acted upon. They are not grave i.e. your computer will not crash but may work in a different manner. You should be aware of them as you may think they are due to other causes.

Device Mode Settings Have Changed

See SAW Errors/Warnings for the full list.

### **Device Mode Settings Have Changed**

Device initialisation information for a given printer driver or display driver has been changed. Therefore, your system may behave differently (depending on what you've done) and that is normal. Applications will take care of this but you should try a test run just in case you missed something...

## **Level 4 : Warnings/Events**

These are warnings / events which inform you that your system configuration has changed in a subtle way that may not be apparent immediately, but may lead to confusion about their nature. You should be aware of them too.

Win.ini has Changed

System Fonts Have Changed

See SAW Errors/Warnings for the full list.

## **Win.ini has Changed**

As you know the **Win.ini** file holds much of the information about your Windows system. When Win.ini changes, some changes **may** only take place when you restart Windows (it will tell you that), but programs (like SAW) may display incorrect information (the one in Win.ini) not the current one.

**Fix:** Re-start Windows if the changes are important. If you only changed the colours for example, then you can leave it.



## **System Fonts Have Changed**

You or an application has added / withdrawn fonts from the font pool. You should not use fonts that change in your documents as they will be replaced (not available) with other fonts.

Some applications add them only for their use but you may use them, if you really want to, but keep that application active while you print your document.

## **Level 5 : Events**

These are common events that occur most of the time and may not be very important but you should notice them all the same. Some may be of some interest to you, you never know...

System Time Has Changed

System Colours Have Changed

Resuming Operation

Resuming Normal Operation

Print Manager is Printing

See SAW Errors/Warnings for the full list.

**System Time Has Changed**

The time of your computer has been changed. If you only adjusted it that's OK, but if you have alarms set then you may have to re-program the alarm time.

### **System Colours Have Changed**

The system colours have changed. The desktop and all applications will re-draw so you can see your masterpiece...

## Resuming (Normal) Operation

This message is sent only on a system that conforms to the Advanced Power Management (APM) BIOS specification. The suspended mode is the state in which the greatest amount of power savings occurs, while all operational data & parameters are preserved. RAM contents are preserved, but many devices are likely to be turned off.

This messages shows that the system is recovering from the suspended state and you should wait to complete its start-up routines.

The *Normal* word is introduced when the system has entered in a critical suspended state before it could inform programs of the shut down. You should check that everything is OK.

### **Print Manager is Printing**

Print Manager sends this message when a print job is added or removed (printed or deleted) from the spool queue. On networks this could give you a discreet feel about the traffic going through the printer or when your document has finished printing.

## Trademarks

Now, this is not supposed to be a official list as you see in advertisements or packages (the small print), but a list of programs I've mentioned in this help file, their manufacturer, etc. This is true at the time of release (start 1995) **as far as I know**.

In this document, the following words in **BOLD** mean:

**Windows** - *Microsoft® Windows™* or *Microsoft® Windows for Workgroups™* versions 3.1 & 3.11. They are made by *Microsoft Corporation* and are the standard "Windows" on the PC. Windows for Workgroups includes peer-to-peer networking (very good) and version 3.11 is even better.

**Windows 4.0, Chicago, Windows 95** - *Microsoft® Windows™* version 4.0 as expected. This should be happening in future so no real information available - Final Beta released start 1995. The new Windows for home-users. Too many names, huh ?

**Windows NT** - *Microsoft® Windows NT™* versions 3.XX. A 32-bit, solid, version of Windows. Maybe the Windows of the future especially if you have the money. (Workstation & Advanced Server versions are separate).

**Daytona** - the new *Microsoft® Windows NT™* version 3.5. Much better than the 3.1 version (runs with 12MB !). Wait for 4.0 (Cairo) if you can...

**Cairo** - the new *Microsoft® Windows NT™* version 4.0 as expected. Should have all the features of Windows 95 but as it is Windows NT has its good and bad points. You should choose depending on what you can afford/need.

**OS/2** - *IBM® OS/2™* version 2.XX. Made by IBM (*International Business Machines®*) and is a 32bit operating system which runs Windows and OS/2 programs with a similar interface (8MB needed). New arrival - **OS/2 Warp** Ver 3.XX should be much better.

**OS/2 for Windows™** - Made by *IBM®*, same version as OS/2. This is an add-on for Windows which "upgrades" your system to full OS/2 at a smaller price since you already have Windows. Get this if you want OS/2.

**SoftPC** - *Insignia® SoftPC™*. This is a Windows emulator program for the *PowerMac™* computers and *Apple Macintosh™*. Very expensive, but OK. Made with Microsoft's blessing.

**WABI** - *WABI™*. This is a Windows emulator for *UNIX™* machines usually running *X-Windows™*. Made without Microsoft's blessing. Commercial software.

**Wine** - *Wine™*. Yet another Windows emulator for *Linux™*, but at the time of writing (start 1995) is free ! Not finalised at the moment, but OK if it goes the way Linux went...

**Linux** - *Linux™*. This is a high quality freeware UNIX clone Operating System that can bring you very low cost UNIX to your PC. If you are mad at DOS/Windows - try this; you might just like it... Source code available, so you can fix it yourself !

**DOS** - *Microsoft® MS-DOS™* any version that runs Windows (5.0 or 6.XX usually). Made by *Microsoft Corporation* and is the usual DOS except for a few machines; usually it is the standard to follow.

**PC-DOS, IBM-DOS** - *IBM® PC-DOS™* any version running Windows. MS-DOS with a few changes (like the name :). Usually found in PS/2s. Different utilities, though, which may make it better value.

**DR-DOS, Novell-DOS** - *Digital Research® DR-DOS™* versions 5.XX and 6.XX and *Novell® DOS™* version 7.0. A similar DOS but different in the same time and a better one. But MS-DOS is the standard

and you could run into incompatibilities with Microsoft applications :) (I wonder why)... Digital Research has been bought by Novell, but I kept the old name.

**i486, Pentium** - Intel® *i486*<sup>™</sup> and *Pentium*<sup>™</sup> - x86 processors. The oldest and still current standard in the x86 family to which all clones aspire to.

**Netware** - Novell® *Netware*<sup>™</sup> versions 3.1X or 4.0. This is *the* network operating system for large organisations and the old rival of Microsoft. Needs full blown servers but is very fast and secure. Still mostly geared towards DOS, though. Peer-to-Peer version available.

**SmartDrive, DoubleSpace, DriveSpace, HIMEM, EMM386** - MS-DOS utilities.

**Stacker** - Stack® *Stacker*<sup>™</sup> is a disk compressor like DoubleSpace/DriveSpace. Better, faster as the company is smaller, so more upgrades. Buy a faster/bigger hard disk instead...

**QEMM386** - Quarterdeck® *QEMM386*<sup>™</sup> is a HIMEM+EMM386 which gives greater performance should you wish to buy it. Includes a great system (memory mostly) information utility which should tell you anything...

**DOOM, WinDOOM** - iD Software® *DOOM*<sup>™</sup>, *WinDOOM*<sup>™</sup>, 3D games by iD Software which have been (and still are) very popular. DOOM was originally released for DOS with ports to SGI, Linux, SUN, etc. WinDOOM is the port to Windows which is handled by the Microsoft guys (beta still). There should be a OS/2 version too. Sequels: *DOOM 2*<sup>™</sup>, *Ultimate DOOM*<sup>™</sup>, *Quake*<sup>™</sup> (not yet !), *Heretic*<sup>™</sup>, *Heretic 2*<sup>™</sup> (with Raven Software®).

All Trademarks are acknowledged and are the property of their respective owners. **E.&O.E.** (errors and omissions excluded)

**Note to Software Manufacturers:** If you have any comments or you want the official thing put in, write specifying the exact phrase / description (max. 10 lines) you want introduced instead of mine, and I will try to put it in. Excuse my ignorance !?! :)



## Cache Disk Information

See [The Cache](#) for a general description. Here, we deal with disk-caches only. This cache may be a program or built-in the hardware. Most modern hard disks and CD-ROMs have an internal cache linked directly with the drive internal electronics. SAW does not detect such a cache.

### Software Caches:

Software Caches are [TSR](#) programs which intercept disk reads / writes and redirects them to a pool of memory. They are as fast as your main processor / memory system but require a few MB of memory plus some KB of base / UMB memory (if they are DOS programs). However, they add extra workload on the processor and if a crash occurs you may lose data if you use write cache as well. They are operating system dependent.

SAW detects only certain cache software programs. The information shown differs with the program and its version. Other caches may not be detected at all if they do not emulate one of the following:

<a href="#">MS-SmartDrive</a>	<a href="#">Norton N-Cache</a>	<a href="#">Norton N-Cache 2</a>
<a href="#">Norton SpeedDrive</a>	<a href="#">HyperDisk</a>	<a href="#">MS-VCache</a>

### Hardware Caches:

These are built-in the disk controller. They consist of a processor with its own BIOS and memory plus a BIOS which takes over the disk related BIOS functions. They are totally transparent, and you will see them only when you need to change some hardware configurations. They are operating system independent, crash-proof and require only a few KB of memory. They are expensive and not that fast since your processor is much faster than the controller's even when heavily used.

SAW does not detect all hardware caches, only the ones from a few companies:

[Promise Controller](#)

**Note:** You can use both software and hardware caches if you want although it is not feasible. If SAW detects a hardware cache it will *not* search for a software cache as well !

## Microsoft SmartDrive

This is a utility included with MS-DOS, PC-DOS and Windows. The latest version can cache CD-ROMs as well. It is therefore free and very good and you should use it - the latest version. Use *SmartDrv.exe* not *SmartDrv.sys*. Also, if you do not want Floppy and CD-ROM caches use the **VCache** utility (in Windows for Workgroups or later) which gives greater performance. See footnote.

**Note:** Norton N-Cache 2 & SpeedDrive emulate SmartDrive, so they return pretty much the same information, therefore, this topic deals with all of them. Some fields may be unavailable.

See your *DOS / Windows* manual for more details.

### Topics:

<u>Cache Type</u>	<u>Version</u>	<u>Actual Cache Size</u>
<u>Min. Cache Size</u>	<u>Element Size</u>	<u>Cache Hits</u>
<u>Cache Misses</u>	<u>Average Hit Rate</u>	<u>CD-ROM Support</u>

### List Box Fields:

<u>Drive</u>	<u>Read Cache</u>	<u>Write Cache</u>
<u>Buffered</u>	<u>DoubleSpaced</u>	

**Note:** The **SmartMonitor** program from the *Windows 3.1+ Resource Kit* provides much the same information, or you can type **smartdrv /s** at the DOS prompt.

**Note:** If you run Windows for Workgroups 3.11 or higher you should not use SmartDrive, but let the 32bit version (**VCache**) do the caching - this gives far greater performance.

## Microsoft V-Cache

Microsoft VCache is the only current (start 1995) cache which is different from any other cache. It is not a DOS Device Driver, but a 32bit Windows Virtual Device Driver (VxD). It can, therefore be used only when running Windows. It comes with Windows for Workgroups 3.11 and later.

VCache caches only hard disks at the moment (CD-ROMs too in Windows 95) and is a high performance cache. If you use Windows most of the time use VCache and not SmartDrive or other DOS cache. You can set up VCache in *Control Panel - 386 Enhanced - Virtual Memory*. See your Windows manual for more details.

### Topics:

<u>Cache Type</u>	<u>Version</u>	<u>Actual Cache Size</u>
<u>Min. Cache Size</u>	<u>Element Size</u>	

## Norton N-Cache

Norton N-Cache is a utility which comes with Norton Utilities package. Very good, but as SmartDrive is *free* you may not want it. Supports more customisation, so reading the manual is recommended.

It is safe to use with the disk utilities provided (like SpeedDisk) while other caches (including DOS's and hardware) are **not**, but the new version of the utilities may offer greater integration.

### Topics:

<u>Cache Type</u>	<u>Version</u>	<u>Status</u>
-------------------	----------------	---------------

**Note:** Type **ncache /?** for more information.

## HyperWare HyperDisk

This is also an utility which usually comes with the *HyperWare Utilities* or as a stand alone product (it is shareware I'm afraid). Very fast and very good too (386 optimised). Sometimes it is distributed with some software packages. Try it against SmartDrive and use the best one.

Read its manual for more information.

### Topics:

<u>Cache Type</u>	<u>Version</u>	<u>Buffers Used</u>
<u>Unflushed Buffers</u>		

### List Box Fields:

<u>Type</u>	<u>Cached</u>	<u>Verified</u>
<u>Stage Writes</u>		

**Note:** A more detailed account of its configuration is displayed at start-up.

## Promise HD Controller

This is a controller usually found on top-end machines. Some years ago only one or two computer manufacturers offered it in UK and it was dreadfully expensive (£399 - 4MB). Now, it makes sense to have one as you will have less trouble with memory configurations and the local bus versions are very fast.

*Promise Technologies®* manufactures such controllers. They are widely used for IDE/SCSI drives.

Please, read the manual as the default values may not work for you and you may not notice immediately that some programs won't work. Also, you may need to configure other cards / software programs although usually this is not needed.

### Topics:

<u>Cache Type</u>	<u>BIOS Address</u>	<u>Manufacturer</u>
<u>BIOS Version</u>	<u>Model/Name</u>	

**Note:** The **sidiag** program on your original disks carries many tests and displays a lot of information but requires a **HARD RESET**. **Do Not** run it within Windows with open files / applications as you will lose all unsaved data !

## Cache Type

This is one of the following:

- **Software Cache:** Manufacturer & Full Program Name, e.g. *Microsoft SmartDrive*.
- **Hardware Cache:** Disk Controller Type, e.g. *Intelligent Disk Controller*.

**Cache Version**

The program version of the software cache, or the version of the emulated cache if it emulates another cache package.



**Actual Cache Size**

The size in kilobytes (KB) of memory the software cache is using at the moment. Usually this should be around 1024-1028KB.

Usually this is extended memory (XMS) but some use expanded (EMS) or, in *desperate* conditions, base memory. Use XMS if possible.

**Minimum Cache Size**

The minimum size in kilobytes that the cache can shrink to if Windows needs more memory. Sometimes this is the same as Actual Cache Size depending on memory management procedures.

## Element Size

The size of the block of memory the cache moves / uses at a time. This should be around 1/100 the size of the cache memory (Actual Cache Size), but generally leave it to the default value. **8KB** seems a good value.

**Cache Hits**

The number of hits (times the data required was found in cache so **no** disk I/O was needed) recorded since the cache was started. See [Cache Hit Rate](#).

## Cache Misses

The number of misses (times the data required was **not** in the cache so disk I/O **was** needed) recorded since the cache was started. See [Cache Hit Rate](#).

### **Cache Average Hit Rate**

The percentage of hits since the cache was started. The highest this number, the better the cache program algorithm is and the faster your machine. You should try to tweak the cache to find the optimum parameters. Usually this is  $\geq 80\%$ .

**Note:** This value differs greatly with the type of applications running.

### **Cache CD-ROM Support Installed**

This shows if the CD-ROM drive (if present) is cached by SmartDrive. If no CD-ROM is present then this may mean that CD-ROM support is installed.

**Note:** Only versions 5.0+ (MS-DOS 6.2+) support this function.

**Drive Cached**

The DOS drive name for which the information is given.



## **Read Cache**

This shows if the drive is *Read Cached*, i.e. the cache tries to supply data from memory instead of reading it (slower) from disk. All local drives should be read cached with the exception of floppy disks and backup devices. Leave it to the default.

**Note:** A drive which is not read nor write cached is no cached at all.

## **Write Cache**

This shows if the drive is *Write Cached*, i.e. the cache writes the data to memory and returns control to operating system; later, when the system is idle, it writes it to disk. Only local, hard drives should be write cached.

**WARNING:** Should your system crash / become unstable you may lose all data which was not yet written to the disk. Do not use write cache if you have an unstable (regardless of reasons) system.

**Note:** A drive which is not read nor write cached is not cached at all.

## Buffered Cache

Some SCSI drives may need to be buffered to maintain memory integrity. Unfortunately, this slows down the cache considerably, so you may consider disabling it.

Use **smartdrv /s** to check if double-buffering is needed.

See your controller and cache manuals.

### **Double Spaced Drive**

If the drive is doubled using the DoubleSpace disk compressor, then the name of the file which constitutes your compressed drive and the drive on which it is placed will be shown.

### **Buffers Used**

The number of buffers the cache uses. This indicates the size of memory the cache program uses. Depending on buffer size, the bigger the better, but the total size should not be more than 2MB.

## **Unflushed Buffers**

The number of buffers that have not yet been written to disk. You should not turn off your system while there are still unflushed buffers or you may lose data !

**Drive Type**

The generic type of drive that the information refers to, e.g. *Floppy*, *Hard Disk*, *CD-ROM*, etc.

**Drive Cached**

This shows if the drive is cached or not. All local drives should be cached with the exception of floppy disks and backup devices. Leave it to the default.



**Verified Data**

This shows if the data is checked / verified after being written to disk to check for disk errors. It is safer to use it, but if you have a stable hard disk system you can disable it to gain speed.

**Stage Writes**

Stage Writing is another name for Write Caching.

## Program Status

The status of the program cache:

- **Enabled:** the program is running normally.
- **Disabled:** no caching is performed. Useful when you do low level disk operations like defragmenting or surface scan / repairs.

### **Controller BIOS Address**

The address in hexadecimal (segment) where the Controller BIOS is placed. This should not overlap **any** other BIOS and should not be shadowed.

**Controller Manufacturer**

The name of the controller manufacturer or the OEM Manufacturer depending on product.

### **Controller BIOS Version**

The BIOS Version of your controller. With controllers, this may not be very important as they emulate the System BIOS, but some device drivers may require certain functions specific to newer versions.

**Controller Model / Name**

The full name of the controller or the model number, depending on product.

## DOS Memory Information

This is how a Windows application *sees* DOS Real Mode memory. This window gives information about the memory available to the *old-school* DOS programs (running in Real / Virtual mode) as provided by Windows. However, the memory available in a DOS window depends on the PIF settings (see the Windows manual for more information). These are the **default** settings.

### Topics:

[Base Memory Info](#)

[UMB Memory Info](#)

[EMS Memory Info](#)

[XMS Memory Info](#)

**Note:** Again, most of this information is gathered by simulating a DOS real mode program, and as we all know, simulation is **not** reality. However, for the interested, it may be interesting...



## UMB (Upper Memory Blocks)

The UMBs are blocks found between memory locations B000-EFFF (or C800-EFFF), the area normally reserved for various BIOS ROMs. But, these do not occupy all the space, so we can use this *holes* to load programs and free base memory. In future, we will not care about this anymore, but we are still stuck with it...

UMBs are provided by an EMM driver like EMM386 or QEMM386. Windows does use UMBs internally for buffers / etc. but we are not concerned with that. Depending on configuration you may have around 160KB of UMB space. If you use EMS memory with a page frame then you lose 64KB. The more fancy cards you have the less UMB space remains free.

You can load TSR programs in the UMBs left using DOS commands. For MS-DOS 6.0+ owners, **MemMaker** comes to the rescue...

### Topics:

- **Total UMB:** The sum (in KB) of all UMB blocks provided by the memory manager (if any).
- **Free UMB:** The sum (in KB) and percentage free of all UMB blocks left free after (if) you loaded you programs. If you have 50%+ free, then you may consider loading more TSRs into UMB.

**Note:** Windows eats up most UMB blocks left (if any) so you are unlikely to see less than 0% free UMBs ! Therefore, check what really is the percentage of free UMB blocks with a DOS snooper (See Utilities) outside Windows ! It's the only sure way to find out...

## EMS (Expanded Memory Specification)

As Base Memory is restricted to about 640KB, a Lotus / Intel / Microsoft (LIM) collaboration developed a technique for making more memory available to programs. The EMS specification bypasses these limits by supporting memory cards (386+ computers usually emulate it with *emm386* or compatible memory manager) that contain 16KB memory pages (banks) that are mapped into the address space by the processor.

Applications must be specially written to use EMS, so most old DOS heavyweight programs use it. You will need to specify a generous amount of it in the PIF if you work with large data.

### Topics:

<u>LIM Version</u>	<u>Page Frame</u>
<u>Total EMS</u>	<u>Free EMS</u>

**Note:** If you don't use any programs which require EMS, adding *NoEMMDriver=on* in *system.ini* under *[386Enh]* will save you memory as the EMS driver will not be installed.

## **LIM Version**

The two kinds of EMS are differentiated by their specification version numbers:

- **3.20** - Moves data in 64KB blocks made up of 4 contiguous 16KB pages - the page frame. This arrangement runs into problems and is insufficient for multitasking.
- **4.00** - This can move data in 1-64 pages; also, it does not need a page frame. This is the latest version (end 1994) and probably the last.

Windows supports both versions.

**Page Frame**

The address of the page frame in hexadecimal if one exists. This is found in the upper memory, somewhere between C800-EFFF.

The page frame is a buffer made up of 4 contiguous 16KB pages which move data around. It is required for programs which want EMS version 3.20. See [LIM Version](#).

**Total EMS**

This is the total default EMS memory which Windows provides to a DOS application (usually 2MB). You can specify more if it is needed but you should specify the minimum required for the DOS application to work satisfactory.

**Free EMS**

Generally this should equal the total EMS, i.e. all EMS memory free. This is not used by Windows itself, but newer versions can balance XMS and EMS memory depending on requirement. Some high-end memory managers do this already.

## **XMS (eXtended Memory Specification)**

Windows uses XMS memory itself provided by **HIMEM.sys** (or compatible memory manager) accessing it directly. Windows also caches XMS through the memory manager, taking advantage of the attributes of Windows applications memory segments.

Also, Protected Mode DOS applications can use DPML and VCPI to access XMS.

### **Topics:**

<u>Versions</u>	<u>A20 Line Status</u>	<u>HMA Status</u>
<u>HMA Free</u>	<u>XMS Free</u>	

**Note:** Do not request more XMS than needed for an application in the PIF that it is necessary as Windows will slow down due to disk swapping to meet your demands (if less XMS is free than available).

## **Versions**

The versions of:

- The memory version provided by the XMS memory manager. Nowadays (1994) HIMEM provides XMS version 3.00. When Windows loads it takes it and provides XMS version 2.00 for applications. Windows 95 provides XMS ver 3.00.
- In brackets is the version of the memory manager itself which is relevant only if you want to upgrade it. Anyway as Window's memory manager is built-in you won't do this very often...



## **A20 Line Status**

The A20 line is the first address line which enables 286+ computers to address more than 1MB RAM. To run Windows HIMEM must install a A20 handler successfully; as you cannot run SAW outside Windows means that the A20 line is **On**.

## HMA Status

HMA (High Memory Area) is a zone of memory above 1024KB, size 64KB which can be accessed by a Real Mode program (like DOS) due to a quirk in the x86 processor architecture.

Usually you should let DOS load in HMA as most programs don't use it, and it will use it very efficiently as it was designed this way. Only one program can use it.

Therefore, the status is **In Use** or **Unused**.

## **HMA Free**

The HMA left free in KB and the percentage. Usually DOS should use it but it does not use exactly 64KB (~80%); if you have a TSR which takes exactly 63-64KB you may like to load that in HMA. Remember, only one program can be loaded in HMA however small !

**XMS Free**

The default XMS provided by Windows for a DOS application and the percentage that this represents out of the total XMS memory. Usually HIMEM takes almost all Extended memory and converts it to XMS.

## Sound Card Information

This window presents you with information about the various Windows Device Drivers installed for your sound card. Therefore, it is not card dependent but rather driver dependent.

If you do have a sound card installed, but no drivers - SAW will say that no card is installed as it does not check the actual hardware.

### Topics:

#### **Wave Driver Info**

Name & Version

Characteristics

Max. Sample Rate

Address

Interrupt

DMA Channel

#### **MIDI Driver Info**

Name & Version

Characteristics

Voice/Notes

MIDI Address

#### **AUX Driver Info**

Name & Version

Characteristics

**Note:** You need *Windows 3.1* or higher to use this function.

**Wave Driver : Name & Version**

The name of the Windows Driver for your sound card that plays sound files (.wav, .voc etc.). This contains the name of the sound card that you have or the one that your card emulates (if non-specific).

The version is also listed. This does not have to be the same as the Windows version, it is the version given by the sound card manufacturer. Ensure you have the latest version.

## Wave Driver : Characteristics

This can be:

- **8bit** - Plays / Records 8bit sound files.
- **16bit** - Plays / Records 8 or 16bit data files. (today's standard)
- **Mono** - Plays / Records Mono (1 channel) sound files only.
- **Stereo** - Plays / Records Mono/Stereo sound files. (today's standard)
- **n channels** - Special - Plays / Records on **n** channels (Mono/Stereo too).

This depends on hardware usually, but sometimes the driver you are using does not support all the card features - check manual to see if they are correct.

### **Wave Driver : Max. Sample Rates**

This is the maximum sample rates (in Hz = Hertz) that your card supports. The higher the better - sound quality is improved but more storage space required. Max. **44.1KHz** (as the CD-Player).

- The first number is the max. sample rate for Mono sound files supported by your card.
- The second (in brackets) is the max. sample rate for Stereo sound files if your card supports it.



**Wave Driver : Address**

The port number in hexadecimal that your sound card uses. Generally **220h**.

Handy to know so you can check that no other peripheral is using it; also you may be asked about it by DOS games.

### **Wave Driver : Interrupt**

The hardware interrupt number that your card uses. Only one peripheral can use it at a given time, so check that this is so. *SoundBlasters* use **Int 7** by default. Note that using interrupts higher than 7 may cause problems with DOS games.

See the [Hardware Interrupts](#) topic for more information.

**Wave Driver : DMA Channel**

The DMA channel(s) (8bit+16bit) used by your sound card. Usually this should be **1** for 8bit sound cards or 8bit simulation. For 16bit cards, the 16bit DMA channel is usually **5**. They are different as one handles 8bit memory access and the other 16bit. New cards may use only one 16bit DMA Channel.

No other peripherals can share the DMA channel with the sound card - otherwise lockups could occur. See [DMA Controller](#) topic for more information.

### **MIDI Driver : Name & Version**

The name of the Windows Driver for your sound card that plays sequencer files (.mid, .rmi). This contains the name of the sound card that you have or the one that your card emulates (if non-specific).

The version is also listed. This does not have to be the same as the Windows version, it is the version given by the sound card manufacturer. This may be different from the *Wave Driver Version* Ensure that you have the latest version.

## **MIDI Driver : Characteristics**

The characteristics of the MIDI driver:

- **MIDI Mapper** - The driver is a MIDI Mapper.
- **FM Synth** - This driver is a FM Synthesiser. (FX)
- **SQ Synth** - This driver is a Square Wave Synthesiser.
- **MIDI Port** - This driver is a MIDI Port driver as well. (should not occur)

**Note:** A driver may have some or all characteristics.

### **MIDI Driver : Voice / Notes**

This lists:

- The number of voices the driver supports.
- The number of notes the driver supports.

**Note:** If both are 0 and the driver is a MIDI port (See Characteristics) then **disregard** this information. SAW was unable to find the FM driver - it has found only the *MIDI Out Port driver* which connects to external MIDI devices.

**MIDI Driver : Address**

The port number in hexadecimal that your sound card uses. Generally **220h**. Usually this is the same as the Wave Driver: Address.

**AUX Driver : Name & Version**

The name of the Windows Driver for your sound card that adds to the *Wave & MIDI* drivers. This contains the name of the sound card that you have or the one that your card emulates (if non-specific).

The version is also listed. This does not have to be the same as the Windows version, it is the version given by the sound card manufacturer. This may be different from the *Wave Driver Version* or *MIDI Driver Version*. Ensure that you have the latest version.



### **AUX Driver : Characteristics**

This can be:

- **CD Audio** - Plays the sound output from a CD-ROM drive.
- **AUX In** - Plays sound output from the AUX Input jacks of the sound card.

## Tasks Information

This shows the names and characteristics of the currently running tasks (applications). It allows you to check how much memory each program uses and which modules it loads / uses. For a different point of view of this situation, see [Modules Info](#) and [Module Usage Info](#) topics.

A module may be a DLL, but sometimes even a driver (DRV extension) or an EXE module. System Fonts (FON) are in fact modules too.

Some of the information may be helpful to programmers in debugging, as well as for the ones that are curious to know who uses what, etc.

### Topics:

<a href="#">Program Name</a>	<a href="#">Program Icon</a>	<a href="#">Program Module</a>
<a href="#">Window Class</a>	<a href="#">File Version</a>	<a href="#">Windows Version</a>
<a href="#">Task Type</a>	<a href="#">File Name</a>	<a href="#">Modules</a>
<a href="#">Task Mem</a>	<a href="#">Custom Mem.</a>	<a href="#">Total Task Mem</a>
<a href="#">Eq. Task Mem</a>	<a href="#">Total Tasks</a>	<a href="#">Hog</a>
<a href="#">Applications</a>		

**Note:** SAW will automatically update the window when a new task is started (you run a program) or killed (you close a program), so you can leave the window *open*.

**Note:** SAW does not know when a program allocates or frees memory - therefore, the **memory used** fields may be old. To update, *re-select* the program from the combo-box: this will update the whole program information.

**Program Name**

This is the main (parent) window caption associated with the respective task. Note that a task may have different / more than one main windows - therefore, different information may appear depending on the state of the program.

## Program Module

This is the name of the module of the respective task. This may not be the same as the **.exe** file name of the program. For example, the **bpw.exe** file name may correspond to the **bpw0700** module...

### **Program Icon**

This is the class icon of the main window associated with the respective task. Some classes do not have a icon (if the program paints / animates the icon itself).



If no icon can be found, the icon on the left will be displayed instead.

### **Window Class Name**

This is the name of the class of the main window associated with the respective task. Most of the program interface characteristics are associated with it.

**Program File Name**

This is the full file name of the module associated with the respective task. The drive, path and .exe file name are shown.

## **Executing Tasks**

This is the number of all the tasks executing at present in the system. This includes hidden tasks, invalid, or multiple threads of the same tasks (when we have different modules).

## **Executing Applications**

This is the number of all tasks which are a normal application, i.e. have a main window, respond to Windows messages and are visible (i.e. not hidden).



## Modules used by the Program

The list box shows information about all the modules used directly or indirectly (through another module) by the given task.

### Topics:

- **Level:** The relation of the module with respect to the given task. **1** indicates a direct usage, anything greater means that the module is used indirectly (by the previous module with the level smaller by 1).
- **Module Name:** The name of the module.
- **Version:** The version of the module and revision.
- **Win:** The expected Windows version for which the module was designed for.
- **Memory Used:** The total (global) memory (in bytes) used by the current module - including code, resources, etc.
- **No. Loads:** This is the *load number* (usage) that shows how many other modules are using the current module. **1** may indicate a custom module. 1 is the smallest possible load number because at **0** Windows unloads the module.

**Note:** The level given is the level of the lowest encounter with the module. A module (like KERNEL) may be encountered many times (level 1, 2, 4, ...) - only the lowest level will be shown.

**Total Memory used by Program**

This is the total (global) memory (in bytes) that the respective program is using. This includes code, resources, etc. associated with the respective task.

### **Total Memory used by Custom Modules (DLLs)**

This is the total (global) memory (in bytes) that all the custom modules are using. This includes code, resources, etc. associated with all the custom modules.

A custom module is a module which is only used by the respective task or by a custom module which, respectively, is owned by the current task. Usually this means that the *load number* (usage) is 1 - but not always: some are loaded dynamically, so these loads may have to be excluded.

## Program File Version

This is the version of the program as listed in the *Version Information* field. The first number is the *major* version, the second the *minor* version and the third is the *revision*.

- For Windows utilities, the first two should be the same as the Windows version, e.g. *File Manager* from Windows 3.10 package might have version 3.10.103.
- Not all Windows programs have this information - *N/A* will be displayed in this case.

**Expected Windows Version**

This is the Windows version on which the program has been designed to run on. Should be equal or less than the current Windows version. Most programs expect a slightly lower Windows version so that more people can run it.

**Note:** Windows might refuse to run old Windows applications, e.g. Windows 3.10+ does not run Windows 2.XX software while Windows 3.00 does.

## **Task Type**

There are only two types:

- **DOS Applications**
- **Windows Applications**

To get a more detailed analysis, see [Module Information](#).

### **Private Task Total Memory**

This is the total memory taken by the task itself and all its custom modules. This is the sum of Total Task Memory & Total Custom Modules Memory.

However, this number becomes less meaningful if more than one application uses the same custom modules like *Word for Windows* and *Excel*. To get better information run only one of them when making measurements or see Total Equivalent Memory.

### **Total Equivalent Memory**

This is the equivalent memory taken by the task and all modules that it uses. This way, each task is said to have its share of the memory consumption.

E.g. if 2 tasks use a 100KB module, each can be thought of using 50KB of memory.

Therefore, if 2 applications (e.g. from a suite) use some big modules, each gets its share - while before only the memory used by the actual application was taken into account.

This number is shown in bytes (or kilo-bytes) and percentage of used memory that belongs to the respective task.



### Task Information: Hog

A task is thought to become a "**hog**" when it uses **50%** more memory than any other task should use in the system. Not all tasks are of the same size, but some are huge compared to others.

For example, if 10 tasks are running, they should each use about 10% of used resources on average. A margin of  $\pm 50\%$  is allowed. But if a task consumes over 15% (depending on system status) then it is considered to be a hog.

The actual formulae is a bit more complicated, but your common sense should reach at the same conclusion.

**Note:** If you run 2 or more big applications, none may be considered as a hog as they are all approximately the same size which is considered "*normal*".

## Module Usage Information

This window shows you a list of all modules loaded by Windows in the left list-box and which module(s) / task(s) use(s) the selected module in the right list-box. This will show you the relations between modules and tasks from another point of view than Tasks Info.

A module may be a DLL, but sometimes even a driver (DRV extension) or an EXE module. System Fonts (FON) are in fact modules too. See the topics shown below:

### Topics

<u>Name</u>	<u>Type</u>	<u>Code</u>
<u>Data</u>	<u>Loads</u>	
<u>Total Modules</u>	<u>Support Modules</u>	

**Note:** SAW will automatically update the window when a new task is started (you run a program) or killed (you close a program), or when a module is loaded / unloaded by a program or Windows, so you can leave the window *open*.

**Note:** SAW does not know when a program allocates or frees memory - therefore, the **memory used** fields may be old. To update, *re-select* the program from the left list-box: this will update the whole module information.

**Module Name**

The name of the module. Note that this may differ from the module file name (not shown).

## Module Type

There are a few types of modules used by Windows. These are:

- **DLL** - *Dynamic Link Libraries* which contain code / data which is usually used by more than one program (shared). Some DLLs contain interrupt handlers. Some contain different code / data and are selected by a program depending on the standard / type required.
- **DRV** - *Drivers* - device drivers which run along applications (Ring 3) and generally use call VxD services (Ring 0) for programs, e.g. buffering / handling data between programs and VxDs. They include printer, display, sound card, keyboard, mouse, etc. drivers.
- **FON** - *Fonts* - contain a font or fonts resource(s) which are used by Windows or other programs. Fonts contain only data, i.e. resources and no code.
- **3GR/2GR** - *Grabbers* - are Enhanced / Standard mode screen grabbers which allow Windows to simulate DOS sessions. They complement the video drivers and could be thought to be one.
- **EXE** - *Executable* - contain programs, i.e. tasks. These are the ones that *really run* and may call any of the other module types to execute a particular request.

**Note:** The EXE type modules shown into the left list-box (e.g. KERNEL, GDI, USER) are not executable program but rather internal Windows components, special DLLs in fact. This is an exception to the rule.

**Note:** The above list is not a definitive one, there are various other types introduced with every new *development* in software. Expect future Windows versions or alien systems to differ somehow.

**Module Code Size**

This shows the code size (in bytes) that this module has loaded at present. This are CPU instructions and specific data. Only modules which have subroutines contain this. (fonts do not).

## Module Data Size

This shows the data size (in bytes) that this module has loaded at present. Data is made up of resources, e.g. bitmaps, fonts, cursors, dialog boxes, strings, custom data, etc. They are used by the module itself or shared with others. Some modules contain only data (resources).

## Module Loads

This is the *load number* (usage) that shows how many other modules are using the current module. **1** may indicate a custom module. 1 is the smallest possible load number because at **0** Windows unloads the module (as it is not needed at present).

**Total Loaded Modules**

This is the total number of modules loaded in the system. These also contain tasks, orphan modules, or any other module that is loaded.



**Total Support Modules**

This is the total number of modules which are not executables, i.e. not tasks. They support programs and the main operating system modules.

## SAW Error Detected - Help

This topic should only be used if SAW displays the "*SAW Error Detected*" dialog box at start-up. This means that a previous instance of SAW has been terminated incorrectly due to an unknown event / error / etc.

However, SAW may not be able to detect the exact cause, so please ignore this message by clicking "**Continue**" if :

- You have **killed** SAW yourself when SAW was working correctly. To **kill** means to unload it from memory using a certain program instead of closing it down or ending the task.
- You have crashed SAW by inadvertently doing something, i.e. SAW crashed when another program has messed up the system, so SAW was unable to continue.
- You have powered off the computer **without** closing down SAW or Windows. *You should shut down Windows before turning the power off !*
- There was a power failure and the computer was turned off.

In case the crash was genuine, i.e. it's SAW's fault, then follow the instructions given in the dialog box.

- Click "**Disable**" if you were using one of the information windows when the crash occurred. This will disable detection procedures which may cause problems on some systems. Check to see if the error persists. If no, then go to "**File-Options**", check "**Save to disk**" check box and click "**OK**". Hopefully, this should be the end of your worries.

- If the above procedure does not solve you problem, this means that SAW is incompatible with something in your system. This can be due to may reasons, but you can check to see if SAW works when **no other** programs are running. If it does, then some program is the problem.

- If SAW still fails when alone, then there's nothing you can do short of not using the respective window, and waiting for a new version of SAW which may solve this problem. Sorry, and **drop me a line !**

## ROM Modules Information

This lists all the ROM Modules (BIOSes) that are installed in your computer. This may include the *Video BIOS*, *System BIOS*, *Disk-Controller BIOS*, *Network Card BIOS*, etc. A very short list should also be shown in the BIOS Extended Data Area.

This window will give a few general details about each one of them if possible. Do note that some of this information is already given for some BIOSes:

- **System BIOS:** see Computer Info, AMI BIOS Info topics.
- **Video BIOS:** see Video Info topic.
- **Cached HD Controller BIOS:** see Disk Cache topic.

### Topics:

<u>Address</u>	<u>Size</u>	<u>Date</u>
<u>Copyright</u>		

## Address

This is the address of the ROM module in hexadecimal. The physical address can be found by adding a **0** to the first value and a **F** to the second, i.e. *C000-C7FF* becomes *C0000h-C7FFFh*.

**Size**

This is the size of the module in kilo-bytes.

**Date**

The date at which the ROM software was created or the date at which this batch of ROM modules was created (blown).

### **Copyright Information String**

This string usually shown the name of the manufacturer / OEM manufacturer. Sometimes, the version, date, ID, of the ROM module is also included depending on manufacturer and the module type.

## Printers Information

This window shows you a list of the installed printers on your system, their drivers, and various information. Also, some information about the other printer-related libraries is shown.

Most of this information is shown by Print Manager as well. To change any of these settings you can use Print Manager or even your Printer Settings dialog box.

### Topics:

<u>Default Printer</u>	<u>Uses Print Spooling</u>
<u>Universal Drv</u>	<u>General Drv</u>
<u>Printer Name</u>	<u>Printer Port</u>
<u>Driver Name</u>	<u>Driver Version</u>



### **Default Printer Name**

The name of the current default printer. This is the printer that is used if you **Print** without selecting a printer first. You can have only one default printer which should be the one you use most of the time.

## **Uses Print Spooling**

Print spooling works by saving the printer information to disk and then sending it to the printer as fast as the printer can process it. Usually, the printer (unless fast Laser) cannot receive the data as fast as the computer can send it and you have to wait until printing is finished to get control to your application.

When Print Spooling is used, you get control of the application quicker and the printing is handled by Print Manager (or compatible) in the background. Do note that you may need appreciable free space on the temporary drive 5-20MB depending on document size / type, etc.

## **Universal Driver**

This is a library that most printer drivers use for most of their tasks. This means that it is easier to create / debug printer drivers. If any of your printers (generally dot-matrix & ink / bubble jets) use it, it is installed on your system.

The version of the library is shown if installed. This should be the same as your Windows version.

## **General Driver**

This is a library that some printer drivers use for some of their tasks. This means that it is easier to create / debug printer drivers. If any of your printers use it, it is installed on your system.

The version of the library is shown if installed. This should be the same as your Windows version.

**Printer Name**

This is the name of the printer the information refers to.

## Printer Driver Name

This is the DOS filename of the Windows printer driver (without extension). If you need to replace it - you will need this name. This is the actual name of the **.DRV** file in your **\windows\system** directory (if your Windows directory **is** *windows* - See [Windows Info](#)).

## **Printer Driver Version**

The version of your Windows printer driver.

This is usually the same as the Windows version itself. If it is smaller, you should get the new drivers (if available) as they should have a few improvements and usually they are more compatible (fewer crashes) - but this is not always the case.

If you can't get a updated version for your printer from the manufacturer, check out the Microsoft WDL (*Windows Driver Library*) on many BBS & FTP sites. See [Video Driver Version](#).

## Printer Port

The port your printer is connected to. This can either be a parallel port (**LPT**x, see [Parallel Ports Info](#)) or serial port (**COM**x, see [Serial Ports Info](#)).

Parallel ports can also be networked.



## Modules Information

This window shows you a list of all modules loaded by Windows and enables you to get detailed information about each of them. You should be able to find out most of what you need to know about any of them. For a different view, try [Tasks Info](#) or [Module Usage Info](#) topics.

### Topics:

<a href="#">Module Name</a>	<a href="#">Module Type</a>	<a href="#">OS Type</a>
<a href="#">Windows Version</a>	<a href="#">File Name</a>	<a href="#">Module Icon</a>
<a href="#">File Description</a>	<a href="#">File Version</a>	<a href="#">Product Description</a>
<a href="#">Product Version</a>	<a href="#">Copyright</a>	<a href="#">Company Name</a>

### Memory Usage Topics:

<a href="#">Memory Type</a>	<a href="#">Number</a>	<a href="#">Memory Used</a>
<a href="#">Percent Used</a>	<a href="#">Total Memory Used</a>	<a href="#">Hog</a>
<a href="#">Total Modules</a>		

### Associated Information:

[Module Exports Information Button.](#)

**Note:** 32bit / 16bit sub-systems: See the ***ReadMe*** file for more details.

## Modules Information - Hog

A module is thought to become a "**hog**" when it uses **50%** more memory than any other module should use in the system. Not all modules are of the same size, but some are **huge** compared to others.

For example, if 100 modules are running, they should each use about 1% of used resources on average. A margin of  $\pm 50\%$  is allowed. But if a module consumes over 1.5-2% (depending on system status) then it is considered to be a hog.

The actual formulae is a bit more complicated, but your common sense should reach at the same conclusion.

**Note:** If you have a few huge modules, none will be considered to be a hog, as this is thought to be *normal*.

## Module Exports Information

This window shows the function(s) exported (if any) by the respective module. Most DLLs, Drivers and even Tasks export a various number of functions - some of which are undocumented.

These are very useful to programmers. However, the parameters are not identified, so the declaration must be known in order to use any of the exported functions / procedures. This window only allows you to find out the name & index in the module.

### Topics:

<b>File Description</b>	A rather longer description of the module.
<b>Exported Name</b>	The name of the function / procedure exported by the module. Listed in alphabetical order.
<b>Index</b>	The index of the respective function within the module.
<b>Total Exports</b>	The total number of exported functions by the module (if any).

**Note:** 32bit modules are slightly different.

## Global Heap Information

This window presents your system's memory in yet another way. This time, you are shown a compacted breakdown image of your system memory. This may help in understanding the ways of Windows and many memory utilities as well as applications.

For other points of view, see [Windows Memory Information](#), [DOS Memory Information](#), [DPML Information](#) and [Base Memory Heap Information](#) topics. If you want a more detailed breakdown, see [LDT Information](#) topic.

### Topics:

<a href="#">Free Items</a>	<a href="#">Discardable Items</a>	<a href="#">Total Items</a>
----------------------------	-----------------------------------	-----------------------------

### ListBox Memory Topics:

<a href="#">Memory Type</a>	<a href="#">Number</a>	<a href="#">Discardable Mem</a>
<a href="#">Total Memory</a>		

### Summary Topics:

<a href="#">Free Memory</a>	<a href="#">Discardable Mem</a>	
<a href="#">Data</a>	<a href="#">Code</a>	<a href="#">Total Memory</a>

**Free Items**

The number of free blocks within the heap out of the Total Number of Items.

**Discardable Items**

The number of currently used memory blocks which can be discarded (freed) if necessary. They have not been freed as they may be needed and this improves performance, but they can be freed if memory is short.

**(Total) Discardable Memory**

The memory (currently used) which can be discarded if needed. This cannot be exactly considered free memory, but may be regarded this way when there is a memory shortage.

The percentage is also shown.

## Free Memory

The sum of all free blocks on the global heap. This is **NOT** the total free memory. It is the memory which has been used but is now free, but does not belong to the main *contiguous* free memory block of memory never used ! This is rectified when Windows does the *garbage collection*.



**Code**

The memory which contains code, i.e. CPU instructions, expressed in kilobytes and percentage.

**Data**

The memory which contains data, e.g. resources (bitmaps, dialogs, menus, pens, information, text, arrays, etc.) which complements the code but are not instructions. Generally, there is more data than code. Expressed in kilobytes and percentage.

## Local Descriptor Table Information

The *Local Descriptor Table* is an array which holds information about most of the selectors in the system for the given VM (Virtual Machine - Task). This depends on the task privilege level (ring). There is also a GDT (*Global Descriptor Table*) which can be accessed for more information. However this is not really available to less privileged tasks.

A selector (this is explained elsewhere) is a handle to a block of memory. This block of memory has various characteristics which must be stored in the GDT / LDT so they can be handled correctly. Otherwise, there would be mayhem. While for a user this is not very interesting, it may be for a programmer.

Yes, this is yet another breakdown of the system memory (see [Global Heap Information](#)) which may prove useful sometime. Remember, though, SAW runs at a low privileged level, so only little system information (Windows) is available.

### Topics:

<u>Descriptor</u>	<u>Size</u>	<u>Address</u>
<u>Ring</u>	<u>Attributes</u>	<u>Type</u>
<u>Owner</u>		

**Note:** As SAW & Windows are currently 16bit programs, there are some limitations. Please see the **ReadMe** file for more information.

## Descriptor

A descriptor is another name for a handle or selector, though there are some differences. This is the selector in hexadecimal format (16bit). **Not** to be confused with a *segment* which is the real mode address handle.

## Size

The size of the memory block represented by the selector in bytes. As you can see, most are limited to **64KB**. The ones bigger than 64KB belong to 32bit applications or the operating system, though usually the 16bit & 32bit systems are separated - therefore not much information available.

## **Address**

The **32bit linear** address of the memory block. Remember, this is a linear address which does not work like real mode stuff. This address can be anywhere in memory, and relocated as needed.

## Ring

The ring or privilege level (or DPL) shows how many rights are available to the descriptor. There are four ring levels in total (0 to 3). The Operating System (Windows) & VxDs run at ring 0 (highest privileged) while applications run at ring 3. In the future, the other rings will be shown as well.

Do note that SAW runs at ring 3, all the descriptors shown are ring 3 ones only. A program running at ring 0 would show information about all descriptors regardless of level.

## Attributes

There are many types of descriptors and each have their own attributes. However, this are a bit beyond the scope of SAW so, I'm afraid, I won't go into details. I will list just a few of them.

<b>Code</b>	Descriptor contains CPU instructions
<b>Data</b>	Descriptor contains resources
<b>R, W, R/W</b>	Read only, Write only, Read/Write allowed
<b>U, D, N</b>	Up, Down, Not Applicable (N/A)
<b>A, N</b>	Accessed, Not Accessed yet
<b>L, U</b>	Loaded, Not Loaded
<b>Owner</b>	

The module (or task) which owns the descriptor. See [Module Information](#), [Module Usage Information](#) and [Tasks Information](#) for a full list of modules running on your system.

If the owning module is not known, it can be assumed to be the operating system itself.





## Keyboard Information

This window gives you information about your keyboard. As the keyboard is still the most important input device, this information may be important to you.

This information is generic; SAW cannot determine crucial facts like feel and type (e.g. normal or special like *Microsoft's Natural Keyboard*) which determine the amount of RSI (Repetitive Strain Injury) that is associated with a particular keyboard.

### Topics:

<u>Model &amp; Type</u>	<u>Function Keys</u>	<u>Intercept</u>
<u>G/S Typematic Rate</u>	<u>Controller</u>	<u>Repeat Speed</u>
<u>Repeat Delay</u>	<u>KEYB Status</u>	

Therefore, choose your keyboard with care: chances are you will spend much time typing or at least using it as the primary input device.

## **Keyboard Model & Type**

The model and type of your keyboard (as installed). This may be not what you have exactly, but what you told Windows you have. However, this is usually correct.

Also the country for which your keyboard was made is shown. If you have a brand name keyboard then the manufacturer's name may also be shown.

## **Keyboard Function Keys**

The true number of keys your keyboard has. Most of today's keyboard are enhanced (12 function keys), but some have more or less keys. You will not be able to use those keys in a program which supports them.

## **Keyboard Intercept**

The Keyboard Intercept is the INT 15h, function 4Fh. Most keyboards have support for this function.

See [Software Interrupts Info](#).

## Keyboard Get/Set Typematic Rate

The Get/Set typematic rate is an enhanced functionality of some keyboards. This enables you to modify the Repeat Speed and Repeat Delay. These two variables can usually be changed from the BIOS. Some DOS programs also do it.

Under Windows, you can set both these variables from **Control Panel - Keyboard**.

**Note:** Some keyboard may support this function even if the query is negative. Usually only IBM keyboards support this query.

## Keyboard Controller

The controller that your keyboard has. Usually this is an **Intel 8042** or compatible. However, some keyboards may have different controllers which support different functions.

## Keyboard Repeat Speed

This shows how fast a key repeats when you press and hold it down. Roughly it means characters/second. In brackets you will see a characterisation of what this number really means.

For example **16** is a **medium** repeat speed; **22** is **fast** where **9** is **slow**. Without this you may not know what this number really means.

Under Windows, you can set both these variables from **Control Panel - Keyboard**.



## Keyboard Repeat Delay

This shows how long the computers waits after you press and hold down a key. In brackets you will see a characterisation of what this number really means.

For example **2** is a **medium** repeat delay (nearly the default) while **0** is **fast**, i.e. the computer will not wait at all. Without this you may not know what this number really means.

Under Windows, you can set both these variables from **Control Panel - Keyboard**.

## Keyboard KEYB Status

The **KEYB** is a DOS TSR program which enables you to specify the country for which the keyboard you are using is made. The default is USA so if you have a UK keyboard some keys are different. For countries which have a different alphabet this is almost obligatory.

SAW will show the KEYB status (Installed or not) and the version of the driver you are using. Some old drivers do not support the new countries added from time to time (e.g. some East European) so if you need support for a keyboard like this you should have the latest driver.

## Joystick Information

This window presents information about the joystick(s) connected to your system. Generally the PC supports **two** joysticks (you must have a two port game card) labelled **A** and **B**. The default one is A. The following information is shown for both joysticks:

### Topics:

<u>Status</u>	<u>X Value</u> <u>1st Button</u>	<u>Y Value</u> <u>2nd Button</u>
---------------	-------------------------------------	-------------------------------------

**Note:** Some devices that emulate a joystick like the **Gravis Advanced GamePad** and **Flight Sticks**; these appear to be two joysticks. For example the Flight Sticks have *throttle* control which looks like the X value of joystick B; the GamePad has *4 buttons* simulating the ones from joystick B.

For this reason the Status field is unique. You *should* know what you've plugged in !

## Joystick Status

- **Installed (OK)** when the joystick(s) is/are detected by the normal procedure.
- **Installed (not standard)** you do have some joystick(s) plugged in, but you may have *more than two* game ports (Sound Cards may have two game ports, and you may have a Multi-I/O card with one/two game ports as well). This may confuse some software (old games). Or, you may have some problems with the speed/ports. Check your system configuration.
- **Not Installed** - no joysticks detected.

## **X Value**

The Horizontal position of your joystick's stick. This is between 0..255 so you should calibrate your joystick to be at **128** when in **Central** position.

## Y Value

The Vertical position of your joystick's stick. This is between 0..255 so you should calibrate your joystick to be at **128** when in **Central** position.

## 1st Button

The status of your first button (the big one). **Up** is unpressed, **Down** is pressed. If you have it on **Turbo Fire** the reading may be inaccurate. Switch it off.

## 2nd Button

The status of your second button (the smaller/secondary one). **Up** is unpressed, **Down** is pressed. If you have it on **Turbo Fire** the reading may be inaccurate. Switch it off.



## Serial Ports Information

This gives you information about your serial ports. The serial ports are Input / Output *gates* which enable your computer to communicate with peripheral devices (mouse, modem, other computers) a bit at a time. Therefore, a byte (8bits) requires 8 cycles to be transmitted.

A PC can have up to 4 serial ports, but some cards support up to 8. SAW will show information only for the first 4 standard ports.

### Topics:

<u>Name</u>	<u>Address</u>	<u>Interrupt</u>
<u>Baud Rate</u>	<u>Data Bits</u>	<u>Stop Bits</u>
<u>Parity</u>	<u>Time Out Error</u>	<u>Time Out Value</u>
<u>UART Chip</u>	<u>Data Ready</u>	<u>Signal Detect</u>
<u>Ring Indicator</u>	<u>Data Set Ready</u>	<u>Clear To Send</u>

**Note:** In Windows you can change most of the communication settings in **Control Panel - Ports**. (or **Control Panel - System - Device Manager - Ports**)

**Note:** As Windows allows **only one** program to have control of a port, some information will be unavailable. Windows will warn you that two applications are trying to use the same port - you should give SAW **priority**.

The port used by mice / graphics tablets cannot be accessed, but then you don't need information about it ! Remember, this is Windows' view not DOS'.

## Serial Port Name

The name of the serial port. Each serial port has a name in the form **COMx** where x is 1..4. The first port is called COM1, the second COM2, etc.

There is another generic name for COM1 and this is AUX. See [Device Drivers Info](#).

Note that some *Accelerated Graphics Cards* have problems with the COM4 port so you may not be able to use it.

## Serial Port Address

The address is a location within the input / output address space used by a device. Each physical port has a unique address. If two devices have the same address then a conflict may occur.

The address is given in hexadecimal form. If you assigned a different address in Windows, this address will be shown, not the default one (if it is valid).

**Note:** If the address of a port is given as **0000h**, the explanation may be:

- The port is used by the mouse driver (See Mouse Info) or by another similar TSR device. In this case, this port is not available for use so, no information is available. This is normal and very frequent.
- The port may not be recognised by the BIOS even if it is there. See your Windows manual for more information about this problem.

## Serial Port Interrupt

An interrupt line is a hardware line over which peripheral devices ask the processor for services (send / receive data, etc.). Typically, each device uses an unique Irq line. However, as there are only 16 interrupts on a PC/AT (15 in fact) some devices share an interrupt but have different addresses. In this case you cannot use both of them in the same time.

By default COM1 and COM3 share IRQ 4 and COM2 and COM4 share IRQ 3.

The interrupt number is shown. If you assigned a different IRQ in Windows, this IRQ will be shown, not the default one (if it is valid). See [Hardware Interrupts](#).

## Serial Port Baud Rate

This is how fast the information is transferred to / from the computer in bits per second (**bps**). The default one is 9600bps.

Of course, if you have a modem you should connect with the fastest one, but take note your UART Chip may not support it or your computer's processor may be too slow in Windows.

Do note that this is not really an indication of how much data (in KB/s) is transferred. This depends on the other settings, type of data, form, etc.

## Serial Port Data Bits

The number of bits that are used for each character. If you have only plain text files (ASCII) with no fancy characters 7bits is OK, but 8bits are needed for **binary** files.

Other values are also available.

## **Serial Port Stop Bits**

The stop bits are not actually bits but timing units between characters so they can be distinguished. Therefore, you can have half stop bits (e.g. 1.5) not only integer values.

## Serial Port Parity

Parity is a method for checking the incoming characters / block for errors. Sometimes, transmission errors may occur, so the bad characters have to be re-send or re-built depending on the hardware / software.

There are many types of parity you can use or you can leave it to **none** if you want high speeds and you are sure of the transmission quality.

Remember, the computer also uses RAM parity checking to trap memory problems or bad connections or damaged memory chips. It happens !



## **Serial Port Time Out Error**

This specifies if the port is in *time out* mode, i.e. busy or not responding. The computer waits a certain time before giving up (see Time Out Value) then displays an error.

Your modem manual should give you full details.

## **Serial Port Time Out Value**

This specifies the time the computer will check the port and if the error remains then the Time Out Error will be displayed. Usually this is given in milliseconds.

Your modem manual should give you full details.

## Serial Port UART Chip

The UART (Universal AsynchRonous Transmitter) Chip is a converter which transforms the data to be send from parallel 8bit to the required standard and vice-versa. It also manages the COM port and the connections. There are a few chips on the market and they are:

- **8250** - (NS8250B) The standard in PC/XTs and early ATs.
- **16450** - The standard chip in PC/ATs. Bug fix for 8250, makes it more stable at high speeds.
- **82450** - A 16450 made by another manufacturer. Detected as 16450.
- **16550** - Added 16byte FIFO buffers, DMA transfer - higher transfer rates, more reliable.
- **16550A** - Bug fix for 16550. This is the one to get now (1994).
- **16550AF, 16550AFN** - Updated 16550A. The fastest yet (end 1994).
- Watch this space...

## Serial Port Extra Information

The fields *Data Ready*, *Signal Detect*, *Ring Indicator*, *Data Set Ready* and *Clear To Send* are usually associated with modems. There are general indications about what they do (the names are pretty well suited), but the modem's manual should tell you what their status should be so I won't get into details.

Anyway, I don't have a modem myself so I really did not *practice* tweaking them...

## Mouse Information

This window presents information about your mouse (rodent). As the mouse is almost obligatory in Windows it pays to have it in order. Usually this means to clean it regularly, but a good mouse driver and settings are just as important.

Most of the information given below is about the driver itself more than the mouse - anyway most are Microsoft compatible.

### Topics:

<u>Status</u>	<u>General Type</u>	<u>Manufacturer</u>
<u>Hardware Type</u>	<u>Interrupt</u>	<u>Port</u>
<u>No. of Buttons</u>	<u>Buttons Swapped</u>	<u>Driver ID.</u>
<u>Driver Version</u>	<u>Driver Type</u>	<u>Cursor Type</u>
<u>Pointer Style/Size</u>	<u>Driver Language</u>	<u>Tracking Speed</u>
<u>Double Click Time</u>		

Unfortunately SAW cannot determine the *quality (feel)* of the mouse. That's left for you to determine: a good mouse can work wonders in Windows especially when you do precision work like CAD or DTP.

## Mouse Status

The status of your mouse. This can be **present** (recognised by Windows) or **absent** (not present or incompatible driver).

**Note:** If the mouse is not present all the following fields will be unavailable.

## General Mouse Type

The general type of your mouse e.g. *Microsoft Compatible or IBM PS/2*. This is the full name of the mouse driver and may include the name of the manufacturer as well.

**Note:** This is the generic type only, not the actual type of mouse.

## Mouse Manufacturer

The name of your mouse manufacturer or the name of the manufacturer whose mice your rodent emulates.

**Note:** All Microsoft compatible mice are detected as Microsoft made as long as you use a Microsoft driver. But the Windows Set-up program may not install the DOS mouse driver if you do not have a original Microsoft mouse...



## Mouse Hardware Type

The type of the connection between you mouse and the computer. This can be:

- **Serial** - connects to a serial port (COM)
- **Bus** - connects to the bus via an interface (Bus Mouse)
- **InPort** - connects to an InPort (special connection)
- **PS/2** - connects to the PS/2 style port on IBM PS/2 computers
- **HP** - Hewlett Packard mouse

**Note:** Some video cards support a bus mouse. You should try to use it if you have a lot of serial devices.

## Mouse Interrupt

The interrupt number that your mouse uses. This is the interrupt of a serial (COM) port. See [Serial Port Interrupt](#) and [Serial Ports Info](#).

**Note:** Non serial mice (like PS/2) do not have an interrupt number.

## Mouse Port

The name port that your mouse is using. This serial port will be unavailable in Windows (naturally). See [Serial Ports Info](#).

**Note:** Non serial mice (like PS/2) do not have an normal COMx port. In this case the generic name of the port is listed (e.g. PS/2).

## Number of Mouse Buttons

The number of mouse buttons that are recognised by the mouse driver, not the number of the buttons on the mouse.

Some 3 button mice can emulate both *Microsoft* or *Mouse Systems* mice; in Microsoft mode they can use 2 buttons and in Mouse Systems mode all 3 buttons. Check your mouse manual.

## Mouse Buttons Swapped

If the buttons on the mouse are swapped or not. For example if you use it with your left hand you might want to swap the mouse buttons (the left button becomes the right button and vice-versa). This is OK until a right handed person tries to use it... :)

You can change this in **Control Panel - Mouse**.

## Mouse Driver ID.

The DOS file name of your mouse driver. The full name is given at General Type. This is important as this is the actual mouse driver. The extension is **.DRV**.

Sometimes the DOS mouse driver has the same ID. but the extension **.SYS** if you load it in Config.sys or **.COM** if you load it in AutoExec.bat.

## **Mouse Driver Version**

The version of your mouse driver. You should always be using the latest version as it usually includes new features (especially under Windows).

The Microsoft compatible DOS mouse drivers that come with Windows are usually left on disk when you update Windows. Also various other drivers are included. You should switch to them. See your Windows manual.

## Mouse Driver Type

The type of your DOS mouse driver. This can be:

- **TSR** - (Terminate and Stay Resident) driver, i.e. stays in memory and services the mouse when it is needed. Most new drivers use this.
- **Device Driver** - See [Device Drivers Info](#). Not many mouse drivers are of this type, but... This must be loaded in Config.sys. Windows 95 uses a VxD instead.

**Note:** As far as you are concerned there is no difference between them.



## Mouse Cursor Type

The type of cursor you are using. In Windows this is almost always **Graphics** (we are in graphics mode), but it can also be **Hardware** i.e. your graphics adapter handles it in hardware.

## Mouse Pointer Style / Size

The style and size of the mouse pointer. This can be configured from the mouse driver set-up panel (See **Control Panel - Mouse**).

The style may be:

- **Normal** - White cursor.
- **Reverse** - Black cursor.
- **Transparent** - Transparent cursor.

The Size may be **Small**, **Medium** or **Large**.

**Note:** Not all mouse drivers allow configuration of the mouse pointer. It is, however, hoped that this functions (already available on very expensive mice) will be introduced in the standard drivers. This has nothing to do with your mouse, just the driver...

## **Mouse Driver Language**

The language for which the mouse is configured. This should be the same as your keyboard language (See [Country/Language](#)) unless you need it otherwise.

## Mouse Tracking Speed

The relative tracking speed for which your mouse is programmed. The tracking speed is the rate at which the mouse pointer travels across the screen. In brackets a characterisation of this speed is given.

This may be:

- **Slow** - The pointer travels slowly for large hand movement. Good for detail work.
- **Medium** - Default. For normal tasks.
- **Fast** - High speed pointer movement for small hand displacement. Good when space is at a premium, but you may fail to see the pointer on some portables.

You can change this settings in **Control Panel - Mouse**.

## Mouse Double Click Time

The maximum amount of time (in milliseconds) that can pass between two mouse *clicks* to be registered by Windows as a *double-click*. An characterisation is given in brackets. Also, the value from **win.ini** is given in the second brackets.

If you notice that your double-clicks become clicks (when you are tired) increase this value.

You can change this settings in **Control Panel - Mouse**.

## Parallel Ports Information

This gives you information about your parallel ports. The parallel ports are Input / Output *gates* which enables your computer to communicate to peripheral devices (printers, backup devices other computers) a byte (8bits) at a time. Therefore, it is faster than a serial port (at same speed).

A PC can have up to 4 parallel ports, but only 3 may be used. SAW will show information only for the first **3** standard ports. Usually parallel ports could communicate only one way but now bi-directional ports have begun to appear.

### Topics:

<u>Name</u>	<u>Address</u>	<u>Interrupt</u>
<u>Access</u>	<u>Type</u>	<u>On Line</u>
<u>Paper Out</u>	<u>I/O Error</u>	<u>Time Out</u>
<u>Time Out Value</u>	<u>Busy</u>	<u>Acknowledge</u>

**Note:** In Windows you can configure the printers in **Control Panel - Printers**

## Parallel Port Name

The name of the parallel port. Each parallel port has a name in the form **LPTx** where x is 1..3. The first port is called LPT1, the second LPT2, etc.

There is another generic name for LPT1 and this is PRN. See [Device Drivers Info](#).

## Parallel Port Address

The address is a location within the input / output address space used by a device. Each physical port has a unique address. If two devices have the same address then a conflict may occur.

The address is given in hexadecimal form.

**Note:** If two ports have the same address it means that one is *redirected*. See [Access](#).

**Note:** If a port has address **0000h** then it may have been not recognised by the [BIOS](#).



## Parallel Port Interrupt

An interrupt line is a hardware line over which peripheral devices ask the processor for service (send / receive data, etc.). Typically, each device uses an unique Irq line. However, as there are only 16 interrupts on a PC/AT (15 in fact) some devices share an interrupt but have different addresses. In this case you cannot use both of them in the same time.

By default LPT1 and LPT 3 share Irq 7 and LPT 2 (and LPT4) share Irq 5. The interrupt number is shown. See [Hardware Interrupts](#).

**Note:** If you have a Sound Card assigned to Irq 7 (as default) you *may* experience sound distortions if you print to LPT 1/3 and play sound in the same time !

**Note:** Some Sound Cards compatible to *Disney Sound Source* create a *virtual* parallel port which *will be* listed by SAW and other programs. Do not send anything down it, because you will just lose that data. See your card manual.

## Parallel Port Access

The way that Windows **should be** talking to that port. It may not be the way Windows is doing it ! This is based on a documented algorithm. The values are:

- **Direct** - The port is physical and OK. Windows can talk faster to it (you may check the *Fast Printing Direct to Port* check box in the **Printer Connect** dialog).
- **Network** - This printer is *network redirected*, i.e. the peripheral is not attached directly to the computer but to another computer (the *Print Manager Computer*) on the network. Windows should talk to it through DOS interrupts (slower).
- **DOS** - Window should talk to it using DOS interrupts (slower). This may be because this port is non-standard or because it *redirected* (maybe with the **MODE** command - see the DOS manual).
- **No** - This port should **not** be used. The BIOS may not recognise it or there's something wrong with it.

## Parallel Port Type

This shows the type of parallel port your machine is equipped with. Until recently, all parallel ports were almost the same, but now, better communication is required which has introduced a few new types.

These may be:

- **EPP** - *Enhanced Parallel Port* - Is a new type of port which has faster throughput (good for external, parallel plugged devices), dual way transmission, etc. With this port, you can get *live* information about your printer status - even the paper position. Complies to IEEE 1284.

If you have fast external hard disks or CD-ROMs plugged into the parallel port you will certainly want one. Also, GDI printers (in which the page mask is created by the computer as the printer does not have a *big* CPU) may require this. All new machines should have it.

- **HP** - *HP Extended BIOS Parallel Port Device* - Is a BIOS extension on *Hewlett-Packard* machines (*Vectra* range) which has extra parallel-port instructions. This should result in higher speed.
- **Normal** - The normal parallel port (standard) and any other proprietary designs or any new ones which are released.

## **Parallel Port On Line**

The port is on line, i.e. ready to receive more data. This also shows if the connected printer is OK. If no peripheral is connected this should be **Yes**, i.e. the port is OK.

See Paper Out and Busy.

## Parallel Port Paper Out

The connected printer is *Out of Paper* or it is not responding. If no peripheral is connected the this should be **Yes**, i.e. the port is not connected.

See On Line and Busy.

## **Parallel Port I/O Error**

If an Input / Output Error has occurred. This may happen due to the cable or the connection. On a network redirected port this may mean that there was a network problem.

## Parallel Port Time Out

This specifies if the port is in *time out* mode, i.e. busy or not responding. The computer waits a certain time before giving up (see Time Out Value) trying then displays an error.

## Parallel Port Time Out Value

This specifies the time the computer will try to check the port and if the error remains then the Time Out Error will be displayed. Usually this is given in seconds.

**Note:** This is the DOS value not the Windows one. In Windows the *Print Manager* will wait the values specified in **Control Panel - Printers - Connect - Device Not Selected / Transmission Retry**.



## Parallel Port Busy

The port is busy, i.e. cannot receive more information. This happens if the printer buffer becomes full so the computer must wait before continuing.

If the port is a virtual port (See Interrupt) then this should be **No**.

If no peripheral is connected the this should be **Yes**.

See Paper Out and On Line.

## Parallel Port Acknowledge

The device has acknowledged a specific command. As we do not send any (just checking its status)- usually this is **No**. However, should this be after an error / busy event occurred it may be useful.

## DPMI Information

What is **DPMI** (DOS Protected-Mode Interface) ? According to the DPMI specification, the term real-mode software refers to code that runs in the low 1-megabyte address space and uses **segment:offset** addressing. With Windows 3.0 and later in protected mode, so-called real-mode software is actually run in virtual-8086 mode. However, because virtual-8086 mode is a close approximation of real mode, both are referred to as real mode in this topic.

Windows 3.0 and later in 386 enhanced mode supports DPMI version 0.9. Windows 3.0 and later in standard mode supports a subset of DPMI that enables applications to call terminate-and-stay-resident (TSR) programs and device drivers running in real (or virtual-8086) mode. Windows also provides specific DPMI APIs which have to be called directly. Some are not documented.

As you can see DPMI is mainly for programmers and such. You, as a user of the Windows Operating System you don't really care what is going on. The programmers may find it useful to know this information. The fields shown should be known to programmers. See the original **Microsoft Ver 0.90, Intel 1.00** or later **DPMI Specification** for more information.

### Topics:

<u>DPMI Version</u>	<u>DPMI Type</u>	<u>Reflected Irqs</u>
<u>Virtual Memory</u>	<u>Virtual Interrupts</u>	<u>Largest Free Block</u>
<u>Max. Pages Avail</u>	<u>Max. Pages Lock</u>	<u>Lin Address Space</u>
<u>Total Unlock Pages</u>	<u>Total Free Pages</u>	<u>Total Phy Pages</u>
<u>Free Lin Addr Spc</u>	<u>Paging File Size</u>	<u>Page Size</u>

**Note:** If *Virtual Memory* is not supported, only some of the fields will be filled. The rest will display **N/A** (Not Applicable).

**DPMI Version**

The version of the emulated DPMI standard. This is the version of the DPMI services supported. Note that this is not the version of any operating system that supports DPMI.

## DPMI Type

The type of the DPMI server. This may be:

- **32bit** i.e. a 80386+ DPMI implementation. (*Enhanced Mode Windows*)
- **16bit** i.e. a x86 DPMI implementation. (*Standard Mode Windows*)

**Reflected Interrupts**

This is the mode the processor is returned in for reflected interrupts. This may be *Real Mode* or *Virtual Mode* depending on DPMI Type.

## **Virtual Memory Support**

If the DPMI implementation supports Virtual Memory or not. In Windows *Enhanced Mode* Virtual Memory is supported.

**Virtual Interrupts**

The state of the virtual hardware interrupts in the current system. This may not reflect the true state of the actual interrupts as they are virtualised by the system.



**Largest Available Free Block**

This field specifies the largest block of contiguous linear memory in bytes that could be allocated if the memory were to be allocated and left unlocked.

**Maximum Pages Available**

This field specifies the number of pages that could be allocated. This is the value returned by Largest Available Free Block

**Maximum Pages Lockable**

This field specifies the largest block of memory in pages that could be allocated and then locked.

**Linear Address Space Size**

This field specifies the size of the total linear address space in pages. This includes all linear address space which has already been allocated.

**Total Number of Unlocked Pages**

This field specifies the total number of pages that are currently unlocked and could be paged out. This value contains any free pages.

**Number of Free Pages**

This field specifies the number of physical pages that are currently not in use.

**Total Number of Physical Pages**

This field specifies the total number of physical pages that the DPML host manages. This value includes all free, locked and unlocked physical pages.

**Free Linear Address Space**

This field specifies the free linear address space available in pages.



**Size of Paging File / Partition (Swap File)**

This field specifies the size of the DPML host's paging partition or file in pages.

**Page Size**

The size of a single memory page in bytes as used by the DPML host. (Block Size)

## Software Interrupts Information

A software interrupt is a group of functions which enables a program to execute a certain task from the group. Each interrupt supports a certain amount of related functions. The PC has room for **256** interrupts pointed to by *vectors* i.e. their addresses. There are two addresses, one for *protected* mode and one for real mode. See [DPMI Info](#).

As your computer runs, several interrupts occur without your knowledge such as the *clock interrupt* (~18.2 times/sec), while others like the *divide by zero* occur only when an error has taken place and the handler informs you (if needed).

When an interrupt occurs, the computer suspends the operation it was performing, saves its state, locates the address and branches to the code that handles the interrupt (the *Owner*). When the interrupt processing is complete, the previous state is restored and the computer resumes the operation as if nothing has happened.

### Topics:

<u>Irq. No.</u>	<u>Address</u>	<u>Description</u>	<u>Owner</u>
-----------------	----------------	--------------------	--------------

See [Plug-and-Play](#) topic.

## Interrupt Number

The interrupt number in this way (the usual way):

- There are **256** *Software* interrupts. The number is given in *hexadecimal* (**00h-FFh**).
- There are **16** *Hardware* interrupts. The number is given in *decimal* (**0-15**).

## Interrupt Address

Two addresses (both in hexadecimal form) are shown as follows:

- **Real Mode - Segment:Offset** form (DOS/Real Mode Windows).
- **Protected Mode - Selector:Offset** (16:16bit) (Enhanced/Standard Mode Windows).

**Note:** 32bit programs use a Protected Mode *Selector:Offset 16:32bit* address; as SAW is 16bit it shows the Protected Mode addresses as 16:16bit. There's nothing wrong with this.

**Interrupt Description**

The *name* of the interrupt if it has one, or a brief description of what is supposed to do otherwise.

## Interrupt Owner

As there are two addresses, there are two *owners* or programs which own the specific interrupt (handlers). The name of the program is given or :

- **Free** - (Real Mode) - The interrupt is not used.
- **Default** - (Protected Mode) - The interrupt is not handled at all in Protected Mode, but reflected to Real / Virtual Mode (see DPMI Info).

## Hardware Interrupts Information

An interrupt line is a hardware line over which peripheral devices ask the processor for service (send / receive data, etc.). Typically, each device uses an unique Irq line. However, as there are only 16 interrupts on a PC/AT some devices share an interrupt but have different addresses. In this case you cannot use both of them in the same time.

As you may have noticed, this interrupts are included in the [Software Interrupts](#), but are given separately because they are a bit different. Generally when speaking of interrupts for ports / sound cards / disks we mean *hardware interrupts*. Programmers generally talk of *software interrupts* as that is what they use in their programs and this is a broader area.

You as a user should only check that you are not using two peripherals with the same interrupt. See [Serial / Parallel Ports](#).

### Topics:

<u>Irq. No.</u> <u>Detected</u>	<u>Address</u> <u>Owner</u>	<u>Description</u>
------------------------------------	--------------------------------	--------------------

**Warning:** [Considerations](#). Also, see [Plug-and-Play](#) topic.



**Interrupt Detected**

Specifies if the interrupt is currently used in this system or not. One interrupt can be used by more than one peripheral - e.g. parallel ports and other devices.

Some interrupts may be used by special hardware (fancy disk controllers) that you might not be aware of. It is usually safe to use an interrupt for a peripheral if it is not detected.

## Hardware Interrupts Considerations

This section has been introduced to clarify the way SAW detects interrupts and their characteristics. The following try to shed some light into the matter...

**Note:** You can *usually* trust what SAW tells you about an interrupt being used or not. The method used is quite precise, but, of course, it does have its limitations...

- **LPT Irqs** - If SAW sees a LPT1 and/or LPT3, then it lists it/them as a user/users of Irq7 ! This does **not** mean that Irq7 is used ! In fact it *may* very well be free ! Similarly for LPT2, 4 and Irq 5. Therefore, you generally *may* use Irq 5 and 7 for peripherals.

For example if LPT1 is attached to a printer and LPT3 does not exist or is not used, then Irq7 (default) **can** be used for a sound card or other device. No conflicts should appear, but some distortion *may* be noticed if you print while you listen to music from the sound card - depending on model. Check !

- **COM Irqs** - COM ports are a bit different from LPT ports in respect to IRQs. If you *use* COM1 which uses Irq 4 then you may use Irq 4 *only* for another COM port (2,3).

For example, if SAW detects COM1 (on Irq 4) then it lists it as a user of Irq 4. If you do use COM1 then Irq 4 is *used*; if you do *not* use COM1 then Irq 4 is **free** ! Similarly for other COM ports.

- **Irq 2 and 9**- As I previously said, these are cascaded. Therefore, you cannot use them both. Some video cards use Irq 9 for full-motion video synchronisation. Also, some sound cards behave erratically when configured to use Irq 2. Generally, user Irq 9 if you really want to.

- **Owners** - How do you know if two devices use the same Irq ? You don't ! SAW or other software programs will be unable to help. That's why:

Say you have two device drivers *DRIVER1* and *DRIVER2*, both listed in Config.sys in this order, and both trying to use Irq 7. When the computer starts, the owner of Irq 7 is the BIOS default handler (owner). When *DRIVER1* loads, it hooks Irq 7 and becomes its owner. When *DRIVER2* loads and (as it usually happens) does not check if Irq 7 is already taken, then it takes control. Now the owner is *DRIVER2*. Programs requiring *DRIVER2* will run just fine while those requiring *DRIVER1* may malfunction.

Note that *DRIVER2* does not handle requests (functions) of *DRIVER1* and passes them to *DRIVER1*, all will be fine ! Only if they do the **same** thing (or close) there will be a problem.

All software detection programs will show only the **last** owner and none of the previous ! How do we detect if we have an conflict ?

- Get a **Hardware Detection Card** which shows when each Irq and DMA channel becomes active. You have to make note, while stepping through Config.sys and AutoExec.bat lines, of the devices that trigger an Irq or DMA channel. You will be surprised, I can tell you ! For Windows, it is a bit more complicated, but not too much.

Problem is, the card's very expensive (£200) - useful if you are an technician...

- Write a **program** which installs before **any** device drivers (Yeah, sure...) and makes a list of **every** previous owner of every Irq ! (traps *Set Int Vector* and *Interrupt Table*) Trickier for DMA channels... Any takers ? I'm waiting...

In fact, I've heard of a few programs that attempt to do it, (for other reasons, though), but the results are not exactly what they should be...

- **Sound Card Blues.** If, for example, your sound card is configured for Irq 7 and requires **no** Real Mode TSR device drivers, you may think Irq 7 is free ! In Protected Mode (Windows), the card has to have an owner (if you installed its driver) ! Check the Sound Card Information topic.

Sorry for all this mumbo-jumbo. So when **is** an Irq free ?

- **Real Mode** - Surely when the *Owner* is **Free**. Possibly if the *Owner* is **System** or **BIOS**.  
**and**
- **Protected Mode** - Usually when the *Owner* is: **Default** - this means that the interrupt is not serviced in Protected Mode at all, but reflected to Real Mode directly. See *Real Mode Owner* in this case.

**Note:** See Plug-and-Play topic for an easy *way out*.

## Plug And Play

If you can, wait and buy a computer with **Plug-and-Play** BIOS / Operating System and get **Plug-and-Play** cards.

### Introduction

*Plug-and-Play* is a set of functions that, if implemented by the BIOS, provides the ability to manage the fundamental system resources which include DMA, Interrupt Request Lines (IRQs), I/O and Memory addresses. These resources, termed **system resources**, are in high demand and commonly are over-allocated or allocated in a conflicting manner in *ISA* systems, leading to system configuration failures. A plug and play system BIOS will play a vital role in helping to manage these resources and ensure a successful launch of the operating system.

A *Plug-and-Play* BIOS takes on the responsibility for configuring *Plug-and-Play* cards, as well as system board devices during the power-up phase. After the process is complete, control of the *Plug-and-Play* device configuration passes from the system BIOS to the system software. The BIOS does, however, provide configuration services for system board devices even after the start-up process is complete.

The *Plug-and-Play* BIOS Specification provides a mechanism whereby a *Plug-and-Play Operating System* may perform this resource allocation dynamically at runtime. The Operating System may directly manipulate the configuration of devices which have traditionally been considered static via a System BIOS.

### What does all this actually mean ?

If you buy a computer which has a *Plug-and-Play* BIOS (ask the salesperson - **get it in writing**) then you can forget about all the problems of peripheral conflict, interrupts and DMA channels. The BIOS and the cards (get new ones that are *Plug-and-Play* compatible) will take care of it all.

A *Plug-and-Play* compatible Operating System like **Windows 95** (*Windows 4.0*) and **Windows NT 4.0**, will also enable you to change the system characteristics at runtime, without rebooting or powering-down the computer. On portables, you can shuffle in and out different **PCMCIA** cards while the Operating System will take care of it all. *Plug-and-Play* software support packs for DOS will appear as well.

An **EISA** or **PCI** system bus may help, as they already take care of some of this tasks themselves.

The latest version of *Plug-and-Play* specification is **1.01** (end 1994).

## Run Command

Use this command to run one of the standard programs included with DOS / Windows. I considered that this applications may help you to get/set information. See [Utilities](#).

### Dialog Box Options - Run Application

Select the application you want to run.

<a href="#">Control Panel</a>	<a href="#">Windows Set-up</a>	<a href="#">Sys Configuration</a>
<a href="#">File Manager</a>	<a href="#">Print Manager</a>	<a href="#">Write</a>
<a href="#">Notepad</a>	<a href="#">Diagnostic Utility</a>	<a href="#">Dr. Watson</a>

**WARNING:** These applications come standard with **Microsoft Windows 3.1X**. SAW has **NO** connection with **any** of them whatsoever: it **just runs** them for you ! For more information, see your Windows manual, or choose a command from the Help menu in the corresponding program.

## **Control Panel**

Control Panel is a Windows application that provides you with a visual way of modifying your system while working with Windows. Each option that you can change is represented by an icon in the Control Panel window.

When you change an option, it is stored in the Win.ini file so that it will be in effect the next time you run Windows.

## **Windows Set-up**

- Using Windows Set-up, you can add / remove any Windows components to your system that you may have chosen not to include when you first set up Windows.
- If you change to a system configuration that is different from the one you had when you set up Windows, you can use Set-up to change your software and hardware settings.
- After you install an application on your system, you can use Set-up to make sure it will run smoothly with Windows.

## **Print Manager**

Print Manager is a Windows application that manages printing. When you print a document from a Windows application, the application passes the printer, font, and file information for the document to Print Manager. Print Manager then controls the printing of your document while you continue working with your other Windows applications. If errors occur while printing, Print Manager notifies you of the problem.

You can view network printer information for any network print queue you specify. You can print several documents at the same time or in succession. You can use the queue to view printing information or to change the type of information displayed and the order in which your documents are printed.



## **Diagnostic Utility**

The Microsoft Diagnostic Utility (**MSD.exe**) is a DOS program which provides you with information about your computer system. It can provide loads of information and can create / print reports as well. Useful if you have problems with drivers / TSR programs as it will help you to avoid conflicts. Supplied with *Windows 3.1* and *MS-DOS 6.0* or later. Use the latest version.

SAW gives some of the information that MSD provides but is geared towards Windows being a Windows application.

## **Dr. Watson**

The Microsoft Doctor Watson utility intercepts *Unrecoverable Application Errors (UAE's)* and writes an error report log file. This log file can assist you in finding the cause of the UAE. If an exception has occurred, the date and time of the exception is shown as well.

Useful if you experience crashes / lockups with a program and you have to call technical support.

## **System Configuration Editor**

The Microsoft System Configuration Editor is a small undocumented program supplied with Windows which allows you to quickly modify the main DOS & Windows initialisation files (*Config.sys*, *AutoExec.bat*, *Win.ini*, *System.ini*).

It is more like Notepad - no fancy stuff - but can help you quite a lot. Use it !

## **File Manager**

The Microsoft File Manager is a tool you can use to organise your files and directories. You can use File Manager to move and copy files, start applications, connect to network drives, print documents, and maintain disks.

In File Manager, your files and directories are displayed in a directory window. The directory window is split: the left half displays the directory tree, and the right half displays the contents of the current directory. When you select a different directory in the directory tree, the contents of that directory are displayed in the right half of the window.

Essential in managing your system.

## **Write for Windows**

Microsoft Write for Windows is a word processor that you can use to create and print documents. It may not be a *Word for Windows* beater, but is great for writing letters and even documents when you don't need 1000's of features. Very fast, uses very little memory and its slick.

**Note:** Windows 95 comes with *WordPad for Windows* which is even better: you can almost mistake it for Word for Windows ! :)

## Notepad for Windows

Microsoft Windows Notepad is a text editor that you can use to edit small text files, such as your *Win.ini*, *System.ini* files. Its extremely useful - you cannot do without it or a replacement. Its not a word processor - but its not supposed to be one.

Only snag is that the **16bit** version has a **30-64KB** file limit. Roll on 32bit Notepad !

## Logical Drives Information

This window shows the list of all the drives recognised by DOS/Windows. You can see most of their characteristics here. Also, you can get extra information should you require it.

Why *Logical* ? These drives are the ones that you actually use in DOS/Windows, but they are logical, i.e. some do not exist (like memory drive - they are not drives but memory) and furthermore, many logical drives may reside on the same hard disk. Also, network drives are remote and not in your machine. They do not tell much about the internal system configuration but are very useful as we care about them and not the actual disks. See [Physical Drives Info](#).

As you may have noticed the list of all drives is shown in **File Manager** style. To select a drive click on it or select it from the list-box. The selected drive is shown as recessed (inwards) and the others outwards. All the information shown refers to the selected drive.

### Topics:

<a href="#">Drive Name</a>	<a href="#">Path</a>	<a href="#">Drive Type</a>
<a href="#">Extended Type</a>	<a href="#">Location</a>	<a href="#">Status</a>
<a href="#">Bytes per Sector</a>	<a href="#">Sectors Per Cluster</a>	
<a href="#">Total Space</a>	<a href="#">Free Space</a>	
<a href="#">Update Button</a>	<a href="#">Extra Info Button</a>	

**Note:** If the information is not available an **N/A** will be shown. This usually happens for removable media when the disk is not in the drive.

## Logical Drive Name

This is given in the usual drive list-box fashion (in Windows). You can select the drive you want from the list-box or click on it.

The format is : **DOS ID**. (e.g. **A:**) **Name** (e.g. **hard disk**). Depending on drive type the name may be one of the following (if any / available). No name will be shown in case of drive error and you will be informed of its nature.

- The removable (e.g. floppy/CD-ROM) disk's name is not shown since no disk may be in drive.
- For fixed disks the name shown is the label of the drive.
- For network drives the connection path is shown or if not available, the name of the network disk.



## Logical Drive Path

The current directory for the selected drive. This is the current / last directory used by an application and it is the default.

The path can be changed with ***File Manager*** or another similar application.

## Logical Drive Type

SAW currently recognises the following drive types:



- **Removable Media:** *3½ inch Floppy Disk* or *Floptical*. See [Extended Type](#) for more information.



- **Removable Media:** *5¼ or 8 inch Floppy Disk*. See [Extended Type](#) for more information.



- **Fixed Disk** (e.g. Hard Disk) and other non-removable, local media.



- **Memory Drive** (created with RAMDrive or compatible).



- **CD-ROM Drive**.



- **Connected Network Drive**. You can use it.



- Existent but **not connected to Network Drive**. This drive is available on network, but you must first connect to it to use it. See your network guru...

## Logical Drive Extended Type

This information is available only if there are variations in drive type:

- **Removable Media.** The type of floppy disk is shown, e.g. **1.44MB 3½** floppy disk. This is the maximum capacity the drive / controller will accept (e.g. 1.44MB drive will accept 720KB disks as well) and not the actual capacity of the floppy disk in drive.

To find that out the actual floppy disk type, press the [Extra Info Button](#). This will give you all the information you may require. Some 1.2MB 5¼ controllers will read / write a 360KB ¼ disks, but you may be unable to read it with a 360KB controller due to hardware incompatibility.

- **Fixed Local Drives** - Specifies one or more of the following:

- **Shrd:** the drive is **shared** across the network, usually in peer-to-peer networks. This may be important to know if you store something important on that drive. Drive could be busy momentarily if it is accessed over the network.

- **Comp:** the drive is a *DoubleSpace/DriveSpace* **compressed** drive. Some fields like *Free Space* may not be very meaningful in this case. A compressed drive is not a real drive as you know, so the true (uncompressed) drive information has to be taken by examining the *host drive*.

- **Host:** the drive is a *DoubleSpace/DriveSpace* **host** (uncompressed) drive. This is the real DOS drive which is used to simulate the compressed drive. Some files (already compressed ones, device drivers, etc.) cannot reside on a compressed drive, so you either put them on a uncompressed drive or on a host drive.

## Logical Drive Location

This may be one of the following:

- **Local** - i.e. the drive, whatever it is, exists in your computer. This includes shared drives on peer-to-peer networks. Naturally, if you are not connected to a network, all drives should be local.
- **Remote** - i.e. located in another computer. This may be a dedicated file server or just another computer. You must be connected to one in order to use it. The actual type of the drive can be seen elsewhere.

**Note:** Be careful with Remote drives: if you are not the administrator don't change anything as you may disrupt / lock other people's machines on the network ! Pay attention to shared files / applications as you may get into trouble...

## Logical Drive Status

This may be one of the following:

- **Ready** - the drive is OK and ready for use.
- **No Disk** - this applies to removable drives. No disk is in drive or a non-system (e.g. non-DOS, not formatted or corrupted) disk is in drive. In this case most of the information is unavailable. No *Extra Information* will be available also.
- **Busy** - Network drives may be busy sometimes due to connection overload or some other problem on the network. Usually, a re-try solves the problem. Hard-Disks / CD-ROMs can also be busy due to various factors.
- **Error** - An error was encountered while interrogating this drive. This can be due to various reasons and a re-try should be attempted. If error persists check drive using a *disk doctor program* to see the actual reason for the error. See Utilities.

## Logical Drive Total Space

The total size of the selected drive (*Total Space*).

The number is given in kilobytes (KB - 1024 bytes) and:

- Megabytes (MB - 1024 KB) if the total size is less or equal to 2048MB.
- Gigabytes (GB - 1024MB) if the total size is greater than 2GB.

**Note:** If the drive is compressed this is the estimated total size given by the compressor based on a 2:1 assumption or the current average ratio (depending on compressor)

## Logical Drive Free Space

The free space of the selected drive (Total Free Space).

The number is given in kilobytes (KB - 1024 bytes) and:

- Megabytes (MB - 1024 KB) if the free space is less or equal to 2048MB.
- Gigabytes (GB - 1024MB) if the free space is greater than 2GB.

The percentage of free space is also shown together with a graphical representation (pie chart) of it. The **Red** represents the used space and the **Green** the free space, so you can see at a glance how much is left.

**Note:** If you use a temporary swap file, enough free space should be left for it. The TEMP directory should also have at least 2-3MB free as many applications create temporary files (or when printing) and become flaky if not enough space is free

**Note:** If the drive is compressed this is the estimated total free space given by the compressor based on a 2:1 assumption or the current average ratio (depending on compressor). Most of the time you will manage to fit less files as this is an optimistic reading.

## **Logical Drive Update Button**

Press this button to get the latest information about this drive. When you select a drive the information is updated automatically, but if you have programs running in the background or monitor a network drive you may want to get the current free space, etc.

If no disk was in drive when you selected it, put one now (if you forgot to put one before selecting the drive) and press Update. SAW will read the disk and show you the then disk information.



## Boot Sector Information

Every DOS Drive begins with a *Boot Sector* or *Boot Record* which is the drive's first sector. This may start the Operating System if it contains it, i.e. *boot-able*. This sector contains most of the information you may ask about a drive and SAW will show it to you.

Most drives have a boot sector; network drives do have one, but can't be accessed in the usual way as they are remote so no information is shown. This is because some servers run Non-DOS operating systems, so the boot sector contains other, specific, information.

### Topics:

<u>Drive Type</u>	<u>Sectors per Track</u>	<u>Heads</u>
<u>Bytes per Sector</u>	<u>Sectors per Cluster</u>	<u>Copies of FAT</u>
<u>Sectors per FAT</u>	<u>Drive Number</u>	<u>Media Descriptor</u>
<u>Formatted With</u>	<u>Max. Root Entries</u>	<u>Boot Signature</u>
<u>Volume Label</u>	<u>Volume ID.</u>	<u>File System Type</u>
<u>Reserved Sectors</u>	<u>Hidden Sectors</u>	<u>Total Sectors</u>
<u>Total Capacity</u>		

## Drive Type

This field specifies the exact type of the disk on which is placed. This is based on the Media Descriptor field.

For example if you have a *720KB* disk in a *1.44MB* drive, the Extended Type field will display 1.44MB (the drive type) while this field will display 720KB - the type of the actual disk.

**Note:** On some operating systems this may be inaccurate. Also, tape drives, removable storage drives (Bernoulli, Floptical, etc.) may be incorrectly detected.

## Drive: Sectors per Track

To store information on disk, DOS divides the disk into several concentric tracks similar to grooves in a record album (LP). Each track is further divided into pie-shaped storage locations - *sectors*.

This field specifies the number of sectors contained within each track. The bigger the disk, the more sectors per track.

**Note:** This is the logical division, not the physical one. As an outside track is longer than an inside one, the disk has more physical sectors per track internally on an outside than on the inside. This information is hidden and you should not bother about it. For simplification the logical number sectors per track is constant for a drive.

## **Drive: Number of Heads**

The number of disk heads the drive supports.

A single-sided disk supports only one head; a double-sided two heads.

A three platter hard disk may have 6 heads (one for each side of the platter), but some have only 4 (the outside platters do not have heads). Larger drives use many platters, so the number of heads on different hard disk types and sizes will vary.

## **Drive: Bytes per Sector**

The number of bytes that can be recorded on a sector. Most DOS drives have **512** and maximum (nowadays) is 2048 (2KB).

A **hard disk** should have 512 to avoid problems with software which assumes a sector holds 512 bytes, but higher values should work also.

**Memory** drives can have any number in fact, but you should also stick to 512.

**Network** drives, especially bigger ones, may more than 512; this is normal especially if the server is a dedicated one (like *Novell Netware*). 512 is again preferred.

## Drive: Sectors per Cluster

The number of sectors that make up a cluster. This number multiplied with Bytes per Sector gives the size of the *allocation unit*. For example 512 bytes per sector and 4 sectors per cluster gives a 2KB allocation unit (see **chkdsk** or **scandisk** DOS utilities).

What is an allocation unit? When DOS/Windows stores a file it uses a round number (integer) of allocation units. For our example any file between 1 and 2048 bytes will take a whole allocation unit - the remaining bytes being wasted (*slack*). Similarly for any file size.

Therefore, the smaller an allocation unit, the smaller the slack if you have many (less than 20KB) files. This will mean less waste and better management.

As hard disks become bigger, the allocation unit increases so some people partition their hard drives into chunks to reduce the allocation unit. But, as files get bigger and bigger, this issue is not that important: **just get a bigger drive !**

## Drive: Copies of FAT

DOS keeps track of where files reside on drive using a FAT (*File Allocation Table*). In general, the FAT serves as a *road map* for DOS to locate files. If it becomes damaged by something DOS will not be able to locate your files - so you lost them.

To reduce the possibility of such an error, DOS keeps usually two copies of FAT on disk. This way it can be checked if all is OK (both are the same) or to repair the first one if it becomes damaged - but this does not always work.

This field specifies the number of FAT DOS keeps.

**Note:** Special DOS drives like memory, CD-ROM, etc. may have only one.

**WARNING:** Use FAT modifying programs (like **Defragmenters**) sparingly: if an error occurs and one FAT becomes damaged it cannot be repaired easily. Also check all disks for viruses; some mess up your FATs.

## **Drive: Sectors per FAT**

See [Copies of FAT](#) for a full explanation.

This field specifies the amount of disk space in sectors that each FAT consumes. Therefore, the larger the copies of FAT, the larger the space taken by them.

This is in direct relation with the size of the drive.



## Drive Physical Number

This field specifies whether this drive is a floppy or a hard disk.

- **00h** for a floppy disk.
- **80h** for a hard disk.

**Note:** This information is available only if you have **DOS 4.0** or higher.

## Drive Media Descriptor

This value specifies the type of drive this is. See [Drive Type](#) for more information.

**Note:** Two different drive types may use the same media descriptor. Detection is based on other values as well to avoid confusion.

**Note:** Some Operating Systems use a this field for other purposes.

## **Drive OEM Name (Formatted With)**

This field usually shows the name of the Operating System (and version) or utility with which the drive was formatted, e.g. **MS-DOS 5.0**, **IBM 3.0** or **MS-WIN 4.0**.

**Note:** Some backup utilities use it for different data storage purposes.

## **Drive: Maximum Root Entries**

When you format a disk, DOS sets aside a specific number of sectors to store the disk's root directory. Every directory uses 32 bytes.

This field specifies the number of entries for which DOS has reserved space.

For example, if a disk has 64 entries (e.g. 160KB floppy) you cannot create more than 64 directories in the root directory; however, you can create as many sub directories as you like - this is only limited by disk space.

Hard disks have 512 (or more) so you should be OK unless you have a 1GB hard disk !

## Drive Boot Signature

This is must be **29h** (hexadecimal) if the extended information is available for this drive. This happens for DOS 4.0 or higher.

**Note:** This information is available only if you have **DOS 4.0** or higher.

## Drive Volume Label

This specifies the disk's name (also shown by the field Drive Name) which is assigned using a *file manager* in Windows or with DOS's LABEL.

This is a means of identifying each drive, especially floppy drives.

**Note:** Some backup / cataloguing utilities use this field for their own purposes, in which case garbage may be shown.

**Note:** This information is available only if you have **DOS 4.0** or higher.

## **Drive Volume ID.**

This is two word field shown in hexadecimal which contains the disk serial number. At formatting this number is selected at random.

**Note:** Some copy protected software use it to identify the original drive on which they were installed. You should install them again if you plan to move them...

**Note:** This information is available only if you have **DOS 4.0** or higher.

## Drive File System Type

This field specifies if the disk uses 12 or 16-bit FATs. Usually, the format of this field is **FAT12** or **FAT16**.

**Note:** Drives over 32MB uses 16-bit FAT, but 32-bit Sector Addressing.

**Note:** This information is available only if you have **DOS 4.0** or higher.



## **Drive: Reserved Sectors**

This field specifies the number of sectors which are reserved for specific use. Generally there is only one sector reserved and this is the *Boot Sector* itself !

## Drive: Hidden Sectors

Every hard disk begins with a collection of sectors which contain the Partition Information or *Master Boot Record*. Typically, most disk manufacturers reserve the first cylinder for this information, i.e. the value of the Sectors per Track field.

This field describes the number of sectors the drive hides for partition table information.

## **Drive: Total Sectors**

This is the total number of sectors on disk, i.e. the number of sectors that are actually used.

You can find the number of cylinders the drive has by dividing this number by Sectors per Track and Number of Heads.

## **Drive: Total Capacity**

This is the total capacity of the drive. It is given in two ways:

- In bytes (usually a very large number).
- And in brackets in:
  - Kilobytes (1KB = 1024bytes) if less than 3MB.
  - Megabytes (1MB = 1024KB) if less than 3GB.
  - Gigabytes (1GB = 1024MB) otherwise.

## Physical Drives Information

This lists the true number/type of drives that you have in your system. You may not know the difference - most programs don't either - but this information is important because this is what you actually have.

**Example:** You may have partitioned hard disks: if you have 2 partitions which may be drives C: and D: - this does not mean that you have 2 hard disks. But you may have indeed 2 hard drives in this case. Also memory drives (created by RAMDrive) are virtual not physical.

### Topics:

<u>Drive</u>	<u>Type</u>	<u>Extended Type</u>
<u>Change Line</u>	<u>CMOS Type</u>	<u>BIOS ID</u>
<u>Heads</u>	<u>Sectors</u>	<u>Cylinders</u>
<u>Total Capacity</u>		

**Note:** Some hard disks with an fancy interface may not be detected ! Also non-standard storage devices.

## Drive

The drive the information refers to.

- *Floppy Disks* are numbered starting from **A:**
- *Hard Disks* are numbered starting from **C:** (usually)

**Type**

The general type of the drive, e.g. *Floppy Disk*. See [Extended Type](#) for more detailed information.

## Extended Type

More information depending on drive type. This may be:

- **Floppy Disks:** Maximum capacity & Format, e.g. *1.44MB 3½" floppy*.
- **Hard Disks:** Controller type, e.g. IDE.



**Change Line - Floppy disks only !**

Shows if the drive supports change line or not, i.e. senses when a disk was removed/inserted in the drive. Older drives do not support this feature, though not many programs take advantage of it.

## CMOS Type - Hard Disks only !

The BIOS has a small database which contains information (Heads, Sectors, Cylinders, WPComp, etc.) about 46 hard disks. If yours is one of these (quite outdated now), you just select it - otherwise you need to enter its characteristics manually (there are ways around this).

Therefore types **1..46** are *built-in*. Type **47** is *user defined*. If you have 2 hard disks, they can both be type 47.

## **BIOS I.D.**

The BIOS ID is a hexadecimal number. It is:

- **00h** for floppy disks.
- **80h** for hard disks.

**Disk Heads**

The number of heads (maximum) the drive has.

**Disk Sectors**

The number of sectors per track the drive has. This number may be untrue as the disk controller hides sometimes the true values from the PC.

## **Disk Cylinders**

The number of usable cylinders (tracks) the drive has. This number may be untrue as the disk controller hides sometimes the true values from the PC.

Some drives reserve a cylinder as a *landing zone* (when you turn the PC off).

**Disk Total Capacity**

The total capacity of the drives in bytes (huge number) kilo-bytes (1KB=1024bytes) and mega-bytes (1MB=1024KB).

In future the capacity will also be shown in giga-bytes (1GB=1024MB) and maybe terra-bytes (1Tb=1024GB).

## Partition Information

Due to the (relatively) large capacity of the hard disks, the Operating System lets you divide it into multiple regions called *partitions*. In this way, you can have multiple operating systems on a single hard disk, e.g. DOS, OS/2, Windows NT and choose between them at boot time. Of course, you can just use them for a single Operating System and store different things on different partitions, e.g. OS on partition 1, programs on partition 2, data/files on partition 3, etc. You have added security (if one gets damaged, the others remain OK), better *housekeeping*, etc.

You can have up to 4 partitions. Each DOS partition has its drive letter assigned. These are Logical Drives. You can also subdivide each partition (under DOS) into multiple logical drives.

### Topics:

<u>Number</u>	<u>Boot</u>	<u>Type</u>	<u>Bits</u>
<u>Starting Head/Sector/Cylinder</u>		<u>Ending Head/Sector/Cylinder</u>	
<u>Total Sectors</u>			

**Note:** On the first half of the window you can see a summary of the physical characteristics of the hard disk the partition information refers to.



**Partition Number**

The number of the partition the information refers to. The range is **1..4**.

## **Boot Partition**

This shows if you can boot (i.e. start the system) using this partition. The partition which contains the Operating System must be configured as *Active* or *Bootable* otherwise the BIOS will not know where is the OS.

## Partition Type

The type of the hard disk partition. This can be:

- Primary DOS
- Extended DOS
- XENIX (old Microsoft UNIX)
- OS/2 (IBM)
- HPFS (NTFS) - Windows NT or OS/2
- UNIX (generic)
- AIX (UNIX version)
- Linux ("*Not-UNIX*")
- SuperStor - the disk-compressor
- DR-DOS protected partition
- Windows 95
- Non DOS

**Note:** Partitions used by other Operating Systems may not be recognised by SAW and will be treated as *Non-DOS* (Unknown).

## Partition FAT Bits

The number of bits the FAT of the partition uses. This can be:

- **12 bits** - usually floppy disks
- **16 bits** - hard disks partitions of 32MB or less
- **16 bits+** - huge partitions, greater than 32MB in size

**Note:** This is relevant only to DOS partitions. *Alien* (i.e. non-DOS) Operating Systems may use totally different file structures.

### **Partition Starting Hard / Sector / Cylinder**

The starting position of the partition on the hard disk.

- **Head** - starting head. The first head is **Head 0**.
- **Sector** - starting sector. Usually 1. The first sector is **Sector 1**.
- **Cylinder** - starting cylinder. The first cylinder is **Cylinder 0**.

**Note:** Logical drives inside a partition are not shown.

### **Partition Ending Hard / Sector / Cylinder**

The ending position of the partition on the hard disk.

- **Head** - ending head. Usually last head. The first head is **Head 0**.
- **Sector** - ending sector. Usually last sector. The first sector is **Sector 1**.
- **Cylinder** - ending cylinder. The first cylinder is **Cylinder 0**.

**Note:** Logical drives inside a partition are not shown.

### **Partition Total Sectors**

The total number of sectors contained by the partition. This gives the size of the partition if you know how many bytes there are per sector.

Under DOS (& *plain* Windows) you can have the following partition sizes:

- DOS 3.3 or less - up to 32MB
- DOS 4.X - up to 512MB
- DOS 5.0 or later - 2GB (2048MB)

See the respective manual for other Operating Systems.

## VxD Information

What is a VxD ? A VxD (*Virtual Device Driver*) is a 32bit, Protected Mode Device Driver. Under Windows 3.XX VxDs are loaded only in *Enhanced Mode*. The VxDs are listed in the *System.ini* file under the *[386Enh]* section (*device=* entries).

Windows is a much more complex Operating System than DOS, and, therefore comes with more (about **30-50**) device drivers which look after (virtualise) the peripherals connected to the system. Every peripheral (video card, sound card, scanner, disk, mouse) comes with its VxD. Memory management, COM/LPT port management, DMA, etc. is also managed by VxDs. The drivers that everyone is talking about (*.DRV*) use the VxDs to accomplish tasks.

In Windows, applications have less rights than VxDs so, if fast, direct, hardware I/O is necessary, a VxD must be created to support the application. Standard mode does not require VxDs as it does not virtualise everything as Enhanced mode does. However, this topic is concerned with Enhanced mode only.

This window gives you various information about the VxDs installed.

### Topics:

<u>ID. No.</u>	<u>P. Mode API</u>	<u>V86 Mode API</u>
<u>Name</u>	<u>Version</u>	<u>DDK Version</u>
<u>Services</u>		

**Note:** VxDs do not currently support *version information*, therefore less information is available about them compared to tasks, modules, etc. Also, a VxD is needed to get VxD information !

**Note: Limitations** - WPS.386 - please see the *ReadMe* file for information about this.



## **VxD ID. Number**

The ID. Number is a *unique* number assigned to a VxD so that they can be distinguished by normal applications. You cannot have two loaded VxDs with the same ID number. However, a updated VxD which replaces a similar default VxD has to have the same ID no. - to ensure compatibility with applications which require API services of the old VxD.

Some VxDs do not export any functions for applications, so they can have no ID number - which is **0000h**. This is the only number that can be shared by two or more loaded VxDs.

## VxD Protected Mode API

This is the address (***Selector:Offset - 16:16bit***) of the API services exported by the VxD for Protected Mode applications, if any. **N/A** will be shown instead if no API services are exported.

32bit applications are given a *16:32bit Selector:Offset*. As SAW is 16bit, it shows the 16:16bit API memory entry.

## VxD Virtual 86 Mode API

This is the address (**Segment:Offset**) of the API services exported by the VxD for Real (*Virtual 86 Mode* in fact) Mode applications, if any. **N/A** will be shown instead if no API services are exported.

## VxD Name

This is the name given to the VxD. These names are listed in the *System.ini* file under the *[386Enh]* section by the *device=* entries. Currently, Windows comes with many VxDs which cater for the various peripherals which may be installed in your system.

Below, a list was made with VxD names, the *Windows version(s)* which has/have them and their descriptions:

Name	Ver	Description
BIOSXLAT	3.X	BIOS API Mapper
	X	
BLOCKDEV	3.1X	Hard Disk Device - Fast Disk Device
COMBUFF	3.1X	UART Buffering Device
DOSMGR	3.X	DOS Manager
	X	
EBIOS	3.X	Extended BIOS Device (PS/2, EISA)
	X	
IFSMgr	3.11	Installable File System Manager Device
Int13	3.1X	Int13h Redirector - Fast Disk Device
PageFile	3.1X	Memory Paging Device
PageSwap	3.X	Swap Paging Device (Virtual Memory)
	X	
Parity	3.X	Parity Error Handler
	X	
Reboot	3.X	Ctl-Alt-Del sequence Handler
	X	
SERIAL	3.11	Serial Port Device
SHELL	3.X	Windows Shell Manager
	X	
VADMAD	3.00	Direct Memory Access (DMA) Device
VCACHE	3.11	Disk Caching Device
VCD	3.X	Communications Device (conflict Handler)
	X	
VCOMM	3.11	Communications Device (conflict Handler)
VDD	3.X	Display Device (various names depending on video card)
	X	
VDMAD	3.1X	Direct Memory Access (DMA) Device
VFAT	3.11	High Performance File Allocation Table (FAT) Manager
VFD	3.X	Floppy Drive Device
	X	
VflatD	3.1X	Video Synchronisation Device
VHD	3.00	Hard Disk Device
VKD	3.X	Keyboard Device
	X	
VMD	3.X	Mouse Device
	X	
VMM	3.X	Virtual Machine Manager
	X	
VMPCD	3.X	Co-Processor Device
	X	
VPD	3.1X	Printer Device (conflict Handler)
VPICD	3.X	Programmable Interrupt Controller (PIC) Device
	X	
VPMTD	3.11	FAX Scheduling Device

VSD	3.1X	Sound Device
VSHARE	3.1X	File Sharing Device
VTD	3.X	Timer Device
	X	
VTDAPI	3.1X	Multimedia Systems Device
WDCTRL	3.1X	Western Digital Hard Disk Device - Fast Disk Device

The table below lists a few VxD that come with Windows or the Windows Driver Library and are optional depending on your computer system and configuration:

Name	Ver	Description
APM	3.X	Advanced Power Management Device
	X	
DOSNET	3.X	MS-Networks Support
	X	
DSVXD	3.1X	DoubleSpace Compressor Device
ISAPnP	3.11	ISA Plug-and-Play Device
MSODISUP	3.11	ODI Network Protocol Support
NETBEUI	3.11	NetBEUI Stack Support Device
NetBIOS	3.1X	Novell Netware NetBIOS Protocol Support
NWNBLINK	3.11	NetBIOS Network Protocol Support
PRODUCER	3.1X	Sound Producer Pro Device
VADLIBWD	3.1X	AdLib Device
VASBID	3.1X	Artisoft Sound Board Device
VILD	3.11	Interlink Network Client Device
VIPX	3.11	IPX Network Protocol Support
VNetSup	3.11	Network Support Device
VNETWARE	3.X	Novell Netware Shell
	X	
VRedir	3.11	Network Redirector Device
VSBID	3.1X	Sound Blaster Device
VSBPD	3.1X	Sound Blaster Pro Device

**Note:** 3.XX means any Windows version with major version number 3, e.g. 3.0, 3.1, 3.11, etc. Similarly 3.1X means Windows versions 3.1, 3.11, etc.

**VxD Version**

This is the version given to the VxD by the VxD programmer. If the VxD is a standard one that comes with Windows, this should be the same as the Windows version. Note that there are a few exceptions for some reasons.

If the VxD is from a third party, it can be the same as the product version it came with, or indeed any version that the company decided to give it.

**VxD DDK Version**

This is the version of the Device Driver Development Kit (DDK) which was used by the programmer to write the respective VxD. Usually, this should be the same or lower than the Windows version - but it can even be higher if the new functions are not used.

## **VxD Services**

A VxD may export services for other VxDs to use and this field shows whether it does so. This services (API for VxDs) are available only for a VxD and not for an application like the API.

See Protected Mode API and Virtual86 Mode API for APIs available for applications.



## Base Memory Heap Information

Even if Windows uses mostly *extended memory*, *base memory* (the first 1Mb of memory) is still very important as Windows needs to communicate with DOS Device Drivers and other DOS modules.

However, you will find that many Windows Modules , Windows Module Usage and Tasks still use a certain amount of base memory, especially video, sound (multimedia) and network drivers. You may have thought that you have finished with this problems in Windows but this is not so.

This Window presents you with a breakdown of base memory information. Different information is shown in Windows Memory Information and DOS Memory Information.

### Topics:

<u>Module</u>	<u>Location</u>	<u>Types</u>
<u>Size</u>	<u>Percent</u>	<u>Hog</u>
<u>Total Memory</u>	<u>Total Modules</u>	

**Note:** SAW updates the information in real time, e.g. when you load/unload a module or an application is started/killed - so leave the window open to monitor your system's progress.

## Module

The name of the module (including tasks) which the information refers to.

To get a list of all modules check [Modules Info](#) and for a complete list of tasks and associated modules, check [Tasks Info](#).

Exception is **Free Blocks** which tells you information about the free memory blocks.

## Location

The location of the memory block(s). This includes:

- **Low - Base Memory** - The first **640KB** of memory (**00000h-9FFFFh**).
- **UMB - Upper Memory Blocks** - The next **384KB** which is shared with video memory and adapter BIOSes (**A0000h-FFFFFh**). See [ROM Modules Information](#).
- **HMA - High Memory Area** - The 64KB that can be accessed by DOS due to a design feature of x86 processors (**100000h-10FFEFh**). See [XMS Memory Info](#).
- **Var - Various** - Different blocks belong to different memory pools (e.g. Low and HMA). Usually Code/Data blocks can be stored elsewhere (e.g. UMB or extended memory) while TDB and MDB reside in Low memory.

**Note:** Strictly speaking, nowadays, only the first 640KB of memory are called *Base Memory*. However, for completeness, we have included UMB and HMA as well.

## Memory Types

There are many types of blocks that can be found in memory and the list is expanding as Windows becomes a more and more complex environment. As their names are rather long and would not fit in the information window, they were abbreviated to their first letter. Usually each module owns a combination of block types.

The main types are listed below:

- **G - DGroup** - Contains the default data segment and the stack for the module.
- **D - Data** - Contains program data. (It may also contain stack and local heap data.)
- **C - Code** - Contains program code.
- **T - Task** - Contains the task database (information about the task).
- **R - Resource** - Contains resources required by the module. There are various resource types, and they are listed below the main class.
- **M - Module** - Contains the module database (information about the module).
- **F - Free Block** - Belongs to the free memory pool.
- **I - Internal** - Specific Windows usage or memory allocated by the module for special use.
- **S - Sentinel** - Either the first or the last object on the global heap. These are the heap borders.
- **B - Burger Master** - Contains a table that maps selectors to arena handles.

## Size

The total size of the memory of blocks which belong to the module.

The *free base memory* is shown by the **Free Blocks** module.

## Percentage

The percentage of memory taken by the module. This allows you to see which modules/tasks create a low memory condition due to their excessive memory requirements.

The percentage of *free base memory* is shown by the **Free Blocks** module.

## **Total Base Memory**

The total size of base memory managed by Windows.

Normally this should be  $640\text{KB} + 384\text{KB} + 64\text{KB} = 1,088\text{KB} = 1,114,112\text{bytes}$ . As DOS takes the HMA in most cases, plus most of the UMBs is taken by ROM modules, plus DOS uses around 80-100KB of base memory for itself + DOS Device Drivers, etc...

Therefore, Windows is left with around **570-600KB** to manage usually.

**Total Modules**

This is the total number of modules which own memory blocks in the Base Memory. For a full list, see [Modules Info](#).



## Base Memory - Hog

A module becomes a **hog** when it has taken more than **4KB** of base memory. As Windows base memory is a scarce system resource - even if you have 100MB you *still* (start 1995) need base memory for network/sound/video/etc. drivers.

Therefore, all programs should use it only if they have to.

All tasks still **need** some 512-1024 bytes for the task database. There is no way to get round this ! On top of this, most need some buffers and stacks to talk to DOS software. This should not take more than 2-4KB.

Modules do not need to use any base memory, but this is allowed as long as they do not eat-up too much.

Do note that some modules (system modules like KERNEL, GDI, DISPLAY) must use memory, so there's no way round that. This should not be taken into considerations and are included for you to get the overall picture.

**Note:** The maximum base memory allowed may vary with the number of loaded modules and free memory. Therefore when memory is low and many modules loaded, all modules which own base memory should shrink it to the minimum.

**Note:** In fact SAW uses a rather complicated formulae which changes from time to time due to continuous development. Just look at the values and make up your own mind if you prefer.

**Note:** There are programs (*Windows Utilities*) like **MoreMem** (shareware), **Fix1MB** (freeware), **RAMDouble** (commercial) which *force* most modules to use other memory - therefore fixing memory allocation problems. Try them if you have problems - they work fine, but they do slow the system down a bit.

Also, don't forget *DOS Memory Managers* like **QEMM**, **MemMax**, etc. which increase DOS free base memory (loading Device Drivers & TSRs high), therefore increasing Windows base memory as well.

## GDI Heap Information

This window presents detailed information about the [GDI Heap](#). See [The Heap](#) to find more about what a heap is. Use [Windows Memory Info](#) for general memory information or [DPMI Info](#) for memory manager information.

Do note that this is Windows' internal information and not really for mere mortals. This is pretty undocumented and requires a fair amount of understanding of the **Windows engine** to interpret the values. This would take books (ages) so its far beyond this topic.

The left list box shows you a breakdown of the items on the heap. In this way you can see what is hogging up your system. Use the **Debug** version of Windows to monitor all this.

### Topics:

<u>Item Type</u>	<u>No of Items</u>	<u>Total Size</u>
<u>Percent of Used</u>	<u>Percent of Total</u>	
<u>Heap Free</u>	<u>Heap Used</u>	

The right list box shows you **some** of the modules which use the GDI heap. I also shows how much of this is taken by the respective module. In this way you can trap the application which eats too much of the heap and close it down.

### Topics:

<u>Module</u>	<u>No of Items</u>	<u>Total Size</u>
<u>Percentage Used</u>		<u>Hog</u>
<u>Total Detected</u>	<u>Total Modules</u>	

**Note:** I can't emphasise more that this is really the only **partially-true** information shown by SAW !

## USER Heap Information

This window presents detailed information about the USER Heap. See The Heap to find more about what a heap is. Use Windows Memory Info for general memory information or DPMI Info for memory manager information.

Do note that this is Windows' internal information and not really for mere mortals. This is pretty undocumented and requires a fair amount of understanding of the **Windows engine** to interpret the values. This would take books (ages) so its far beyond this topic.

The left list box shows you a breakdown of the items on the heap. In this way you can see what is hogging up your system. Use the **Debug** version of Windows to monitor all this.

### Topics:

<u>Item Type</u>	<u>No of Items</u>	<u>Total Size</u>
<u>Percent of Used</u>	<u>Percent of Total</u>	
<u>Heap Free</u>	<u>Heap Used</u>	

The right list box shows you **some** of the modules which use the USER heap. I also shows how much of this is taken by the respective module. In this way you can trap the application which eats too much of the heap and close it down.

### Topics:

<u>Module</u>	<u>No of Items</u>	<u>Total Size</u>
<u>Percentage Used</u>		<u>Hog</u>
<u>Total Detected</u>	<u>Total Modules</u>	

**Note:** If you are running the **Retail** version of Windows instead of Debug (very likely, see Windows Info) the information in the right left box (modules) is even more misleading ! *The Total Size and Percentage Used* fields **are not** accurate !

This is because the Windows Kernel does not tell SAW what it asks, so there's no way SAW can find out through other methods (who would you ask ???)

**Note:** I can't emphasise more that this is really the only **partially-true** information shown by SAW !

## Item Type

This is the name of the item which takes heap space. Most of them have familiar names if you are a programmer; they are too many to explain, so please read a book. Some are undocumented.

- **Internal** - means internal Window items / structures - undocumented.
- **Small Blocks** - These are free blocks, but are too small to be used. They arise due to shrinking of used blocks and as Windows does not do garbage collection, they stay unused. If the heap is very fragmented, then you may run out of resources.
- **Free Blocks** - are free blocks which can be used by the system.
- **Total Free** - is the sum of all/most free blocks.

**Number of Items**

The total number of items of the respective type found of the heap at the time. **0** means that no items of the specified type are on the heap or they are not reported !

**Total Size**

The total size (in bytes) of the blocks of the respective type found on the heap at the time. **0** means that no items of the specified type are on the heap or they are not reported !

### Percentage of Used Size

The percentage taken by the respective item from the used heap, i.e. how much of the used heap is due to the item. **0%** means that its not using any heap.

**N/A** is shown for *Total Free Blocks* because this are not included in the used sum - as they are free blocks !

### **Percentage of Total Size**

The percentage taken by the respective item from the total heap, i.e. how much of the whole heap is due to the item. **0%** means that its not using any heap or is using far too little to see.



**Percent Free**

The percentage of the heap that is free. This information is duplicated in [Windows Memory Info](#) window and is included only for completeness.

**Percent Used**

The percentage of the heap that is used. This information is duplicated in Windows Memory Info window and is included only for completeness.

**Module Name**

The name of the module (including tasks) which the information refers to.

To get a list of all modules check [Modules Info](#) and for a complete list of tasks and associated modules, check [Tasks Info](#).

**Number of Items**

The total number of items taken by the respective module. This is a sum of the number of certain types item taken by the module.

They are not listed separately, but may be shown in this way in the future.

**Total Size**

The total size (in bytes) of the blocks taken by the respective module. This is **part** of the total size taken by the module from the heap.

### **Percentage of Used**

The share (percentage) taken by the module with respect to the total size taken by all the modules which use the heap.

This is not out of the total size or used size, but out of the used size *detected* by SAW which is less than the total used size due to Internal / Unknown blocks.

### **Total Detected**

The sum of all used blocks detected by SAW from the heap and successfully assigned to an owner. Used orphan (Internal / Unknown) blocks are **not** included in this sum.

Therefore this sum is smaller than the real sum of all used blocks on the heap.

**Total Modules**

This is the number of modules which use the heap and were detected. This may not be a complete list due to hidden / internal stuff which SAW cannot track.



## Heap Usage Information - Hog

This is *awarded* to modules which use **50%** more heap space than they should. The heap is a scarce system resource (at least in Windows 3.0 - 3.11) so programs should not use more than they need to.

How does that work ?

Well, if there is 100KB of used heap and 10 modules running, each should take around 10KB or less. If one of them uses over 15-20KB (**50%** more) - depending on free space - then the module hogs the memory. This means it is using more than 15-20% of the system resources.

Do note that some modules (system modules like KERNEL, GDI, DISPLAY) must use memory, so there's no way round that. This should not be taken into considerations and are included for you to get the overall picture.

**Note:** In fact SAW uses a rather complicated formulae which changes from time to time due to continuous development. Just look at the values and make up your own mind if you prefer.

## OLE Servers Information

**OLE** or **Object Linking & Embedding** is a method of exchanging information between running applications

You can create a document that includes information such as charts, graphics, and spreadsheet data created by other applications. You can embed objects created by applications that support Windows and object embedding. You must have enough memory to run all of the applications you are using at once.

All the information used to create the object is kept with the document which contains it. To edit the object, you simply select it. The server is started so you make the editing or formatting changes you want. When you switch back, the object is updated with the changes you made.

The **server** is the application which creates / edits the object. The **client** is the application in which you insert the object.

<u>Object Name</u>	<u>Object ID</u>	<u>General Name</u>
<u>Server Icon</u>	<u>OLE Type</u>	<u>DDE Support</u>
<u>Server Type</u>	<u>Server OS</u>	<u>Server Location</u>
<u>Server Description</u>	<u>Server Version</u>	<u>Product Description</u>
<u>Product Version</u>	<u>Copyright</u>	<u>Company Name</u>

**Note:** SAW assumes that all the applications are installed correctly and the information supplied by them is correct. There is no *real* standard, so SAW assumes the one used by Microsoft in its latest *Microsoft Office Ver 4.X* product.

**Note:** The OLE is not the only object embedding standard. We can list **OpenDoc** (IBM / Apple), **CORBA** (OMG), etc. Also, the object oriented operating systems are, **NeXT Step** (NeXT), **Taligent** (IBM / Apple / Hewlett-Packard) and even **Cairo** (Microsoft Windows NT 4.0).

**OLE - Object Name**

The name of the object. This often contains the name of the server application, version and type of the object that you can insert, e.g. *Microsoft Word 6.0 Document*. Do note that one server may support many object types as well as compatible ones.

**OLE - Object ID.**

This is the internal name (Class Name) of the object. It is the one that is generally used to identify items that belong to an application.

## OLE - General Object Name

This is the generic / version independent object name. As new versions of servers appear which support many more functions (e.g. OLE 2), there had to be a way to identify them from lower versions. However, they had to be compatible with objects created by older versions.

Therefore, the **object name** may be *Microsoft Excel 5.0 Chart* while the **general object name** should be *Microsoft Excel Chart* only.

## **OLE - Object Icon**

This is the icon of the server application. It is **not** the icon which identifies the object itself, though the two may be the same. SAW shows server information, not object information.

Usually, the object icon contains the picture of a sheet of paper with a smaller version of the server icon placed upon it - see *Microsoft Word 6.0+*. This visual image is used to identify files as well in Windows 95' Explorer - rather like *MacIntosh Finder*.

**Note:** If the server is not found or is 32bit, the window OLE generic icon will be displayed instead.

## OLE - OLE Type

There are currently two OLE types today (1995):

- **OLE 1.0** - the first OLE standard.
- **OLE 2.0** - enables *in-situ* editing of objects, i.e. the server application *takes over* the client application - which makes easier editing. All new *Software Suites* support this for better integration.

**Note:** SAW does not establish an OLE link, so the information may be inconclusive.

## **OLE - DDE Support**

DDE or *Dynamic Data Exchange* is a set of procedures which are used by applications to *communicate* with one another using defined protocols which may not be application specific. Information contained in a document created by another application, for example, can be updated automatically when it changes - ensuring information synchronisation.

Here, it means that the server has defined the DDE commands that can be used to communicate with it. The available commands vary from application to application, but there are a set of main functions.

**Note:** SAW does not establish a DDE link, so the information may be inconclusive.



## Module Type

There are various types of modules, and there is an ever-expanding list of them. The main ones are:

- **Task** - Application. The module is a Windows Application.
- **DLL** - The module is a Windows function library (*Dynamic Link Library*).
- **Driver** - The Module is a Windows Device Driver. This is in fact a Ring 3 DLL and must not be confused with VxDs. There are various types of Windows device drivers:
  - **Comms** - A Windows Communication Device Driver.
  - **Printer** - A Windows Printer Driver.
  - **Keyboard** - A Windows Keyboard Driver.
  - **Language** - A Windows Language Driver - used when the country is not the US.
  - **Video** - A Windows Display Driver (video card).
  - **Mouse** - A Windows Mouse Driver.
  - **Network** - A Windows Network Driver.
  - **System** - A Windows System Driver - Windows Internal.
  - **Install** - A Windows Installable Driver - i.e. can be installed / uninstalled as needed. May be any other driver type (e.g. Sound Driver).
  - **Sound** - A Windows Sound Driver.
- **Font** - The module is a font. There are various font types:
  - **ATM** - An ATM Font. This font is scalable. Like TrueType but not supported internally by Windows.
  - **Raster** - A Raster Font. This file is made up of dots and does not look too good when scaled up.
  - **Vector** - A Vector Font. This is generally a plotter font which is made up by lines. If blown, areas may remain unfilled.
  - **TrueType** - A TrueType Font. This is a scalable font and is used for both printer and display.
- **VxD** - Contains a Windows Virtual Device Driver, a Ring 0 device.
- **Static** - Contains a static library which is linked to the application by the linker. Therefore, there is no need for a DLL.

**Module Location (File Name)**

The path and file name of the module.

## **Module Operating System**

The operating system for which the module was designed to run on:

- **16bit Windows** - Windows 3.0+ or compatible.
- **32bit Windows** - Windows 3.1+ & Win32s or Windows NT.
- **Windows NT** - Only Windows NT.

**Note:** While most 16bit applications run on Windows NT or 95, they may behave unpredictably.

**Note:** Some 32bit applications run on Windows 3.1 or later + Win32s, other use Windows NT functions not supported by Win32s. Most Windows 95 applications would run on Windows NT and vice-versa.

**Module Description**

The description of the module as given by the software company. This may include the name of the application package and a very brief note about the role of the module, i.e. what it does.

**Note:** Some modules lack this information.

**Module Version**

The version of the module file name and the revision. This should be the same as the Windows Build Version if it came with the Windows package. Otherwise, it may be the same as the package version, though there are exceptions.

**Product Description**

The description of the whole package (product) to which the module belongs to as given by the software company. This should include the name of the application package and maybe version.

**Note:** Some modules lack this information.

**Product Version**

The main version of the package (product) which contains the module and its revision. This should be the same as the Windows Build Version if it came with the Windows package.

**Product Copyright**

The copyright owner of the software module. This is usually the software company which created the whole package (product) which contains the module, though there are exceptions. It may also include the date(s) of first / last publishing.



**Product Company Name**

The name of the software company which created the module and which usually is the copyright holder.



