

# 1 Small System Monitor

A small program that has been created to monitor system health/performance.

Created by: Sytse B. van Werkhoven (spicts.nl)

This document is based on SSM version: 1.4

## 1.1 General Syntax

Syntax:

SSM.exe <Action> [<Parameter1> [<Parameter2> [...]]] [Debug] [Smart]

Additional to the <Parameters> of the <Action> you can specify “Debug” or “Smart”, meaning:

- **Debug:** the script will output intermediate debug information during the processing of the requested action. Without this parameter all the actions output **ONLY** the return values of the requested action.
- **Smart:** Enables the smart-functions (where applicable).

## 2 System Monitor Actions

Valid Actions:

- CPU Actions:
  - CPU.Counter
  - CPU.Load
  - CPU.QueueLength
- Disk Actions:
  - Disk.AverageDisksecPerRead
  - Disk.AverageDisksecPerWrite
  - Disk.Free
  - Disk.IdleTime
  - Disk.QueueLength
  - Disk.Size
  - Disk.SmartFree (Obsolete, use Smart parameter instead)
  - Disk.SmartUsed (Obsolete, use Smart parameter instead)
  - Disk.Status
  - Disk.Type
  - Disk.Used
- Memory Actions:
  - Memory.Free
  - Memory.PageFaults
  - Memory.Size
  - Memory.Used



- Network Actions:
  - Network.ConnectedNICs
  - Network.QueueLength
  - Network.Utilization
- Swap Actions:
  - Swap.Free
  - Swap.Size
  - Swap.Used
- System Actions:
  - System.Uptime
- Volume Actions:
  - Volume.Free
  - Volume.FreeTrigger
  - Volume.Size
  - Volume.Used

These actions are explained in more detail in the following sections.

## **2.1 CPU Functions**

These functions provide information about the CPU.

### **2.1.1 CPU.Counter**

Syntax: SSM CPU.Counter

Returns the number of CPU's available to the Operating System, this also included the number of cores. E.g. on system with 2 dual-core CPU's the function returns 4.

### **2.1.2 CPU.Load**

Syntax: SSM CPU.Load [<Option>]

Where:

<Option>: The load you would like to query, possible options are:

- Load: The CPU load of the system in percent (Default)
- Idle: The CPU Idle time in percent
- Priviledged: The CPU Priviledged time in percent
- User: The CPU User time in percent

Returns the CPU times averaged for the number of CPU's on a scale from 0 to 100. E.g. the Load on a dual-core system with one core running at 100% and the other running idle the function returns 50.

### **2.1.3 CPU.QueueLength**

Syntax: SSM CPU.QueueLength



This function returns the number of threads that are ready to run (thus wanting execution time). A value observed here greater than 2 means that that the delays in execution may occur.

## **2.2 Disk Functions**

These functions provide information about the drives currently available to the Operating System.

### **2.2.1 Disk.AverageDisksecPerRead**

Syntax: SSM Disk.AverageDisksecPerRead [<DriveLetter>]

Where:

<DriveLetter>: The drive that you would like to query (Defaults to C)

This function returns the current average time of a read operation of data from the logical disk <DriveLetter>.

Note: The output of this function is probably in milliseconds.

### **2.2.2 Disk.AverageDisksecPerWrite**

Syntax: SSM Disk.AverageDisksecPerWrite [<DriveLetter>]

Where:

<DriveLetter>: The drive that you would like to query (Defaults to C)

This function returns the current average time of a write operation of data to the logical disk <DriveLetter>.

Note: The output of this function is probably in milliseconds.

### **2.2.3 Disk.Free**

Syntax: SSM Disk.Free [<DriveLetter> [<MB/Percent>]] [<PageFileTrigger> [Smart]]

Where:

<DriveLetter>: The drive that you would like to check (Defaults to C)

<MB/Percent>: Output in MegaBytes (MB) or percentage (Percent) (Defaults to MB)

<PageFileTrigger>: Percentage of disk when disk becomes a swap drive (Defaults to 80)

This function returns the free space on the specified drive, depending on the second parameter in megabytes or percentage of the total disk size.

When the Smart parameter is supplied the function returns no output when:

- the <DriveLetter> is not a local drive
- the <DriveLetter> is a local drive that is used as swap drive

A swap drive is considered to be a local drive that contains a pagefile that occupies more than <PageFileTrigger> percent of the total disk space.



Note: Returns 0 also when it is unable to retrieve the free space.

Note2: <PageFileTrigger> parameter is only used when the Smart parameter is supplied

#### **2.2.4 Disk.IdleTime**

Syntax: SSM Disk.IdleTime [<DriveLetter>]

Where:

<DriveLetter>: The drive that you would like to query (Defaults to C)

This function returns the current idle time in percent of the logical disk <DriveLetter>.

#### **2.2.5 Disk.QueueLength**

Syntax: SSM Disk.QueueLength

This function returns the current length of the queue with pending disk operations. A value observed here greater than 2 can lead to system delays.

#### **2.2.6 Disk.Size**

Syntax: SSM Disk.Size [<DriveLetter>]

Where:

<DriveLetter>: The drive that you would like to check (Defaults to C)

This function returns the total size of the specified drive in megabytes.

Note: Returns 0 also when it is unable to retrieve the disk size.

#### **2.2.7 Disk.SmartFree**

This function is obsolete. Please use the Disk.Free function together with the Smart parameter.

#### **2.2.8 Disk.SmartUsed**

This function is obsolete. Please use the Disk.Used function together with the Smart parameter.

#### **2.2.9 Disk.Status**

Syntax: SSM Disk.Status

This function returns the combines SMART status of all the local disks in the system.

Note: The reported status is the status of the last disk that is not OK, e.g. in a system with 4 disks with disk 1 and 2 “OK”, disk 3 “Pred Fail” and disk 4 “OK” the returned status is: “Pred Fail”.



Possible return values are:

"OK", "Error", "Degraded", "Unknown", "Pred Fail", "Starting", "Stopping", "Service", "Stressed", "NonRecover", "No Contact", "Lost Comm"

### 2.2.10 Disk.Type

Syntax: SSM Disk.Type [<DriveLetter>]

Where:

<DriveLetter>: The drive that you would like to check (Defaults to C)

This function returns the type of the drive specified by <DriveLetter>.

Possible return values:

- CompactDisk: <DriveLetter> is a CD- or DVD-Drive
- LocalDisk: <DriveLetter> is a local drive
- NetworkDrive: <DriveLetter> is a network drive
- RAMDisk: <DriveLetter> is a RAM drive
- Removable: <DriveLetter> is a drive with removable media (excl. CD/DVD)
- Unknown: the type of <DriveLetter> can't be determined

### 2.2.11 Disk.Used

Syntax: SSM Disk.Used [<DriveLetter> [<MB/Percent>]] [<PageFileTrigger> [Smart]]

Where:

<DriveLetter>: The drive that you would like to check (Defaults to C)

<MB/Percent>: Output in MegaBytes (MB) or percentage (Percent) (Defaults to MB)

<PageFileTrigger>: Percentage of disk when disk becomes a swap drive (Defaults to 80)

This function returns the used space on the specified drive, depending on the second parameter in megabytes or percentage of the total disk size.

When the Smart parameter is supplied the function returns no output when:

- the <DriveLetter> is not a local drive
- the <DriveLetter> is a local drive that is used as swap drive

A swap drive is considered to be a local drive that contains a pagefile that occupies more than <PageFileTrigger> percent of the total disk space.

Note: Returns 0 also when it is unable to retrieve the used space.

Note2: <PageFileTrigger> parameter is only used when the Smart parameter is supplied

## 2.3 Memory Actions

These functions provide information about the physical memory that is currently available to the Operating System.



### **2.3.1 Memory.Free**

Syntax: SSM Memory.Free [<MB/Percent>]

Where:

<MB/Percent>: Output in MegaBytes (MB) or percentage (Percent) (Defaults to MB)

This function returns the free physical memory, depending on the second parameter in megabytes or percentage of the total physical memory size.

Note: Returns 0 also when it is unable to retrieve the free memory.

### **2.3.2 Memory.PageFaults**

Syntax: SSM Memory.PageFaults [<MeasureTime>]

Where

<MeasureTime>: Time in seconds to measure the data (Defaults to 1 second)

This function returns the number of hard page faults that have occurred in the <MeasureTime>. A hard page fault is a page fault that is recovered by reading the requested memory page from Virtual Memory (in addition to soft page faults that can be recovered by loading a page from another location in physical memory).

### **2.3.3 Memory.Size**

Syntax: SSM Memory.Size

This function returns the total size of the physical memory in megabytes that is present in the system. This number does not include virtual memory.

Note: Returns 0 also when it is unable to retrieve the memory size.

### **2.3.4 Memory.Used**

Syntax: SSM Memory.Used [<MB/Percent>]

Where:

<MB/Percent>: Output in MegaBytes (MB) or percentage (Percent) (Defaults to MB)

This function returns the used physical memory, depending on the second parameter in megabytes or percentage of the total physical memory size.

Note: Returns 0 also when it is unable to retrieve the used memory.

## **2.4 Network Actions**

These functions provide information about the network interface controller (NIC) in the system.



### **2.4.1 Network.ConnectedNICs**

Syntax: SSM Network.ConnectedNICs

This function returns the number of network cards that are currently connected (that have a functional network connection).

Note: The local loopback interface is not included in this number.

### **2.4.2 Network.QueueLength**

Syntax: SSM Network.QueueLength

This function returns the current length of the queue with pending network operations. A value observed here greater than 2 can lead to network delays.

### **2.4.3 Network.Utilization**

Syntax: SSM Network.Utilization [<bps/Percent> [<MeasureTime>]]

Where:

<Bps/Percent>: Output in bytes/second (Bps) or percentage (Percent) (Defaults to Bps)

<MeasureTime>: Time in seconds to measure the data (Defaults to 1 second)

This function returns the combined utilization of all the connected network cards in bytes per second or percentage of the current bandwidth.

## **2.5 Swap Actions**

These functions provide information about the current amount of virtual memory available to the Operating System. When the system has its virtual memory (swap files) divided onto multiple disks the total of all used swap files is used.

### **2.5.1 Swap.Free**

Syntax: SSM Swap.Free [<MB/Percent>]

Where:

<MB/Percent>: Output in MegaBytes (MB) or percentage (Percent) (Defaults to MB)

This function returns the free virtual memory, depending on the second parameter in megabytes or percentage of the total virtual memory size.

Note: Returns 0 also when it is unable to retrieve the free virtual memory.

### **2.5.2 Swap.Size**

Syntax: SSM Swap.Size

This function returns the maximum size of the virtual memory in megabytes that the Operating System is allowed to allocate.



Note: Returns 0 also when it is unable to retrieve the memory size.

### **2.5.3 Swap.Used**

Syntax: SSM Swap.Used [<MB/Percent>]

Where:

<MB/Percent>: Output in MegaBytes (MB) or percentage (Percent) (Defaults to MB)

This function returns the used virtual memory, depending on the second parameter in megabytes or percentage of the total virtual memory size.

Note: Returns 0 also when it is unable to retrieve the used virtual memory.

## **2.6 System Actions**

These functions provide information about the current running Operating System.

### **2.6.1 System.Uptime**

Syntax: SSM System.Uptime

This function returns the number of seconds that the system is running since the last boot.

Note: This function returns a value with engineering precision.

## **2.7 Volume Actions**

These functions provide information about the volumes that are available in the system and can therefore also be used on volume mountpoints. When you want to use them on local disks you should use these functions with the name of the disk as <VolumeName>.

### **2.7.1 Volume.Free**

Syntax: SSM Volume.Free <VolumeLabel> [<MB/Percent>]

Where:

<VolumeLabel>: The name of the volume you want to query

<MB/Percent>: Output in MegaBytes (MB) or percentage (Percent) (Defaults to MB)

This function returns the free space of the volume with the name <VolumeLabel>, depending on the second parameter in megabytes or percentage of the total size of the volume.

Note: Returns nothing when it is unable to find the specified volume.



## 2.7.2 Volume.FreeTrigger

Syntax: SSM Volume.FreeTrigger [<Trigger>]

Where:

<Trigger >: The trigger value in percent (Defaults to 10)

This function returns the names of the volumes where the free space is below <Trigger> percentage of the total size of the volume.

Note: Returns nothing when all volumes are above the trigger value.

## 2.7.3 Volume.Size

Syntax: SSM Volume.Size <VolumeLabel>

Where:

<VolumeLabel>: The name of the volume you want to query

This function returns the size of the volume with the name <VolumeLabel>.

Note: Returns nothing when it is unable to find the specified volume.

## 2.7.4 Volume.Used

Syntax: SSM Volume.Used <VolumeLabel> [<MB/Percent>]

Where:

<VolumeLabel>: The name of the volume you want to query

<MB/Percent>: Output in MegaBytes (MB) or percentage (Percent) (Defaults to MB)

This function returns the used space of the volume with name <VolumeLabel>, depending on the second parameter in megabytes or percentage of the total size of the volume.

Note: Returns nothing when it is unable to find the specified volume.

