

# **SysInfo 10.1.1 H5 User Guide**

Copyright © 2023 MagniComp™

---

<b>COLLABORATORS</b>
----------------------

	<i>TITLE :</i> SysInfo 10.1.1 H5 User Guide	
<i>ACTION</i>	<i>NAME</i>	<i>DATE</i>
WRITTEN BY		October 13, 2023
		<i>SIGNATURE</i>

<b>REVISION HISTORY</b>
-------------------------

NUMBER	DATE	DESCRIPTION	NAME

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Related Documentation . . . . .	1
1.2	Target Audience . . . . .	1
1.3	Overview . . . . .	1
<b>2</b>	<b>Account Data Class</b>	<b>2</b>
2.1	Introduction . . . . .	2
2.2	Account Data Class Attributes . . . . .	2
2.2.1	Account Data Details . . . . .	2
<b>3</b>	<b>Bios Data Class</b>	<b>4</b>
3.1	Introduction . . . . .	4
3.2	Bios Data Class Attributes . . . . .	4
3.2.1	Bios Data Details . . . . .	4
<b>4</b>	<b>General Data Class</b>	<b>5</b>
4.1	Introduction . . . . .	5
4.2	General Data Class Attributes . . . . .	5
4.2.1	General Data Details . . . . .	5
<b>5</b>	<b>Hardware Data Class</b>	<b>7</b>
5.1	Introduction . . . . .	7
5.2	DevInfo Data Class Attributes . . . . .	7
5.2.1	DevInfo Data Details . . . . .	7
5.3	DiskDrive Data Class Attributes . . . . .	7
5.3.1	DiskDrive Data Details . . . . .	7
5.4	FrameBuffer Data Class Attributes . . . . .	7
5.4.1	FrameBuffer Data Details . . . . .	7
5.5	Monitor Data Class Attributes . . . . .	7
5.5.1	Monitor Data Details . . . . .	7
5.6	PartInfo Data Class Attributes . . . . .	7
5.6.1	PartInfo Data Details . . . . .	7
5.7	Hardware Class Notes . . . . .	7
5.7.1	SCSI Bus, Unit, and LUN . . . . .	7

---

---

<b>6</b>	<b>Job Data Class</b>	<b>13</b>
6.1	Introduction . . . . .	13
6.2	Job Data Class Attributes . . . . .	13
6.2.1	Job Data Details . . . . .	13
<b>7</b>	<b>KernelVar Data Class</b>	<b>15</b>
7.1	Introduction . . . . .	15
7.2	KernelVar Data Class Attributes . . . . .	15
7.2.1	KernelVar Data Details . . . . .	15
<b>8</b>	<b>License Data Class</b>	<b>16</b>
8.1	Introduction . . . . .	16
8.2	License Data Class Attributes . . . . .	16
8.2.1	License Data Details . . . . .	16
<b>9</b>	<b>Netif Data Class</b>	<b>18</b>
9.1	Introduction . . . . .	18
9.2	NetIf Data Class Attributes . . . . .	18
9.2.1	NetIfAddr Data Details . . . . .	18
9.2.2	NetIf Data Details . . . . .	18
<b>10</b>	<b>Network Data Class</b>	<b>20</b>
10.1	Introduction . . . . .	20
10.2	Network Data Class Attributes . . . . .	20
10.2.1	Network Data Details . . . . .	20
<b>11</b>	<b>Patch Data Class</b>	<b>22</b>
11.1	Introduction . . . . .	22
11.2	Patch Data Class Attributes . . . . .	22
11.2.1	Patch Data Details . . . . .	22
<b>12</b>	<b>Printer Data Class</b>	<b>23</b>
12.1	Introduction . . . . .	23
12.2	Printer Data Class Attributes . . . . .	23
12.2.1	Printer Data Details . . . . .	23
<b>13</b>	<b>Process Data Class</b>	<b>24</b>
13.1	Introduction . . . . .	24
13.2	Process Data Class Attributes . . . . .	24
13.2.1	Process Data Details . . . . .	24

---

---

<b>14 Service Data Class</b>	<b>25</b>
14.1 Introduction	25
14.2 Service Data Class Attributes	25
14.2.1 Service Data Details	25
<b>15 SiteInfo Data Class</b>	<b>26</b>
15.1 Introduction	26
15.1.1 Setting SiteInfo Data	26
15.1.1.1 Setting SiteInfo Data - File Method	26
15.1.1.2 Setting SiteInfo Data - Using the CLI	26
15.2 SiteInfo Data Class Attributes	27
15.2.1 SiteInfo Data Details	27
15.3 More Information	27
<b>16 Software Data Class</b>	<b>28</b>
16.1 Introduction	28
16.2 SoftInfo Data Class Attributes	28
16.2.1 SoftInfo Data Details	28
16.3 SoftFile Data Class Attributes	28
16.3.1 SoftFile Data Details	28
<b>17 Storage System Data Class</b>	<b>31</b>
17.1 Introduction	31
17.2 EMC Support	31
17.2.1 Introduction	31
17.3 Network Appliance Filer Support	31
17.3.1 Introduction	31
17.3.2 Discovery of Filers	31
17.3.3 Data Acquisition	32
17.4 StorSys Data Class Attributes	32
17.4.1 MicroCode Data Details	32
17.4.2 FibrePort Data Details	32
17.4.3 StorPort Data Details	32
17.4.4 StorFanBank Data Details	32
17.4.5 StorPwrSupply Data Details	32
17.4.6 StorEnc Data Details	34
17.4.7 StorMemBrd Data Details	34
17.4.8 StorRemData Data Details	34
17.4.9 StorSnap Data Details	34
17.4.10 StorPart Data Details	34

---

17.4.11 StorMeta Data Details . . . . .	36
17.4.12 StorGroup Data Details . . . . .	36
17.4.13 StorHostDev Data Details . . . . .	36
17.4.14 StorDevice Data Details . . . . .	36
17.4.15 StorDisk Data Details . . . . .	36
17.4.16 StorCtrl Data Details . . . . .	36
17.4.17 StorLicense Data Details . . . . .	36
17.4.18 StorSys Data Details . . . . .	36
<b>18 StorVol Data Class</b>	<b>41</b>
18.1 Introduction . . . . .	41
18.2 StorVol Data Class Attributes . . . . .	41
18.2.1 StorVolSw Data Details . . . . .	41
18.2.2 StorVolDisk Data Details . . . . .	41
18.2.3 StorVolPart Data Details . . . . .	42
18.2.4 StorVolPlex Data Details . . . . .	42
18.2.5 StorVolume Data Details . . . . .	42
18.2.6 StorVolGroup Data Details . . . . .	42
18.2.7 StorVol Data Details . . . . .	42
<b>19 SysConf Data Class</b>	<b>45</b>
19.1 Introduction . . . . .	45
19.2 SysConf Data Class Attributes . . . . .	45
19.2.1 SysConf Data Details . . . . .	45
<b>20 Virtual Machine Guests Data Class</b>	<b>46</b>
20.1 Introduction . . . . .	46
20.2 VmGuest Data Class Attributes . . . . .	46
20.2.1 VmGuest Data Details . . . . .	46
<b>21 Virtual Machine Host Data Class</b>	<b>48</b>
21.1 Introduction . . . . .	48
21.2 Data Details . . . . .	48
<b>22 Software Discovery Language™ (SDL)</b>	<b>50</b>
22.1 Introduction . . . . .	50
22.1.1 Syntax . . . . .	50
22.1.1.1 Data Types . . . . .	51
22.1.1.2 Variables . . . . .	51
22.1.1.3 Logical Constructs . . . . .	51
22.1.1.3.1 IF Statements . . . . .	51

---

---

22.1.1.3.2	FOR Statements . . . . .	52
22.1.1.3.3	FOREACH Statements . . . . .	52
22.1.1.3.4	WHILE Statements . . . . .	52
22.1.1.3.5	Special Loop Instructions . . . . .	52
22.2	Standard Functions . . . . .	52
22.3	Custom Functions . . . . .	55
22.4	Basic Example Script . . . . .	55
22.5	SDL Script Detection . . . . .	56
22.6	Testing and Debugging . . . . .	56

---

# List of Tables

2.1	Account Data Details . . . . .	3
3.1	Bios Data Details . . . . .	4
4.1	General Data Details . . . . .	6
5.1	DevInfo Data Details . . . . .	8
5.2	DiskDrive Data Details . . . . .	9
5.3	FrameBuffer Data Details . . . . .	10
5.4	Monitor Data Details . . . . .	10
5.5	PartInfo Data Details . . . . .	11
5.6	SCSI ID Mapping . . . . .	11
6.1	Job Data Details . . . . .	14
7.1	KernelVar Data Details . . . . .	15
8.1	License Discovery Support on Windows . . . . .	16
8.2	License Data Details . . . . .	17
9.1	NetIfAddr Data Details . . . . .	18
9.2	NetIf Data Details . . . . .	19
10.1	Network Data Details . . . . .	21
11.1	Patch Data Details . . . . .	22
12.1	Printer Data Details . . . . .	23
13.1	Process Data Details . . . . .	24
14.1	Service Data Details . . . . .	25
15.1	SiteInfo Data Details . . . . .	27
16.1	SoftInfo Data Details . . . . .	29

---

---

16.2	SoftFile Data Details . . . . .	30
17.1	MicroCode Data Details . . . . .	32
17.2	FibrePort Data Details . . . . .	32
17.3	StorPort Data Details . . . . .	33
17.4	StorFanBank Data Details . . . . .	33
17.5	StorPwrSupply Data Details . . . . .	33
17.6	StorEnc Data Details . . . . .	33
17.7	StorMemBrd Data Details . . . . .	34
17.8	StorRemData Data Details . . . . .	34
17.9	StorSnap Data Details . . . . .	34
17.10	StorPart Data Details . . . . .	35
17.11	StorMeta Data Details . . . . .	35
17.12	StorGroup Data Details . . . . .	36
17.13	StorHostDev Data Details . . . . .	36
17.14	StorDevice Data Details . . . . .	37
17.15	StorDisk Data Details . . . . .	38
17.16	StorCtrlr Data Details . . . . .	39
17.17	StorLicense Data Details . . . . .	39
17.18	StorSys Data Details . . . . .	40
18.1	StorVolSw Data Details . . . . .	41
18.2	StorVolDisk Data Details . . . . .	41
18.3	StorVolPart Data Details . . . . .	42
18.4	StorVolPlex Data Details . . . . .	43
18.5	StorVolume Data Details . . . . .	43
18.6	StorVolGroup Data Details . . . . .	44
18.7	StorVol Data Details . . . . .	44
19.1	SysConf Data Details . . . . .	45
20.1	VmGuest Data Details . . . . .	47
21.1	VM Host Data Details . . . . .	49
22.1	SDL Data Types . . . . .	51
22.2	Standard Functions . . . . .	53
22.3	SoftInfoSet() Key Definetions . . . . .	54

---

# Chapter 1

## Introduction

### 1.1 Related Documentation

- [Release Notes](#)
- [Installation Guide](#)
- [Reference Manual](#)

### 1.2 Target Audience

The target audience for this guide are all users of MagniComp™'s SysInfo™ software. The audience includes IT professionals, System Administrators, Software Engineers, and end-users interested in hardware and software asset information as well as system configuration information.

### 1.3 Overview

This book provides information on how SysInfo™ functions and operates. The [Reference Manual](#) contains the most detailed information on SysInfo™ configuration and operation. This document provides operational details not normally found in the [Reference Manual](#).

---

## Chapter 2

# Account Data Class

### 2.1 Introduction

The Account data class provides data about all system (user) accounts defined on the local system. On some systems, such as Unix/Linux based systems which have NIS/NIS+ configured, accounts may be reported which are managed by a central database of accounts. The Account class is not intended to report all accounts defined in server based databases like Active Directory, NIS/NIS+, and LDAP.

On Windows based systems, Active Directory accounts are explicitly not reported since those accounts are not defined exclusively on the local system being scanned.

### 2.2 Account Data Class Attributes

#### 2.2.1 Account Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
User Name	UserName	Username used to login to system.	
Unique ID	UserID	Unique system identifier for user.	
Type	Type	Type of account.	
Status	Status	Status of account.	
Account Database	Database	Database source where account is defined.	
First Name	FirstName	First name of account user.	
Last Name	LastName	Last name of account user.	
Full Name	FullName	Full name (first middle last) of account user.	
Comment	Comment	Comment about this account.	
Home Directory	HomeDir	Home (login) directory.	
Shell	Shell	User's shell program executed at login.	
Primary Group Name	PrimaryGroupName	Name of user's primary group membership.	
Primary Group ID	PrimaryGroupID	ID of user's primary group membership.	
Group Names	GroupNames	List of system groups user is a member of.	
Group IDs	GroupIDs	List of system group IDs user is a member of.	

Table 2.1: Account Data Details

## Chapter 3

# Bios Data Class

### 3.1 Introduction

The Bios data class provides data about the system BIOS(s). This class is primarily limited to X86 platforms.

### 3.2 Bios Data Class Attributes

#### 3.2.1 Bios Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Name	Name	Name of this BIOS	
Description	Desc	Description of BIOS	
Vendor	Vendor	Hardware Vendor	
Model	Model	Model of BIOS	
Version	Version	Version of BIOS	
Revision	Revision	Revision of BIOS	
Release Date	ReleaseDate	Release Date	
Release Date (Number)	ReleaseDateNum	Release Date Number	

Table 3.1: Bios Data Details

## Chapter 4

# General Data Class

### 4.1 Introduction

The General data class provides a high-level summary of a system. Information includes CPU counts, host name, IP addresses, Operating System name and versions, and the system model.

### 4.2 General Data Class Attributes

#### 4.2.1 General Data Details

Name	Keyword	Description	Examples
Host Name	hostname	The canonical hostname.	speed.acme.com
Host Name Aliases	hostaliases	Aliases the host is known by.	quick, fast
Host IP Addresses	hostaddrs	The IP address(es) used by the host.	192.168.1.1, 192.168.99.1
System Host Identifier	hostid	The host identifier of the system. This is not usually the system serial number, but some form of unique ID based on the systems boot PROM or ethernet MAC address.	05c0201a
System Serial Number	serial	The system's unique serial number which is typically assigned by the system manufacturer. On blade servers this value is the serial number of the blade module. This value may be overridden by creating a SysInfo SiteInfo entry using the key "systemserial".	A1234567BC
Chassis Serial Number	chassisserial	The chassis's unique serial number which is typically assigned by the system manufacturer. On blade servers this value is the serial number of the enclosure which is shared by all blade modules. This value may be overridden by creating a SysInfo SiteInfo entry using the key "chassisserial".	C124365B7C
System Manufacturer	man	The manufacturer of the system combining the short and long names.	Acme (Acme International Inc)
System Manufacturer (Short)	manshort	A short string specifying the manufacturer of the system.	Acme
System Manufacturer (Full)	manlong	A multi-word string specifying the manufacturer of the system.	Acme International Inc
System Model	model	The hardware system model.	Excalibur 4500
System Memory	memory	The amount of main system memory (RAM).	512 MB
System Virtual Memory	virtmem	The amount of virtual memory. This value usually includes the System Memory plus the swap/paging space.	1.2 GB
Application Architecture	arch	The application architecture of the system. This value represents the most common architecture supported by the system and CPU platform.	x86
Kernel Architecture	karch	The kernel architecture of the system. This value represents the architecture the system kernel supports. It's typically the most specific architecture	x64

## Chapter 5

# Hardware Data Class

### 5.1 Introduction

The Hardware data class (aka Device class) provides information on physical hardware detected on the system. The discovered hardware is classified into device type and sub types (known in SysInfo™ as Type Classes).

### 5.2 DevInfo Data Class Attributes

#### 5.2.1 DevInfo Data Details

### 5.3 DiskDrive Data Class Attributes

#### 5.3.1 DiskDrive Data Details

### 5.4 FrameBuffer Data Class Attributes

#### 5.4.1 FrameBuffer Data Details

### 5.5 Monitor Data Class Attributes

#### 5.5.1 Monitor Data Details

### 5.6 PartInfo Data Class Attributes

#### 5.6.1 PartInfo Data Details

### 5.7 Hardware Class Notes

#### 5.7.1 SCSI Bus, Unit, and LUN

A device's SCSI Bus, SCSI Unit, and SCSI LUN are reported as the device attributes as shown in the table below when the Device Type Class is SCSI. Currently on HP-UX and Linux provide this data due to limitations in other operating systems.

In SysInfo™ Report format these attributes appear in the `device|data|...` fields. For example:

---

Name	Keyword	Description	Examples
Name of Device	Name	The primary (canonical) name of the device as assigned by the operation system.	c0t0d0
Node Identifier	NodeID	The unique node ID typically used internally by the OS.	
Node Path	NodePath	The unique system node path typically used internally by the OS.	
Alternative Name	AltName	A well known alternative name used by the OS to refer to this device.	sd0
Aliases for Name	Aliases	A list of device name aliases this device is also known as to the operating system.	disk0, sdisk0
Compatibility Names for Device	CompatNames	A list of device class names this device is compatible with.	vga, display
Driver Name	Driver	The name of the device driver managing this device.	sd
Device Type	Type	The primary device type classification.	diskdrive
Device Class Type	ClassType	The secondary device type (sub) classification.	scsi
Vendor	Vendor	The hardware vendor or built this device.	Acme
Model	Model	The hardware model name.	A12345
Model Description	ModelDesc	A description of the model and/or additional info describing the hardware.	High Speed
Capacity	Capacity	The capacity (size, etc) of the hardware.	100 GB
Capacity (bytes)	CapacityNum	The capacity (size, etc) of the hardware in bytes.	100000000000
Speed	Speed	The speed of the device. Typically this is clock speed (in hertz), bits/second, or RPMs.	40 Mb
Part Number	Part	The part number assigned by the Vendor.	X1234
Revision	Revision	The hardware revision of the device.	A01
Serial Number	Serial	The serial number assigned by the Vendor to uniquely identify the device.	XN1234546
World Wide Identifier	WWID	Unique identifier which includes a unique vendor code and device identifier.	3600a098000a264070000042a5
Physical Location	Location	The physical location of the device. Usually this is a slot ID or mainboard location coordinate.	SLOT0
Status	Status	The status of the device. Typically this is whether the device is running or not.	ONLINE
State	State	The state of the device. Typically this is whether the device is functional or broken.	RUNNING
Device Files	Files	A list of device files made available by the OS to	/dev/dsk/c0t0d0,

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Data Type	DataType	The source of this data.	
Disk Label	Label	The label on the disk as usually written by the OS	
Unit Number	Unit	The unit number of the disk.	
Slave Number	Slave	The slave number of the disk.	
Data Cylinders	DataCyl	The number of usable data cylinders.	
Physical Cylinders	PhyCyl	The number of physical cylinders.	
Alternate Cylinders	AltCyl	The number of alternative cylinders	
Cylinder Skew	CylSkew	Cylinder Skew	
Alternates per Cylinder	APC	The number of alternates per cylinders.	
Tracks	Tracks	The number of tracks	
Alternate Tracks per Zone	AltTracksPerZone	The number of alternate tracks per zone	
Alternate Tracks per Volume	AltTracksPerVol	The number of alternate tracks per volume	
Track Skew	TrackSkew	Track Skew	
Sectors	Sect	The number of usable sectors per track	
Sector Size (bytes)	SecSize	The size of a sector in bytes	
Physical Sectors	PhySect	The number of physical sectors per track.	
Alternate Sectors per Zone	AltSectPerZone	The number of alternate sectors per zone.	
Sector Gap	SectGap	The length between sectors.	
Steps per Track	StepsPerTrack	The number of steps per track.	
Revolutions Per Minute	RPM	Speed in Revolutions Per Minute.	
Interleave Factor	IntrLv	The interleave factor.	
PROM Revision	PROMRev	The PROM revision.	
Size (MB)	Size	The capacity size.	
Flags	Flags	Flags	
Disk Controller	Ctlr	Controller the disk belongs to.	
Partition Label Type	PartLabelType	The type of partition label used to partition the disk.	
Partition Information	PartInfo	Partitions of the disk.	

Table 5.2: DiskDrive Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Height	Height	Height in pixels.	
Width	Width	Width in pixels.	
Vertical Refresh (Hz)	VRefresh	Vertical refresh rate in Hz.	
Depth	Depth	Color depth in bits/pixel.	
Size of Memory	Size	Total memory size in bytes.	
Video Memory Size	VideoMemSize	Size of video memory in bytes.	
Color Map Size	ColorMapSize	Size of color map (# of entries).	

Table 5.3: FrameBuffer Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Maximum Horizontal Size (cm)	MaxHorSize	Maximum Horizontal Size in cm	
Maximum Vertical Size (cm)	MaxVerSize	Maximum Vertical Size in cm	
Resolutions	Resolutions	List of supported video resolutions	
Minimum Vertical Refresh (Hz)	MinVerRefresh	Minimum Vertical Refresh rate supported	
Maximum Vertical Refresh (Hz)	MaxVerRefresh	Maximum Vertical Refresh rate supported	
Minimum Horizontal Refresh (KHz)	MinHorRefresh	Minimum Horizontal Refresh rate supported	
Maximum Horizontal Refresh (KHz)	MaxHorRefresh	Maximum Horizontal Refresh rate supported	
Maximum Pixel Clock (MHz)	MaxPixelClock	Maximum Pixel Clock rate supported	

Table 5.4: Monitor Data Details

Name	Keyword	Description	Examples
Title	Title	Title of partition.	
Device Path	DevPath	Path to block device.	/dev/dsk/c0t0s1
Device Path Raw	DevPathRaw	Path to raw device.	/dev/rdisk/c0t0s1
Device Name	DevName	Name of device.	c0t0
Base Device Name	BaseName	Base name of device.	c0t0s1
Partition Name	Name	Name of partition.	s1
Partition Number	Number	Partition Number.	1
Mount Name	MntName	Mounted name.	/usr
Mount Options	MntOpts	Mount options.	rw, quota
Type	Type	Mount Type.	nfs
Type Description	TypeDesc	Mount Type Description.	nfs
Type Number	TypeNum	Numeric value for type.	
Usage Status	UsageStatus	Usage status.	
Manager of Partition	Manager	Subsystem which manages/controls the partition.	
Size (MB)	Size	Size of partition in megabytes.	
Used Space (MB)	AmtUsed	Amount of space used in megabytes.	
Sector Size (bytes)	SectSize	Size of each sector in bytes.	
Starting Sector	StartSect	Sector of disk where partition begins.	
Ending Sector	EndSect	Sector of disk where partition ends.	
Number of Sectors	NumSect	The number of sectors on the disk that partition occupies.	
Usage Type	Usage	Usage type of partition.	

Table 5.5: PartInfo Data Details

SCSI ID	SysInfo™ Device Field
SCSI Bus #	Bus
SCSI Target/Unit #	BusDevNum
SCSI LUN	Port

Table 5.6: SCSI ID Mapping

```
device|data|||sdd|||Bus|Bus Number|1
device|data|||sdd|||BusDevNum|Bus Device Number|10
device|data|||sdd|||Port|Bus Port Number|0
```

In this example the SCSI Bus number is 1, the SCSI Target is 10, and the SCSI LUN is 0.

## Chapter 6

# Job Data Class

### 6.1 Introduction

The Job data class provides information about all scheduled jobs (sometimes called tasks) on a given system.

On Unix/Linux based systems **cron(1)** based jobs are reported. All system (root) and user defined **cron(1)** jobs are reported. SysInfo™ will scan all standard **cron(1)** locations for **crontab(5)** files which are parsed and reported as Job entries.

On Windows based systems the Windows Scheduled Tasks are scanned and reported.

### 6.2 Job Data Class Attributes

#### 6.2.1 Job Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Job ID	ID	Identifier for a job.	
Job Name	Name	Name of Job.	
Job Scheduler Name	SchedulerName	Name of job scheduler used to run and manage this job.	
Job Scheduler File	SchedulerFile	Path to file job is defined in.	
Program	Program	The basename of the program being run.	
ProgramPath	ProgramPath	The full pathname to the program being run.	
Program Arguments	ProgramArgs	The full list of command arguments used to run the job.	
Program Command Line	ProgramCmd	The ProgramPath and the full list of command arguments used to run the job.	
Runtime Directory	RunDir	The directory the job is started from.	
Owner User Name	OwnerUserName	The username the job is owned/created by.	
Owner User ID	OwnerUserID	The user identifier the job is owned/created by.	
Run As User Name	RunAsUserName	The user name that the job runs as.	
Run As User ID	RunAsUserID	The user identifier that the job runs as.	
Schedule Type	ScheduleType	The type of schedule the job runs on.	
Start Date	StartDate	The date the job is scheduled to first run.	
Start Time	StartTime	The time the job is scheduled to run.	
End Date	EndDate	The date the job is scheduled to last run.	
End Time	EndTime	The time the job is scheduled to finish.	
Months Job Runs	RunMonths	List of numeric months (1-12) that the job runs.	
Days of Month Job Runs	RunMonthDays	List of numeric days (1-31) of a month that the job runs.	
Days of Week Job Runs	RunWeekDays	List of days (3 letter abbreviated day names) of a week that the job runs.	
Hours of Day Job Runs	RunHours	List of numeric hours (0-23 where 0=midnite) of a day that the job runs.	
Minutes of an Hour Job Runs	RunMinutes	List of numeric minutes (0-59) of an hour that the job runs.	
Run Every N Days	RunEveryDays	Number of days between repeating interval the job is run.	
Run Every N Weeks	RunEveryWeeks	Number of weeks between repeating interval the job is run.	
Run Every N Months	RunEveryMonths	Number of months between repeating interval the job is run.	
Maximum Time Job	RunMaxTime	Maximum amount of time job is permitted to run.	
Next Run Time	NextRunTime	The time and date the job will next run.	
		The time and date the job	

## Chapter 7

# KernelVar Data Class

### 7.1 Introduction

The KernelVar data class provides data about kernel variables. Supported is limited primarily to UNIX based operating systems.

### 7.2 KernelVar Data Class Attributes

#### 7.2.1 KernelVar Data Details

Name	Keyword	Description	Examples
Kernel Variable Name	Name		
Kernel Variable Description	Description		
Kernel Variable Numeric Type	Type		
Kernel Variable Value	Value		

Table 7.1: KernelVar Data Details

## Chapter 8

# License Data Class

### 8.1 Introduction

The License data class provides data about software licenses found on the system. The license class may contain a more comprehensive list of licenses and more license data than is reported in the Software data class.

On Linux/UNIX systems supported is limited to FLEXlm based licenses. FLEXlm is a widely used license product used by many vendors to provide floating and node locked licenses.

The following table describes the licenses that should be reported on Microsoft Windows based systems.

Vendor	Products	Notes
Adobe	Acrobat 7 and later	
Adobe	Other	Other Adobe products released after 2005 will likely be reported.
Microsoft	Office 2003, XP, 2007, 2010	
Microsoft	Windows XP, Vista, 7	
Microsoft	Windows Server 2000, 2003, 2008	
Microsoft	Other	Most Microsoft products released after 2003 should be discovered.

Table 8.1: License Discovery Support on Windows

Other products not listed in the above table may be reported if they confirm to common methods of storing license data in the Windows Registry.

### 8.2 License Data Class Attributes

#### 8.2.1 License Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Software Name	SoftName	Name of software.	
Software Version	SoftVer	Version of software	
Software Vendor Name	SoftVendor	Name of software vendor.	
License Type	Type	Type of license.	
License Category	Category	Category of license.	
Port Number	Port	Port number license server runs on.	
Allowed License Count	Count	Maximum number of licenses allowed to be used.	
File	File	File license resides in.	
Server Software Name	ServerSoft	Name of license server software.	
Server Software Version	ServerVer	Version of license server software.	
Vendor License Server Software	VendorServerSoft	Name of vendor license server software.	
Vendor License Server Version	VendorServerVer	Version of vendor license server software.	
License Issuer	Issuer	Name of license issuing organization.	
Host ID	Hostid	The Host ID license is tied/intended for.	
Issue Date	IssueDate	The date the license was issued.	
Start Date	StartDate	The date the license starts.	
Expire Date	ExpireDate	The date the license expires.	
Serial Number	Serial	License serial number.	
Key	Key	License key suitable for use during installation or authorization of the product.	
Key Encrypted	KeyEnc	Encrypted License Key.	

Table 8.2: License Data Details

## Chapter 9

# Netif Data Class

### 9.1 Introduction

The NetIf data class provides data about each network interface on a system.

### 9.2 Netif Data Class Attributes

#### 9.2.1 NetIfAddr Data Details

Name	Keyword	Description	Examples
Address Type	AddrType	The address type.	
Host Address	HostAddr	Address assigned for use on network.	
Host Name	HostName	Host name corresponding to Host Address.	
Network Address	NetAddr	Network Address portion of address.	
Network Name	NetName	Name of network for Network Address.	
Network Mask	NetMask	Network Mask	
Broadcast Address	Broadcast	Broadcast Address portion of Network Address.	
Gateway Address	GatewayAddr	Address of default gateway.	
Gateway Name	GatewayName	Name of host used as Gateway Address.	
Address MTU Size	MTU	Address MTU size	
Address Interface Flags	Flags	Address specific interface flags	

Table 9.1: NetIfAddr Data Details

#### 9.2.2 Netif Data Details

Name	Keyword	Description	Examples
Name of Interface	Name	The name of the device as assigned by the operation system.	
Driver Name	Driver	Device driver name.	
Unique Identifier	ID	Unique identifier for this device.	
Description of interface	Description	Description of the interface.	
Alternate Description	AltDescription	Alternate description of the interface.	
Interface Type	IfType	The type of interface.	
Current MAC Address	MACaddr	Current MAC Address.	
Current MAC Host Name	MACname	Current host name associated with MAC address.	
Factory MAC Address	FacMACaddr	Factory assigned MAC address.	
Factory MAC Host Name	FacMACname	Name associated with factory MAC address.	
Status	Status	Current status of interface.	
State	State	Current state of interface.	
Media Type	MediaType	Type of physical media.	
Media Sub Type	MediaSubType	Sub type of physical media.	
Link Speed	Speed	Current speed of link.	
Link Duplex	Duplex	Link duplex.	
MTU Size	MTU	MTU Size	
Unit number	Unit	Unit number.	
Model	Model	Model of hardware.	
Interface Flags	Flags	Interface flags	
DHCP Data	Dhcp	DHCP client data.	

Table 9.2: NetIf Data Details

## **Chapter 10**

# **Network Data Class**

### **10.1 Introduction**

The Network data class provides data about general network parameters on a system.

### **10.2 Network Data Class Attributes**

#### **10.2.1 Network Data Details**

---

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Default Gateway IPv4	defgwip	The default gateway IPv4 address	
Default Gateway IPv6	defgwip6	The default gateway IPv6 address	
Default Gateway IPv4 Hostname	defgwname	Hostname of default IPv4 gateway	
Default Gateway IPv6 Hostname	defgwname6	Hostname of default IPv6 gateway	
IP Forwarding Enabled?	IpForwarding	Is IP Forwarding Enabled?	
DNS Domain Name	DnsDomain	DNS Domain Name system belongs to	
DNS Server IP Addresses	DnsServerIPs	DNS Server IP addresses.	
DNS Server Names	DnsServerNames	DNS Server Hostnames.	
NIS Domain Name	NisDomain	NIS Domain Name	
NIS Server Binding Method	NisServerMethod	The method NIS uses to bind/find NIS server(s).	
NIS Server IPs	NisServerIPs	NIS Server IP addresses.	
NIS Server Names	NisServerNames	NIS Server Hostnames.	
Current NIS Server	nissserver	Current NIS server hostname system is bound to	
NIS Master Hostname	NisMaster	NIS Master Hostname	
Windows Workgroup Name	WinWorkgroupName	Name of Windows Workgroup system uses	
Windows Domain Name	WinDomainName	Name of Windows Domain system uses	
Windows AD Server Names	WinAdServerNames	Windows Active Directory Server Names	
Windows Primary Domain Controller	WinPdcName	Name of Windows Primary Domain Controller	
WINS Proxy Enabled?	WinWinsProxyEnabled	WINS Proxy is Enabled?	

Table 10.1: Network Data Details

## Chapter 11

# Patch Data Class

### 11.1 Introduction

The Patch data class provides data about all the operating system patches installed on a system.

### 11.2 Patch Data Class Attributes

#### 11.2.1 Patch Data Details

Name	Keyword	Description	Examples
Patch ID	ID	Unique identifier/number	
Summary of Patch	Summary	Quick 1 line summary	
Description (Verbose)	DescVerbose	Verbose description	
Date of Install	DateInstalled_String	Date of patch install	
Date of Release	DateReleased_String	Date of patch Release	
Search Keywords	Keywords	Keywords for searching	
Status	Status	Status of this patch	
URL	URL	Vendor's URL for info on this patch	
OS Name	OsName	Operating System Name Patch applies to	
OS Version	OsVersion	Operating System Version Patch applies to	
Application Architectures	AppArches	Application Architectures patch applies to	
README File	ReadmeFile	REAMDE File for Patch.	
Bug Fix IDs	BugFixIDs	List of Bugs fixed by this patch	
Patched Files	PatchedFiles	List of files patched by this patch	

Table 11.1: Patch Data Details

## Chapter 12

# Printer Data Class

### 12.1 Introduction

The Printer data class provides data about all the printers which are configured on a system.

### 12.2 Printer Data Class Attributes

#### 12.2.1 Printer Data Details

Name	Keyword	Description	Examples
Printer Queue Name	Queue	Name of printer queue	
Queue Aliases	Aliases	List of other names for queue	
Description	Desc	Description of Printer/Queue	
Vendor Name	Vendor	Vendor Name	
Model Name	Model	Model Name	
Driver Description	DriverDesc	Printer Driver Description	
Location	Location	Physical Location	
Protocol	Protocol	Protocol used to communicate with printer	
Device File	Device	Device File Path	
Server Hostname	Server	Server Hostname	
Server Canonical Hostname	ServerCan	Server Canonical Hostname	
Server IPv4 Address	ServerIPv4	Server IPv4 Address	
Server Port Number	ServerPort	Port number on server for this printer	
Remote Queue	RemoteQueue	Name of Printer Queue on Remote Server	
Printer Languages	Langs	List of Printer Languages Supported	
Spool Directory	SpoolDir	Spool Directory for this queue	
Max Job Size (KB)	MaxJobSize	Maximum Job Size Allowed (KB)	

Table 12.1: Printer Data Details

## Chapter 13

# Process Data Class

### 13.1 Introduction

The Process data class provides information on all system processes running on a system. This is the same data as provided by the `ps(1)` command on Unix/Linux systems or the Task Bar on Windows. The process data reported by SysInfo™ is provided in a single, consistent format and set of units across all platforms.

### 13.2 Process Data Class Attributes

#### 13.2.1 Process Data Details

Name	Keyword	Description	Examples
Process ID	ProcID	The unique process ID.	123
Command with arguments	Cmd	Full command with arguments.	<code>init -s</code>
Command Name	CmdName	Command name without arguments.	<code>init</code>
User Name	UserName	The username the process is running as.	<code>jsmith</code>
Terminal TTY	Tty	The TTY device process is running from.	<code>jsmith</code>
CPU Runtime	CpuTime_IntegerLarge	CPU Runtime formatted.	30
CPU Runtime	CpuTime_String	CPU Runtime formatted.	30
Percent of CPU	CpuPercent	Percent of CPU	5.1
Total Resident Memory Used (KB)	ResMem_IntegerLarge	Total Resident (RSS) Memory Used (KBYTES).	
Total Resident Memory Used (KB)	ResMem_String	Total Resident (RSS) Memory Used (KBYTES).	
Total Virtual Memory Used (KB)	VirtMem_IntegerLarge	Total Virtual Memory Used.	
Total Virtual Memory Used (KB)	VirtMem_String	Total Virtual Memory Used.	

Table 13.1: Process Data Details

## Chapter 14

# Service Data Class

### 14.1 Introduction

The Service data class provides information on services provided by the system. These are typically services started during system boot which provide services like remote login, login service, authentication, directory service, and network file access.

### 14.2 Service Data Class Attributes

#### 14.2.1 Service Data Details

Name	Keyword	Description	Examples
Name	Name	Name of service	rlogind
Identifier	ID	Identifier	A1234
Description	Descript	Description summary	Provides remote login service for rlogin command.
Description Verbose	DescVerbose	Verbose description	Provides remote login service for rlogin command. Allows authorized users to login remotely.
State	State	State of service	running, stopped, failed
Boot Time Enabled	BootEnabled	Enabled for start during system boot?	YES
RC Command	RcCmd	Boot time rc command to start/stop service	/etc/init.d/rlogind
Configuration File	ConfigFile	Configuration File	/etc/default/rlogind
Start Number	StartNum	Order number for startup sequence	95
Stop Number	StopNum	Order number for stop sequence	15
Boot Levels	BootLevels	List of system boot levels service is enabled for	1, 2, 3

Table 14.1: Service Data Details

## Chapter 15

# SiteInfo Data Class

### 15.1 Introduction

The SiteInfo data class provides arbitrary information that is user (site) defined. Users can set any data which can be represented in string form and have it reported when SysInfo™ is run with the **--class siteinfo** option. The common types of user defined data are site/company issued asset tracking tags, physical location of system, and department name system belongs too.

Each SiteInfo data consists of a keyword, a multiword description, and a value. Any data which can be described in string form may be used in any of these siteinfo entries.

#### 15.1.1 Setting SiteInfo Data

SiteInfo data may be set by manual creation of one or more `siteinfo` files or via the `mcsysinfocli(1)` command line.

##### 15.1.1.1 Setting SiteInfo Data - File Method

You manually create, edit, and modify the `siteinfo` files directly using an editor or your own tools. SysInfo™ searches for and reports data for each of the following `siteinfo` files:

- `$HOME/.mcsysinfo/siteinfo`
- `/etc/mcsysinfo/siteinfo`
- `$SysInfoDir/config/siteinfo`

where `$SysInfoDir` is the top level directory where SysInfo is installed (`/opt/sysinfo` is the default). The format of each `siteinfo` is defined in the `mcsysinfositeinfo(5)` reference manual.

For example, if you wanted to define a company issued asset tag value, you could run this command:

```
mkdir /etc/mcsysinfo
echo "AssetTag|Acme Asset Tag|A12345" > /etc/mcsysinfo/siteinfo
```

##### 15.1.1.2 Setting SiteInfo Data - Using the CLI

To set SiteInfo data using the SysInfo™ command line, use the following syntax for each SiteInfo data you wish to defined:

```
mcsysinfo \
  --siteinfo-varvar \
  --siteinfo-descdescription \
  --siteinfo-valuevalue \
  --siteinfo-scope usersystemsitem
```

The `--siteinfo-scope` argument specifies what `siteinfo` file the data is added to.

Here's an example of how to define a company issued asset tag value in the system's `siteinfo` file:

```
mcsysinfo --siteinfo-var AssetTag --siteinfo-desc 'Acme Asset Tag #' \  
--siteinfo-value A123456 --siteinfo-scope system
```

## 15.2 SiteInfo Data Class Attributes

### 15.2.1 SiteInfo Data Details

Name	Keyword	Description	Examples
Variable Name	Var	Variable (Keyword) Name	
Value	Value	Value of this entry	
Description	Desc	Description of entry	

Table 15.1: SiteInfo Data Details

## 15.3 More Information

More information is available in the [mcsysinfocli\(1\)](#) and [mcsysinfositeinfo\(5\)](#) reference manuals.

## Chapter 16

# Software Data Class

### 16.1 Introduction

The Software data class provides data about all the software installed on a system. All software which is registered with the operating system registration database is reported. Support for discovery of all other software is limited to specific software products. See the Release Notes for details of what products are supported for discovery.

### 16.2 SoftInfo Data Class Attributes

#### 16.2.1 SoftInfo Data Details

### 16.3 SoftFile Data Class Attributes

#### 16.3.1 SoftFile Data Details

---

Name	Keyword	Description	Examples
Entry Type	EntryType	Entry Type	
Name	Name	Name of package/product	
Software Unique Identification	ID	Unique value which identifies the software.	
Version	Version	Version	
Revision	Revision	Revision	
Description Summary	Description	Description Summary	
Description Verbose	DescVerbose	Description Verbose	
URL	URL	Product URL	
Help URL	URLhelp	Product Help URL	
Update URL	URLupdate	Product Update URL	
Entitlement (License) Required?	EntitlementRequired	Indicates wheather an entitlement (license) is required to legally use the software.	
Legal License Information	License	Information about how the software is licensed.	
License Key	LicenseKey	License Key (Serial Number) typically issued by software's manufacturer.	
License/Activiation Status	LicenseStatus	Current License Status.	
License Status (String)	LicenseStatusStr	License Status (String).	
Channel Name	ChannelName	Channel Name software targeted for.	
Channel Type	ChannelType	Channel Type software targeted for.	
Channel Type (String)	ChannelTypeStr	Channel Type software targeted for.	
Customer Type	CustomerType	Customer Type software targeted for.	
Customer Type (String)	CustomerTypeStr	Customer Type software targeted for.	
Copyright	Copyright	Copyright Message	
Category Type	CategoryType	Standardized category type. This type will be consistant across platforms.	
Category	Category	Category is a free-form string which is take directly from the OS/package. This field will vary widely from platform-to-platform.	
Sub Category	SubCategory	Sub Category entry belongs to. This field will vary widely from platform-to-platform.	
OS Name	OSname	Operating System Name	
OS Version	OSversion	Operating System Version	
Application Architecture	Arch	Application Architecture product runs on	
Instruction Set Architecture	ISArch	Instruction Set Architecture Product runs on	
Installation Date	InstDate	Installation Date	
Installation Date (Time)	InstDateTime	Installation Date (Time)	
Install Source	InstSource	Installation source	
Release Date	ReleaseDate	Release Date	
Release Date (Time)	ReleaseDateTime	Release Date (Time)	
Build Date	BuildDate	Build Date	
Build Date (Time)	BuildDateTime	Build Date (Time)	
Production Stamp	ProdStamp	Production Stamp	
Base Directory	BaseDir	Base Directory where installed	
Package Dependencies	PkgDeps	List of packages product depends upon	

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
File Type	Type	File Type	
Path	Path	File Path	
Link To	LinkTo	Link To This File	
File Size (bytes)	FileSize	File Size	
MD5 Checksum	MD5	MD5 Checksum	
Platform Checksum	CheckSum	Platform specific checksum	
Package Names	PkgNames	List of packages file is associated with	

Table 16.2: SoftFile Data Details

## Chapter 17

# Storage System Data Class

### 17.1 Introduction

The Storage System data class provides information on Network Attached Storage (NAS) and Storage Area Network (SAN) storage systems. Typically SysInfo™ reports the Storage Systems which are in use or known to the local host.

Since Storage System vendors provide their own unique, usually proprietary interfaces, SysInfo™ utilizes its own drivers to interface with each vendor's Storage System. Due to the lack of adoption of industry standards for Storage System interfaces, it's not possible for SysInfo™ to provide generic Storage System support. Only the following Storage Systems are supported:

- [Network Appliance Filers](#)
- [EMC Symmetrix and Clariion](#)

Please see the vendor specific sections for details of each system.

### 17.2 EMC Support

#### 17.2.1 Introduction

SysInfo™ supports all [EMC Symmetrix and Clariion](#) systems. SysInfo™ utilizes the EMC Solutions Enabler Developers Suite API Software Developer Kit (SDK) to discovery and obtain data. All supported EMC systems known to the local system's EMC software will be identified and reported by SysInfo™.

### 17.3 Network Appliance Filer Support

#### 17.3.1 Introduction

SysInfo™ supports all [Network Appliance Filers](#) which run the Network Appliance ONTAP operating system. See the [SysInfo™ Release Notes](#) for which specific ONTAP versions are supported.

#### 17.3.2 Discovery of Filers

Discovery of NetApp Filers is performed using the following procedure:

1. Obtain a list of all mounted filesystems on the local system.

2. For each NFS server found in the mount list we perform an SNMP probe of the system using a NetApp SNMP MIB.
3. For each NFS server found in the mount list we attempt to interrogate the system using *rsh(1c)*. Hence, the local system must be in the filer's `hosts` file for this to succeed. If rsh access succeeds, a number of ONTAP commands are used to supplement and enhance the data obtained via SNMP.

### 17.3.3 Data Acquisition

SysInfo™ uses SNMP queries as well as the ONTAP Command Line Interface via **rsh** to acquire most data on a Filer.

While SNMP queries of a Filer provides the bulk of the data we search for, critical components are only available via the Filer's Command Line Interface (CLI) which we access via **rsh**. Some of this data includes disk drive serial numbers, installed cards/controllers, and firmware information. Future versions of ONTAP will hopefully expose this type of detail via SNMP. Until then, **rsh** access will still be necessary for obtaining complete detailed data.

If you wish SysInfo™ to obtain full data from a Filer, you must place the hostname of the system SysInfo™ is run on in the Filer's `/vol/vol0/etc/hosts` file. Without this, **rsh** access will fail and thus, SysInfo™ will be unable to obtain all the data it's capable of. See your Network Appliance ONTAP documentation for instructions on how to setup **rsh** access.

## 17.4 StorSys Data Class Attributes

### 17.4.1 MicroCode Data Details

Name	Keyword	Description	Examples
Version (String)	uCodeVersionStr		
Version (Numeric)	uCodeVersionNum		
Version Date	uCodeDate		
Patch Level	uCodePatchLevel		
Patch Date	uCodePatchDate		

Table 17.1: MicroCode Data Details

### 17.4.2 FibrePort Data Details

Name	Keyword	Description	Examples
Type of Port	Type		
Controller's Port Number	CtlrPortNum		
Hub's Version	HubVersion		
Hub's ID	HubID		
Number of Ports on Hub	NumHubPorts		
Number of Ports on Switch	NumSwitchPorts		

Table 17.2: FibrePort Data Details

### 17.4.3 StorPort Data Details

### 17.4.4 StorFanBank Data Details

### 17.4.5 StorPwrSupply Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Controller Name	CtlrName		
Device Name on Host	DevName		
Status of Port	Status		
Status of Link	LinkStatus		
SCSI Width	ScsiWidth		
SCSI Method	ScsiMethod		
Device's Target ID	TargetID		
Device's Logical Unit Number	LUN		
Host's LUN for this Device	HostLUN		
Is Switch Present?	SwitchPresent		
Port Speed	Speed		
Port Number	PortNum		

Table 17.3: StorPort Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Identifier	Ident		
Status	Status		

Table 17.4: StorFanBank Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Identifier	Ident		
Type	Type		
Status	Status		

Table 17.5: StorPwrSupply Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Identifier	Ident		
Type	Type		

Table 17.6: StorEnc Data Details

### 17.4.6 StorEnc Data Details

### 17.4.7 StorMemBrd Data Details

Name	Keyword	Description	Examples
Slot Number	SlotNum		
Size	Size		

Table 17.7: StorMemBrd Data Details

### 17.4.8 StorRemData Data Details

Name	Keyword	Description	Examples
Vendor Term	VendorTerm	Vendor's Term for this data	
Remote Storage System Name	RemoteSysName		
Remote Storage System Name	RemoteSysDevName		
Group Number	GroupNum		
Remote Device Type	Type		
Status	Status		
Local Device Status	DevStatus		
Remote Device Status	RemDevStatus		
Link Status	LinkStatus		
Link Configuration	LinkConfig		
Remote Mode	Mode		
State of Pair	PairState		

Table 17.8: StorRemData Data Details

### 17.4.9 StorSnap Data Details

Name	Keyword	Description	Examples
Vendor's Term	VendorTerm	Vendor's Term for this data	
Serial # of Device	Serial		
System Device Name	SysDevName		
Serial # for Snap Device	SnapSerial		
Snap System Device Name	SnapSysDevName	System's device name for snap dev	
Group Name	GroupName		
Snap Group Name	SnapGroupName		
State of Pair	PairState		
Snap Device Status	SnapDevStatus		
Snap Mirror State	SnapMirrorState		

Table 17.9: StorSnap Data Details

### 17.4.10 StorPart Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Controller's Name	CtlrName		
Vendor Term	VendorTerm		
Controller's Logical Volume Number	CtlrVolNum		
Device Partition Number	PartNum		
Type	Type		
Status	Status		
RAID Group Number	RaidGroupNum		
Disk's SCSI ID	DiskScsiID		
Disk's Model Name	DiskModel		
Disk's Model Revision	DiskModelRev		
Disk's Serial Number	DiskSerial		
Disk's Cache Size	DiskCacheSize		
Disk's Cache Size (KB)	DiskCacheSizeNum		
Disk's Total Capacity	DiskCapacity		
Disk's Total Capacity (MB)	DiskCapacityNum		
Disk's RPM	DiskRpm		
Disk's Actual Capacity	DiskActCapacity		
Disk's Actual Capacity (MB)	DiskActCapacityNum		
Capacity	Capacity		
Capacity (MB)	CapacityNum		

Table 17.10: StorPart Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Vendor's Term for this data	VendorTerm		
Capacity	Capacity		
Capacity (MB)	CapacityNum		
Number of Member Devices	NumMembers		
System Device Name List	SysDevList		
Status of Meta Device	Status		
Configuration	Config		
Size of Stripe (KB)	StripeSize		

Table 17.11: StorMeta Data Details

**17.4.11 StorMeta Data Details****17.4.12 StorGroup Data Details**

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Names of Storage Controllers in Group	CtlrNames		
Vendor's Term for this data	VendorTerm		
Group Name	Name		
Unique ID	UID		
Host's Names of StorDevice's in Group	HostDevNames		
System's Names of StorDevice's in Group	SysDevNames		
RAID Type	RaidType		
Capacity	Capacity		
Capacity (MB)	CapacityNum		
Used Capacity	UsedCapacity		
Used Capacity (MB)	UsedCapacityNum		
Free Capacity	FreeCapacity		
Free Capacity (MB)	FreeCapacityNum		

Table 17.12: StorGroup Data Details

**17.4.13 StorHostDev Data Details**

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Host's Name for Device	Name		
Host Path Name	Path		
SCSI Target ID	TargetID		
SCSI Logical Unit Number	LUN		

Table 17.13: StorHostDev Data Details

**17.4.14 StorDevice Data Details****17.4.15 StorDisk Data Details****17.4.16 StorCtlr Data Details****17.4.17 StorLicense Data Details****17.4.18 StorSys Data Details**

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Vendor's Term for this data	VendorTerm		
Name of Storage Controller	CtlrName		
Physical Device Name	Name		
System's Name for Device	SysDevName		
Logical Device Name	LogDevName		
Type	Type		
Group Name	GroupName		
Vendor	Vendor		
Model	Model		
Revision	Revision		
Serial Number	Serial		
User Label	UserLabel		
Device Block Size (bytes)	BlockSize		
Stripe Size (KB)	StripeSize		
Capacity	Capacity		
Capacity (MB)	CapacityNum		
Number of Cylinders	Cylinders		
Capacity Used	Used		
Used Capacity (MB)	UsedNum		
SCSI Target ID	TargetID		
SCSI LUN	LUN		
SCSI Width	ScsiWidth		
SCSI Method	ScsiMethod		
Disk Names	DiskNames	Names of disks used for this device	
Status	Status		
Host Device List	HostDevices		
Remote Devices List	RemData		
Snap List Info	Snap		
Alternate Port Info	AltPort		
Partition Info	Parts		
Meta Info	Meta		

Table 17.14: StorDevice Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Name of Controller	CtlrName		
Enclosure Number	Enclosure		
Bay Number in Enclosure	Bay		
Name	Name		
Vendor	Vendor		
Model	Model		
Revision	Revision		
Serial Number	Serial		
Type	Type		
Capacity	Capacity		
Capacity (MB)	CapacityNum		
Used Amount	Used		
Used Amount (MB)	UsedNum		
Block Size (bytes)	BlockSize		
Usage	Usage		
Status	Status		
Bus Number	Bus		
SCSI Target ID	TargetID		
SCSI Logical Unit Number	LUN		
Name of Meta Info	MetaName		

Table 17.15: StorDisk Data Details

Name	Keyword	Description	Examples
Vendor's Term for this data	VendorTerm		
Name	Name		
Type	Type		
Vendor	Vendor		
Model	Model		
Model Description	ModelDesc		
Part Number	Part		
Revision	Revision		
Serial Number	Serial		
Location in System	Location		
Micro Code Version	MicroCodeVer		
Unit Number	Unit		
Slot Number	Slot		
Number of Ports	NumPorts		
Number of Partitions Serviced	NumParts		
Status	Status		
SCSI Width	ScsiWidth		
Total Physical Memory (MB)	TotalMem		
Total Physical Memory	TotalMemStr		
System Buffer Reserved Memory Size	SysBufferSize		
System Buffer Reserved Memory Size (MB)	SysBufferSizeNum		
Read Cache Size	ReadCacheSize		
Read Cache Size (MB)	ReadCacheSizeNum		
Write Cache Size	WriteCacheSize		
Write Cache Size (MB)	WriteCacheSizeNum		
Read Cache Status	ReadCacheStatus		
Write Cache Status	WriteCacheStatus		
Free Memory Size	FreeMemSize		
Free Memory Size (MB)	FreeMemSizeNum		
Port Info	Ports		
Fibre Ports Info	FibrePorts		
Devices Attached	Devices		
Physical Disks Attached	Disks		
Enclosures Attached to this Controller	Enclosures		
Device Type	DevType		
Device Class Type	DevClassType		
Attached DevInfo Children	DevInfoChildren		

Table 17.16: StorCtrl Data Details

Name	Keyword	Description	Examples
Name	Name		
Description	Desc		
License Enabled/Available?	Enabled		

Table 17.17: StorLicense Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Name	Name		
Vendor	Vendor		
Model	Model		
Model Description	ModelDesc		
Serial Number	Serial		
System ID	Id		
Name of OS	OsName		
OS Version	OsVersion		
OS Release	OsRelease		
Size of Cache (MB)	CacheSize		
Size of Cache	CacheSizeStr		
Total Capacity	Capacity		
Total Capacity (MB)	CapacityNum		
Number of Physical Disks	NumDisks		
Number of Spare Disks	NumDisksSpare		
Number of Active Disks	NumDisksActive		
Number of Failed Disks	NumDisksFailed		
Number of Disks not Configured	NumUnConfigDisks		
Number of System Devices	NumSysDevs		
Number of Host Physical Devices	NumPhysDevs		
Physical Device List	PhysDevList		
Time of System Power On (seconds)	PwrOnTime		
Time of System Power On	PwrOnTimeStr		
Time of last Initial Primary Load	LastIplTime		
Time of last Initial Primary Load	LastIplTimeStr		
Time of last Fast Initial Primary Load	LastFastIplTime		
Time of last Fast Initial Primary Load	LastFastIplTimeStr		
System Up Time (seconds)	Uptime		
System Up Time (days)	UptimeDays		
System Micro Code	MicroCode		
Controllers	Controllers		
Memory Boards	MemBoards		
Enclosures	Enclosures		
Physical Disks	Disks		
Storage Groups	Groups		
Attached DevInfo Children	DevInfoChildren		
List of Cluster Partner Hostnames	ClusterPartners		
Devices Attached to System	Devices		
Licensed Features	Licenses		

Table 17.18: StorSys Data Details

## Chapter 18

# StorVol Data Class

### 18.1 Introduction

The StorVol data class provides data about all storage volumes found on a system. These are typically managed through some type of logical volume manager.

### 18.2 StorVol Data Class Attributes

#### 18.2.1 StorVolSw Data Details

Name	Keyword	Description	Examples
Volume Manager Vendor	VmVendor		
Volume Manager Product	VmProduct		
VolMgr Software Version	VmVersion		

Table 18.1: StorVolSw Data Details

#### 18.2.2 StorVolDisk Data Details

Name	Keyword	Description	Examples
Vol Mgr Name for Disk	Name		
System Name for Disk	SysName		
Unique Identifier (Serial Number)	ID		
Status of Disk	Status		
State of Disk	State		
Capacity of Disk	Capacity		
Capacity of Disk (MB)	CapacityNum		
Number of Blocks	BlkCount		
Vol Mgr Software	VmSw		
Volume Name	VolName	Name of Volume this disk is a member of.	

Table 18.2: StorVolDisk Data Details

### 18.2.3 StorVolPart Data Details

Name	Keyword	Description	Examples
Vendor's term for this data	VendorTerm		
Name	Name		
Unique Identifier (Serial Number)	ID		
Device Path	DevPath		
Used For	Usage		
Name of Parent Plex	PlexName		
OS Disk Name	OsDiskName		
Vol Mgr Name for Disk	VmDiskName		
OS Starting Block on OS Disk	OsStartBlk		
Vol Mgr Starting Block on Disk	VmStartBlk		
Number of Blocks Partition Assigned	BlkCount		
Status	Status		
State	State		
Capacity of Partition	Capacity		
Capacity of Partition (MB)	CapacityNum		
Amount Used	AmtUsed		
Amount Used (MB)	AmtUsedNum		
Amount Available	AmtAvail		
Amount Available (MB)	AmtAvailNum		
Device Paths	DevPaths		

Table 18.3: StorVolPart Data Details

### 18.2.4 StorVolPlex Data Details

### 18.2.5 StorVolume Data Details

### 18.2.6 StorVolGroup Data Details

### 18.2.7 StorVol Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Vendor's term for this data	VendorTerm		
Name	Name		
Unique Identifier (Serial Number)	ID		
Device Path	DevPath		
Volume Name of Parent	VolName		
Status	Status		
State	State		
Type (Layout) of Plex	Type		
Capacity	Capacity		
Capacity (MB)	CapacityNum		
Partitions Which Make Up Plex	Parts		

Table 18.4: StorVolPlex Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Vendor's term for this data	VendorTerm		
Name of Volume	Name		
Unique Identifier (Serial Number)	ID		
Parent's Volume Group Name	VolGrp		
RAID Level	RaidLevel		
Status	Status		
State	State		
Used For	Usage		
Capacity	Capacity		
Capacity (MB)	CapacityNum		
Amount Used	AmtUsed		
Amount Used (MB)	AmtUsedNum		
Amount Available	AmtAvail		
Amount Available (MB)	AmtAvailNum		
Character Device Path	DevPath		
Raw (block) Device Path	RawDevPath		
Volume Manager Software	VmSw		
Storage Plexes	Plexes		
Partitions which make up plex	Parts		

Table 18.5: StorVolume Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
Vendor's term for this data	VendorTerm		
Name of Disk Group	Name		
Unique Identifier (Serial Number)	ID		
State	State		
Status	Status		
Character Device Path	DevPath		
Capacity	Capacity		
Capacity (MB)	CapacityNum		
Amount Used	AmtUsed		
Amount Used (MB)	AmtUsedNum		
Amount Available	AmtAvail		
Amount Available (MB)	AmtAvailNum		
Volume Manager Software	VmSw		
Child Storage Volumes	Volumes		
Disks Belonging to this Group	Disks		
Partitions Belonging to this Group	Parts		
File Systems using this Group	FileSystems		

Table 18.6: StorVolGroup Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
All Volume Groups	VolGroups		
All Storage Volumes	Volumes		
All Disks Used by Volume Managers	Disks		

Table 18.7: StorVol Data Details

## Chapter 19

# SysConf Data Class

### 19.1 Introduction

The SysConf data class provides data about system configuration parameters. These parameters provide details about the operating system's capabilities and compliance with standards such as POSIX.

### 19.2 SysConf Data Class Attributes

#### 19.2.1 SysConf Data Details

Name	Keyword	Description	Examples
Variable Name	Name		
Variable Description	Description		
Variable Type	Type		
Variable Value	Value_String		
Variable Value	Value_Boolean		
Variable Value	Value_IntegerLarge		

Table 19.1: SysConf Data Details

## Chapter 20

# Virtual Machine Guests Data Class

### 20.1 Introduction

The Virtual Machine Guests (vmguest) data class provides information on each Virtual Machine Guest OS configured by a VM product. Each VM Guest is "hosted" by the VM software product and provides a virtual interface to the top-level OS where the VM software is installed. The vmguest SysInfo™ class lists each Guest OS and its attributes.

SysInfo™ must be run under the real (non-hosted) OS where the VM product itself is installed.

See the Release Notes for which VM products are supported in this release.

### 20.2 VmGuest Data Class Attributes

#### 20.2.1 VmGuest Data Details

<b>Name</b>	<b>Keyword</b>	<b>Description</b>	<b>Examples</b>
VM Software Name	VmSoftwareName	Name of VM Software	VMware ESX
VM Software Version	VmSoftwareVersion	Version of VM Software	3.5
VM Software Vendor	VmSoftwareVendor	Vendor of VM Software	VMware
VM Network Interface Name	VmNetIfName	VM Network Interface Name	Bridged0
VM Guest Path	VmGuestPath	VM Guest Path	/vol/vm/guest1
Name	Name	Name of VM	buzzvm1
Label	Label	Customer label for VM	QA Test VM 1
UUID	UUID	Unique ID	124-JDKJF-334I
Local ID	LID	Local ID	1
State	State	State of VM	running
Boot Type	BootType	Boot Type	manual
Guest OS Name	GuestOsName	Name of OS installed in Guest VM	Solaris
Guest OS Version	GuestOsVersion	Version of OS installed in Guest VM	10
Guest Serial	GuestSerial	Guest Serial Number of VM	V9834856
Guest Memory Size	GuestMemorySize	Memory Size assigned to Guest	512 MB
Guest CPU Count	GuestCpuCount	Count of CPUs assigned to Guest	2
Guest IP Address	GuestIpAddr	Guest IP Address	10.1.2.55
Guest IP Hostname	GuestIpHostname	Guest IP Hostname	www.acme.com
Guest MAC Address	GuestMacAddr	Guest MAC Address	00:0B:DC:12:34

Table 20.1: VmGuest Data Details

## Chapter 21

# Virtual Machine Host Data Class

### 21.1 Introduction

The Virtual Machine Host (vmhost) data class provides information on whether the OS under which SysInfo™ is run is running under a Virtual Machine hosting product. If VM Host detection is available for the platform SysInfo™ is run on, it will indicate whether the OS is hosted by a VM product or not. If it is hosted by a VM product, then details of the VM product will be provided including the VM software and vendor.

See the Release Notes for which VM products are supported in this release.

### 21.2 Data Details

The following table provides details on what data is provided by this class. Not all data types are provided on all platforms.

Name	Keyword	Description	Examples
Running in a VM Host	invmhost	If SysInfo™ is run inside of a VM Host, the value of this field is YES. If SysInfo™ supports detection of VM Hosts on the platform it's run on and it does not detect it's running in a VM Host, the value reported is NO.	YES
VM Host Name	name	The name of the VM Host as defined by the VM software product. This is not necessarily the same as the hostname configured in the OS.	buzzvm1
VM Software Name	softwarename	The name of the software which is providing the VM Host environment.	VMware Workstation.
VM Software Version	version	The version of VM Software which is providing the VM Host environment.	1.0
VM Software Vendor	vendormame	The vendor of the VM Software which is providing the VM Host environment.	VMware

Table 21.1: VM Host Data Details

## Chapter 22

# Software Discovery Language™ (SDL)

### 22.1 Introduction

The Software Discovery Language™ (SDL) is an interpreted scripting language implemented in SysInfo™ to discover software products. SysInfo™ uses SDL scripts to discovery and collect SysInfo™ Software class data for software products which do not register themselves with the native operating system software registry (Windows Registry, RPM, swinstall, pkgs, etc).

SDL scripts are interpreted at runtime and thus are able to be added, deleted, and modified to an existing SysInfo™ binary installation. Customers can choose to add and maintain their own set of SDL scripts to support their specific environment and applications.

The primary work of an SDL script is to discovery it's intended software product. The script can use many of the SDL standard functions or the programmer can write their own custom functions built on top of the SDL standard functions much like in a shell script.

When an SDL script discovers data about it's intended software product, it uses a series of SDL standard functions to record the relevent data. When all of the relevent data is discovered, the SDL script calls the `SoftInfoAdd()` to add the discovered product to the internal software list. When all Software discovery, both SDL and the discovery built into SysInfo™, is complete, the discovered products are reported.

By convention each SDL script should discover and report on a single software product. While there is no SDL syntax limitation to supporting multiple products in a single SDL script, this is not yet a supported convention.

#### 22.1.1 Syntax

An SDL script is a text file which has a syntax that is a blend of C and UNIX Bourne Shell syntax.

An SDL script contains a series of statements. Statements can contain variables, custom functions, and calls to custom or standard SDL functions. Statements are terminated by the ';' (semi-colon) character. Statements can be grouped together inside of '{}' (curly braces).

The "///" (two forward slashes) sequence denotes a comment. Anything appearing after this sequence on the same line is ignored.

A fairly simple SDL script with builtin product definetions is:

---

**Example 22.1** Simple SDL Script example

```
$PRODUCT_NAME = "splicer"; // Required
$VENDOR_NAME = "Acme Inc"; // Required
$SoftInfo = SoftInfoCreate($PRODUCT_NAME, "1.0");
SoftInfoAdd($SoftInfo);
Exit(0);
```

---

In this example, a product called "splicer" version 1.0 is created and added to the list of products to report. Note that the call to `SoftInfoAdd()` is always required in order to report on the discovered product. If you do not call the `SoftInfoAdd()` function your product will not be reported by SysInfo™.

---

### 22.1.1.1 Data Types

SDL supports the following data types:

Type	Description
Boolean_t	Value can either be True or False.
String_t	A string value.
Array_t	An array of any other supported data type.
Number_t	A numeric integer or floating point value.
XmlHandle_t	A handle used to manipulate XML data.
SoftInfo_t	A handle used to manipulate discovered software data.
DataTree_t	A generic list of data in a hierarchy tree structure.

Table 22.1: SDL Data Types

### 22.1.1.2 Variables

Variables have local function scope. A variable declared in the global scope is available through-out the entire script. A variable declared inside a function has a scope limited to that function.

Variable names start with the traditional '\$' and consist of letters, digits, and '\_' (underscore) characters. Variables are set using the '=' (equal) char and can be set to literal string or numeric values, as well as to other variables or data returned by functions.

#### Example 22.2 Variable Names

```
$String1 = "fun time";
$Num = 42;
$MyStrArray = StrSplit(":", "/usr/bin:/bin:/etc");
```

### 22.1.1.3 Logical Constructs

SDL supports the `if`, `if else`, `for`, `foreach`, and `while`, logical constructs.

#### 22.1.1.3.1 IF Statements

IF statements are logical conditional statements that are very similar to C statements. The syntax supports both a simple "if" conditional as well as an "if else" syntax.

#### Example 22.3 IF Simple

```
if (...) {
    [statements]
}
```

#### Example 22.4 IF Else

```
if (...) {
    [statements]
} else {
    [statements]
}
```

### 22.1.1.3.2 FOR Statements

FOR statements are logical conditional loop statements that are very similar to C statements. The syntax is as follows:

---

**Example 22.5** FOR Syntax

---

```
for ([first time statements]; [while true]; [loop statements]) {
    [statements]
}
```

---

### 22.1.1.3.3 FOREACH Statements

FOREACH statements are logical conditional loop statements that are very similar to the Perl "foreach" loop. The FOREACH statement takes a single value which must be an Array\_t value of any supported type. Each value in the Array\_t is iterated until the end of the array is reached or a "Break" or "Return" statement is reached. Each time through the loop the first argument is set as a variable specifying the value of the iterated array. The syntax is as follows:

---

**Example 22.6** FOREACH Syntax

---

```
foreach $NewVar ($ArrayValue) {
    [statements]
}
```

---

### 22.1.1.3.4 WHILE Statements

WHILE statements are logical conditional loop statements that are very similar to C while loops. The statements inside the while loop are executed so long as the logical loop test evaluates to true. The syntax is as follows:

---

**Example 22.7** WHILE Syntax

---

```
while ([true statements]) {
    [statements]
}
```

---

### 22.1.1.3.5 Special Loop Instructions

There are a number of special instructions which apply to the loop constructs `for`, `foreach`, and `while`.

When the `break` instruction is encountered inside of a loop, the execution of that loop immediately stops and continues after the loop construct.

When the `continue` instruction is encountered inside of a loop, the execution of the loop returns to the "top" of the loop instead of continuing to the next logical statement inside of the loop.

## 22.2 Standard Functions

The following table specifies all the standard functions which are built into SDL.

The `SoftInfoSet()` takes a `String_t` key value which indicates what type of data to set into the `SoftInfo_t` handle. The following table describes the valid values for this key:

---

Syntax	Return Value	Description
CmdClose(Cmd_t \$Handle)		Close a Cmd_t handle.
ArrayLen(Array_t \$Array)	Number_t	Return the number of members in array \$Array.
CmdOpen(String_t \$Command, String_t \$Args, String \$Opts)	Cmd_t	Open a new command. \$Command should be the path to the command to run and \$Args are the arguments to pass to \$Command. \$Opts is a comma separated list of options understood by CmdOpen() itself. The only currently supported option is "withstderr" which includes standard error output in the input stream read by CmdReadLine().
CmdReadLine(Cmd_t \$Handle)	String_t	Read one line of output from the command handle \$Handle.
DataTreeGetIndex(DataTree_t \$Data, Number_t \$Index)	DataTree_t	Get the DataTree_t at offset \$Index from \$Data.
DataTreeGetTree(DataTree_t \$Data, String_t \$Path, String_t \$Attr, String_t \$Opts)	DataTree_t	Search through data tree \$Data for attribute matching \$Path and optionally \$Attr. \$Opts is a comma separated list of options. Valid options are "CHILDREN" (return Children of match, not the top node itself).
DataTreeGetValue(DataTree_t \$Data, String_t \$Attr)	String_t	Search through data tree \$Data for attribute matching \$Attr.
DataTreeGetKeys(DataTree_t \$Data)	Array_t	Return an Array of Key values for each member of \$Data and it's peer nodes. Does not descend.
DataTreeGetValues(DataTree_t \$Data)	Array_t	Return an Array of Values for each member of \$Data and it's peer nodes. Does not descend.
Debug(String_t \$Msg)		If SDL debugging is enabled, then output \$Msg.
Error(String_t \$Msg)		Print an error message.
Exit(Number_t \$Num)		Exit the current SDL script and return \$Num to the invoker as the return value for this script.
FileClose(File_t \$Handle)		Close a File_t handle.
FileOpen(String_t \$Path, String_t \$Mode, String \$Opts)	File_t	Open a new file handle. \$Path should be the path of the file to be opened. \$Mode should be "READ" to open the file in read mode. \$Opts is not used at this time.
FileReadLine(File_t \$Handle)	String_t	Read one line of output from the file handle \$Handle.
FileSize(String_t \$File)	Number_t	Get the size of file \$File in bytes.
GetEnv(String_t \$Var)	String_t	Get an environment variable \$Var and return it's value.
GetGrEnt(String_t \$Name)	String_t	Look for a system group called \$Name. If found returns an string array consisting of: [0] group name, [1] group password, [2], GID, [3] list of members seperated by comma.
GetPwEnt(String_t \$Name)	String_t	Look for a system user (defined in /etc/passwd or similar location) called \$Name. If found returns an string array consisting of: [0] user name, [1] password, [2], UID, [3] GID, [4] full name (GECOS), [5] home directory, [6] shell.
PathBaseName(String_t \$Path)	String_t	Get the base (file) name of \$Path.
PathDirName(String_t \$Path)	String_t	Get the directory name of \$Path.
PathExists(String_t \$Path, String_t \$Type)	Boolean_t	If \$Path exists and is of file type \$Type (optional), return True.

<b>Key</b>	<b>Description</b>
EntryType	The type of entry either "pkg" or "product".
Name	The short, unique name of the product.
Version	The version of the product.
Revision	The revision of the product.
Desc	The short, one-line summary description.
DescVerbose	The verbose, possibly multi-line description.
URL	Product URL.
URLhelp	The product help URL.
URLupdate	The product update URL.
License	The license of the product.
Copyright	The product copyright notice.
Category	The category the product belongs to.
SubCategory	The sub-category the product belongs to.
OSname	The OS name the product requires.
OSversion	The OS version the product requires.
Arch	The hardware architecture the product requires.
ISArch	The instruction set architecture the product requires.
InstDate	Installation date.
BuildDate	Build date.
ProdStamp	Production stamp.
BaseDir	Base directory where product is installed.
PkgDeps	Package dependencies.
DiskUsage	The disk usage the product consumes in bytes.
VendorName	Name of product vendor.
VendorEmail	Email of product vendor.
VendorPhone	Phone of product vendor.
VendorStock	Stock number from product vendor.
Child	Name of child SoftInfo_t nodes.

Table 22.3: SoftInfoSet() Key Definitions

## 22.3 Custom Functions

Each SDL script can contain custom functions much like UNIX shell scripts and C functions. A function can appear in any part of an SDL script. You define a function by specifying:

---

### Example 22.8 Function Definition

---

```
Name (Parameters) {
    [statements]
}
```

---

For example, here is a function definition which accepts two arguments and prints each in a separate debug call:

---

### Example 22.9 Simple Function Definition

---

```
PrintMyArgs(String_t $Str1, String_t $Str2) {
    Debug("Arg1 is $Arg1");
    Debug("Arg2 is $Arg2");
}
```

---

## 22.4 Basic Example Script

The following is a basic example of an SDL script to discover the Apache HTTPD web server.

---

### Example 22.10 Basic Example Script

---

```
//
// Apache HTTPD Server
//

$PRODUCT_NAME = "apachehttpd"; // Required
$VENDOR_NAME = "Apache Foundation"; // Required

Debug("Looking for Apache httpd web server");

$HttpdPaths = "/usr/apache/bin/httpd:/sbin/httpd:/usr/sbin/httpd:/usr/local/sbin/httpd:/usr ←
    /local/bin/httpd:/bin/httpd:/usr/bin/httpd";

$Paths = StrSplit(":", $HttpdPaths);

foreach $path ($Paths) {
    Debug("Looking for $path");
    if (PathExists($path, "FILE")) {
        if (HttpdProbe($path)) {
            break;
        }
    }
}

//
// Probe using $HttpdPath as path to httpd
//
HttpdProbe(String_t $HttpdPath)
{
    Debug("Probe using httpd ${HttpdPath}");

    $CmdHandle = CmdOpen($HttpdPath, "-v");
```

---

```

while ($Line = CmdReadLine($CmdHandle)) {
Debug("READ: ${Line}");

if (StrMatchN($Line, "Server version:", 15)) {
    Debug("Found version line.");
    if (StrRegExMatch($Line, "Apache")) {
        Debug("Matched line ${Line}");
        // Line looks like: Server version: Apache/2.2.3
        $Version = StrEdit(".*/", $Line, "", NULL);
    }
} else {
    Debug("Check for built in ${Line}");
    if (StrMatchN($Line, "Server built:", 13)) {
        Debug("Found built info.");
        $BuildDate = StrEdit("Server built:", $Line, "", NULL);
        $BuildDate = StrEdit("^.[ ]+", $BuildDate, "", NULL);
    }
}
}
CmdClose($CmdHandle);

if (!$Version) {
Debug("No Version was found - skipping $HttpdPath");
Return(0);
}

$SoftInfo = SoftInfoCreate($PRODUCT_NAME, $Version);
SoftInfoSet($SoftInfo, "VendorName", $VENDOR_NAME);
SoftInfoSet($SoftInfo, "Desc", "HTTP Web Server");
SoftInfoSet($SoftInfo, "VendorName", $VENDOR_NAME);
if ($BuildDate) {
SoftInfoSet($SoftInfo, "BuildDate", $BuildDate);
}
if ($dir = PathDirName($HttpdPath)) {
SoftInfoSet($SoftInfo, "BaseDir", $dir);
}
if ($size = FileSize($HttpdPath)) {
SoftInfoSet($SoftInfo, "DiskUsage", $size);
}

SoftInfoAdd($SoftInfo); // Add to tree

Return(1);
}

```

## 22.5 SDL Script Detection

When SysInfo™ is run, it looks for SDL scripts automatically under the `$prefix/lib/sdl` directory tree. All files ending with ".sdl" are run.

You can have SysInfo™ search your own set of SDL scripts by specifying the `--sdlpath $path` command line option or by setting the `MCSYSINFO_SDL_PATH` environment variable to a comma separated list of directories to search for ".sdl" files.

## 22.6 Testing and Debugging

A useful means of testing and debugging SDL scripts is to use the `sdl` standalone command. This command, which is found in the same location as the `mcsysinfo`, is an easy means of running and debugging one or many SDL scripts.

If you want to run a single SDL script, you simple run:

```
sdl file.sdl
```

If you want to enable debugging output add "-d" options:

```
sdl -d file.sdl
```

Each time "-d" is specified the amount of debugging output increases:

```
sdl -d -d file.sdl
```