SIB '04 CI info functions

By Hari, Prathibha and Raghunand, CSY

Enhancements summary:

- The DEVINFO evaluator function returns information about the logical devices attached to the system. It is a CI interface to AIFDEVICEGET.
- The SPOOLINFO evaluator function returns information about input and output Spoolfiles. It is a CI Interface to AIFSPFGET.
- The VOLINFO evaluator function returns information about volumes/volume classes and volume sets. VOLINFO is a CI interface to HPVOLINFO.

DEVINFO

The DEVINFO CI function returns information about devices in a system. It is a CI interface to the existing AIFDEVICEGET intrinsic. Most of the items are serviced through the AIFDEVICEGET API. However, there are a few items that use the AIFSPPGET API (spooler items) and internal OS routines (device list items). Depending on the information desired, DEVINFO returns a string, Boolean, or an integer.

Syntax

DEVINFO (DeviceID, item[,status])

Parameters

DEVINFO accepts three parameters :

- A device ID, the logical device number of the device about which information is returned.
- An item indicating what information is desired.
- An optional status parameter to return the status of the function call.

Device ID

The device ID identifies a device via its LDEV number. For ease of use, the LDEV number can be a numeric string or an integer. There are cases (calls to DEVINFO that return a list of LDEV numbers) when this parameter is not required. In such cases, the device ID can be set to the empty string (""), the string "0" or an integer 0.

Item

This parameter is a key word that indicates what information about an LDEV is desired. It is a string parameter. A list of supported items and their description can be found in a separate section below. The list includes the item name, its aliases, return type of DEVINFO and a description of the item.

Status

Status is an optional 32 bit integer parameter. It is the CI variable name

where a DEVINFO error number can be returned. Zero indicates success. Non-zero indicates a failure. Error values returned in 'status' are the same as the CIERROR value when an error occurs. If the 'status' parm is passed, the CI will not set CIERROR and will not report an error message. If 'status' is not specified the CI performs its normal error handling.

Description of DEVINFO Items

The following table summarizes the items of the DEVINFO CI function. The description includes the item string, one or more aliases, the return type, and a brief description of the item.

Item Name	Type	Description
AllPrinterDevList	string	List of printer LDEVs separated by a space Format:"Total/LDEV1 LDEV2 ... LDEVN" *
AllTermDevList	string	List of terminal LDEVs separated by a space Format:"Total/LDEV1 LDEV2 ... LDEVN" *
AltOwnerPin	int	Device's alternate owner PIN
AutoReply	bool	True means device automatically replies to tape requests
Available Up	bool	True if the device is available to the System
BlockModeType	int	7=both line and DTC style block mode, 15=PAD terminal supporting page block mode
BOT	bool	True means tape device is at beginning of tape mark
CommDevList	string	List of data comm devices. Format: "Total/LDEV1 LDEV2 ... LDEVN" *
DataAccepting	bool	True if device accepts DATA logons
DevClass	string	IO Device Class: "NOT CONFIG","DISK","TAPE", "TERM","CIPER PRINTER","PRINTER","SPOOLED", "DATA COMM","DS TERM","DS PRINTER",
DevName	string	Device name as used in the "adev" SYSGEN command
DevSubType	int	Device subtype. See IODEFAULT.PUB.SYS
DevType	string	Device type: "DISK","TERM","TAPE","PRINTER"
DiscDevList DiskDevList	string	List of disk LDEVs separated by a space. Format:"Total/LDEV1 LDEV2 ... LDEVN" *
DiskSizeMB DiscSizeMB	int	Disk size in megabytes
DiskSizePages DiscSizePages	int	Disk size in 2k pages (8 sectors per physical page)
Down	bool	True if device is unavailable to the system (not up) and no down pending
DownPend	bool	True if device has a down request pending
DSPrinterDevList	string	List of DS printer LDEVs separated by a space. Format:"Total/LDEV1 LDEV2 ... LDEVN" *
DSTermDevList	string	List of DS terminal LDEVs separated by a space. Format:"Total/LDEV1 LDEV2 ... LDEVN" *
Duplicative	bool	True all input operations are automatically Echoed
EOT	bool	True if tape is positioned at the end of Tape
Exists Exist	bool	True if the device is configured
FileName	string	Formal device file designator, eg $STDIN. Will be blanks for non-terminal devices
FormsMounted	bool	True if special forms are mounted for this Device
HeaderDisabled	bool	True if the header pages are suppressed in Printing
IntDevClass	int	0=not config,1=disk,2=tape,3=terminal, 4=CIPER printer, 5=printer(non-ciper), 6=spooled, 7=data comm, 8=DS teminal, 9=DS printer, 10=user-defined
IntDevType	int	0-7=disk, 16=terminal, 24=tape, 32=printer
Interactive	bool	True if device requires human intervention for all input

		operations
IntSpoolerState	int	Spooler process state for LDEV: 0=initializing, 1=release, 2=start, 3=stop, 4=stop pending, 5=suspend, 6=suspend pending, 7=resume, 8=active, 9=shut pending, 10=idle
IntSubClass	int	Device subclass: 0=unknown,1=connected to TMUX, 2=terminal,3=printer,4=virtual terminal,5=virtual printer,6=PAD terminal,7=PAD printer,12=null terminal,13=DHCF terminal,14=pseudo terminal, 15=pseudo null terminal. For tape: 1=tape. For disk: 1=disk
IntTapeError	int	Fatal tape error value: 0=No fatal error, 1= tape runaway, 2=multiple tracks in error, 3= timing error, 4=command reject, 5=unit failure, 6=data parity error, 7=command parity error
IntTermType	int	Terminal Type: 10=TT10, 18=TT18, 21=TT21, 22=TT22, 24=TT24, 26=TT26
InvalidLogonCnt	int	Invalid logon counts for interactive device **
JobAccepting	bool	True if device accepts HELLO or JOB logons
LineSpeed	int	Terminal line speed setting: 30, 120, 480, 960, 19200, 38400
NotConfigDevList	string	List of unconfigured LDEVs separated by a space. Format:"Total/LDEV1 LDEV2 ... LDEVN" *
Opened	bool	True means device exists and is opened/owned
OwnerJobNum	string	Device's owner's job/session number, "J|Snnn"
OwnerPIN	int	PIN for device's owner process, which is a JSMAIN process
Ownership	string	Device's ownership state: "OWNED"=owned by a process, "OS"=temporarily owned by the operating system, ""=not owned by a process
PrinterDevList	string	List of printers which are not DS nor serial Format:"Total/LDEV1 LDEV2 ... LDEVN" *
ReadTimeOut	int	Number of seconds for next (timed) read, 0 or negative means timeout is disabled
RecWidth	int	Device record width in bytes
SecurityDown	bool	True if the device is downed by HP Security Monitor **
SerialPrinterDevList	str	List of serial printers separated by a space.Format:"Total/LDEV1 LDEV2 ... LDEVN" *
SpoolerDevList	string	List of spooler device separated by a space. Format:"Total/LDEV1 LDEV2 ... LDEVN" *
SpoolerOutfence	int	Current outfence for the output spooler device, 0..14
SpoolerState	string	Spooler process state for LDEV: "INIT", "RELEASE","START","STOP","STOP PEND","SUSP", "SUSP PEND", "RESUME","ACTIVE","SHUT PEND", "IDLE"
SpoolFileID	string	"Onnn" ID of the spoolfile currently being printed on the device
SpoolingEnabled	bool	True if spooling has been enabled for this Device
SpoolType	string	Device's spooler type: "INSP"=owned by input spooler, "OUTSP"=owned by output spooler, "" =not owned by a spooler process
SubClass	string	Device subclass. "UNKNOWN","TMUX","TERM", "PRINTER", "VIRT TERM","VIRT PRINTER","PAD TERM","PAD PRINTER","NULL TERM","DHCF TERM", "PSEUDO TERM", "PSEUDO NULL TERM", "TAPE", "DISK"
TapeBuffering	bool	True means tape device is operating in "immediate report mode", meaning the device buffers data until it has enough to flush to tape
TapeDensity	int	0=unknown, 800 (BPI), 1600 (BPI), 6250 (BPI)
TapeDevList	string	List of all tape devices separated by a space. Format:"Total/LDEV1 LDEV2 ... LDEVN" *
TapeEOF	bool	True when tape is positioned at the end-of- file marker
TapeError	string	Fatal tape error: "NON-FATAL","RUNAWAY", "MULTI-TRACK","TIMING","CMD REJECT","UNIT", "DATA

		PARITY","CMD PARITY"
TapeOnline	bool	True means the tape drive in online
TapeTrackError	bool	True means a single track error was found
TapeWriteProtect	bool	True means tape is write protected
TermDevList	string	List of all terminal device separated by a space. Format:"Total/LDEV1 LDEV2 ... LDEVN" *
TrailerDisabled	bool	True if the trailer pages are suppressed in Printing
TypeAhead	bool	True means typeahead is enabled

* There are chances the output may exceed the usual limit of 1024 characters. When this happens, the overflow is handled by removing trailing LDEV numbers from the list. Due to the presence of a bug, the total count is also corrected and so there is no way of detecting the overflow. This bug will be fixed in a future CI patch. Till such time, it is recommended not to retrieve a list of device numbers where the output can exceed 1024 characters.

** Requires HP Security Monitor/iX

Examples

```
:calc DEVINFO(20, "ownership")
OWNED
:calc DEVINFO(0, "DiskDevList")
6/1 2 3 5 190 191
:calc DEVINFO(hpldevin, "Interactive")
TRUE
```

Additional Information

1) Parameter 1 (Device ID) can be “” or 0 for items that return a list of LDEV numbers. The following are the items that return a list:

- AllPrinterDevList
- AllTermDevList
- CommDevList
- DiscDevList / DiskDevList
- DSPrinterDevList
- DSTermDevList
- NotConfigDevList
- PrinterDevList
- SerialPrinterDevList
- SpoolerDevList
- TapeDevList
- TermDevList

AllPrinterDevlist includes DS printers and serial printers. AllTermDevList includes all MPE session terminal numbers and DS terminal numbers.

For all items that return a list, this result will be truncated if it is in excess of 1024 characters. The algorithm removes trailing LDEV numbers from the list till it is lesser or equal to the size of a CI variable.

2) BlockmodeType: This item works on Terminal LDEVs only. It has two possible values – 7 for line and DTC type terminals and 15 for PAD terminals.

3) Filename: This item returns the formal device file designator for only terminal LDEVs; for non-terminals, blanks are returned.

4) Opened: Device is opened if it is owned. Transitory owning of the

device by the OS does not indicate that the device is opened.

5) Terminal Type: The behavior of this item varies with the connection type (for example, DTC direct connect, PAD, or Telnet/iX). Console device managers only support type 10.

6) Read TimeOut: This returns the timeout in seconds and depends on the connection type.

SPOOLINFO

The SPOOLINFO CI function provides information about output and input spoolfiles stored in the system. It returns string, integer, or Boolean values depending upon the item queried. It is a CI Interface to AIFSPFGET.

Syntax

```
SPOOLINFO(spoolID, item[, status])
```

Parameters

SPOOLINFO accepts a spoolfile ID, item, and status as parameters. Out of these, the spoolfile ID and item parameters are mandatory and the status parameter is optional.

Spoolfile ID

This parameter is of type string. It identifies the target spoolfile as “[#] Onnnn” or “[#]Innnn”, where “nnnn” is a number. When specifying the spoolfile ID, # can be retained or omitted. After O or I in the string, a number should be specified which is the ID of the spoolfile. O and I identify whether the spoolfile is an output spoolfile or an input spoolfile. Following are the valid and invalid specifications of spoolfile ID.

Spoolfile ID	Validity	Reason
1234	Invalid	O/I is not present to identify the type of file.
O1435	Valid	Output spoolfile with ID 1435.
#I43435	Valid	Input spoolfile with ID 43435.
O15@53	Invalid	15@53 is not a valid number.

Item

This parameter is also of type string. It specifies which information about a spoolfile is to be returned. There are about 37 items which can be queried using SPOOLINFO. All the items are not valid for both output and input spoolfiles. Some items are valid if used only with output spoolfiles and some items are valid when used with any type of spoolfile. There is only one item, “RESTART” which can be used only with input spoolfile and invalid when used with output spoolfile. The following table summarizes the list of items which can be queried with SPOOLINFO.

Items valid for		
Output spoolfiles only	Any type of file	Input spoolfiles only

Aborted	Active	Restart
ActiveDeviceName	DeviceClass	
Broadcastable	DevRecSize	
Disposition	DevSubType	
EnvFileName	DevType	
FormsId	Exists	
FormsMsg	FileDesignator	
Incomplete	FmtCreateDate	
NumCopies	FmtCreateTime	
NumCopiesDone	InputType	
NumPages	IntCreateDate	
OutPriority	IntCreateTime	
RestartPageNum	Job User Account	
	JobName	
	JobNum	
	NumRecords	
	NumSectors	
	Opened	
	OpenState	
	Private	
	Ready	
	SpoolfileState	
	User Account	

Status

Status parameter is optional. It is the CI variable name where a SPOOLINFO error number can be returned. Zero indicates success. Non-zero indicates a failure. Error values returned in 'status' are the same as the CIERROR value when an error occurs. If the 'status' parm is passed the CI will not set CIERROR and will not report an error message. If 'status' is not specified the CI performs its normal error handling.

Description of SPOOLINFO items

The following table summarizes the description, the output type returned by all the items and the aliases for the items accepted by the SPOOLINFO CI function.

Item String	Type	Description
Aborted	bool	Returns TRUE if the spoolfile is the $STDLST of an aborted job otherwise returns FALSE.
Active	bool	Returns TRUE if input spoolfile state is 1 (active), or output spoolfile state is 5 (transfer), 6 (print), 10 (interrupt), otherwise returns FALSE.
ActiveDeviceName ActiveDev	string	Returns the name of the last device that has picked up a copy of the spoolfile and is currently processing the spoolfile.
Broadcastable	bool	Returns TRUE if the broadcastable field of an output spoolfile is set, otherwise returns FALSE. Used to print additional copies of SPSAVE'd output spoolfiles.
DeviceClass DevClass	string	Returns the destination device name or device class for the spoolfile.
DevRecSize	int	Returns the record size of the target device of the spoolfile in bytes.
DevSubType	int	Returns the device subtype of the target device of the spoolfile.
DevType	int	Returns the device type of the target device of the spoolfile
Disposition Disp	string	Returns "SAVE" or "PURGE" depending upon whether the spoolfile has to be saved or purged after it has been printed.
EnvFileName	string	Returns the environment file name of the spoolfile.

Env		
Exists Exist	bool	Returns TRUE if the spoolfile exists in any state, otherwise returns FALSE.
FileDesignator	string	Returns the formal file designator of the spoolfile if present, else “” (Null string) is returned.
FmtCreateDate	string	Returns the date when the spoolfile was created as a formatted string, “DAY, MONTH DATE, YEAR”.
FmtCreateTime	string	Returns the time when the spoolfile was created as a formatted string, “HH:MM AM/PM”.
FormsId	string	Returns the forms ID of the spoolfile.
FormsMsg	bool	Returns TRUE if the spoolfile has a forms message associated with it, otherwise returns FALSE.
Incomplete	bool	Returns TRUE if the spoolfile is incomplete, otherwise returns FALSE.
InputType	string	Returns “JOB” or “DATA” depending upon whether the input spoolfile is a job or data file. For output spoolfiles, “” (Null string) is returned.
IntCreateDate	int	Returns the date when the spoolfile was created as a YYYYMMDD integer.
IntCreateTime	int	Returns the time when the spoolfile was created as a HHMMSS integer.
Job User Account JobUserAccount	string	Returns “JOB,USER.ACCOUNT” of the creator of the spoolfile.
JobName	string	Returns the job name under which the spoolfile was created if supplied, else “” (Null string) is returned.
JobNum	string	Returns the Job/Session number under which the spoolfile was created as a formatted string, “#J/Snnnn”.
NumCopies Copies	int	Returns the total number of copies requested for the spoolfile.
NumCopiesDone	int	Returns the number of copy that has already been printed.
NumPages Pages	int	Returns the number of pages in the spoolfile.
NumRecords Records	int	Returns the number of records in the spoolfile.
NumSectors	int	Returns the number of sectors in the spoolfile.
Opened	bool	Returns TRUE if the input spoolfile state is 0 (open) or the output spoolfile state is 2 (create).
OpenState	string	Returns “OPEN” or “OPEN EXCL” depending upon whether the spoolfile is currently opened or exclusively opened. “” (Null string) is returned if the spoolfile is not opened.
Outpriority Pri	int	Returns the output priority of the spoolfile, a value in the range 0..14.
Private	bool	Returns TRUE if the spoolfile is a private spoolfile, otherwise FALSE.
Ready	bool	Returns TRUE if the input spoolfile state is 4 (ready), or the output spoolfile state is 3 (defer), 4 (ready), 7 (problem), 8 (delpend), 9 (spsave), otherwise returns FALSE.
Restart	bool	Returns TRUE if the job is restartable, otherwise returns FALSE.
RestartPageNum	int	Returns the page at which to restart if the printing of the spoolfile has been suspended.
SpoolfileState State	string	Returns state of the spoolfile, "OPEN", "ACTIVE", "CREATE", "DEFER", “READY", "TRANSFER", "PRINT", "PROBLEM", "DELPEND", "SPSAVE", or "INTERRUPT".
User Account UserAccount	string	Returns “USER.ACCOUNT” of the creator of the spoolfile.

Additional Information

1. **EnvFileName/Env**

   To query this item, the system should not be rebooted since the

spoolfile was created. Otherwise, “” (Null string) will be returned as output.

2. **FmtCreateDate**

   To query this item, the spoolfile should be past the ACTIVE/CREATE state. It should be at least in the READY state. If the spoolfile is in ACTIVE/CREATE state, an error is reported with SPOOLINFO.

   The output is formatted as “DAY, MONTH DATE, YEAR”.

   E.g.: If the spoolfile was created on 12/20/2005, then the output will be
   TUE, DEC 20, 2005.

3. **FmtCreateTime**

   To query this item, the spoolfile should be past the ACTIVE/CREATE state. It should be at least in the READY state. If the spoolfile is in ACTIVE/CREATE state, an error is reported with SPOOLINFO.

   The output is formatted as “HH:MM AM/PM”.

   E.g.: If the spoolfile was created at 21:59, then the output will be
   9:59 PM.

4. **InputType**

   This item is appropriate when used with input spoolfiles. It returns “JOB” or “DATA” depending upon whether the input spoolfile is a job or data file. However, when this item is used with output spoolfiles, “” (Null string) is returned as output.

5. **IntCreateDate**

   To query this item, the spoolfile should be past the ACTIVE/CREATE state. It should be at least in the READY state. If the spoolfile is in ACTIVE/CREATE state, an error is reported with SPOOLINFO.

   The output is formatted as a YYYYMMDD integer.

   E.g.: If the spoolfile was created on 12/20/2005, then the output will be
   20051220.

6. **IntCreateTime**

   To query this item, the spoolfile should be past the ACTIVE/CREATE state. It should be at least in the READY state. If the spoolfile is in ACTIVE/CREATE state, an error is reported with SPOOLINFO.

   The output is formatted as HHMMSS integer.

   E.g.: If the spoolfile was created at 21:59 PM and 32 secs, then the output will be
   215932.

   Consider the following example.

E.g.: A spoolfile was created at 08:02 AM and 05 secs, then the output will be
80205.

Notice that the output is not 080205 as the output type is integer (Leading zeroes are not seen). So, whenever the output integer has fewer than 6 digits, it has to be interpreted with a required number of leading zeroes. In the above example 80205, it is evident that the spoolfile was created at 8 hours, 2 mins and 5 secs.

This can be consistently dealt via the CI as:

E.g.: setvar x decimal(spoolinfo("o123","IntCreateTime))
setvar x rht("000000",6 - len(x)) + x

When used as shown above, x will be CI variable of type string and it will contain leading 0s if output has fewer than 6 digits.

7. **NumPages/Pages**

   This item returns the number of pages in the spoolfile. A positive number indicates the actual number of pages in the spoolfile. A negative number indicates that the spoolfile has never been printed before, and the number is only an estimation.

8. **NumRecords/NumRecs**

   This item returns the number of records in the spoolfile. The spoolfile must be past the CREATE state. That is, the spoolfile must be created successfully. If the spoolfile is in CREATE state, 0 is returned as number of records for the spoolfile.

9. **OpenState**

   This item returns “OPEN” or “OPEN EXCL” depending upon whether the spoolfile is currently opened or exclusively opened. “” (Null string) is returned if the spoolfile is not opened.

10. **Private**

    This item returns whether the spoolfile is a private spoolfile. All input spoolfiles are created with the private option.

11. **Job User Account/JobUserAccount**

    This item returns “JOB,USER.ACCOUNT” of the creator of the spoolfile. When there is no jobname specified, only USER.ACCOUNT is returned.

12. **DevRecSize**

    This item returns the record size of the target device of the spoolfile in bytes. This is usually 132 for output spoolfiles and 256 for input spoolfiles.

VOLINFO

The VOLINFO CI function returns information about volumes, volume classes, and volume sets. It is a CI interface to the existing HPVOLINFO

intrinsic. Depending on the information desired, VOLINFO returns a string, Boolean, or an integer.

Syntax

VOLINFO (VolumeID, item[,status])

Parameters

VOLINFO accepts three parameters:

- A volume ID: the volume, volume class or volume set about which information is returned.
- An item indicating what information is desired.
- An optional status parameter to return the status of the function call.

Volume ID

The volume ID parameter identifies a volume, volume class, or volume set. Volumes can be specified by their name or their LDEV number. For convenience, an LDEV number can be an integer or a string. A volume within a volume set is specified as "VolumeSet:VolumeName" or via its corresponding LDEV number. A volume class within a volume set is specified as "VolumeSet:VolumeClassName". Since "volset:volname" and "volset:volclass" are syntactically ambiguous, all volume class items have "volclass" in their name. For example, the value of parameter one could be "abc:def" (which could name a volume or volume class), but if the item selected is "VolClassExists" then it is obvious that parameter one contains a volume class name. In all cases, it should be clear from the item description what is expected in parameter one.

There are cases where parameter one is not a volume set, volume class, LDEV, or volume name. For instance, to retrieve the number of volume sets ("NumVS" item) the volume ID parameter can be wildcarded or set to the empty string, "", meaning all or "@".

Item

This parameter is a key-word that indicates what information is required. It is a string parameter. A list of supported items and their description can be found in a separate section below. The list includes the item name, its aliases, return type of VOLINFO and a description of the item.

Status

Status is an optional 32 bit integer parameter. It is the CI variable name where a VOLINFO error number can be returned. Zero indicates success. Non-zero indicates a failure. Error values returned in 'status' are the same as the CIERROR value when an error occurs. If the 'status' parm is passed, the CI will not set CIERROR and will not report an error message. If 'status' is not specified the CI performs its normal error handling.

Description of VOLINFO Items

The following table summarizes the items of the VOLINFO CI function. The description includes the item string, one or more aliases, the return type, and a brief description of the item.

Item String	Type	Description
Drivetype	String	Returns the drive type of the specified volset:volname or LDEV. The drive type is the product ID of the hard-disk. E.g. ST15150W

Exists Exist	bool	True if LDEV, volset:volname, or volset exists, regardless of whether it is mounted or opened. "Exists" confirms the input value is configured. Use "mounted" to check if a volume is mounted. Use "opened" to check if a volume has been vsopen'd. Use "volclass exists" for volume classes
LDEV VolLdev	int	Returns the LDEV of the specified volset:volname
Loner	bool	True if LDEV or volset:volname exists and is in the "loner" state. See also "exists", "mounted", and "opened"
MemberVolClassNames MemberVolClasses	string	A list of volume class names, separated by a space, which the specified LDEV or volset:volname is a member of. Format:"total/name1 name2..nameN"
Mounted	bool	True if LDEV or volset:volname exists and is mounted. See also "exists" and "opened"
NumVolumeClasses NumVolClass	int	Returns the number of volume classes in the volume set
NumVolClassVolumes NumVolClassVols	int	Returns the number of volumes in the Volume class
NumVolSetVolumes NumVSVols	int	Returns the number of volumes in the volume set, including the master volume
NumVolumeSets NumVS	int	Returns the number of volume setsin the system. VolID is "", meaning "@", or a wildcard pattern. e.g. "@TEST@","USER@"
Opened	bool	True if the LDEV or volset:volname is mounted and vsopen'd. See also "exists" and "mounted"
SystemVolume	bool	True if LDEV or volset:volname is a system volume
UserVolume	bool	True if LDEV or volset:volname is a user volume
VolClassDirSpaceMB	int	Returns the directory space overhead in megabytes of the specified volume class
VolClassDirSpaceSectors	string	Returns the directory space overhead in sectors of the specified volume class as a numeric string
VolClasses VSVolClassNames	string	A list of volume class names, separated by a space, belonging to the specified volume set. Format:"total/name1 name2..nameN"
VolClassExists VolClassExist	bool	True if volset:volclass exists. Use "exists" for LDEV, volset:volume or volset names
VolClassFileLabelMB	int	Returns the File Label overhead in megabytes of the specified volume class
VolClassFileLabelSectors	string	Returns the File Label overhead in sectors of the specified volume class as a numeric string
VolClassFreeSpaceMB	int	Returns the total free space in megabytes in the specified volume class
VolClassFreeSpaceSectors	string	Returns the total free space in sectors in the specified volume class as a numeric string
VolClassLargestSpaceMB	int	Returns the largest contiguous free space in megabytes in the specified volume class
VolClassLargestSpaceSectors	string	Returns the largest contiguous free space in sectors in the specified volume class as a numeric string
VolClassMaxTransSpaceMB	int	Returns the maximum configured transient space in megabytes of the specified volume class
VolClassMaxTransSpaceSectors	string	Returns the maximum configured transient space in sectors of the specified volume class as a numeric string
VolClassOverheadMB	int	Returns the total MPE/iX overhead in megabytes of the specified volume class
VolClassOverheadSectors	string	Returns the total MPE/iX overhead in sectors of the specified volume class as a numeric string
VolClassPermFilesMB	int	Returns the permanent file disc usage in megabytes of the specified volume class
VolClassPermFilesSectors	string	Returns the permanent file disc usage in sectors of the specified volume class as a numeric string

VolClassSizeMB	int	Returns the total size in megabytes of the specified volume class
VolClassSizeSectors	string	Returns the total size in sectors of the specified volume class as a numeric string
VolClassSpoolSpaceMB	int	Returns the spoolfile space in megabytes of the specified volume class
VolClassSpoolSpaceSectors	string	Returns the spoolfile space in sectors of the specified volume class as a numeric string
VolClassTransSpaceMB	int	Returns the transient space overhead in megabytes of the specified volume class
VolClassTransSpaceSectors	string	Returns the transient space overhead in sectors of the specified volume class as a numeric string
VolClassVols VolClassVolumeNames	string	A list of volume names, separated by a space, belonging to the specified volume class. Format:"total/name1 name2 ..nameN"
VolClassXMMB	int	Returns the Transaction Management (XM) overhead in megabytes of the specified volume class
VolClassXMSectors	string	Returns the Transaction Management (XM) overhead in sectors of the specified volume class as a numeric string
VolDirSpaceMB	int	Returns the directory space overhead in megabytes of the specified LDEV or volset:volname
VolDirSpaceSectors	string	Returns the directory space overhead in sectors of the specified LDEV or volset:volname as a numeric string
VolFileLabelMB	int	Returns the File Label overhead in megabytes of the specified LDEV or volset:volname
VolFileLabelSectors	string	Returns the File Label overhead in sectors of the specified LDEV or volset:volname as a numeric string
VolFreeSpaceMB	int	Returns the total free space in megabytes in the specified LDEV or volset:volname
VolFreeSpaceSectors	string	Returns the total free space in sectors in the specified LDEV or volset:volname as a numeric string
VolLargestSpaceMB	int	Returns the largest contiguous free space in megabytes in the specified LDEV or volset:volname
VolLargestSpaceSectors	string	Returns the largest contiguous free space in sectors in the specified LDEV or volset:volname as a numeric string
VolMaxTransSpaceMB	int	Returns the maximum configured transient space in megabytes of the specified LDEV or volset:volname
VolMaxTransSpaceSectors	string	Returns the maximum configured transient space in sectors of the specified LDEV or volset:volname as a numeric string
VolName	string	Returns the volume name of the specified LDEV
VolOverheadMB	int	Returns the total MPE/iX overhead in megabytes of the specified LDEV or volset:volname
VolOverheadSectors	string	Returns the total MPE/iX overhead in sectors of the specified LDEV or volset:volname as a numeric string
VolPermFilesMB	int	Returns the permanent file disc usage in megabytes of the specified LDEV or volset:volname
VolPermFilesSectors	string	Returns the permanent file disc usage in sectors of the specified LDEV or volset:volname as a numeric string
VolSetDirSpaceMB	int	Returns the directory space overhead inmegabytes of the specified volume set
VolSetDirSpaceSectors	string	Returns the directory space overhead in sectors of the specified volume set as a numeric string
VolSetFileLabelMB	int	Returns the File Label overhead in megabytes of the specified volume set
VolSetFileLabelSectors	string	Returns the File Label overhead in sectors of the specified volume set as a numeric string
VolSetFreeSpaceMB	int	Returns the total free space in megabytes in the specified volume set

VolSetFreeSpaceSectors	string	Returns the total free space in sectors in the specified volume set as a numeric string
VolSetLargestSpaceMB	int	Returns the largest contiguous free space in megabytes in the specified volume set
VolSetLargestSpaceSectors	string	Returns the largest contiguous free space in sectors in the specified volume set as a numeric string
VolSetMaxTransSpaceMB	int	Returns the maximum configured transient space in megabytes of the specified volume set
VolSetMaxTransSpaceSectors	string	Returns the maximum configured transient space in sectors of the specified volume set as a numeric String
VolSetName	string	Returns the volume set name of thespecified LDEV
VolSetOverheadMB	int	Returns the total MPE/iX overhead in megabytes of the specified volume set
VolSetOverheadSectors	string	Returns the total MPE/iX overhead in sectors of the specified volume set as a numeric string
VolSetPermFilesMB	int	Returns the permanent file disc usage in megabytes of the specified volume set
VolSetPermFilesSectors	string	Returns the permanent file disc usage in sectors of the specified volume set as a numeric string
VolSets VolumeSetNames	string	A list of volume set names separated by a space. Format:"total/name1 name2..nameN". VolId can be "", meaning "@", or a wildcarded pattern
VolSetSizeMB	int	Returns the total size in megabytes of the specified volume set
VolSetSizeSectors	string	Returns the total size in sectors of the specified volume set as a numeric string
VolSetSpoolSpaceMB	int	Returns the spoolfile space in megabytes of the specified volume set
VolSetSpoolSpaceSectors	string	Returns the spoolfile space in sectors of the specified volume set as a numeric string
VolSetTransSpaceMB	int	Returns the transient space overhead in megabytes of the specified volume set
VolSetTransSpaceSectors	string	Returns the transient space overhead in sectors of the specified volume set as a numeric string
VolSetXMMB	int	Returns the Transaction Management (XM) overhead in megabytes of the specified volume set
VolSetXMSectors	string	Returns the Transaction Management (XM) overhead in sectors of the specified volume set as a numeric String
VolSizeMB	int	Returns the size in megabytes of the specified LDEV or volset:volname
VolSizeSectors	string	Returns the size in sectors of the specified LDEV or volset:volname as a numeric string
VolSpoolSpaceMB	int	Returns the spoolfile space in mega bytes of the specified LDEV or volset:volname
VolSpoolSpaceSectors	string	Returns the spoolfile space in sectors of the specified LDEV or volset:volname as a numeric string
VolTransSpaceMB	int	Returns the transient space overhead in megabytes of the specified LDEV or volset:volname
VolTransSpaceSectors	string	Returns the transient space overhead in sectors of the specified LDEV or volset:volname as a numeric string
VolType	string	Returns the status of the LDEV or volset:volname as displayed by the DSTAT command:"MASTER" "MEMBER","LONER", "SCRATCH","UNKNOWN", "DISABLED", "LOCKED"
VolXMMB	int	Returns the Transaction Management (XM) overhead in megabytes of the specified LDEV or volset:volname
VolXMSectors	string	Returns the Transaction Management (XM) overhead in sectors of the specified LDEV or volset:volname as a numeric string
VSVols	string	A list of volume names, separated by a space, belonging to

VSVolumeNames		the specified volume set. Format:"total/name1 name2..nameN"

* A CI string variable cannot exceed 1024 characters and thus the result could be truncated. Truncation is evident by checking wordcnt() < total. E.g.

```
setvar x VOLINFO('@','VolSets')
if wordcnt(x) < ![word(x,'/')] then
  # truncated list...
```

Examples

```
:calc                                    VOLINFO("@",
"VOLSETS")
2/MPEXL_SYSTEM_VOLUME_SET
RATVOL_VOLMGT_QASYSXL

:calc            VOLINFO("MPEXL_SYSTEM_VOLUME_SET",
"Exist")
TRUE

:calc    VOLINFO("MPEXL_SYSTEM_VOLUME_SET:MEMBER1",
"LDEV")
1
```

Additional Information

MPE/iX Transient Space Overhead
This consists of the volume space that is used for temporary processes, such as stacks, heaps and the operating system datastructures.

MPE/iX Virtual Memory Overhead
The virtual memory overhead is the volume space that is allocated for saving segments of main memory temporarily. This amount of configured virtual memory space is equivalent to the amount of used virtual memory space. Virtual memory is allocated only on system volumes; there is no virtual memory on private volumes.

MPE/iX Directory Space Overhead
Directory space is area on system and non-system volumes reserved for accounting information. It consists of the directory space used for permanent files.

MPE/iX File Label Overhead
On MPE/iX, each volume has its own label table, which contains file labels and extent descriptors for files that begin on that volume.

MPE/iX transaction management overhead
Transaction management overhead consists of any logging information that is maintained in order to provide file consistence and file recovery.

Spool file disk usage
Spoolfile space consists of the volume space that is used by hidden spoolfiles. Hidden spoolfiles are not part of the permanent file space. This space can be found only on system volumes that are configured with the device class SPOOL. Spool file space is not allocated on nonsystem volumes.

Total MPE/iX Overhead
This consists of everything on a volume that is not set aside for file space use. This is the volume space used for operating system

purposes. This includes volume label, file label table, directory, volume set information table, free space map, transient space, and transaction management overhead.

Size Representation
The maximum integer value that can be returned is a value that can be represented in 32 bits. For items that return a number of sectors, the return value is a numeric string as this number can exceed the 32 bit limit.

Nested Calls
VOLINFO supports nested calls. The maximum level of nesting is same as the limit set by the MPE/iX CI.