

```

--
-- file: NMSysMgr.mib
--
-- COMTEK Services, Inc.
-- MIB for VMS NM*SysMgr Subagent
-- Release      3.5
-- Date        January 2002
-- Author      NF
--
-- Copyright 1995-2002 COMTEK Services, Inc. All Rights Reserved.
--
-- This COMTEK Services VMS NM*SysMgr SNMP Management Information Base
-- Specification (MIB) embodies COMTEK Services' confidential and
-- proprietary intellectual property. COMTEK Services retains all
-- title and ownership in the Specification, including any
-- revisions.
--
-- This Specification is supplied "AS IS," and COMTEK Services makes
-- no warranty, either express or implied, as to the use,
-- operation, condition, or performance of the Specification.
--

COMTEK-VMS-NM-SYSMGR-MIB DEFINITIONS ::= BEGIN

IMPORTS
    DisplayString                FROM RFC1213-MIB
    OBJECT-TYPE                  FROM RFC-1212
    TimeTicks, Counter           FROM RFC1155-SMI
    comtekVmsNMSysMgrMib        FROM COMTEK-DEFINITIONS-MIB;

-- This lists system resources available (system, CPU, disk, queue)
sResources OBJECT IDENTIFIER ::= { comtekVmsNMSysMgrMib 1 }

-- This is a table of VMS process statistics
sProcesses OBJECT IDENTIFIER ::= { comtekVmsNMSysMgrMib 2 }

-- This is trap data
sTrapInfo OBJECT IDENTIFIER ::= { comtekVmsNMSysMgrMib 3 }

-- This lists error information (hardware, software, operator log)
sErrInfo OBJECT IDENTIFIER ::= { comtekVmsNMSysMgrMib 4 }

-- This lists NM*SysMgr Subagent configuration parameters
sCfg OBJECT IDENTIFIER ::= { comtekVmsNMSysMgrMib 5 }

sYi OBJECT IDENTIFIER ::= { sResources 1 }
sCpu OBJECT IDENTIFIER ::= { sResources 2 }
sDsk OBJECT IDENTIFIER ::= { sResources 3 }
sShd OBJECT IDENTIFIER ::= { sResources 4 }
sQue OBJECT IDENTIFIER ::= { sResources 5 }

sProcInfo OBJECT IDENTIFIER ::= { sProcesses 1 }
sCritInfo OBJECT IDENTIFIER ::= { sProcesses 2 }

sHwErrInfo OBJECT IDENTIFIER ::= { sErrInfo 1 }
sSwErrInfo OBJECT IDENTIFIER ::= { sErrInfo 2 }

```

```

-- System Information
sSyiNodeName OBJECT-TYPE
    SYNTAX DisplayString (SIZE (0..15))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The name of the node."
    ::= { sSyi 1 }

sSyiHwName OBJECT-TYPE
    SYNTAX DisplayString (SIZE (0..31))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The hardware model name."
    ::= { sSyi 2 }

sSyiBootTime OBJECT-TYPE
    SYNTAX DisplayString (SIZE (23))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The date and time when the system was booted. This value
        is in the form dd-mmm-yyyy hh:mm:ss.cc."
    ::= { sSyi 3 }

sSyiVersion OBJECT-TYPE
    SYNTAX DisplayString (SIZE (0..8))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The software version number of the OpenVMS operating
        system running on the node."
    ::= { sSyi 4 }

sSyiMemSize OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total number of pages of physical memory in the system
        configuration. This value is static."
    ::= { sSyi 5 }

sSyiAvailCpuCnt OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of CPUs which were available at system bootstrap.
        This value is static."
    ::= { sSyi 6 }

sSyiActiveCpuCnt OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only

```

STATUS mandatory
DESCRIPTION
 "Number of active CPUs."
::= { sSyi 7 }

sSyiPgSize OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The number of pages in the currently installed page files."
::= { sSyi 8 }

sSyiPgFree OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The number of free pages in the currently installed page
 files."
::= { sSyi 9 }

sSyiPgUsedPercent OBJECT-TYPE
SYNTAX INTEGER (0..100)
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The percent full of the currently installed page files."
::= { sSyi 10 }

sSyiSwpSize OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The number of pages in the currently installed swapping
 files."
::= { sSyi 11 }

sSyiSwpFree OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The number of free pages in the currently installed
 swapping files."
::= { sSyi 12 }

sSyiSwpUsedPercent OBJECT-TYPE
SYNTAX INTEGER (0..100)
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The percent full of the currently installed swapping files."
::= { sSyi 13 }

sSyiCpuPgSize OBJECT-TYPE

```

SYNTAX    INTEGER
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "The number of CPU-specific bytes per page in the system.  On
    VAX systems, this value is always 512.  On Alpha systems, CPU
    page size varies from system to system."
::= { sSyi 14 }

sSyiTime OBJECT-TYPE
SYNTAX    TimeTicks
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "Subagent up time (sCfgUpTime) when system information (sSyi)
    was last gathered.  sCfgSysInfoTimer controls how often system
    statistics are gathered."
::= { sSyi 15 }

sSyiMemFreePg OBJECT-TYPE
SYNTAX    INTEGER
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "Number of free pages of physical memory on the system."
::= { sSyi 16 }

sSyiMemUsed OBJECT-TYPE
SYNTAX    INTEGER
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "Physical memory percentage used."
::= { sSyi 17 }

sSyiCondition OBJECT-TYPE
SYNTAX    INTEGER { unknown(1), ok(2), degraded(3), failed(4) }
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "Indicates current condition of System-related statistics.
    DEGRADED indicates that one or more of the following has
    exceeded the defined limit: page file usage, swap file usage,
    memory usage."
::= { sSyi 18 }

-- CPU Statistics
sCpuPercentUsed OBJECT-TYPE
SYNTAX    INTEGER (0..100)
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "The percent utilization of all CPUs, i.e., the percent of
    time the combined CPUs were not idle during the last one
    minute interval."
::= { sCpu 1 }

```

```

sIntPercentUsed OBJECT-TYPE
    SYNTAX  INTEGER (0..100)
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The percent of time the combined CPUs spent in interrupt
        mode during the last one minute interval."
    ::= { sCpu 2 }

sCpuTicks OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of interval timer ticks that have occurred since the
        last time CPU statistics were gathered (approximately one
        minute). This value is the sum of the one minute mode tick
        counters. This number has been normalized to account for all
        active CPUs on the system. This value may be used with the
        one minute CPU mode times to determine the percent of time
        spent in each CPU mode."
    ::= { sCpu 3 }

-- CPU Statistics for the past one minute interval
sCpuLoadOneMinute OBJECT IDENTIFIER ::= { sCpu 4 }

sKernelOne OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Time in Kernel mode for all CPUs during the last one minute
        interval. This value indicates the number of ticks in kernel
        mode on the kernel stack, when no spinlock busy wait is
        active."
    ::= { sCpuLoadOneMinute 1 }

sExecOne OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of interval timer ticks in Executive mode for all
        CPUs during the last one minute interval."
    ::= { sCpuLoadOneMinute 2 }

sSuprOne OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of interval timer ticks in Supervisor mode for all
        CPUs during the last one minute interval."
    ::= { sCpuLoadOneMinute 3 }

sUserOne OBJECT-TYPE

```

```

SYNTAX    INTEGER
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "Number of interval timer ticks in User mode for all CPUs
    during the last one minute interval."
::= { sCpuLoadOneMinute 4 }

sIntrOne OBJECT-TYPE
SYNTAX    INTEGER
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "Time in Interrupt mode for all CPUs during the last one
    minute interval. This value indicates the number of interval
    timer ticks in Kernel mode on the Interrupt stack."
::= { sCpuLoadOneMinute 5 }

sCompOne OBJECT-TYPE
SYNTAX    INTEGER
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "Number of interval timer ticks in Compatibility mode for all
    CPUs during the last one minute interval."
::= { sCpuLoadOneMinute 6 }

sSpinOne OBJECT-TYPE
SYNTAX    INTEGER
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "Number of interval timer ticks in mpsynch mode (i.e., Kernel
    mode on kernel or interrupt stack, when spinlock busy wait
    is active) for all CPUs during the last one minute interval."
::= { sCpuLoadOneMinute 7 }

sNullOne OBJECT-TYPE
SYNTAX    INTEGER
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "CPU idle time for all CPUs during the last one minute interval.
    This value is measured in interval timer ticks."
::= { sCpuLoadOneMinute 8 }

-- CPU Statistics for the past five minutes
sCpuLoadFiveMinute OBJECT IDENTIFIER ::= { sCpu 5 }

sKernelFive OBJECT-TYPE
SYNTAX    INTEGER
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "Number of ticks spent in Kernel mode during the past five
    minutes."

```

```
 ::= { sCpuLoadFiveMinute 1 }

sExecFive OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of ticks spent in Executive mode during the past five
        minutes."
    ::= { sCpuLoadFiveMinute 2 }

sSuprFive OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of ticks spent in Supervisor mode during the past five
        minutes."
    ::= { sCpuLoadFiveMinute 3 }

sUserFive OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of ticks spent in User mode during the past five
        minutes."
    ::= { sCpuLoadFiveMinute 4 }

sIntrFive OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of ticks spent in Interrupt mode during the past five
        minutes."
    ::= { sCpuLoadFiveMinute 5 }

sCompFive OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of ticks spent in Compatibility mode during the past
        five minutes."
    ::= { sCpuLoadFiveMinute 6 }

sSpinFive OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of ticks spent in Spin mode during the past five
        minutes."
    ::= { sCpuLoadFiveMinute 7 }

sNullFive OBJECT-TYPE
```

```

SYNTAX    INTEGER
ACCESS    read-only
STATUS    mandatory
DESCRIPTION
    "Number of ticks the CPUs were idle during the past five
    minutes."
::= { sCpuLoadFiveMinute 8 }

-- CPU Statistics for the past fifteen minutes
sCpuLoadFifteenMinute OBJECT IDENTIFIER ::= { sCpu 6 }

sKernelFifteen OBJECT-TYPE
    SYNTAX    INTEGER
    ACCESS    read-only
    STATUS    mandatory
    DESCRIPTION
        "Number of ticks spent in Kernel mode during the past fifteen
        minutes."
    ::= { sCpuLoadFifteenMinute 1 }

sExecFifteen OBJECT-TYPE
    SYNTAX    INTEGER
    ACCESS    read-only
    STATUS    mandatory
    DESCRIPTION
        "Number of ticks spent in Executive mode during the past
        fifteen minutes."
    ::= { sCpuLoadFifteenMinute 2 }

sSuprFifteen OBJECT-TYPE
    SYNTAX    INTEGER
    ACCESS    read-only
    STATUS    mandatory
    DESCRIPTION
        "Number of ticks spent in Supervisor mode during the past
        fifteen minutes."
    ::= { sCpuLoadFifteenMinute 3 }

sUserFifteen OBJECT-TYPE
    SYNTAX    INTEGER
    ACCESS    read-only
    STATUS    mandatory
    DESCRIPTION
        "Number of ticks spent in User mode during the past fifteen
        minutes."
    ::= { sCpuLoadFifteenMinute 4 }

sIntrFifteen OBJECT-TYPE
    SYNTAX    INTEGER
    ACCESS    read-only
    STATUS    mandatory
    DESCRIPTION
        "Number of ticks spent in Interrupt mode during the past
        fifteen minutes."
    ::= { sCpuLoadFifteenMinute 5 }

```

```

sCompFifteen OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of ticks spent in Compatibility mode during the past
        fifteen minutes."
    ::= { sCpuLoadFifteenMinute 6 }

sSpinFifteen OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of ticks spent in Spin mode during the past fifteen
        minutes."
    ::= { sCpuLoadFifteenMinute 7 }

sNullFifteen OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of ticks the CPUs were idle during the past fifteen
        minutes."
    ::= { sCpuLoadFifteenMinute 8 }

sCpuCondition OBJECT-TYPE
    SYNTAX  INTEGER { unknown(1), ok(2), degraded(3), failed(4) }
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Indicates current condition of CPU-related statistics.
        DEGRADED indicates that CPU or Interrupt usage (or both)
        has exceeded the defined limit."
    ::= { sCpu 7 }

-- Disk Information
--
-- The disk table is a potentially sparse table of disk related information.
-- Each time disk information is collected, new rows are added to the bottom
-- of the disk table for new disks that are discovered.  For disks that
-- are no longer available, the corresponding row in the disk table becomes
-- inaccessible.  This allows the disk table to be walked to retrieve only
-- the current disk information while ensuring that a disk index (and
-- therefore the row in the disk table) relates to a specific disk.
-- When the NM*SysMgr Subagent is reinitialized (see sCfgReinitSubagent),
-- any inaccessible rows in the disk table are removed and the disk
-- indexes and rows are re-numbered to be contiguous.
--
-- Two configuration file flags are available to fine tune which disks are
-- monitored: sCfgLocalDisksOnly and sCfgCritDisksOnly.  These flags can
-- be used alone or in combination to control which disks appear in the
-- disk table (sDskTbl).  Disk related traps can only be sent for disks
-- that are in the disk table.
--
-- Note:  When disks are shadowed, disk information is reported only

```

-- for the shadow set virtual unit.

```
sDskTblCnt OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of entries in the disk table."
    ::= { sDsk 1 }
```

```
sDskTblTime OBJECT-TYPE
    SYNTAX  TimeTicks
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Subagent up time (sCfgUpTime) when disk statistics (sDskTbl)
        were last gathered. sCfgDskTimer controls how often disk
        statistics are gathered."
    ::= { sDsk 2 }
```

```
sDskTbl OBJECT-TYPE
    SYNTAX  SEQUENCE OF SDskEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "Table of disk information."
    ::= { sDsk 3 }
```

```
sDskEntry OBJECT-TYPE
    SYNTAX  SDskEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "A row in the disk table."
    INDEX   { sDskIndex }
    ::= { sDskTbl 1 }
```

```
SDskEntry ::= SEQUENCE {
    sDskIndex      INTEGER,
    sDskName       DisplayString,
    sDskUsedPercent INTEGER,
    sDskOps        INTEGER,
    sDskMountCnt   INTEGER,
    sDskRefCnt     INTEGER,
    sDskTransCnt   INTEGER,
    sDskMediaName  DisplayString,
    sDskOpCnt      INTEGER,
    sDskFreeBlocks INTEGER,
    sDskMaxBlocks  INTEGER,
    sDskStatus     INTEGER,
    sDskDevChar    INTEGER,
    sDskLogVolName DisplayString,
    sDskTrapPercent INTEGER,
    sDskTrapBlocks INTEGER,
    sDskShdCount   Counter,
    sDskTrapOpSec  INTEGER,
    sDskQueueLength Counter,
```

```

    sDskRemote          INTEGER,
    sDskEntryCondition INTEGER,
    sDskQueueLengthDelta INTEGER,
    sDskQueueLengthLimit INTEGER
}

sDskIndex OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Location in the disk table."
    ::= { sDskEntry 1 }

sDskName OBJECT-TYPE
    SYNTAX  DisplayString (SIZE (0..64))
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Device name."
    ::= { sDskEntry 2 }

sDskUsedPercent OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The disk percent full."
    ::= { sDskEntry 3 }

sDskOps OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The average number of disk operations per second during the
        last disk statistics update interval. (The disk statistics
        update frequency is determined by the variable sCfgDskTimer.)"
    ::= { sDskEntry 4 }

sDskMountCnt OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The mount count for the volume."
    ::= { sDskEntry 5 }

sDskRefCnt OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The number of channels assigned to the device."
    ::= { sDskEntry 6 }

sDskTransCnt OBJECT-TYPE

```

SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The transaction count for the volume."
::= { sDskEntry 7 }

sDskMediaName OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..64))
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The name of the volume type."
::= { sDskEntry 8 }

sDskOpCnt OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The operation count for the volume."
::= { sDskEntry 9 }

sDskFreeBlocks OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The number of free blocks on the disk."
::= { sDskEntry 10 }

sDskMaxBlocks OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The maximum number of blocks on the volume."
::= { sDskEntry 11 }

sDskStatus OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "The device unit status for the physical disk or virtual disk.
 This value is a bit vector which, when set, corresponds to the
 following states:

| Bit | Description |
|-----|---|
| 0 | TIM (time-out is enabled) |
| 1 | INT (Interrupt is expected) |
| 2 | ERLOGIP (Error log in progress on unit) |
| 3 | CANCEL (I/O on unit is canceled) |
| 4 | ONLINE (Unit is online) |
| 5 | POWER (Power failed while unit busy) |
| 6 | TIMOUT (Unit timed out) |
| 7 | INTTYPE (Receiver interrupt) |

```

8      BSY (Unit is busy)
9      MOUNTING (Device is being mounted)
10     DEADMO (Deallocate at dismount)
11     VALID (Volume is software valid)
12     UNLOAD (Unload volume at dismount)
13     TEMPLATE (Template UCB)
14     MNTVERIP (Mount verification is in progress)
15     WRONGVOL (Wrong volume detected during mount
verification)
16     DELETEUCB (Delete this UCB when reference count
equals 0)
17     LCL_VALID (Volume is valid on local node)
18     SUPMVMSG (Suppress mount verification messages if
they indicate success)
19     MNTVERPEND (Mount verification is pending on the
device and the device is busy)
20     DISMOUNT (Dismount in progress)
21     CLUTRAN (VAXcluster state transition in progress)
22     WRTLOCKMV (Write-locked mount verification in
progress)
23     SVPN_END (Last byte used from page is mapped by a
system virtual page number )
24     ALTBSY (Unit is busy via an alternate startup path)
25     SNAPSHOT (Restart verification is in progress)
26     NO_ASSIGN
27     EXFUNC_SUPP
28     FAST_PATH
29     PATHVERIP

```

A sDskStatus trap message is sent when a device transitions into the POWER, TIMEOUT, or MNTVERIP state."

```
::= { sDskEntry 12 }
```

sDskDevChar OBJECT-TYPE

```

SYNTAX  INTEGER
ACCESS  read-only
STATUS  mandatory
DESCRIPTION

```

"The disk device-independent characteristics of the disk. This value is a bit vector which, when set, corresponds to the following characteristic:

| Bit | Description |
|-----|--|
| 0 | REC (Device is record oriented) |
| 1 | CCL (Device is a carriage control device) |
| 2 | TRM (Device is a terminal) |
| 3 | DIR (Device is directory structured) |
| 4 | SDI (Device is single-directory structured) |
| 5 | SQD (Device is sequential and block oriented) |
| 6 | SPL (Device is being spooled) |
| 7 | OPR (Device is an operator) |
| 8 | RCT (Disk contains Revector Cache Table. This bit is set for every DAA disk) |
| 13 | NET (Device is a network device) |
| 14 | FOD (Device is files oriented) |
| 15 | DUA (Device is dual ported) |
| 16 | SHR (Device is shareable) |

```

17     GEN (Device is a generic device)
18     AVL (Device is available for use)
19     MNT (Device is mounted)
20     MBX (Device is a mailbox)
21     DMT (Device is marked for dismount)
22     ELG (Device has error logging enabled)
23     ALL (Device is allocated)
24     FOR (Device is mounted foreign)
25     SWL (Device is software write locked)
26     IDV (Device can provide input)
27     ODV (Device can provide output)
28     RND (Device allows random access)
29     RTM (Device is a real-time device)
30     RCK (Device has read-checking enabled)
31     WCK (Device has write-checking enabled)"
 ::= { sDskEntry 13 }

```

```

sDskLogVolName OBJECT-TYPE
SYNTAX  DisplayString (SIZE (0..64))
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
        "The logical name of the volume or volume set."
 ::= { sDskEntry 14 }

```

```

sDskTrapPercent OBJECT-TYPE
SYNTAX  INTEGER (0..100)
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
        "The percentage of disk space utilization at which excessive
        disk utilization (sDskFull) traps begin to be sent for this
        disk.  A value of 0 indicates that either a disk block free
        count is being used for this disk (see sDskTrapBlocks), or
        the system default disk use limit (identified by sCfgDskLimit),
        or the system default disk minimum free block limit
        (identified by sCfgDskMinFreeBlks) for the system is being
        used.  Disks which have a specific usage limit set are
        contained in the critical disk file.  See sCfgCritDsk for
        more information about this file."
 ::= { sDskEntry 15 }

```

```

sDskTrapBlocks OBJECT-TYPE
SYNTAX  INTEGER
ACCESS  read-only
STATUS  mandatory
DESCRIPTION
        "The minimum number of free disk blocks required for this disk.
        If the actual number of free blocks on the disk drops below this
        level, disk utilization (sDskFull) traps begin to be sent for
        this disk.  A value of 0 indicates that either a percent full
        is being used for this disk (see sDskTrapPercent) or
        the system default disk use limit (identified by sCfgDskLimit),
        or the system default disk minimum free block limit
        (identified by sCfgDskMinFreeBlks) for the system is being
        used.  Disks which have a specific usage limit set are
        contained in the critical disk file.  See sCfgCritDsk for

```

```

        more information about this file."
 ::= { sDskEntry 16 }

sDskShdCount OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of shadow set members that make up this virtual
        unit.  If this number is zero, then this is not a virtual
        disk.  This number indicates the number of shadow set
        entries that will be found in the (sShdTbl) for this disk."
 ::= { sDskEntry 17 }

sDskTrapOpSec OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The number of operations per second at which excessive
        disk (sDskOpSec) traps begin to be sent for this disk.
        A value of 0 indicates that the system default disk
        operations per second limit (identified by sCfgDskOps),
        is being used.  Disks which have a specific limit set
        are contained in the critical disk file.  If both this value
        and sCfgDskOps are zero, disk operations per second traps are
        disabled.  See sCfgCritDsk for more information about the
        critical disk file."
 ::= { sDskEntry 18 }

sDskQueueLength OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Total I/O queue length for this disk since it was mounted."
 ::= { sDskEntry 19 }

sDskRemote OBJECT-TYPE
    SYNTAX INTEGER { true(1), false(2) }
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "A value of true(1) indicates that the device is a remote
        device; a value of false(2) indicates that it is not a remote
        device.  A remote device is a device that is not directly
        connected to the local node, but instead is visible through
        the VMScluster system."
 ::= { sDskEntry 20 }

sDskEntryCondition OBJECT-TYPE
    SYNTAX INTEGER { unknown(1), ok(2), degraded(3), failed(4) }
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Indicates current condition of this disk entry.
        DEGRADED indicates that one or more of the following

```

```

        conditions exists: this disk's usage exceeds the defined
        threshold, this disk's operations/second exceeds the defined
        threshold, this disk's status indicates a potential problem,
        or a shadow set member has a status of failed, catch-up
        copying, or merge-copy."
 ::= { sDskEntry 21 }

sDskQueueLengthDelta OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "I/O queue length for this disk during the last statistics
        gathering period."
 ::= { sDskEntry 22 }

sDskQueueLengthLimit OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "I/O queue length threshold for this disk. When the I/O
        queue length (sDskQueueLengthDelta) exceeds this value, a
        sDskIOQLen(28) trap."
 ::= { sDskEntry 23 }

sDskCondition OBJECT-TYPE
    SYNTAX  INTEGER { unknown(1), ok(2), degraded(3), failed(4) }
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Indicates current condition of Disk-related statistics.
        DEGRADED indicates that one or more of the following
        conditions exists: a disk's usage exceeds the defined
        threshold, a disk's operations/second exceeds the defined
        threshold, a disk's status indicates a potential problem,
        or a shadow set member has a status of failed, catch-up
        copying, or merge-copy."
 ::= { sDsk 4 }

-- Shadow set information
--
-- The shadow table is a potentially sparse table of disk shadow information.
-- Each time shadow information is collected, new rows are added to the
-- bottom of the shadow table for new shadow sets that are discovered for the
-- corresponding disk. For shadow sets that are no longer available, the
-- corresponding row in the shadow table becomes inaccessible.
-- When the NM*SysMgr Subagent is reinitialized (see sCfgReinitSubagent),
-- any inaccessible rows in the shadow table are removed and the shadow
-- indexes and rows are re-numbered to be contiguous.

sShdTbl OBJECT-TYPE
    SYNTAX  SEQUENCE OF SShdEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "Table of shadow set information."

```

```
::= { sShd 1 }
```

sShdEntry OBJECT-TYPE

SYNTAX SShdEntry

ACCESS not-accessible

STATUS mandatory

DESCRIPTION

"A row in the shadow table. The combination of the sDskIndex and sShdIndex indices relates each shadow set table entry (sShdEntry) to its virtual unit entry in the disk table (sDskTbl)."

INDEX { sDskIndex, sShdIndex }

```
::= { sShdTbl 1 }
```

SShdEntry ::= SEQUENCE {

sShdIndex INTEGER,

sShdName DisplayString,

sShdFail INTEGER,

sShdCopy INTEGER,

sShdMerge INTEGER

}

sShdIndex OBJECT-TYPE

SYNTAX INTEGER

ACCESS read-only

STATUS mandatory

DESCRIPTION

"Location in the shadow set table."

```
::= { sShdEntry 1 }
```

sShdName OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..64))

ACCESS read-only

STATUS mandatory

DESCRIPTION

"Device name of the shadow set disk."

```
::= { sShdEntry 2 }
```

sShdFail OBJECT-TYPE

SYNTAX INTEGER { true(1), false(2) }

ACCESS read-only

STATUS mandatory

DESCRIPTION

"Indicator for shadow set member failure. A value of true(1) indicates that this member has failed out of the shadow set. "

```
::= { sShdEntry 3 }
```

sShdCopy OBJECT-TYPE

SYNTAX INTEGER { true(1), false(2) }

ACCESS read-only

STATUS mandatory

DESCRIPTION

"Indicator for shadow set catch-up copying. A value of true(1) indicates that this member is in a catch-up copy state."

```
::= { sShdEntry 4 }
```

```

sShdMerge OBJECT-TYPE
    SYNTAX  INTEGER { true(1), false(2) }
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Indicator for shadow set member merge.  A value of
        true(1) indicates that this member is in a merge
        state."
    ::= { sShdEntry 5 }

-- Queue information.  Names of queues to be monitored must be listed
-- in the sCfgQueFile queue configuration file.  The following MIB
-- information relates only to those queues listed in this file.
-- Refer to sCfgQueFile for information on updating the monitored
-- queue information.
sQueCnt OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of queues currently being monitored.  This
        number is the number of queue names found in the
        sCfgQueFile file."
    ::= { sQue 1 }

sQueTime OBJECT-TYPE
    SYNTAX  TimeTicks
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Subagent up time (sCfgUpTime) when queue statistics (sQTbl)
        were last gathered.  sCfgQueTimer controls how often queue
        statistics are gathered."
    ::= { sQue 2 }

sQTbl OBJECT-TYPE
    SYNTAX  SEQUENCE OF SQTblEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "Table of queue information."
    ::= { sQue 3 }

sQTblEntry OBJECT-TYPE
    SYNTAX  SQTblEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "A row in the queue table."
    INDEX   { sQIndex }
    ::= { sQTbl 1 }

SQTblEntry ::= SEQUENCE {
    sQIndex          INTEGER,
    sQName           DisplayString,
    sQMonitor        INTEGER,

```

```

        sQStatus          INTEGER,
        sQEntryCount      INTEGER,
        sQCritJobCount    INTEGER
    }

sQIndex OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Location in the queue table."
    ::= { sQTblEntry 1 }

sQName OBJECT-TYPE
    SYNTAX  DisplayString (SIZE (0..31))
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Name of the queue."
    ::= { sQTblEntry 2 }

sQMonitor OBJECT-TYPE
    SYNTAX  INTEGER { true(1), false(2) }
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
        "Indicates if the queue and its entries are currently being
        monitored.  A value of true(1) indicates that this queue
        and its entries are currently being monitored.  A value of
        false(2) indicates that monitoring is currently disabled on
        this queue and its entries."
    DEFVAL  { true }
    ::= { sQTblEntry 3 }

sQStatus OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Queue status indicates the current settings of the queue status
        flags.  This value is a bit vector which, when set, corresponds
        to the following characteristic:

        Bit      Description
        0         Queue is aligning
        1         Queue is idle:  no jobs executing and none available for
                   execution
        2         Lowercase device
        3         Queue is doing /OPERATOR (intentionally not documented)
        4         Queue is paused
        5         Queue is pausing
        6         Remote device
        7         Incomplete remote request to reset  (OBSOLETE)
        8         Queue is resuming from paused state
        9         /DEVICE=SERVER or reported to be a served queue by the
                   symbiont
        10        Device is stalled
    
```

```

11      Queue is starting
12      Queue is stopped
13      Queue is stopping
14      Device is unavailable
15      Queue is closed
16      Queue is executing as many jobs as it is capable of
      executing
17      Queue does not yet exist in db or is being removed
      from db
18      At least one job is executing, but the queue is
      capable of executing more
19      Queue is not able to accept or process jobs; its
      QSIs are inactive
20      AUTOSTART queue explicitly stopped
21      Queue will be stopped when work currently in
      progress has completed"
 ::= { sQTblEntry 4 }

sQEntryCount OBJECT-TYPE
  SYNTAX  INTEGER
  ACCESS  read-only
  STATUS  mandatory
  DESCRIPTION
    "Number of current queue entries. This number also
    identifies the number of rows in the sQEntryTbl for
    this queue."
 ::= { sQTblEntry 5 }

sQCritJobCount OBJECT-TYPE
  SYNTAX  INTEGER
  ACCESS  read-only
  STATUS  mandatory
  DESCRIPTION
    "Number of critical jobs for this queue. This number also
    identifies the number of rows in the sCritJobTbl for
    this queue."
 ::= { sQTblEntry 6 }

sQEntryTbl OBJECT-TYPE
  SYNTAX  SEQUENCE OF SQEntry
  ACCESS  not-accessible
  STATUS  mandatory
  DESCRIPTION
    "Table of queue entry information."
 ::= { sQue 4 }

sQEntry OBJECT-TYPE
  SYNTAX  SQEntry
  ACCESS  not-accessible
  STATUS  mandatory
  DESCRIPTION
    "A row in the queue entry table."
  INDEX   { sQIndex, sQEntryNum }
 ::= { sQEntryTbl 1 }

SQEntry ::= SEQUENCE {
  sQEntryNum          INTEGER,

```

```

        sQEntryJobname          DisplayString,
        sQEntryJobStatus        INTEGER
    }

sQEntryNum OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The queue entry number of this job."
    ::= { sQEntry 1 }

sQEntryJobname OBJECT-TYPE
    SYNTAX  DisplayString (SIZE (0..39))
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The job name of this queue entry."
    ::= { sQEntry 2 }

sQEntryJobStatus OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Queue status indicates the current settings of the queue status
        flags. This value is a bit vector which, when set, corresponds
        to the following characteristic:

        Bit      Description
        0        Job is aborting
        1        Job is executing
        2        Job is holding for /HOLD
        3        Job is inaccessible
        4        Job was refused by symbiont
        5        Job will requeue after abort (intentionally not
        documented)
        6        Job started execution at least once
        (intentionally not documented)
        7        Job was retained by /RETAIN
        8        Job is starting
        9        Job is holding for /AFTER
        10       Job is suspended by STOP/QUEUE command
        11       Job is pending
        12       Job does not yet exist in db or is being removed from db
        13       Current job on stalled queue
        14       Job was processed by the symbiont but further
        processing is necessary
        15       Job has been completely processed"
    ::= { sQEntry 3 }

sQueBatch OBJECT IDENTIFIER ::= { sQue 5 }

sQueBatchPending OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory

```

```

DESCRIPTION
    "Number of pending jobs in the batch queue."
 ::= { sQueBatch 1 }

sQueBatchExecuting OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of executing jobs in the batch queue."
 ::= { sQueBatch 2 }

sQueBatchTimed OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of timed jobs in the batch queue."
 ::= { sQueBatch 3 }

sQueBatchHolding OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of holding jobs in the batch queue."
 ::= { sQueBatch 4 }

sQueBatchRetained OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of retained jobs in the batch queue."
 ::= { sQueBatch 5 }

sQueGeneric OBJECT IDENTIFIER ::= { sQue 6 }

sQueGenericPending OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of pending jobs in generic queues."
 ::= { sQueGeneric 1 }

sQueGenericExecuting OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of executing jobs in generic queues."
 ::= { sQueGeneric 2 }

sQueGenericTimed OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only

```

STATUS mandatory
DESCRIPTION
"Number of timed jobs in generic queues."
::= { sQueGeneric 3 }

sQueGenericHolding OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Number of holding jobs in generic queues."
::= { sQueGeneric 4 }

sQueGenericRetained OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Number of retained jobs in generic queues."
::= { sQueGeneric 5 }

sQuePrinter OBJECT IDENTIFIER ::= { sQue 7 }

sQuePrinterPending OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Number of pending jobs in printer queues."
::= { sQuePrinter 1 }

sQuePrinterExecuting OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Number of executing jobs in printer queues."
::= { sQuePrinter 2 }

sQuePrinterTimed OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Number of timed jobs in printer queues."
::= { sQuePrinter 3 }

sQuePrinterHolding OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
"Number of holding jobs in printer queues."
::= { sQuePrinter 4 }

sQuePrinterRetained OBJECT-TYPE
SYNTAX INTEGER

```
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Number of retained jobs in printer queues."
 ::= { sQuePrinter 5 }

sQueServer OBJECT IDENTIFIER ::= { sQue 8 }

sQueServerPending OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Number of pending jobs in server queues."
 ::= { sQueServer 1 }

sQueServerExecuting OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Number of executing jobs in server queues."
 ::= { sQueServer 2 }

sQueServerTimed OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Number of timed jobs in server queues."
 ::= { sQueServer 3 }

sQueServerHolding OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Number of holding jobs in server queues."
 ::= { sQueServer 4 }

sQueServerRetained OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Number of retained jobs in server queues."
 ::= { sQueServer 5 }

sQueSymbiont OBJECT IDENTIFIER ::= { sQue 9 }

sQueSymbiontPending OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Number of pending jobs in symbiont queues."
 ::= { sQueSymbiont 1 }
```

sQueSymbiontExecuting OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "Number of executing jobs in symbiont queues."
 ::= { sQueSymbiont 2 }

sQueSymbiontTimed OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "Number of timed jobs in symbiont queues."
 ::= { sQueSymbiont 3 }

sQueSymbiontHolding OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "Number of holding jobs in symbiont queues."
 ::= { sQueSymbiont 4 }

sQueSymbiontRetained OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "Number of retained jobs in symbiont queues."
 ::= { sQueSymbiont 5 }

sQueTerminal OBJECT IDENTIFIER ::= { sQue 10 }

sQueTerminalPending OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "Number of pending jobs in terminal queues."
 ::= { sQueTerminal 1 }

sQueTerminalExecuting OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "Number of executing jobs in terminal queues."
 ::= { sQueTerminal 2 }

sQueTerminalTimed OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "Number of timed jobs in terminal queues."

```

 ::= { sQueTerminal 3 }

sQueTerminalHolding OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of holding jobs in terminal queues."
 ::= { sQueTerminal 4 }

sQueTerminalRetained OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of retained jobs in terminal queues."
 ::= { sQueTerminal 5 }

SQMan OBJECT IDENTIFIER ::= { sQue 11 }

SQManCount OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of entries in the queue manager table."
 ::= { sQMan 1 }

SQManTbl OBJECT-TYPE
    SYNTAX  SEQUENCE OF SQManTblEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "Table of queue manager information."
 ::= { sQMan 2 }

SQManTblEntry OBJECT-TYPE
    SYNTAX  SQManTblEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "A row in the queue manager table."
    INDEX   { sQManIndex }
 ::= { sQManTbl 1 }

SQManTblEntry ::= SEQUENCE {
    sQManIndex          INTEGER,
    sQManName           DisplayString,
    sQManStatus         INTEGER
}

sQManIndex OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Location in the queue manager table."

```

```
::= { sQManTblEntry 1 }
```

sQManName OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..31))

ACCESS read-only

STATUS mandatory

DESCRIPTION

"Name of the queue manager."

```
::= { sQManTblEntry 2 }
```

sQManStatus OBJECT-TYPE

SYNTAX INTEGER

ACCESS read-only

STATUS mandatory

DESCRIPTION

"Queue manager status indicates the current settings of the queue manager status flags. This value is a bit vector which, when set, corresponds to the following characteristic:

| Bit | Description |
|-----|---------------|
| 0 | Start pending |
| 1 | Starting |
| 2 | Running |
| 3 | Failover |
| 4 | Stopping |
| 5 | Stopped" |

```
::= { sQManTblEntry 3 }
```

sQCondition OBJECT-TYPE

SYNTAX INTEGER { unknown(1), ok(2), degraded(3), failed(4) }

ACCESS read-only

STATUS mandatory

DESCRIPTION

"Indicates current condition of Queue-related statistics. DEGRADED indicates that either status or a queue entry status indicates a potential problem."

```
::= { sQue 12 }
```

sCritJobCondition OBJECT-TYPE

SYNTAX INTEGER { unknown(1), ok(2), degraded(3), failed(4) }

ACCESS read-only

STATUS mandatory

DESCRIPTION

"Indicates current condition of critical jobs. DEGRADED indicates that at least one critical job is missing from its specified queue."

```
::= { sQue 13 }
```

sCritJobTbl OBJECT-TYPE

SYNTAX SEQUENCE OF SCritJobEntry

ACCESS not-accessible

STATUS mandatory

DESCRIPTION

"Table of critical job information."

```
::= { sQue 14 }
```

sCritJobEntry OBJECT-TYPE

```

SYNTAX   SCritJobEntry
ACCESS   not-accessible
STATUS   mandatory
DESCRIPTION
        "A row in the critical job table."
INDEX    { sQIndex, sCritJobNum }
::= { sCritJobTbl 1 }

SCritJobEntry ::= SEQUENCE {
    sCritJobNum          INTEGER,
    sCritJobName         DisplayString,
    sCritJobFound        INTEGER
}

sCritJobNum OBJECT-TYPE
    SYNTAX   INTEGER
    ACCESS   read-only
    STATUS   mandatory
    DESCRIPTION
        "The row number for the critical job table for this
        queue."
    ::= { sCritJobEntry 1 }

sCritJobName OBJECT-TYPE
    SYNTAX   DisplayString (SIZE (0..39))
    ACCESS   read-only
    STATUS   mandatory
    DESCRIPTION
        "The job name of the critical job."
    ::= { sCritJobEntry 2 }

sCritJobFound OBJECT-TYPE
    SYNTAX   INTEGER { true(1), false(2) }
    ACCESS   read-only
    STATUS   mandatory
    DESCRIPTION
        "Indicates if this job is present in this queue."
    ::= { sCritJobEntry 3 }

-- Process statistics for all processes currently running on the system
sPSCnt OBJECT-TYPE
    SYNTAX   INTEGER
    ACCESS   read-only
    STATUS   mandatory
    DESCRIPTION
        "Number of processes currently running of the system."
    ::= { sProcInfo 1 }

sPSTime OBJECT-TYPE
    SYNTAX   TimeTicks
    ACCESS   read-only
    STATUS   mandatory
    DESCRIPTION
        "Subagent up time (sCfGUpTime) when process statistics (sPsTbl)
        were last gathered. sCfGPsTimer controls how often process
        statistics are gathered."
    ::= { sProcInfo 2 }

```

```

sPsTbl OBJECT-TYPE
    SYNTAX SEQUENCE OF SPsEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "Table of process information."
    ::= { sProcInfo 3 }

sPsEntry OBJECT-TYPE
    SYNTAX SPsEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION
        "A row in the process table."
    INDEX { sPsPID }
    ::= { sPsTbl 1 }

SPsEntry ::= SEQUENCE {
    sPsPID                INTEGER,
    sPsProcName           DisplayString,
    sPsState              INTEGER,
    sPsPriority            INTEGER,
    sPsDirectIO           INTEGER,
    sPsCpuTime            TimeTicks,
    sPsPgFaults           INTEGER,
    sPsWorkSetSize        INTEGER,
    sPsUsername           DisplayString,
    sPsPhysTerm           DisplayString,
    sPsImageName          DisplayString,
    sPsLoginTime          DisplayString,
    sPsPgTblCnt           INTEGER,
    sPsMode               INTEGER,
    sPsRWState            INTEGER
}

sPsPID OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The VMS process identification (PID) of the process."
    ::= { sPsEntry 1 }

sPsProcName OBJECT-TYPE
    SYNTAX DisplayString (SIZE (0..15))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The process name."
    ::= { sPsEntry 2 }

sPsState OBJECT-TYPE
    SYNTAX INTEGER {
        colpg(1),
        mwait(2),
        cef(3),

```

```

        pfw(4),
        lef(5),
        lefo(6),
        hib(7),
        hibo(8),
        susp(9),
        suspo(10),
        fpg(11),
        com(12),
        como(13),
        cur(14) }
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The state of the process."
 ::= { sPsEntry 3 }

sPsPriority OBJECT-TYPE
SYNTAX INTEGER (0..31)
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The current priority of the process."
 ::= { sPsEntry 4 }

sPsDirectIO OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The count of the direct I/O operations performed by this
    process. If sCfgAllPsData is set to no(2), this value may
    be zero."
 ::= { sPsEntry 5 }

sPsCpuTime OBJECT-TYPE
SYNTAX TimeTicks
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The process' accumulated CPU time. If sCfgAllPsData is set
    to no(2), this value may be zero."
 ::= { sPsEntry 6 }

sPsPgFaults OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "The total number of page faults incurred by the process.
    If sCfgAllPsData is set to no(2), this value may be zero."
 ::= { sPsEntry 7 }

sPsWorkSetSize OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory

```

DESCRIPTION

"The current working set size (in pages) of the process.
If sCfgAllPsData is set to no(2), this value may be zero."

::= { sPsEntry 8 }

sPsUsername OBJECT-TYPE

SYNTAX DisplayString (SIZE (12))

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The owner username of the process. This string is always
blank padded to a length of 12 characters."

::= { sPsEntry 9 }

sPsPhysTerm OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..8))

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The physical device name of the terminal associated with
the process."

::= { sPsEntry 10 }

sPsImageName OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..255))

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The directory specification and image file name. If
sCfgAllPsData is set to no(2), this value may be zero."

::= { sPsEntry 11 }

sPsLoginTime OBJECT-TYPE

SYNTAX DisplayString

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The time at which the process was created. This value is
in dd-mmm-yyyy hh:mm:ss.cc format. If sCfgAllPsData is set
to no(2), this value may be zero."

::= { sPsEntry 12 }

sPsPgTblCnt OBJECT-TYPE

SYNTAX INTEGER

ACCESS read-only

STATUS mandatory

DESCRIPTION

"The number of pages the process has in its working set."

::= { sPsEntry 13 }

sPsMode OBJECT-TYPE

SYNTAX INTEGER {
other(1),
network(2),
batch(3),
interactive(4) }

ACCESS read-only

STATUS mandatory
DESCRIPTION
 "The mode of the process."
::= { sPsEntry 14 }

sPsRWState OBJECT-TYPE

SYNTAX INTEGER {
 astwait(1),
 mailbox(2),
 npdynmem(3),
 pgfile(4),
 pgdynmem(5),
 mplempty(11),
 mpwbusy(12),
 scs(13),
 clustran(14),
 cpucap(15),
 clusrv(16),
 psxfrk(18),
 mutex(19) }

ACCESS read-only

STATUS mandatory

DESCRIPTION

 "The resource wait state of the process. This variable is only meaningful when a process is in the MWAIT state (i.e., when sPsState is set to MWAIT(2)), otherwise this variable contains no data. The above listed resource wait states have the following meanings:

 AST - AST wait
 MBX - Mailbox full
 NPG - Nonpaged dynamic memory
 PFF - Page file full
 PAG - Paged dynamic memory
 MPE - Modified page list empty
 MPB - Modified page writer busy
 SCS - Distributed lock manager wait
 CLU - VMScluster transition
 CAP - CPU capability
 CSV - VMScluster server process
 PSXFRK - POSIX fork creation
 MUTEX - Locked resource"

::= { sPsEntry 15 }

sPsCOMQueue OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION

 "Number of processes that are waiting to compute, i.e., the number of processes with a sPsState of COM."

::= { sProcInfo 4 }

sPsCOMOQueue OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION

```
        "Number of outswapped processes that are waiting to compute,  
        i.e., the number of processes with a sPsState of COMO."  
 ::= { sProcInfo 5 }
```

sPsOther OBJECT-TYPE

```
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "Count of other mode processes i.e., those processes  
    that are not network, batch or interactive."  
 ::= { sProcInfo 6 }
```

sPsNetwork OBJECT-TYPE

```
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "Count of network processes."  
 ::= { sProcInfo 7 }
```

sPsBatch OBJECT-TYPE

```
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "Count of batch processes."  
 ::= { sProcInfo 8 }
```

sPsInteractive OBJECT-TYPE

```
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "Count of Interactive processes."  
 ::= { sProcInfo 9 }
```

sPsAvailProcSlots OBJECT-TYPE

```
SYNTAX Counter  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "Count of Available Process Slots."  
 ::= { sProcInfo 10 }
```

sPsCondition OBJECT-TYPE

```
SYNTAX INTEGER { unknown(1), ok(2), degraded(3), failed(4) }  
ACCESS read-only  
STATUS mandatory  
DESCRIPTION  
    "Indicates current condition of Process-related statistics.  
    DEGRADED indicates that one or more of the following  
    conditions exist: too many processes in COM state, too  
    many processes in COMO state, one or more processes in  
    MWAIT state."  
 ::= { sProcInfo 11 }
```

```

sPsmMwaitCount OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The number of processes currently in Mwait state."
    ::= { sProcInfo 12 }

-- Critical process table
-- This table reflects the current status of the critical processes
-- contained in the critical process file specified in sCfgCritFile
sCritCnt OBJECT-TYPE
    SYNTAX  INTEGER (0..100)
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of critical process table entries."
    ::= { sCritInfo 1 }

sCritTime OBJECT-TYPE
    SYNTAX  TimeTicks
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Subagent up time (sCfgUpTime) when critical process
        statistics (sCritTbl) were last gathered. sCfgCritTimer
        specifies how often critical process statistics are gathered."
    ::= { sCritInfo 2 }

sCritTbl OBJECT-TYPE
    SYNTAX  SEQUENCE OF SCritEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "Table of critical process information."
    ::= { sCritInfo 3 }

sCritEntry OBJECT-TYPE
    SYNTAX  SCritEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "A row in the critical process table."
    INDEX  { sCritIndex }
    ::= { sCritTbl 1 }

SCritEntry ::= SEQUENCE {
    sCritIndex      INTEGER,
    sCritName       DisplayString,
    sCritReqCnt     INTEGER,
    sCritCurCnt    INTEGER,
    sCritBufIO      Counter,
    sCritCpuTime    Counter,
    sCritDirIO      Counter,
    sCritPageFlts   Counter
}

```

```

sCritIndex OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Location in the critical process table."
    ::= { sCritEntry 1 }

sCritName OBJECT-TYPE
    SYNTAX  DisplayString (SIZE (0..15))
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Name of a critical process.  Critical process names
        may contain * and % wildcard characters."
    ::= { sCritEntry 2 }

sCritReqCnt OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Minimum number of copies of this process required to be
        running on the system."
    ::= { sCritEntry 3 }

sCritCurCnt OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The current number of copies of this process running on
        the system."
    ::= { sCritEntry 4 }

sCritBufIO OBJECT-TYPE
    SYNTAX  Counter
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The total buffered I/O for all instances of this process."
    ::= { sCritEntry 5 }

sCritCpuTime OBJECT-TYPE
    SYNTAX  Counter
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The total CPU time for all instances of this process."
    ::= { sCritEntry 6 }

sCritDirIO OBJECT-TYPE
    SYNTAX  Counter
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The total direct I/O for all instances of this process."

```

```

        the system."
 ::= { sCritEntry 7 }

sCritPageFlts OBJECT-TYPE
    SYNTAX Counter
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The total pages faults for all instances of this process."
 ::= { sCritEntry 8 }

sCritCondition OBJECT-TYPE
    SYNTAX INTEGER { unknown(1), ok(2), degraded(3), failed(4) }
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Indicates current condition of Critical Process statistics.
        DEGRADED indicates that at least one critical process
        is missing."
 ::= { sCritInfo 4 }

-- Trap Information
-- The NM*SysMgr Subagent process maintains an internal trap table
-- containing the latest traps that have been sent by the subagent.
-- The number of traps that are retained in the internal trap table
-- is controlled by the sCfgTrapTblSize variable. Each trap that is
-- sent by NM*SysMgr includes a trap sequence number and a timestamp
-- as well as any trap specific data. The trap sequence number may
-- be used to request that a trap be resent to the manager.
--
-- The number of traps per second that may be sent is governed by the
-- sCfgMaxTrapSec variable. sTrapLastSeqNumSent identifies the
-- sequence number of the last trap that was actually sent, if no
-- backlog of traps exists, this number will be one less than
-- sTrapNextSeqNum. If a backlog exists due to throttling, then
-- the size of the backlog may be determined by comparing these
-- numbers. If the size of the backlog exceeds the size of the
-- trap table (as set by sCfgTrapTblSize), the backlogged traps
-- will be lost as they are overwritten by new traps, the number
-- of traps that have been lost is contained in sTrapLostCount.

sTrapNextSeqNum OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The next sequence number in the trap table. When this
        variable is sent as part of a trap, it identifies the
        sequence number assigned to that trap. This is a 32-bit
        value and will wrap after 4294967295."
 ::= { sTrapInfo 1 }

sTrapTime OBJECT-TYPE
    SYNTAX TimeTicks
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION

```

"Subagent up time (sCfgUpTime) when the trap was created."
 ::= { sTrapInfo 2 }

sTrapResendSeqNum OBJECT-TYPE

SYNTAX INTEGER

ACCESS read-write

STATUS mandatory

DESCRIPTION

"Sequence number of a trap message to be resent to the manager. Setting this variable to a trap sequence number causes the specified trap message to be resent. If the set-request contains a number that is not available (the number of items retained in the trap table is dependent on the value of sCfgTrapTblSize), an error indicator will be returned."

::= { sTrapInfo 3 }

sTrapLastSeqNumSent OBJECT-TYPE

SYNTAX INTEGER

ACCESS read-only

STATUS mandatory

DESCRIPTION

"Sequence number of the last trap sent. This value may be compared to sTrapNextSeqNum to determine if a backlog of traps exists. If this value is one less than sTrapNextSeqNum, then no backlog exists. A backlog of traps may exist if more traps are generated per second than may be sent (as specified by the variable sCfgMaxTrapSec)."

::= { sTrapInfo 4 }

sTrapLostCount OBJECT-TYPE

SYNTAX Counter

ACCESS read-only

STATUS mandatory

DESCRIPTION

"This variable indicates the number of traps that have been lost. A trap is considered to be lost if it is overwritten in the internal trap table before it is sent to the NMS. This may occur if the trap table is not large enough (see sCfgTrapTblSize) to hold all of unsent traps. How quickly traps are sent to the NMS is governed by the sCfgMaxTrapSec variable. To prevent trap loss, set these two variables so that the trap backlog does not overflow the trap table and so that traps do not flood the network."

::= { sTrapInfo 5 }

sTrapTextTbl OBJECT-TYPE

SYNTAX SEQUENCE OF STrapTextEntry

ACCESS not-accessible

STATUS mandatory

DESCRIPTION

"Table of textual trap messages."

::= { sTrapInfo 6 }

sTrapTextEntry OBJECT-TYPE

SYNTAX STrapTextEntry

ACCESS not-accessible

STATUS mandatory
DESCRIPTION
 "A row in the trap text table."
INDEX { sTrapTextSeqNum }
::= { sTrapTextTbl 1 }

sTrapTextEntry ::= SEQUENCE {
 sTrapTextSeqNum INTEGER,
 sTrapTimeStamp DisplayString,
 sTrapType INTEGER,
 sTrapText1 DisplayString,
 sTrapText2 DisplayString,
 sTrapText3 DisplayString,
 sTrapText4 DisplayString,
 sTrapText5 DisplayString,
 sTrapText6 DisplayString,
 sTrapText7 DisplayString
}

sTrapTextSeqNum OBJECT-TYPE
SYNTAX INTEGER
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "Sequence number of the trap."
::= { sTrapTextEntry 1 }

sTrapTimeStamp OBJECT-TYPE
SYNTAX DisplayString
ACCESS read-only
STATUS mandatory
DESCRIPTION
 "System time when trap occurred."
::= { sTrapTextEntry 2 }

sTrapType OBJECT-TYPE
SYNTAX INTEGER {
 coldStart(1),
 error(2),
 hardwareError(3),
 cpuUsage(4),
 intUsage(5),
 diskBusy(6),
 diskFull(7),
 diskFullClear(8),
 swapFull(9),
 swapFullClear(10),
 pageFull(11),
 pageFullClear(12),
 criticalProcess(13),
 opcom(14),
 terminate(15),
 diskStatus(16),
 warmStart(17),
 shadowFail(18),
 shadowCopy(19),
 shadowMerge(20),

```

        queueStatus(21),
        jobStatus(22),
        processStatus(23),
        memoryFull(24),
        memoryFullClear(25),
        comQueue(26),
        comoQueue(27),
        critProcClear(28),
        dskIOQLen(29),
        jobMissing(30),
        mwaitLimit(31) }
ACCESS   read-only
STATUS   mandatory
DESCRIPTION
        "Type of trap."
::= { sTrapTextEntry 3 }

sTrapText1 OBJECT-TYPE
SYNTAX   DisplayString
ACCESS   read-only
STATUS   mandatory
DESCRIPTION
        "Part 1 of the trap text."
::= { sTrapTextEntry 4 }

sTrapText2 OBJECT-TYPE
SYNTAX   DisplayString
ACCESS   read-only
STATUS   mandatory
DESCRIPTION
        "Part 2 of the trap text."
::= { sTrapTextEntry 5 }

sTrapText3 OBJECT-TYPE
SYNTAX   DisplayString
ACCESS   read-only
STATUS   mandatory
DESCRIPTION
        "Part 3 of the trap text."
::= { sTrapTextEntry 6 }

sTrapText4 OBJECT-TYPE
SYNTAX   DisplayString
ACCESS   read-only
STATUS   mandatory
DESCRIPTION
        "Part 4 of the trap text."
::= { sTrapTextEntry 7 }

sTrapText5 OBJECT-TYPE
SYNTAX   DisplayString
ACCESS   read-only
STATUS   mandatory
DESCRIPTION
        "Part 5 of the trap text."
::= { sTrapTextEntry 8 }

```

```

sTrapText6 OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Part 6 of the trap text."
    ::= { sTrapTextEntry 9 }

sTrapText7 OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Part 7 of the trap text."
    ::= { sTrapTextEntry 10 }

-- Hardware error information for disks, tapes, buses, CPU, and memory
--
-- The hardware error table is a potentially sparse table of device errors.
-- Each time hardware error information is collected, new rows are added
-- to the bottom of the hardware error table for devices that had not
-- previously reported errors. For devices that are no longer available,
-- the corresponding row in the hardware error table becomes inaccessible.
-- This allows the hardware error table to be walked to retrieve only the
-- current error information while ensuring that a hardware error table
-- index (and therefore the row in the hardware error table) relates to a
-- specific device. When the NM*SysMgr Subagent is reinitialized
-- (see sCfgReinitSubagent), any inaccessible rows in the hardware error
-- table are removed and the hardware error table indexes and rows are
-- re-numbered to be contiguous.
--
-- Note: Disk errors for shadowed disks are reported by physical shadow
-- set member, they are not reported for the shadow set virtual unit.

sHwErrDeviceCnt OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Number of devices reporting errors."
    ::= { sHwErrInfo 1 }

sHwErrTime OBJECT-TYPE
    SYNTAX TimeTicks
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Subagent up time (sCfgUpTime) when hardware error statistics
        were last gathered."
    ::= { sHwErrInfo 2 }

sHwErrTbl OBJECT-TYPE
    SYNTAX SEQUENCE OF SHwErrEntry
    ACCESS not-accessible
    STATUS mandatory
    DESCRIPTION

```

```

        "Table of hardware errors."
 ::= { sHwErrInfo 3 }

sHwErrEntry OBJECT-TYPE
    SYNTAX  SHwErrEntry
    ACCESS  not-accessible
    STATUS  mandatory
    DESCRIPTION
        "A row in the hardware error table."
    INDEX   { sHwErrIndex }
    ::= { sHwErrTbl 1 }

SHwErrEntry ::= SEQUENCE {
    sHwErrIndex      INTEGER,
    sHwErrDeviceName DisplayString,
    sHwErrCnt        INTEGER,
    sHwErrLastTime   TimeTicks
}

sHwErrIndex OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Location in the hardware error table."
    ::= { sHwErrEntry 1 }

sHwErrDeviceName OBJECT-TYPE
    SYNTAX  DisplayString
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Name of the device."
    ::= { sHwErrEntry 2 }

sHwErrCnt OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "Number of errors reported by the device since last reboot
        of the system."
    ::= { sHwErrEntry 3 }

sHwErrLastTime OBJECT-TYPE
    SYNTAX  TimeTicks
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "sCfgUpTime when the latest error was detected by the
        NM*SysMgr Subagent."
    ::= { sHwErrEntry 4 }

-- When a software error is encountered, a sSwErr trap is generated containing
-- the following software error description items.  Invalid sets of the
-- sCfgFile or sCfgCritFile variable will cause a sSwErr trap to be generated
-- to explain the error.

```

```

sSwErrStatus OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The latest software status code."
    ::= { sSwErrInfo 1 }

sSwErrFile OBJECT-TYPE
    SYNTAX DisplayString (SIZE (0..255))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The file name of the software element which generated the
        latest software status code."
    ::= { sSwErrInfo 2 }

sSwErrLineNum OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The line number in the file which generated the latest
        software status code."
    ::= { sSwErrInfo 3 }

sSwErrLastTime OBJECT-TYPE
    SYNTAX TimeTicks
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "sCfgUpTime when the most recent software status code
        was generated."
    ::= { sSwErrInfo 4 }

sSwErrMessage OBJECT-TYPE
    SYNTAX DisplayString
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Description of the latest software error."
    ::= { sSwErrInfo 5 }

-- Operator (OpCom) information
sOpcomOne OBJECT-TYPE
    SYNTAX DisplayString (SIZE (1..255))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "The first 255 characters of the text of the most recent
        message in the operator log. For opcom messages that
        exceed 255 characters, the remainder of the message will be
        contained in sOpcomTwo, sOpcomThree, sOpcomFour, sOpcomFive,
        sOpcomSix, and sOpcomSeven depending on the length of the
        message.

        All operator log messages are sent as sOpcomMessage traps and

```

```

        retained in the internal trap table. To retrieve a previous
        message, resend the trap (see sTrapResendSeqNum).
 ::= { sErrInfo 3 }

sOpcomTwo OBJECT-TYPE
    SYNTAX DisplayString (SIZE (1..255))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Part 2 of an opcom message."
 ::= { sErrInfo 4 }

sOpcomThree OBJECT-TYPE
    SYNTAX DisplayString (SIZE (1..255))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Part 3 of an opcom message."
 ::= { sErrInfo 5 }

sOpcomFour OBJECT-TYPE
    SYNTAX DisplayString (SIZE (1..255))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Part 4 of an opcom message."
 ::= { sErrInfo 6 }

sOpcomFive OBJECT-TYPE
    SYNTAX DisplayString (SIZE (1..255))
    ACCESS read-only
    STATUS mandatory
    DESCRIPTION
        "Part 5 of an opcom message."
 ::= { sErrInfo 7 }

-- The following sOpcom variables may be used to send a reply to an
-- operator request.
sOpcomReplyId OBJECT-TYPE
    SYNTAX INTEGER
    ACCESS read-write
    STATUS mandatory
    DESCRIPTION
        "Request number to be associated with the operator reply.
        This number identifies a previous Opcom request. This
        variable is reset to zero when a reply is successfully sent.
        (See sOpcomReplySend.)"
    DEFVAL { 0 }
 ::= { sErrInfo 8 }

sOpcomReplyStatus OBJECT-TYPE
    SYNTAX INTEGER {
        noStatus(1),
        blankTape(2),
        initializeTape(3),
        requestComplete(4),
        requestPending(5),

```

```

        requestAborted(6)
    }
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Completion status to accompany opcom reply. This variable
    is reset to noStatus when a reply is successfully sent.
    (See sOpcomReplySend.)"
DEFVAL { noStatus }
::= { sErrInfo 9 }

sOpcomReplyText OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..255))
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Text to be sent for an operator reply message. This
    variable is reset to null when a reply is successfully sent.
    (See sOpcomReplySend.)"
DEFVAL { "" }
::= { sErrInfo 10 }

sOpcomReplySend OBJECT-TYPE
SYNTAX INTEGER { true(1), false(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Flag which is used to trigger sending of an operator reply
    message. To send an operator reply, set the sOpcomReplyId,
    sOpcomReplyStatus, and sOpcomReplyText variables and then
    set the sOpcomReplySend to true. All opcom reply variables
    will be reset following a successful send."
DEFVAL { false }
::= { sErrInfo 11 }

sOpcomSix OBJECT-TYPE
SYNTAX DisplayString (SIZE (1..255))
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Part 6 of an opcom message."
::= { sErrInfo 12 }

sOpcomSeven OBJECT-TYPE
SYNTAX DisplayString (SIZE (1..255))
ACCESS read-only
STATUS mandatory
DESCRIPTION
    "Part 7 of an opcom message."
::= { sErrInfo 13 }

-- Configuration parameters which are read from disk and may be
-- viewed/changed by the network manager. All changes to these
-- parameters are written to the disk file.
sCfgFile OBJECT-TYPE
SYNTAX DisplayString (SIZE (0..255))
ACCESS read-write

```

STATUS mandatory

DESCRIPTION

"Name of the file currently being used for NM*SysMgr initialization parameters. This file is identified by the logical name COMTEK\$SYSMGR_CONFIG. Changing this variable redefines the COMTEK\$SYSMGR_CONFIG logical name. NM*SysMgr must be reinitialized (by setting the variable sCfgReinitSubagent to true) before the new configuration file is used. If a set-request is performed on this variable and the specified file name does not exist, is not accessible by the subagent, or contains invalid data, the file name is not changed, and an error is returned, and a sSwErr trap is generated describing the error.

Set-requests: Take effect only after reinitialization
(sCfgReinitSubagent)

NOTE: To make changes to this value permanent across system reboots, the definition of this logical name must also be changed in the system startup procedures."

::= { sCfg 1 }

sCfgLogFile OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..255))

ACCESS read-only

STATUS mandatory

DESCRIPTION

"Name of the log file currently being used by NM*SysMgr. If this process is run interactively, this variable will have the value SYS\$OUTPUT: (standard output)."

::= { sCfg 2 }

sCfgCritFile OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..255))

ACCESS read-write

STATUS mandatory

DESCRIPTION

"Name of the file currently being used to identify critical processes. This file is associated with the logical name COMTEK\$SYSMGR_CRITPROC. Changing this variable redefines the COMTEK\$SYSMGR_CRITPROC logical name. NM*SysMgr must be reinitialized (by setting the variable sCfgReinitSubagent to true) before the new critical process file is used. If a set-request is performed on this variable and the specified file name does not exist, is not accessible by the subagent, or contains invalid data, the file name is not changed, an error is returned, and a sSwErr trap is generated describing the error.

Set-requests: Take effect only after reinitialization
(sCfgReinitSubagent)

NOTE: To make changes to this value permanent across system reboots, the definition of this logical name must also be changed in the system startup procedures."

::= { sCfg 3 }

sCfgReinitSubagent OBJECT-TYPE
SYNTAX INTEGER { true(1), false(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Flag which may be set to force the NM*SysMgr Subagent to reinitialize. Reinitialization involves resetting the process configuration using the parameters contained in the sCfgFile and resetting the critical process, critical disk, and critical queue data as specified in the sCfgCritFile, sCfgCritDsk, and sCfgCritQue files and sending the sWarmStart trap."
DEFVAL { false }
 ::= { sCfg 4 }

sCfgTraps OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Flag which determines if any trap messages are sent to the NM*Master Agent. Setting this variable to disabled(2) disables all traps from being sent."

Configuration Variable: Traps
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
 ::= { sCfg 5 }

sCfgHwErrTraps OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Flag which determines if hardware error traps (sHwErr) should be sent. Setting this variable to enabled(1) causes a trap to be sent for every error detected. Setting this variable to disabled(2) disables hardware error traps from being sent."

Configuration Variable: HwErrTrap
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
 ::= { sCfg 6 }

sCfgCpuLimit OBJECT-TYPE
SYNTAX INTEGER (0..100)
ACCESS read-write
STATUS mandatory
DESCRIPTION
"The percentage of CPU utilization at which excessive CPU utilization (sCpuUsage) traps begin to be sent. A value of 0 disables these traps."

The variable sCfgCpuPersist can be used to specify the number of consecutive minutes the CPU utilization threshold must be exceeded before a trap is sent.

Configuration Variable: CpuUtil
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."

DEFVAL { 90 }
::= { sCfg 7 }

sCfgIntLimit OBJECT-TYPE

SYNTAX INTEGER (0..100)
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The percentage of interrupt CPU utilization at which excessive interrupt utilization (sIntUsage) traps begin to be sent. A value of 0 disables these traps.

Configuration Variable: IntUtil
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."

DEFVAL { 20 }
::= { sCfg 8 }

sCfgDskOps OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The threshold at which excessive disk busy traps (sDskOpSec) begin to be sent. This value is expressed in terms of disk operations (reads and writes) per second. At expiration of each sCfgDskTimer interval, the number of disk operations during the intervening interval is computed. A trap is sent if the number of operations during that interval exceeds the this limit. A value of 0 disables disk busy traps. Different thresholds may be set on individual disks by specifying an OpSec limit in the critical disk file.

Configuration Variable: DskOps
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."

DEFVAL { 25 }
::= { sCfg 9 }

sCfgDskLimit OBJECT-TYPE

SYNTAX INTEGER (0..100)
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The percentage of disk space utilization at which excessive disk utilization (sDskFull) traps begin to be sent. If this

variable is set to a value other than zero, it is used to determine when to send sDskFull traps for disks not specified in the critical disk table (see sCfgCritDsk).

This value and sCfgDskMinFreeBlks are mutually exclusive, i.e., only one of these values is in effect at any given time. If neither value is specified in the configuration file, then the default value of sCfgDskLimit is used for disk monitoring and sCfgDskMinFreeBlks is set to zero. To disable default disk full monitoring, both of these values must be zero.

Set-requests to this value will fail unless sCfgDskMinFreeBlks is set to zero.

Configuration Variable: DskUse
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."

DEFVAL { 90 }
::= { sCfg 10 }

sCfgPgLimit OBJECT-TYPE

SYNTAX INTEGER (0..100)
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The percentage of page file utilization at which excessive page file utilization (sPgFull) traps begin to be sent. A value of 0 disables these traps.

Configuration Variable: PgUse
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."

DEFVAL { 90 }
::= { sCfg 11 }

sCfgSwpLimit OBJECT-TYPE

SYNTAX INTEGER (0..100)
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The percentage of swap file utilization at which excessive swap file utilization (sSwpFull) traps begin to be sent. A value of 0 disables these traps.

Configuration Variable: SwpUse
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."

DEFVAL { 90 }
::= { sCfg 12 }

sCfgPsTimer OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory

DESCRIPTION

"The number of minutes between updates of the process statistics table (sPsTbl) data. A value of 0 disables gathering of process related data.

Configuration Variable: PsTimer
Set-requests: Take effect on next timer expiration

Changes to this value are written to the sCfgFile."

DEFVAL { 5 }
::= { sCfg 13 }

sCfgDskTimer OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The number of minutes between updates of the disk table (sDskTbl) data. A value of 0 disables gathering of disk related data.

Configuration Variable: DskTimer
Set-requests: Take effect on next timer expiration

Changes to this value are written to the sCfgFile."

DEFVAL { 1 }
::= { sCfg 14 }

sCfgHwErrTimer OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The number of minutes between updates of the hardware error table (sHwErrTbl) data. A value of 0 disables gathering of hardware error data.

Configuration Variable: HwErrTimer
Set-requests: Take effect on next timer expiration

Changes to this value are written to the sCfgFile."

DEFVAL { 1 }
::= { sCfg 15 }

sCfgSysInfoTimer OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The number of minutes between updates of the system information (sSyi) data. A value of 0 disables gathering of system information data.

Configuration Variable: SysInfoTimer

Set-requests: Take effect on next timer expiration

Changes to this value are written to the sCfgFile."

DEFVAL { 1 }
::= { sCfg 16 }

sCfgCritTimer OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The number of minutes between critical process table (sCritTbl) updates. A value of 0 disables critical process missing processing.

Configuration Variable: CritTimer
Set-requests: Take effect on next timer expiration

Changes to this value are written to the sCfgFile."

DEFVAL { 1 }
::= { sCfg 17 }

sCfgDskAlarm OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The number of minutes between repeat traps for disk (sDskFull), page file (sPgFull), or swap file full (sSwpFull) conditions. A value of 0 causes only the initial instance of the disk, page file, or swap file full condition to trigger a trap.

Configuration Variable: RptDskAlarm
Set-requests: Take effect on next timer expiration

Changes to this value are written to the sCfgFile."

DEFVAL { 5 }
::= { sCfg 18 }

sCfgCritAlarm OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The number of minutes between repeat traps for a given critical process missing condition. Setting this variable to 0 causes only the first instance to generate a trap.

Configuration Variable: RptCritAlarm
Set-requests: Take effect on next timer expiration

Changes to this value are written to the sCfgFile."

DEFVAL { 5 }

```
::= { sCfg 19 }
```

```
sCfgTrapTblSize OBJECT-TYPE
```

```
SYNTAX INTEGER (10..5000)
```

```
ACCESS read-write
```

```
STATUS mandatory
```

```
DESCRIPTION
```

"The maximum number of traps to be retained. Traps that are retained may be resent (see sTrapResendSeqNum) unless too many subsequent traps have been generated. This variable controls how many traps are retained in NM*SysMgr's internal trap table.

Configuration Variable: TrapTableSize

Set-Requests: Take effect immediately

Changes to this value are written to the sCfgFile."

```
DEFVAL { 100 }
```

```
::= { sCfg 20 }
```

```
sCfgHostName OBJECT-TYPE
```

```
SYNTAX DisplayString (SIZE(0..255))
```

```
ACCESS read-only
```

```
STATUS mandatory
```

```
DESCRIPTION
```

"The name of the host that the subagent is communicating with. This is the host name where the NM*Master Agent resides. This variable is not currently used.

Configuration Variable: Host"

```
DEFVAL { "localhost" }
```

```
::= { sCfg 21 }
```

```
sCfgAllPsData OBJECT-TYPE
```

```
SYNTAX INTEGER { yes(1), no(2) }
```

```
ACCESS read-write
```

```
STATUS mandatory
```

```
DESCRIPTION
```

"This variable controls the amount of data that is available in the process statistics table (sPsTbl). If this flag is set to no(2), only the process statistics that are contained in the process control block (PCB) or job information block (JIB) for each process are available (data in the process header (PHD) may also be available). Setting this flag to yes makes all statistics in the sPsTbl available but incurs an additional performance overhead (to retrieve out swapped processes and deliver kernel mode ASTs to all processes).

Configuration variable: AllPsData

Set-requests: Take effect immediately.

Changes to this value are written to the sCfgFile."

```
DEFVAL { no }
```

```
::= { sCfg 22 }
```

```
sCfgTimeout OBJECT-TYPE
```

```
SYNTAX INTEGER (0..3600)
```

ACCESS read-only
STATUS mandatory
DESCRIPTION
"This variable identifies the time-out value that is to be used by the subagent during subtree registration. If this value is set to zero, the default value (saDefaultTimeout) contained in the NM*Master Agent will be used. If the value specified exceeds the current maximum that NM*Master will permit (saMaxTimeout), the maximum value will be used. This value is specified in seconds.

Configuration Variable: Timeout"

DEFVAL { 0 }
::= { sCfg 23 }

sCfgOpcomSecurity OBJECT-TYPE

SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION

"Flag which determines if the opcom security messages are to be sent as opcom traps. Setting this variable to enabled(1) causes opcom trap messages to be sent for security messages.

Configuration Variable: OpcomSecurity
Set-requests: Take effect at next process start-up

Changes to this value are written to the tCfgFile."

DEFVAL { enabled }
::= { sCfg 24 }

sCfgControlTermProc OBJECT-TYPE

SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-only
STATUS mandatory
DESCRIPTION

"Flag which may be used to disable process termination by the sCfgTermProc variable. Setting this variable to enabled(1) allows the sCfgTermProc variable to be set to true(1) to terminate the NM*SysMgr process. Setting this variable to disabled(2) prevents the sCfgTermProc variable from being set.

Configuration Variable: ControlTermProc"

DEFVAL { disabled }
::= { sCfg 25 }

sCfgTermProc OBJECT-TYPE

SYNTAX INTEGER { true(1), false(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION

"Flag which may be used to shutdown the NM*SysMgr process. Setting this flag to true causes the NM*SysMgr process to respond to the set-request, send the sTermProc trap, close the connection to the NM*Master Agent, and shutdown. If the sCfgControlTermProc flag state is disabled(2), the sCfgTermProc

variable can not be set.

Set-requests: Take effect immediately if
sCfgControlTermProc is enabled(2)"

DEFVAL { false }
::= { sCfg 26 }

sCfgVersion OBJECT-TYPE

SYNTAX DisplayString
ACCESS read-only
STATUS mandatory
DESCRIPTION

"Current version of the NM*SysMgr Subagent software."
::= { sCfg 27 }

sCfgUpTime OBJECT-TYPE

SYNTAX TimeTicks
ACCESS read-only
STATUS mandatory
DESCRIPTION

"Subagent up time."
::= { sCfg 28 }

sCfgMaxTrapSec OBJECT-TYPE

SYNTAX INTEGER (0..100)
ACCESS read-write
STATUS mandatory
DESCRIPTION

"This value specifies the maximum number of traps per second that may be sent. A value of zero disables trap throttling, allowing all traps to be sent as they are generated. It is recommended that this value and sCfgTrapTblSize be tuned to each operational environment.

Configuration Variable: MaxTrapSec
Set-requests: Take effect immediately.

Changes to this value are written to the sCfgFile."

DEFVAL { 0 }
::= { sCfg 29 }

sCfgCritDsk OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..255))
ACCESS read-write
STATUS mandatory
DESCRIPTION

"Name of the file currently being used to identify critical disk space and operations per second trap thresholds for individual disks. This file contains:

disk name (required)
percent full or number of free blocks (optional)
operations per second (optional)

The disk name may take the form of a physical disk name (including the leading underscore (_) and trailing colon (:)), a logical volume name, or, for shadowed disks, the name of the

virtual unit. If two entries specify the same disk (one by physical name and the other by logical volume name), only the entry for the physical disk name will be used for that disk.

Once the disk usage exceeds the specified percent full or the number of free blocks drops below the specified free disk blocks, disk full (sDskFull) traps will begin to be sent. Disk busy (sDskOpSec) traps are sent if the number of operations per second exceeds the limit specified for this disk.

The following are valid entries in the critical disk file:

```
_ALPHA$DKA0:,blocks=652829
disk$alphasys,percent=99,ops=120
_DSA1:,ops=300
```

This file is associated with the system logical name COMTEK\$SYSMGR_CRITDSK. Changing this variable redefines the COMTEK\$SYSMGR_CRITDSK logical name. NM*SysMgr must be reinitialized (by setting the variable sCfgReinitSubagent to true) before a new critical disk file is used. If a set-request is performed on this variable and the specified file name does not exist, is not accessible by the subagent, or contains invalid data, the file name is not changed, an error is returned, and a sSwErr trap is generated describing the error.

Set-requests: Take effect only after reinitialization
(sCfgReinitSubagent)

NOTE: To make changes to this value permanent across system reboots, the definition of this logical name must also be changed in the system startup procedures."

```
::= { sCfg 30 }
```

sCfgCritQue OBJECT-TYPE

SYNTAX DisplayString (SIZE (0..255))

ACCESS read-write

STATUS mandatory

DESCRIPTION

"Name of the file currently being used to identify queues to be monitored. This file is associated with the logical name COMTEK\$SYSMGR_QUEUE. Changing this variable redefines the COMTEK\$SYSMGR_QUEUE logical name. NM*SysMgr must be reinitialized (by setting the variable sCfgReinitSubagent to true) before the new queue file is used. If a set-request is performed on this variable and the specified file name does not exist, is not accessible by the subagent, or contains invalid data, the file name is not changed, an error is returned, and a sSwErr trap is generated describing the error.

Set-requests: Take effect only after reinitialization
(sCfgReinitSubagent)

NOTE: To make changes to this value permanent across system reboots, the definition of this logical name must also be

changed in the system startup procedures."
::= { sCfg 31 }

sCfgQueTimer OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The number of minutes between updates of the queue statistics table (sQTbl) data. A value of 0 disables gathering of queue related data.

Configuration Variable: QueTimer
Set-requests: Take effect on next timer expiration

Changes to this value are written to the sCfgFile."

DEFVAL { 1 }
::= { sCfg 32 }

sCfgDskMinFreeBlks OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The minimum number of disk blocks that must be free on a disk. When the number of free blocks on a disk drops below this level, excessive disk utilization (sDskFull) traps begin to be sent. If this variable is set to a value other than zero, it is used to determine when to send sDskFull traps for disks not specified in the critical disk table (see sCfgCritDsk).

This value and sCfgDskLimit are mutually exclusive, i.e., only one of these values is in effect at any given time. If neither value is specified in the configuration file, then the default value of sCfgDskLimit is used for disk monitoring and sCfgDskMinFreeBlks is set to zero. To disable default disk full monitoring, both of these values must be zero.

Set-requests to this value will fail unless sCfgDskLimit is set to zero.

Configuration Variable: DskMinFreeBlks
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."
::= { sCfg 33 }

sCfgMemLimit OBJECT-TYPE

SYNTAX INTEGER (0..100)
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The percentage of physical memory utilization at which excessive memory utilization (sMemFull) traps begin to be sent. A value of 0 disables these traps.

Configuration Variable: MemUse
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."
DEFVAL { 90 }
::= { sCfg 34 }

sCfgCOMQueueLimit OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The number of processes in COM state at which excessive compute queue (sCOMQue) traps begin to be sent. A value of 0 disables these traps.

Configuration Variable: COMQueue
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."
DEFVAL { 5 }
::= { sCfg 35 }

sCfgCOMOQueueLimit OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The number of processes in COMO state at which excessive outswapped compute queue (sCOMOQue) traps begin to be sent. A value of 0 disables these traps.

Configuration Variable: COMOQueue
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."
DEFVAL { 1 }
::= { sCfg 36 }

sCfgOpcomCards OBJECT-TYPE

SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION

"Flag which determines if NM*SysMgr will receive opcom CARDS class messages. Setting this variable to disabled(2) disables NM*SysMgr from receiving CARDS opcom messages.

Configuration Variable: Cards
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 37 }

sCfgOpcomCentral OBJECT-TYPE

SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Flag which determines if NM*SysMgr will receive opcom
CENTRAL class messages. Setting this variable to disabled(2)
disables NM*SysMgr from receiving CENTRAL opcom messages.

Configuration Variable: Central
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 38 }

sCfgOpcomCluster OBJECT-TYPE

SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Flag which determines if NM*SysMgr will receive opcom
CLUSTER class messages. Setting this variable to disabled(2)
disables NM*SysMgr from receiving CLUSTER opcom messages.

Configuration Variable: Cluster
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 39 }

sCfgOpcomDevices OBJECT-TYPE

SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Flag which determines if NM*SysMgr will receive opcom
DEVICES class messages. Setting this variable to disabled(2)
disables NM*SysMgr from receiving DEVICES opcom messages.

Configuration Variable: Devices
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 40 }

sCfgOpcomDisks OBJECT-TYPE

SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Flag which determines if NM*SysMgr will receive opcom
DISKS class messages. Setting this variable to disabled(2)
disables NM*SysMgr from receiving DISKS opcom messages.

Configuration Variable: Disks

```

                Set-requests:                Take effect on next process start-up

                Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 41 }

sCfgOpcomLicense OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Flag which determines if NM*SysMgr will receive opcom
    LICENSE class messages.  Setting this variable to disabled(2)
    disables NM*SysMgr from receiving LICENSE opcom messages.

    Configuration Variable: License
    Set-requests:                Take effect on next process start-up

    Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 42 }

sCfgOpcomNetwork OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Flag which determines if NM*SysMgr will receive opcom
    NETWORK class messages.  Setting this variable to disabled(2)
    disables NM*SysMgr from receiving NETWORK opcom messages.

    Configuration Variable: Network
    Set-requests:                Take effect on next process start-up

    Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 43 }

sCfgOpcomOper1 OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Flag which determines if NM*SysMgr will receive opcom
    OPER1 class messages.  Setting this variable to disabled(2)
    disables NM*SysMgr from receiving OPER1 opcom messages.

    Configuration Variable: Oper1
    Set-requests:                Take effect on next process start-up

    Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 44 }

sCfgOpcomOper2 OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write

```

STATUS mandatory
DESCRIPTION
"Flag which determines if NM*SysMgr will receive opcom
OPER2 class messages. Setting this variable to disabled(2)
disables NM*SysMgr from receiving OPER2 opcom messages.

Configuration Variable: Oper2
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 45 }

sCfgOpcomOper3 OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Flag which determines if NM*SysMgr will receive opcom
OPER3 class messages. Setting this variable to disabled(2)
disables NM*SysMgr from receiving OPER3 opcom messages.

Configuration Variable: Oper3
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 46 }

sCfgOpcomOper4 OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Flag which determines if NM*SysMgr will receive opcom
OPER4 class messages. Setting this variable to disabled(2)
disables NM*SysMgr from receiving OPER4 opcom messages.

Configuration Variable: Oper4
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 47 }

sCfgOpcomOper5 OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
"Flag which determines if NM*SysMgr will receive opcom
OPER5 class messages. Setting this variable to disabled(2)
disables NM*SysMgr from receiving OPER5 opcom messages.

Configuration Variable: Oper5
Set-requests: Take effect on next process start-up

```

        Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 48 }

sCfgOpcomOper6 OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Flag which determines if NM*SysMgr will receive opcom
    OPER6 class messages. Setting this variable to disabled(2)
    disables NM*SysMgr from receiving OPER6 opcom messages.

    Configuration Variable: Oper6
    Set-requests:           Take effect on next process start-up

    Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 49 }

sCfgOpcomOper7 OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Flag which determines if NM*SysMgr will receive opcom
    OPER7 class messages. Setting this variable to disabled(2)
    disables NM*SysMgr from receiving OPER7 opcom messages.

    Configuration Variable: Oper7
    Set-requests:           Take effect on next process start-up

    Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 50 }

sCfgOpcomOper8 OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION
    "Flag which determines if NM*SysMgr will receive opcom
    OPER8 class messages. Setting this variable to disabled(2)
    disables NM*SysMgr from receiving OPER8 opcom messages.

    Configuration Variable: Oper8
    Set-requests:           Take effect on next process start-up

    Changes to this value are written to the sCfgFile."
DEFVAL { enabled }
::= { sCfg 51 }

sCfgOpcomOper9 OBJECT-TYPE
SYNTAX INTEGER { enabled(1), disabled(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION

```

"Flag which determines if NM*SysMgr will receive opcom OPER9 class messages. Setting this variable to disabled(2) disables NM*SysMgr from receiving OPER9 opcom messages.

Configuration Variable: Oper9
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."

DEFVAL { enabled }
::= { sCfg 52 }

sCfgOpcomOper10 OBJECT-TYPE

SYNTAX INTEGER { enabled(1), disabled(2) }

ACCESS read-write

STATUS mandatory

DESCRIPTION

"Flag which determines if NM*SysMgr will receive opcom OPER10 class messages. Setting this variable to disabled(2) disables NM*SysMgr from receiving OPER10 opcom messages.

Configuration Variable: Oper10
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."

DEFVAL { enabled }
::= { sCfg 53 }

sCfgOpcomOper11 OBJECT-TYPE

SYNTAX INTEGER { enabled(1), disabled(2) }

ACCESS read-write

STATUS mandatory

DESCRIPTION

"Flag which determines if NM*SysMgr will receive opcom OPER11 class messages. Setting this variable to disabled(2) disables NM*SysMgr from receiving OPER11 opcom messages.

Configuration Variable: Oper11
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."

DEFVAL { enabled }
::= { sCfg 54 }

sCfgOpcomOper12 OBJECT-TYPE

SYNTAX INTEGER { enabled(1), disabled(2) }

ACCESS read-write

STATUS mandatory

DESCRIPTION

"Flag which determines if NM*SysMgr will receive opcom OPER12 class messages. Setting this variable to disabled(2) disables NM*SysMgr from receiving OPER12 opcom messages.

Configuration Variable: Oper12
Set-requests: Take effect on next process start-up

Changes to this value are written to the sCfgFile."

DEFVAL { enabled }

```
::= { sCfg 55 }
```

```
sCfgOpcomPrinter OBJECT-TYPE
```

```
SYNTAX INTEGER { enabled(1), disabled(2) }
```

```
ACCESS read-write
```

```
STATUS mandatory
```

```
DESCRIPTION
```

```
"Flag which determines if NM*SysMgr will receive opcom  
PRINTER class messages. Setting this variable to disabled(2)  
disables NM*SysMgr from receiving PRINTER opcom messages.
```

```
Configuration Variable: Printer
```

```
Set-requests: Take effect on next process start-up
```

```
Changes to this value are written to the sCfgFile."
```

```
DEFVAL { enabled }
```

```
::= { sCfg 56 }
```

```
sCfgOpcomTapes OBJECT-TYPE
```

```
SYNTAX INTEGER { enabled(1), disabled(2) }
```

```
ACCESS read-write
```

```
STATUS mandatory
```

```
DESCRIPTION
```

```
"Flag which determines if NM*SysMgr will receive opcom  
TAPES class messages. Setting this variable to disabled(2)  
disables NM*SysMgr from receiving TAPES opcom messages.
```

```
Configuration Variable: Tapes
```

```
Set-requests: Take effect on next process start-up
```

```
Changes to this value are written to the sCfgFile."
```

```
DEFVAL { enabled }
```

```
::= { sCfg 57 }
```

```
sCfgOpcomFilter OBJECT-TYPE
```

```
SYNTAX INTEGER { dropMatch(1), keepMatch(2) }
```

```
ACCESS read-write
```

```
STATUS mandatory
```

```
DESCRIPTION
```

```
that "Flag which determines if NM*SysMgr will drop opcom messages  
that match filter criteria or drop all opcom messages except those  
that match filter criteria.
```

```
Configuration Variable: Filter
```

```
Set-requests: Take effect on next process start-up
```

```
Changes to this value are written to the sCfgFile."
```

```
DEFVAL { dropMatch }
```

```
::= { sCfg 58 }
```

```
sCfgLocalDisksOnly OBJECT-TYPE
```

```
SYNTAX INTEGER { true(1), false(2) }
```

```
ACCESS read-write
```

```
STATUS mandatory
```

```
DESCRIPTION
```

"Flag which may be used to instruct NM*SysMgr to gather disk statistics either on all available disks or only on local disks. A disk is considered to be local if the variable sDskRemote for the disk is FALSE(2), indicating that the disk is not remote. If this flag is set to TRUE(1), the disk table (sDskTbl) will only contain data on local disks. If this flag is set to FALSE(2), the disk table (sDskTbl) will contain data on all available disks.

Configuration Variable: LocalDisksOnly
Set-requests: Take effect on next timer expiration

Changes to this value are written to the sCfgFile."

DEFVAL { false }
::= { sCfg 59 }

sCfgCritDisksOnly OBJECT-TYPE

SYNTAX INTEGER { true(1), false(2) }
ACCESS read-write
STATUS mandatory
DESCRIPTION

"Flag which may be used to instruct NM*SysMgr to gather disk statistics either on all available disks or only on disks listed in the Critical Disk file (COMTEK\$SYSMGR_CRITDSK). If this flag is set to TRUE(1), the disk table (sDskTbl) will only contain data on disks listed in the Critical Disk file. If this flag is set to FALSE(2), the disk table (tblsDskTbl) will contain data on all available disks.

Configuration Variable: CritDisksOnly
Set-requests: Take effect on next timer expiration

Changes to this value are written to the sCfgFile."

DEFVAL { false }
::= { sCfg 60 }

sCfgCpuPersist OBJECT-TYPE

SYNTAX INTEGER
ACCESS read-write
STATUS mandatory
DESCRIPTION

"The number of minutes the CPU must remain above the CPU utilization threshold (sCfgCpuLimit) before an excessive CPU utilization trap is sent. The value zero is not permitted for this variable.

Configuration Variable: CpuPersist
Set-requests: Take effect immediately

Changes to this value are written to the sCfgFile."

DEFVAL { 1 }
::= { sCfg 61 }

```
sCfgMWAITLimit OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-write
    STATUS  mandatory
    DESCRIPTION
        "If the number of processes in the MWAIT state exceeds
        this threshold, the sMWAITExcessive trap is sent.  Set
        this threshold to zero to disable the sMWAITExcessive
        traps.

        Configuration Variable:  MWAITLimit
        Set-requests:            Take effect immediately

        Changes to this value are written to the sCfgFile."
    DEFVAL  { 0 }
    ::= { sCfg 62 }
```

END