Network Working Group
Request for Comments: 2790
Obsoletes: 1514
Category: Standards Track

S. Waldbusser
Lucent Technologies Inc.
P. Grillo
WeSync.com

March 2000

Host Resources MIB

Status of this Memo

Copyright Notice

Abstract

This memo defines a portion of the Management Information Base (MIB) for use with network management protocols in the Internet community. This memo obsoletes RFC 1514, the "Host Resources MIB". This memo extends that specification by clarifying changes based on implementation and deployment experience and documenting the Host Resources MIB in SMIv2 format while remaining semantically identical to the existing SMIv1-based MIB.

This memo defines a MIB for use with managing host systems. The term "host" is construed to mean any computer that communicates with other similar computers attached to the internet and that is directly used by one or more human beings. Although this MIB does not necessarily apply to devices whose primary function is communications services (e.g., terminal servers, routers, bridges, monitoring equipment), such relevance is not explicitly precluded. This MIB instruments attributes common to all internet hosts including, for example, both personal computers and systems that run variants of Unix.

Table of Contents

1 The SNMP Management Framework 2
2 Host Resources MIB 3
3 IANA Considerations 4
4 Definitions 4
4.1 Textual Conventions 6
4.2 The Host Resources System Group 7
4.3 The Host Resources Storage Group 9
4.4 The Host Resources Device Group 12
4.5 The Host Resources Running Software Group 26
4.6 The Host Resources Running Software Performance Group 29
4.7 The Host Resources Installed Software Group 30
4.8 Conformance Definitions 33
5 Type Definitions 36
6 Internationalization Considerations 44
7 Security Considerations 45
8 References 46
9 Acknowledgments 48
10 Authors' Addresses 49
11 Intellectual Property 49
12 Full Copyright Statement 50

1. The SNMP Management Framework

The SNMP Management Framework presently consists of five major components:

- An overall architecture, described in RFC 2571 [RFC2571].

- Mechanisms for describing and naming objects and events for the purpose of management. The first version of this Structure of Management Information (SMI) is called SMIv1 and described in STD 16, RFC 1155 [RFC1155], STD 16, RFC 1212 [RFC1212] and RFC 1215 [RFC1215]. The second version, called SMIv2, is described in STD 58, RFC 2578 [RFC2578], RFC 2579 [RFC2579] and RFC 2580 [RFC2580].

- Message protocols for transferring management information. The first version of the SNMP message protocol is called SNMPv1 and described in STD 15, RFC 1157 [RFC1157]. A second version of the SNMP message protocol, which is not an Internet standards track protocol, is called SNMPv2c and described in RFC 1901 [RFC1901] and RFC 1906 [RFC1906]. The third version of the message protocol is called SNMPv3 and described in RFC 1906 [RFC1906], RFC 2572 [RFC2572] and RFC 2574 [RFC2574].

- Protocol operations for accessing management information. The first set of protocol operations and associated PDU formats is described in STD 15, RFC 1157 [RFC1157]. A second set of protocol operations and associated PDU formats is described in RFC 1905 [RFC1905].

- A set of fundamental applications described in RFC 2573 [RFC2573] and the view-based access control mechanism described in RFC 2575 [RFC2575].

A more detailed introduction to the current SNMP Management Framework can be found in RFC 2570 [RFC2570].

Managed objects are accessed via a virtual information store, termed the Management Information Base or MIB. Objects in the MIB are defined using the mechanisms defined in the SMI.

This memo specifies a MIB module that is compliant to the SMIv2. A MIB conforming to the SMIv1 can be produced through the appropriate translations. The resulting translated MIB must be semantically equivalent, except where objects or events are omitted because no translation is possible (use of Counter64). Some machine readable information in SMIv2 will be converted into textual descriptions in SMIv1 during the translation process. However, this loss of machine readable information is not considered to change the semantics of the MIB.

2. Host Resources MIB

The Host Resources MIB defines a uniform set of objects useful for the management of host computers. Host computers are independent of the operating system, network services, or any software application.

The Host Resources MIB defines objects which are common across many computer system architectures.

In addition, there are objects in the SNMPv2-MIB [RFC1907] and IF-MIB [RFC2233] which also provide host management functionality. Implementation of the System and Interfaces groups is mandatory for implementors of the Host Resources MIB.

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED","MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. IANA Considerations

This MIB contains type definitions for storage types, device types, and file system types for use as values for the hrStorageType, hrDeviceType, and hrFSType objects, respectively. As new computing technologies are developed, new types need to be registered for these technologies. The IANA (Internet Assigned Numbers Authority) is designated as the registration authority for new registrations beyond those published in this document. The IANA will maintain the HOST-RESOURCES-TYPES module as new registrations are added and publish new versions of this module.

Given the large number of such technologies and potential confusion in naming of these technologies (such as a technology known by two names or a name and an acronym), there is a real danger that more than one registration might be created for what is essentially the same technology. In order to ensure that future type registrations are performed correctly, applications for new types will be reviewed by a Designated Expert appointed by the IESG.

4. Definitions

```
HOST-RESOURCES-MIB DEFINITIONS ::= BEGIN

IMPORTS
MODULE-IDENTITY, OBJECT-TYPE, mib-2,
Integer32, Counter32, Gauge32, TimeTicks  FROM SNMPv2-SMI

TEXTUAL-CONVENTION, DisplayString,
TruthValue, DateAndTime, AutonomousType   FROM SNMPv2-TC

MODULE-COMPLIANCE, OBJECT-GROUP           FROM SNMPv2-CONF

InterfaceIndexOrZero                      FROM IF-MIB;

hostResourcesMibModule MODULE-IDENTITY
   LAST-UPDATED "200003060000Z"    -- 6 March 2000
   ORGANIZATION "IETF Host Resources MIB Working Group"
   CONTACT-INFO
       "Steve Waldbusser
       Postal: Lucent Technologies, Inc.
               1213 Innsbruck Dr.
               Sunnyvale, CA 94089
               USA
       Phone:  650-318-1251
       Fax:    650-318-1633
       Email:  waldbusser@lucent.com
```

```
        In addition, the Host Resources MIB mailing list is
        dedicated to discussion of this MIB. To join the
        mailing list, send a request message to
        hostmib-request@andrew.cmu.edu. The mailing list
        address is hostmib@andrew.cmu.edu."

    DESCRIPTION
        "This MIB is for use in managing host systems. The term
        'host' is construed to mean any computer that communicates
        with other similar computers attached to the internet and
        that is directly used by one or more human beings. Although
        this MIB does not necessarily apply to devices whose primary
        function is communications services (e.g., terminal servers,
        routers, bridges, monitoring equipment), such relevance is
        not explicitly precluded.  This MIB instruments attributes
        common to all internet hosts including, for example, both
        personal computers and systems that run variants of Unix."

    REVISION "200003060000Z"        -- 6 March 2000
    DESCRIPTION
        "Clarifications and bug fixes based on implementation
        experience.  This revision was also reformatted in the SMIv2
        format. The revisions made were:

        New RFC document standards:
           Added Copyright notice, updated introduction to SNMP
           Framework, updated references section, added reference to
           RFC 2119, and added a meaningful Security Considerations
           section.

        New IANA considerations section for registration of new types

        Conversion to new SMIv2 syntax for the following types and
        macros:
            Counter32, Integer32, Gauge32, MODULE-IDENTITY,
            OBJECT-TYPE, TEXTUAL-CONVENTION, OBJECT-IDENTITY,
            MODULE-COMPLIANCE, OBJECT-GROUP

        Used new Textual Conventions:
            TruthValue, DateAndTime, AutonomousType,
            InterfaceIndexOrZero

        Fixed typo in hrPrinterStatus.

        Added missing error bits to hrPrinterDetectedErrorState and
        clarified confusion resulting from suggested mappings to
        hrPrinterStatus.
```

```
          Clarified that size of objects of type
          InternationalDisplayString is number of octets, not number
          of encoded symbols.

          Clarified the use of the following objects based on
          implementation experience:
              hrSystemInitialLoadDevice, hrSystemInitialLoadParameters,
              hrMemorySize, hrStorageSize, hrStorageAllocationFailures,
              hrDeviceErrors, hrProcessorLoad, hrNetworkIfIndex,
              hrDiskStorageCapacity, hrSWRunStatus, hrSWRunPerfCPU,
              and hrSWInstalledDate.

          Clarified implementation technique for hrSWInstalledTable.

          Used new AUGMENTS clause for hrSWRunPerfTable.

          Added Internationalization Considerations section.

   This revision published as RFC2790."

      REVISION "9910202200Z"    -- 20 October, 1999
      DESCRIPTION
          "The original version of this MIB, published as
          RFC1514."
      ::= { hrMIBAdminInfo 1 }

   host     OBJECT IDENTIFIER ::= { mib-2 25 }

   hrSystem        OBJECT IDENTIFIER ::= { host 1 }
   hrStorage       OBJECT IDENTIFIER ::= { host 2 }
   hrDevice        OBJECT IDENTIFIER ::= { host 3 }
   hrSWRun         OBJECT IDENTIFIER ::= { host 4 }
   hrSWRunPerf     OBJECT IDENTIFIER ::= { host 5 }
   hrSWInstalled   OBJECT IDENTIFIER ::= { host 6 }
   hrMIBAdminInfo  OBJECT IDENTIFIER ::= { host 7 }

   -- textual conventions

   KBytes ::= TEXTUAL-CONVENTION
       STATUS current
       DESCRIPTION
           "Storage size, expressed in units of 1024 bytes."
       SYNTAX Integer32 (0..2147483647)

   ProductID ::= TEXTUAL-CONVENTION
       STATUS current
       DESCRIPTION
           "This textual convention is intended to identify the
```

```
            manufacturer, model, and version of a specific
            hardware or software product.  It is suggested that
            these OBJECT IDENTIFIERs are allocated such that all
            products from a particular manufacturer are registered
            under a subtree distinct to that manufacturer.  In
            addition, all versions of a product should be
            registered under a subtree distinct to that product.
            With this strategy, a management station may uniquely
            determine the manufacturer and/or model of a product
            whose productID is unknown to the management station.
            Objects of this type may be useful for inventory
            purposes or for automatically detecting
            incompatibilities or version mismatches between
            various hardware and software components on a system.

            For example, the product ID for the ACME 4860 66MHz
            clock doubled processor might be:
            enterprises.acme.acmeProcessors.a4860DX2.MHz66

            A software product might be registered as:
            enterprises.acme.acmeOperatingSystems.acmeDOS.six(6).one(1)
            "
        SYNTAX OBJECT IDENTIFIER

    -- unknownProduct will be used for any unknown ProductID
    -- unknownProduct OBJECT IDENTIFIER ::= { 0 0 }

    InternationalDisplayString ::= TEXTUAL-CONVENTION
        STATUS current
        DESCRIPTION
            "This data type is used to model textual information
            in some character set.  A network management station
            should use a local algorithm to determine which
            character set is in use and how it should be
            displayed.  Note that this character set may be
            encoded with more than one octet per symbol, but will
            most often be NVT ASCII. When a size clause is
            specified for an object of this type, the size refers
            to the length in octets, not the number of symbols."
        SYNTAX OCTET STRING

    -- The Host Resources System Group

    hrSystemUptime OBJECT-TYPE
        SYNTAX     TimeTicks
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
```

```
            "The amount of time since this host was last
            initialized.  Note that this is different from
            sysUpTime in the SNMPv2-MIB [RFC1907] because
            sysUpTime is the uptime of the network management
            portion of the system."
        ::= { hrSystem 1 }

    hrSystemDate OBJECT-TYPE
        SYNTAX     DateAndTime
        MAX-ACCESS read-write
        STATUS     current
        DESCRIPTION
            "The host's notion of the local date and time of day."
        ::= { hrSystem 2 }

    hrSystemInitialLoadDevice OBJECT-TYPE
        SYNTAX     Integer32 (1..2147483647)
        MAX-ACCESS read-write
        STATUS     current
        DESCRIPTION
            "The index of the hrDeviceEntry for the device from
            which this host is configured to load its initial
            operating system configuration (i.e., which operating
            system code and/or boot parameters).

            Note that writing to this object just changes the
            configuration that will be used the next time the
            operating system is loaded and does not actually cause
            the reload to occur."
        ::= { hrSystem 3 }

    hrSystemInitialLoadParameters OBJECT-TYPE
        SYNTAX     InternationalDisplayString (SIZE (0..128))
        MAX-ACCESS read-write
        STATUS     current
        DESCRIPTION
            "This object contains the parameters (e.g. a pathname
            and parameter) supplied to the load device when
            requesting the initial operating system configuration
            from that device.

         Note that writing to this object just changes the
         configuration that will be used the next time the
         operating system is loaded and does not actually cause
         the reload to occur."
        ::= { hrSystem 4 }

    hrSystemNumUsers OBJECT-TYPE
```

```
        SYNTAX     Gauge32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The number of user sessions for which this host is
            storing state information.  A session is a collection
            of processes requiring a single act of user
            authentication and possibly subject to collective job
            control."
        ::= { hrSystem 5 }

    hrSystemProcesses OBJECT-TYPE
        SYNTAX     Gauge32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The number of process contexts currently loaded or
            running on this system."
        ::= { hrSystem 6 }

    hrSystemMaxProcesses OBJECT-TYPE
        SYNTAX     Integer32 (0..2147483647)
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The maximum number of process contexts this system
            can support.  If there is no fixed maximum, the value
            should be zero.  On systems that have a fixed maximum,
            this object can help diagnose failures that occur when
            this maximum is reached."
        ::= { hrSystem 7 }

    -- The Host Resources Storage Group

    -- Registration point for storage types, for use with hrStorageType.
    -- These are defined in the HOST-RESOURCES-TYPES module.
    hrStorageTypes          OBJECT IDENTIFIER ::= { hrStorage 1 }

    hrMemorySize OBJECT-TYPE
        SYNTAX     KBytes
        UNITS      "KBytes"
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The amount of physical read-write main memory,
            typically RAM, contained by the host."
        ::= { hrStorage 2 }
```

```
   hrStorageTable OBJECT-TYPE
       SYNTAX     SEQUENCE OF HrStorageEntry
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
           "The (conceptual) table of logical storage areas on
           the host.

           An entry shall be placed in the storage table for each
           logical area of storage that is allocated and has
           fixed resource limits.  The amount of storage
           represented in an entity is the amount actually usable
           by the requesting entity, and excludes loss due to
           formatting or file system reference information.

           These entries are associated with logical storage
           areas, as might be seen by an application, rather than
           physical storage entities which are typically seen by
           an operating system.  Storage such as tapes and
           floppies without file systems on them are typically
           not allocated in chunks by the operating system to
           requesting applications, and therefore shouldn't
           appear in this table.  Examples of valid storage for
           this table include disk partitions, file systems, ram
           (for some architectures this is further segmented into
           regular memory, extended memory, and so on), backing
           store for virtual memory ('swap space').

           This table is intended to be a useful diagnostic for
           'out of memory' and 'out of buffers' types of
           failures.  In addition, it can be a useful performance
           monitoring tool for tracking memory, disk, or buffer
           usage."
       ::= { hrStorage 3 }

   hrStorageEntry OBJECT-TYPE
       SYNTAX     HrStorageEntry
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
           "A (conceptual) entry for one logical storage area on
           the host.  As an example, an instance of the
           hrStorageType object might be named hrStorageType.3"
       INDEX { hrStorageIndex }
       ::= { hrStorageTable 1 }

   HrStorageEntry ::= SEQUENCE {
           hrStorageIndex              Integer32,
```

```
            hrStorageType               AutonomousType,
            hrStorageDescr              DisplayString,
            hrStorageAllocationUnits    Integer32,
            hrStorageSize               Integer32,
            hrStorageUsed               Integer32,
            hrStorageAllocationFailures Counter32
        }

    hrStorageIndex OBJECT-TYPE
        SYNTAX     Integer32 (1..2147483647)
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "A unique value for each logical storage area
            contained by the host."
        ::= { hrStorageEntry 1 }

    hrStorageType OBJECT-TYPE
        SYNTAX     AutonomousType
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The type of storage represented by this entry."
        ::= { hrStorageEntry 2 }

    hrStorageDescr OBJECT-TYPE
        SYNTAX     DisplayString
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "A description of the type and instance of the storage
            described by this entry."
        ::= { hrStorageEntry 3 }

    hrStorageAllocationUnits OBJECT-TYPE
        SYNTAX     Integer32 (1..2147483647)
        UNITS      "Bytes"
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The size, in bytes, of the data objects allocated
            from this pool.  If this entry is monitoring sectors,
            blocks, buffers, or packets, for example, this number
            will commonly be greater than one.  Otherwise this
            number will typically be one."
        ::= { hrStorageEntry 4 }

    hrStorageSize OBJECT-TYPE
```

```
        SYNTAX     Integer32 (0..2147483647)
        MAX-ACCESS read-write
        STATUS     current
        DESCRIPTION
            "The size of the storage represented by this entry, in
            units of hrStorageAllocationUnits. This object is
            writable to allow remote configuration of the size of
            the storage area in those cases where such an
            operation makes sense and is possible on the
            underlying system. For example, the amount of main
            memory allocated to a buffer pool might be modified or
            the amount of disk space allocated to virtual memory
            might be modified."
        ::= { hrStorageEntry 5 }

    hrStorageUsed OBJECT-TYPE
        SYNTAX     Integer32 (0..2147483647)
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The amount of the storage represented by this entry
            that is allocated, in units of
            hrStorageAllocationUnits."
        ::= { hrStorageEntry 6 }

    hrStorageAllocationFailures OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The number of requests for storage represented by
            this entry that could not be honored due to not enough
            storage.  It should be noted that as this object has a
            SYNTAX of Counter32, that it does not have a defined
            initial value.  However, it is recommended that this
            object be initialized to zero, even though management
            stations must not depend on such an initialization."
        ::= { hrStorageEntry 7 }

    -- The Host Resources Device Group
    --
    -- The device group is useful for identifying and diagnosing the
    -- devices on a system.  The hrDeviceTable contains common
    -- information for any type of device.  In addition, some devices
    -- have device-specific tables for more detailed information.  More
    -- such tables may be defined in the future for other device types.

    -- Registration point for device types, for use with hrDeviceType.
```

```
   -- These are defined in the HOST-RESOURCES-TYPES module.
   hrDeviceTypes             OBJECT IDENTIFIER ::= { hrDevice 1 }

   hrDeviceTable OBJECT-TYPE
       SYNTAX     SEQUENCE OF HrDeviceEntry
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
           "The (conceptual) table of devices contained by the
           host."
       ::= { hrDevice 2 }

   hrDeviceEntry OBJECT-TYPE
       SYNTAX     HrDeviceEntry
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
           "A (conceptual) entry for one device contained by the
           host.  As an example, an instance of the hrDeviceType
           object might be named hrDeviceType.3"
       INDEX { hrDeviceIndex }
       ::= { hrDeviceTable 1 }

   HrDeviceEntry ::= SEQUENCE {
           hrDeviceIndex           Integer32,
           hrDeviceType            AutonomousType,
           hrDeviceDescr           DisplayString,
           hrDeviceID              ProductID,
           hrDeviceStatus          INTEGER,
           hrDeviceErrors          Counter32
       }

   hrDeviceIndex OBJECT-TYPE
       SYNTAX     Integer32 (1..2147483647)
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "A unique value for each device contained by the host.
           The value for each device must remain constant at
           least from one re-initialization of the agent to the
           next re-initialization."
       ::= { hrDeviceEntry 1 }

   hrDeviceType OBJECT-TYPE
       SYNTAX     AutonomousType
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
```

```
            "An indication of the type of device.

            If this value is
            'hrDeviceProcessor { hrDeviceTypes 3 }' then an entry
            exists in the hrProcessorTable which corresponds to
            this device.

            If this value is
            'hrDeviceNetwork { hrDeviceTypes 4 }', then an entry
            exists in the hrNetworkTable which corresponds to this
            device.

            If this value is
            'hrDevicePrinter { hrDeviceTypes 5 }', then an entry
            exists in the hrPrinterTable which corresponds to this
            device.

            If this value is
            'hrDeviceDiskStorage { hrDeviceTypes 6 }', then an
            entry exists in the hrDiskStorageTable which
            corresponds to this device."
        ::= { hrDeviceEntry 2 }

    hrDeviceDescr OBJECT-TYPE
        SYNTAX     DisplayString (SIZE (0..64))
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "A textual description of this device, including the
            device's manufacturer and revision, and optionally,
            its serial number."
        ::= { hrDeviceEntry 3 }

    hrDeviceID OBJECT-TYPE
        SYNTAX     ProductID
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The product ID for this device."
        ::= { hrDeviceEntry 4 }

    hrDeviceStatus OBJECT-TYPE
        SYNTAX     INTEGER {
                       unknown(1),
                       running(2),
                       warning(3),
                       testing(4),
                       down(5)
```

```
                }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The current operational state of the device described
            by this row of the table.  A value unknown(1)
            indicates that the current state of the device is
            unknown.  running(2) indicates that the device is up
            and running and that no unusual error conditions are
            known.  The warning(3) state indicates that agent has
            been informed of an unusual error condition by the
            operational software (e.g., a disk device driver) but
            that the device is still 'operational'.  An example
            would be a high number of soft errors on a disk.  A
            value of testing(4), indicates that the device is not
            available for use because it is in the testing state.
            The state of down(5) is used only when the agent has
            been informed that the device is not available for any
            use."
        ::= { hrDeviceEntry 5 }

    hrDeviceErrors OBJECT-TYPE
        SYNTAX     Counter32
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The number of errors detected on this device.  It
            should be noted that as this object has a SYNTAX of
            Counter32, that it does not have a defined initial
            value.  However, it is recommended that this object be
            initialized to zero, even though management stations
            must not depend on such an initialization."
        ::= { hrDeviceEntry 6 }

    hrProcessorTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF HrProcessorEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "The (conceptual) table of processors contained by the
            host.

            Note that this table is potentially sparse: a
            (conceptual) entry exists only if the correspondent
            value of the hrDeviceType object is
            'hrDeviceProcessor'."
        ::= { hrDevice 3 }
```

```
   hrProcessorEntry OBJECT-TYPE
       SYNTAX     HrProcessorEntry
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
           "A (conceptual) entry for one processor contained by
           the host.  The hrDeviceIndex in the index represents
           the entry in the hrDeviceTable that corresponds to the
           hrProcessorEntry.

           As an example of how objects in this table are named,
           an instance of the hrProcessorFrwID object might be
           named hrProcessorFrwID.3"
       INDEX { hrDeviceIndex }
       ::= { hrProcessorTable 1 }

   HrProcessorEntry ::= SEQUENCE {
           hrProcessorFrwID            ProductID,
           hrProcessorLoad             Integer32
       }

   hrProcessorFrwID OBJECT-TYPE
       SYNTAX     ProductID
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "The product ID of the firmware associated with the
           processor."
       ::= { hrProcessorEntry 1 }

   hrProcessorLoad OBJECT-TYPE
       SYNTAX     Integer32 (0..100)
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "The average, over the last minute, of the percentage
           of time that this processor was not idle.
           Implementations may approximate this one minute
           smoothing period if necessary."
       ::= { hrProcessorEntry 2 }

   hrNetworkTable OBJECT-TYPE
       SYNTAX     SEQUENCE OF HrNetworkEntry
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
           "The (conceptual) table of network devices contained
           by the host.
```

```
            Note that this table is potentially sparse: a
            (conceptual) entry exists only if the correspondent
            value of the hrDeviceType object is
            'hrDeviceNetwork'."
        ::= { hrDevice 4 }

    hrNetworkEntry OBJECT-TYPE
        SYNTAX     HrNetworkEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "A (conceptual) entry for one network device contained
            by the host.  The hrDeviceIndex in the index
            represents the entry in the hrDeviceTable that
            corresponds to the hrNetworkEntry.

            As an example of how objects in this table are named,
            an instance of the hrNetworkIfIndex object might be
            named hrNetworkIfIndex.3"
        INDEX { hrDeviceIndex }
        ::= { hrNetworkTable 1 }

    HrNetworkEntry ::= SEQUENCE {
            hrNetworkIfIndex    InterfaceIndexOrZero
        }

    hrNetworkIfIndex OBJECT-TYPE
        SYNTAX     InterfaceIndexOrZero
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The value of ifIndex which corresponds to this
            network device. If this device is not represented in
            the ifTable, then this value shall be zero."
        ::= { hrNetworkEntry 1 }

    hrPrinterTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF HrPrinterEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "The (conceptual) table of printers local to the host.

            Note that this table is potentially sparse: a
            (conceptual) entry exists only if the correspondent
            value of the hrDeviceType object is
            'hrDevicePrinter'."
        ::= { hrDevice 5 }
```

```
   hrPrinterEntry OBJECT-TYPE
       SYNTAX     HrPrinterEntry
       MAX-ACCESS not-accessible
       STATUS     current
       DESCRIPTION
           "A (conceptual) entry for one printer local to the
           host.  The hrDeviceIndex in the index represents the
           entry in the hrDeviceTable that corresponds to the
           hrPrinterEntry.

           As an example of how objects in this table are named,
           an instance of the hrPrinterStatus object might be
           named hrPrinterStatus.3"
       INDEX { hrDeviceIndex }
       ::= { hrPrinterTable 1 }

   HrPrinterEntry ::= SEQUENCE {
           hrPrinterStatus             INTEGER,
           hrPrinterDetectedErrorState OCTET STRING
       }

   hrPrinterStatus OBJECT-TYPE
       SYNTAX     INTEGER {
                      other(1),
                      unknown(2),
                      idle(3),
                      printing(4),
                      warmup(5)
                  }
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "The current status of this printer device."
       ::= { hrPrinterEntry 1 }

   hrPrinterDetectedErrorState OBJECT-TYPE
       SYNTAX     OCTET STRING
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "This object represents any error conditions detected
           by the printer.  The error conditions are encoded as
           bits in an octet string, with the following
           definitions:

                Condition         Bit #

                lowPaper              0
```

```
                noPaper              1
                lowToner             2
                noToner              3
                doorOpen             4
                jammed               5
                offline              6
                serviceRequested     7
                inputTrayMissing     8
                outputTrayMissing    9
                markerSupplyMissing  10
                outputNearFull       11
                outputFull           12
                inputTrayEmpty       13
                overduePreventMaint  14

            Bits are numbered starting with the most significant
            bit of the first byte being bit 0, the least
            significant bit of the first byte being bit 7, the
            most significant bit of the second byte being bit 8,
            and so on.  A one bit encodes that the condition was
            detected, while a zero bit encodes that the condition
            was not detected.

            This object is useful for alerting an operator to
            specific warning or error conditions that may occur,
            especially those requiring human intervention."
        ::= { hrPrinterEntry 2 }

    hrDiskStorageTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF HrDiskStorageEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "The (conceptual) table of long-term storage devices
            contained by the host.  In particular, disk devices
            accessed remotely over a network are not included
            here.

            Note that this table is potentially sparse: a
            (conceptual) entry exists only if the correspondent
            value of the hrDeviceType object is
            'hrDeviceDiskStorage'."
        ::= { hrDevice 6 }

    hrDiskStorageEntry OBJECT-TYPE
        SYNTAX     HrDiskStorageEntry
        MAX-ACCESS not-accessible
        STATUS     current
```

```
        DESCRIPTION
            "A (conceptual) entry for one long-term storage device
            contained by the host.  The hrDeviceIndex in the index
            represents the entry in the hrDeviceTable that
            corresponds to the hrDiskStorageEntry. As an example,
            an instance of the hrDiskStorageCapacity object might
            be named hrDiskStorageCapacity.3"
        INDEX { hrDeviceIndex }
        ::= { hrDiskStorageTable 1 }

    HrDiskStorageEntry ::= SEQUENCE {
            hrDiskStorageAccess         INTEGER,
            hrDiskStorageMedia          INTEGER,
            hrDiskStorageRemoveble      TruthValue,
            hrDiskStorageCapacity       KBytes
        }

    hrDiskStorageAccess OBJECT-TYPE
        SYNTAX     INTEGER {
                       readWrite(1),
                       readOnly(2)
                   }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "An indication if this long-term storage device is
            readable and writable or only readable.  This should
            reflect the media type, any write-protect mechanism,
            and any device configuration that affects the entire
            device."
        ::= { hrDiskStorageEntry 1 }

    hrDiskStorageMedia OBJECT-TYPE
        SYNTAX     INTEGER {
                       other(1),
                       unknown(2),
                       hardDisk(3),
                       floppyDisk(4),
                       opticalDiskROM(5),
                       opticalDiskWORM(6),     -- Write Once Read Many
                       opticalDiskRW(7),
                       ramDisk(8)
                   }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "An indication of the type of media used in this long-
            term storage device."
```

```
        ::= { hrDiskStorageEntry 2 }

    hrDiskStorageRemoveble OBJECT-TYPE
        SYNTAX     TruthValue
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "Denotes whether or not the disk media may be removed
            from the drive."
        ::= { hrDiskStorageEntry 3 }

    hrDiskStorageCapacity OBJECT-TYPE
        SYNTAX     KBytes
        UNITS      "KBytes"
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The total size for this long-term storage device. If
            the media is removable and is currently removed, this
            value should be zero."
        ::= { hrDiskStorageEntry 4 }

    hrPartitionTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF HrPartitionEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "The (conceptual) table of partitions for long-term
            storage devices contained by the host.  In particular,
            partitions accessed remotely over a network are not
            included here."
        ::= { hrDevice 7 }

    hrPartitionEntry OBJECT-TYPE
        SYNTAX     HrPartitionEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "A (conceptual) entry for one partition.  The
            hrDeviceIndex in the index represents the entry in the
            hrDeviceTable that corresponds to the
            hrPartitionEntry.

            As an example of how objects in this table are named,
            an instance of the hrPartitionSize object might be
            named hrPartitionSize.3.1"
        INDEX { hrDeviceIndex, hrPartitionIndex }
        ::= { hrPartitionTable 1 }
```

```
   HrPartitionEntry ::= SEQUENCE {
           hrPartitionIndex                Integer32,
           hrPartitionLabel                InternationalDisplayString,
           hrPartitionID                   OCTET STRING,
           hrPartitionSize                 KBytes,
           hrPartitionFSIndex              Integer32
       }

   hrPartitionIndex OBJECT-TYPE
       SYNTAX     Integer32 (1..2147483647)
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "A unique value for each partition on this long-term
           storage device.  The value for each long-term storage
           device must remain constant at least from one re-
           initialization of the agent to the next re-
           initialization."
       ::= { hrPartitionEntry 1 }

   hrPartitionLabel OBJECT-TYPE
       SYNTAX     InternationalDisplayString (SIZE (0..128))
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "A textual description of this partition."
       ::= { hrPartitionEntry 2 }

   hrPartitionID OBJECT-TYPE
       SYNTAX     OCTET STRING
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "A descriptor which uniquely represents this partition
           to the responsible operating system.  On some systems,
           this might take on a binary representation."
       ::= { hrPartitionEntry 3 }

   hrPartitionSize OBJECT-TYPE
       SYNTAX     KBytes
       UNITS      "KBytes"
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "The size of this partition."
       ::= { hrPartitionEntry 4 }

   hrPartitionFSIndex OBJECT-TYPE
```

```
        SYNTAX     Integer32 (0..2147483647)
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The index of the file system mounted on this
            partition.  If no file system is mounted on this
            partition, then this value shall be zero.  Note that
            multiple partitions may point to one file system,
            denoting that that file system resides on those
            partitions.  Multiple file systems may not reside on
            one partition."
        ::= { hrPartitionEntry 5 }

    -- The File System Table

    -- Registration point for popular File System types,
    -- for use with hrFSType. These are defined in the
    -- HOST-RESOURCES-TYPES module.
    hrFSTypes               OBJECT IDENTIFIER ::= { hrDevice 9 }

    hrFSTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF HrFSEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "The (conceptual) table of file systems local to this
            host or remotely mounted from a file server.  File
            systems that are in only one user's environment on a
            multi-user system will not be included in this table."
        ::= { hrDevice 8 }

    hrFSEntry OBJECT-TYPE
        SYNTAX     HrFSEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "A (conceptual) entry for one file system local to
            this host or remotely mounted from a file server.
            File systems that are in only one user's environment
            on a multi-user system will not be included in this
            table.

            As an example of how objects in this table are named,
            an instance of the hrFSMountPoint object might be
            named hrFSMountPoint.3"
        INDEX { hrFSIndex }
        ::= { hrFSTable 1 }
```

```
   HrFSEntry ::= SEQUENCE {
           hrFSIndex                   Integer32,
           hrFSMountPoint              InternationalDisplayString,
           hrFSRemoteMountPoint        InternationalDisplayString,
           hrFSType                    AutonomousType,
           hrFSAccess                  INTEGER,
           hrFSBootable                TruthValue,
           hrFSStorageIndex            Integer32,
           hrFSLastFullBackupDate      DateAndTime,
           hrFSLastPartialBackupDate   DateAndTime
       }

   hrFSIndex OBJECT-TYPE
       SYNTAX     Integer32 (1..2147483647)
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "A unique value for each file system local to this
           host.  The value for each file system must remain
           constant at least from one re-initialization of the
           agent to the next re-initialization."
       ::= { hrFSEntry 1 }

   hrFSMountPoint OBJECT-TYPE
       SYNTAX     InternationalDisplayString (SIZE(0..128))
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "The path name of the root of this file system."
       ::= { hrFSEntry 2 }

   hrFSRemoteMountPoint OBJECT-TYPE
       SYNTAX     InternationalDisplayString (SIZE(0..128))
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "A description of the name and/or address of the
           server that this file system is mounted from.  This
           may also include parameters such as the mount point on
           the remote file system.  If this is not a remote file
           system, this string should have a length of zero."
       ::= { hrFSEntry 3 }

   hrFSType OBJECT-TYPE
       SYNTAX     AutonomousType
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
```

```
            "The value of this object identifies the type of this
            file system."
        ::= { hrFSEntry 4 }

    hrFSAccess OBJECT-TYPE
        SYNTAX     INTEGER {
                       readWrite(1),
                       readOnly(2)
                   }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "An indication if this file system is logically
            configured by the operating system to be readable and
            writable or only readable.  This does not represent
            any local access-control policy, except one that is
            applied to the file system as a whole."
        ::= { hrFSEntry 5 }

    hrFSBootable OBJECT-TYPE
        SYNTAX     TruthValue
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "A flag indicating whether this file system is
            bootable."
        ::= { hrFSEntry 6 }

    hrFSStorageIndex OBJECT-TYPE
        SYNTAX     Integer32 (0..2147483647)
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The index of the hrStorageEntry that represents
            information about this file system.  If there is no
            such information available, then this value shall be
            zero.  The relevant storage entry will be useful in
            tracking the percent usage of this file system and
            diagnosing errors that may occur when it runs out of
            space."
        ::= { hrFSEntry 7 }

    hrFSLastFullBackupDate OBJECT-TYPE
        SYNTAX     DateAndTime
        MAX-ACCESS read-write
        STATUS     current
        DESCRIPTION
            "The last date at which this complete file system was
```

```
            copied to another storage device for backup.  This
            information is useful for ensuring that backups are
            being performed regularly.

            If this information is not known, then this variable
            shall have the value corresponding to January 1, year
            0000, 00:00:00.0, which is encoded as
            (hex)'00 00 01 01 00 00 00 00'."
        ::= { hrFSEntry 8 }

    hrFSLastPartialBackupDate OBJECT-TYPE
        SYNTAX     DateAndTime
        MAX-ACCESS read-write
        STATUS     current
        DESCRIPTION
            "The last date at which a portion of this file system
            was copied to another storage device for backup.  This
            information is useful for ensuring that backups are
            being performed regularly.

            If this information is not known, then this variable
            shall have the value corresponding to January 1, year
            0000, 00:00:00.0, which is encoded as
            (hex)'00 00 01 01 00 00 00 00'."
        ::= { hrFSEntry 9 }

    -- The Host Resources Running Software Group
    --
    -- The hrSWRunTable contains an entry for each distinct piece of
    -- software that is running or loaded into physical or virtual
    -- memory in preparation for running.  This includes the host's
    -- operating system, device drivers, and applications.

    hrSWOSIndex OBJECT-TYPE
        SYNTAX     Integer32 (1..2147483647)
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The value of the hrSWRunIndex for the hrSWRunEntry
            that represents the primary operating system running
            on this host.  This object is useful for quickly and
            uniquely identifying that primary operating system."
        ::= { hrSWRun 1 }

    hrSWRunTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF HrSWRunEntry
        MAX-ACCESS not-accessible
        STATUS     current
```

```
        DESCRIPTION
            "The (conceptual) table of software running on the
            host."
        ::= { hrSWRun 2 }

    hrSWRunEntry OBJECT-TYPE
        SYNTAX     HrSWRunEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "A (conceptual) entry for one piece of software
            running on the host Note that because the installed
            software table only contains information for software
            stored locally on this host, not every piece of
            running software will be found in the installed
            software table.  This is true of software that was
            loaded and run from a non-local source, such as a
            network-mounted file system.

            As an example of how objects in this table are named,
            an instance of the hrSWRunName object might be named
            hrSWRunName.1287"
        INDEX { hrSWRunIndex }
        ::= { hrSWRunTable 1 }

    HrSWRunEntry ::= SEQUENCE {
            hrSWRunIndex       Integer32,
            hrSWRunName        InternationalDisplayString,
            hrSWRunID          ProductID,
            hrSWRunPath        InternationalDisplayString,
            hrSWRunParameters  InternationalDisplayString,
            hrSWRunType        INTEGER,
            hrSWRunStatus      INTEGER
        }

    hrSWRunIndex OBJECT-TYPE
        SYNTAX     Integer32 (1..2147483647)
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "A unique value for each piece of software running on
            the host.  Wherever possible, this should be the
            system's native, unique identification number."
        ::= { hrSWRunEntry 1 }

    hrSWRunName OBJECT-TYPE
        SYNTAX     InternationalDisplayString (SIZE (0..64))
        MAX-ACCESS read-only
```

```
        STATUS     current
        DESCRIPTION
            "A textual description of this running piece of
            software, including the manufacturer, revision,  and
            the name by which it is commonly known.  If this
            software was installed locally, this should be the
            same string as used in the corresponding
            hrSWInstalledName."
        ::= { hrSWRunEntry 2 }

    hrSWRunID OBJECT-TYPE
        SYNTAX     ProductID
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The product ID of this running piece of software."
        ::= { hrSWRunEntry 3 }

    hrSWRunPath OBJECT-TYPE
        SYNTAX     InternationalDisplayString (SIZE(0..128))
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "A description of the location on long-term storage
            (e.g. a disk drive) from which this software was
            loaded."
        ::= { hrSWRunEntry 4 }

    hrSWRunParameters OBJECT-TYPE
        SYNTAX     InternationalDisplayString (SIZE(0..128))
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "A description of the parameters supplied to this
            software when it was initially loaded."
        ::= { hrSWRunEntry 5 }

    hrSWRunType OBJECT-TYPE
        SYNTAX     INTEGER {
                       unknown(1),
                       operatingSystem(2),
                       deviceDriver(3),
                       application(4)
                   }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The type of this software."
```

```
        ::= { hrSWRunEntry 6 }

    hrSWRunStatus OBJECT-TYPE
        SYNTAX     INTEGER {
                       running(1),
                       runnable(2),    -- waiting for resource
                                       -- (i.e., CPU, memory, IO)
                       notRunnable(3), -- loaded but waiting for event
                       invalid(4)      -- not loaded
                   }
        MAX-ACCESS read-write
        STATUS     current
        DESCRIPTION
            "The status of this running piece of software.
            Setting this value to invalid(4) shall cause this
            software to stop running and to be unloaded. Sets to
            other values are not valid."
        ::= { hrSWRunEntry 7 }

    -- The Host Resources Running Software Performance Group
    --
    -- The hrSWRunPerfTable contains an entry corresponding to
    -- each entry in the hrSWRunTable.

    hrSWRunPerfTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF HrSWRunPerfEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "The (conceptual) table of running software
            performance metrics."
        ::= { hrSWRunPerf 1 }

    hrSWRunPerfEntry OBJECT-TYPE
        SYNTAX     HrSWRunPerfEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "A (conceptual) entry containing software performance
            metrics.  As an example, an instance of the
            hrSWRunPerfCPU object might be named
            hrSWRunPerfCPU.1287"
        AUGMENTS { hrSWRunEntry }  -- This table augments information in
                                   -- the hrSWRunTable.
        ::= { hrSWRunPerfTable 1 }

    HrSWRunPerfEntry ::= SEQUENCE {
            hrSWRunPerfCPU          Integer32,
```

```
        hrSWRunPerfMem          KBytes
    }

    hrSWRunPerfCPU OBJECT-TYPE
        SYNTAX     Integer32 (0..2147483647)
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The number of centi-seconds of the total system's CPU
            resources consumed by this process.  Note that on a
            multi-processor system, this value may increment by
            more than one centi-second in one centi-second of real
            (wall clock) time."
        ::= { hrSWRunPerfEntry 1 }

    hrSWRunPerfMem OBJECT-TYPE
        SYNTAX     KBytes
        UNITS      "KBytes"
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The total amount of real system memory allocated to
            this process."
        ::= { hrSWRunPerfEntry 2 }

    -- The Host Resources Installed Software Group
    --
    -- The hrSWInstalledTable contains an entry for each piece
    -- of software installed in long-term storage (e.g. a disk
    -- drive) locally on this host.  Note that this does not
    -- include software loadable remotely from a network
    -- server.
    --
    -- Different implementations may track software in varying
    -- ways. For example, while some implementations may track
    -- executable files as distinct pieces of software, other
    -- implementations may use other strategies such as keeping
    -- track of software "packages" (e.g., related groups of files)
    -- or keeping track of system or application "patches".
    --
    -- This table is useful for identifying and inventorying
    -- software on a host and for diagnosing incompatibility
    -- and version mismatch problems between various pieces
    -- of hardware and software.

    hrSWInstalledLastChange OBJECT-TYPE
        SYNTAX     TimeTicks
        MAX-ACCESS read-only
```

```
        STATUS     current
        DESCRIPTION
            "The value of sysUpTime when an entry in the
            hrSWInstalledTable was last added, renamed, or
            deleted.  Because this table is likely to contain many
            entries, polling of this object allows a management
            station to determine when re-downloading of the table
            might be useful."
        ::= { hrSWInstalled 1 }

    hrSWInstalledLastUpdateTime OBJECT-TYPE
        SYNTAX     TimeTicks
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The value of sysUpTime when the hrSWInstalledTable
            was last completely updated.  Because caching of this
            data will be a popular implementation strategy,
            retrieval of this object allows a management station
            to obtain a guarantee that no data in this table is
            older than the indicated time."
        ::= { hrSWInstalled 2 }

    hrSWInstalledTable OBJECT-TYPE
        SYNTAX     SEQUENCE OF HrSWInstalledEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "The (conceptual) table of software installed on this
            host."
        ::= { hrSWInstalled 3 }

    hrSWInstalledEntry OBJECT-TYPE
        SYNTAX     HrSWInstalledEntry
        MAX-ACCESS not-accessible
        STATUS     current
        DESCRIPTION
            "A (conceptual) entry for a piece of software
            installed on this host.

            As an example of how objects in this table are named,
            an instance of the hrSWInstalledName object might be
            named hrSWInstalledName.96"
        INDEX { hrSWInstalledIndex }
        ::= { hrSWInstalledTable 1 }

    HrSWInstalledEntry ::= SEQUENCE {
            hrSWInstalledIndex       Integer32,
```

```
            hrSWInstalledName        InternationalDisplayString,
            hrSWInstalledID          ProductID,
            hrSWInstalledType        INTEGER,
            hrSWInstalledDate        DateAndTime
    }

    hrSWInstalledIndex OBJECT-TYPE
        SYNTAX     Integer32 (1..2147483647)
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "A unique value for each piece of software installed
            on the host.  This value shall be in the range from 1
            to the number of pieces of software installed on the
            host."
        ::= { hrSWInstalledEntry 1 }

    hrSWInstalledName OBJECT-TYPE
        SYNTAX     InternationalDisplayString (SIZE (0..64))
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "A textual description of this installed piece of
            software, including the manufacturer, revision, the
            name by which it is commonly known, and optionally,
            its serial number."
        ::= { hrSWInstalledEntry 2 }

    hrSWInstalledID OBJECT-TYPE
        SYNTAX     ProductID
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The product ID of this installed piece of software."
        ::= { hrSWInstalledEntry 3 }

    hrSWInstalledType OBJECT-TYPE
        SYNTAX     INTEGER {
                       unknown(1),
                       operatingSystem(2),
                       deviceDriver(3),
                       application(4)
                   }
        MAX-ACCESS read-only
        STATUS     current
        DESCRIPTION
            "The type of this software."
        ::= { hrSWInstalledEntry 4 }
```

```
   hrSWInstalledDate OBJECT-TYPE
       SYNTAX     DateAndTime
       MAX-ACCESS read-only
       STATUS     current
       DESCRIPTION
           "The last-modification date of this application as it
           would appear in a directory listing.

           If this information is not known, then this variable
           shall have the value corresponding to January 1, year
           0000, 00:00:00.0, which is encoded as
           (hex)'00 00 01 01 00 00 00 00'."
       ::= { hrSWInstalledEntry 5 }

   -- Conformance information

   hrMIBCompliances OBJECT IDENTIFIER ::= { hrMIBAdminInfo 2 }
   hrMIBGroups      OBJECT IDENTIFIER ::= { hrMIBAdminInfo 3 }

   -- Compliance Statements
   hrMIBCompliance MODULE-COMPLIANCE
       STATUS current
       DESCRIPTION
           "The requirements for conformance to the Host Resources MIB."
       MODULE -- this module
         MANDATORY-GROUPS { hrSystemGroup, hrStorageGroup,
                            hrDeviceGroup }

         OBJECT hrSystemDate
             MIN-ACCESS read-only
             DESCRIPTION
                 "Write access is not required."

         OBJECT hrSystemInitialLoadDevice
             MIN-ACCESS read-only
             DESCRIPTION
                 "Write access is not required."

         OBJECT hrSystemInitialLoadParameters
             MIN-ACCESS read-only
             DESCRIPTION
                 "Write access is not required."

         OBJECT hrStorageSize
             MIN-ACCESS read-only
             DESCRIPTION
                 "Write access is not required."
```

```
        OBJECT hrFSLastFullBackupDate
            MIN-ACCESS read-only
            DESCRIPTION
                "Write access is not required."

        OBJECT hrFSLastPartialBackupDate
            MIN-ACCESS read-only
            DESCRIPTION
                "Write access is not required."

        GROUP hrSWRunGroup
            DESCRIPTION
                "The Running Software Group. Implementation
                of this group is mandatory only when the
                hrSWRunPerfGroup is implemented."

        OBJECT hrSWRunStatus
            MIN-ACCESS read-only
            DESCRIPTION
                "Write access is not required."

        GROUP hrSWRunPerfGroup
            DESCRIPTION
                "The Running Software Performance Group.
                Implementation of this group is at the discretion
                of the implementor."

        GROUP hrSWInstalledGroup
            DESCRIPTION
                "The Installed Software Group.
                Implementation of this group is at the discretion
                of the implementor."

    ::= { hrMIBCompliances 1 }

    hrSystemGroup OBJECT-GROUP
        OBJECTS {
            hrSystemUptime, hrSystemDate,
            hrSystemInitialLoadDevice,
            hrSystemInitialLoadParameters,
            hrSystemNumUsers, hrSystemProcesses,
            hrSystemMaxProcesses
        }
        STATUS current
        DESCRIPTION
            "The Host Resources System Group."
        ::= { hrMIBGroups 1 }
```

```
    hrStorageGroup OBJECT-GROUP
        OBJECTS {
            hrMemorySize, hrStorageIndex, hrStorageType,
            hrStorageDescr, hrStorageAllocationUnits,
            hrStorageSize, hrStorageUsed,
            hrStorageAllocationFailures
        }
        STATUS current
        DESCRIPTION
            "The Host Resources Storage Group."
        ::= { hrMIBGroups 2 }

    hrDeviceGroup OBJECT-GROUP
        OBJECTS {
            hrDeviceIndex, hrDeviceType, hrDeviceDescr,
            hrDeviceID, hrDeviceStatus, hrDeviceErrors,
            hrProcessorFrwID, hrProcessorLoad,
            hrNetworkIfIndex, hrPrinterStatus,
            hrPrinterDetectedErrorState,
            hrDiskStorageAccess, hrDiskStorageMedia,
            hrDiskStorageRemoveble, hrDiskStorageCapacity,
            hrPartitionIndex, hrPartitionLabel,
            hrPartitionID, hrPartitionSize,
            hrPartitionFSIndex, hrFSIndex, hrFSMountPoint,
            hrFSRemoteMountPoint, hrFSType, hrFSAccess,
            hrFSBootable, hrFSStorageIndex,
            hrFSLastFullBackupDate,
            hrFSLastPartialBackupDate
        }
        STATUS current
        DESCRIPTION
            "The Host Resources Device Group."
        ::= { hrMIBGroups 3 }

    hrSWRunGroup OBJECT-GROUP
        OBJECTS {
            hrSWOSIndex, hrSWRunIndex, hrSWRunName,
            hrSWRunID, hrSWRunPath, hrSWRunParameters,
            hrSWRunType, hrSWRunStatus
        }
        STATUS current
        DESCRIPTION
            "The Host Resources Running Software Group."
        ::= { hrMIBGroups 4 }

    hrSWRunPerfGroup OBJECT-GROUP
        OBJECTS { hrSWRunPerfCPU, hrSWRunPerfMem }
        STATUS current
```

```
            DESCRIPTION
                "The Host Resources Running Software
                Performance Group."
            ::= { hrMIBGroups 5 }

        hrSWInstalledGroup OBJECT-GROUP
            OBJECTS {
                hrSWInstalledLastChange,
                hrSWInstalledLastUpdateTime,
                hrSWInstalledIndex, hrSWInstalledName,
                hrSWInstalledID, hrSWInstalledType,
                hrSWInstalledDate
            }
            STATUS current
            DESCRIPTION
                "The Host Resources Installed Software Group."
            ::= { hrMIBGroups 6 }

    END
```

5. Type Definitions

```
    HOST-RESOURCES-TYPES DEFINITIONS ::= BEGIN

    IMPORTS
      MODULE-IDENTITY, OBJECT-IDENTITY        FROM SNMPv2-SMI
      hrMIBAdminInfo, hrStorage, hrDevice     FROM HOST-RESOURCES-MIB;

    hostResourcesTypesModule MODULE-IDENTITY
      LAST-UPDATED "200003060000Z"    -- 6 March, 2000
      ORGANIZATION "IETF Host Resources MIB Working Group"
      CONTACT-INFO
          "Steve Waldbusser
          Postal: Lucent Technologies, Inc.
                  1213 Innsbruck Dr.
                  Sunnyvale, CA 94089
                  USA
          Phone: 650-318-1251
          Fax:   650-318-1633
          Email: waldbusser@ins.com

          In addition, the Host Resources MIB mailing list is dedicated
          to discussion of this MIB. To join the mailing list, send a
          request message to hostmib-request@andrew.cmu.edu. The mailing
          list address is hostmib@andrew.cmu.edu."
      DESCRIPTION
          "This MIB module registers type definitions for
          storage types, device types, and file system types.
```

```
        After the initial revision, this module will be
        maintained by IANA."
    REVISION "200003060000Z"    -- 6 March 2000
    DESCRIPTION
        "The original version of this module, published as RFC
        2790."
    ::= { hrMIBAdminInfo 4 }

-- Registrations for some storage types, for use with hrStorageType
hrStorageTypes          OBJECT IDENTIFIER ::= { hrStorage 1 }

hrStorageOther OBJECT-IDENTITY
    STATUS current
    DESCRIPTION
        "The storage type identifier used when no other defined
        type is appropriate."
    ::= { hrStorageTypes 1 }

hrStorageRam OBJECT-IDENTITY
    STATUS current
    DESCRIPTION
        "The storage type identifier used for RAM."
    ::= { hrStorageTypes 2 }

hrStorageVirtualMemory OBJECT-IDENTITY
    STATUS current
    DESCRIPTION
        "The storage type identifier used for virtual memory,
        temporary storage of swapped or paged memory."
    ::= { hrStorageTypes 3 }

hrStorageFixedDisk OBJECT-IDENTITY
    STATUS current
    DESCRIPTION
        "The storage type identifier used for non-removable
        rigid rotating magnetic storage devices."
    ::= { hrStorageTypes 4 }

hrStorageRemovableDisk OBJECT-IDENTITY
    STATUS current
    DESCRIPTION
        "The storage type identifier used for removable rigid
        rotating magnetic storage devices."
    ::= { hrStorageTypes 5 }

hrStorageFloppyDisk OBJECT-IDENTITY
    STATUS current
    DESCRIPTION
```

```
            "The storage type identifier used for non-rigid rotating
            magnetic storage devices."
        ::= { hrStorageTypes 6 }

    hrStorageCompactDisc OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
            "The storage type identifier used for read-only rotating
            optical storage devices."
        ::= { hrStorageTypes 7 }

    hrStorageRamDisk OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
            "The storage type identifier used for a file system that
            is stored in RAM."
        ::= { hrStorageTypes 8 }

    hrStorageFlashMemory OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
            "The storage type identifier used for flash memory."
        ::= { hrStorageTypes 9 }

    hrStorageNetworkDisk OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
            "The storage type identifier used for a
            networked file system."
        ::= { hrStorageTypes 10 }

    -- Registrations for some device types, for use with hrDeviceType
    hrDeviceTypes             OBJECT IDENTIFIER ::= { hrDevice 1 }

    hrDeviceOther OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
            "The device type identifier used when no other defined
            type is appropriate."
        ::= { hrDeviceTypes 1 }

    hrDeviceUnknown OBJECT-IDENTITY
        STATUS current
        DESCRIPTION
            "The device type identifier used when the device type is
            unknown."
        ::= { hrDeviceTypes 2 }
```

```
   hrDeviceProcessor OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a CPU."
       ::= { hrDeviceTypes 3 }

   hrDeviceNetwork OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a network interface."
       ::= { hrDeviceTypes 4 }

   hrDevicePrinter OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a printer."
       ::= { hrDeviceTypes 5 }

   hrDeviceDiskStorage OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a disk drive."
       ::= { hrDeviceTypes 6 }

   hrDeviceVideo OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a video device."
       ::= { hrDeviceTypes 10 }

   hrDeviceAudio OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for an audio device."
       ::= { hrDeviceTypes 11 }

   hrDeviceCoprocessor OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a co-processor."
       ::= { hrDeviceTypes 12 }

   hrDeviceKeyboard OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a keyboard device."
       ::= { hrDeviceTypes 13 }
```

```
   hrDeviceModem OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a modem."
       ::= { hrDeviceTypes 14 }

   hrDeviceParallelPort OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a parallel port."
       ::= { hrDeviceTypes 15 }

   hrDevicePointing OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a pointing device
           (e.g., a mouse)."
       ::= { hrDeviceTypes 16 }

   hrDeviceSerialPort OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a serial port."
       ::= { hrDeviceTypes 17 }

   hrDeviceTape OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a tape storage device."
       ::= { hrDeviceTypes 18 }

   hrDeviceClock OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a clock device."
       ::= { hrDeviceTypes 19 }

   hrDeviceVolatileMemory OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a volatile memory
           storage device."
       ::= { hrDeviceTypes 20 }

   hrDeviceNonVolatileMemory OBJECT-IDENTITY
       STATUS current
       DESCRIPTION
           "The device type identifier used for a non-volatile memory
```

```
            storage device."
        ::= { hrDeviceTypes 21 }

    -- Registrations for some popular File System types,
    -- for use with hrFSType.
    hrFSTypes               OBJECT IDENTIFIER ::= { hrDevice 9 }

    hrFSOther OBJECT-IDENTITY
        STATUS  current
        DESCRIPTION
            "The file system type identifier used when no other
            defined type is appropriate."
        ::= { hrFSTypes 1 }

    hrFSUnknown OBJECT-IDENTITY
        STATUS  current
        DESCRIPTION
            "The file system type identifier used when the type of
            file system is unknown."
        ::= { hrFSTypes 2 }

    hrFSBerkeleyFFS OBJECT-IDENTITY
        STATUS  current
        DESCRIPTION
            "The file system type identifier used for the
            Berkeley Fast File System."
        ::= { hrFSTypes 3 }

    hrFSSys5FS OBJECT-IDENTITY
        STATUS  current
        DESCRIPTION
            "The file system type identifier used for the
            System V File System."
        ::= { hrFSTypes 4 }

    hrFSFat OBJECT-IDENTITY
        STATUS  current
        DESCRIPTION
            "The file system type identifier used for
            DOS's FAT file system."
        ::= { hrFSTypes 5 }

    hrFSHPFS OBJECT-IDENTITY
        STATUS  current
        DESCRIPTION
            "The file system type identifier used for OS/2's
            High Performance File System."
        ::= { hrFSTypes 6 }
```

```
   hrFSHFS OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           Macintosh Hierarchical File System."
       ::= { hrFSTypes 7 }

   hrFSMFS OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           Macintosh File System."
       ::= { hrFSTypes 8 }

   hrFSNTFS OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           Windows NT File System."
       ::= { hrFSTypes 9 }

   hrFSVNode OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           VNode File System."
       ::= { hrFSTypes 10 }

   hrFSJournaled OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           Journaled File System."
       ::= { hrFSTypes 11 }

   hrFSiso9660 OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           ISO 9660 File System for CD's."
       ::= { hrFSTypes 12 }

   hrFSRockRidge OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           RockRidge File System for CD's."
       ::= { hrFSTypes 13 }
```

```
   hrFSNFS OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           NFS File System."
       ::= { hrFSTypes 14 }

   hrFSNetware OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           Netware File System."
       ::= { hrFSTypes 15 }

   hrFSAFS OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           Andrew File System."
       ::= { hrFSTypes 16 }

   hrFSDFS OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           OSF DCE Distributed File System."
       ::= { hrFSTypes 17 }

   hrFSAppleshare OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           AppleShare File System."
       ::= { hrFSTypes 18 }

   hrFSRFS OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           RFS File System."
       ::= { hrFSTypes 19 }

   hrFSDGCFS OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           Data General DGCFS."
       ::= { hrFSTypes 20 }
```

```
   hrFSBFS OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           SVR4 Boot File System."
       ::= { hrFSTypes 21 }

   hrFSFAT32 OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           Windows FAT32 File System."
       ::= { hrFSTypes 22 }

   hrFSLinuxExt2 OBJECT-IDENTITY
       STATUS  current
       DESCRIPTION
           "The file system type identifier used for the
           Linux EXT2 File System."
       ::= { hrFSTypes 23 }

   END
```

6. Internationalization Considerations

This MIB has many objects that identify file-system pathnames on the managed host. Many file systems allow pathnames to be encoded in a variety of character sets (other than ASCII), but do not support the encoding of the actual character set used with the pathname. The implementation strategy is that user interfaces (i.e. character-based shells or graphical applications) will have configuration options that control with which character set they will interpret and display all pathnames. This is often a per-user configuration (e.g. an environment variable), so that users using different languages and character sets on a multi-user system may each work effectively with their preferred character set. A human usually controls this configuration. If an application is not configured or is configured incorrectly, it will often have trouble displaying pathnames in the intended character set.

This situation made it important for this MIB to handle two issues:

1) Pathname objects must be able to transfer a variety of character sets with potentially multi-byte encodings; and,

2) HostMIB agents will generally not be correctly configured for the appropriate character set to be used for all files on the system, particularly on a system with multiple users using different character sets. It was thus impossible to mandate that the agent tag pathnames with the character set in use.

These issues were solved with the introduction of the InternationalDisplayString textual convention, which supports multi-byte encodings. Network management stations should use a local algorithm to determine which character set is in use and how it should be displayed. It is expected that network management station applications will rely on human configuration to choose which character set in which to interpret InternationalDisplayString objects, much like an application running locally on that host.

7. Security Considerations

There are a number of management objects defined in this MIB that have a MAX-ACCESS clause of read-write. Such objects may be considered sensitive or vulnerable in some network environments. The support for SET operations in a non-secure environment without proper protection can have a negative effect on system operations.

There are a number of managed objects in this MIB that may contain sensitive information. The objects in the Running Software Group list information about running software on the system (including the operating system software and version). Some may wish not to disclose to others what software they are running. Further, an inventory of the running software and versions may be helpful to an attacker who hopes to exploit software bugs in certain applications. The same issues exist for the objects in the Installed Software Group.

It is thus important to control even GET access to these objects and possibly to even encrypt the values of these object when sending them over the network via SNMP. Not all versions of SNMP provide features for such a secure environment.

SNMPv1 by itself is not a secure environment. Even if the network itself is secure (for example by using IPSec), even then, there is no control as to who on the secure network is allowed to access and GET/SET (read/change/create/delete) the objects in this MIB.

It is recommended that the implementers consider the security features as provided by the SNMPv3 framework. Specifically, the use of the User-based Security Model RFC 2574 [RFC2574] and the View-based Access Control Model RFC 2575 [RFC2575] is recommended.

It is then a customer/user responsibility to ensure that the SNMP entity giving access to an instance of this MIB, is properly configured to give access to the objects only to those principals (users) that have legitimate rights to indeed GET or SET (change/create/delete) them.

8. References

[RFC2571] Harrington, D., Presuhn, R. and B. Wijnen, "An Architecture for Describing SNMP Management Frameworks", RFC 2571, April 1999.

[RFC1155] Rose, M. and K. McCloghrie, "Structure and Identification of Management Information for TCP/IP-based Internets", STD 16, RFC 1155, May 1990.

[RFC1212] Rose, M. and K. McCloghrie, "Concise MIB Definitions", STD 16, RFC 1212, March 1991.

[RFC1215] Rose, M., "A Convention for Defining Traps for use with the SNMP", RFC 1215, March 1991.

[RFC2578] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Structure of Management Information Version 2 (SMIv2)", STD 58, RFC 2578, April 1999.

[RFC2579] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Textual Conventions for SMIv2", STD 58, RFC 2579, April 1999.

[RFC2580] McCloghrie, K., Perkins, D., Schoenwaelder, J., Case, J., Rose, M. and S. Waldbusser, "Conformance Statements for SMIv2", STD 58, RFC 2580, April 1999.

[RFC1157] Case, J., Fedor, M., Schoffstall, M. and J. Davin, "Simple Network Management Protocol", STD 15, RFC 1157, May 1990.

[RFC1901] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Introduction to Community-based SNMPv2", RFC 1901, January 1996.

[RFC1906] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Transport Mappings for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1906, January 1996.

[RFC2572] Case, J., Harrington D., Presuhn R. and B. Wijnen, "Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)", RFC 2572, April 1999

[RFC2574] Blumenthal, U. and B. Wijnen, "User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)", RFC 2574, April 1999.

[RFC1905] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Protocol Operations for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1905, January 1996.

[RFC2573] Levi, D., Meyer, P. and B. Stewart, "SNMPv3 Applications", RFC 2573, April 1999.

[RFC2575] Wijnen, B., Presuhn, R. and K. McCloghrie, "View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)", RFC 2575, April 1999.

[RFC2570] Case, J., Mundy, R., Partain, D. and B. Stewart, "Introduction to Version 3 of the Internet- standard Network Management Framework", RFC 2570, April 1999.

[RFC1907] Case, J., McCloghrie, K., Rose, M. and S. Waldbusser, "Management Information Base for Version 2 of the Simple Network Management Protocol (SNMPv2)", RFC 1907, January 1996.

[RFC2233] McCloghrie, K. and F. Kastenholz, "The Interfaces Group MIB", RFC 2233, November 1997.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

9. Acknowledgments

This document was produced by the Host Resources MIB working group.

Bobby Krupczak's efforts were particularly helpful in the creation of the draft standard version of this document.

In addition, the authors gratefully acknowledge the comments of the following individuals:

Amatzia Ben-Artzi	NetManage
Ron Bergman	Hitachi, Inc.
Steve Bostock	Novell
Stephen Bush	GE Information Systems
Jeff Case	SNMP Research
Chuck Davin	Bellcore
Ray Edgarton	Bell Atlantic
Mike Erlinger	Aerospace Corporation
Tim Farley	Magee Enterprises
Mark Kepke	Hewlett Packard
Bobby Krupczak	Empire Technologies, Inc.
Cheryl Krupczak	Empire Technologies, Inc.
Harry Lewis	IBM Corp.
Keith McCloghrie	Cisco Systems
Greg Minshall	Novell
Steve Moulton	SNMP Research
Dave Perkins	Synoptics
Ed Reeder	Objective Systems Integrators
Mike Ritter	Apple Computer
Marshall Rose	Dover Beach Consulting
Jon Saperia	DEC
Rodney Thayer	Sable Technology
Kaj Tesink	Bellcore
Dean Throop	Data General
Bert Wijnen	Lucent
Lloyd Young	Lexmark International

10. Authors' Addresses

Pete Grillo
WeSync.com
1001 SW Fifth Ave, Fifth Floor
Portland, OR 97204

Phone: 503-425-5051
Fax: 503-827-6718
email: pete@wesync.com
Phone: +1 503 827 6717

Steven Waldbusser
Lucent Technologies, Inc.
1213 Innsbruck Dr.
Sunnyvale CA 94089

Phone: +1 650 318 1251
Fax: +1 650 318 1633
EMail: waldbusser@ins.com

11. Intellectual Property

12. Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.