

INTERNET-DRAFT

[draft-hazewinkel-appl-mib-01.txt](#)

July 18, 1997

Harrie Hazewinkel

DESIRE / MUSIQ

CEO Programme

Joint Research Centre of the E.C.

harrie.hazewinkel@jrc.it

July 17 1997

Survey of Defined Managed Objects for Applications Management

<[draft-hazewinkel-appl-mib-01.txt](#)>

Status of this Memo

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as ``work in progress.''

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt' listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Distribution of this document is unlimited. Please send comments to harrie.hazewinkel@jrc.it.

Epires January 18, 1998

[Page 1]

Abstract

This document was produced as the result of a survey on MIBs related to application management. The goal was to identify overlapping or duplicated objects and discover problems within the relationships between those MIBs. The survey covered MIBs already defined and those currently being defined within IETF WGs, namely:

- * the SYSAPPL-MIB,
- * the APPLICATION-MIB,
- * the WWW MIB,
- * the NSM-MIB,
- * the DSA-MIB,
- * the HR-MIB, and
- * the RDBMS-MIB.

It is interesting to note that both the SYSAPPL-MIB and NSM-MIB claim to be generic MIBs (the SYSAPPL-MIB under an OS/host perspective and the NSM-MIB under a networked/distributed perspective). The HR-MIB describes a host as a set of hardware and software groups. On the other hand, either the DSA-MIB, the RDBMS-MIB and the WWW-MIB are application specific MIBs.

There are many overlapping and duplicated objects in the surveyed MIBs. For example, the software groups of the HR-MIB can be replaced with the recently defined SYSAPPL-MIB. Other overlaps or duplications consist mainly of objects which have more detail.

The current MIB definitions pose two major problems:

- 1) indexing with managed objects of other MIBs, and
- 2) mapping of applications on both in the SYSAPPL-MIB and the NSM-MIB.

Implementation problems arise when indexes from other MIBs are used. Is implementation of the MIB containing the objects used for indexes required? Within the current SNMP framework there are no rules or hints given to these questions. If an agent implementor decides not to implement the referred MIB he has to choose his own index values in such a way that no conflicts will occur.

At the moment, host-based applications map cleanly onto the SYSAPPL-MIB and networked applications fit onto the NSM-MIB. However, an application can be made of multiple elements and processes while being a multiple networked applications (services) at the same time. Although a mapping into both MIBs is possible, their relation is not specified anywhere in the existing MIBs. A relationship table is, therefore, required to supplement the existing MIBs. Otherwise, no coherence between the SYSAPPL-MIB and the NSM-MIB

exists.

Epires January 18, 1998

[Page 2]

What's new or updated.

- A) The WWW-MIB sections are changed due to changes of the document itself.
- B) The Application Management MIB is added.
- C) A possible solution with an integration MIB to bind application of the SYSAPPL-MIB and NSM-MIB is added in the conclusions.
- D) The indexing scheme of the RDBMS-MIB is added.

1. Introduction

Distributed applications became increasingly important for organisations. Companies are more geographically spread and their core applications are often connected together across computer networks. The need for instrumentation that provides realtime monitoring and configuration capabilities for these systems is of critical importance. The Simple Network Management Protocol (SNMP) framework can be used to provide this infrastructure.

The SNMP framework is used by operators to manage physical or logical device remotely. The devices are represented through managed objects which are accessed via a virtual information store, termed the Management Information Base or MIB. This MIB (and the SNMP framework) allows extensions of new defined managed objects according to a standard scheme.

The IETF continues to make efforts leading to the definition of MIBs that can be used for applications. This internet draft analysis those efforts and describes:

- 1) contents of application MIBs,
- 2) duplicated and overlapping objects and the
- 3) indexing scheme of application related MIBs.

The following MIBs are analysed:

- * the SYSAPPL-MIB,
- * the APPLCIATION-MIB,
- * the WWW MIB,
- * the NSM-MIB,
- * the DSA-MIB,
- * the HR-MIB, and
- * the RDBMS-MIB.

1.1 How to read this internet draft.

This document relates the continuing IETF efforts in application MIB definition with the related. This internet-draft is organised as follows:

- * [Section 2](#) describes the Working Groups (WG) of the surveyed MIBs.
- * [Section 3](#) describes a summary of the surveyed MIBs.
- * [Section 4](#) describes the similarities of the MIBs.
- * [Section 5](#) describes the indexing scheme within the current MIBs

2. The Working Groups of Application Related MIBs

In this section we provide a short description of the IETF Working Groups working application management related MIBs and present the current status of their work. At this moment only the SYSAPPL WG and the MADMAN WG are defining management objects to manage applications.

The APPLMIB WG is chartered to define managed objects for the monitoring and control of distributed applications. The desire was to support distributed applications, but not at the expense of creating a MIB which cannot or will not be implemented. Consensus was reached on the fact that the APPLMIB WG would focus initially on applications resident in a single host, but that there should be a possibility of additional growth. Features for distributed applications will be added unless the required support would make the MIB over complex. The current status is that the first MIB, the SYSAPPL-MIB which does not require instrumentation of the applications for its population, is ready. Currently, the APPLMIB WG works on two MIBs, an application management MIB requiring instrumentation within applications and a WWW-MIB for monitoring and control of Web servers.

The MADMAN WG is chartered to review the managed objects already defined by the WG in January 1994. The WG in 1994 finished their work under the Network Management Area and resulted in 4 internet drafts which are the Network Services Monitoring MIB, the X.500 Directory Monitoring MIB, the Mail Monitoring MIB and Directory Alarms MIB. In this survey only the first two MIBs are described. The review is currently in its final stage and some changes have been done the original documents.

The Host Resources WG was chartered to define a set of managed objects useful for management of host computers, the Host Resources (HR) MIB. Host computers were defined as a system independent of the operating system, the network service and the applications running on the host. An important requirement during the definition of the HR MIB was that the managed object had to be system independent. The HR WG has finished their work and results are in [RFC 1514](#).

The Relational DataBase Management System WG was chartered to define a set of managed objects useful for the management of database systems. The work is finished and results published in [RFC 1697](#).

Epires January 18, 1998

[Page 4]

3. The application related MIBs.

This section is a summary of the documents describing the definition of application related MIBs.

3.1 The System Application MIB

The SYSAPPL-MIB is designed to provide the requirements for monitoring the behaviour, the faults and the performance of applications. The design is made in such a way that no instrumentation of an application is required. The MIB defines managed objects under an operating system and a file-system perspective. The structure of the SYSAPPL-MIB reflects four groups of applications within a system

- 1) the installed applications
- 2) the applications that are currently running on the system
- 3) the applications that have run on the system in the past
- 4) the mapping from running towards the installed elements

The Installed Applications Group.

--- -----

The installed applications group provides information on the software installed on a system. This group has a highly static character, since installed applications tend to not change very fast. The group makes a distinction between an application and the elements from which it is composed. By using the information provided by this group a user can obtain information about the currently running elements (processes) or those elements (processes) that have run in applications.

The group consists of:

- * "sysApplInstallPkgTable" provides information on the installed packages such as name, version, location and installed date.
- * "sysApplInstallElmtTable" provides detailed information on the application by means of a set of elements acting together as an application. Each element is described in an entry and provides information such as whether the element is executable, name, installed date.

The Applications Run Group.

--- -----

The applications run group reflects the dynamic information from the applications on a system. It describes applications that are currently running or that have run on a system. The elements provide historic

information about the applications. The group is divided in the same fashion as the installed application group, i.e. an application table and an element table. Besides that division, the tables are also divided into applications which are running or have been run.

Epires January 18, 1998

[Page 5]

Due to the historical character of this group, the 'have run' tables can consume an unlimited size and number of resources. Therefore, managed objects are added to control and limit the amount of used resources.

The group defines the following objects:

- * "sysApplRunTable" provides information on applications currently running on the system. For instance, the time an application started running, the number files it has open and the number of connections it has.
- * "sysApplPastRunTable" provides historical information about applications that have been running. For instance, the time an application started running, the time it stopped and the last known number of connections.
- * "sysApplElmtRunTable" provides information on all processes running and (if known) to which application a process belongs. For instance, the time the process started running and the number of connections the process has.
- * "sysApplElmtPastRunTable" provides information on processes that have been running. For instance, the time a process started running and the time it stopped running.
- * "sysApplPastRunTableSizeLimit" provides the maximum number of entries in the "sysApplPastRunTable". When this limit is reached the oldest entry will be removed.
- * "sysApplPastRunTableItemsRem" provides a count of the number of items removed from the "sysApplPastRunTable".
- * "sysApplPastRunTimeLimit" provides the minimum time that should pass before an entry may be removed from the "sysApplPastRunTable".
- * "sysApplElemPastRunSizeLimit" provides the maximum size of entry growth of the "sysApplElmtPastRunTable" before the oldest entry will be removed.
- * "sysApplElmtPastRunTableItemsRem" provides a count of the number of items removed from the "sysApplElmtPastRunTable".
- * "sysApplPastRunTimeLimit" provid the minimum amount of time before an entry may be removed from the "sysApplPastRunTable".
- * "sysApplPollInterval" provides the minimum time in seconds that an agent will poll the status of the managed resources.

The Mapping Group

--- -----

The mapping group provides an easy mapping from running elements in the running element table towards the installed elements table.

The managed objects are:

- * "sysApplMapTable" provides the mapping from a running element entry to an installed element entry.

3.2 The Application Management MIB

The Application Management MIB defines a set of managed object to monitor 'internal' application behavior. This requires that applications has to be instrumented in order to populate this MIB. This is in contrast of the SYSAPPL-MIB which does not require the instrumentation of applications. This MIB focusses on the management of running elements. Therefore, this MIB should be used in conjunction with SYSAPPL-MIB.

The structure of the MIB reflects the throughput, reports of unit of work, and generic resource management and can be grouped like:

- 1) relations group,
- 2) files group,
- 3) connections group,
- 4) transaction statistics group,
- 5) running element behavior.

The Relation Group

--- -----

The relation group connects the 'active' running elements of an application to the networked applications as defined in the NSM-MIB. This mapping is required, since there is no 1:1 relationship between the networked application and the running element. Multiple relations (both ways) are possible.

The managed objects are:

- * "nsmToSysAppElmtRunTable" provides a linking between running elements as managed in this MIB and both the NSM-MIB and SYSAPPL-MIB.

The files group

--- -----

This group provides information about the use of files. The group is organised per running application element and connects them to the open files.

The managed objects of this group are:

- * "applOpenFileTable" provides management information about the file and in particular the activity of the application element with this file.
The table consists of objects like the number of read requests, last time the file was read and the numebr of bytes associated with the read requests.
- * "applOpenFileXrefTable" provides an alternative for the indexing used in the "applOpenFileTable".

The connections group

--- -----

The connections group maintains usefull management information about the current and previous network connections at a per running element sorting. Although, the description of the group implies a very tight connection to the NSM-MIB.

The managed objects of this group are:

- * "applOpenConnectionTable" provides information about connections of a running element. This table explicitly provides only managed information of the connections and it doesn't interpret the passed information in any form.
- * "applExConnectionControlTable" allows a manager to set the boundaries of the "applExConnectionTable". The boundaries are the amount and the 'time-to-live' of the entries in the "applExConnectionTable".
- * "applExConnectionTable" provides a history of the connection of the running elements. Entries exists only in this table after the connection is closed and existed in the "applOpenConnectionTable".

The Transaction Group

--- -----

The transaction group provides besides the connection group also network traffic related management information. This group looks with detail to the traffic past at the connections of the running element.

The managed objects are:

- * "applTransactionControltable" provides a manager a to set up for which open files or connections transactions are defined and the unit of work.
- * "applTransactionStreamTable" provides information about units of work in streams, such as number of requests, number of responses and the accumulated time between tehm seen from the running element.

The running element behavior group

--- -----

This group provides monitor and control functionality for the running elements within the system.

The managed objects are:

- * "applElmtRunStatusTable" provides the ability to monitor the behavior of the running element.
- * "applElmtRunControlTable" provides the control to, for instance, suspend or reconfigure the running element.

3.3 The World Wide Web MIB

The WWW-MIB is designed to fulfil the requirements to manage entities responsible for documents exchange over the World Wide Web. The emphasis of the definitions is at the service provided by the entity. An entity is defined as a logical device which can behave as a client, a server, a proxy or a cached-proxy accessible via an unique fully qualified domain name and service port. Between the entities documents are transported through a 'Document Transfer Protocol' (DTP) which is based on the request/response paradigm and abstracts from the actual used protocol. Therefore, the WWW-MIB can be used in a wide range of document transfer protocols and also allows the introduction of new DTPs based on the client-server paradigm. HTTP, FTP and NNTP are examples of such protocols.

The structure of the WWW-MIB reflects the four major groups:

- 1) entity administration ("wwwSystem"),
- 2) DTP traffic ("wwwServiceStatistics"),
- 3) accessed or stored document ("wwwDocuments").

The System Group

--- -----

The system group provides general information on the WWW entities in the system. Those entities exchange messages (PDUs) between each other with the abstract defined 'Document Transfer Protocol'.

The group consists of one table.

- * "wwwEntityTable" provides general administrative information of an entity. For instance, contact address, the used 'DTP' protocol and the fully qualified domain name the entity listens to. The index used to uniquely define the entities is an WWW-MIB internal arbitrary number.

The Service Statistics Group

--- -----

The service statistics group provides network management information on network traffic transmitted and/or received by the WWW-entity.

The group consists of the following tables.

- * "wwwSummaryTable" provides a summary on the network traffic statistics of the associated entity. The index used is equal to the "wwwEntityTable". For instance, the number of requests received or transmitted and the number of unknown protocol type messages.
- * "wwwRequestInTable" provides the statistics on the received requests whereby for each request type statistics is maintained. The indexing of this table is defined by the "wwwRequestInIndex"

and the "wwwEntityIndex" which associates the requesttype with an entity. For instance, a recieved GET-request of HTTP is kept in an entry.

* "wwwRequestOutTable" provides the statistics on the transmitted

responses whereby for each response type statistics is maintained. The indexing of this table is defined by the "wwwRequestInIndex" and the "wwwEntityIndex" which associates the requesttype with an entity.

For instance, a transmitted GET-request of HTTP is kept in an entry.

- * "wwwResponseInTable" provides the statistics on the received responses whereby for each response type statistics is maintained. The indexing of this table is defined by the "wwwResponseInIndex" and the "wwwEntityIndex" which associates the responsetype with an entity. For instance, a received 200-reply (OK) of HTTP is kept in an entry.
- * "wwwResponseOutTable" provides the statistics on the transmitted responses whereby for each response type statistics is maintained. The indexing of this table is defined by the "wwwResponseOutIndex" and the "wwwEntityIndex" which associates the responsetype with an entity. For instance, a transmitted 200-reply (OK) of HTTP is kept in an entry.

The Document Group

--- -----

The "wwwDocumentGroup" defines management information on the documents which in the past were attempted to read. The management information is ordered by the entities and has a bucket approach. The bucket approach is used to reduce unmeaningfull information, but document statistics should still be possible.

The document group has the following defined objects:

- * "wwwDocCtrlTable" provides the manager the possiblity to configure the document statistics tables, such as table size and bucket creation time.
- * "wwwDocLastNTable" provides a list of the last N documents which were attempted to read.
- * "wwwDocBucketTable" provides the summary statistics of the read attempts which were captured by the bucket.
- * "wwwDocBucketTopNtable" provides a 'top N' of the documents which were attempted to read. The sort order with which the 'top N' is created can be configured with the "wwwDocCtrlTable".

3.4 The Network Services Monitoring MIB

The NSM-MIB is designed to monitor the behaviour of networked applications. No application specific objects are defined in the NSM-MIB. Therefore, it can be used for a wide range of applications which make use of network services and are distributed.

The structure of the NSM-MIB reflects the coherent behavior of applications in conjunction with the network service on which the applications rely.

Epires January 18, 1998

[Page 10]

The NSM-MIB definitions consists of the following objects.

- * "applTable" provides information on the networked application. For instance, the name of the application, the number of inbound and outbound connections.
- * "assocTable" provides information on the associations with the other/remote applications. For instance, the remote address and the duration.

3.5 The X.500 Directory Monitoring MIB

The X.500 Directory Monitoring MIB (DSA-MIB) is designed to manage a Directory Systems Agent (DSA), a component of the OSI Directory Framework. The MIB allows managers to monitor operational aspects and performance of a DSA. The DSA-MIB is an application specific and relies on the NSM-MIB.

The structure of the DSA-MIB reflects only the DSA-applications specific parts in 3 defined object groups the DSA operations, the DSA statistics and cache performance and the interaction with peer DSAs.

The DSA-MIB definitions consist of the following objects.

- * "dsaOpsTable" provides the summary information on accesses, operations and errors occurred within the DSA.
- * "dsaEntriesTable" provides statistical and cache performance information on DSA applications.
- * "dsaIntTable" provides some information on interactions done with their respective peer DSAs. These interaction are done in the past by the monitored DSAs.

3.6 The Host Resources MIB

The Host Resources MIB (HR-MIB) contains a set of management objects useful for hosts system. The term 'host' refers to any type of computer connected with other computers via a network. Although MIB-II is especially defined for the protocol stacks used in the host, the HR-MIB supplies additional information on the hardware, operating system, installed and running software.

The HR-MIB is structured by three hardware groups ("hrSystem", "hrStorage" and "hrDevice") and three software groups ("hrSWRun", "hrSWRunPerf" and "hrSWInstalled").

NOTE: The operating system is seen as having both hardware and

software components.

The Hardware Groups

--- -----

The system group, "hrSystem", provides additional system information besides the system group of MIB-II. The group provides general hardware variables and OS parameters.

The definitions consists of the following scalars.

- * "hrSystemUpTime" provides the amount of time since the host was last initialized. The difference with the "sysUpTime" is that the "hrSystemUpTime" provides the host up time and the "sysUpTime" the SNMP-entity uptime.
- * "hrSystemDate" provides the date and time of the system.
- * "hrSystemInitialLoadDevice" provides a pointer to the device in the "hrDevicetable" from which the system is booted.
- * "hrSystemInitialLoadParameters" provides the parameters with which the system is booted.
- * "hrSystemNumUsers" provides the number of user on the system.
- * "hrSystemProcesses" provides the number of processes currently loaded or running

- * "hrSystemMaxProcesses" provides the maximum number of processes which the context can support.

The storage group, "hrStorage", describes all logical components which can store data. The group definitions abstracts from the implemented storage techniques; it only provides a clustering of those techniques.

The group definitions consist of:

- * the definitions of storage types, "hrStorageTypes". For instance, the type 'floppy-disc' is defined.
- * "hrMemorySize" provides the amount of main memory contained by the system.
- * "hrStorageTable" provides information of all data storage components of the system. For example, if the system contains a floppy-disc the storage table has an entry describing that floppy-disc.

The device group, "hrDevice", contains information of all hardware components of the system. This group is useful for identifying and diagnosing faulty devices on a system.

The group definitions consist of the following objects:

- * the definitions of device types, "hrDeviceTypes". For instance, types are defined for processor and co-processor.
- * "hrDeviceTable" provides general information on the devices such as type, description and status. The table also references if a

more detailed table of the device is available.

- * "hrProcessorTable" provides additional information on the device type "hrDeviceProcessor".
- * "hrNetworkTable" provides the relationship between the network devices of the "hrDeviceTable" and the associating interface in

the "ifTable".

- * "hrPrintertable" provides additional information on printer devices.
- * "hrDiskStorageTable" provides additional type information on the disk storage devices.
- * "hrPartitionTable" provides information on partitions in a storage device used for long-term storage in a system.
- * "hrFSTable" provides information on the local or remotely mounted file system components.
- * the definitions of file system types ("hrFSTypes"). For instance, the Andrew file system and Windows NT file system.

The Software Groups

--- -----

The software groups define management information on the software used in the system. The information provided is grouped in (1) the currently running, (2) the performance and (3) the installed applications.

The software run group, "hrSWRun", contains information of each software part that is running or loaded into the physical or virtual memory in preparation of running.

The group definitions consist of the following objects:

- * "hrSWOSIndex" provides the index of the entry of the "hrSWRunTable" which represents the operating system.
- * "hrSWRunTable" provides detailed information on each piece of software which runs or will be loaded in preparation of running on the system.

The software run performance group, "hrSWRunPerf", consists of objects providing the performance of the running software.

The group consists of one table:

- * "hrSWRunTable" provides performance information on every piece of software running on the system. The entries of the table are link to the "hrSWRunTable" by "hrSWRunIndex".

The HR software installed group, "hrSWInstalled", is useful for identifying and inventoring software installed on the system. This does not include software loadable remotely from the network.

The group definitions consist of the following objects:

- * "hrSWInstalledlastChange" provides the "sysUpTime" of the last change of an entry in the "hrSWInstalledtable".

- * "hrSWInstalledlastChange" provides the "sysUpTime" of the last complete change of the "hrSWInstalledtable".
- * "hrSWinstalledTable" provides information on the installed software.

3.7 The Relational Database Management System MIB

The Relational Database Management System MIB (RDBMS-MIB) consists of a set of management objects for relational databases. In this MIB a separation is made between a database which is seen as a collection of interrelated data organized according a relational schema and an entity which provides access to the database, termed the database server. The database server may exist independently of the database, but this condition is not required. The server can be seen as an extra layer with a well defined API through which the data is accessed.

The structure of the RDBMS-MIB reflects the separation of (1) the database - where the actual data is stored - , (2) the server - access mechanism to the data and (3) the relationship between a database and an access mechanism.

The Database Group

--- -----

Meaningful information on the databases installed on the system.

The group consists of the following 4 tables:

- * "rdbmsDbTable" provides the general management information on the database. General parameters such as vendor name and contact for the database.
- * "rdbmsDbInfoTable" provides administrative information required when the database is actively opened on a host.
- * "rdbmsDbParamTable" describes the parameter for the configuration of the database (This table describes on a per entry basis for each parameter).
- * "rdbmsDbLimitedResourceTable" provides the limitations in resources per database (The provision of this information may only be done if the database is in operation (actively open)).

The Database Server Group

--- -----

This group provides more detailed information on the servers used for accessing of the databases:

The group consists of the following 4 tables:

- * "rdbmsSrvTable" provides general administrative information on running or installed database servers on a system.
- * "rdbmsSrvInfoTable" provides database server specific information which requires additional semantics then defined in

the "applTable" of the NSM-MIB. (An entry may only available in this table if the associated "applOperStatus" has the value "up(1)").

* "rdbmsSrvParamTable" provides the configuration parameters of the

database server.

- * "rdbmsSrvLimitedResourceTable" provides the limitations of resources used by the database server.

The Relations Group

--- -----

A relationship between database servers and databases is not uniquely defined. The design of the MIB allows cross connections between different databases and database servers.

The group consists of the following object:

- * "rdbmsRelTable" provides the relationship by the index of the "rdbmsDbIndex" and the "applTable". the supported relationships are 1:1, 1:many, many:1 and many:many. The table also provides the status of the relationship.

Traps

Two traps are defined to inform the manager of urgent matters concerning databases.

The following traps can be generated:

- * "rdbmsStateChange" provides that a certain relationship between a database server and a database has hanged in state and the state change made it less accessible for use.
- * "rdbmsOutOfSpace" provides that one of the databases is unable to allocate more space for its operation.

4 The overlapping objects of the MIBs

All of the surveyed MIBs in this document define managed objects for applications. That makes it possible to have certain overlaps duplications in their defintions. This section describes duplications or overlaps recognized between the surveyed MIBs.

NOTE: Whenever there is an overlap or duplication recognized between MIBs both are are mentioned in the title of the section. In these sections is then a detailed comparison made on scalar- or columnar-object level (the managed objects). The arrows between the objects express the relation and scaled in a certain level. If an arrow is draw with a '=' symbol object objects can be seen as equal. If an '~' symbol is used there is no real duplication, but only more detail is provided. It is also possible that multiple object can

express something similar than other objects.

Epires January 18, 1998

[Page 15]

4.1 The SYSAPPL-MIB and the HR-MIB.

The SYSAPPL-MIB provides information on installed applications and monitoring of processes which are running or have been running on a system. Basically the software groups of the HR-MIB provide the same kind of information. Both do not require instrumentation of applications in order to populate the MIBs.

Therefore, both MIBs could be interchanged without losing the ability to manage the processes/applications in a system. However, there is also a huge difference between both MIBs. The HR-MIB also provides information on hardware, not available in the SYSAPPL-MIB.

The Installed Software

The "hrSWInstalledTable" is somehow a subset of the "sysApplCfgeElmntTable". The reasoning behind this is that the "hrSWInstalledTable" has an entry for each installed piece of runnable software which is also provided by the "sysApplCfgeElmntTable", but the "sysApplCfgeElmntTable" has also information of non runnable elements of an applications. Below a detailed comparison at the level of the objects instances is shown in figure 1.

SYSAPPL-MIB	HR-MIB
sysApplInstalledTable	hrSWInstalledLastChange
sysApplInstalledEntry	hrSWInstalledLastUpdateTime
sysApplInstalledIndex<-----+	hrSWInstalledTable
sysApplInstalledManufacturer<-+	hrSWInstalledEntry
sysApplInstalledProductName<-+ +-- = --->	hrSWInstalledIndex
sysApplInstalledVersion<-----+	+-->hrSWInstalledName
sysApplInstalledSerialNumber<-+ -- ~	+-->hrSWInstalledID
sysApplInstalledDate<-----+ +- = -->	hrSWInstalledType
sysApplInstalledLocation +----- ~	--->hrSWInstalledDate
sysApplInstallElmtTable	
sysApplInstallElmtEntry	
sysApplInstallElmtIndex<---- ----+	
sysApplInstallElmtName<---- +	
sysApplInstallElmtType<---- -----+	
sysApplInstallElmtDate<-----+	
sysApplInstallElmtPath	
sysApplInstallElmtSize	
sysApplInstallElmtRole	
sysApplInstallElmtModifyDate	

Figure 1: Installed package overlap of the SYSAPPL-MIB and HR-MIB.

Although the HR-MIB has two objects to provide a time indication of the change in the "hrSWInstaledTable", the information can be seen as a subset provided of the SYSAPPL-MIB.

Epires January 18, 1998

[Page 16]

The "hrSWRunName" and "hrSWRunID" do not have similar objects in the run groups of the SYSAPPL-MIB, but this information can be retrieved from the SYSAPPL-MIB through the "sysAppElmtRunInstallID", which points to the "sysAppInstallElmtTable".

Although the overlap is here not direct recognisable, the software run group is contained by the more recent SYSAPPL-MIB. The exception is the "hrSWOSIndex" providing the index of the process which instantiates the OS (kernel).

[4.2](#) The SYSAPPL-MIB and the NSM-MIB.

The NSM-MIB provides a subset of applications of the SYSAPPL-MIB. That is, the subset consists of only those applications which have networked associations with other applications. Currently, the SYSAPPL-MIB is defined as the MIB with which the basic management operations can be done in respect of applications management. Strangely enough the NSM-MIB is the same, but the only for networked applications. The overlap between these MIBs consists of some general administrative data and some basic objects providing some running information.

The Administrative Application Objects

Since both MIBs are seen as the basis and generic platform on which other MIBs should be based, both define some administrative managed objects. In fact, there is no complete overlap, only partial overlap as it can be seen from figure 3.

SYSAPPL-MIB		NSM-MIB
sysApplInstalledTable		applTable
sysApplInstalledEntry		applEntry
sysApplInstalledIndex		applIndex
sysApplInstalledManufacturer		+>applName
sysApplInstalledProductName<+ ~		+>applDirectoryName
sysApplInstalledVersion<----		= >applVersion
sysApplInstalledSerialNumber		applUptime
sysApplInstalledDate		applOperStatus
sysApplInstalledLocation<----+		applLastChange
		applInboundAssociations
sysApplInstallElmtTable		applOutboundAssociations
sysApplInstallElmtEntry		applAccumulatedInboundAss..
sysApplInstallElmtIndex		applAccumulatedOutboundAss..
sysApplInstallElmtName<-----+		applLastInboundActivity
sysApplInstallElmtType		applLastOutboundActivity
sysApplInstallElmtDate		applRejectedInboundAssoc..
sysApplInstallElmtPath		applFailedOutboundAssiations
sysApplInstallElmtSize		applDescription
sysApplInstallElmtRole		applURL

Figure 3: Administrative overlap of the SYSAPPL-MIB and NSM-MIB.

The Overlap Of Runtime Objects

--- -----

Since both MIBs are seen as the basis and generic platform on which other MIBs should be based, both define some administrative managed objects. In fact, there is no complete overlap only partial overlap which can be seen in the figure 4.

SYSAPPL-MIB		NSM-MIB
sysApplRunTable		applTable
sysApplRunEntry		applEntry
sysApplRunIndex		applIndex
sysApplRunInstalledIndex		applName
sysApplRunLastStarted<-----+>		applDirectoryName
sysApplRunCurrentState<-----+>		applVersion
	+- ~ ->	applUptime
sysApplElmtRunTable	+--- ~ ->	applOperStatus
sysApplElmtRunEntry		applLastChange
sysApplElmtRunInvocId		applInboundAssociations
sysApplElmtRunIndex		applOutboundAssociations
sysApplElmtRunInstallID		applAccumulatedInbound..
sysApplElmtRunTimeStarted<---- +>		applAccumulatedOutbound..
sysApplElmtRunState<-----+>		applLastInboundActivity
sysApplElmtRunName		applLastOutboundActivity
sysApplElmtRunParameters		applRejectedInbound..
sysApplElmtRunCPU		applFailedOutbound..
sysApplElmtRunMemory		applDescription
sysApplElmtRunNumTCon		applURL
sysApplElmtRunNumFiles		

Figure 4: Runtime overlap of the SYSAPPL-MIB and NSM-MIB.

Overlap in runtime information exists only for the objects expressing the state of the application and time it is running.

4.3 The HR-MIB and the NSM-MIB.

The Host Resources MIB was the early start in which managed objects are defined to manage applications. In relation with the NSM they both provide management information of applications/software. The HR-MIB does this with respect to the running processes and the installed software in the system. The NSM-MIB shows only a part of this information with the extension of network related issues. The overlap consists only in some general administrative objects describing the application.

HR-MIB		NSM-MIB
hrSWOSIndex		applTable
		applEntry
hrSWRunTable		applIndex
hrSWRunEntry	+--- ~ ->	applName
hrSWRunIndex	+----- = ->	applDirectoryName
hrSWRunName<-----	+~ ->	applVersion
hrSWRunID<-----	- + applUptime	
hrSWRunPath<-----+	applOperStatus	
hrSWRunParameters	applLastChange	
hrSWRunType	applInboundAssociations	
	applOutboundAssociations	
hrSWInstalledLastChange	applAccumulatedInboundAss..	
	applAccumulatedOutboundAss..	
hrSWInstalledLastUpdateTime	applLastInboundActivity	
	applLastOutboundActivity	
hrSWInstalledTable	applRejectedInboundAssociations	
hrSWInstalledentry	applFailedOutboundAssociations	
hrSWInstalledIndex	applDescription	
hrSWInstalledName<-----+	applURL	
hrSWInstalledID<-----+		
hrSWInstalledType		
hrSWInstalledDate		

Figure 5: Runtime overlap of the HR-MIB and NSM-MIB.

[4.4](#) The WWW-MIB and the SYSAPPL-MIB

The WWW-MIB is defined at almost the same period as the SYSAPPL-MIB. Therefore, certain objects can be overlapping or redundant. This is due to changes made to the SYSAPPL-MIB. Both MIBs have more groups, but the only groups with some overlap are the "wwwSystem" and the "sysApplInstalled" group.

WWW-MIB		SYSAPPL-MIB
wwwEntityTable		sysApplInstalledTable
wwwEntityEntry		sysApplInstalledEntry
wwwEntityDescription<-----+		sysApplInstalledIndex
wwwEntityContact	+-- ~ --->	sysApplInstalledManufacturer
wwwEntityProtocol	+-- ~ --->	sysApplInstalledProductName
wwwEntityProtocolVersion	+-- ~ --->	sysApplInstalledVersion
wwwEntityName		sysApplInstalledSerialNumber
wwwEntityAddress		sysApplInstalledDate
wwwEntityType		sysApplInstalledLocation
wwwEntityUptime		

wwwEntityOperStatus
wwwEntityLastChange

Figure 6: Administrative overlap of the WWW-MIB and SYSAPPL-MIB.

Epires January 18, 1998

[Page 20]

As seen above (in figure 6) there is no big overlap in information. Only a sort of summarizing object within the WWW-MIB to describe an installed package.

4.5 The WWW-MIB and the NSM-MIB

These MIBs are related since they both provide network management information concerning network traffic with a certain underlying services.

The Administrative Objects

--- -----

In the tables which describe the general information only the version describing objects can be seen as overlapping. This is not completely true, because an application can have a total different version then its protocol. Also a newer version of an application can still implement the same protocol version.

WWW-MIB

NSM-MIB

wwwEntityTable		applTable
wwwEntityEntry		applEntry
wwwEntityDescription<-----+		applIndex
wwwEntityContact +---	~	-->applName
wwwEntityProtocol<-----+		applDirectoryName
wwwEntityName +---	~	-->applVersion
wwwEntityType +-----	=	-->applUptime
wwwEntityUpTime<-----+	+---	= -->applOperStatus
wwwEntityOperStatus<-----+	+ =	-->applLastChange
wwwEntityLastChange<-----+		applInboundAssociations
		applOutboundAssociations
		applAccumulatedInboundAs..
		applAccumulatedOutboundAs..
		applLastInboundActivity
		applLastOutboundActivity
		applRejectedInboundAss..
		applFailedOutboundAss..
		applDescription
		applURL

Figure 7: Administrative overlap of the WWW-MIB and NSM-MIB.

The "wwwEntityTable" defines more detail on the networked application. Overlap does not really exists. In the "wwwEntityTable" is an entry uniquely defined through its Fully Qualified Domain Name which is not reuired for the applTable.

The Network Traffic Objects

--- -----

The overlap at the network traffic level is far obvious. Within the WWW-MIB a summary table is providing a quick view of the network traffic. This is almost similar to the network information which is provided by the application table of the NSM-MIB.

WWW-MIB	NSM-MIB
wwwSummaryTable	applTable
wwwSummaryEntry	applEntry
wwwSummaryInRequests<----+	applIndex
wwwSummaryOutRequest<-----+	applName
wwwSummaryInResponses<---+	applDirectoryName
wwwSummaryOutResponses<----+	applVersion
wwwSummaryInBytes	applUptime
wwwSummaryOutBytes	applOperStatus
	applLastChange
	applInboundAssociations
	applOutboundAssociations
	+-- - (sum) ->applAccumulatedInboundAs..
	+- (sum) ->applAccumulatedOutboundAs..
	applLastInboundActivity
	applLastOutboundActivity
	applRejectedInboundAss..
	applFailedOutboundAss..
	applDescription
	applURL

Figure 8: Network traffic overlap of the WWW-MIB and NSM-MIB.

All indexes could be coupled between both MIBs. The difference is that the WWW-MIB is much more detailed information concerning the used document transfer protocol. This is due to the fact that the WWW-MIB describes network management information for the instantiated protocol providing a certain service. On the contrary, the NS-MIB defines the network management information of the provided service and abstract from protocol issues.

The relation between the WWW MIB and the NSM MIB isnot very explicit and built on the fact that a certain service is provided both by a protocol and an underlying service (a stack of protocol layers). Although entities and services are distributed, these MIBs define only information for one specific point (address/ host). To fill the gap for the remote relations/ connections the NSM-MIB has an association table in which the information of the current connections are kept.

In a certain way the network service information at a service access point can be seen as a summary of the underlying protocol information.

Epires January 18, 1998

[Page 22]

possibility that the referred object/entry doesn't exist.
In this section all the index links, pointers within one of the surveyed MIBs or between two of them are described. Mostly these pointers or indexes are quite straightforward, but it provides also

As figure 10 (above) shows, all indexes (except one) are made with pointers. The reason for this is that in the RunTables and RunPastTables every process should be shown and if possible grouped by application. The direct index is only possible between the

"sysApplInstallPackageTable" and the "sysApplInstallElementTable" because it is not required that every process is registered within an "sysApplInstallPackageTable".

A problem with this MIB (especially under an implementing point of view) is the grouping of the elements into a composite application. This could be done by reading the element organisation from a configuration file. The configuration file should also provide the information on what element is from which application during runtime. The element could be a part of an already started application (an entry already exists in the "sysApplRunTable") or a complete new application (a new entry has to be inserted in the "sysApplRunTable").

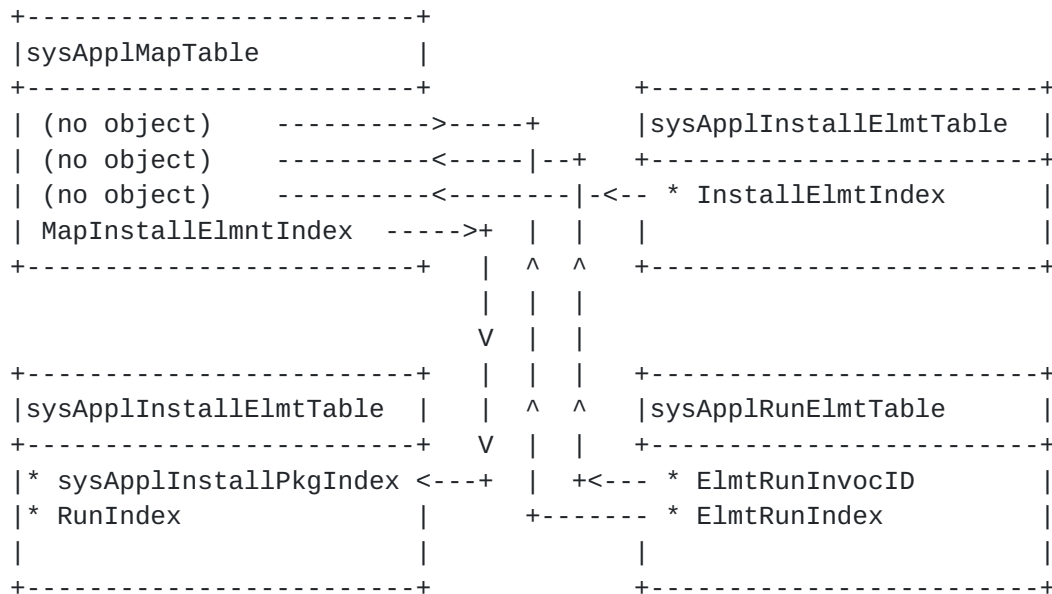


Figure 11: Links sysApplMapTable.

Although the linkage for this table is very complex, it provides the manager an easy mapping between running elements and the installed elements. The burden for this table is, therefore, put in the SNMP-agent and not in the SNMP-manager application.

5.2 The Application Management MIB

The Application Management MIB provides the management information per running element for which it requires instrumentation inside the running element.

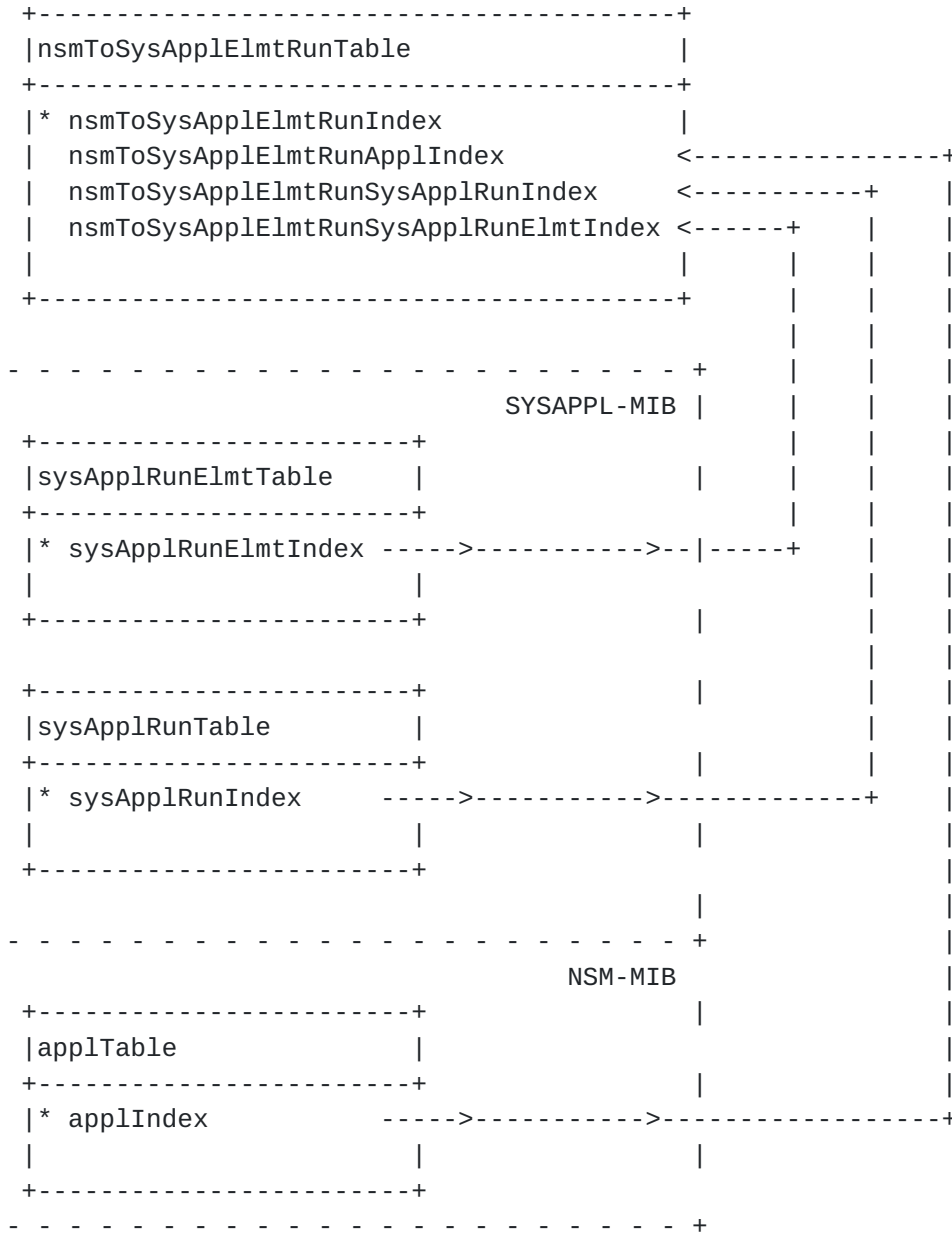


Figure 12: The relationships with the SYSAPPL-MIB and NSM-MIB.

The indexing scheme within this MIB relies on the existence of the "sysappElmtRunIndex" of the SYSAPPL-MIB. Therefore, it is required to at least implement this object. Seen from the goals for as well the SYSAPPL-MIB as the Application Management MIB you should

use/implement the framework which exists of the indexing scheme.
An overview of the relationships of the Application Management MIB
is given in Figure 13.

As seen in the MIB definitions of the Application MIB most of the

tables are combined for counter of both Counter32 and Counter64 application types. It is the editor's view that in order to create less overhead and a more clearly structure the tables should be split in a table defining the Counter32 counters and a table for the Counter64 counters. The Implementor should teh be able to choose via MODULE-COMPLIANCE statements which complaince he selects. of course, this choice depends highly on the used SNMP version and the architecture the MIB is implemented on.

```

+-----+
|sysAppElmtRunTable | |
+-----+
|* sysAppElmtRunIndex |<-|-----+<---< -----(no object) |
| | | ^ +-->* applOpenFileId |
+-----+ | +-----> applOpenFileName |
          SYSAPPL-MIB | | | |
- - - - - + | | | +-----+
          | | |
+-----+ ^ | | |
|applOpenFileXRefTable | | | +-----+
+-----+ | | | |applOpenConnectionTable |
| (no object) ---->---->+ | | +-----+
| (no object) ---->---->+<---|---- (no object) |
| (no object) ---->---->+ |* applOpenIndex |
+-----+ | | +-----+
          |
+-----+ ^ +-----+
|applExConnectionControlTable | | |applExConnectionTable |
+-----+ | | +-----+
| (no object) ---->---->+<---<--- (no object) |
| | | |* applExConnectionId |
+-----+ | | +-----+
          |
+-----+ ^ +-----+
|applTransactionControlTable | | |applTransactionStreamTable |
+-----+ | | +-----+
| (no object) ---->---->+<---<--- (no object) |
|* applTransactionStreamId <-----<----- (no object) |
| | | |
+-----+ | | +-----+
          |
+-----+ ^ +-----+
|applAlmtRunStatusTable | | |applElmtRunControlTable |
+-----+ | | +-----+
| (no object) ---->---->+<---<--- (no object) |

```



Figure 13: The Application Management MIB indexing scheme.

Figure 14: Links within the core of the WWW-MIB.

Epires January 18, 1998

[Page 28]

In the latest version of the WWW-MIB is the defined relationship (with "applIndex") taken away and a internal indexing scheme is used. This solution is chosen because the NSM-MIB is not precise enough in what is a networked service and how can they be distinguished. This solution benefits also that no instrumentation is required from the NSM-MIB to implement the WWW-MIB.

5.4 The NSM-MIB

This MIB does not have pointers to other MIBs, due to its generic nature for management of networked applications. Compared with the SYSAPPL-MIB the NSM-MIB is much simpler, consisting only of two tables. One of the tables provides more detailed information on the application and the other tables provides the associations with remote (other) applications. Figure 15 shows the internal linking structure of the NSM-MIB.

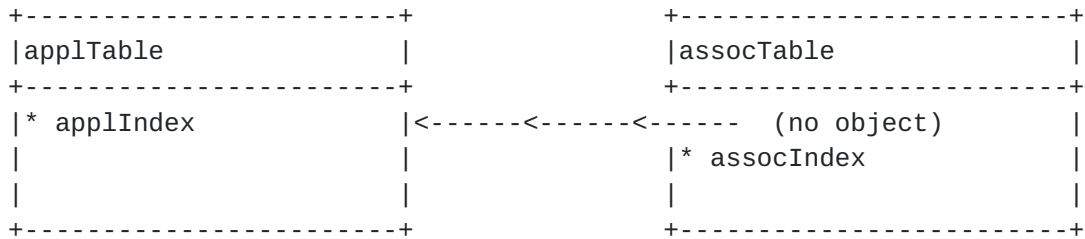


Figure 15: Links within the NSM MIB.

5.5 The X.500 DSA-MIB.

This MIB consists of three tables providing application specific information for a networked and distributed application. Therefore, the DSA MIB is connected to the NSM-MIB through the "applIndex".

As figure 17 shows (next page), the core of the WWW MIB is defined without using external indexes. Previously, the indexing of the WWW-MIB was done by means of the "applIndex" defined in the NSM-MIB. Due to no clear definitions used in the NSM-MIB and indexing problems. The WWW-MIB will not be defined with the indexing scheme of the NSM-MIB. The problem is that unclaeer definitions makes it possible to have lots of non-unique relations between both MIBs. During the implementation of the WWW-MIB the problem is how to uniquely define the index. Questions such as "May I choose my own index number?" or "Is it required to implement the NSM-MIB?" can be raised. Unfortenately, the SNMP-framework doesn't give an

concrete answer based on arguments consistent with the SNMP framework documents. A solution is only to implement the "applIndex" of the "applTable", due to the maximum access clause of "not-accessible" the "applIndex"-instance can never be retrieved.

Epires January 18, 1998

[Page 29]

Epires January 18, 1998

[Page 30]

The hardware groups

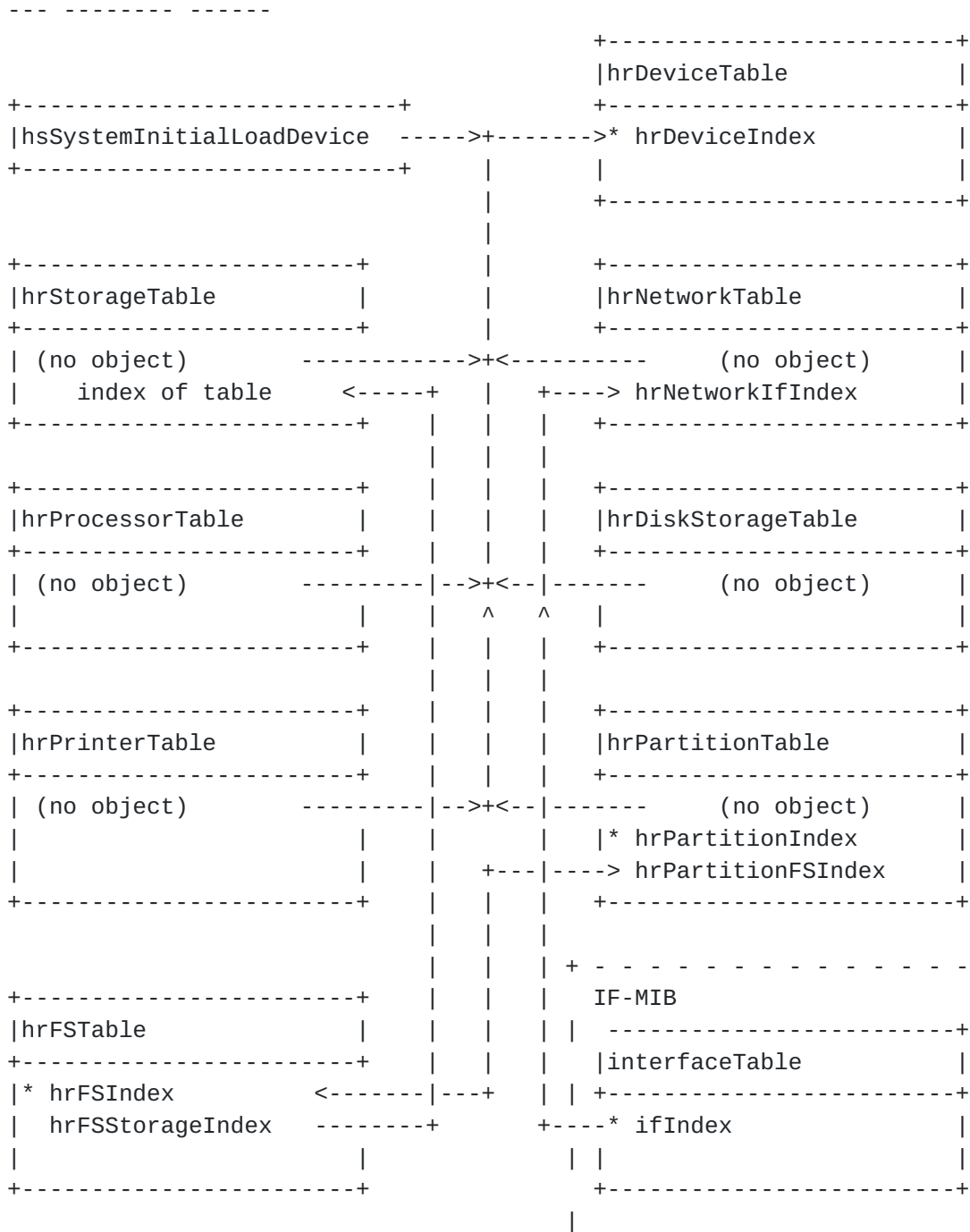


Figure 17: Links of the hardware groups within the HR MIB.

The software groups

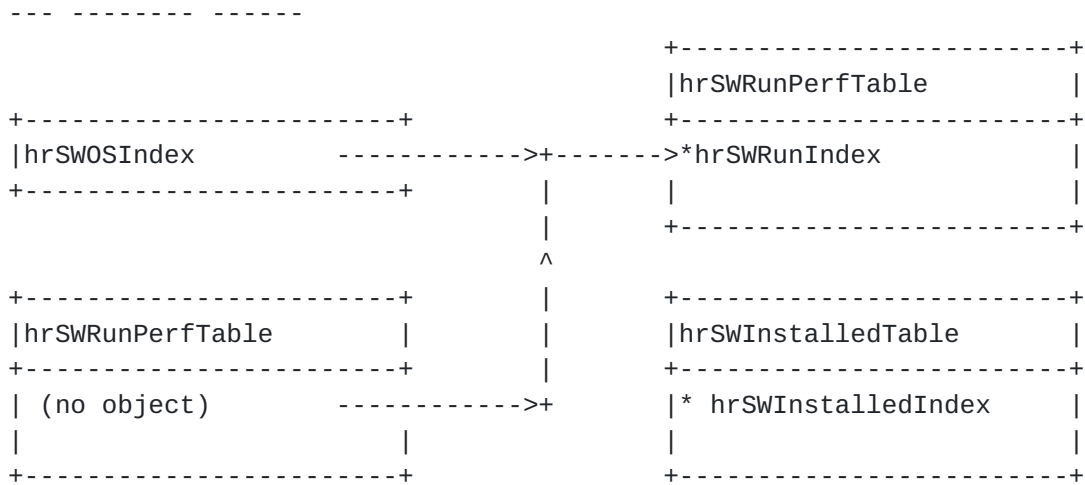


Figure 18: Links of the software within the HR MIB.

The above figure (18) shows that there are a few links between the tables and that no connection is made to other MIBs. Therefore, the software groups can be implemented without big difficulties, not requiring instrumentation of other MIBs, whether or not available.

5.7 The RDBMS-MIB.

The RDBMS-MIB is an application specific MIB which is used for DataBase Management.

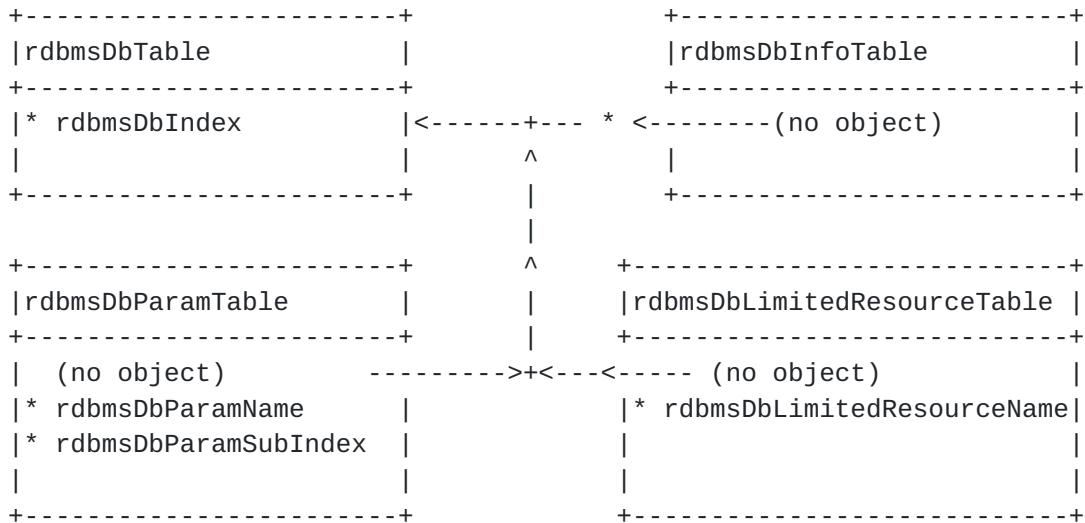


Figure 19: Links within the database group of the RDBMS MIB.

The database group complete depends of the rdbmsDbTable in which general management data is defined for a database.

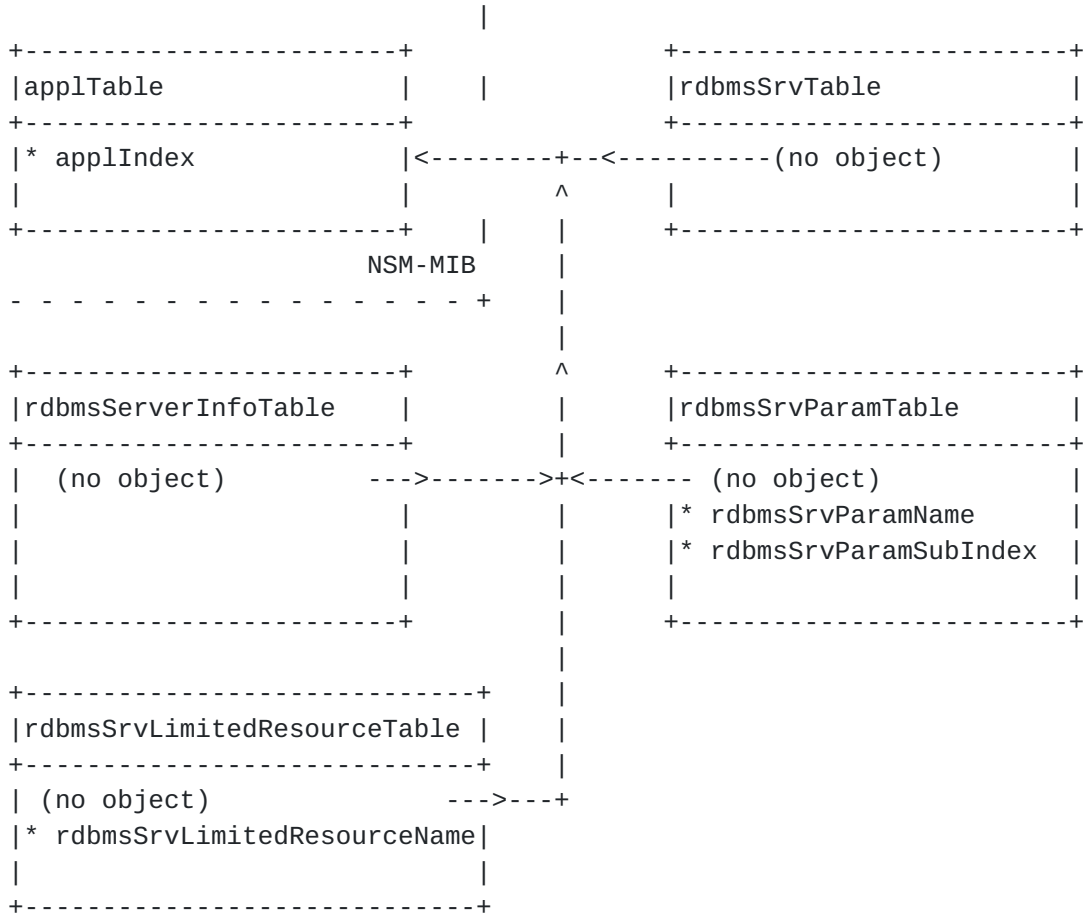
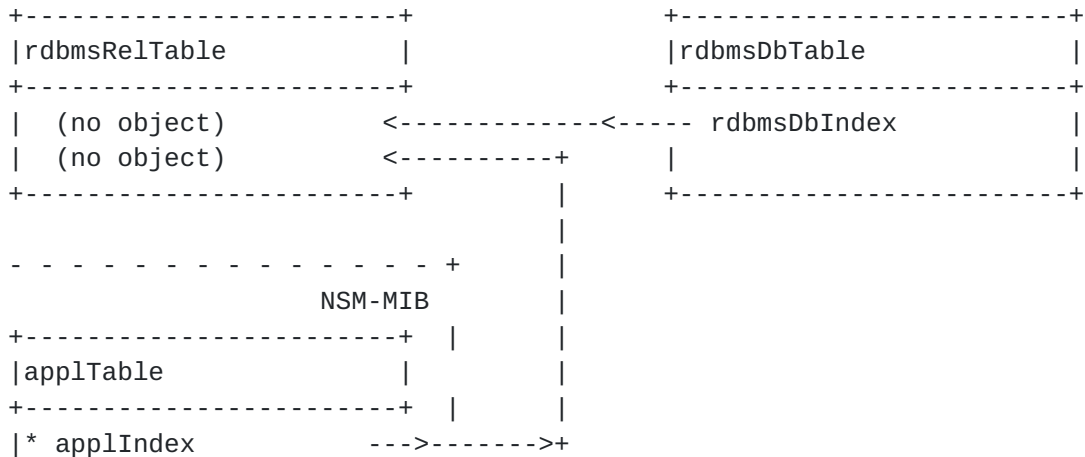


Figure 20: Links within the database server group of the RDBMS MIB.

This figure (20) shows that the database server is seen in terms of a network service, since its indexing scheme depends on the applIndex. The applIndex is defined withi the NSM-MIB.



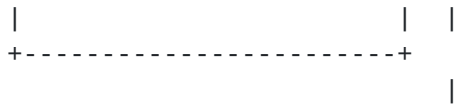


Figure 21: Links connecting the database and database server group.

In order to have full control of which database is used by which database server a relations table is added. An advantage of this method is that a user can decide which database and/or which database server he uses and at the same time the manager has enough information for proper management.

6 Conclusions

This section presents the major conclusions found during this study. The study analysed the currently existing MIBs defined in IETF WGs which can be used for application management. The investigated MIBs were:

- * the SYSAPPL-MIB,
- * the APPLICATION-MIB,
- * the WWW MIB,
- * the NSM-MIB,
- * the DSA-MIB,
- * the HR-MIB, and
- * the RDBMS-MIB.

The objective of the analyses was to identify:

- a) the contents of the MIBs,
- b) the duplications and the overlaps of the MIBs, and
- c) the indexing scheme of the MIBs.

The SYSAPPL-MIB framework

--- -----

Within the APPLMIB WG a complete framework is defined for proper management of applications using SNMP. The output of this WG will be 3 MIBs which are the SYSAPPL-MIB, Application Management MIB, and the WWW-MIB. This also requires that a complete indexing scheme must be setup and except for the WWW-MIB the MIBs have a complex network of indexing. Especially, the indexing scheme with in the SYSAPPL-MIB. A good reason for this is that an application only once installed may only be once in the 'installed'-tables, but they certainly can be run multiple times and spin-off multiple running elements. Although, the WWW-MIB is defined within the APPLMIB WG and therefore must fullfill the requirements of the framework, a service view is selected for this MIB. The service view allows the WWW-MIB to be independent of the others. Managers of application software, and in particular WWW-servers, do not have to implement the full framework and carry the burden which comes with it.

The NSM-MIB

--- -----

Although a definition of a networked application in the NSM-MIB is provided, but it defines not properly a service on which the definition depends. Questions like 'Is a networked application only listening to one single transport address or listening to multiple?' or 'What mapping into the "applTable" has to be used in case of a networked application using the "multi-home-host" approach?' can be raised.

This last question presents an arbitrary choice for MIB implementors to define the "applTable" entries for such a networked application. For example, a "multi-homed-host" with a networked application can be defined in one entry, but also in multiple entries. It is not even required that in case of multiple host and entries the relation is 1:1.

SYSAPPL MIB overlapping the HR MIB software groups

The managed objects in the software groups of the HR MIB are almost completely overlap by the newer SYSAPPL MIB. Although the HR MIB is more process oriented, this information can be extracted from the SYSAPPL-MIB which even combines the processes into a coherent application.

Linking with other MIBs

There are two general approaches in linking objects across different MIBs. The first approach is to define an object which contains a value, object identifier (OID) of the referred object. This approach is simple and even possible to use whenever the referred object is not defined in the SNMP agent, where the zero pointer can be used. The second approach is to use the value of the referred object as an index (is in this draft called a "direct index"). This approach is more consistent since links are uniquely defined. On the contrary, it raises problems when the referred object is not implemented in the SNMP agent. In such a case, the referred index cannot be discovered and the agent implementor has to make his own decision regarding the index values.

Mapping between the SYSAPPL-MIB and the NSM-MIB

Nowadays two 'major' MIBs, SYSAPPL-MIB and NSM-MIB, are available for

application management. Both claim to be generic and could, therefore, be used as platform to hook on other application specific MIBs.

The definition of an application within the SYSAPPL-MIB is:

"An application is one or more units of executable code and other resources, installed on a single host system that a manager may think of as a single object for management purposes."

The definition of an application (or networked application) within the NSM-MIB is not really defined, but implicit within the draft as:

"A networked application is a unit (executable code) within a single host which operates in conjunction with other units over a network."

Unfortunately, the mapping between both types of applications is not well and uniquely defined. This is what we directly see taking WWW-servers as an example. Normally, the mapping between an application in the SYSAPPL-MIB and an application in the NSM-MIB should be no problem. However, when using virtual hosts on a single machine the application components are going to behave very differently. WWW-servers, like Apache, is one application (as seen from a SYSAPPL-MIB point of view), but it are also multiple networked applications (as seen from a NSM-MIB point of view) by using virtual hosts.

7 References

- [] McCloghrie, K., and M. Rose, Editors, "Management Information Base for Network Management of TCP/IP-based internets: MIB-II", STD 17, [RFC 1213](#), Hughes LAN Systems, Performance Systems International, March 1991.
- [] Grillo, P., and S. Waldbusser, "Host Resources MIB", [RFC 1514](#), Network Innovations, Intel Corporation, Carnegie Mellon University, September 1993.
- [] Kille, S., and N. Freed, "Network Services Monitoring MIB", [RFC 1565](#), ISODE Consortium, Innosoft, January 1994
- [] Saperia, J., C. Krupczak, R. Sturm, and J. Weinstock, "Definition of Managed Objects for Applications", [draft-ietf-applmib-sysapplmib-08.txt](#), BGS Systems, Empire Technologies, Enterprise Management Professional Services, Bellcore, April 1997.
- [] Kalbfleisch, C., C. Krupczak, R. Presuhn, J. Saperia, "Application Management MIB", [draft-ietf-applmib-mib-03.txt](#), September 1997.
- [] Kalbfleisch, C., Hazewinkel, H., Schoenwaelder, J., "Definitions of Managed Objects for WWW Servers", [draft-ietf-applmib-wwwmib-03.txt](#), June 1997.
- [] Krupczak, C. and S. Waldbusser, "Applicability of Host Resources MIB to Application Management", Empire Technologies, Inc., International Network Services, October 1995.
- [] Kille, S., and N. Freed, "Mail Monitoring MIB", [RFC 1566](#), ISODE Consortium, Innosoft, January 1994.
- [] Mansfield, G., and S. Kille, "X.500 Directory Monitoring MIB", [RFC 1567](#), AIC Systems Laboratory, ISODE Consortium, January 1994.
- [] Mansfield, G., and S. Kille, "X.500 Directory Monitoring MIB", [RFC 1567](#), AIC Systems Laboratory, ISODE Consortium, January 1994.
- [] brower, D., Purvy, B., Daniel, A., Sinykin, M., Smith, J., "Relational Database management System (RDBMS) Management Information Base (MIB) using SMIV2", [RFC1697](#), INGRES DBMS Development, August 1994.
- [] Kalbfleisch, C., "Applicability of Standards Track MIBs to Management of World Wide Web Servers", [RFC-2039](#), OnRamp Technologies, November 1996.

8 Authors address

Harrie Hazewinkel
postal: CEO Programme
 Joint Research Centre of the E.C
 TP 950
 21020 Ispra Italy
email: harrie.hazewinkel@jrc.it
www: <http://porto.jrc.it/~harrie>

9 Acknowledgements

This document benefited from the comments of the following persons:

Eric van Hengstum	<hengstum@cs.utwente.nl>
Carl Kalbfleish	<cwk@verio.net>
Cheryl Krupczak	<cheryl@empiretech.com>
Rui Meneses	<rui.meneses@jrc.it>
Aiko Pras	<pras@cs.uwtente.nl>
randy Presuhn	<rpresuhn@peer.com>
Jon Saperia	<saperia@networks.bgs.com>
Jeurgen Schoenwaelder	<schoenw@cs.utwente.nl>

10. Table of Contents

Abstract2

[1.](#) Introduction3

[1.1](#) How to read this internet draft4

[2.](#) The Working Groups of Application Related MIBs4

[3.](#) The application related MIBs5

[3.1](#) The System Application MIB5

[3.2](#) The Application Management MIB7

[3.3](#) The World Wide Web MIB9

[3.4](#) The Network Services Monitoring MIB10

[3.5](#) The X.500 Directory Monitoring MIB11

[3.6](#) The Host Resources MIB11

[3.7](#) The Relational Database Management System MIB14

[4.](#) The overlapping objects of the MIBs15

[4.1](#) The SYSAPPL-MIB and the HR-MIB16

[4.2](#) The SYSAPPL-MIB and the NSM-MIB18

[4.3](#) The HR-MIB and the NSM-MIB19

[4.4](#) The WWW-MIB and the SYSAPPL-MIB20

[4.5](#) The WWW-MIB and the NSM-MIB21

[4.6](#) The WWW-MIB and the HR-MIB23

[5.](#) Relationships/Indexing between managed objects23

[5.1](#) The SYSAPPL-MIB24

[5.2](#) The Application Management MIB26

[5.3](#) The WWW-MIB28

[5.4](#) The NSM-MIB29

[5.5](#) The X.500 DSA-MIB29

[5.6](#) The HR-MIB30

[5.7](#) The RDBMS-MIB32

[6.](#) Conclusions34

[7.](#) References37

[8.](#) Author's address38

[9.](#) Acknowledgements38

[10.](#) Table of Contents39

