

DHCP

czyli automatyczna konfiguracja interfejsu sieciowego

Skąd komputer bierze adres IP, maskę, oraz inne elementy konfiguracji interfejsu sieciowego?

Historia: RARP (tylko adres IP), BOOTP, DHCP (rozszerzenie BOOTP, obecnie używane...)

RARP – podobne do ARP, działa w drugą stronę, pyta/odpowiada, broadcast sprzętowy, losowe opóźnienie przy wysyłaniu odpowiedzi przez serwer rarp (istotne w etch...)

BOOTP – jedna wymiana komunikatów z serwerem BOOTP

taki sam komunikat w pytaniu i odpowiedzi, w odpowiedzi wypełniają się nieznane pola, możliwość użycia przełącznika, nazwa pliku startowego (obraz dysku), komputer bezdyskowy

DHCP – rozszerzenie BOOTP, trochę bardziej skomplikowane...

trzy sposoby przydzielania adresu:

1. adres etch → adres IP, przydzielany „ręcznie”
2. adres IP przydzielany dynamicznie z puli, gdy pojawi się pytanie
3. adres IP przydzielany dynamicznie/ na określony czas („leasing”) po upływie okresu wynajęcia musi być odnawiany!!
tylko ta metoda pozwala na odzyskanie adresu IP !!!

BOOTP i DHCP działają nad UDP (porty 67 i 68)

jak to możliwe gdy nie ma adresu IP? odpowiedź: broadcast, adres docelowy IP „same 1”

Linux:

klient dhcp : program **dhcpcd**; serwer dhcp: program dhcpcd lub **dhcpcd3**
po stronie klienta: „dhcpcd eth0”

DHCP

Jakie informacje dostarcza dhcp?

Adr ip, maskę, default gw, adr ser DNS, wart MTU, ...

Komunikaty BOOTP i DHCP...

w DHCP jest kilka typów komunikatów (w opcjach, z uwagi na kompat z BOOTP)
kli może otrzymać kilka odpowiedzi, od kilku ser, wybiera z nich jedną...

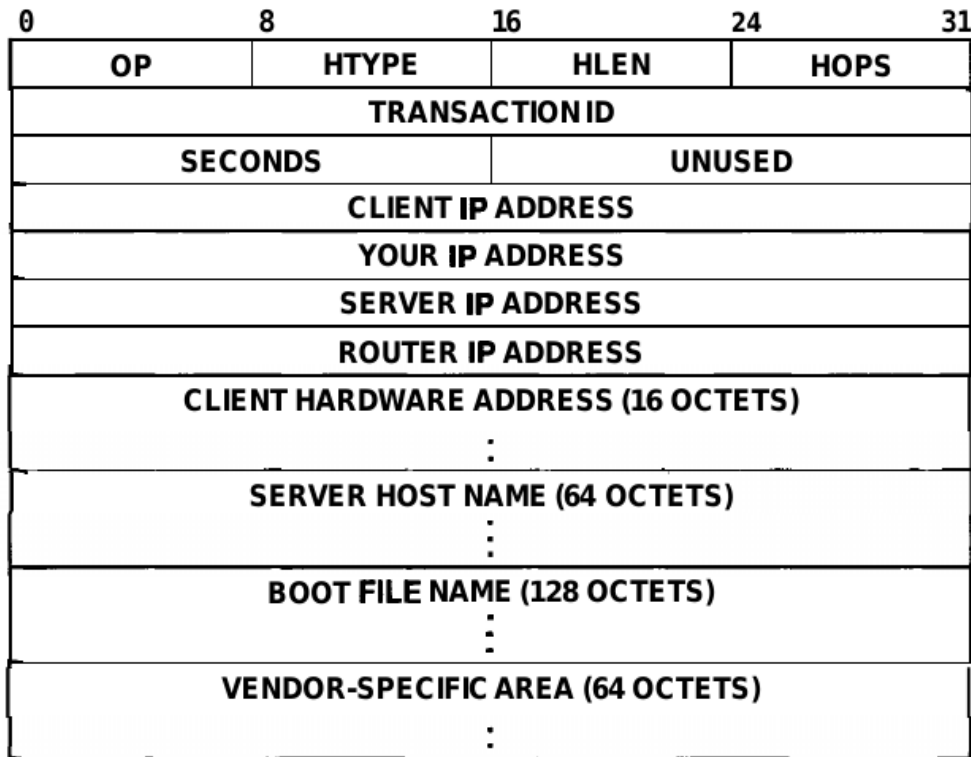


Figure 23.1 The format of a BOOTP message. To keep implementations small enough to fit in ROM, all fields have fixed length.

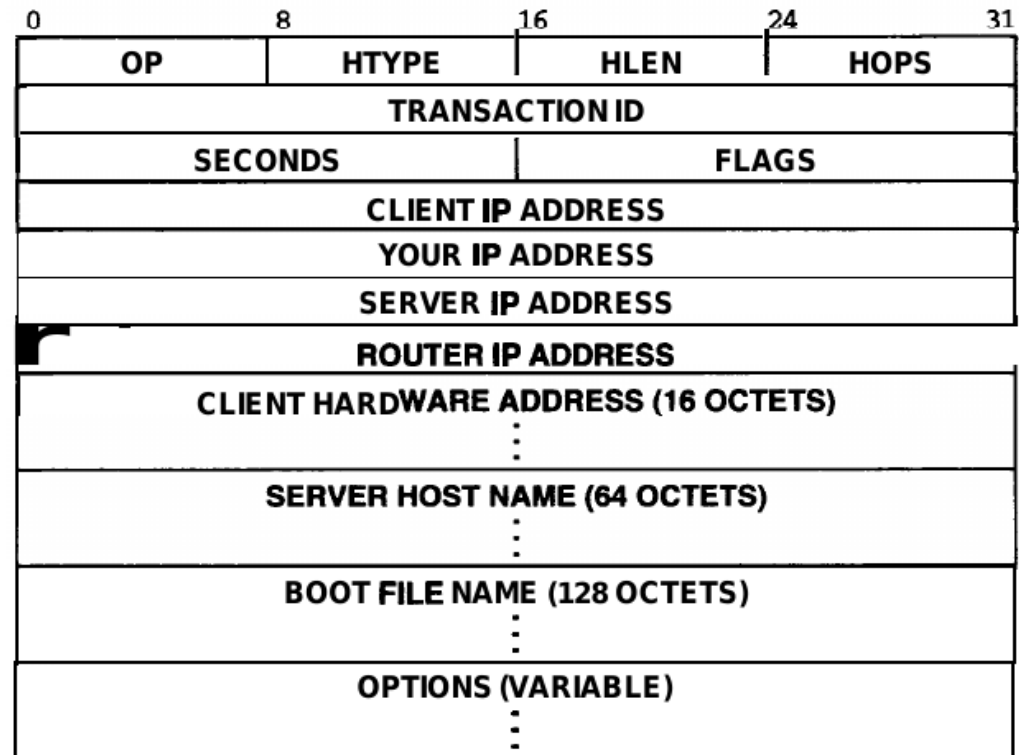


Figure 23.5 The format of a DHCP message, which is an extension of a BOOTP message. The options field is variable length; a client must be prepared to accept at least 312 octets of options.

DHCP

Stany kli dhcp...

normalny stan działania to „bound”
„DHCP*” to nazwy komunikatów

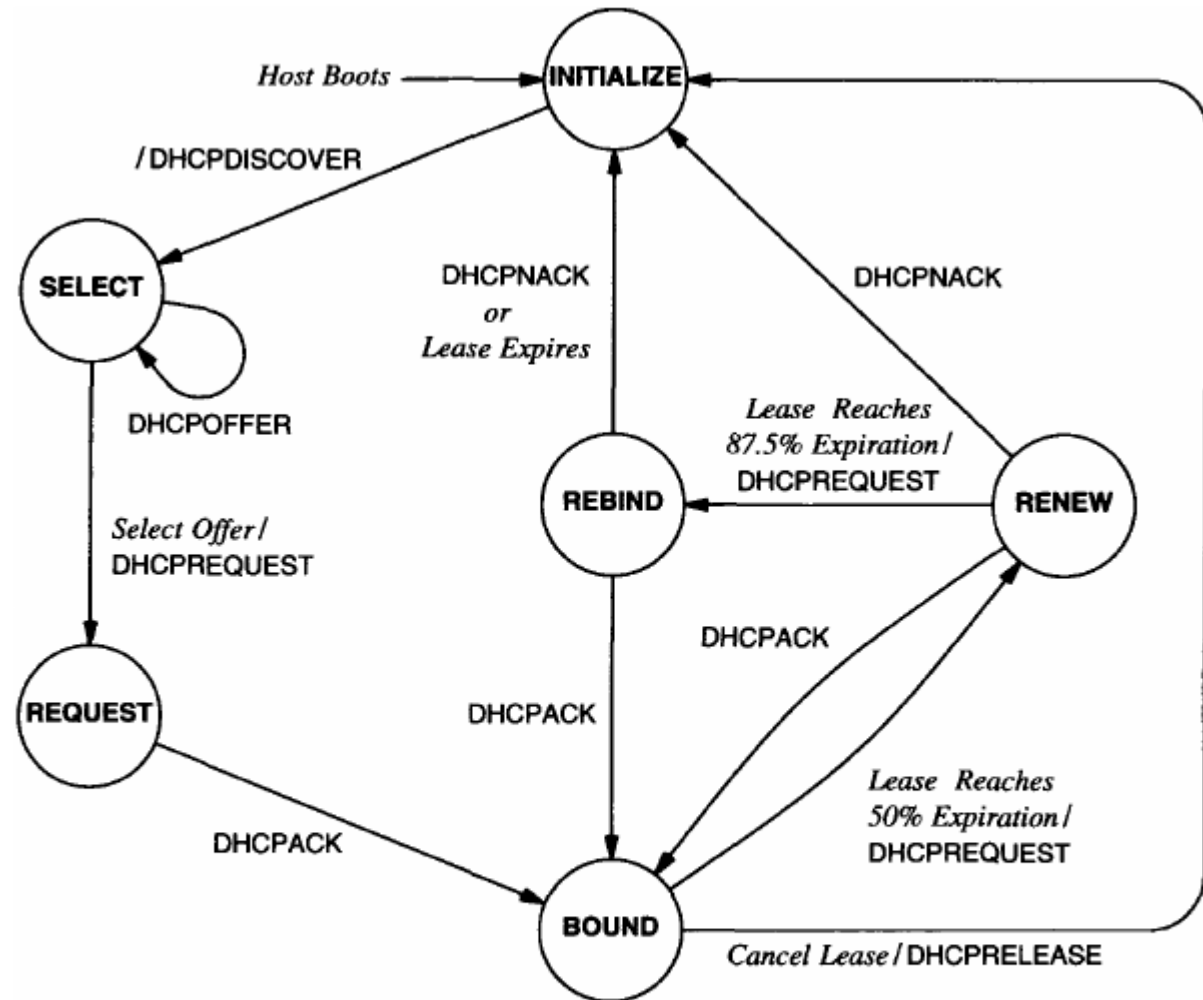


Figure 23.4 The six main states of a DHCP client and transitions among them. Each label on a transition lists the incoming message or event that causes the transmission, followed by a slash and the message the client sends.

IPv6

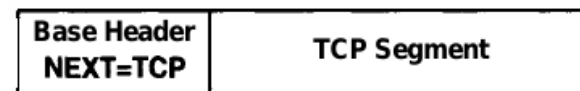
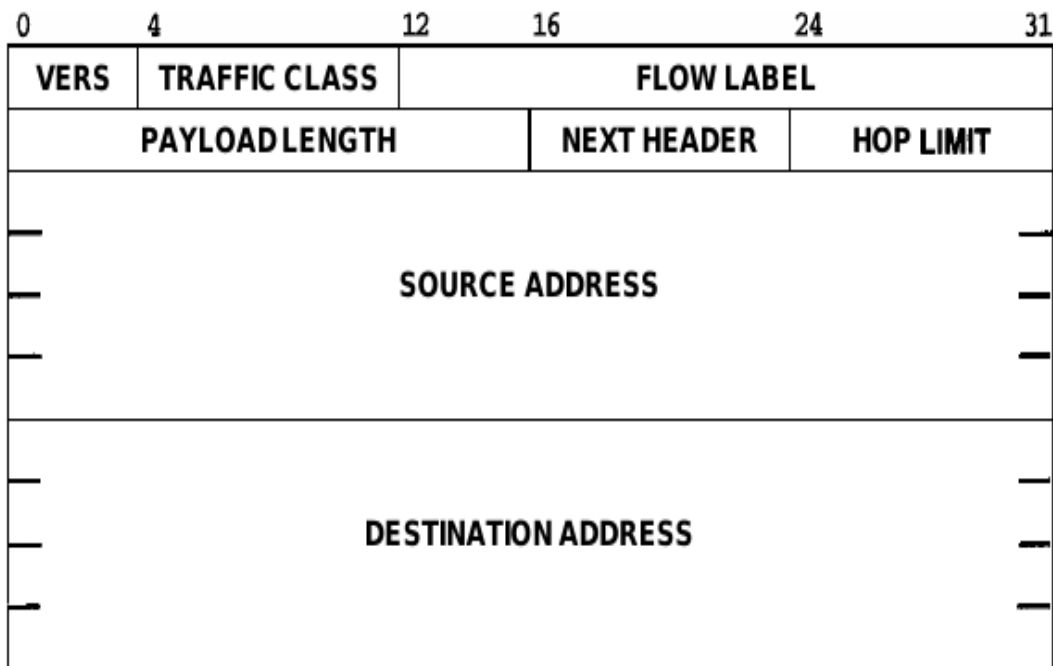
czyli ulepszenie war. 3 (IPv4)

IPv6 to modyfikacja IPv4 ,czyli wymiana 3 warstwy prot...

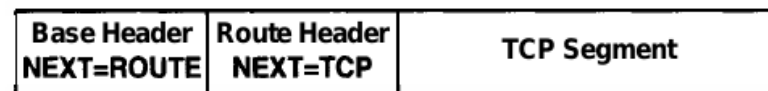
główna cecha: dłuższe adr: ipv4 32bit, ipv6 128bit (na pewno ich wystarczy...)

Uproszczony nagłówek ipv6 w porównaniu do ipv4, 40bajtów,
 hop limit = ttl, traffic class i flow label = na użytek QoS, payload len = długość danych,
 next header = typ następnego nagłówka (w danych)...

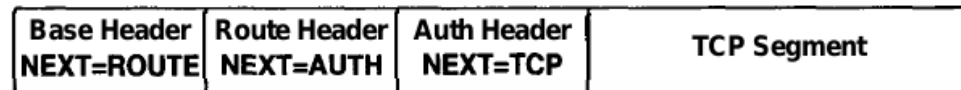
może to być nagłówek dodatkowy ipv6 lub nagłówek wyższej warstwy (UDP, TCP)



(a)



(b)



(c)

Figure 33.3 Three datagrams with (a) only a base header, (b) a base header and one extension, and (c) a base header plus two extensions. The NEXT HEADER field in each header specifies the type of the following header.

IPv6

Adresy ipv6 i ich skróty:

2001:0db8:0000:0000:0000:ff00:0042:8329

2001:db8:0:0:0:ff00:42:8329

2001:db8::ff00:42:8329

„::” oznacza ciąg zer, może wystąpić tylko raz! „::1” to localhost

Adresy budowane na podstawie innych:

Adresy LL = Local-Link, adr ipv6 na podstawie adr eth, ważne tylko w 1 sieci fizycznej,

FE80 :: X1 X2 : X3 FF : FE X4 : X5 X6

X1:X2:X3:X4:X5:X6 to adr eth ze zmodyfikowanym 1 bajtem (2 najmłodszy bit)

UWAGA: adr LL wymagają info o interfejsie bo mogą się powtarzać:

„adr_ipv6%eth0” (warto zrobic eksperyemnt !!!)

są też inne możliwości: np. ::FFFF:y1.y2.y3.y4 – adr ipv6 na bazie adr ipv4

Jak się włącza ipv6 na linuxie?

jeśli ipv6 działa to plik /proc/net/if_inet6 istnieje

„modprobe ipv6” włącza działanie ipv6 w linuxie od kernela 2.6 !!!

w linuxie, przydziela się adres LL do interf...

Współistnienie ipv4 i ipv6:

1. dwa stosy prot, z ipv4 i ipv6
2. tunelowanie pkg ipv6 w pkg ipv4

Typy adr dst:

Ipv4: unicast, broadcast, multicast

Ipv6: unicast, multicast, anycast (wysyłamy do jednego hosta ze zbioru)

NIE ma pojęcia broadcasting !!

IPv6

Typy adresów ipv6 na podstawie **RFC 4291** (prefiks decyduje):

Address type	Binary prefix	IPv6 notation	Section
Unspecified	00...0 (128 bits)	::/128	2.5.2
Loopback	00...1 (128 bits)	::1/128	2.5.3
Multicast	11111111	FF00::/8	2.7
Link-Local unicast	1111111010	FE80::/10	2.5.6
Global Unicast	(everything else)		

Jeden if (np. eth0) może mieć: 1 adr ipv4, >=1 adr ipv6...

```
# ip addr add ::ffff:192.168.1.3/120 dev eth0
# ^ ipv6 oparty na ipv4!! maska nie obejmuje ostat bajtu!
# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr 08:9E:01:1C:9C:70
          inet addr:192.168.1.3  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::a9e:1ff:fe1c:9c70/64 Scope:Link
          inet6 addr: ::ffff:192.168.1.3/120 Scope:Global
          # ^ 1 adr ipv6 ma scope link, drugi ma scope gloobal
```

Tabl routinowa osobna dla ipv4 i ipv6 („route -n -A inet/inet6” albo „ip route”)

```
# route -n -A inet6
Kernel IPv6 routing table
Destination „Next Hop” Flags Metric Ref Use Iface
::1/128 :: Un 0 2 0 lo
fe80::9ead:97ff:fe84:af1d/128 :: Un 0 2 0 wlan0
fe80::/64 :: U 256 1 0 wlan0
ff00::/8 :: U 256 2 0 wlan0
::/0 :: !n -1 1 0 lo
# ^ to są reguły dla sieci bliskich...
```

SNMP

czyli zarządzanie sieciami komp

Manager i agent SNMP (działa na urządzeniu sieciowym)...

Obiekty/zmienne MIB opisane w języku SMI (syntaktyka z ASN.1)

Agent utrzymuje zmienne opisujące urządzenie

Manager może odczytywać te zmienne i modyfikować

(na tym polega proto manager \leftrightarrow agent, także oparty na ASN.1/encoder/dekoder)

Jest możliwość czekania na zdarzenie od agenta (trap)

Przykładowe oprogramowanie: program „scotty”...

Std drzewo obiektów/zmiennych ISO/ITU...

ścieżka korzeń liść – zm. prosta,

ścieżka korzeń „nie liść” - zm. złożona (obiekt ?), np. tablica

ścieżka może być ciągiem liczb lub etykiet...

Poddrzewo tego drzewa zawiera „MIB-2”,

którego impl jest obowiązkowa dla agentów snmp...

SNMP

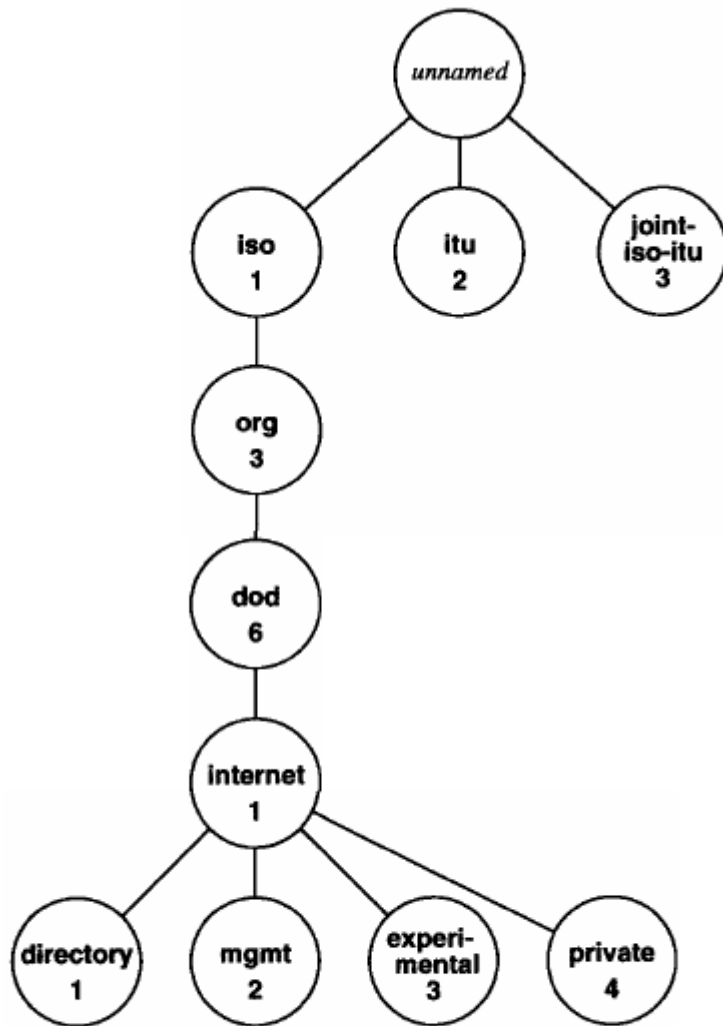


Figure 30.4 Part of the hierarchical object identifier namespace used to name MIB variables. An object's name consists of the numeric label along a path from the root to the object.

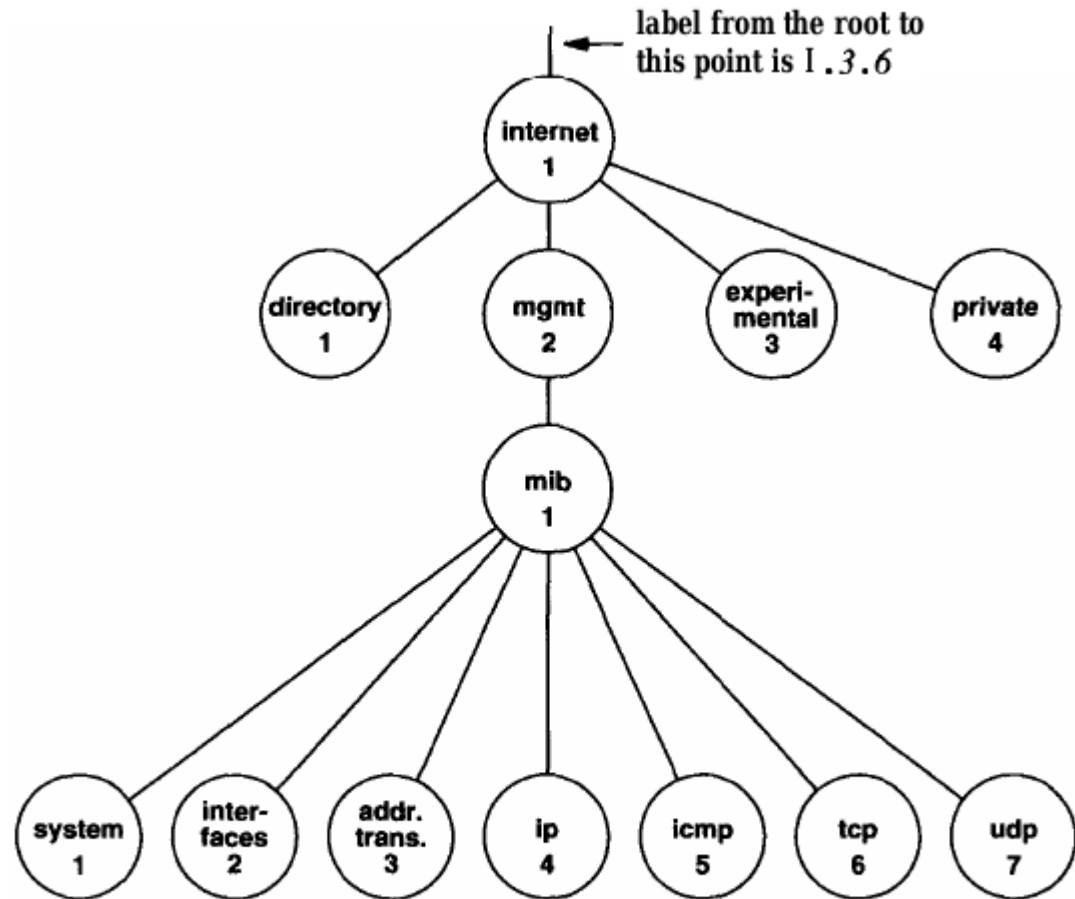


Figure 30.5 Part of the object identifier namespace under the IAB mib node. Each subtree corresponds to one of the categories of MIB variables.

