# Compilateurs MIB et chargement des MIB

## Contenu



## Introduction

La plupart des systèmes d'administration de réseaux (NMS) fournissent une manière pour que l'utilisateur charge les MIB. Le chargement d'un MIB est une manière dont les NMS peuvent se renseigner sur de nouveaux objets MIB, tels que leurs noms, identifiants d'objet (OID), et le genre de type de données (par exemple, compteur).

Le MIB pourrait être analysé quand il est chargé ou il pourrait se produire plus tard, par exemple, quand une application NMS fonctionne. Le logiciel qui exécute analyser est un compilateur MIB.

N'importe quel MIB syntactiquement correct devrait être avec succès analysé par le compilateur MIB de n'importe quel constructeur. Malheureusement, les différents compilateurs MIB peuvent montrer différents caprices.

Cisco fait des efforts continus de s'assurer que le MIB édité aux clients est syntactiquement correct. Cisco évite également les élaborations MIB qui se sont avérées problématiques dans des Produits populaires NMS. En dépit de ces efforts, il n'est pas possible de satisfaire les caprices dans tous les compilateurs MIB dans le domaine.

Ce document aborde certains des problèmes courants et suggère des contournements. Si vous rencontrez l'un de ces problèmes avec le compilateur MIB de votre constructeur (excepté le RFC

14xx contre la question RFC 19xx), elle est due à une insuffisance dans ce compilateur MIB. Vous pouvez souhaiter inviter votre constructeur ou constructeurs pour réparer leurs compilateurs.

# Conditions préalables

## Conditions requises

Les lecteurs de ce document devraient être familiarisés avec le MIB.

## Composants utilisés

Ce document n'est pas limité à des versions de matériel et de logiciel spécifiques.

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

## Conventions

For more information on document conventions, refer to the Cisco Technical Tips Conventions.

# Problèmes communs de chargement MIB

## Commande de chargement

La commande de chargement est la plus importante et le problème courant quand vous chargez le MIB. Beaucoup de MIB utilise les définitions qui sont définies dans l'autre MIB. Ces définitions sont répertoriées dans la clause d'IMPORTATIONS près du dessus du MIB.

Par exemple, si le **marmonnement** MIB importe une définition de MIB **gaffez**, quelques compilateurs MIB exigent de vous de charger le MIB **gaffent** avant de charger le **marmonnement** MIB. Si vous obtenez la commande de chargement fausse, le compilateur réclamera que le MIB importé est non défini.

C'est une liste de MIB duquel importation de beaucoup l'autre de MIB, et de la commande dans laquelle vous devriez les charger. Ceci prend probablement soin de 95 pour cent de questions de commande de chargement (plus de l'autre MIB peut être chargé dans n'importe quelle commande) :

- SNMPv2-SMI.my
- SNMPv2-TC.my
- SNMPv2-MIB.my
- RFC1213-MIB.my
- IF-MIB.my
- CISCO-SMI.my
- CISCO-PRODUCTS-MIB.my
- CISCO-TC.my

**Remarque:** Si vous chargez les versions v1 de ce MIB, le nom du fichier MIB ressemblera à réellement **IF-MIB- V1SMI.my** (« -V1SMI » est ajouté au nom du MIB qui ont été convertis de v2 en v1). L'exception à ceci est le MIB RFC1213-MIB.my, qui existe seulement comme version v1 (c'est-à-dire, il n'y a aucun RFC1213-MIB-V1SMI.my).

Si vous tentez de charger un autre MIB, et si le compilateur se plaint au sujet des éléments non définis, alors l'identifiez quel MIB ce MIB importe et vérifiez que vous avez chargé tout l'autre MIB d'abord.

**Remarque:** Pour chaque MIB, vous pouvez voir la liste précise du MIB qui doit être chargé avant elle — avec la commande précise de compilation — dans le MIB de navigateur > de vue et de téléchargement d'objet SNMP ; **MIB** choisi de **dépendances et de téléchargement MIB de vue**.

## Non-concordances sur des définitions de Datatype

Bien que des définitions de datatype MIB de Cisco ne soient pas mal adaptées, vous pouvez trouver cela pour être le point de droit pour un certain MIB standard RFC. Exemple :

- **Le marmonnement** MIB définit : `SomeDatatype : : = INTEGER(0..100)`
- Le MIB **gaffent** définit : `SomeDatatype : : = INTEGER(1..50)`

Cet exemple est considéré une erreur insignifiante et le MIB charge avec succès avec un message d'avertissement.

L'exemple suivant est considéré une erreur non triviale (quoique les deux définitions sont essentiellement équivalentes), et le MIB n'est pas avec succès analysé.

- **Le marmonnement** MIB définit : `SomeDatatype : : = DisplayString`
- Le MIB **gaffent** définit : `SomeDatatype : : = OCTET STRING (SIZE(0..255))`

Si votre compilateur MIB traite ces derniers comme erreurs, ou si vous souhaitez se débarasser des messages d'avertissement, alors éditez un du MIB qui définissent ce même datatype de sorte que les définitions s'assortissent.

## Redéfinitions d'identifiant d'objet

Vous pouvez rencontrer des redéfinitions OID si vous chargez ce MIB (bien qu'il peut y avoir d'autres exemples où cette erreur se produit) :

- OLD-CISCO-CPU-MIB.my
- OLD-CISCO-ENV-MIB.my
- OLD-CISCO-MEMORY-MIB.my
- OLD-CISCO-SYSTEM-MIB.my

Exemple :

- OLD-CISCO-CPU-MIB.my définit : `IDENTIFIANT de` lcpuOBJECT `: : = {gens du pays 1}`
- OLD-CISCO-ENV-MIB.my définit : `IDENTIFIANT de` lenvOBJECT `: : = {gens du pays 1}`

Quand vous chargez ce deux MIB, le compilateur MIB peut se plaindre au sujet de l'`IDENTIFIANT d'OBJET de lcpu` étant redéfini avec un nouveau `lenv de` nom. Les OLD-CISCO-MEMORY-MIB.my et les OLD-CISCO-SYSTEM-MIB.my donnent pareillement de nouveaux noms à {gens du pays`1}`.

Ceci est traité pendant qu'une erreur insignifiante et le MIB charge avec succès avec un message

d'avertissement.

Si le MIB ne charge pas avec succès, ou si vous souhaitez se débarasser du message d'avertissement, alors éditez un du MIB de sorte que tout les MIB utilise le même nom.

## Définitions de Datatypes intégré

Beaucoup de compilateurs MIB ont la connaissance intégrée de quelques datatypes, tels que DisplayString. Certains de ces compilateurs se plaignent s'ils voient une définition pour ces datatypes dans un MIB. Par exemple, DisplayString est défini dans SNMPv2-TC.

Le contournement est de retirer ou commenter la définition offensante dans le fichier MIB.

## Tailles alternatives

C'est un exemple syntactiquement valide, qui indique qu'une valeur de type `MyDatatype` sera 0, 5, ou 20 octets de longueur :

```
MyDatatype ::= OCTET STRING (SIZE(0 | 5 | 20))
```

Quelques compilateurs MIB ne reçoivent pas cette syntaxe. Habituellement, un contournement suffisant est de sélectionner une des tailles et de retirer les autres. Vous devriez garder la plus grande taille. Par exemple, l'exemple précédent serait changé à ceci :

```
MyDatatype ::= OCTET STRING (SIZE(20))
```

## Identifiants impairs d'objet

Quelques OID sont considérés impairs parce qu'ils ne se rapportent pas à un noeud dans le SMI (comme la plupart des `identifiants d'OBJET`). Cependant, ils sont syntactiquement valides. Un exemple classique est l'identifiant nul d'objet, par exemple, `{0 0}`. Quelques compilateurs MIB n'entretiennent pas les `identifiants d'OBJET` qui ne correspondent pas à un noeud dans le SMI. Ce sont des exemples de syntaxe MIB qui pourraient poser des problèmes pour ces compilateurs :

```
zeroDotZero OBJECT IDENTIFIER ::= { 0 0 }
myMIBObject OBJECT-TYPE
DEFVAL { {0 0} }
```

Le contournement est de retirer ou commenter ces types de références dans le MIB classent.

## Définitions de déroutement

Dans le MIB SNMPv1, des déroutements sont définis avec la macro-instruction `TRAP-TYPE`. Dans le MIB SNMPv2, des déroutements sont définis utilisant le `TYPE DE NOTIFICATION` macro-instruction.

Quelques compilateurs MIB n'aiment pas ces définitions dans les fichiers MIB qu'ils analysent (ils ne prennent en charge pas ces macros-instructions).

Si c'est le cas, vous pouvez retirer les définitions de déroutement ou commenter les définitions (par exemple, mettez le délimiteur de commentaire MIB `--` au début des lignes).

### RFC 14xx a basé des compilateurs contre les compilateurs basés par 19xx RFC

RFC 1442 définissent jusqu'en 1452 le SNMPv2 basé sur interlocuteur. Ces RFC obsoleted par les RFC plus nouveaux de projet de norme 1902 à 1908.

Quant à la syntaxe MIB, il y a très peu de différences entre ces deux versions de SNMPv2 ; il y en a, cependant. Le MIB de Cisco est actuellement basé sur des règles RFC 19xx.

**Remarque:** Il y a quelques années, quand le MIB de Cisco était RFC 14xx-based, quelques compilateurs RFC 19xx-based se plaindraient au sujet de l'`Unsigned32 : : =` ligne `TEXTUAL-CONVENTION` dans le MIB CISCO-TC.my et PNNI-MIB.my. C'est parce qu'Unsigned32 est un datatype de prédéfinis dans RFC 19xx. Pour cette raison, Cisco avait des versions alternatives de ce MIB (CISCO-TC-NO-U32.my et PNNI-MIB-NO-U32.my) sans la définition pour Unsigned32, à charger dans les compilateurs qui savent déjà ce type de données. Ce ne s'applique plus.

## Chargeant et compilant le MIB dans la tierce partie NMS

Le meilleur et la plupart de façon efficace de charger le MIB, les déroutements, et les icônes de Cisco dans la tierce partie NMS est d'utiliser l'utilitaire d'intégration de CiscoWorks (utilitaire d'intégration), qui est disponible en tant qu'élément des services communs de CiscoWorks (ou autonome de http://www.cisco.com/cgi-bin/tablebuild.pl/cw2000-utility), avec l'adaptateur de service correspondant d'intégration de http://www.cisco.com/tacpage/sw-center/cw2000/cmc3rd.shtml et des plus défuntes données d'intégration de la gestion de réseau empaquetez (NMIDB). Documentation de service d'intégration de contrôle pour plus de détails.

Alternativement, vous pouvez consulter la documentation de la tierce partie NMS sur le chargement et la compilation MIB. Ce document comporte des instructions pour le HP OpenView et l'IBM NetView ; mais vous devriez encore consulter la documentation de HP ou IBM, car les Produits peuvent changer.

### Du GUI du HP OpenView ou de l'IBM NetView

Suivez ces étapes pour charger le MIB de Cisco que vous voulez :

1. Copiez les fichiers dans le répertoire /usr/OV/snmp_mibs de la station de Gestion de réseau.C'est le répertoire par défaut où le HP OpenView et l'IBM NetView recherchent des documents MIB. Si vous les placez ailleurs, spécifiez les noms de chemin explicites dans l'interface graphique de loadmib.
2. Placez les autorisations de sorte que vous ayez l'accès en lecture au MIB.
3. Du menu GUI, choisissez les **options > MIB de charge/décharge**.
4. Suivez les instructions dans la documentation de plate-forme, de compiler ou charger le MIB de Cisco.

### De l'interface de ligne de commande du HP OpenView ou de l'IBM NetView

Émettez **/opt/OV/bin/xnmloadmib - chargez la** commande de *nom du fichier*, de charger le fichier MIB.

## Informations connexes

- Support et documentation techniques - Cisco Systems