Présentation de l’ILMI sur les interfaces ATM

Table des matières

[Introduction](#)
[Avant de commencer](#)
[Conventions](#)
[Conditions préalables](#)
[Composants utilisés](#)
[Configuration du circuit virtuel permanent ILMI](#)
[Présentation de la MIB de liaison](#)
[Couche physique](#)
[Couche ATM](#)
[Connexions de chemin virtuel (VPC)](#)
[Connexions de canal virtuel (VCC)](#)
[Enregistrement des adresses sur les interfaces UNI](#)
[pièges coldStart](#)
[Réponses erronées](#)
[Exemple de négociation](#)
[ILMI sur les interfaces NNI](#)
[Informations connexes](#)

Introduction

L’interface ILMI (Interim Local Management Interface) est un protocole défini par l’ATM Forum pour définir et capturer les paramètres de la couche physique, de la couche ATM, du chemin virtuel et du circuit virtuel sur les interfaces ATM. ILMI utilise des messages SNMP (Simple Network Management Protocol) sans UDP (User Datagram Protocol) ni IP, et organise les objets gérés en quatre bases d'informations de gestion (MIB) :

- MIB de conventions textuelles - Définit plusieurs conventions textuelles et ID d'objet, tels que le nombre d'octets pour les adresses de système d'extrémité ATM et les préfixes réseau. Ce document ne couvre pas cette MIB.
- MIB de gestion des liaisons - Fournit quatre groupes d’objets pour toutes les interfaces ATM :
    - Couche physique : l'interface ILMI 4.0 abandonne ou « déprécie » les anciennes valeurs ILMI de couche physique et spécifie l'utilisation de la base MIB d'interface standard (RFC 1213). Voici des exemples de valeurs précédentes dans ce groupe :
        - atmfTransmissionTypes, tels que atmfSonetType, atmfSonetSTS3c, atmfDs3 et atmfT1.

    - atmfMediaTypes, tels que atmfMediaUnknownType, atmfMediaCoaxCable et atmfMediaSingleMode.
  - Couche ATM : indique le nombre de bits disponibles pour les valeurs VPI (Virtual Path Identifier) et VCI (Virtual Channel Identifier) dans l'en-tête de cellule ATM, le nombre maximal de connexions de chemin virtuel (VPC) et de connexions de canal virtuel (VCC) autorisé, le nombre de chemins virtuels permanents configurés et de canaux virtuels permanents, etc.
  - Connexion de chemin virtuel : indique l'état actif ou inactif d'un VPC et ses paramètres de qualité de service (QoS).
  - Connexion au canal virtuel : indique l'état actif ou inactif du VCC et de ses paramètres QoS.
- MIB d'enregistrement d'adresse - Fournit un mécanisme d'enregistrement d'adresse qui permet aux commutateurs de configurer automatiquement les préfixes réseau dans les systèmes d'extrémité.
- MIB Service Registry - Fournit un registre de services à usage général pour localiser des services réseau ATM tels qu'un serveur LECS (LAN Emulation Configuration Server) dans LANE.

Il est important que vous compreniez l'interface ILMI, car les interfaces ATM utilisent ces ID d'objet SNMP (Simple Network Management Protocol) dans des fonctions réseau telles que la configuration automatique d'un client d'émulation LAN (LEC) dans des environnements LANE, les keepalives et même la détection automatique de circuits virtuels permanents (PVC), particulièrement utile dans les applications DSL (Digital Subscriber Line).

Ce document vous aide à comprendre l'ILMI et fournit quelques exemples de débogages pour vous aider à résoudre les problèmes que vous rencontrez.

Remarque : ce document se concentre sur la mise en oeuvre de l'ILMI sur les routeurs Cisco. Pour obtenir des informations générales sur l'interface ILMI, reportez-vous aux spécifications de l'interface ILMI sur la page Approved ATM Forum Specifications ou consultez les livres de la liste Suggested Reading de la page ATM Technologies.

Avant de commencer

Conventions

Pour plus d'informations sur les conventions des documents, consultez la section Conventions relatives aux conseils techniques de Cisco.

Conditions préalables

Il n'existe aucune condition préalable spécifique pour ce document.

Composants utilisés

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

Les informations présentées dans ce document ont été créées à partir de périphériques dans un environnement de travaux pratiques spécifique. All of the devices used in this document started with a cleared (default) configuration. Si vous travaillez sur un réseau actif, assurez-vous de bien comprendre l’impact potentiel de toute commande avant de l’utiliser.

Configuration du circuit virtuel permanent ILMI

Lorsque deux interfaces ATM exécutent le protocole ILMI, elles échangent des paquets ILMI sur la connexion physique. Ces paquets sont constitués de messages SNMP pouvant atteindre 484 octets. Les interfaces ATM encapsulent ces messages dans une queue de bande de couche d’adaptation ATM 5 (AAL5), segmentent le paquet en cellules et planifient la transmission des cellules.

Comme ILMI spécifie des valeurs particulières pour la queue de bande AAL5, nous définissons l'encapsulation comme ILMI lors de la création du circuit virtuel permanent qui transportera les messages ILMI. Par défaut, un circuit virtuel permanent avec les valeurs VPI=0 et VCI=16 transporte les messages ILMI. Nous pouvons voir dans le résultat de la commande show atm ilmi-status ci-dessous que l'ILMI utilise les valeurs par défaut 0/16.

<#root>

```
Switch#

show atm ilmi-status atm 0/0/0


   Interface : ATM0/0/0 Interface Type : Private UNI (Network-side)


ILMI VCC : (0, 16)

 ILMI Keepalive : Disabled
   ILMI State: UpAndNormal
   Peer IP Addr: 10.10.10.4    Peer IF Name: ATM2
   Peer MaxVPIbits: 0          Peer MaxVCIbits: 10
   Peer MaxVPCs: 0             Peer MaxVCCs: 4096
   Peer MaxSvccVpi: 0          Peer MinSvccVci: 0
   Peer MaxSvpcVpi: 0
   Configured Prefix(s) :
   47.0091.8100.0000.0060.3e5a.8f01
```

Sur les commutateurs ATM tels que les gammes Cisco LightStream 1010 et Catalyst 8500, un circuit virtuel permanent ILMI de 0/16 est configuré automatiquement sur chaque interface. La commande show atm vc illustre cette configuration automatique. Notez comment le circuit virtuel ILMI de chaque port se connecte à ATM 2/0/0, qui est le port de gestion interne du commutateur. Les messages ILMI étant des messages de contrôle, ils doivent être envoyés et traités par le

processeur.

<#root>

```
   Switch#
```

show atm vc

```
   Interface VPI VCI Type    X-Interface X-VPI X-VCI Encap Status
   ATM0/0/0 0 5    PVC ATM2/0/0 0    39 QSAAL UP
```

ATM0/0/0 0 16   PVC ATM2/0/0 0   35 ILMI UP

```
   ATM0/0/1 0 5    PVC ATM2/0/0 0    40 QSAAL DOWN
```

ATM0/0/1 0 16   PVC ATM2/0/0 0   36 ILMI DOWN

```
   ATM0/0/1 4 50    PVC ATM2/0/0 0    230 SNAP DOWN
   ATM0/0/2 0 5    PVC ATM2/0/0 0    41 QSAAL UP
```

ATM0/0/2 0 16   PVC ATM2/0/0 0   37 ILMI UP

```
   ATM0/0/2 0 55    PVC ATM0/0/3 0    50 UP
   ATM0/0/2 2 40    PVC ATM2/0/0 0    89 SNAP UP
   ATM0/0/2 4 66    PVC ATM2/0/0 0    66 SNAP UP
   ATM0/0/3 0 5    PVC ATM2/0/0 0    42 QSAAL UP
```

ATM0/0/3 0 16   PVC ATM2/0/0 0   38 ILMI UP

Vous pouvez éventuellement configurer des valeurs autres que les valeurs par défaut pour le circuit virtuel permanent ILMI à l'aide de la procédure suivante. Cliquez ici pour plus d'informations.

<#root>

Switch(config)# interface atm 0/0/0

```
   Switch(config-if)# atm manual-well-known-vc delete
   Okay to delete well-known VCs for this interface? [no]: y
   Switch(config-if)# atm pvc 1 35 interface atm0 any-vci encap ilmi
   Switch(config-if)# end

   Switch#
```

show atm vc interface atm 0/0/0

```
   Interface VPI VCI Type    X-Interface X-VPI X-VCI Encap Status
```

```
   ATM0/0/0 1 35    PVC ATM0    0 150 ILMI UP

 Caution: It is not recommended to change the default values
```

Attention : il n'est pas recommandé de modifier les valeurs par défaut du circuit virtuel permanent ILMI, car cela peut entraîner la panne de votre réseau. Le même circuit virtuel permanent doit être utilisé entre le périphérique final et le commutateur. En outre, la configuration manuelle d'un autre circuit virtuel permanent ILMI rend le dépannage et la maintenance plus difficiles.

Présentation de la MIB de liaison

La MIB de liaison de la MIB ILMI se compose des quatre groupes d'objets suivants :

- Couche physique
- Couche ATM
- Connexions de chemin virtuel (VPC)
- Connexions de canal virtuel (VCC)

Les sections suivantes décrivent les objets de chaque groupe.

Couche physique

L'interface ILMI 4.0 abandonne ou « déprécie » les anciennes valeurs ILMI de couche physique dans le groupe de ports et spécifie l'utilisation de la base MIB d'interface standard (RFC 1213). Ce groupe comprend également des objets qui permettent aux systèmes voisins de tenir à jour une table de systèmes adjacents pour faciliter la découverte automatique et le traçage des connexions ATM.

- atmfPortMonNomSi
- atmfPortMyIfIdentifier
- AdresseNmIpAtmf
- atmfIdentificateurMonSystème

La commande show atm ilmi-status affiche les valeurs envoyées par l'homologue pour ces objets.

<#root>

```
   Switch#
show atm ilmi-status atm 0/0/0

   Interface : ATM0/0/0 Interface Type : Private UNI (Network-side)
   ILMI VCC : (0, 16) ILMI Keepalive : Disabled
   ILMI State: UpAndNormal
```

```
Peer IP Addr: 10.10.10.4 Peer IF    Name: ATM2


   Peer MaxVPIbits: 0    Peer MaxVCIbits: 10
   Peer MaxVPCs: 0    Peer MaxVCCs: 4096
   Peer MaxSvccVpi: 0    Peer MinSvccVci: 0
   Peer MaxSvpcVpi: 0
   Configured Prefix(s) :
   47.0091.8100.0000.0060.3e5a.8f01
```

Le résultat de debug atm ilmi capture également les valeurs telles qu'elles sont annoncées.

<#root>

```
   1w1d: ILMI(ATM0/0/0): KeepAlive disabled
   1w1d: ILMI: Sending Per-Switch prefix
   1w1d: ILMI: Registering prefix with end-system 47.0091.8100.0000.0060.3e5a.8f01


1w1d: ILMI: The Neighbor's IfName on Intf (ATM0/0/0) is ATM2
   1w1d: ILMI: The Neighbor's IP on Intf (ATM0/0/0) is 168430084
```

atmfMySystemIdentifier est un identifiant 48 bits provenant de l’espace d’adressage MAC universellement administré par l’IEEE (Institute of Electrical and Electronic Engineers), qui identifie de manière unique le périphérique ATM.

Couche ATM

Les attributs suivants d’une interface ATM forment le groupe de couches ATM, qui stocke ses valeurs dans la table atmfAtmLayerGroup. Chaque interface a une entrée atmfAtmLayerIndex dans la table.

- index d'interface
- Nombre maximal de bits VPI actifs
- Nombre maximal de bits VCI actifs
- Nombre maximal de VPC
- Nombre maximal de VCC
- Nombre de VPC configurés
- Nombre de VCC configurés
- VPI SVPC maximum
- VPI SVCC maximum

- VCI SVCC minimum
- Type d’interface ATM
- Type de périphérique ATM
- version ILMI
- version de signalisation UNI
- version de signalisation NNI

Lors du choix des valeurs maximales à utiliser, chaque côté compare les valeurs de l'homologue avec ses propres valeurs. Définissez le nombre réel sur la valeur commune la plus élevée pour garantir l'interopérabilité.

Connexions de chemin virtuel (VPC)

Les attributs suivants d'un VPC forment le Virtual Path Group, qui stocke les valeurs dans la table atmfVpcGroup. Chaque VPC est indexé dans la table par un atmfVpcPortIndex pour identifier le port physique et un atmfVpcVpi pour identifier le numéro VPI.

- index d'interface
- Valeur VPI
- État opérationnel
- Descripteur de trafic de transmission
- Descripteur de trafic de réception
- Indicateur Best effort
- Classe Transmit QoS
- Classe de réception QoS
- Catégorie de service

Connexions de canal virtuel (VCC)

Les attributs suivants d'un VCC forment le groupe de canaux virtuels, qui stocke les valeurs dans atmfVccGroup. Chaque VCC est indexé dans la table par l'index d'interface (atmfVccPortIndex), la valeur VPI (atmfVccVpi) et la valeur VCI (atmfVccVci). Seuls les circuits virtuels permanents sont représentés dans ce groupe, y compris les circuits virtuels de signalisation réservés ou bien connus, les circuits virtuels ilmi et les circuits virtuels LECS.

- index d'interface

- Valeur VPI
- État opérationnel
- Descripteur de trafic de transmission
- Descripteur de trafic de réception
- Indicateur Best effort
- Classe Transmit QoS
- Classe de réception QoS
- Catégorie de service

Enregistrement des adresses sur les interfaces UNI

La base MIB Address Registration fournit des objets SNMP pour l'échange dynamique d'informations d'adresse ATM. Ces informations se composent de deux tableaux :

- Network Prefix (Préfixe réseau) : implémenté sur le système d'extrémité ATM via le groupe atmfNetPrefixGroup. Le commutateur ATM envoie un message SetRequest avec le préfixe de 13 octets d'ordre haut configuré sur ce port de commutateur. Lors de l'initialisation, l'enregistrement des préfixes réseau intervient en premier.

  <#root>

  ```
  1w1d: ILMI(ATM0/0/0): KeepAlive disabled

  1w1d: ILMI: Sending
  Per-Switch prefix 1w1d: ILMI: Registering prefix with end-system


  47.0091.8100.0000.0060.3e5a.8f01 1w1d: ILMI: The Neighbor's IfName
  on Intf (ATM0/0/0) is ATM2 1w1d: ILMI: The Neighbor's IP on
  Intf (ATM0/0/0) is 168430084
  ```

- Adresse ATM : mise en oeuvre sur le commutateur ATM via le groupe d'adresses atmf. Le système d'extrémité ATM reçoit d'abord un SetRequest avec le préfixe réseau et enregistre ce préfixe dans sa table de préfixes. Ensuite, le système d'extrémité ATM combine le préfixe avec sa partie ESI (End-Station Identifier) et envoie une requête SetRequest avec l'adresse ATM complète de 20 octets. Enfin, le commutateur ATM choisit d'enregistrer l'adresse dans sa table d'adresses ATM. La table d'adresses ATM utilise deux objets clés :
  - atmfAddressAtmAddress : l'objet d'adresse ATM se compose de l'adresse ATM privée complète de 20 octets
  - atmfAddressStatus - L'objet ATM Address Status indique la validité d'une adresse

ATM. Un système d’extrémité ATM configure une nouvelle adresse ATM en envoyant une requête SetRequest avec l’objet ATM Address Status défini sur un état valide. Un système d’extrémité ATM supprime une adresse ATM existante en envoyant une requête SetRequest avec l’objet ATM Address Status défini sur un état non valide.

Le système d’extrémité ATM et le commutateur ATM doivent tous deux tenir à jour des tables d’adressage précises puisque les adresses sont utilisées dans les champs d’élément d’information Numéro de l’appelant et Numéro de l’appelé des messages de signalisation envoyés lors de l’établissement de circuits virtuels commutés.

L'objet atmfAddressRegistrationAdminStatus indique la prise en charge des groupes Prefix et Address. L’interface ILMI 4.0 impose l’utilisation des groupes de préfixes et d’adresses au niveau d’une interface UNI privée. Si l'extrémité distante renvoie une erreur noTelName indiquant qu'il s'agit d'un périphérique antérieur à ILMI 4.0, l'extrémité proche doit supposer que l'extrémité distante prend en charge l'enregistrement d'adresse. Si un seul côté prend en charge l'enregistrement d'adresse, la spécification ILMI 4.0 suggère que le côté prenant en charge signale une condition d'alarme UNI-misconfiguration ou choisisse de tenter l'enregistrement quand même, puisque le côté distant devrait simplement renvoyer les erreurs noThisName à de telles demandes d'enregistrement.

Commutateur ATM (côté réseau)	
Action	Lors de la réception d'une requête SetRequest d'un système d'extrémité pour une entrée dans la table d'adresses ATM, le commutateur ATM valide l'adresse annoncée pour empêcher l'enregistrement d'adresses en double.
Si la validation échoue	Répond avec une GetResponse contenant une erreur badValue.
Si la validation réussit	Répond avec un GetResponse indiquant noError et met à jour la table d'adresses.

Lorsqu’un système d’extrémité ATM annule l’enregistrement d’une adresse ATM, le commutateur ATM ne doit pas effacer les connexions/appels associés à l’adresse annulée.

Système d'extrémité ATM (côté utilisateur)	
Action	Valide un SetRequest pour l'objet Network Prefix.
Si la validation échoue	Répond avec un GetResponse contenant l'erreur appropriée.
Si la	Répond avec un GetResponse indiquant

validation réussit	noError et met à jour la table Préfixe réseau si le préfixe n'est pas déjà inscrit.

pièges coldStart

Le protocole SNMP utilise des déroutements pour permettre à un périphérique géré de signaler à la station de gestion les événements inhabituels. Il définit plusieurs déroutements dits génériques, dont l'un est le déroutement coldStart. L'interface ILMI utilise le déroutement coldStart lors de l'initialisation ou de la réinitialisation pour effacer ou vider les entrées existantes dans les tables de préfixe réseau ou d'adresses ATM. Regardons comment cela fonctionne :

- Le système d'extrémité ATM envoie une requête ILMI GetNextRequest pour lire la première instance de l'objet ATM Address Status du commutateur ATM. Si la réponse inclut une valeur, le système d'extrémité ATM envoie une interruption coldStart pour demander au commutateur ATM d'initialiser la table d'adresses ATM.
- Le commutateur ATM envoie une requête ILMI GetNextRequest pour lire la première instance de la table des préfixes réseau du système d'extrémité. Si la réponse inclut une valeur, le commutateur envoie un déroutement coldStart pour demander au système d'extrémité ATM d'initialiser la table de préfixes réseau.

Dans l'exemple de résultat suivant, la configuration automatique ILMI échoue et l'interface ATM 1/0/0 envoie une interruption coldStart à l'interface ATM homologue.

<#root>

```
   May 11 15:11:19: ILMI: Post trap Config Check Failed. Interface Restarted
   May 11 15:11:19: %ATM-4-ILMICONFIGCHANGE: ILMI(ATM1/0/0): Restarting ATM signal.
   May 11 15:11:19: ILMI(ATM1/0/0):Setting Local (Pre-Negotiated) PNNI version as d
   May 11 15:11:19: ILMI(ATM1/0/0):Setting Local (Pre-Negotiated) UNI version as i1
   May 11 15:11:19: ILMI(ATM1/0/0):Registering New port


May 11 15:11:19: ILMI: Sending coldstrart trap to peer
   May 11 15:11:19: ILMI(ATM1/0/0): Sending ilmiColdStart trap
   May 11 15:11:19: ILMI(ATM1/0/0): Sending ilmiColdStart trap


   May 11 15:11:19: ILMI(ATM1/0/0): Querying peer device type.
```

ILMI 4.0 spécifie uniquement le déroutement coldStart et les déroutements spécifiques à l'entreprise (c'est-à-dire spécifiques au fournisseur). Les commutateurs ATM utilisent le déroutement ilmiVccChange, comme indiqué dans l'exemple de résultat suivant.

<#root>

```
   1w1d: %LINK-3-UPDOWN: Interface ATM0/0/0, changed state to up
   1w1d: ILMI: Received Interface Up (ATM0/0/0)
   1w1d: ILMI(ATM0/0/0):Setting Local (Pre-Negotiated) PNNI version as ilmiPnniVersion1point0
   1w1d: ILMI(ATM0/0/0):Setting Local (Pre-Negotiated) UNI version as ilmiUniVersion4point0
```

```
   1w1d: ILMI(ATM0/0/0):Registering New port
   1w1d: ILMI: Sending coldstrart trap to peer
   1w1d: ILMI(ATM0/0/0): Sending ilmiColdStart trap (ATM0/0/0)


1w1d: ILMI(ATM0/0/0): Sending ilmiVCCChange trap (ATM0/0/0)
   1w1d: ILMI(ATM0/0/0): Sending ilmiVCCChange trap (ATM0/0/0)


   1w1d: ILMI(ATM0/0/0): Sending ilmiColdStart trap
```

Utilisez la commande disable-ilmi-enterprise-traps hidden pour désactiver les pièges d'entreprise ILMI.

Attention : les commandes masquées ne sont pas officiellement prises en charge par Cisco.

Réponses erronées

Dans certains cas, la sortie de debug atm ilmi retourne un message semblable à ce qui suit :

```
*Sep 1 01:30:11: ILMI(ATM5/0): Errored response
Function Type = ilmiPeerDeviceTypeInfo
```

En regardant cet exemple de trace de renifleur, nous pouvons voir qu'un en-tête SNMP standard comprend les champs suivants :

```
------------ SNMP Header ------------
   SNMP: Version = 0
   SNMP: Community = ILMI
   SNMP: PDU = GetRequest
   SNMP: Request identifier = 0x348 (840)
   SNMP: Error status = noError (0)
   SNMP: Error index = 0
```

L'ID de requête est un entier qui correspond aux messages envoyés et reçus, et permet en fait à un périphérique ATM d'envoyer rapidement plusieurs messages SNMP dans une ligne, comme nous pouvons le voir ci-dessous.

Le champ d'état d'erreur, lorsqu'il est différent de zéro, indique qu'une exception s'est produite lors du traitement de la demande. Le champ error-status utilise les valeurs d'erreur suivantes :

Valeur	Description
tropGrand	Les résultats d'une opération ne tiennent pas dans un seul message SNMP.

noTelName	L'opération demandée a identifié un nom de variable inconnu, selon le profil de la communauté.
mauvaiseValeur	L'opération demandée a spécifié une syntaxe ou une valeur incorrecte lors de la tentative de modification d'une variable.
lecture seule	L'opération demandée a tenté de modifier une variable pour laquelle le profil de communauté n'autorise pas l'accès en écriture.
genError	Toutes les autres conditions d'erreur.

Une valeur différente de zéro pour le champ d'index d'erreur indique quelle variable de la demande était erronée. Les valeurs non nulles ne sont possibles que pour les valeurs d'erreur noTelName, badValue et readOnly.

Exemple de négociation

Examinons un exemple de messages ILMI échangés entre deux interfaces ATM.

Lors de l'initialisation et de la réinitialisation, une interface ATM envoie plusieurs messages GetRequest avec différents numéros d'ordre. Le résultat de debug snmp packet révèle le contenu unique de chaque message GetRequest. Dans l'exemple de sortie suivant, l'interface ATM 0/0/0 envoie six requêtes avec des numéros d'ordre compris entre 6551 et 6556. Examinons les GetRequests en les divisant en deux ensembles.

Dans le premier jeu, ATM 0/0/0 envoie les deux GetRequests suivantes :

ID de demande	Action et résultats
6551	Interroge l'ID d'objet atmfAtmLayerDeviceType de l'interface ATM homologue. Les systèmes d'extrémité ATM prennent la valeur de l'utilisateur (1), tandis que les commutateurs réseau ATM prennent la valeur du noeud (2).
6552	Interroge l'ID d'objet atmfAtmLayerUniType de l'interface ATM homologue. Les valeurs prises en charge sont public et private.

<#root>

```
1w1d: ILMI(ATM0/0/0): Querying peer device type.
1w1d: ILMI:peerDeviceTypeQuery not completed
```

```
   1w1d: ILMI:peerPortTypeQuery not completed
   1w1d: ILMI(ATM0/0/0): From Restarting To WaitDevAndPort


1w1d: ILMI(ATM0/0/0):Sending out Request 6551
   1w1d: ILMI(ATM0/0/0):Sending out Request 6552
   1w1d: SNMP: Response, reqid 6551, errstat 0, erridx    0
   atmfAtmLayerEntry.10.0 = 1
   1w1d: ILMI(ATM0/0/0):Response received for request    6551
   1w1d: SNMP: Response, reqid 6552, errstat 0, erridx    0
   atmfAtmLayerEntry.8.0 = 2
   1w1d: ILMI(ATM0/0/0):Response received for request    6552
   1w1d: ILMI(ATM0/0/0): Peer Device Type is 1
   1w1d: The peer UNI Type on (ATM0/0/0) is 2

   1w1d: ILMI(ATM0/0/0): From WaitDevAndPort To DeviceAndPortComplete

   1w1d: ILMI(ATM0/0/0): From DeviceAndPortComplete To NodeConfigComplete

   1w1d: ILMI: My Device type is set to Node (ATM0/0/0)
```

Dans ce deuxième jeu de résultats, le commutateur envoie cinq GetRequests. Chacun d'entre eux est répertorié dans le tableau ci-dessous. Pour faciliter la compréhension, nous avons mis en surbrillance chaque série de messages dans une couleur différente sous ce tableau.

ID de demande	Action et résultats

6553	Interroge l'objet atmfNetPrefixGroup et implémente peerAddressTableCheck. Nous recevons une GetResponse avec une erreur. En faisant correspondre la sortie debug snmp packet à la sortie debug atm ilmi, nous voyons que SetRequest a interrogé une variable inconnue, selon le profil de la communauté. Le résultat suivant est également mis en évidence en gras ci-dessous.
	
	`1w1d: SNMP: Response, reqid 6553, errstat 2, erridx 1`
	`atmfNetPrefixGroup.1 = NULL TYPE/VALUE`
	`1w1d: ILMI(ATM0/0/0):Response received for request 6553`
	`1w1d: ILMI(ATM0/0/0): Errored response  Function`
	`Type = ilmiAddressTableCheck`

6554	Interroge trois objets dans la table atmfAtmLayer. En faisant correspondre la sortie debug snmp packet à la sortie debug atm ilmi, nous voyons que ces objets sont : • Nombre maximal de bits VPI actifs • Nombre maximal de bits VCI actifs • version de signalisation UNI Le résultat suivant est également mis en surbrillance en bleu ci-dessous.

	1w1d: SNMP: Response, reqid 6554, errstat 0, erridx 0 atmfAtmLayerEntry.6.0 = 0 atmfAtmLayerEntry.7.0 = 10 atmfAtmLayerEntry.9.0 = 4 1w1d: ILMI(ATM0/0/0): The Maximum # of VPI Bits is 0 1w1d: ILMI(ATM0/0/0): The Maximum # of VCI Bits is 10 1w1d: ILMI(ATM0/0/0): The UNI version is negotiated as ilmiUniVersion4point0
6555	Interroge cinq objets supplémentaires dans la table atmfAtmLayer. En faisant correspondre la sortie du paquet debug snmp à la sortie debug atm ilmi, nous voyons que ces objets sont : • Nombre maximal de VPC • Nombre maximal de VCC • VPI SVPC maximum • VPI SVCC maximum • VCI SVCC minimum Le résultat suivant est également mis en surbrillance en italique gras ci-dessous. 1w1d: SNMP: Response, reqid 6555, errstat 0, erridx 0 atmfAtmLayerEntry.2.0 = 0 atmfAtmLayerEntry.3.0 = 4096 atmfAtmLayerEntry.13.0 = 0 atmfAtmLayerEntry.14.0 = 0 atmfAtmLayerEntry.15.0 = 0 1w1d: ILMI(ATM0/0/0):Response received for request 6555 1w1d: ILMI(ATM0/0/0): Peer Max Vpcs is 0 1w1d: ILMI(ATM0/0/0): Peer Max Vccs is 4096 1w1d: ILMI(ATM0/0/0): Peer Max SvpcVpi is 0 1w1d: ILMI(ATM0/0/0): Peer Max SvccVpi is 0 1w1d: ILMI(ATM0/0/0): Peer Min SvccVci is 0
6556	Interroge deux objets dans le groupe de ports physiques : • atmfPortMonNomSi • AdresseNmIpAtmf Le résultat suivant est également mis en surbrillance en italique ci-dessous. 1w1d: SNMP: Response, reqid 6556, errstat 0, erridx 0 atmfPortEntry.7.0 = ATM2 atmfPhysicalGroup.2.0 = 10.10.10.4 1w1d: ILMI(ATM0/0/0):Response received for request 6556 1w1d: ILMI: The Neighbor's IfName on Intf (ATM0/0/0) is ATM2 1w1d: ILMI: The Neighbor's IP on Intf (ATM0/0/0) is 168430084

6557	Envoie un SetRequest avec son préfixe réseau et l'extrémité distante confirme la validation et l'enregistrement de ce préfixe. Le résultat suivant est également mis en surbrillance en italique gras bleu ci-dessous.
	`1w1d: ILMI(ATM0/0/0):Sending out Request 6557`
	`1w1d: SNMP: Response, reqid 6557, errstat 0, erridx 0`
	` atmfNetPrefixEntry.3.0.13.71.0.145.129.0.0.0.0.96.62.90.143.1 = 1`
	`1w1d: ILMI(ATM0/0/0):Response received for request 6557`

<#root>

```
   1w1d: ILMI(ATM0/0/0): Checking Peer Config and Address Table
   1w1d: ILMI:peerAddressTableCheck not completed
   1w1d: ILMI:peerConfigQuery not completed
   1w1d: ILMI:peerRangeConfigQuery not completed
   1w1d: ILMI(ATM0/0/0): From NodeConfigComplete To AwaitRestartAck

   1w1d: ILMI(ATM0/0/0):Sending out Request 6553
   1w1d: ILMI(ATM0/0/0):Sending out Request 6554
   1w1d: ILMI(ATM0/0/0):Sending out Request 6555
   1w1d: ILMI(ATM0/0/0):Sending out Request 6556


1w1d: SNMP: Response, reqid 6553, errstat 2, erridx 1
   atmfNetPrefixGroup.1 = NULL TYPE/VALUE
   1w1d: ILMI(ATM0/0/0):Response received for request 6553
   1w1d: ILMI(ATM0/0/0): Errored response
   Function Type = ilmiAddressTableCheck


 1w1d: SNMP: Response, reqid 6554, errstat 0, erridx 0 atmfAtmLayerEntry.6.0 = 0 atmfAtmLayerEntry.7.0


 1w1d: SNMP: Response, reqid 6555, errstat 0, erridx 0 atmfAtmLayerEntry.2.0 = 0 atmfAtmLayerEntry.3.0 =



   1w1d: SNMP: Response, reqid 6556, errstat 0, erridx 0
   atmfPortEntry.7.0 = ATM2
   atmfPhysicalGroup.2.0 = 10.10.10.4
   1w1d: ILMI(ATM0/0/0):Response received for request 6556

 1w1d: ILMI(ATM0/0/0): The Maximum # of VPI Bits is 0 1w1d: ILMI(ATM0/0/0): The Maximum # of VCI Bits i


 1w1d: ILMI(ATM0/0/0): Peer Max Vpcs is 0 1w1d: ILMI(ATM0/0/0): Peer Max Vccs is 4096 1w1d: ILMI(ATM0/0


   1w1d: ILMI(ATM0/0/0): From AwaitRestartAck To UpAndNormal

   1w1d: ILMI: Auto Port determination enabled
   1w1d: ILMI(ATM0/0/0): Link determination completed
   1w1d: Peer Device Type: ilmiDeviceTypeUser
   1w1d: Peer Port Type: ilmiUniTypePrivate
   1w1d: Peer MaxVpiBits: 0
   1w1d: Peer MaxVciBits: 10
   1w1d: Peer MaxVpcs:    0
```

```
1w1d: Peer MaxVccs:    4096
1w1d: Peer MaxSvpcVpi: 0
1w1d: Peer MaxSvccVpi: 0
1w1d: Peer MinSvccVci: 0
1w1d: Peer UNI version: ilmiUniVersion4point0
1w1d: Neg. UNI Version: ilmiUniVersion4point0
1w1d: Local Device Type: ilmiDeviceTypeNode
1w1d: Local Port Type: ilmiPrivateUNINetworkSide
1w1d: Local System ID:
1w1d: ILMI(ATM0/0/0): KeepAlive disabled
1w1d: ILMI: Sending Per-Switch prefix
1w1d: ILMI: Registering prefix with end-system 47.0091.8100.0000.0060.3e5a.8f01

1w1d: ILMI: The Neighbor's IfName on Intf    (ATM0/0/0) is ATM2
1w1d: ILMI: The Neighbor's IP on Intf (ATM0/0/0) is 168430084
1w1d: ILMI(ATM0/0/0):Sending out Request 6557
1w1d: SNMP: Response, reqid 6557, errstat 0, erridx 0


atmfNetPrefixEntry.3.0.13.71.0.145.129.0.0.0.0.96.62.90.143.1 = 1 1w1d: ILMI(ATM0/0/0):Response receive
```

ILMI sur les interfaces NNI

Les interfaces réseau à réseau (NNI) définissent la connexion entre deux interfaces ATM. Outre tous les paramètres UNI décrits ci-dessus, les ports NNI négocient l'objet atmfAtmLayernniSigVersion pour le groupe de couches ATM. Cet objet indique la dernière version de la spécification de signalisation PNNI du forum ATM prise en charge par ce port ATM. Cet objet ne détermine pas la version de routage PNNI.

Les valeurs de atmfAtmLayerNniSigVersion sont les suivantes :

- fai (2)
- pnniVersion1point0 (3)

Remarque : la version de signalisation UNI utilisée sur les interfaces IISP (Interswitch Signaling Protocol) est déterminée en trouvant la valeur commune la plus élevée annoncée dans l'objet atmfAtmL LayerUniVersion. Le type d'interface est côté utilisateur si l'atmfMySystemIdentifier local est plus grand que l'atmfMySystemIdentifier de l'homologue, et côté réseau si l'atmfMySystemIdentifier local est plus petit que l'atmfMySystemIdentifier de l'homologue.

Remarque : bien que la spécification IISP 1.0 indique que les liaisons IISP 1.0 n'utilisent pas l'interface ILMI, la spécification ILMI 4.0 spécifie éventuellement que les fonctions ILMI autres que l'enregistrement d'adresse peuvent être exécutées sur des liaisons IISP.

Informations connexes

- Pages d'assistance technique ATM

- Support technique - Cisco Systems
- Plus d'informations ATM

À propos de cette traduction
Cisco a traduit ce document en traduction automatisée vérifiée par une personne dans le cadre d'un service mondial permettant à nos utilisateurs d'obtenir le contenu d'assistance dans leur propre langue.

Il convient cependant de noter que même la meilleure traduction automatisée ne sera pas aussi précise que celle fournie par un traducteur professionnel.