

Présentation du protocole ILMI dans les interfaces ATM

Contenu

[Introduction](#)

[Avant de commencer](#)

[Conventions](#)

[Conditions préalables](#)

[Components Used](#)

[Configuration du circuit virtuel permanent ILMI](#)

[Présentation de la MIB de liaison](#)

[Couche physique](#)

[Couche ATM](#)

[Connexions du chemin virtuel \(VPC\)](#)

[Connexions de canal virtuel \(VCC\)](#)

[Enregistrement des adresses sur les interfaces UNI](#)

[Interruptions ColdStart](#)

[Réponses erronées](#)

[Exemple de négociation](#)

[ILMI sur les interfaces NNI](#)

[Informations connexes](#)

[Introduction](#)

L'interface ILMI (Interior Local Management Interface) est un protocole défini par le forum ATM pour définir et capturer les paramètres de couche physique, de couche ATM, de chemin virtuel et de circuit virtuel sur les interfaces ATM. ILMI utilise des messages SNMP (Simple Network Management Protocol) sans protocole UDP (User Datagram Protocol) et IP et organise les objets gérés dans les quatre bases MIB suivantes :

- **MIB de conventions textuelles** - Définit plusieurs conventions textuelles et ID d'objet, tels que le nombre d'octets pour les adresses du 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 : ILMI 4.0 supprime ou « désapprouve » les valeurs ILMI de couche physique antérieures et spécifie l'utilisation de la 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 la cellule ATM, le nombre maximal de connexions

VPC (Virtual Path Connection) et de connexions VCC (Virtual Channel Connexions) autorisées, le nombre de chemins virtuels permanents et de canaux virtuels permanents configurés, etc. Virtual path connection (Connexion du chemin virtuel) : indique l'état up ou down d'un VPC et ses paramètres de qualité de service (QoS). Connexion de canal virtuel : indique l'état actif ou inactif du VCC et de ses paramètres QoS.

- **Base 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.
- **Base MIB du Registre de services** - Fournit un registre de services à usage général pour localiser des services réseau ATM tels qu'un serveur de configuration d'émulation de réseau local (LECS) dans LANE.

Il est important de comprendre l'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 de réseau local (LEC) dans des environnements LANE, des keepalives et même la découverte 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 des exemples de débogages pour vous aider à résoudre les problèmes que vous rencontrez.

Remarque : Ce document porte sur la mise en oeuvre de l'ILMI sur les routeurs Cisco. Pour obtenir des renseignements généraux sur l'ILMI, veuillez consulter la spécification ILMI sur la page [Spécifications approuvées du forum ATM](#) ou consulter les livres de la liste [Suggested Reading](#) de la page Technologies ATM.

[Avant de commencer](#)

[Conventions](#)

Pour plus d'informations sur les conventions des documents, référez-vous aux [Conventions utilisées pour les conseils techniques de Cisco](#).

[Conditions préalables](#)

Aucune condition préalable spécifique n'est requise pour ce document.

[Components Used](#)

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 laboratoire spécifique. All of the devices used in this document started with a cleared (default) configuration. Si vous travaillez dans un réseau opérationnel, 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 se composent de messages SNMP d'une taille pouvant atteindre 484 octets. Les interfaces ATM encapsulent ces messages dans une queue de bande AAL5 (ATM adaptation layer 5), segmentent le paquet en cellules et planifient la transmission des cellules.

Puisque 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 ILMI utilise les valeurs par défaut 0/16.

```
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. Puisque les messages ILMI sont des messages de contrôle, ils doivent être envoyés et traités par le processeur.

```
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.

```
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 défaillance 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 du chemin virtuel \(VPC\)](#)
- [Connexions de canal virtuel \(VCC\)](#)

Les sections suivantes décrivent les objets de chaque groupe.

Couche physique

ILMI 4.0 arrête ou « désapprouve » les valeurs ILMI de couche physique antérieures dans le groupe de ports et spécifie l'utilisation de la MIB d'interface standard (RFC 1213). Ce groupe inclut également des objets qui permettent aux systèmes voisins de tenir à jour une table des systèmes adjacents afin de faciliter la détection automatique et le suivi des connexions ATM.

- atmPortMonNomSi
- atmPortMyIflIdentifier
- atmMyIpnAddress
- atmMySystemIdentifier

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

```
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 MaxVCbits: 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
```

La sortie de **debug atm ilmi** capture également les valeurs au fur et à mesure qu'elles sont annoncées.

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

```
lwd: ILMI: Sending Per-Switch prefix
lwd: ILMI: Registering prefix with end-system 47.0091.8100.0000.0060.3e5a.8f01
lwd: ILMI: The Neighbor's IfName on Intf (ATM0/0/0) is ATM2
lwd: ILMI: The Neighbor's IP on Intf (ATM0/0/0) is 168430084
```

atmfMySystemIdentifier est un identificateur 48 bits provenant de l'espace d'adresses MAC universellement administré par l'Institut des ingénieurs électriques et électroniques (IEEE), 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 possède 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 de la détermination des valeurs maximales à utiliser, chaque partie compare les valeurs de l'homologue à ses propres valeurs. Définissez le nombre réel sur la valeur commune la plus élevée pour garantir l'interopérabilité.

Connexions du chemin virtuel (VPC)

Les attributs suivants d'un VPC forment le groupe de chemins virtuels, 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
- Transmettre le descripteur de trafic
- Réception du descripteur de trafic
- Indicateur du meilleur effort
- Classe QoS de transmission
- Classe QoS de réception
- 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 atmVccGroup. Chaque VCC est indexé dans le tableau par l'index d'interface (atmVccPortIndex), la valeur VPI (atmVccVpi) et la valeur VCI (atmVccVci). Seuls les circuits virtuels permanents sont représentés dans ce groupe, y compris la signalisation bien connue ou réservée, les circuits virtuels ilmi et LECS.

- Index d'interface
- Valeur VPI
- État opérationnel
- Transmettre le descripteur de trafic
- Réception du descripteur de trafic
- Indicateur du meilleur effort
- Classe QoS de transmission
- Classe QoS de réception
- Catégorie de service

Enregistrement des adresses sur les interfaces UNI

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

- **Préfixe réseau** : implémenté sur le système d'extrémité ATM via le groupe atmNetPrefix. Le commutateur ATM envoie un message SetRequest avec le préfixe haut de gamme de 13 octets configuré sur ce port de commutateur. Lors de l'initialisation, l'enregistrement des préfixes réseau a lieu en premier.

```
1wld: ILMI(ATM0/0/0): KeepAlive disabled 1wld: ILMI: Sending  
Per-Switch prefix 1wld: ILMI: Registering prefix with end-system  
47.0091.8100.0000.0060.3e5a.8f01 1wld: ILMI: The Neighbor's IfName  
on Intf (ATM0/0/0) is ATM2 1wld: ILMI: The Neighbor's IP on  
Intf (ATM0/0/0) is 168430084
```
- **Adresse ATM** : mise en oeuvre sur le commutateur ATM via atmAddressGroup. 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 associe le préfixe à sa partie ESI (End Station Identifier) et envoie un 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 : atmAddressAtmAddress - L'objet ATM Address se compose de l'adresse ATM privée complète de 20 octets atmAddressStatus - objet ATM Address Status indique la validité d'une adresse ATM. Un système d'extrémité ATM configure une nouvelle adresse ATM en envoyant un SetRequest avec l'objet ATM Address Status défini sur un état valide. Un système d'extrémité ATM supprime une adresse ATM existante en envoyant un 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, car les adresses sont utilisées dans les champs d'éléments 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. ILMI 4.0 impose l'utilisation des groupes Prefix et Address sur une interface UNI privée. Si le périphérique distant renvoie une erreur `noThisName` indiquant qu'il s'agit d'un périphérique pré-ILMI 4.0, le périphérique principal doit supposer que le périphérique distant 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 support signale une condition d'alarme UNI-erreur de configuration ou décide de tenter de l'enregistrer de toute façon, puisque le poste distant devrait simplement renvoyer les erreurs `noLikeName` à toute demande d'enregistrement de ce type.

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

Lorsqu'un système d'extrémité ATM désenregistre une adresse ATM, le commutateur ATM ne doit pas effacer les connexions/appels associés à l'adresse désenregistrée.

Système d'extrémité ATM (côté utilisateur)	
Action	Valide un <code>SetRequest</code> pour l'objet Network Prefix.
Si la validation échoue	Répond par une <code>GetResponse</code> contenant l'erreur appropriée.
Si la validation réussit	Répond avec une <code>GetResponse</code> indiquant <code>noError</code> et met à jour la table Network Prefix si le préfixe n'est pas déjà enregistré.

Interruptions ColdStart

SNMP utilise des interruptions pour permettre à un périphérique géré de signaler un événement inhabituel à la station de gestion. Il définit plusieurs dérivés dits génériques, dont l'un est le déroutement `coldStart`. ILMI utilise le déroutement `coldStart` lors de l'initialisation ou de la réinitialisation pour effacer ou vider les entrées existantes dans les tables Network Prefix ou ATM Address. Voyons comment cela fonctionne :

- Le système d'extrémité ATM envoie une requête GetNextRequest ILMI 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 un déroutement coldStart pour indiquer au commutateur ATM d'initialiser la table d'adresses ATM.
- Le commutateur ATM envoie une requête GetNextRequest ILMI 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 indiquer au système d'extrémité ATM d'initialiser la table de préfixe réseau.

Dans l'exemple de sortie suivant, la configuration automatique ILMI échoue et l'interface ATM 1/0/0 envoie un déroutement coldStart à l'interface ATM homologue.

```

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 il
May 11 15:11:19: ILMI(ATM1/0/0):Registering New port
May 11 15:11:19: ILMI: Sending coldstart 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 tout déroutement spécifique à l'entreprise (c'est-à-dire spécifique au fournisseur). Les commutateurs ATM utilisent le piège ilmiVccChange, comme illustré dans l'exemple de sortie suivant.

```

1wld: %LINK-3-UPDOWN: Interface ATM0/0/0, changed state to up
1wld: ILMI: Received Interface Up (ATM0/0/0)
1wld: ILMI(ATM0/0/0):Setting Local (Pre-Negotiated) PNNI version as ilmiPnniVersion1point0
1wld: ILMI(ATM0/0/0):Setting Local (Pre-Negotiated) UNI version as ilmiUniVersion4point0
1wld: ILMI(ATM0/0/0):Registering New port
1wld: ILMI: Sending coldstart trap to peer
1wld: ILMI(ATM0/0/0): Sending ilmiColdStart trap (ATM0/0/0)
1wld: ILMI(ATM0/0/0): Sending ilmiVCCChange trap (ATM0/0/0)
1wld: ILMI(ATM0/0/0): Sending ilmiVCCChange trap (ATM0/0/0)
1wld: ILMI(ATM0/0/0): Sending ilmiColdStart trap

```

Utilisez la commande `disable-ilmi-enterprise-traps` cachée pour désactiver les interruptions 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` renvoie un message similaire à ce qui suit :

```

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

```

En regardant cet exemple de trace Sniffer, nous pouvons voir qu'un en-tête SNMP standard inclut 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 demande est un entier qui correspond aux messages envoyés et reçus, et qui permet en fait à un périphérique ATM d'envoyer rapidement plusieurs messages SNMP d'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
trop grand	Les résultats d'une opération ne s'insèrent pas dans un seul message SNMP.
noThisName	L'opération demandée a identifié un nom de variable inconnu, selon le profil de la communauté.
valeur incorrecte	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 à laquelle le profil de communauté n'autorise pas l'accès en écriture.
genError	Toutes les autres conditions d'erreur.

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

Exemple de négociation

Examinons un exemple des 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 de séquence. La sortie du **paquet debug snmp** révèle le contenu unique de chaque message GetRequest. Dans l'exemple de résultat suivant, l'interface ATM 0/0/0 envoie six requêtes avec des numéros d'ordre compris entre 6551 et 6556. Examinons les demandes 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 atmAtmLayerDeviceType de l'interface ATM homologue. Les systèmes d'extrémité ATM prennent la valeur de l'utilisateur (1), tandis que les commutateurs de réseau ATM

	prennent la valeur du noeud (2).
655 2	Interroge l'ID d'objet atmAtmLayerUniType de l'interface ATM homologue. Les valeurs prises en charge sont publiques et privées.

```

1wld: ILMI(ATM0/0/0): Querying peer device type.
1wld: ILMI:peerDeviceTypeQuery not completed
1wld: ILMI:peerPortTypeQuery not completed
1wld: ILMI(ATM0/0/0): From Restarting To WaitDevAndPort
1wld: ILMI(ATM0/0/0):Sending out Request 6551
1wld: ILMI(ATM0/0/0):Sending out Request 6552
1wld: SNMP: Response, reqid 6551, errstat 0, erridx 0
atmAtmLayerEntry.10.0 = 1
1wld: ILMI(ATM0/0/0):Response received for request 6551
1wld: SNMP: Response, reqid 6552, errstat 0, erridx 0
atmAtmLayerEntry.8.0 = 2
1wld: ILMI(ATM0/0/0):Response received for request 6552
1wld: ILMI(ATM0/0/0): Peer Device Type is 1
1wld: The peer UNI Type on (ATM0/0/0) is 2
1wld: ILMI(ATM0/0/0): From WaitDevAndPort To DeviceAndPortComplete

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

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

```

Dans ce deuxième jeu de résultats, le commutateur envoie cinq demandes GetRequests. Chacune est répertoriée 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
655 3	Interroge l'objet atmNetPrefixGroup et implémente peerAddressTableCheck. Nous recevons une réponse GetResponse avec une erreur. Associant la sortie du paquet debug snmp à la sortie debug atm ilmi , nous voyons que SetRequest a interrogé une variable inconnue, selon le profil de communauté. Le résultat suivant est également mis en évidence en gras ci-dessous. <pre> 1wld: SNMP: Response, reqid 6553, errstat 2, erridx 1 atmNetPrefixGroup.1 = NULL TYPE/VALUE 1wld: ILMI(ATM0/0/0):Response received for request 6553 1wld: ILMI(ATM0/0/0): Errored response Function Type = ilmiAddressTableCheck </pre>
655 4	Interroge trois objets dans la table atmAtmLayer. Associant la sortie de paquet debug snmp à la sortie debug atm ilmi , nous voyons que ces objets sont : <ul style="list-style-type: none"> • Nombre maximal de bits VPI actifs • Nombre maximal de bits VCI actifs • Version de signalisation UNI

	<p>Le résultat suivant est également mis en surbrillance en bleu ci-dessous.</p> <pre> lwd: SNMP: Response, reqid 6554, errstat 0, erridx 0 atmfAtmLayerEntry.6.0 = 0 atmfAtmLayerEntry.7.0 = 10 atmfAtmLayerEntry.9.0 = 4 lwd: ILMI(ATM0/0/0): The Maximum # of VPI Bits is 0 lwd: ILMI(ATM0/0/0): The Maximum # of VCI Bits is 10 lwd: ILMI(ATM0/0/0): The UNI version is negotiated as ilmiUniVersion4point0 </pre>
655 5	<p>Interroge cinq objets supplémentaires dans la table atmfAtmLayer. Associant la sortie de paquet debug snmp à la sortie ilmi debug atm, nous voyons que ces objets sont :</p> <ul style="list-style-type: none"> • Nombre maximal de VPC • Nombre maximal de VCC • VPI SVPC maximum • VPI SVCC maximum • VCI SVCC minimum <p>Le résultat suivant est également mis en évidence en <i>gras en italique</i> ci-dessous.</p> <pre> lwd: 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 lwd: ILMI(ATM0/0/0):Response received for request 6555 lwd: ILMI(ATM0/0/0): Peer Max Vpcs is 0 lwd: ILMI(ATM0/0/0): Peer Max Vccs is 4096 lwd: ILMI(ATM0/0/0): Peer Max SvpcVpi is 0 lwd: ILMI(ATM0/0/0): Peer Max SvccVpi is 0 lwd: ILMI(ATM0/0/0): Peer Min SvccVci is 0 </pre>
655 6	<p>Interroge deux objets dans le groupe de ports physiques :</p> <ul style="list-style-type: none"> • atmfPortMonNomSi • atmfMyIpNmAddress <p>Le résultat suivant est également mis en évidence en <i>italique</i> ci-dessous.</p> <pre> lwd: SNMP: Response, reqid 6556, errstat 0, erridx 0 atmfPortEntry.7.0 = ATM2 atmfPhysicalGroup.2.0 = 10.10.10.4 lwd: ILMI(ATM0/0/0):Response received for request 6556 lwd: ILMI: The Neighbor's IfName on Intf (ATM0/0/0) is ATM2 lwd: ILMI: The Neighbor's IP on Intf (ATM0/0/0) is 168430084 </pre>
655	<p>Envoie un SetRequest avec son préfixe de réseau</p>

7

et l'extrémité distante confirme la validation et l'enregistrement de ce préfixe. Le résultat suivant est également mis en évidence en caractères gras bleus ci-dessous.

```
1wld: ILMI(ATM0/0/0):Sending out Request 6557
1wld: SNMP: Response, reqid 6557, errstat 0, erridx 0

atmfNetPrefixEntry.3.0.13.71.0.145.129.0.0.0.0.96.6
2.90.143.1 = 1
1wld: ILMI(ATM0/0/0):Response received for request 6557
```

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

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

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

```
1wld: SNMP: Response, reqid 6554, errstat 0, erridx 0 atmfAtmLayerEntry.6.0 = 0
atmfAtmLayerEntry.7.0 = 10 atmfAtmLayerEntry.9.0 = 4 1wld: ILMI(ATM0/0/0):Response received for request 6554
```

```
1wld: 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 1wld: ILMI(ATM0/0/0):Response received for request 6555
```

```
1wld: SNMP: Response, reqid 6556, errstat 0, erridx 0 atmfPortEntry.7.0 = ATM2
atmfPhysicalGroup.2.0 = 10.10.10.4 1wld: ILMI(ATM0/0/0):Response received for request 6556 1wld:
ILMI(ATM0/0/0): The Maximum # of VPI Bits is 0 1wld: ILMI(ATM0/0/0): The Maximum # of VCI Bits
is 10 1wld: ILMI(ATM0/0/0): The UNI version is negotiated as ilmiUniVersion4point0
1wld: ILMI(ATM0/0/0): Peer Max Vpcs is 0 1wld: ILMI(ATM0/0/0): Peer Max Vccs is 4096 1wld:
ILMI(ATM0/0/0): Peer Max SvpcVpi is 0 1wld: ILMI(ATM0/0/0): Peer Max SvccVpi is 0 1wld:
ILMI(ATM0/0/0): Peer Min SvccVci is 0
```

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

```
1wld: ILMI: Auto Port determination enabled
1wld: ILMI(ATM0/0/0): Link determination completed
1wld: Peer Device Type: ilmiDeviceTypeUser
1wld: Peer Port Type: ilmiUniTypePrivate
1wld: Peer MaxVpiBits: 0
1wld: Peer MaxVciBits: 10
1wld: Peer MaxVpcs: 0
1wld: Peer MaxVccs: 4096
1wld: Peer MaxSvpcVpi: 0
1wld: Peer MaxSvccVpi: 0
1wld: Peer MinSvccVci: 0
1wld: Peer UNI version: ilmiUniVersion4point0
1wld: Neg. UNI Version: ilmiUniVersion4point0
1wld: Local Device Type: ilmiDeviceTypeNode
1wld: Local Port Type: ilmiPrivateUNINetworkSide
1wld: Local System ID:
1wld: 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 received for request 6557
```

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 ATM Forum prise en charge par ce port ATM. Cet objet ne détermine pas la version de routage PNNI.

Les valeurs de atmfAtmLayerNniSigVersion sont les suivantes :

- iisp (2)
- pnniVersion1point0 (3)

Remarque : La version de signalisation UNI utilisée sur les interfaces IISP (Inter-switch Signaling Protocol) est déterminée en recherchant la valeur commune la plus élevée annoncée dans l'objet ATMFatmLayerUniVersion. 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 ILMI, la spécification ILMI 4.0 spécifie éventuellement que les fonctions ILMI autres que l'enregistrement d'adresse peuvent s'exécuter sur des liaisons IISP.

Informations connexes

- [Pages d'assistance technique ATM](#)
- [Support technique - Cisco Systems](#)
- [Plus d'informations ATM](#)