

Groupe de travail Réseau  
**Request for Comments : 4104**  
**Met à jour la RFC 3703**  
 Catégorie : Sur la voie de la normalisation  
 Traduction Claude Brière de L'Isle

M. Pana, MetaSolv  
 A. Reyes, Computer Architecture, UTC  
 A. Barba & D. Moron, UTC  
 M. Brunner, NEC  
 juin 2005

## Schéma d'extension de cœur de politique du protocole léger d'accès à un répertoire (PCELS)

### Statut de ce mémoire

Le présent document spécifie un protocole de l'Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et suggestions pour son amélioration. Prière de se référer à l'édition en cours des "Normes officielles des protocoles de l'Internet" (STD 1) pour connaître l'état de la normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

### Notice de copyright

Copyright (C) The Internet Society (2005).

### Résumé

Le présent document définit un certain nombre de changements et extensions au schéma de cœur de politique du protocole léger d'accès à un répertoire (LDAP, *Lightweight Directory Access Protocol*) (RFC 3703) fondé sur les extensions au modèle définies par les extensions du modèle d'information de cœur de politique (PCIM, *Policy Core Information Model*) (RFC 3460). Ces changements et extensions consistent en nouvelles classes et types d'attributs d'objet LDAP. Certains des éléments de schéma définis dans ce document remettent en œuvre des concepts existants en accord avec leur nouvelle sémantique introduite par la RFC 3460. Les autres éléments de schéma mettent en œuvre de nouveaux concepts, non couverts par la RFC 3703. Le présent document met à jour la RFC 3703.

## Table des Matières

1. Introduction.....	2
1.1 Spécification des exigences.....	2
2. Relations avec les autres documents de cadre de politique.....	2
3. Héritage de hiérarchie pour PCELS.....	3
4. Discussion générale de la transposition des extensions du modèle d'informations de cœur de politique à LDAP.....	6
4.1 Sommaire des transpositions de classe.....	6
4.2 Sommaire des transpositions d'association.....	7
4.3 Sommaire des changements depuis PCLS.....	8
4.4 Relations aux classes PCLS.....	9
4.5 Impact sur les mises en œuvre existantes du schéma LDAP de cœur de politique.....	9
4.6 Association de variables et valeurs de politique pour PolicySimpleCondition et PolicySimpleAction.....	10
4.7 Agrégation de PolicyRules et PolicyGroups dans PolicySets.....	10
4.8 Agrégation de Actions/Conditions dans PolicyRules et CompoundActions/CompoundConditions.....	11
5. Définition des classes.....	14
5.1 Classe abstraite Class pcelsPolicySet.....	15
5.2 Classe structurelle pcelsPolicySetAssociation.....	17
5.3 Les trois classes de groupe de politique.....	18
5.4. Les trois classes de règles de politique.....	18
5.5 Classe structurelle pcelsConditionAssociation.....	21
5.6 Classe structurelle pcelsActionAssociation.....	22
5.7 Classe auxiliaire pcelsSimpleConditionAuxClass.....	22
5.8 Classe auxiliaire pcelsCompoundConditionAuxClass.....	23
5.9 Classe auxiliaire pcelsCompoundFilterConditionAuxClass.....	24
5.10 Classe auxiliaire pcelsSimpleActionAuxClass.....	24
5.11 Classe auxiliaire pcelsCompoundActionAuxClass.....	25
5.12 Classe abstraite pcelsVariable.....	25
5.13 Classe auxiliaire pcelsExplicitVariableAuxClass.....	26
5.14 Classe auxiliaire pcelsImplicitVariableAuxClass.....	27
5.15 Sous classes de pcelsImplicitVariableAuxClass.....	28
5.16 Classe auxiliaire pcelsValueAuxClass.....	32
5.17 Sous classes de pcelsValueAuxClass.....	33

5.18 Les trois classes de conteneur de politique réutilisable.....	36
5.19 Classe structurelle pcelsRoleCollection.....	38
5.20 Classe abstraite pcelsFilterEntryBase.....	39
5.21 Classe structurelle pcelsIPHeadersFilter.....	40
5.22 Classe structurelle pcels8021Filter.....	44
5.23 Classe auxiliaire pcelsFilterListAuxClass.....	46
5.24 Classe auxiliaire pcelsVendorVariableAuxClass.....	47
5.25 Classe auxiliaire pcelsVendorValueAuxClass.....	48
6. Considérations sur la sécurité.....	49
7. Considérations relatives à l'IANA.....	49
7.1 Identifiants d'objet.....	49
7.2 Descripteurs d'identifiant d'objet.....	50
8. Remerciements.....	52
9. Références normatives.....	52
10. Références pour information.....	53
Adresse des auteurs.....	53
Déclaration complète de droits de reproduction.....	53

## 1. Introduction

Le présent document définit un certain nombre de changements et extensions au schéma de cœur de politique du protocole léger d'accès à un répertoire (LDAP, *Lightweight Directory Access Protocol*) [RFC3703] fondé sur les extensions au modèle défini dans les extensions au modèle d'informations de cœur de politique (PCIM, *Policy Core Information Model*) [RFC3460]. Ces changements et extensions consistent en de nouvelles classes d'objets LDAP et types d'attributs [RFC3377]. Certains des éléments du schéma définis dans le présent document remettent en œuvre des concepts existants en accord avec la nouvelle sémantique introduite par la [RFC3460]. Les autres éléments de schéma mettent en œuvre de nouveaux concepts, non couverts par la [RFC3703]. Le présent document met à jour la [RFC3703].

En plus des concepts définis par la [RFC3460], le présent document introduit deux nouvelles classes : pcelsVendorVariableAuxClass et pcelsVendorValueAuxClass. Ces classes fournissent un mécanisme d'extension standard pour les variables et valeurs de politique spécifiques de fabricant qui n'ont pas été spécifiquement modélisées.

Dans le contexte du présent document, le terme "PCELS" (*Policy Core Extension LDAP Schema*, schéma LDAP d'extension de cœur de politique) est utilisé pour se référer à la classe d'objet LDAP, aux définitions de type d'attribut et aux recommandations associées contenues dans le présent document.

### 1.1 Spécification des exigences

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" en majuscules dans ce document sont à interpréter comme décrit dans le BCP 14, [RFC2119].

## 2. Relations avec les autres documents de cadre de politique

Le présent document contient une transposition de schéma LDAP pour les classes définies dans les "extensions au modèle d'information de cœur de politique (PCIM)" [RFC3460]. Le schéma LDAP défini dans le présent document est une extension au "schéma de cœur de politique du protocole léger d'accès à un répertoire (LDAP)" [RFC3703], qui définit la transposition de la "spécification du modèle d'information de cœur de politique -- version 1" [RFC3060] en un schéma LDAP.

Ces trois documents ([RFC3060], [RFC3460] et [RFC3703]) sont des prérequis pour la lecture et la compréhension du présent document.

D'autres documents pourront être produits ultérieurement avec des transpositions du même modèle dans d'autres technologies de mémorisation ou de transport .

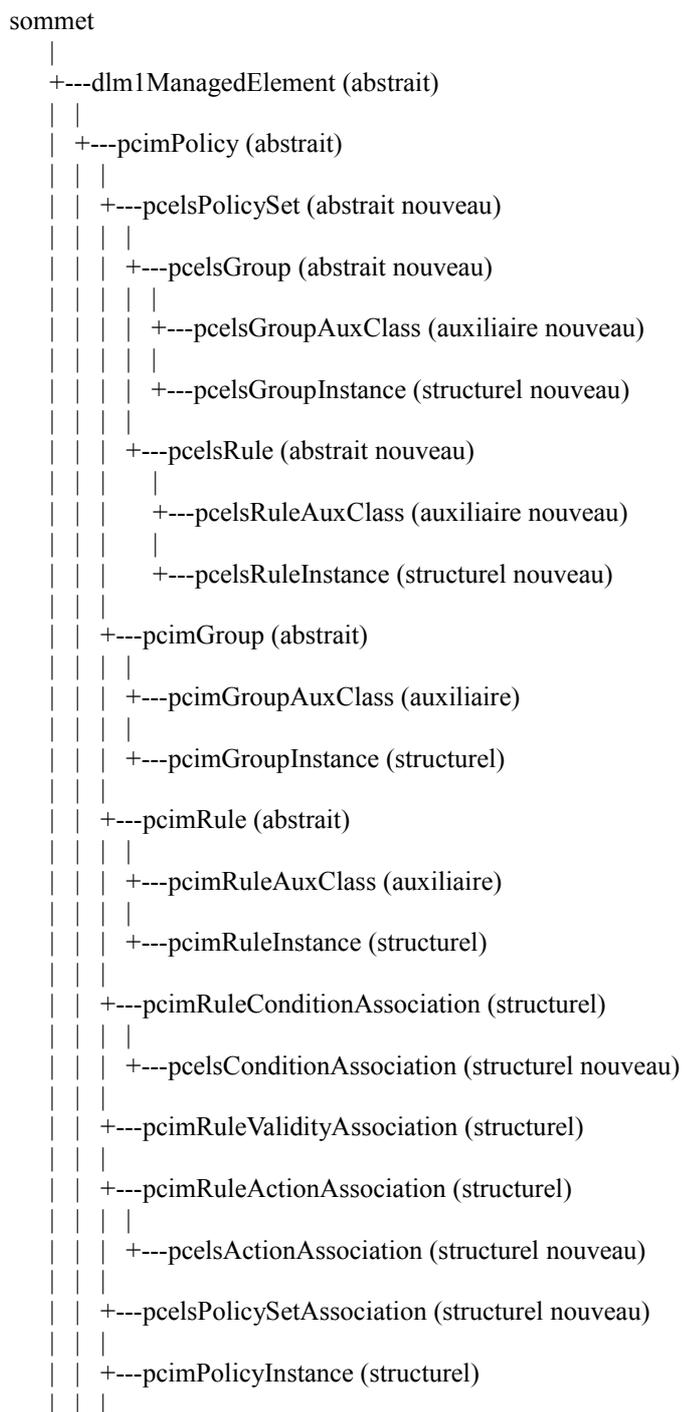
### 3. Héritage de hiérarchie pour PCELS

Les noms de classe d'objet et de type d'attribut définis dans le présent document ont pour préfixe "pcels".

Le diagramme ci-dessous illustre la hiérarchie de classe combinée pour les classes d'objet LDAP définies dans les documents suivants :

- Les noms de classe qui commencent par "pcels" sont définis dans le présent document.
- Les noms de classe qui commencent par "pcim" sont définis dans la [RFC3703].
- Les noms de classe qui commencent pas "d1m1" sont définis dans [CIM\_LDAP].
- La classe nommée "somet" est définie dans la [RFC2256].

Toutes les nouvelles classes d'objet sauf pcelsVendorVariableAuxClass et pcelsVendorValueAuxClass, sont transposées à partir de concepts définis ou modifiés dans la [RFC3460]. Les classes pcelsVendorVariableAuxClass et pcelsVendorValueAuxClass ne sont pas transposées de la [RFC3460]. Elles représentent des concepts introduits dans le présent document.



```

| | +---pcimElementAuxClass (auxiliaire)
| | |
| | +---pcelsRoleCollection (structurel nouveau)
| | |
| | +---pcelsFilterEntryBase (abstrait nouveau)
| | |
| | | +---pcelsIPHeadersFilter (structurel nouveau)
| | | |
| | | +---pcels8021Filter (structurel nouveau)
| | |
| +---dmlManagedSystemElement (abstrait)
| |
| | +---dmlLogicalElement (abstrait)
| | |
| | | +---dmlSystem (abstrait)
| | | |
| | | | +---dmlAdminDomain (abstrait)
| | | | |
| | | | | +---pcimRepository (abstrait)
| | | | | |
| | | | | | +---pcimRepositoryAuxClass (auxiliaire)
| | | | | | |
| | | | | | +---pcimRepositoryInstance (structurel)
| | | | | | |
| | | | | | +---pcelsReusableContainer (abstrait nouveau)
| | | | | | |
| | | | | | | +---pcelsReusableContainerAuxClass (auxiliaire nouveau)
| | | | | | | |
| | | | | | | +---pcelsReusableContainerInstance (structurel nouveau)
| +---pcimConditionAuxClass (auxiliaire)
| |
| | +---pcimTPCAuxClass (auxiliaire)
| | |
| | +---pcimConditionVendorAuxClass (auxiliaire)
| | |
| | +---pcelsSimpleConditionAuxClass (auxiliaire nouveau)
| | |
| | +---pcelsCompoundConditionAuxClass (auxiliaire nouveau)
| | |
| | | +---pcelsCompoundFilterConditionAuxClass (auxiliaire nouveau)
| | |
| | +---pcelsFilterListAuxClass (auxiliaire nouveau)
|
| +---pcimActionAuxClass (auxiliaire)
| |
| | +---pcimActionVendorAuxClass (auxiliaire)
| | |
| | +---pcelsSimpleActionAuxClass (auxiliaire nouveau)
| | |
| | +---pcelsCompoundActionAuxClass (auxiliaire nouveau)
|
| +---pcelsVariable (abstrait nouveau)
| |
| | +---pcelsVendorVariableAuxClass (auxiliaire nouveau)
| | |
| | +---pcelsExplicitVariableAuxClass (auxiliaire nouveau)
| | |
| | +---pcelsImplicitVariableAuxClass (auxiliaire nouveau)
| | |
| | | +---pcelsSourceIPv4VariableAuxClass (auxiliaire nouveau)
| | | |
| | | +---pcelsSourceIPv6VariableAuxClass (auxiliaire nouveau)
| | |
| |

```

```

| +---pcelsDestinationIPv4VariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsDestinationIPv6VariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsSourcePortVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsDestinationPortVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsIPProtocolVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsIPVersionVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsIPToSVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsDSCPVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsFlowIdVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsSourceMACVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsDestinationMACVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsVLANVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsCoSVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsEthertypeVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsSourceSAPVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsDestinationSAPVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsSNAPOUIVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsSNAPTypeVariableAuxClass (auxiliaire nouveau)
| |
| +---pcelsFlowDirectionVariableAuxClass (auxiliaire nouveau)
|
+---pcelsValueAuxClass (auxiliaire nouveau)
| |
| +---pcelsVendorValueAuxClass (auxiliaire nouveau)
| |
| +---pcelsIPv4AddrValueAuxClass (auxiliaire nouveau)
| |
| +---pcelsIPv6AddrValueAuxClass (auxiliaire nouveau)
| |
| +---pcelsMACAddrValueAuxClass (auxiliaire nouveau)
| |
| +---pcelsStringValueAuxClass (auxiliaire nouveau)
| |
| +---pcelsBitStringValueAuxClass (auxiliaire nouveau)
| |
| +---pcelsIntegerValueAuxClass (auxiliaire nouveau)
| |
| +---pcelsBooleanValueAuxClass (auxiliaire nouveau)
|
+---pcimSubtreesPtrAuxClass (auxiliaire)
|
+---pcimGroupContainmentAuxClass (auxiliaire)
|
+---pcimRuleContainmentAuxClass (auxiliaire)

```

**Figure 1. Hiérarchie d'héritage de classe LDAP pour PCELS**

## 4. Discussion générale de la transposition des extensions du modèle d'informations de cœur de politique à LDAP

Les classes d'objet décrites dans le présent document contiennent certaines optimisations pour un répertoire qui utilise LDAP comme protocole d'accès. Un exemple est l'utilisation du rattachement de classe auxiliaire aux entrées LDAP pour la réalisation de certaines des associations définies dans le modèle d'information. Par exemple, l'agrégation d'une SimplePolicyCondition à une PolicyRule réutilisable [RFC3460] peut être réalisée en rattachant une pcelsSimpleConditionAuxClass spécifique à une entrée pcelsRuleInstance.

Noter que d'autres magasins de données pourraient devoir mettre en œuvre différemment les associations.

### 4.1 Sommaire des transpositions de classe

Les classes et leurs propriétés définies dans le modèle d'information [RFC3460] se transposent directement en classes d'objets et types d'attributs LDAP.

Les détails de cette transposition sont discutés cas par cas à la Section 5.

<b>Modèle d'information (PCIM_EXT)</b>	<b>Classes LDAP</b>
PolicySet	pcelsPolicySet
PolicyGroup	pcelsGroup pcelsGroupAuxClass pcelsGroupInstance
PolicyRule	pcelsRule pcelsRuleAuxClass pcelsRuleInstance
SimplePolicyCondition	pcelsSimpleConditionAuxClass
CompoundPolicyCondition	pcelsCompoundConditionAuxClass
CompoundFilterCondition	pcelsCompoundFilterConditionAuxClass
SimplePolicyAction	pcelsSimpleActionAuxClass
CompoundPolicyAction	pcelsCompoundActionAuxClass
PolicyVariable	pcelsVariable pcelsVendorVariableAuxClass
PolicyExplicitVariable	pcelsExplicitVariableAuxClass
PolicyImplicitVariable	pcelsImplicitVariableAuxClass
PolicySourceIPv4Variable	pcelsSourceIPv4VariableAuxClass
PolicySourceIPv6Variable	pcelsSourceIPv6VariableAuxClass
PolicyDestinationIPv4Variable	pcelsDestinationIPv4VariableAuxClass
PolicyDestinationIPv6Variable	pcelsDestinationIPv6VariableAuxClass
PolicySourcePortVariable	pcelsSourcePortVariableAuxClass
PolicyDestinationPortVariable	pcelsDestinationPortVariableAuxClass
PolicyIPProtocolVariable	pcelsIPProtocolVariableAuxClass
PolicyIPVersionVariable	pcelsIPVersionVariableAuxClass
PolicyIPToSVariable	pcelsIPToSVariableAuxClass
PolicyDSCPVariable	pcelsDSCPVariableAuxClass
PolicyFlowIDVariable	pcelsFlowIDVariableAuxClass
PolicySourceMACVariable	pcelsSourceMACVariableAuxClass
PolicyDestinationMACVariable	pcelsDestinationMACVariableAuxClass
PolicyVLANVariable	pcelsVLANVariableAuxClass
PolicyCoSVariable	pcelsCoSVariableAuxClass
PolicyEthertypeVariable	pcelsEthertypeVariableAuxClass
PolicySourceSAPVariable	pcelsSourceSAPVariableAuxClass
PolicyDestinationSAPVariable	pcelsDestinationSAPVariableAuxClass
PolicySNAPOUIVariable	pcelsSNAPOUIVariableAuxClass
PolicySNAPTypeVariable	pcelsSNAPTypeVariableAuxClass
PolicyFlowDirectionVariable	pcelsFlowDirectionVariableAuxClass
PolicyValue	pcelsValueAuxClass pcelsVendorValueAuxClass
PolicyIPv4AddrValue	pcelsIPv4AddrValueAuxClass

PolicyIPv6AddrValue	pcelsIPv6AddrValueAuxClass
PolicyMACAddrValue	pcelsMACAddrValueAuxClass
PolicyStringValue	pcelsStringValueAuxClass
PolicyBitStringValue	pcelsBitStringValueAuxClass
PolicyIntegerValue	pcelsIntegerValueAuxClass
PolicyBooleanValue	pcelsBooleanValueAuxClass
PolicyRoleCollection	pcelsRoleCollection
ReusablePolicyContainer	pcelsReusableContainer
	pcelsReusableContainerAuxClass
	pcelsReusableContainerInstance
FilterEntryBase	pcelsFilterEntryBase
IPHeadersFilter	pcelsIPHeadersFilter
8021Filter	pcels8021Filter
FilterList	pcelsFilterListAuxClass

**Figure 2. Transposition des classes d'extension de modèle d'information à LDAP**

Les classes `pcelsVendorVariableAuxClass` et `pcelsVendorValueAuxClass` ne sont pas transposées de la [RFC3460]. Ces classes sont introduites dans le présent document comme nouveau mécanisme d'extension pour les variables et valeurs de politique spécifiques de fabricant qui n'ont pas été spécifiquement modélisées. Comme tout autre élément de schéma défini dans le présent document ou dans la [RFC3703], un schéma particulier de sous modèle n'aura généralement pas besoin d'utiliser des classes de variables et valeurs spécifiques de fabricant. Les schémas de sous modèle DEVRAIENT appliquer les recommandations du paragraphe 5.10 de la [RFC3460] sur les éléments pris ou non en charge.

## 4.2 Sommaire des transpositions d'association

Les associations dans le modèle d'information se transposent en une ou plusieurs des options suivantes :

1. Attributs qui font référence aux noms distinctifs (DN, *Distinguished Name*)
2. Placement dans l'arborescence d'informations de répertoire (DIT, *Directory Information Tree*) (c'est-à-dire, relation supérieur-subordonné) dans LDAP
3. Rattachement de classe auxiliaire
4. Association de classes d'objets et d'attributs qui font référence aux DN.

Les détails de cette transposition sont discutés au cas par cas à la Section 5.

<b>Association de modèle d'information</b>	<b>Attribut/classe LDAP</b>
PolicySetComponent	pcelsPolicySetComponentList dans pcelsPolicySet et pcelsPolicySetDN dans pcelsPolicySetAsociation
PolicySetInSystem	Placement dans la DIT et pcelsPolicySetDN dans pcelsPolicySetAsociation
PolicyGroupInSystem	Placement dans la DIT et pcelsPolicySetDN dans pcelsPolicySetAsociation
PolicyRuleInSystem	Placement dans la DIT et pcelsPolicySetDN dans pcelsPolicySetAsociation
PolicyConditionStructure	pcimConditionDN dans pcelsConditionAssociation
PolicyConditionInPolicyRule	pcelsConditionList dans pcelsRule et pcimConditionDN dans pcelsConditionAssociation
PolicyConditionInPolicyCondition	pcelsConditionList dans pcelsCompoundConditionAuxClass et pcimConditionDN dans pcelsConditionAssociation
PolicyActionStructure	pcimActionDN dans pcelsActionAssociation
PolicyActionInPolicyRule	pcelsActionList dans pcelsRule et pcimActionDN dans pcelsActionAssociation
PolicyActionInPolicyAction	pcelsActionList dans pcelsCompoundActionAuxClass et pcimActionDN dans pcelsActionAssociation
PolicyVariableInSimplePolicyCondition	pcelsVariableDN dans pcelsSimpleConditionAuxClass
PolicyValueInSimplePolicyCondition	pcelsValueDN dans pcelsSimpleConditionAuxClass
PolicyVariableInSimplePolicyAction	pcelsVariableDN dans pcelsSimpleActionAuxClass
PolicyValueInSimplePolicyAction	pcelsValueDN dans pcelsSimpleActionAuxClass
ReusablePolicy	Placement dans la DIT
ExpectedPolicyValuesForVariable	pcelsExpectedValueList dans pcelsVariable
ContainedDomain	Placement dans la DIT ou pcelsReusableContainerList dans pcelsReusableContainer
EntriesInFilterList	pcelsFilterEntryList dans pcelsFilterListAuxClass
ElementInPolicyRoleCollection	Placement dans la DIT ou pcelsElementList dans pcelsRoleCollection
PolicyRoleCollectionInSystem	Placement dans la DIT

**Figure 3. Transposition des associations d'extension de modèle d'information à LDAP**

Deux associations de la [RFC3460] sont transposées en placement dans la DIT :

- PolicyRoleCollectionInSystem est une association faible et les associations faibles se transposent bien en placement dans la DIT [CIM\_LDAP] (sans se limiter à cette transposition). En l'absence de contraintes supplémentaires, le placement dans la DIT est choisi ici comme transposition optimale d'association.
- ReusablePolicy est transposé en placement dans la DIT pour des raisons d'adaptabilité. On s'attend à ce que les applications associent un grand nombre d'instances de politique à un ReusablePolicyContainer et le placement dans la DIT est un type d'association qui s'adapte bien.

### 4.3 Sommaire des changements depuis PCLS

Ce paragraphe fait un survol des changements par rapport à la [RFC3703] qui sont définis dans le présent document :

1. Le concept d'ensemble de politiques est introduit par deux nouvelles classes d'objet : pcelsPolicySet et pcelsPolicySetAssociation. Ces classes permettent l'agrégation et la priorisation relative des politiques (règles et/ou groupes). L'attribut pcelsPriority est utilisé par les instances de pcelsPolicySetAssociation pour indiquer la priorité d'une politique par rapport aux autres politiques agrégées par le même ensemble. Les applications peuvent utiliser cet attribut pour appliquer un ordre approprié aux politiques agrégées. Ce nouveau mécanisme d'agrégation de politique donne une solution de remplacement au mécanisme d'agrégation défini par la [RFC3703] (qui définit pcimRuleContainmentAuxClass et/ou pcimGroupContainmentAuxClass pour rattacher des composants à un pcimGroup).
2. L'attribut pcimRoles défini dans la [RFC3703] est utilisé ici par la classe d'objets pcelsPolicySet. Donc, le mécanisme de choix de politique fondé sur le rôle est étendu à toutes les sous classes de pcelsPolicySet.
3. Un nouvel attribut pcelsDecisionStrategy est ajouté à la classe pcelsPolicySet comme transposition du mécanisme de décision.
4. Une nouvelle classe pcelsGroup (avec deux sous classes) met en œuvre la sémantique modifiée de PolicyGroup en accord avec la [RFC3460]. Cette nouvelle classe hérite de sa superclasse pcelsPolicySet la capacité d'agréger (avec une priorité relative) d'autres règles ou groupes de politique.
5. Une nouvelle classe pcelsRule (avec deux sous classes) met en œuvre la sémantique modifiée de PolicyRule en accord avec la [RFC3460]. Elle n'inclut pas d'attribut de priorité absolue, mais les instances de sous classes non abstraites de pcelsRule peuvent recevoir une priorité par rapport à chaque autre au sein d'un système (comportement hérité de sa super classe : pcelsPolicySet). La classe pcelsRule hérite aussi de pcelsPolicySet la capacité d'agréger d'autres règles ou groupes de politiques, et donc, la capacité de construire des structures de règles incorporées de complexité arbitraire.
6. Un nouvel attribut pcelsExecutionStrategy est ajouté aux classes pcelsRule et pcelsCompoundActionAuxClass pour permettre la spécification du comportement attendu en cas de multiples actions agrégées par une règle ou par une action composée.
7. Conditions composées : la classe pcelsCompoundConditionAuxClass est ajoutée afin de transposer la classe CompoundPolicyCondition. Une nouvelle classe, pcelsConditionAssociation est utilisée pour agréger les conditions de politique dans une classe pcelsCompoundConditionAuxClass. La même classe est aussi utilisée pour agréger les conditions de politique dans une pcelsRule.
8. Actions composées : la classe pcelsCompoundActionAuxClass est ajoutée afin de transposer la classe CompoundPolicyAction. Une nouvelle classe, pcelsActionAssociation est utilisée pour agréger les actions de politique dans une classe pcelsCompoundActionAuxClass. La même classe est aussi utilisée pour agréger les actions de politique dans une pcelsRule.
9. Conditions simples, actions simples, variables et valeurs : les classes Condition simple, Action simple, variables et valeurs définies dans la [RFC3460] sont directement transposées en classes d'objet LDAP. Ce sont : pcelsSimpleConditionAuxClass, pcelsSimpleActionAuxClass, pcelsVariable et ses sous classes, et pcelsValueAuxClass et ses sous classes.
10. Un mécanisme d'extension général est introduit pour représenter les variables et valeurs de politique qui n'ont pas été spécifiquement modélisées. Le mécanisme est destiné aux extensions spécifiques de fabricant.
11. Répertoire de politique réutilisable : une nouvelle classe (avec deux sous classes) pcelsReusableContainer est créée comme sous classe de pcimRepository. Tout en conservant la compatibilité avec les mises en œuvre plus anciennes

[RFC3703], l'ajout de cette classe reconnaît les intentions de la [RFC3460] d'éviter une potentielle confusion avec le composant de cadre de politique appelé Répertoire de politique. La nouvelle classe permet des associations de plusieurs à plusieurs entre des conteneurs de politique réutilisables.

12. L'association ReusablePolicy définie dans la [RFC3460] est réalisée par la subordination à une instance de la sous classe non abstraite de pcelsReusableContainer. Donc, les composants de politique réutilisables (groupes, règles, conditions, actions, variables et valeurs) peuvent être définis comme entités autonomes ou groupes autonomes d'entités relatives subordonnées (contenues dans la DIT) à un pcelsReusableContainer.
13. Des classes de filtre de niveau appareil sont ajoutées au schéma.
14. La classe pcelsRoleCollection est ajoutée au schéma pour permettre l'association de rôles de politique aux ressources représentées comme des entrées LDAP.

#### 4.4 Relations aux classes PCLS

Plusieurs classes [RFC3703] sont utilisées dans le présent document pour déduire d'autres classes. Si une application PCELS exige une fonctionnalité fournie par une des classes dérivées, la classe [RFC3703] DOIT alors aussi être prise en charge par les mises en œuvre PCELS. Ces classes sont :

pcimPolicy  
pcimRuleConditionAssociation  
pcimRuleActionAssociation  
pcimConditionAuxClass  
pcimActionAuxClass  
pcimRepository

D'autres classes [RFC3703] ne sont ni déduites ni supplantées par des classes définies dans le présent document. Si une application PCELS exige une fonctionnalité fournie par une de ces classes, la classe [RFC3703] DEVRAIT alors être utilisée. Ces classes sont :

pcimRuleValidityAssociation  
pcimTPCAuxClass  
pcimConditionVendorAuxClass  
pcimActionVendorAuxClass  
pcimPolicyInstance  
pcimElementAuxClass  
pcimSubtreesPtrAuxClass

Parmi les classes définies dans le présent document, certaines mettent en œuvre des concepts qui supplantent ceux mis en œuvre par des classes [RFC3703] similaires. Les mises en œuvre PCELS PEUVENT prendre en charge de telles classes [RFC3703]. Ces classes sont :

pcimGroup et ses sous classes  
pcimRule et ses sous classes  
pcimGroupContainmentAuxClass  
pcimRuleContainmentAuxClass  
Les sous classes de pcimRepository

#### 4.5 Impact sur les mises en œuvre existantes du schéma LDAP de cœur de politique

En général, l'intention de PCELS est d'étendre les fonctionnalités offertes par le schéma LDAP de cœur de politique. Pour la plus grande part, la compatibilité avec la [RFC3703] est préservée. Les quelques cas dans lesquels la compatibilité ne peut pas être achevée du fait de changements fondamentaux imposés par la [RFC3460], sont définis ici comme des solutions de remplacement à la mise en œuvre d'origine.

PCELS ne rend pas obsolète ni ne déconseille les concepts mis en œuvre par la [RFC3703]. Les nouveaux éléments du schéma LDAP sont définis dans le présent document d'une façon qui évite, dans la mesure du possible, les interférences avec le fonctionnement normal d'une mise en œuvre raisonnablement bien exécutée de la [RFC3703]. L'intention est de permettre au moins une coexistence sans dommage des deux modèles dans le même répertoire de données.

Cependant, on devrait noter que PCELS introduit les changements suivants qui peuvent avoir un impact sur certaines mises en œuvre de la [RFC3703] :



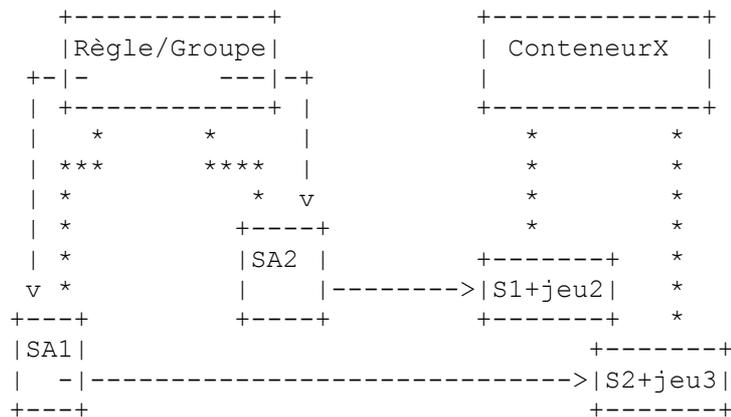
Jeu# : classe auxiliaire pcelsRuleAuxClass ou pcelsGroupAuxClass.

SA# : classe structurelle pcelsPolicySetAssociation.

**Figure 4. Ensemble de politique avec composants spécifiques**

Le pcelsPolicySet incorporant se réfère aux instances de pcelsPolicySetAssociation en utilisant l'attribut pcelsPolicySetList. Ces classes d'association structurelles sont subordonnées (contenues dans la DIT) à une instance d'une sous classe non abstraite de pcelsPolicySet et représentent l'association entre le PolicySet et ses règles/groupes incorporés. Les instances incorporées de sous classes auxiliaires de pcelsPolicySet sont rattachées aux entrées d'association.

Second cas : jeux incorporés réutilisables (règles/groupes).



Légende :

\*\*\*\*\* DIT

+ rattachement auxiliaire

----> référence DN

Jeu# : classe auxiliaire pcelsRuleAuxClass ou pcelsGroupAuxClass.

SA# : classe structurelle PolicySetAssociation.

S# : classe structurelle .

**Figure 5. Ensemble de politique avec composants réutilisables**

Le pcelsPolicySet incorporant se réfère aux instances de pcelsPolicySetAssociation en utilisant l'attribut pcelsPolicySetList. Ces classes d'association structurelles sont subordonnées (contenues dans la DIT) à une instance d'une sous classe non abstraite de pcelsPolicySet et représentent l'association entre le PolicySet et ses règles/groupes incorporés. Les règles/groupes réutilisables sont instanciées ici comme classes auxiliaires et rattachées aux entrées de pcimPolicyInstance dans le conteneur réutilisable. Une autre option est d'utiliser les sous classes structurelles pour définir des règles/groupes réutilisables. Les classes d'association appartenant à un jeu de politique incorporant sont des références aux règles/groupes réutilisables en utilisant l'attribut pcelsPolicySetDN. Une combinaison de composants spécifiques et réutilisables est aussi permise pour le même jeu de politiques.

#### 4.8 Agrégation de Actions/Conditions dans PolicyRules et CompoundActions/CompoundConditions

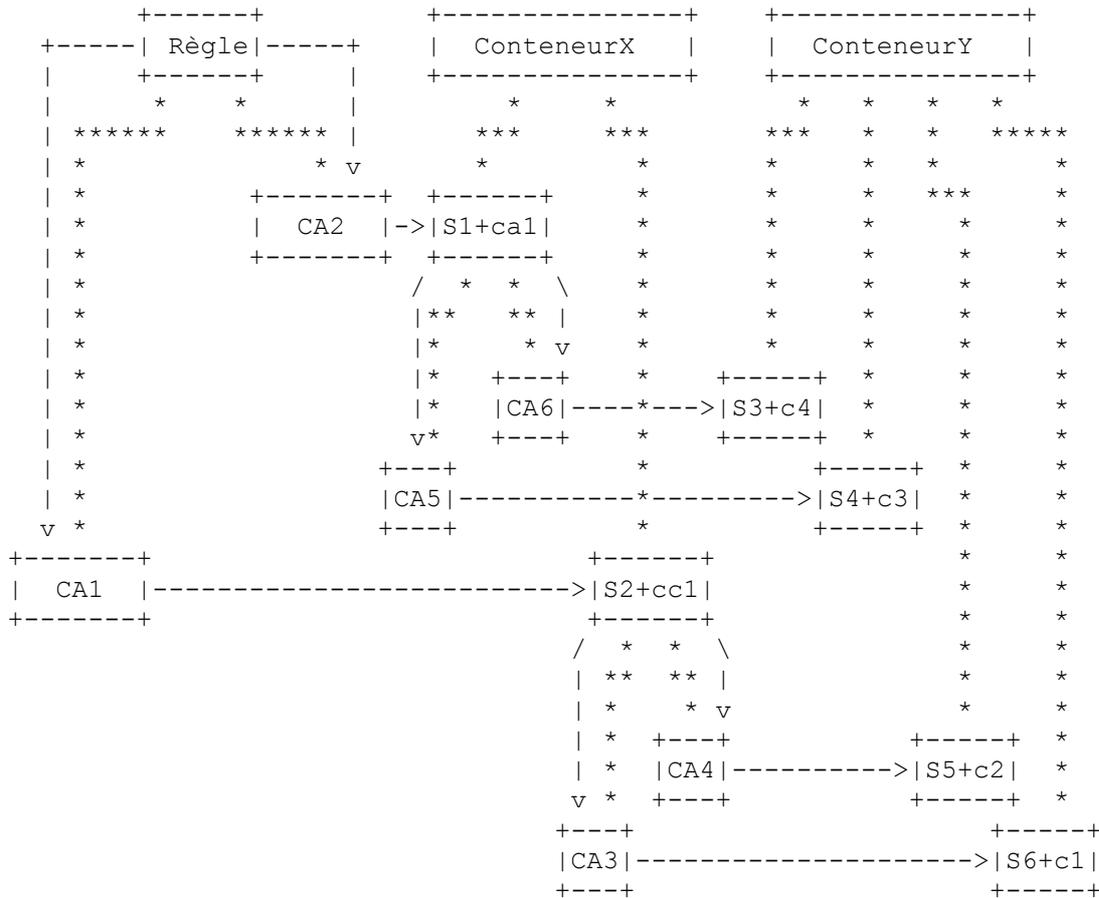
La [RFC3460] définit deux nouvelles classes qui permettent au concepteur de créer des conditions et actions plus complexes. Les classes CompoundPolicyCondition et CompoundPolicyAction sont transposées dans le présent document en les classes pcelsCompoundConditionAuxClass et pcelsCompoundActionAuxClass qui sont des sous classes de pcimConditionAuxClass/pcimActionAuxClass. Les conditions/actions composées définies dans la [RFC3460] étendent la capacité de la règle d'associer, grouper et évaluer les conditions ou exécuter les actions. Les conditions/actions sont associées aux conditions/actions composées de la même façon qu'elles sont associées aux règles.

Dans ce paragraphe, on explique comment mémoriser les instances de ces classes dans un répertoire LDAP. En règle générale, des conditions/actions spécifiques sont subordonnées (contenues dans la DIT) à la règle ou condition/action composée qui les agrège et sont rattachées aux instances de classe d'association. Les conditions/actions réutilisables sont subordonnées aux instances de pcelsReusableContainer et rattachées aux instances de pcimPolicyInstance.

Les exemples ci-dessous illustrent les quatre cas possibles combinant les condition/action spécifique/réutilisable composée/non composée. La règle a deux conditions composées, chacune a deux conditions différentes. Les schémas







Légende :

\*\*\*\*\* DIT

+ rattachement auxiliaire

----> référence de DN

# : Numéro.

CA# : classe structurelle pcelsConditionAssociation.

cc# : classe auxiliaire pcelsCompoundConditionAuxClass.

c# : sous classe de pcimConditionAuxClass.

S# : classe structurelle

**Figure 9. Conditions composées réutilisables avec composants réutilisables**

Toutes les conditions/actions sont réutilisables et sont donc mémorisées dans des conteneurs réutilisables. La figure ci-dessus illustre deux conteneurs de politique réutilisables différents, mais le nombre de conteneurs dans le système est décidé sur la base de raisons administratives. Les conditions, actions, etc. peuvent être mémorisées dans les mêmes conteneurs ou des conteneurs différents sans impact sur la sémantique de définition de politique.

## 5. Définition des classes

La sémantique des classes d'information de politique qui sont à transposer directement du modèle d'information en une représentation LDAP est détaillée dans la [RFC3460]. Par conséquent, le présent document présente seulement une brève référence à cette sémantique. On met ici l'accent sur la transposition à partir du modèle d'information (qui est indépendant du type de répertoire et du protocole d'accès) en une forme à laquelle on peut accéder en utilisant LDAP. Pour diverses raisons incluant une optimisation spécifique de LDAP, cette transposition n'est pas toujours biunivoque. Certaines nouvelles classes et attributs (qui ne faisaient pas partie de la [RFC3060] ou de la [RFC3460]) avaient besoin d'être créées afin de mettre en œuvre la transposition en LDAP. Ces nouvelles classes seulement LDAP sont définies dans le présent document.

Les notes suivantes s'appliquent à toute cette section :

Note 1 : le langage formel pour spécifier les classes, attributs, structures et règles de contenu de la DIT est celui défini dans la [RFC2252]. Dans les définitions qui suivent, les définitions de classe et d'attribut suivent la [RFC2252] mais elles vont à la ligne pour faciliter la lecture.

Note 2 : même si ce n'est pas explicitement noté dans les définitions de classe et d'attribut suivantes, les mises en œuvre peuvent définir les règles de structure et de contenu de DIT lorsque elles sont applicables et prises en charge par l'infrastructure LDAP sous-jacente. Dans de tels cas, les considérations de règle de structure de DIT discutées à la Section 5 de la [RFC3703] doivent être aussi appliquées aux mises en œuvre de PCELS. Les raisons et les détails sont présentés dans [X.501].

Note 3 : chaque fois que possible, une règle d'égalité, une règle de correspondance de sous chaînes et d'ordre sont définies pour un certain attribut. Cela procure plus de souplesse à la mise en œuvre. Cependant, dans certains cas, la sémantique de correspondance de LDAP peut ne pas couvrir tous les besoins de l'application. Par exemple, des valeurs différentes de `pcelsIPv4AddrList` peuvent être sémantiquement équivalentes. La règle de correspondance d'égalité, `caseIgnoreMatch`, associée à ce type d'attribut ne convient pas pour détecter cette équivalence. Les mises en œuvre ne devrait pas s'appuyer seulement sur les règles de syntaxe et de correspondance de LDAP pour être cohérentes avec la présente spécification.

Note 4 : les définitions d'attribut suivantes utilisent seulement les définitions de syntaxe et de règles de correspondance de LDAP tirées des [RFC2252], [RFC2256] et [RFC3698]. Les règles de correspondances X.500 équivalentes sont définies dans [X.520].

Note 5 : certains des types d'attributs suivants DOIVENT se conformer aux contraintes supplémentaires sur divers types de données (par exemple, les seules valeurs valides pour `pcelsDecisionStrategy` sont 1 et 2). Tout comme la sémantique d'attribut, la définition des structures de valeurs, des gammes valides, etc. est couverte par la [RFC3460] pour les propriétés correspondantes tandis que de telles contraintes sont seulement mentionnées brièvement dans le présent document. Dans tous les cas, si une contrainte est violée, l'entrée DEVRAIT être traitée comme invalide et les règles ou groupes de politique qui s'y réfèrent DEVRAIENT être traités comme étant désactivés, ce qui signifie que l'exécution de telles règles ou groupes de politique DEVRAIENT être arrêtée.

Note 6 : certaines des classes d'objet définies dans cette section ne peuvent pas ou ne devraient pas être directement instanciées parce que elles sont soit définies comme abstraites, soit ne mettent pas en œuvre une sémantique autonome (par exemple, `pcelsValueAuxClass`). En ce qui concerne les instances d'objets qui héritent de telles classes, le texte se réfère aux "instances de <nom\_de\_classe>" alors qu'en fait l'expression strictement correcte serait "instances d'objets qui appartiennent aux sous classes non abstraites de <nom\_de\_classe>". L'omission est intentionnelle ; elle rend plus facile la lecture du texte.

## 5.1 Classe abstraite Class `pcelsPolicySet`

La classe `pcelsPolicySet` représente un ensemble de politiques avec une stratégie de décision commune et un ensemble commun de rôles de politique. Cette classe, avec la classe `pcelsPolicySetAssociation` définie dans un paragraphe ultérieur du présent document fournit des informations suffisantes pour permettre aux applications d'appliquer un ordre approprié à un ensemble de politiques. L'ensemble `pcelsPolicySet` est transposé de la classe `PolicySet` de la [RFC3460]. La classe `pcelsPolicySet` est une classe d'objets abstraits et elle est déduite de la classe `pcimPolicy` [RFC3703].

L'attribut `pcelsPolicySetList` d'une instance de `pcelsPolicySet` fait références aux entrées subordonnées de `pcelsPolicySetAssociation`. Les instances de `pcelsPolicySet` agrégées sont soit rattachées aux entrées de `pcelsPolicySetAssociation` comme classes d'objets auxiliaires, soit référencées par les entrées de `pcelsPolicySetAssociation` en utilisant l'attribut `pcelsPolicySetDN`.

La classe `pcelsPolicySet` est définie comme suit :

```
( 1.3.6.1.1.9.1.1
  NOM 'pcelsPolicySet'
  DESC 'Jeu de politiques'
  SUP pcimPolicy
  ABSTRAITE
  PEUT ( pcelsPolicySetName $ pcelsDecisionStrategy $ pcimRoles $ pcelsPolicySetList )
)
```

Un des attributs de la classe `pcelsPolicySet`, `pcimRoles` est défini au paragraphe 5.3 de la [RFC3703]. Dans la classe

pcelsPolicySet l'attribut pcimRole conserve sa syntaxe et sa sémantique définies par les [RFC3703] et [RFC3060].

Le type d'attribut pcelsPolicySetName peut être utilisé comme attribut de dénomination pour les entrées de pcelsPolicySet. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de caseIgnoreMatch, une règle de correspondance d'ordre de caseIgnoreOrderingMatch et une règle de correspondance de sous chaînes de caseIgnoreSubstringsMatch [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type d'attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.1
  NOM 'pcelsPolicySetName'
  DESC 'Nom facile à mémoriser d'un jeu de politiques'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
  VALEUR UNIQUE
)
```

Le type d'attribut pcelsDecisionStrategy indique la méthode d'évaluation pour les politiques agrégées dans le jeu de politiques. Il est transposé de la propriété PolicySet.PolicyDecisionStrategy [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de integerMatch [RFC2252] et une règle de correspondance d'ordre de integerOrderingMatch [RFC3698]. Les attributs de ce type peuvent seulement avoir une seule valeur. Les seules valeurs admises pour les attributs de ce type sont 1 (PremièreCorrespondance) et 2 (TouteCorrespondance). Si cet attribut manque dans une instance de pcelsPolicySet, les applications DOIVENT supposer une stratégie de décision de PremièreCorrespondance pour le jeu de politiques.

Ce type d'attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.2
  NOM 'pcelsDecisionStrategy'
  DESC 'Méthode d'évaluation pour les composants d'un pcelsPolicySet'
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
  VALEUR UNIQUE
)
```

Le type d'attribut pcelsPolicySetList est utilisé dans la réalisation de l'association PolicySetComponent [RFC3460]. Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de distinguishedNameMatch [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs pcelsPolicySetList sont des DN d'entrées de pcelsPolicySetAssociation. Dans un pcelsPolicySet, l'attribut pcelsPolicySetList représente les associations entre ce jeu de politiques et ses composants.

Ce type d'attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.3
  NOM 'pcelsPolicySetList'
  DESC 'Jeu non ordonné de DN d'entrées de pcelsPolicySetAssociation'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.12
)
```

Note : Une instance de pcelsPolicySet agrège d'autres instances de pcelsPolicySet en utilisant les entrées de pcelsPolicySetAssociation (définies au paragraphe suivant). Les applications peuvent trier les composants d'un pcelsPolicySet en utilisant les attributs des entrées de pcelsPolicySetAssociation. Cependant, les mises en œuvre ne devraient pas s'attendre à ce que le magasin de données LDAP fournisse un ordre utile des valeurs de pcelsPolicySetList dans une instance de pcelsPolicySet ou retourne des ensembles d'entrées correspondantes de pcelsPolicySetAssociation dans un ordre significatif. Les applications DEVRAIENT plutôt mettre en œuvre leurs propres moyens pour mettre en ordre après la restitution des règles/groupes de politique sur la base des valeurs de pcelsPolicySetAssociation.pcelsPriority.

## 5.2 Classe structurelle pcelsPolicySetAssociation

La classe pcelsPolicySetAssociation est utilisée pour associer les instances de PolicySet [RFC3460] aux autres entrées. Les entrées de pcelsPolicySetAssociation sont toujours subordonnées à l'entrée agrégeante. Lorsque elle est subordonnée à une instance de pcelsPolicySet, pcelsPolicySetAssociation réalise une association de PolicySetComponent [RFC3460]. Lorsque subordonnée à une instance de dlm1System [CIM\_LDAP], pcelsPolicySetAssociation réalise une association de PolicySetInSystem [RFC3460].

La classe pcelsPolicySetAssociation est une classe d'objets structurelle et elle est déduite de la classe pcimPolicy [RFC3703].

L'agrégation d'une instance réutilisable de pcelsPolicySet est réalisée via l'attribut pcelsPolicySetDN. Une instance non réutilisable de pcelsPolicySet est rattachée (comme sous classe auxiliaire de pcelsPolicySet) directement à une entrée de pcelsPolicySetAssociation.

Lorsque on lit une instance de pcelsPolicySetAssociation qui a un pcelsPolicySet rattaché, l'attribut pcelsPolicySetDN DOIT être ignoré. Les applications DEVRAIENT retirer la valeur de pcelsPolicySetDN d'une pcelsPolicySetAssociation au rattachement d'un pcelsPolicySet à l'entrée.

La classe pcelsPolicySetAssociation est définie comme suit :

```
( 1.3.6.1.1.9.1.2
  NOM 'pcelsPolicySetAssociation'
  DESC 'Associe un jeu de politique à une entrée agrégeante'
  SUP pcimPolicy
  STRUCTURELLE
  DOIT ( pcelsPriority )
  PEUT ( pcelsPolicySetName $ pcelsPolicySetDN )
)
```

Le type d'attribut pcelsPriority indique la priorité d'un composant de jeu de politique. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de integerMatch [RFC2252] et une règle de correspondance d'ordre de integerOrderingMatch [RFC3698]. Les attributs de ce type peuvent seulement avoir des valeurs uniques. Les seules valeurs admises pour les attributs de ce type sont des entiers non négatifs. Dans l'ensemble des entrées de pcelsPolicySetAssociation directement subordonnées à un pcelsPolicySet ou un dlm1System [CIM\_LDAP], les valeurs de pcelsPriority DOIVENT être uniques.

Ce type d'attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.4
  NOM 'pcelsPriority'
  DESC 'Priorité d'un composant'
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
  VALEUR UNIQUE
)
```

Le type d'attribut pcelsPolicySetDN est utilisé dans l'agrégation d'instances de PolicySet [RFC3460]. Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de distinguishedNameMatch [RFC2252]. Les attributs de ce type peuvent seulement avoir des valeurs uniques. Les seules valeurs admises pour les attributs pcelsPolicySetDN sont des DN d'entrées de pcelsPolicySet.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.5
  NOM 'pcelsPolicySetDN'
  DESC 'DN d'une entrée de pcelsPolicySet'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.12
  VALEUR UNIQUE
)
```

### 5.3 Les trois classes de groupe de politique

La classe `pcelsGroup` est la classe de base pour représenter un groupe de politiques. Elle est transposée de la classe modifiée `PolicyGroup` [RFC3460]. La classe `pcelsGroup` est dérivée de la classe `pcelsPolicySet`. Pour maximiser la souplesse, la classe `pcelsGroup` est définie comme abstraite. Une sous classe auxiliaire `pcelsGroupAuxClass` permet le rattachement d'un groupe de politiques à une entrée existante, tandis qu'une sous classe structurelle `pcelsGroupInstance` permet la représentation d'un groupe de politiques comme entrée autonome.

La classe `pcelsGroup` est définie comme suit :

```
( 1.3.6.1.1.9.1.3
  NOM 'pcelsGroup'
  DESC 'Classe de base pour représenter un groupe de politiques'
  SUP pcelsPolicySet
  ABSTRAITE
  PEUT ( pcimGroupName )
)
```

La classe `pcelsGroupAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.4
  NOM 'pcelsGroupAuxClass'
  DESC 'Classe auxiliaire pour représenter un groupe de politiques'
  SUP pcelsGroup
  AUXILIAIRE
)
```

La classe `pcelsGroupInstance` est définie comme suit :

```
( 1.3.6.1.1.9.1.5
  NOM 'pcelsGroupInstance'
  DESC 'Classe structurelle pour représenter un groupe de politiques'
  SUP pcelsGroup
  STRUCTURELLE
)
```

Le type d'attribut `pcimGroupName` utilisé par la classe `pcelsGroup` est défini au paragraphe 5.2 de la [RFC3703]. Dans la classe d'objets `pcelsGroup`, cet attribut conserve sa syntaxe et sa sémantique comme définies par les [RFC3703] et [RFC3060].

Note : les mises en œuvre de PCELS DEVRAIENT prendre en charge `pcelsGroup` et ses deux sous classes et PEUVENT aussi prendre en charge `pcimGroup` et ses deux sous classes [RFC3703]. Les applications qui choisissent de prendre en charge `pcelsGroup` et ses deux sous classes DOIVENT utiliser le mécanisme d'agrégation fourni par `pcelsPolicySetAssociation` pour agréger les groupes de politiques ou règles de politique dans les groupes de politiques représentés comme des instances de `pcelsGroup`.

### 5.4 Les trois classes de règles de politique

La classe `pcelsRule` est la classe de base pour représenter une règle de politique. Elle est transposée de la classe modifiée `PolicyRule` [RFC3460]. La classe `pcelsRule` est dérivée de la classe `pcelsPolicySet`. Pour maximiser la souplesse, la classe `pcelsRule` est définie comme abstraite. Une sous classe auxiliaire `pcelsRuleAuxClass` permet le rattachement d'une règle de politique à une entrée existante, tandis qu'une sous classe structurelle `pcelsRuleInstance` permet la représentation d'une règle de politique comme entrée autonome.

Lorsque on lit une instance de `pcelsRule` qui a une `pcimConditionAuxClass` rattachée, du point de vue de la règle de politique, l'attribut `pcelsConditionList` DOIT être ignoré. Par exemple, si il est présent, l'attribut NE DOIT PAS être considéré comme une association entre cette règle de politique et une condition de politique. De telles situations peuvent se produire, par exemple, quand une `pcelsCompoundConditionAuxClass` est rattachée à une instance de `pcelsRule`.

Lorsque on lit une instance de `pcelsRule` qui a une `pcimActionAuxClass` rattachée, du point de vue de la règle de politique l'attribut `pcelsActionList` DOIT être ignoré. Par exemple, si il est présent, l'attribut NE DOIT PAS être considéré comme une association entre cette règle de politique et une action de politique. De telles situations peuvent se produire, par

exemple, quand une `pcelsCompoundActionAuxClass` est rattachée à une instance de `pcelsRule`.

La classe `pcelsRule` est définie comme suit :

```
( 1.3.6.1.1.9.1.6
  NOM 'pcelsRule'
  DESC 'Classe de base pour représenter une règle de politique'
  SUP pcelsPolicySet
  ABSTRAITE
  PEUT ( pcimRuleName $ pcimRuleEnabled $ pcimRuleUsage $ pcimRuleMandatory $ pcelsRuleValidityPeriodList
        $ pcelsConditionListType $ pcelsConditionList $ pcelsActionList $ pcelsSequencedActions
        $ pcelsExecutionStrategy )
)
```

La classe `pcelsRuleAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.7
  NOM 'pcelsRuleAuxClass'
  DESC 'Classe auxiliaire pour représenter une règle de politique'
  SUP pcelsRule
  AUXILIAIRE
)
```

La classe `pcelsRuleInstance` est définie comme suit :

```
( 1.3.6.1.1.9.1.8
  NOM 'pcelsRuleInstance'
  DESC 'Classe structurelle pour représenter une règle de politique'
  SUP pcelsRule
  STRUCTURELLE
)
```

Quatre des attributs utilisés par la classe `pcelsRule` sont définis au paragraphe 5.3 de la [RFC3703]. Ces attributs sont : `pcimRuleName`, `pcimRuleEnabled`, `pcimRuleUsage` et `pcimRuleMandatory`. Dans la classe d'objets `pcelsRule`, ces attributs conservent leur syntaxe et leur sémantique comme définies par les [RFC3703] et [RFC3060].

Les attributs `pcimRuleValidityPeriodList`, `pcimRuleConditionListType`, `pcimRuleConditionList`, `pcimRuleActionList` et `pcimRuleSequencedActions` définis dans la [RFC3703] ne sont pas utilisés par `pcelsRule`. À la place, cette classe utilise les nouveaux attributs `pcelsRuleValidityPeriodList`, `pcelsConditionListType`, `pcelsConditionList`, `pcelsActionList` et `pcelsSequencedActions`. Excepté pour `pcelsRuleValidityPeriodList`, les nouveaux attributs sont aussi utilisés pour un objet similaire par `pcelsCompoundConditionAuxClass` ou `pcelsCompoundActionAuxClass`.

Le type d'attribut `pcelsRuleValidityPeriodList` est utilisé dans la réalisation de l'association `PolicyRuleValidityPeriod` ([RFC3460] et [RFC3060]). Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de `distinguishedNameMatch` [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs `pcelsRuleValidityPeriodList` sont les DN des entrées de `pcimRuleValidityAssociation`. Dans une `pcelsRule`, l'attribut `pcelsRuleValidityPeriodList` représente les associations entre cette règle de politique et ses conditions de période de temps.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.62
  NOM 'pcelsRuleValidityPeriodList'
  DESC 'Ensemble non ordonné de DN d'entrées de pcimRuleValidityAssociation'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.12
)
```

Le type d'attribut `pcelsConditionListType` indique si l'ensemble de conditions agrégées est en forme normale disjonctive ou conjonctive. Il est transposé de la propriété `PolicyRule.ConditionListType` [RFC3060] (identique à la propriété `CompoundPolicyCondition.ConditionListType` définie dans la [RFC3460]). Ce type d'attribut a la syntaxe `Integer` [RFC2252]. Il a une règle de correspondance pour égalité de `integerMatch` [RFC2252] et une règle de correspondance d'ordre de `integerOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules

valeurs admises pour les attributs de ce type sont 1 (disjonctive) et 2 (conjonctive). Si cet attribut manque dans une instance de pcelsRule, les applications DOIVENT supposer que l'ensemble des conditions agrégées est en forme normale disjonctive.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.6
  NOM 'pcelsConditionListType'
  DESC 'Indique le type d'agrégation de conditions'
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
  VALEUR UNIQUE
)
```

Le type d'attribut pcelsConditionList est utilisé dans la réalisation de l'association PolicyConditionStructure [RFC3460]. Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de distinguishedNameMatch [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs pcelsConditionList sont des DN d'entrées de pcelsConditionAssociation. Dans une pcelsRule, l'attribut pcelsConditionList représente les associations entre cette règle de politique et ses conditions.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.7
  NOM 'pcelsConditionList'
  DESC 'Ensemble non ordonné de DN d'entrées de pcelsConditionAssociation'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.12
)
```

Le type d'attribut pcelsActionList est utilisé dans la réalisation de l'association PolicyActionStructure [RFC3460]. Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de distinguishedNameMatch [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs pcelsActionList sont les DN des entrées de pcelsActionAssociation. Dans une pcelsRule, l'attribut pcelsActionList représente les associations entre cette règle de politique et ses actions.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.8
  NOM 'pcelsActionList'
  DESC 'Ensemble non ordonné de DN d'entrées de pcelsActionAssociation'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXER 1.3.6.1.4.1.1466.115.121.1.12
)
```

Le type d'attribut pcelsSequencedActions indique si l'exécution ordonnée d'actions dans un agrégat est Obligatoire, Recommandée ou SansImportance. Il est transposé de la propriété PolicyRule.SequencedActions [RFC3060] (identique à la propriété CompoundPolicyAction.SequencedActions définie dans la [RFC3460]). Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de integerMatch [RFC2252] et une règle de correspondance d'ordre de integerOrderingMatch [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont 1 (Obligatoire), 2 (Recommandée) et 3 (SansImportance). Si cet attribut manque dans une instance de pcelsRule, les applications DOIVENT supposer que l'exécution ordonnées des actions dans cette règle est SansImportance.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.9
  NOM 'pcelsSequencedActions'
  DESC 'Indique k'importance du séquençage de l'action '
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
  VALEUR UNIQUE
```

)

Le type d'attribut `pcelsExecutionStrategy` indique si les actions dans un agrégat sont à exécuter jusqu'au succès, toutes (indépendamment de leur résultat) ou jusqu'à échec. Il est transposé de la propriété `PolicyRule.ExecutionStrategy` [RFC3460] (identique à la propriété `CompoundPolicyAction.ExecutionStrategy`). Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de `integerMatch` [RFC2252] et une règle de correspondance d'ordre de `integerOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont 1 (Faire jusqu'au succès), 2 (Faire tout) et 3 (Faire jusqu'à échec). Si cet attribut manque dans une instance de `pcelsRule`, les applications DOIVENT supposer que toutes les actions sont à exécuter (Faire tout).

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.10
  NOM 'pcelsExecutionStrategy'
  DESC 'Indique la stratégie d'exécution de l'action'
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
  VALEUR UNIQUE
)
```

Note 1 : les périodes de validité pour une instance de `pcelsRule` sont réalisées en utilisant l'attribut `pcelsRuleValidityPeriodList` et les entrées de `pcimRuleValidityAssociation` [RFC3703] subordonnées à la règle. Si les règles de structure de DIT et les formes de noms sont écrites pour une mise en œuvre PCELS (comme suggéré au paragraphe 5.5 de la [RFC3703]) cela va exiger qu'une instance de la classe `pcimRuleValidityAssociation` ait comme supérieur une instance de la classe `pcelsRule` ou, si applicable, une instance de la classe `pcimRule`. Toutes les règles de structure et formes de noms qui exigent qu'une instance de la classe `pcimRuleValidityAssociation` n'ait son supérieur que dans une instance de la classe `pcimRule` sont en conflit et DOIVENT être supprimées.

Note 2 : les mises en œuvre de PCELS DEVRAIENT prendre en charge `pcelsRule` et ses deux sous classes et PEUVENT aussi prendre en charge `pcimRule` et ses deux sous classes [RFC3703]. Les applications qui choisissent de prendre en charge `pcelsRule` et ses deux sous classes DOIVENT utiliser le mécanisme d'agrégation fourni par `pcelsPolicySetAssociation` pour agréger des groupes de politiques ou des règles de politique dans des règles de politique représentées comme des instances de `pcelsRule`.

## 5.5 Classe structurelle `pcelsConditionAssociation`

La classe `pcelsConditionAssociation` est utilisée pour l'agrégation d'instances de `PolicyCondition` [RFC3060]. Les entrées de `pcelsConditionAssociation` sont toujours subordonnées à l'entrée agrégante. Lorsque elle est subordonnée à une instance de `pcelsRule`, l'entrée de `pcelsConditionAssociation` réalise l'association `PolicyConditionInPolicyRule` [RFC3460]. Lorsque elle est subordonnée à une instance de `pcelsCompoundConditionAuxClass`, l'entrée de `pcelsConditionAssociation` réalise l'association `PolicyConditionInPolicyCondition` [RFC3460].

La classe `pcelsConditionAssociation` est une classe d'objets structurelle et elle est déduite de la classe `pcimRuleConditionAssociation` [RFC3703].

L'agrégation d'une instance réutilisable de `pcimConditionAuxClass` est réalisée via l'attribut `pcimConditionDN`. Une instance non réutilisable de `pcimConditionAuxClass` est rattachée directement à l'entrée de `pcelsConditionAssociation`.

Lorsque on lit une instance d'entrée de `pcelsConditionAssociation` qui a une instance de `pcimConditionAuxClass` rattachée, l'attribut `pcimConditionDN` DOIT être ignoré. Les applications DEVRAIENT supprimer la valeur de `pcimConditionDN` d'une `pcelsConditionAssociation` lors du rattachement d'une `pcimConditionAuxClass` à l'entrée.

La classe `pcelsConditionAssociation` est définie comme suit :

```
( 1.3.6.1.1.9.1.9
  NOM 'pcelsConditionAssociation'
  DESC 'Associe une condition de politique à une entrée agrégante'
  SUP pcimRuleConditionAssociation
  STRUCTURELLE
)
```

Cette classe étend la sémantique de la classe d'objet `pcimRuleConditionAssociation` sans utiliser de nouvel attribut. Tous ses attributs sont hérités de la `pcimRuleConditionAssociation` définie au paragraphe 5.4 de la [RFC3703].

### 5.6 Classe structurelle `pcelsActionAssociation`

La classe `pcelsActionAssociation` est utilisée pour l'agrégation d'instances de `PolicyAction` [RFC3060]. Les entrées de `pcelsActionAssociation` sont toujours subordonnées à l'entrée agrégeante. Lorsque subordonnée à une instance de `pcelsRule`, l'entrée de `pcelsActionAssociation` réalise l'association de `PolicyActionInPolicyRule` [RFC3460]. Lorsque subordonnée à une instance de `pcelsCompoundActionAuxClass`, l'entrée de `pcelsActionAssociation` réalise l'association de `PolicyActionInPolicyAction` [RFC3460].

La classe `pcelsActionAssociation` est une classe d'objets structurelle et elle est déduite de la classe `pcimRuleActionAssociation` [RFC3703].

L'agrégation d'une instance réutilisable de `pcimActionAuxClass` est réalisée via l'attribut `pcimActionDN`. Une instance non réutilisable de `pcimActionAuxClass` est rattachée directement à l'entrée `pcelsActionAssociation`.

Lorsque on lit une instance d'entrée de `pcelsActionAssociation` qui a une instance de `pcimActionAuxClass` rattachée, l'attribut `pcimActionDN` DOIT être ignoré. Les applications DEVRAIENT supprimer la valeur de `pcimActionDN` d'une `pcelsActionAssociation` lors du rattachement d'une `pcimActionAuxClass` à l'entrée.

La classe `pcelsActionAssociation` est définie comme suit :

```
( 1.3.6.1.1.9.1.10
  NOM 'pcelsActionAssociation'
  DESC 'Associe une condition de politique à une entrée agrégeante'
  SUP pcimRuleActionAssociation
  STRUCTURELLE
)
```

Cette classe étend la sémantique de la classe d'objets `pcimRuleActionAssociation` sans utiliser de nouvel attribut. Tous ses attributs sont hérités de la `pcimRuleActionAssociation` définie au paragraphe 5.6 de la [RFC3703].

### 5.7 Classe auxiliaire `pcelsSimpleConditionAuxClass`

La classe `pcelsSimpleConditionAuxClass` met en œuvre une condition de correspondance de valeur pour une variable. Elle est transposée de la classe `SimplePolicyCondition` de la [RFC3460]. La classe `pcelsSimpleConditionAuxClass` est une classe d'objet auxiliaire et elle est déduite de la classe `pcimConditionAuxClass` [RFC3703].

Une variable/valeur réutilisable est associée à une `pcelsSimpleConditionAuxClass` via la référence `pcelsVariableDN/pcelsValueDN` d'une instance de la condition simple. Une variable/valeur non réutilisable est associée directement comme classe d'objet auxiliaire à la même entrée que l'instance de `pcelsSimpleConditionAuxClass`.

Lorsque on lit une instance de `pcelsSimpleConditionAuxClass` qui est une instance de `pcelsVariable` rattachée, l'attribut `pcelsVariableDN` DOIT être ignoré. Les applications DEVRAIENT supprimer la valeur de `pcelsVariableDN` d'une instance de `pcelsSimpleConditionAuxClass` lors du rattachement d'une instance de `pcelsVariable` à la même entrée.

Lorsque on lit une instance de `pcelsSimpleConditionAuxClass` qui a une instance de `pcelsValue` rattachée, l'attribut `pcelsValueDN` DOIT être ignoré. Les applications DEVRAIENT supprimer la valeur de `pcelsValueDN` d'une instance de `pcelsSimpleConditionAuxClass` lors du rattachement d'une instance de `pcelsValue` à la même entrée.

La classe `pcelsSimpleConditionAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.11
  NOM 'pcelsSimpleConditionAuxClass'
  DESC 'Condition de correspondance de valeur pour une variable de politique'
  SUP pcimConditionAuxClass
  AUXILIAIRE
  PEUT ( pcelsVariableDN $ pcelsValueDN )
)
```

Le type d'attribut `pcelsVariableDN` réalise l'association de `PolicyVariableInSimplePolicyCondition` [RFC3460]. Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de `distinguishedNameMatch` [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs `pcelsVariableDN` sont les DN des entrées de `pcelsVariable`. Dans une `pcelsSimpleConditionAuxClass`, l'attribut `pcelsVariableDN` représente l'association entre cette simple condition de politique et sa variable de politique.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.11
  NOM 'pcelsVariableDN'
  DESC 'DN d'une entrée de pcelsVariable'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.12
  VALEUR UNIQUE
)
```

Le type d'attribut `pcelsValueDN` réalise l'association de `PolicyValueInSimplePolicyCondition` [RFC3460]. Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de `distinguishedNameMatch` [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs `pcelsValueDN` sont les DN des entrées de `pcelsValueAuxClass`. Dans une `pcelsSimpleConditionAuxClass`, l'attribut `pcelsValueDN` représente l'association entre cette simple condition de politique et sa valeur de politique.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.12
  NOM 'pcelsValueDN'
  DESC 'DN d'une entrée de pcelsValueAuxClass'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.12
  VALEUR UNIQUE
)
```

Note : une instance de `pcelsSimpleActionAuxClass` et une instance de `pcelsSimpleConditionAuxClass` NE DOIVENT PAS être rattachées à la même entrée. Parce que les deux classes utilisent les mêmes mécanismes pour associer variables et valeurs, cette restriction est nécessaire pour éviter les ambiguïtés.

## 5.8 Classe auxiliaire `pcelsCompoundConditionAuxClass`

La classe `pcelsCompoundConditionAuxClass` représente une condition de politique composée formée par l'agrégation d'autres conditions de politique. Elle est transposée de la classe `CompoundPolicyCondition` [RFC3460]. La classe `pcelsCompoundConditionAuxClass` est une classe d'objets auxiliaire et elle est déduite de la classe `pcimConditionAuxClass` [RFC3703].

La classe `pcelsCompoundConditionAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.12
  NOM 'pcelsCompoundConditionAuxClass'
  DESC 'Combinaison booléenne de conditions plus simples'
  SUP pcimConditionAuxClass
  AUXILIAIRE
  PEUT ( pcelsConditionListType $ pcelsConditionList )
)
```

Si l'attribut `pcelsConditionListType` manque dans une instance de `pcelsCompoundConditionAuxClass`, les applications DOIVENT supposer que l'ensemble de conditions agrégées est une forme disjonctive normale.

Dans une instance de `pcelsCompoundConditionAuxClass`, l'attribut `pcelsConditionList` représente les associations entre cette condition de politique composée et les conditions composées.

Ces types d'attributs sont définis au paragraphe 5.4.

Comme `pcelsRule`, les instances de `pcelsCompoundConditionAuxClass` utilisent les valeurs de `pcelsConditionList` et les entrées subordonnées de `pcelsConditionAssociation` pour agréger les conditions de politique.

### 5.9 Classe auxiliaire `pcelsCompoundFilterConditionAuxClass`

La classe `pcelsCompoundFilterConditionAuxClass` représente un filtre de niveau domaine. Elle est transposée de la classe `CompoundFilterCondition` [RFC3460]. La classe `pcelsCompoundFilterConditionAuxClass` est une classe d'objets auxiliaire et elle est déduite de la classe `pcelsCompoundConditionAuxClass`.

La classe `pcelsCompoundFilterConditionAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.13
  NOM 'pcelsCompoundFilterConditionAuxClass'
  DESC 'Condition composée avec des capacités de miroir'
  SUP pcelsCompoundConditionAuxClass
  AUXILIAIRE
  PEUT ( pcelsIsMirrored )
)
```

Le type d'attribut `pcelsIsMirrored` indique si le trafic qui reflète le filtre spécifié est à traiter comme correspondant au filtre. Il est transposé de la propriété `CompoundFilterCondition.IsMirrored` [RFC3460]. Ce type d'attribut a la syntaxe Booléen [RFC2252]. Il a une règle de correspondance pour égalité de `booleanMatch` [RFC3698].

Les attributs de ce type ne peuvent avoir qu'une seule valeur. Si cet attribut manque dans une instance de `pcelsCompoundFilterConditionAuxClass`, les applications DOIVENT supposer que le filtre n'est pas reflété.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.13
  NOM 'pcelsIsMirrored'
  DESC 'Indique si le trafic reflété correspond'
  ÉGALITÉ booleanMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.7
  VALEUR UNIQUE
)
```

### 5.10 Classe auxiliaire `pcelsSimpleActionAuxClass`

La classe `pcelsSimpleActionAuxClass` met en œuvre l'action d'allouer une valeur à une variable. Elle est transposée de la classe `SimplePolicyAction` [RFC3460]. La classe `pcelsSimpleActionAuxClass` est une classe d'objets auxiliaire et elle est déduite de la classe `pcimActionAuxClass` [RFC3703].

Une variable/valeur réutilisable est associée à une `pcelsSimpleActionAuxClass` via la référence `pcelsVariableDN/pcelsValueDN` provenant de l'instance d'action simple. Une variable/valeur non réutilisable est associée directement comme classe d'objets auxiliaire à la même entrée que l'instance de `pcelsSimpleActionAuxClass`.

Lorsque on lit une instance de `pcelsSimpleActionAuxClass` qui a une instance de `pcelsVariable` rattachée, l'attribut `pcelsVariableDN` DOIT être ignoré. Les applications DEVRAIENT supprimer la valeur de `pcelsVariableDN` provenant d'une instance de `pcelsSimpleActionAuxClass` au moment du rattachement d'une instance de `pcelsVariable` à la même entrée.

Lorsque on lit une instance de `pcelsSimpleActionAuxClass` qui a une instance de `pcelsValue` rattachée, l'attribut `pcelsValueDN` DOIT être ignoré. Les applications DEVRAIENT supprimer la valeur de `pcelsValueDN` d'une instance de `pcelsSimpleActionAuxClass` instance au moment du rattachement d'une instance de `pcelsValue` à la même entrée.

La classe `pcelsSimpleActionAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.14
  NOM 'pcelsSimpleActionAuxClass'
  DESC 'Action d'allocation d'une valeur à une variable de politique'
  SUP pcimActionAuxClass
)
```

```
AUXILIAIRE
PEUT ( pcelsVariableDN $ pcelsValueDN )
)
```

Dans une `pcelsSimpleActionAuxClass`, l'attribut `pcelsVariableDN` représente l'association entre cette action de politique simple et sa variable de politique. Il réalise l'association de `PolicyVariableInSimplePolicyAction` [RFC3460].

Dans une `pcelsSimpleActionAuxClass`, l'attribut `pcelsValueDN` représente l'association entre cette action de politique simple et sa valeur de politique. Il réalise l'association de `PolicyValueInSimplePolicyAction` [RFC3460].

Ces attributs sont définis au paragraphe 5.7.

Note : une instance de `pcelsSimpleActionAuxClass` et une instance de `pcelsSimpleConditionAuxClass` NE DOIVENT PAS être rattachées à la même entrée. Parce que les deux classes utilisent les mêmes mécanismes pour associer variables et valeurs, cette restriction est nécessaire afin d'éviter les ambiguïtés.

### 5.11 Classe auxiliaire `pcelsCompoundActionAuxClass`

La classe `pcelsCompoundActionAuxClass` représente une action de politique composée formée par l'agrégation d'autres actions de politique. Elle est transposée de la classe `CompoundPolicyCondition` [RFC3460]. La classe `pcelsCompoundActionAuxClass` est une classe d'objets auxiliaire et elle est déduite de la classe `pcimActionAuxClass` [RFC3703].

La classe `pcelsCompoundActionAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.15
  NOM 'pcelsCompoundActionAuxClass'
  DESC 'Séquence d'actions avec une stratégie d'exécution spécifique'
  SUP pcimActionAuxClass
  AUXILIAIRE
  PEUT ( pcelsActionList $ pcelsSequencedActions $ pcelsExecutionStrategy )
)
```

Dans une instance de `pcelsCompoundActionAuxClass`, l'attribut `pcelsActionList` représente les associations entre cette règle de politique et ses actions.

Si l'attribut `pcelsSequencedActions` manque dans une instance de `pcelsCompoundActionAuxClass`, les applications DOIVENT supposer que l'exécution ordonnée des actions dans cette action de politique composée n'est pas importante (SansImportance).

Si l'attribut `pcelsExecutionStrategy` manque dans une instance de `pcelsCompoundActionAuxClass`, les applications DOIVENT supposer que toutes les actions sont à exécuter (Faire tout).

Ces types d'attributs sont définis au paragraphe 5.4.

Comme `pcelsRule`, les instances de `pcelsCompoundActionAuxClass` utilisent les valeurs de `pcelsActionList` et les entrées subordonnées de `pcelsActionAssociation` pour agréger les actions de politique.

### 5.12 Classe abstraite `pcelsVariable`

La classe `pcelsVariable` est transposée de la classe `PolicyVariable` [RFC3460]. La classe `pcelsVariable` est une classe d'objets abstraite et elle est dérivée directement de la classe d'objet "sommet" [RFC2256].

Une instance de `pcelsVariable` peut être associée à un ensemble d'instances de `pcelsValueAuxClass` qui représentent ses valeurs attendues. Les valeurs attendues pour une variable peuvent être indiquées par :

- (1) des références de `pcelsExpectedValueList` à des instances réutilisables de `pcelsValueAuxClass`, ou
- (2) des références de `pcelsExpectedValueList` à des instances subordonnées non réutilisables de `pcelsValueAuxClass`

La classe `pcelsVariable` est définie comme suit :

```
( 1.3.6.1.1.9.1.16
```

```

NOM 'pcelsVariable'
DESC 'Classe de base pour représenter une variable de politique'
SUP sommet
ABSTRAITE
PEUT ( pcelsVariableName $ pcelsExpectedValueList )
)

```

Le type d'attribut `pcelsVariableName` peut être utilisé comme attribut de dénomination pour des entrées de `pcelsVariable`. Ce type d'attribut a la syntaxe Chaîne de répertoire (*Directory String*) [RFC2252]. Il a une règle de correspondance pour égalité de `caseIgnoreMatch`, une règle de correspondance d'ordre de `caseIgnoreOrderingMatch` et une règle de correspondance de sous chaîne de `caseIgnoreSubstringsMatch` [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type attribut est défini comme suit :

```

( 1.3.6.1.1.9.2.14
  NOM 'pcelsVariableName'
  DESC 'Nom facile à mémoriser d'une variable.'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
  VALEUR UNIQUE
)

```

Le type d'attribut `pcelsExpectedValueList` réalise l'association `ExpectedPolicyValuesForVariable` [RFC3460]. Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de `distinguishedNameMatch` [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs `pcelsExpectedValueList` sont des DN d'entrées de `pcelsValueAuxClass`. Dans une `pcelsVariable`, l'attribut `pcelsExpectedValueList` représente les associations entre cette variable de politique et ses valeurs attendues.

Ce type attribut est défini comme suit :

```

( 1.3.6.1.1.9.2.15
  NOM 'pcelsExpectedValueList'
  DESC 'Ensemble non ordonné de DN d'entrées de pcelsValueAuxClass représentant les valeur attendues pour une variable de politique'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.12
)

```

### 5.13 Classe auxiliaire `pcelsExplicitVariableAuxClass`

La classe `pcelsExplicitVariableAuxClass` est transposée de la classe `PolicyExplicitVariable` [RFC3460]. La classe `pcelsExplicitVariableAuxClass` est une classe d'objets auxiliaire et elle est déduite de la classe `pcelsVariable`.

La classe `pcelsExplicitVariableAuxClass` est définie comme suit :

```

( 1.3.6.1.1.9.1.17
  NOM 'pcelsExplicitVariableAuxClass'
  DESC 'Variable de politique explicitement définie'
  SUP pcelsVariable
  AUXILIAIRE
  DOIT ( pcelsVariableModelClass $ pcelsVariableModelProperty )
)

```

Le type d'attribut `pcelsVariableModelClass` identifie [CIM] une classe dont la propriété est évaluée ou réglée comme une variable. Il est transposé de la propriété `PolicyExplicitVariable.ModelClass` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de `caseIgnoreMatch` [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.16
  NOM 'pcelsVariableModelClass'
  DESC 'Identifie une classe CIM'
  ÉGALITÉ caseIgnoreMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
  VALEUR UNIQUE
)
```

Le type d'attribut `pcelsVariableModelProperty` identifie l'attribut d'une classe [CIM], qui est évaluée ou réglée comme une variable. Il est transposé de la propriété `PolicyExplicitVariable.ModelProperty` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de `caseIgnoreMatch` [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type d'attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.17
  NOM 'pcelsVariableModelProperty'
  DESC 'Identifie la propriété d'une classe CIM'
  ÉGALITÉ caseIgnoreMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
  VALEUR UNIQUE
)
```

#### 5.14 Classe auxiliaire `pcelsImplicitVariableAuxClass`

La classe `pcelsImplicitVariableAuxClass` est transposée de la classe `PolicyImplicitVariable` [RFC3460]. La classe `pcelsImplicitVariableAuxClass` est une classe d'objets auxiliaire et elle est déduite de la classe `pcelsVariable`.

La classe `pcelsImplicitVariableAuxClass` ne représente pas de variables réelles ; celles-ci sont introduites par ses sous classes. `pcelsImplicitVariableAuxClass` introduit la sémantique d'être une variable de politique définie implicitement et cette sémantique est héritée par toutes ses sous classes. Cette sémantique inclut celle héritée de `pcelsVariable` qui représente éventuellement des variables de politique spécifiques de règle ou réutilisables.

Afin de préserver la capacité de représenter des variables spécifiques de règle ou réutilisables, toutes les sous classes de `pcelsImplicitVariableAuxClass` DOIVENT aussi être des classes auxiliaires.

La classe `pcelsImplicitVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.18
  NOM 'pcelsImplicitVariableAuxClass'
  DESC 'Variable de politique implicitement définie'
  SUP pcelsVariable
  AUXILIAIRE
  PEUT ( pcelsExpectedValueTypes )
)
```

Le type d'attribut `pcelsExpectedValueTypes` représente l'ensemble des types de valeurs de politiques qui peuvent être utilisés avec cette variable de politique. Il est transposé de la propriété `PolicyImplicitVariable.ValueTypes` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de `caseIgnoreMatch` [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.18
  NOM 'pcelsExpectedValueTypes'
  DESC 'Identifie par nom les sous classes de pcelsValueAuxClass'
  ÉGALITÉ caseIgnoreMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
)
```

### 5.15 Sous classes de `pcelsImplicitVariableAuxClass`

Les classes suivantes sont dérivées de la classe `pcelsImplicitVariableAuxClass`. Elles sont transposées des sous classes correspondantes de la classe `PolicyImplicitVariable` [RFC3460]. Toutes les classes définies ci-dessous sont des classes d'objets auxiliaires.

Chacune des classes définies dans ce paragraphe introduit des restrictions spécifiques aux valeurs des attributs de `pcelsExpectedValueTypes`. Si cet attribut manque, les applications DOIVENT supposer que tous les types de valeurs permises sont attendus pour la variable de politique.

Certaines des classes ont des restrictions supplémentaires sur les valeur réelles des instances de valeur de politique associée (par exemple, seulement des entiers dans la gamme 0 à 65 535 doivent être utilisés avec une variable `SourcePort`). L'association entre une instance de `pcelsImplicitVariableAuxClass` et une instance de `pcelsValueAuxClass` qui contient des valeurs hors de la gamme ou ensemble valide pour cette variable DEVRAIT être considérée comme invalide. L'entrée qui réalise une telle association DEVRAIT être traitée comme invalide et les règles ou groupes de politiques qui se réfèrent à elle DEVRAIENT être traités comme désactivés, ce qui signifie que l'exécution de telles règles ou groupes de politiques DEVRAIT être arrêtée.

La classe `pcelsSourceIPv4VariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.19
  NOM 'pcelsSourceIPv4VariableAuxClass'
  DESC 'Adresse IPv4 de source'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsSourceIPv4VariableAuxClass`, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsIPv4AddrValueAuxClass`'.

La classe `pcelsSourceIPv6VariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.20
  NOM 'pcelsSourceIPv6VariableAuxClass'
  DESC 'Adresse IPv6 de source'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsSourceIPv6VariableAuxClass` instance, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsIPv6AddrValueAuxClass`'.

La classe `pcelsDestinationIPv4VariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.21
  NOM 'pcelsDestinationIPv4VariableAuxClass'
  DESC 'Adresse IPv4 de destination'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsDestinationIPv4VariableAuxClass`, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsIPv4AddrValueAuxClass`'.

La classe `pcelsDestinationIPv6VariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.22
  NOM 'pcelsDestinationIPv6VariableAuxClass'
  DESC 'Adresse IPv6 de destination'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsDestinationIPv6VariableAuxClass`, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsIPv6AddrValueAuxClass`'.

La classe `pcelsSourcePortVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.23
  NOM 'pcelsSourcePortVariableAuxClass'
  DESC 'Accès de source'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsSourcePortVariableAuxClass`, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsIntegerValueAuxClass`'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0..65535 (inclus) DEVRAIENT être utilisées avec les instances de `pcelsSourcePortVariableAuxClass`.

La classe `pcelsDestinationPortVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.24
  NOM 'pcelsDestinationPortVariableAuxClass'
  DESC 'Accès de destination'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsDestinationPortVariableAuxClass` instance, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsIntegerValueAuxClass`'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 65 535 (inclus) DEVRAIENT être utilisées avec les instances de `pcelsDestinationPortVariableAuxClass`.

La classe `pcelsIPProtocolVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.25
  NOM 'pcelsIPProtocolVariableAuxClass'
  DESC 'Numéro de protocole IP'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsIPProtocolVariableAuxClass`, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsIntegerValueAuxClass`'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 255 (inclus) DEVRAIENT être utilisées avec des instances de `pcelsIPProtocolVariableAuxClass`.

La classe `pcelsIPVersionVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.26
  NOM 'pcelsIPVersionVariableAuxClass'
  DESC 'Numéro de version IP'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsIPVersionVariableAuxClass`, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsIntegerValueAuxClass`'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 15 (inclus) DEVRAIENT être utilisées avec des instances de `pcelsIPVersionVariableAuxClass`.

La classe `pcelsIPToSVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.27
  NOM 'pcelsIPToSVariableAuxClass'
  DESC 'octet ToS IP'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsIPToSVariableAuxClass`, les seules valeurs admises pour l'attribut `pcelsExpectedValueTypes` sont '`pcelsIntegerValueAuxClass`' et '`pcelsBitStringValueAuxClass`'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 255 (inclus) ou des chaînes binaires de 8 bits DEVRAIENT être utilisées avec des instances de `pcelsIPToSVariableAuxClass`.

La classe `pcelsDSCPVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.28
  NOM 'pcelsDSCPVariableAuxClass'
  DESC 'Codet DiffServ'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsDSCPVariableAuxClass`, les seules valeurs admises pour l'attribut `pcelsExpectedValueTypes` sont '`pcelsIntegerValueAuxClass`' et '`pcelsBitStringValueAuxClass`'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 63 (inclus) ou des chaînes binaires de 6 bits DEVRAIENT être utilisées avec des instances de `pcelsDSCPVariableAuxClass`.

La classe `pcelsFlowIdVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.29
  NOM 'pcelsFlowIdVariableAuxClass'
  DESC 'Identifiant de flux'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsFlowIdVariableAuxClass`, les seules valeurs admises pour l'attribut `pcelsExpectedValueTypes` sont '`pcelsIntegerValueAuxClass`' et '`pcelsBitStringValueAuxClass`'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 1 048 575 (inclus) ou des chaînes binaires de 20 bits DEVRAIENT être utilisées avec des instances de `pcelsFlowIdVariableAuxClass`.

La classe `pcelsSourceMACVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.30
  NOM 'pcelsSourceMACVariableAuxClass'
  DESC 'Adresse MAC de source'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsSourceMACVariableAuxClass`, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsMACAddrValueAuxClass`'.

La classe `pcelsDestinationMACVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.31
  NOM 'pcelsDestinationMACVariableAuxClass'
  DESC 'Adresse MAC de destination'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsDestinationMACVariableAuxClass`, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsMACAddrValueAuxClass`'.

La classe `pcelsVLANVariableAuxClass` class définie comme suit :

```
( 1.3.6.1.1.9.1.32
  NOM 'pcelsVLANVariableAuxClass'
  DESC 'VLAN'
```

```

SUP pcelsImplicitVariableAuxClass
AUXILIAIRE
)

```

Dans une instance de pcelsVLANVariableAuxClass, les seules valeurs admises pour l'attribut pcelsExpectedValueTypes sont 'pcelsIntegerValueAuxClass' et 'pcelsBitStringValueAuxClass'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 4095 (inclus) ou des chaînes binaires de 12 bits DEVRAIENT être utilisées avec des instances de pcelsVLANVariableAuxClass.

La classe pcelsCoSVariableAuxClass est définie comme suit :

```

( 1.3.6.1.1.9.1.33
  NOM 'pcelsCoSVariableAuxClass'
  DESC 'Classe de service'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)

```

Dans une instance de pcelsCoSVariableAuxClass, les seules valeurs admises pour l'attribut pcelsExpectedValueTypes sont 'pcelsIntegerValueAuxClass' et 'pcelsBitStringValueAuxClass'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 7 (inclus) ou des chaînes binaires de 3 bits DEVRAIENT être utilisées avec des instances de pcelsCoSVariableAuxClass.

La classe pcelsEthertypeVariableAuxClass est définie comme suit :

```

( 1.3.6.1.1.9.1.34
  NOM 'pcelsEthertypeVariableAuxClass'
  DESC 'Ethertype'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)

```

Dans une instance de pcelsEthertypeVariableAuxClass, les seules valeurs admises pour l'attribut pcelsExpectedValueTypes sont 'pcelsIntegerValueAuxClass' et 'pcelsBitStringValueAuxClass'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 65 535 (inclus) ou des chaînes binaires de 16 bits DEVRAIENT être utilisées avec des instances de pcelsEthertypeVariableAuxClass.

La classe pcelsSourceSAPVariableAuxClass est définie comme suit :

```

( 1.3.6.1.1.9.1.35
  NOM 'pcelsSourceSAPVariableAuxClass'
  DESC 'SAP de source'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)

```

Dans une instance de pcelsSourceSAPVariableAuxClass, les seules valeurs admises pour l'attribut pcelsExpectedValueTypes sont 'pcelsIntegerValueAuxClass' et 'pcelsBitStringValueAuxClass'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 255 (inclus) ou des chaînes binaires de 8 bits DEVRAIENT être utilisées avec des instances de pcelsSourceSAPVariableAuxClass.

La classe pcelsDestinationSAPVariableAuxClass est définie comme suit :

```

( 1.3.6.1.1.9.1.36
  NOM 'pcelsDestinationSAPVariableAuxClass'
  DESC 'SAP de destination'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)

```

Dans une instance de pcelsDestinationSAPVariableAuxClass, les seules valeurs admises pour l'attribut pcelsExpectedValueTypes sont 'pcelsIntegerValueAuxClass' et 'pcelsBitStringValueAuxClass'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 255 (inclus) ou des chaînes binaires de 8 bits DEVRAIENT être

utilisées avec des instances de `pcelsDestinationSAPVariableAuxClass`.

La classe `pcelsSNAPOUIVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.37
  NOM 'pcelsSNAPOUIVariableAuxClass'
  DESC 'OUI SNAP'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsSNAPOUIVariableAuxClass`, les seules valeurs admises pour l'attribut `pcelsExpectedValueTypes` sont '`pcelsIntegerValueAuxClass`' et '`pcelsBitStringValueAuxClass`'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 16 777 215 (inclus) ou des chaînes binaires de 24 bits DEVRAIENT être utilisées avec des instances de `pcelsSNAPOUIVariableAuxClass`.

La classe `pcelsSNAPTypeVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.38
  NOM 'pcelsSNAPTypeVariableAuxClass'
  DESC 'Type SNAP'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsSNAPTypeVariableAuxClass`, les seules valeurs admises pour l'attribut `pcelsExpectedValueTypes` sont '`pcelsIntegerValueAuxClass`' et '`pcelsBitStringValueAuxClass`'. De plus, seules les valeurs de politique qui représentent des entiers dans la gamme 0 à 65 535 (inclus) ou des chaînes binaires de 16 bits DEVRAIENT être utilisées avec des instances de `pcelsSNAPTypeVariableAuxClass`.

La classe `pcelsFlowDirectionVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.39
  NOM 'pcelsFlowDirectionVariableAuxClass'
  DESC 'Direction du flux'
  SUP pcelsImplicitVariableAuxClass
  AUXILIAIRE
)
```

Dans une instance de `pcelsFlowDirectionVariableAuxClass`, la seule valeur admise pour l'attribut `pcelsExpectedValueTypes` est '`pcelsStringValueAuxClass`'. De plus, seules les valeurs de politique qui représentent les chaînes 'IN' et 'OUT' DEVRAIENT être utilisées avec des instances de `pcelsFlowDirectionVariableAuxClass`.

## 5.16 Classe auxiliaire `pcelsValueAuxClass`

La classe `pcelsValueAuxClass` est la classe de base pour représenter une valeur de politique. Elle est transposée de la classe `PolicyValue` [RFC3460]. `pcelsValueAuxClass` est une classe d'objets auxiliaire et elle est directement dérivée de la classe d'objets "sommet" [RFC2256].

La classe `pcelsValueAuxClass` ne représente pas des valeurs réelles ; celles-ci sont introduites par ses sous classes. `pcelsValueAuxClass` introduit la sémantique d'être une valeur de politique qui est héritée par toute ses sous classes. Parmi ces sémantiques se trouvent celles de représenter des valeurs de politique spécifiques de règle ou réutilisables.

Afin de préserver la capacité de représenter des valeurs spécifiques de règle ou réutilisables, toutes les sous classes de `pcelsValueAuxClass` DOIVENT aussi être des classes auxiliaires.

La classe `pcelsValueAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.40
  NOM 'pcelsValueAuxClass'
  DESC 'Classe de base pour représenter une valeur de politique'
  SUP top
```

```

AUXILIAIRE
PEUT ( pcelsValueName )
)

```

Le type d'attribut `pcelsValueName` peut être utilisé comme attribut de désignation pour les entrées de `pcelsValueAuxClass`. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de `caseIgnoreMatch`, une règle de correspondance d'ordre de `caseIgnoreOrderingMatch` et une règle de correspondance de sous chaîne de `caseIgnoreSubstringsMatch` [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type attribut est défini comme suit :

```

( 1.3.6.1.1.9.2.19
  NOM 'pcelsValueName'
  DESC 'Nom facile à retenir d'une valeur'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
  VALEUR UNIQUE
)

```

### 5.17 Sous classes de `pcelsValueAuxClass`

Les classes suivantes sont dérivées de la classe `pcelsValueAuxClass`. Elles sont transposées des sous classes correspondantes de la classe `PolicyValue` [RFC3460]. Toutes les classes définies ci-dessous sont des classes d'objets auxiliaires.

La classe `pcelsIPv4AddrValueAuxClass` représente une valeur de politique qui fournit un ensemble non ordonné d'adresses IPv4, de gammes d'adresses IPv4 ou d'hôtes. Elle est transposée de la classe `PolicyIPv4AddrValue` [RFC3460].

La classe `pcelsIPv4AddrValueAuxClass` est définie comme suit :

```

( 1.3.6.1.1.9.1.41
  NOM 'pcelsIPv4AddrValueAuxClass'
  DESC 'Fournit des adresses IPv4'
  SUP pcelsValueAuxClass
  AUXILIAIRE
  DOIT ( pcelsIPv4AddrList )
)

```

Le type d'attribut `pcelsIPv4AddrList` représente un ensemble non ordonné d'adresses IPv4, de gammes d'adresses IPv4 ou d'hôtes. Il est transposé de la propriété `PolicyIPv4AddrValue.IPv4AddrList` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de `caseIgnoreMatch`, une règle de correspondance d'ordre de `caseIgnoreOrderingMatch` et une règle de correspondance de sous chaînes de `caseIgnoreSubstringsMatch` [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs de ce type sont des chaînes qui se conforment à un des formats définis pour la propriété `IPv4AddrList` [RFC3460].

Ce type attribut est défini comme suit :

```

( 1.3.6.1.1.9.2.20
  NOM 'pcelsIPv4AddrList'
  DESC 'Ensemble non ordonné d'adresses IPv4, de gammes d'adresses IPv4 ou d'hôtes'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
)

```

La classe `pcelsIPv6AddrValueAuxClass` représente une valeur de politique qui fournit un ensemble non ordonné d'adresses IPv6, de gammes d'adresses IPv6 ou d'hôtes. Elle est transposée de la classe `PolicyIPv6AddrValue` [RFC3460].

La classe pcelsIPv6AddrValueAuxClass est définie comme suit :

```
( 1.3.6.1.1.9.1.42
  NOM 'pcelsIPv6AddrValueAuxClass'
  DESC 'Fournit des adresses IPv6'
  SUP pcelsValueAuxClass
  AUXILIAIRE
  DOIT ( pcelsIPv6AddrList )
)
```

Le type d'attribut pcelsIPv6AddrList représente un ensemble non ordonné d'adresses IPv6, de gammes d'adresses ou d'hôtes IPv6. Il est transposé de la propriété PolicyIPv6AddrValue.IPv6AddrList [RFC3460]. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de caseIgnoreMatch, une règle de correspondance d'ordre de caseIgnoreOrderingMatch et une règle de correspondance de sous chaînes de caseIgnoreSubstringsMatch [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs de ce type sont des chaînes qui se conforment à un des formats définis pour la propriété IPv6AddrList [RFC3460].

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.21
  NOM 'pcelsIPv6AddrList'
  DESC 'Ensemble non ordonné d'adresses IPv6, de gammes d'adresses ou hôtes IPv6'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
)
```

La classe pcelsMACAddrValueAuxClass représente une valeur de politique qui fournit un ensemble non ordonné d'adresses MAC ou de gammes d'adresses MAC. Elle est transposée de la classe PolicyMACAddrValue [RFC3460].

La classe pcelsMACAddrValueAuxClass est définie comme suit :

```
( 1.3.6.1.1.9.1.43
  NOM 'pcelsMACAddrValueAuxClass'
  DESC 'Fournit des adresses MAC'
  SUP pcelsValueAuxClass
  AUXILIAIRE
  DOIT ( pcelsMACAddrList )
)
```

Le type d'attribut pcelsMACAddrList représente un ensemble non ordonné d'adresses MAC ou de gammes d'adresses MAC. Il est transposé de la propriété PolicyMACAddrValue.MACAddrList [RFC3460]. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de caseIgnoreMatch, une règle de correspondance d'ordre de caseIgnoreOrderingMatch et une règle de correspondance de sous chaînes de caseIgnoreSubstringsMatch [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs de ce type sont des chaînes qui se conforment à un des formats définis pour la propriété MACAddrList [RFC3460].

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.22
  NOM 'pcelsMACAddrList'
  DESC 'Ensemble non ordonné d'adresses MAC ou de gammes d'adresses MAC'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
)
```

La classe pcelsStringValueAuxClass représente une valeur de politique qui fournit un ensemble non ordonné de chaînes avec des caractères génériques. Elle est transposée de la classe PolicyStringValue [RFC3460].

La classe `pcelsStringValueAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.44
  NOM 'pcelsStringValueAuxClass'
  DESC 'Fournit des chaînes de valeurs'
  SUP pcelsValueAuxClass
  AUXILIAIRE
  DOIT ( pcelsStringList )
)
```

Le type d'attribut `pcelsStringList` représente un ensemble non ordonné de chaînes avec des caractères génériques. Il est transposé de la propriété `PolicyStringValue.StringList` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de `caseIgnoreMatch`, une règle de correspondance d'ordre de `caseIgnoreOrderingMatch` et une règle de correspondance de sous chaînes de `caseIgnoreSubstringsMatch` [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs de ce type sont des chaînes qui se conforment au format défini pour la propriété `StringList` [RFC3460].

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.23
  NOM 'pcelsStringList'
  DESC 'Ensemble non ordonné de chaînes avec des caractères génériques'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
)
```

La classe `pcelsBitStringValueAuxClass` représente une valeur de politique qui fournit un ensemble non ordonné de chaînes binaires ou de gammes de chaînes binaires. Il est transposé de la classe `PolicyBitStringValue` [RFC3460].

La classe `pcelsBitStringValueAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.45
  NOM 'pcelsBitStringValueAuxClass'
  DESC 'Fournit des chaînes binaires'
  SUP pcelsValueAuxClass
  AUXILIAIRE
  DOIT ( pcelsBitStringList )
)
```

Le type d'attribut `pcelsBitStringList` représente un ensemble non ordonné de chaînes binaires ou de gammes de chaînes binaires. Il est transposé de la propriété `PolicyBitStringValue.BitStringList` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de `caseIgnoreMatch`, une règle de correspondance d'ordre de `caseIgnoreOrderingMatch` et une règle de correspondance de sous chaînes de `caseIgnoreSubstringsMatch` [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour ce type d'attribut sont des chaînes se conformant à un des formats définis pour la propriété `BitStringList` [RFC3460].

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.24
  NOM 'pcelsBitStringList'
  DESC 'Ensemble non ordonné de chaînes binaires ou de gammes de chaînes binaires'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
)
```

La classe `pcelsIntegerValueAuxClass` représente une valeur de politique qui fournit un ensemble non ordonné d'entiers ou de gammes d'entiers. Il est transposé de la classe `PolicyIntegerValue` [RFC3460].

La classe `pcelsIntegerValueAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.46
  NOM 'pcelsIntegerValueAuxClass'
  DESC 'Fournit des valeurs d'entiers'
  SUP pcelsValueAuxClass
  AUXILIAIRE
  DOIT ( pcelsIntegerList )
)
```

Le type d'attribut `pcelsIntegerList` représente un ensemble non ordonné d'entiers ou de gammes d'entiers. Il est transposé de la propriété `PolicyIntegerValue.IntegerList` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de `caseIgnoreMatch`, une règle de correspondance d'ordre de `caseIgnoreOrderingMatch` et une règle de correspondance de sous chaînes de `caseIgnoreSubstringsMatch` [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs de ce type sont des chaînes qui se conforment au format défini pour la propriété `IntegerList` [RFC3460].

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.25
  NOM 'pcelsIntegerList'
  DESC 'Ensemble non ordonné d'entiers ou de gammes d'entiers'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
)
```

La classe `pcelsBooleanValueAuxClass` représente une valeur de politique qui fournit un booléen. Elle est transposée de la classe `PolicyIntegerValue` [RFC3460].

La classe `pcelsBooleanValueAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.47
  NOM 'pcelsBooleanValueAuxClass'
  DESC 'Fournit une valeur booléenne'
  SUP pcelsValueAuxClass
  AUXILIAIRE
  DOIT ( pcelsBoolean )
)
```

Le type d'attribut `pcelsBoolean` représente un booléen. Il est transposé de la propriété `PolicyBooleanValue.BooleanValue` [RFC3460]. Ce type d'attribut a la syntaxe `Boolean` [RFC2252]. Il a une règle de correspondance pour égalité de `booleanMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.26
  NOM 'pcelsBoolean'
  DESC 'Valeur booléenne'
  ÉGALITÉ booleanMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.7
  VALEUR UNIQUE
)
```

## 5.18 Les trois classes de conteneur de politique réutilisable

La classe `pcelsReusableContainer` représente un conteneur d'éléments de politique réutilisables. Elle est transposée de la classe `ReusablePolicyContainer` [RFC3460]. La classe `pcelsReusableContainer` est dérivée de la classe `pcimRepository` [RFC3703]. Pour maximiser la souplesse, la classe `pcelsReusableContainer` est définie comme abstraite. Une sous classe auxiliaire `pcelsReusableContainerAuxClass` permet le rattachement d'un conteneur de politique réutilisable à une entrée existante, tandis qu'une sous classe structurelle `pcelsReusableContainerInstance` permet la représentation d'un conteneur de

politique réutilisable comme entrée autonome.

Les éléments contenus dans un conteneur de politique réutilisable sont agrégés via la subordination à une instance de `pcelsReusableContainer` (contenue dans la DIT). Un conteneur de politique réutilisable peut inclure les éléments d'un autre conteneur de politique réutilisable en agrégeant le conteneur lui-même. Ceci est réalisé par l'inclusion dans la DIT lorsque les conteneurs de politique sont subordonnés l'un à l'autre, ou par référence quand le conteneur de politique agrégateur fait référence à celui agrégé en utilisant l'attribut `pcelsReusableContainerList`.

La classe `pcelsReusableContainer` est définie comme suit :

```
( 1.3.6.1.1.9.1.48
  NOM 'pcelsReusableContainer'
  DESC 'Conteneur pour informations de politique réutilisables'
  SUP pcimRepository
  ABSTRAITE
  PEUT ( pcelsReusableContainerName $ pcelsReusableContainerList )
)
```

La classe `pcelsReusableContainerAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.49
  NOM 'pcelsReusableContainerAuxClass '
  DESC 'Conteneur pour informations de politique réutilisables'
  SUP pcelsReusableContainer
  AUXILIAIRE
)
```

La classe `pcelsReusableContainerInstance` est définie comme suit :

```
( 1.3.6.1.1.9.1.50
  NOM 'pcelsReusableContainerInstance'
  DESC 'Conteneur pour informations de politique réutilisables'
  SUP pcelsReusableContainer
  STRUCTURELLE
)
```

Le type d'attribut `pcelsReusableContainerName` peut être utilisé comme attribut de désignation pour des entrées de `pcelsReusableContainer`. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de `caseIgnoreMatch`, une règle de correspondance d'ordre de `caseIgnoreOrderingMatch` et une règle de correspondance de sous chaînes de `caseIgnoreSubstringsMatch` [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.27
  NOM 'pcelsReusableContainerName'
  DESC 'Nom facile à mémoriser d'un conteneur de politique réutilisable'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
  VALEUR UNIQUE
)
```

Le type d'attribut `pcelsReusableContainerList` réalise l'association `ContainedDomain` [RFC3460]. Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de `distinguishedNameMatch` [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs `pcelsReusableContainerList` sont les DN des entrées de `pcelsReusableContainer`. Dans un `pcelsReusableContainer`, l'attribut `pcelsReusableContainerList` représente les associations entre ce conteneur de politique réutilisable et les autres afin de les inclure comme conteneurs incorporés.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.28
  NOM 'pcelsReusableContainerList'
  DESC 'Ensemble non ordonné de DN des entrées de pcelsReusableContainer'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.12
)
```

Note : les mises en œuvre de PCELS DEVRAIENT prendre en charge pcelsReusableContainer et ses deux sous classes et PEUVENT aussi prendre en charge les deux sous classes de pcimRepository [RFC3703].

### 5.19 Classe structurelle pcelsRoleCollection

La classe pcelsRoleCollection représente une collection d'éléments gérés qui partagent un rôle commun. Elle est transposée de la classe PolicyRoleCollection [RFC3460]. La classe pcelsRoleCollection est une classe d'objets structurelle et elle est déduite de la classe pcimPolicy [RFC3703].

La classe pcelsRoleCollection est définie comme suit :

```
( 1.3.6.1.1.9.1.51
  NOM 'pcelsRoleCollection'
  DESC 'Collection d'éléments gérés qui partagent un rôle commun'
  SUP pcimPolicy
  STRUCTURELLE
  DOIT ( pcelsRole )
  PEUT ( pcelsRoleCollectionName $ pcelsElementList )
)
```

Le type d'attribut pcelsRole représente le rôle associé à une collection d'éléments gérés. Il est transposé de la propriété PolicyRoleCollection.PolicyRole [RFC3460]. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de caseIgnoreMatch, une règle de correspondance d'ordre de caseIgnoreOrderingMatch et une règle de correspondance de sous chaînes de caseIgnoreSubstringsMatch [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.29
  NOM 'pcelsRole'
  DESC 'Chaîne représentant un rôle.'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
  VALEUR UNIQUE
)
```

Le type d'attribut pcelsRoleCollectionName peut être utilisé comme attribut de désignation pour les entrées de pcelsRoleCollection. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de caseIgnoreMatch, une règle de correspondance d'ordre de caseIgnoreOrderingMatch et une règle de correspondance de sous chaînes de caseIgnoreSubstringsMatch [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.30
  NOM 'pcelsRoleCollectionName'
  DESC 'Nom facile à mémoriser pour une collection de rôles'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
  VALEUR UNIQUE
)
```

Le type d'attribut pcelsElementList réalise l'association de ElementInPolicyRoleCollection [RFC3460]. Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de distinguishedNameMatch [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Dans une pcelsRoleCollection, l'attribut pcelsElementList représente les associations entre cette collection de rôles et ses membres.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.31
  NOM 'pcelsElementList'
  DESC 'Ensemble non ordonné d'éléments gérés'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.12
)
```

## 5.20 Classe abstraite pcelsFilterEntryBase

La classe pcelsFilterEntryBase est la classe de base pour définir des filtres de message ou de paquet. Elle est transposée de la classe FilterEntryBase [RFC3460]. La classe pcelsFilterEntryBase est une classe d'objets abstraite et elle est déduite de la classe pcimPolicy [RFC3703].

La classe pcelsFilterEntryBase est définie comme suit :

```
( 1.3.6.1.1.9.1.52
  NOM 'pcelsFilterEntryBase'
  DESC 'Classe de base pour les filtres de message ou de paquet'
  SUP pcimPolicy
  ABSTRAITE
  PEUT ( pcelsFilterName $ pcelsFilterIsNegated )
)
```

Le type d'attribut pcelsFilterName peut être utilisé comme attribut de désignation pour les entrées de pcelsFilterEntryBase. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de caseIgnoreMatch, une règle de correspondance d'ordre de caseIgnoreOrderingMatch et une règle de correspondance de sous chaînes de caseIgnoreSubstringsMatch [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.32
  NOM 'pcelsFilterName'
  DESC 'Nom facile à mémoriser pour une entrée de filtre'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
  VALEUR UNIQUE
)
```

Le type d'attribut pcelsFilterIsNegated indique si les informations de correspondance spécifiées dans une pcelsFilterEntryBase sont satisfaites ou non. Il est transposé de la propriété FilterEntryBase.IsNegated [RFC3460]. Ce type d'attribut a la syntaxe Boolean [RFC2252]. Il a une règle de correspondance pour égalité de booleanMatch [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Si cet attribut manque dans une instance de pcelsFilterEntryBase, les applications DOIVENT supposer que le filtre n'est pas nié.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.33
  NOM 'pcelsFilterIsNegated'
  DESC 'Indique si le filtre est nié'
  ÉGALITÉ booleanMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.7
  VALEUR UNIQUE
)
```

)

## 5.21 Classe structurelle pcelsIPHeadersFilter

La classe pcelsIPHeadersFilter fournit les attributs le plus fréquemment exigés pour effectuer un filtrage sur les en-têtes IP, TCP ou UDP. Elle est transposée de la classe IpHeadersFilter [RFC3460]. C'est une classe d'objets structurelle dérivée de la classe pcelsFilterEntryBase.

La classe pcelsIPHeadersFilter est définie comme suit :

( 1.3.6.1.1.9.1.53

NOM 'pcelsIPHeadersFilter'

DESC 'Filtre d'en-tête IP'

SUP pcelsFilterEntryBase

STRUCTURELLE

PEUT ( pcelsIPHdrVersion \$ pcelsIPHdrSourceAddress \$ pcelsIPHdrSourceAddressEndOfRange  
 \$ pcelsIPHdrSourceMask \$ pcelsIPHdrDestAddress \$ pcelsIPHdrDestAddressEndOfRange  
 \$ pcelsIPHdrDestMask \$ pcelsIPHdrProtocolID \$ pcelsIPHdrSourcePortStart \$ pcelsIPHdrSourcePortEnd  
 \$ pcelsIPHdrDestPortStart \$ pcelsIPHdrDestPortEnd \$ pcelsIPHdrDSCPList \$ pcelsIPHdrFlowLabel )

)

Les applications DOIVENT supposer 'toutes valeurs' pour les attributs facultatifs (PEUT) qui ne sont pas présents dans une entrée de pcelsIPHeadersFilter.

La [RFC3460] définit plusieurs contraintes pour la classe IpHeadersFilter et ses propriétés. Toutes ces contraintes (même celles qui, pour abrégé, ne sont pas répétées dans le présent document) s'appliquent à la classe pcelsIPHeadersFilter et à ses attributs. Une entrée de pcelsIPHeadersFilter qui viole une de ces contraintes DEVRAIT être traitée comme invalide et les règles ou groupes de politiques associés à cette entrée DEVRAIENT être traités comme étant désactivés, ce qui signifie que l'exécution de telles règles ou groupes de politiques DEVRAIT être arrêtée.

Le type d'attribut pcelsIPHdrVersion indique la version des adresses IP à filtrer. Il est transposé de la propriété IpHeadersFilter.HdrIpVersion [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de integerMatch [RFC2252] et une règle de correspondance d'ordre de integerOrderingMatch [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont 4 et 6.

Dans une entrée de pcelsIPHeadersFilter, le type d'attribut pcelsIPHdrVersion détermine la taille des valeurs d'attribut selon la version IP. Ces attributs sont : pcelsIPHdrSourceAddress, pcelsIPHdrSourceAddressEndOfRange, pcelsIPHdrSourceMask, pcelsIPHdrDestAddress, pcelsIPHdrDestAddressEndOfRange et pcelsIPHdrDestMask. Leurs valeurs valides sont :

pour IPv4 : des chaînes de 4 octets

pour IPv6 : des chaînes de 16 ou 20 octets

Si l'attribut pcelsIPHdrVersion manque dans une instance de pcelsFilterEntryBase, le filtre ne prend alors pas en compte la version IP pour choisir les paquets qui correspondent. Dans ce cas, les attributs qui dépendent de la version IP (cités ci-dessus) ne doivent pas être présents dans l'entrée de filtre.

Ce type attribut est défini comme suit :

( 1.3.6.1.1.9.2.34

NOM 'pcelsIPHdrVersion'

DESC 'Version IP'

ÉGALITÉ integerMatch

ORDRE integerOrderingMatch

SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27

VALEUR UNIQUE

)

Le type d'attribut pcelsIPHdrSourceAddress représente une adresse IP de source. Il est transposé de la propriété IpHeadersFilter.HdrSrcAddress [RFC3460]. Ce type d'attribut a la syntaxe OctetString [RFC2252]. Il a une règle de correspondance pour égalité de octetStringMatch [RFC2256] et une règle de correspondance d'ordre de octetStringOrderingMatch [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs

admises pour les attributs de ce type sont des chaînes d'octets d'une taille de 4, 16, ou 20.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.35
  NOM 'pcelsIPHdrSourceAddress'
  DESC 'Adresse IP de source'
  ÉGALITÉ octetStringMatch
  ORDRE octetStringOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
  VALEUR UNIQUE
)
```

Le type d'attribut `pcelsIPHdrSourceAddressEndOfRange` représente la fin d'une gamme d'adresses IP de source. Il est transposé de la propriété `IpHeadersFilter.HdrSrcAddressEndOfRange` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de `octetStringMatch` [RFC2256] et une règle de correspondance d'ordre de `octetStringOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des chaînes de 4, 16, ou 20 octets.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.36
  NOM 'pcelsIPHdrSourceAddressEndOfRange'
  DESC 'Fin d'une gamme d'adresses IP de source'
  ÉGALITÉ octetStringMatch
  ORDRE octetStringOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
  VALEUR UNIQUE
)
```

Le type d'attribut `pcelsIPHdrSourceMask` représente le gabarit à utiliser pour comparer l'adresse IP de source. Il est transposé de la propriété `IpHeadersFilter.HdrSrcMask` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de `octetStringMatch` [RFC2256] et une règle de correspondance d'ordre de `octetStringOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des chaînes de 4, 16, ou 20 octets.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.37
  NOM 'pcelsIPHdrSourceMask'
  DESC 'Gabarit à utiliser pour comparer l'adresse IP de source'
  ÉGALITÉ octetStringMatch
  ORDRE octetStringOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
  VALEUR UNIQUE
)
```

Le type d'attribut `pcelsIPHdrDestAddress` représente une adresse IP de destination. Il est transposé de la propriété `IpHeadersFilter.HdrDestAddress` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de `octetStringMatch` [RFC2256] et une règle de correspondance d'ordre de `octetStringOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des chaînes de 4, 16, ou 20 octets.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.38
  NOM 'pcelsIPHdrDestAddress'
  DESC 'Adresse IP de destination'
  ÉGALITÉ octetStringMatch
  ORDRE octetStringOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
  VALEUR UNIQUE
)
```

Le type d'attribut `pcelsIPHdrDestAddressEndOfRange` représente la fin d'une gamme d'adresses IP de destination. Il est transposé de la propriété `IpHeadersFilter.HdrDestAddressEndOfRange` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de `octetStringMatch` [RFC2256] et une règle de correspondance d'ordre de `octetStringOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des chaînes de 4, 16, ou 20 octets.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.39
  NOM 'pcelsIPHdrDestAddressEndOfRange'
  DESC 'Fin d'une gamme d'adresses IP de destination'
  ÉGALITÉ octetStringMatch
  ORDRE octetStringOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
  VALEUR UNIQUE
)
```

Le type d'attribut `pcelsIPHdrDestMask` représente un gabarit à utiliser pour comparer l'adresse IP de destination. Il est transposé de la propriété `IpHeadersFilter.HdrDestMask` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de `octetStringMatch` [RFC2256] et une règle de correspondance d'ordre de `octetStringOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des chaînes de 4, 16, ou 20 octets.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.40
  NOM 'pcelsIPHdrDestMask'
  DESC 'Gabarit à utiliser pour comparer l'adresse IP de destination'
  ÉGALITÉ octetStringMatch
  ORDRE octetStringOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
  VALEUR UNIQUE
)
```

Le type d'attribut `pcelsIPHdrProtocolID` indique un type de protocole IP. Il est transposé de la propriété `IpHeadersFilter.HdrProtocolID` [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de `integerMatch` [RFC2252] et une règle de correspondance d'ordre de `integerOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des entiers dans la gamme 0 à 255 (inclus).

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.41
  NOM 'pcelsIPHdrProtocolID'
  DESC 'Type de protocole IP'
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
  VALEUR UNIQUE
)
```

Le type d'attribut `pcelsIPHdrSourcePortStart` représente l'extrémité inférieure d'une gamme d'accès de source UDP ou TCP. Il est transposé de la propriété `IpHeadersFilter.HdrSrcPortStart` [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de `integerMatch` [RFC2252] et une règle de correspondance d'ordre de `integerOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des entiers dans la gamme 0 à 65 535 (inclus).

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.42
  NOM 'pcelsIPHdrSourcePortStart'
  DESC 'Extrémité inférieure d'une gamme d'accès de source UDP ou TCP'
```

ÉGALITÉ integerMatch  
 ORDRE integerOrderingMatch  
 SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27  
 VALEUR UNIQUE  
 )

Le type d'attribut pcelsIPHdrSourcePortEnd représente l'extrémité supérieure d'une gamme d'accès de source UDP ou TCP. Il est transposé de la propriété IpHeadersFilter.HdrSrcPortEnd [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de integerMatch [RFC2252] et une règle de correspondance d'ordre de integerOrderingMatch [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des entiers dans la gamme 0 à 65 535 (inclus).

Ce type attribut est défini comme suit :

( 1.3.6.1.1.9.2.43  
 NOM 'pcelsIPHdrSourcePortEnd'  
 DESC 'Extrémité supérieure d'une gamme d'accès de source UDP ou TCP'  
 ÉGALITÉ integerMatch  
 ORDRE integerOrderingMatch  
 SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27  
 VALEUR UNIQUE  
 )

Le type d'attribut pcelsIPHdrDestPortStart représente extrémité inférieure d'une gamme d'accès de destination UDP ou TCP. Il est transposé de la propriété IpHeadersFilter.HdrDestPortStart [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de integerMatch [RFC2252] et une règle de correspondance d'ordre de integerOrderingMatch [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des entiers dans la gamme 0 à 65 535 (inclus).

Ce type attribut est défini comme suit :

( 1.3.6.1.1.9.2.44  
 NOM 'pcelsIPHdrDestPortStart'  
 DESC 'Extrémité inférieure d'une gamme d'accès de destination UDP ou TCP'  
 ÉGALITÉ integerMatch  
 ORDRE integerOrderingMatch  
 SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27  
 VALEUR UNIQUE  
 )

Le type d'attribut pcelsIPHdrDestPortEnd représente l'extrémité supérieure d'une gamme d'accès de destination UDP ou TCP. Il est transposé de la propriété IpHeadersFilter.HdrDestPortEnd [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de integerMatch [RFC2252] et une règle de correspondance d'ordre de integerOrderingMatch [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des entiers dans la gamme 0 à 65 535 (inclus).

Ce type attribut est défini comme suit :

( 1.3.6.1.1.9.2.45  
 NOM 'pcelsIPHdrDestPortEnd'  
 DESC 'extrémité supérieure d'une gamme d'accès de destination UDP ou TCP'  
 ÉGALITÉ integerMatch  
 ORDRE integerOrderingMatch  
 SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27  
 VALEUR UNIQUE  
 )

Le type d'attribut pcelsIPHdrDSCPList est transposé de la propriété IpHeadersFilter.HdrDSCP [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de integerMatch [RFC2252] et une règle de correspondance d'ordre de integerOrderingMatch [RFC3698]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs de ce type sont des entiers dans la gamme 0 à 63 (inclus).

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.46
  NOM 'pcelsIPHdrDSCPList'
  DESC 'Valeurs DSCP'
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
)
```

Le type d'attribut pcelsIPHdrFlowLabel est transposé de la propriété IpHeadersFilter.HdrFlowLabel [RFC3460]. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de octetStringMatch [RFC2256] et une règle de correspondance d'ordre de octetStringOrderingMatch [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des chaînes de trois octets (soit 24 bits) qui contiennent dans les 20 bits de droite une valeur d'étiquette de flux bourrée à gauche avec "0000".

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.47
  NOM 'pcelsIPHdrFlowLabel'
  DESC 'Étiquette de flux IP'
  ÉGALITÉ octetStringMatch
  ORDRE octetStringOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
  VALEUR UNIQUE
)
```

## 5.22 Classe structurelle pcels8021Filter

La classe pcels8021Filter fournit des attributs 802.1 pour effectuer un filtrage sur les en-têtes 802.1. Elle est transposée de la classe 8021Filter [RFC3460]. La classe pcels8021Filter est une classe d'objets structurelle et elle est déduite de la classe pcelsFilterEntryBase.

La classe pcels8021Filter est définie comme suit :

```
( 1.3.6.1.1.9.1.54
  NOM 'pcels8021Filter'
  DESC 'Filtre d'en-tête 802.1'
  SUP pcelsFilterEntryBase
  STRUCTURAL
  PEUT ( pcels8021HdrSourceMACAddress $ pcels8021HdrSourceMACMask $ pcels8021HdrDestMACAddress
        $ pcels8021HdrDestMACMask $ pcels8021HdrProtocolID $ pcels8021HdrPriority $ pcels8021HdrVLANID )
)
```

Les applications DOIVENT supposer "toutes les valeurs" pour les attributs facultatifs (PEUT) qui ne sont pas présents dans une entrée de pcels8021Filter.

La [RFC3460] définit plusieurs contraintes pour la classe 8021Filter et ses propriétés. Toutes ces contraintes (même celles qui, pour alléger, ne sont pas répétées dans le présent document) s'appliquent à la classe pcels8021Filter et ses attributs. Une entrée pcels8021Filter qui viole une de ces contraintes DEVRAIT être traitée comme invalide et les règles ou groupes de politique associés à cette entrée DEVRAIENT être traités comme étant désactivés, ce qui signifie que l'exécution de telles règles ou groupes de politiques DEVRAIT être arrêtée.

Le type d'attribut pcels8021HdrSourceMACAddress représente une adresse MAC de source. Il est transposé de la propriété 8021Filter.8021HdrSrcMACAddr [RFC3460]. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de octetStringMatch [RFC2256] et une règle de correspondance d'ordre de octetStringOrderingMatch [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des chaînes de 6 octets.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.48
  NOM 'pcels8021HdrSourceMACAddress'
```

```

DESC 'Adresse MAC de source'
ÉGALITÉ octetStringMatch
ORDRE octetStringOrderingMatch
SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
VALEUR UNIQUE
)

```

Le type d'attribut `pcels8021HdrSourceMACMask` représente le gabarit à utiliser pour comparer l'adresse MAC de source. Il est transposé de la propriété `8021Filter.8021HdrSrcMACMask` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de `octetStringMatch` [RFC2256] et une règle de correspondance d'ordre de `octetStringOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des chaînes de 6 octets.

Ce type attribut est défini comme suit :

```

( 1.3.6.1.1.9.2.49
NOM 'pcels8021HdrSourceMACMask'
DESC 'Gabarit d'adresse MAC de source'
ÉGALITÉ octetStringMatch
ORDRE octetStringOrderingMatch
SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
VALEUR UNIQUE
)

```

Le type d'attribut `pcels8021HdrDestMACAddress` représente une adresse MAC de destination. Il est transposé de la propriété `8021Filter.8021HdrDestMACAddr` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de `octetStringMatch` [RFC2256] et une règle de correspondance d'ordre de `octetStringOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des chaînes de 6 octets.

Ce type attribut est défini comme suit :

```

( 1.3.6.1.1.9.2.50
NOM 'pcels8021HdrDestMACAddress'
DESC 'Adresse MAC de destination'
ÉGALITÉ octetStringMatch
ORDRE octetStringOrderingMatch
SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
VALEUR UNIQUE
)

```

Le type d'attribut `pcels8021HdrDestMACMask` représente le gabarit à utiliser pour comparer l'adresse MAC de destination. Il est transposé de la propriété `8021Filter.8021HdrDestMACMask` [RFC3460]. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de `octetStringMatch` [RFC2256] et une règle de correspondance d'ordre de `octetStringOrderingMatch` [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont des chaînes de 6 octets.

Ce type attribut est défini comme suit :

```

( 1.3.6.1.1.9.2.51
NOM 'pcels8021HdrDestMACMask'
DESC 'Gabarit d'adresse MAC de destination'
ÉGALITÉ octetStringMatch
ORDRE octetStringOrderingMatch
SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
VALEUR UNIQUE
)

```

Le type d'attribut `pcels8021HdrProtocolID` indique un type de protocole Ethernet. Il est transposé de la propriété `8021Filter.8021HdrProtocolID` [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de `integerMatch` [RFC2252] et une règle de correspondance d'ordre de `integerOrderingMatch` [RFC3698]. Les attributs de ce type peuvent avoir plusieurs valeurs. Aucun ordre n'est impliqué. Les seules valeurs admises pour les attributs de ce type sont des entiers dans la gamme 0 à 65 535 (inclus).

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.52
  NOM 'pcels8021HdrProtocolID'
  DESC 'Identifiant de protocole Ethernet'
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
)
```

Le type d'attribut `pcels8021HdrPriority` indique une priorité 802.1Q. Il est transposé de la propriété `8021Filter.8021HdrPriorityValue` [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de `integerMatch` [RFC2252] et une règle de correspondance d'ordre de `integerOrderingMatch` [RFC3698]. Les attributs de ce type peuvent avoir plusieurs valeurs. Aucun ordre n'est impliqué. Les seules valeurs admises pour les attributs de ce type sont des entiers dans la gamme 0 à 7 (inclus).

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.53
  NOM 'pcels8021HdrPriority'
  DESC 'Priorité 802.1Q'
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
)
```

Le type d'attribut `pcels8021HdrVLANID` indique un identifiant de VLAN 802.1Q. Il est transposé de la propriété `8021Filter.8021HdrVLANID` [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de `integerMatch` [RFC2252] et une règle de correspondance d'ordre de `integerOrderingMatch` [RFC3698]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs de ce type sont des entiers dans la gamme 0 à 4095 (inclus).

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.54
  NOM 'pcels8021HdrVLANID'
  DESC 'Identifiant de VLAN 802.1Q'
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
)
```

### 5.23 Classe auxiliaire `pcelsFilterListAuxClass`

La classe `pcelsFilterListAuxClass` représente une collection de filtres de niveau appareil agrégés dans une condition de politique. Elle est transposée de la classe `FilterList` [RFC3460]. Les instances de `pcelsFilterListAuxClass` peuvent être utilisées comme conditions dans des règles de politique ou comme composants dans des conditions composées. La classe `pcelsFilterListAuxClass` est une classe d'objets auxiliaire et elle est déduite de la classe `pcimConditionAuxClass` [RFC3703].

La classe `pcelsFilterListAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.55
  NOM 'pcelsFilterListAuxClass'
  DESC 'Collection de filtres de pcelsFilterEntryBase'
  SUP pcimConditionAuxClass
  AUXILIAIRE
  PEUT ( pcelsFilterListName $ pcelsFilterDirection $ pcelsFilterEntryList )
)
```

Le type d'attribut `pcelsFilterListName` peut être utilisé comme attribut de désignation pour des entrées de

pcelsFilterListAuxClass. Ce type d'attribut a la syntaxe Chaîne de répertoire [RFC2252]. Il a une règle de correspondance pour égalité de caseIgnoreMatch, une règle de correspondance d'ordre de caseIgnoreOrderingMatch et une règle de correspondance de sous chaînes de caseIgnoreSubstringsMatch [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.55
  NOM 'pcelsFilterListName'
  DESC 'Nom facile à mémoriser d'une liste de filtres'
  ÉGALITÉ caseIgnoreMatch
  ORDRE caseIgnoreOrderingMatch
  SOUS CHAÎNE caseIgnoreSubstringsMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.15
  VALEUR UNIQUE
)
```

Le type d'attribut pcelsFilterDirection indique la direction des paquets ou messages par rapport à l'interface où le filtre est appliqué. Il est transposé de la propriété FilterList.Direction [RFC3460]. Ce type d'attribut a la syntaxe Entier [RFC2252]. Il a une règle de correspondance pour égalité de integerMatch [RFC2252] et une règle de correspondance d'ordre de integerOrderingMatch [RFC3698]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Les seules valeurs admises pour les attributs de ce type sont 0 (NonApplicable), 1 (Entrée), 2 (Sortie), 3 (LesDeux) et 4 (Miroir). Si cet attribut manque dans une instance de pcelsFilterListAuxClass, les applications DOIVENT supposer qu'une direction n'est pas applicable.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.56
  NOM 'pcelsFilterDirection'
  DESC 'Direction dans laquelle ce filtre est appliqué'
  ÉGALITÉ integerMatch
  ORDRE integerOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.27
  VALEUR UNIQUE
)
```

Le type d'attribut pcelsFilterEntryList réalise l'association de EntriesInFilterList [RFC3460]. Ce type d'attribut a la syntaxe DN [RFC2252]. Il a une règle de correspondance pour égalité de distinguishedNameMatch [RFC2252]. Les attributs de ce type peuvent avoir plusieurs valeurs. Les seules valeurs admises pour les attributs pcelsFilterEntryList sont les DN des entrées de pcelsFilterEntryBase. Dans une pcelsFilterListAuxClass, l'attribut pcelsFilterEntryList représente les associations entre cette collection de filtres et ses composants.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.57
  NOM 'pcelsFilterEntryList'
  DESC 'E,semble non ordonné de DN d'entrées de pcelsFilterEntryBase'
  ÉGALITÉ distinguishedNameMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.12
)
```

La propriété EntrySequence de l'association EntriesInFilterList se restreint à une seule valeur ('0') [RFC3460] qui la rend redondante. Donc, sa transposition à un élément du schéma LDAP n'est pas nécessaire.

## 5.24 Classe auxiliaire pcelsVendorVariableAuxClass

La classe pcelsVendorVariableAuxClass fournit un mécanisme général d'extension pour représenter les variables de politique qui n'ont pas été spécifiquement modélisées. Ses deux propriétés sont plutôt utilisées pour définir le contenu et le format de la variable, comme expliqué ci-dessous. Cette classe est destinée aux extensions spécifiques de fabricant qui ne sont pas éligibles à l'utilisation de pcelsVariable ; les extensions normalisées NE DEVRAIENT PAS utiliser cette classe.

La classe pcelsVendorVariableAuxClass est une classe d'objets auxiliaire et elle est déduite de la classe pcelsVariable.

La classe `pcelsVendorVariableAuxClass` est définie comme suit :

```
( 1.3.6.1.1.9.1.56
  NOM 'pcelsVendorVariableAuxClass'
  DESC 'Définit des moyens enregistrés pour décrire une variable de politique'
  SUP pcelsVariable
  AUXILIAIRE
  PEUT ( pcelsVendorVariableData $ pcelsVendorVariableEncoding )
)
```

L'attribut `pcelsVendorVariableData` fournit un mécanisme général pour représenter les variables de politique qui n'ont pas été spécifiquement modélisées. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de `octetStringMatch` [RFC2256] et une règle de correspondance d'ordre de `octetStringOrderingMatch` [RFC3698]. Les attributs de ce type peuvent avoir plusieurs valeurs. Dans les instances de `pcelsVendorVariableAuxClass`, le format des valeurs pour les attributs de ce type est identifié par l'OID mémorisé dans l'attribut `pcelsVendorVariableEncoding`.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.58
  NOM 'pcelsVendorVariableData'
  DESC 'Mécanisme pour représenter des variables qui n'ont pas été spécifiquement modélisées'
  ÉGALITÉ octetStringMatch
  ORDRE octetStringOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
)
```

L'attribut `pcelsVendorVariableEncoding` identifie le format pour représenter les variables de politique qui n'ont pas été spécifiquement modélisées. Ce type d'attribut a la syntaxe OID [RFC2252]. Il a une règle de correspondance pour égalité de `objectIdentifierMatch` [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Dans les instances de `pcelsVendorVariableAuxClass`, l'attribut `pcelsVendorVariableEncoding` est utilisé pour identifier le format et la sémantique des valeurs d'attribut de `pcelsVendorVariableData`.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.59
  NOM 'pcelsVendorVariableEncoding'
  DESC 'Identifie le format et la sémantique des variables de politique'
  ÉGALITÉ objectIdentifierMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.38
  VALEUR UNIQUE
)
```

## 5.25 Classe auxiliaire `pcelsVendorValueAuxClass`

La classe `pcelsVendorValueAuxClass` fournit un mécanisme général d'extension pour représenter des valeurs de politique qui n'ont pas été spécifiquement modélisées. Ses deux propriétés sont plutôt utilisées pour définir le contenu et le format de la valeur de politique, comme expliqué ci-dessous. Cette classe est destinée aux extensions spécifiques de fabricant qui ne sont pas éligibles à l'utilisation de la `pcelsValueAuxClass` ; les extensions normalisées NE DEVRAIENT PAS utiliser cette classe.

La classe `pcelsVendorValueAuxClass` est une classe d'objets auxiliaire et elle est déduite de la classe `pcelsValueAuxClass`.

La classe `pcelsVendorValueAuxClass` class est définie comme suit :

```
( 1.3.6.1.1.9.1.57
  NOM 'pcelsVendorValueAuxClass'
  DESC 'Définit des moyens enregistrés de décrire une valeur de politique'
  SUP pcelsValueAuxClass
  AUXILIAIRE
  PEUT ( pcelsVendorValueData $ pcelsVendorValueEncoding )
)
```

)

L'attribut `pcelsVendorValueData` fournit un mécanisme général pour représenter des valeurs de politique qui n'ont pas été spécifiquement modélisées. Ce type d'attribut a la syntaxe Chaîne d'octets [RFC2252]. Il a une règle de correspondance pour égalité de `octetStringMatch` [RFC2256] et une règle de correspondance d'ordre de `octetStringOrderingMatch` [RFC3698]. Les attributs de ce type peuvent avoir plusieurs valeurs. Dans les instances de `pcelsVendorValueAuxClass`, le format des valeurs pour les attributs de ce type est identifié par l'OID mémorisé dans l'attribut `pcelsVendorValueEncoding`.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.60
  NOM 'pcelsVendorValueData'
  DESC 'Mécanisme pour représenter les valeurs qui n'ont pas été spécifiquement modélisées'
  ÉGALITÉ octetStringMatch
  ORDRE octetStringOrderingMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.40
)
```

L'attribut `pcelsVendorValueEncoding` identifie le format pour représenter les valeurs de politique qui n'ont pas été spécifiquement modélisées. Ce type d'attribut a la syntaxe OID [RFC2252]. Il a une règle de correspondance pour égalité de `objectIdentifierMatch` [RFC2252]. Les attributs de ce type ne peuvent avoir qu'une seule valeur. Dans les instances de `pcelsVendorValueAuxClass`, l'attribut `pcelsVendorValueEncoding` est utilisé pour identifier le format et la sémantique pour les valeurs d'attribut de `pcelsVendorValueData`.

Ce type attribut est défini comme suit :

```
( 1.3.6.1.1.9.2.61
  NOM 'pcelsVendorValueEncoding'
  DESC 'Identifie le format et la sémantique des valeurs de politique'
  ÉGALITÉ objectIdentifierMatch
  SYNTAXE 1.3.6.1.4.1.1466.115.121.1.38
  VALEUR UNIQUE
)
```

## 6. Considérations sur la sécurité

Le schéma de cœur de politique LDAP [RFC3703] décrit les considérations générales de sécurité relatives au schéma général de cœur de politique. Les extensions définies dans le présent document n'introduisent aucune considération supplémentaire à l'égard de la sécurité.

## 7. Considérations relatives à l'IANA

Se référer à la [RFC3383], "Autorité d'allocation des numéros de l'Internet (IANA) : Considérations sur le protocole léger d'accès à un répertoire (LDAP)".

### 7.1 Identifiants d'objet

L'IANA a enregistré un identifiant d'objet LDAP à utiliser dans cette spécification technique conformément au gabarit suivant :

Sujet : Demande d'enregistrement d'OID LDAP

Adresse de messagerie à contacter pour plus d'informations : Mircea Pana ( [mpana@metasolv.com](mailto:mpana@metasolv.com) )

Spécification : RFC 4104

Auteur/Contrôleur des changements : IESG

Commentaires : L'OID alloué est utilisé comme base de l'identification d'un certain nombre d'éléments de schéma définis dans le présent document.

L'IANA a alloué un OID de 1.3.6.1.1.9 du nom de `pcelsSchema` à cet enregistrement dans le registre suivant : <http://www.iana.org/assignments/smi-numbers>

## 7.2 Descripteurs d'identifiant d'objet

L'IANA a enregistré les descripteurs LDAP utilisés dans cette spécification technique comme précisé dans le gabarit suivant :

Sujet : Demande de mise à jour d'enregistrement de descripteur LDAP

Descripteur (nom abrégé) : voir le commentaire

Identifiant d'objet : voir le commentaire

Adresse de messagerie à contacter pour plus d'informations : Mircea Pana ( [mpana@metasolv.com](mailto:mpana@metasolv.com) )

Usage : voir le commentaire

Spécification : RFC 4104

Auteur/Contrôleur des changements : IESG

Commentaire : Les descripteurs suivants ont été ajoutés :

Nom	Type	OID
pcelsPolicySet	O	1.3.6.1.1.9.1.1
pcelsPolicySetAssociation	O	1.3.6.1.1.9.1.2
pcelsGroup	O	1.3.6.1.1.9.1.3
pcelsGroupAuxClass	O	1.3.6.1.1.9.1.4
pcelsGroupInstance	O	1.3.6.1.1.9.1.5
pcelsRule	O	1.3.6.1.1.9.1.6
pcelsRuleAuxClass	O	1.3.6.1.1.9.1.7
pcelsRuleInstance	O	1.3.6.1.1.9.1.8
pcelsConditionAssociation	O	1.3.6.1.1.9.1.9
pcelsActionAssociation	O	1.3.6.1.1.9.1.10
pcelsSimpleConditionAuxClass	O	1.3.6.1.1.9.1.11
pcelsCompoundConditionAuxClass	O	1.3.6.1.1.9.1.12
pcelsCompoundFilterConditionAuxClass	O	1.3.6.1.1.9.1.13
pcelsSimpleActionAuxClass	O	1.3.6.1.1.9.1.14
pcelsCompoundActionAuxClass	O	1.3.6.1.1.9.1.15
pcelsVariable	O	1.3.6.1.1.9.1.16
pcelsExplicitVariableAuxClass	O	1.3.6.1.1.9.1.17
pcelsImplicitVariableAuxClass	O	1.3.6.1.1.9.1.18
pcelsSourceIPv4VariableAuxClass	O	1.3.6.1.1.9.1.19
pcelsSourceIPv6VariableAuxClass	O	1.3.6.1.1.9.1.20
pcelsDestinationIPv4VariableAuxClass	O	1.3.6.1.1.9.1.21
pcelsDestinationIPv6VariableAuxClass	O	1.3.6.1.1.9.1.22
pcelsSourcePortVariableAuxClass	O	1.3.6.1.1.9.1.23
pcelsDestinationPortVariableAuxClass	O	1.3.6.1.1.9.1.24
pcelsIPProtocolVariableAuxClass	O	1.3.6.1.1.9.1.25
pcelsIPVersionVariableAuxClass	O	1.3.6.1.1.9.1.26
pcelsIPToSVariableAuxClass	O	1.3.6.1.1.9.1.27
pcelsDSCPVariableAuxClass	O	1.3.6.1.1.9.1.28
pcelsFlowIdVariableAuxClass	O	1.3.6.1.1.9.1.29
pcelsSourceMACVariableAuxClass	O	1.3.6.1.1.9.1.30
pcelsDestinationMACVariableAuxClass	O	1.3.6.1.1.9.1.31
pcelsVLANVariableAuxClass	O	1.3.6.1.1.9.1.32
pcelsCoSVariableAuxClass	O	1.3.6.1.1.9.1.33
pcelsEthertypeVariableAuxClass	O	1.3.6.1.1.9.1.34
pcelsSourceSAPVariableAuxClass	O	1.3.6.1.1.9.1.35
pcelsDestinationSAPVariableAuxClass	O	1.3.6.1.1.9.1.36
pcelsSNAPOUIVariableAuxClass	O	1.3.6.1.1.9.1.37
pcelsSNAPTypeVariableAuxClass	O	1.3.6.1.1.9.1.38
pcelsFlowDirectionVariableAuxClass	O	1.3.6.1.1.9.1.39
pcelsValueAuxClass	O	1.3.6.1.1.9.1.40
pcelsIPv4AddrValueAuxClass	O	1.3.6.1.1.9.1.41
pcelsIPv6AddrValueAuxClass	O	1.3.6.1.1.9.1.42
pcelsMACAddrValueAuxClass	O	1.3.6.1.1.9.1.43
pcelsStringValueAuxClass	O	1.3.6.1.1.9.1.44
pcelsBitStringValueAuxClass	O	1.3.6.1.1.9.1.45
pcelsIntegerValueAuxClass	O	1.3.6.1.1.9.1.46

pcelsBooleanValueAuxClass	O	1.3.6.1.1.9.1.47
pcelsReusableContainer	O	1.3.6.1.1.9.1.48
pcelsReusableContainerAuxClass	O	1.3.6.1.1.9.1.49
pcelsReusableContainerInstance	O	1.3.6.1.1.9.1.50
pcelsRoleCollection	O	1.3.6.1.1.9.1.51
pcelsFilterEntryBase	O	1.3.6.1.1.9.1.52
pcelsIPHeadersFilter	O	1.3.6.1.1.9.1.53
pcels8021Filter	O	1.3.6.1.1.9.1.54
pcelsFilterListAuxClass	O	<sup>2</sup> 1.3.6.1.1.9.1.55
pcelsVendorVariableAuxClass	O	1.3.6.1.1.9.1.56
pcelsVendorValueAuxClass	O	1.3.6.1.1.9.1.57
pcelsPolicySetName	A	1.3.6.1.1.9.2.1
pcelsDecisionStrategy	A	1.3.6.1.1.9.2.2
pcelsPolicySetList	A	1.3.6.1.1.9.2.3
pcelsPriority	A	1.3.6.1.1.9.2.4
pcelsPolicySetDN	A	1.3.6.1.1.9.2.5
pcelsConditionListType	A	1.3.6.1.1.9.2.6
pcelsConditionList	A	1.3.6.1.1.9.2.7
pcelsActionList	A	1.3.6.1.1.9.2.8
pcelsSequencedActions	A	1.3.6.1.1.9.2.9
pcelsExecutionStrategy	A	1.3.6.1.1.9.2.10
pcelsVariableDN	A	1.3.6.1.1.9.2.11
pcelsValueDN	A	1.3.6.1.1.9.2.12
pcelsIsMirrored	A	1.3.6.1.1.9.2.13
pcelsVariableName	A	1.3.6.1.1.9.2.14
pcelsExpectedValueList	A	1.3.6.1.1.9.2.15
pcelsVariableModelClass	A	1.3.6.1.1.9.2.16
pcelsVariableModelProperty	A	1.3.6.1.1.9.2.17
pcelsExpectedValueTypes	A	1.3.6.1.1.9.2.18
pcelsValueName	A	1.3.6.1.1.9.2.19
pcelsIPv4AddrList	A	1.3.6.1.1.9.2.20
pcelsIPv6AddrList	A	1.3.6.1.1.9.2.21
pcelsMACAddrList	A	1.3.6.1.1.9.2.22
pcelsStringList	A	1.3.6.1.1.9.2.23
pcelsBitStringList	A	1.3.6.1.1.9.2.24
pcelsIntegerList	A	1.3.6.1.1.9.2.25
pcelsBoolean	A	1.3.6.1.1.9.2.26
pcelsReusableContainerName	A	1.3.6.1.1.9.2.27
pcelsReusableContainerList	A	1.3.6.1.1.9.2.28
pcelsRole	A	1.3.6.1.1.9.2.29
pcelsRoleCollectionName	A	1.3.6.1.1.9.2.30
pcelsElementList	A	1.3.6.1.1.9.2.31
pcelsFilterName	A	1.3.6.1.1.9.2.32
pcelsFilterIsNegated	A	1.3.6.1.1.9.2.33
pcelsIPHdrVersion	A	1.3.6.1.1.9.2.34
pcelsIPHdrSourceAddress	A	1.3.6.1.1.9.2.35
pcelsIPHdrSourceAddressEndOfRange	A	1.3.6.1.1.9.2.36
pcelsIPHdrSourceMask	A	1.3.6.1.1.9.2.37
pcelsIPHdrDestAddress	A	1.3.6.1.1.9.2.38
pcelsIPHdrDestAddressEndOfRange	A	1.3.6.1.1.9.2.39
pcelsIPHdrDestMask	A	1.3.6.1.1.9.2.40
pcelsIPHdrProtocolID	A	1.3.6.1.1.9.2.41
pcelsIPHdrSourcePortStart	A	1.3.6.1.1.9.2.42
pcelsIPHdrSourcePortEnd	A	1.3.6.1.1.9.2.43
pcelsIPHdrDestPortStart	A	1.3.6.1.1.9.2.44
pcelsIPHdrDestPortEnd	A	1.3.6.1.1.9.2.45
pcelsIPHdrDSCPList	A	1.3.6.1.1.9.2.46
pcelsIPHdrFlowLabel	A	1.3.6.1.1.9.2.47
pcels8021HdrSourceMACAddress	A	1.3.6.1.1.9.2.48
pcels8021HdrSourceMACMask	A	1.3.6.1.1.9.2.49
pcels8021HdrDestMACAddress	A	1.3.6.1.1.9.2.50
pcels8021HdrDestMACMask	A	1.3.6.1.1.9.2.51
pcels8021HdrProtocolID	A	1.3.6.1.1.9.2.52

pcels8021HdrPriority	A	1.3.6.1.1.9.2.53
pcels8021HdrVLANID	A	1.3.6.1.1.9.2.54
pcelsFilterListName	A	1.3.6.1.1.9.2.55
pcelsFilterDirection	A	1.3.6.1.1.9.2.56
pcelsFilterEntryList	A	1.3.6.1.1.9.2.57
pcelsVendorVariableData	A	1.3.6.1.1.9.2.58
pcelsVendorVariableEncoding	A	1.3.6.1.1.9.2.59
pcelsVendorValueData	A	1.3.6.1.1.9.2.60
pcelsVendorValueEncoding	A	1.3.6.1.1.9.2.61
pcelsRuleValidityPeriodList	A	1.3.6.1.1.9.2.62

Type A, attribut ; Type O, classe d'objet

Ces allocations sont enregistrées dans le registre : <http://www.iana.org/assignments/ldap-parameters>

## 8. Remerciements

Merci à Kurt Zeilenga, Bert Wijnen, Ryan Moats, John Strassner, David McTavish, Larry Bartz et à tous les autres membres du groupe de travail Cadre de politique pour la relecture du document et les nombreuses et utiles suggestions et corrections.

Merci aussi à Joel Halpern (coprésident du groupe de travail Cadre de politique) pour son soutien, pour avoir soumis le présent document à l'attention du groupe de travail Cadre de politique et pour sa régulation des interactions résultantes.

## 9. Références normatives

- [CIM] Distributed Management Task Force, Inc., "Common Information Model (CIM) Specification", Version 2.2, 14 juin 1999, <http://www.dmtf.org/standards/documents/CIM/DSP0004.pdf>
- [CIM\_LDAP] Distributed Management Task Force, Inc., "DMTF LDAP Schema for the CIM v2.5 Core Information Model", 15 avril 2002, <http://www.dmtf.org/standards/documents/DEN/DSP0123.pdf>
- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. (MàJ par [RFC8174](#))
- [RFC2252] M. Wahl, A. Coulbeck, T. Howes, S. Kille, "[Protocole léger d'accès à un répertoire](#) (v3) : Définitions de syntaxe d'attribut", décembre 1997. (*Obsolète, voir [RFC4510](#), [RFC4517](#), [RFC4523](#), [RFC4512](#)*) (MàJ par [RFC3377](#)) (P.S.)
- [RFC2256] M. Wahl, "Résumé du schéma d'utilisateur X.500(96) à utiliser avec LDAPv3", décembre 1997. (*Obsolète, voir [RFC4517](#), [RFC4519](#), [RFC4523](#), [RFC4512](#), [RFC4510](#)*) (P.S.)
- [RFC3060] B. Moore et autres, "Spécification du [modèle d'information de cœur de politique](#) -- version 1", février 2001. (MàJ par [RFC3460](#)) (P.S.)
- [RFC3377] J. Hodges, R. Morgan, "Protocole léger d'accès à un répertoire (v3) : Spécification technique", septembre 2002. *Obsolète, voir [RFC4510](#)* (P.S.)
- [RFC3460] B. Moore, éd., "[Extensions au modèle d'information](#) de cœur de politique (PCIM)", janvier 2003. (P.S.)
- [RFC3698] K. Zeilenga, éd., "Protocole léger d'accès à un répertoire (LDAP) : [règles de correspondance supplémentaires](#)", février 2004. (MàJ par [RFC4517](#)) (P.S.)
- [RFC3703] J. Strassner et autres, "[Schéma de cœur de politique](#) du protocole léger d'accès à un répertoire (LDAP)", février 2004. (MàJ par [RFC4104](#)) (P.S.)
- [X.501] Recommandation UIT-T X.501, "L'annuaire : modèles". 2001.
- [X.520] Recommandation UIT-T X.520, "L'annuaire : Types d'attribut choisis". 2001.

## 10. Références pour information

[RFC3383] K. Zeilenga, "Autorité d'allocation des numéros de l'Internet (IANA) : Considérations sur le protocole léger d'accès à un répertoire (LDAP)", septembre 2002. (*Obsolète, voir RFC4520*)

### Adresse des auteurs

Mircea Pana  
MetaSolv Software Inc.  
360 Legget Drive  
Ottawa, Ontario, Canada  
K2K 3N1  
mél : [mpana@metasolv.com](mailto:mpana@metasolv.com)

Angelica Reyes  
Department of Computer Architecture  
Technical University of Catalonia  
Campus Castelldefels  
Spain  
mél : [mreyes@ac.upc.edu](mailto:mreyes@ac.upc.edu)

Antoni Barba  
Technical University of Catalonia  
Jordi-Girona 1-3  
08034 Barcelona  
Spain  
mél : [telabm@mat.upc.es](mailto:telabm@mat.upc.es)

David Moron  
Technical University of Catalonia  
Jordi-Girona 1-3  
08034 Barcelona  
Spain  
mél : [dmor4477@hotmail.com](mailto:dmor4477@hotmail.com)

Marcus Brunner  
NEC Europe Ltd.  
Kurfuersten-Anlage 36  
D-69115 Heidelberg  
Germany  
mél : [brunner@netlab.nec.de](mailto:brunner@netlab.nec.de)

### Déclaration complète de droits de reproduction

Copyright (C) The Internet Society (2005).

Le présent document est soumis aux droits, licences et restrictions contenus dans le BCP 78, et à [www.rfc-editor.org](http://www.rfc-editor.org), et sauf pour ce qui est mentionné ci-après, les auteurs conservent tous leurs droits.

Le présent document et les informations contenues sont fournies sur une base "EN L'ÉTAT" et le contributeur, l'organisation qu'il ou elle représente ou qui le/la finance (s'il en est), la INTERNET SOCIETY et la INTERNET ENGINEERING TASK FORCE déclinent toutes garanties, exprimées ou implicites, y compris mais non limitées à toute garantie que l'utilisation des informations ci encloses ne violent aucun droit ou aucune garantie implicite de commercialisation ou d'aptitude à un objet particulier.

### Propriété intellectuelle

L'IETF ne prend pas position sur la validité et la portée de tout droit de propriété intellectuelle ou autres droits qui pourraient être revendiqués au titre de la mise en œuvre ou l'utilisation de la technologie décrite dans le présent document ou sur la mesure dans laquelle toute licence sur de tels droits pourrait être ou n'être pas disponible ; pas plus qu'elle ne prétend avoir accompli aucun effort pour identifier de tels droits. Les informations sur les procédures de l'ISOC au sujet des droits dans les documents de l'ISOC figurent dans les BCP 78 et BCP 79.

Des copies des dépôts d'IPR faites au secrétariat de l'IETF et toutes assurances de disponibilité de licences, ou le résultat de tentatives faites pour obtenir une licence ou permission générale d'utilisation de tels droits de propriété par ceux qui mettent en œuvre ou utilisent la présente spécification peuvent être obtenues sur répertoire en ligne des IPR de l'IETF à <http://www.ietf.org/ipr>.

L'IETF invite toute partie intéressée à porter son attention sur tous copyrights, licences ou applications de licence, ou autres droits de propriété qui pourraient couvrir les technologies qui peuvent être nécessaires pour mettre en œuvre la présente norme. Prière d'adresser les informations à l'IETF à [ietf-ipr@ietf.org](mailto:ietf-ipr@ietf.org).

### Remerciement

Le financement de la fonction d'édition des RFC est actuellement fourni par la Internet Society.