université de BORDEAUX

DECOUVRIR LES ANNUAIRES

Aquinetic
INNOVATION TECHNOLOGIES LIBRES EN NOUVELLE-AQUITAINE

Les architectures nouvelles supportés par une technologie de plus en plus sophistiquée impliquent des cycles de décision de plus en plus courts qui s'appuient sur des modèles de connaissance de plus en plus complexes nécessitant l'exploitation rapide d'une masse d'informations sans cesse croissante.

Dans ce cadre, le partage de l'information est essentiel. L'information pertinente doit être disponible pour les utilisateurs sous une forme compréhensible et exploitable. Ce partage ne signifie pas que l'information doit être systématiquement diffusée ou accessible à tous les niveaux.

Le partage de l'information disponible est essentiel : il contribue directement au renforcement des capacités propres d'appréciation de situation, au travail en commun, à la cohésion dans l'action.

Le partage de l'information est rendu possible par la capacité d'interopérabilité du système d'information et celle du système de communication qui est chargée de la transporter. Cela suppose aussi que l'information soit formulée dans un format interprétable (le format de message), et dans certains cas structurée de manière commune (modèles de données).

L'annuaire est en passe de devenir un point central permettant un partage d'information simplifié de part la fédération qu'il assure entre les différents systèmes d'information déployés.

Sommaire :

CHAPITRE 1 :

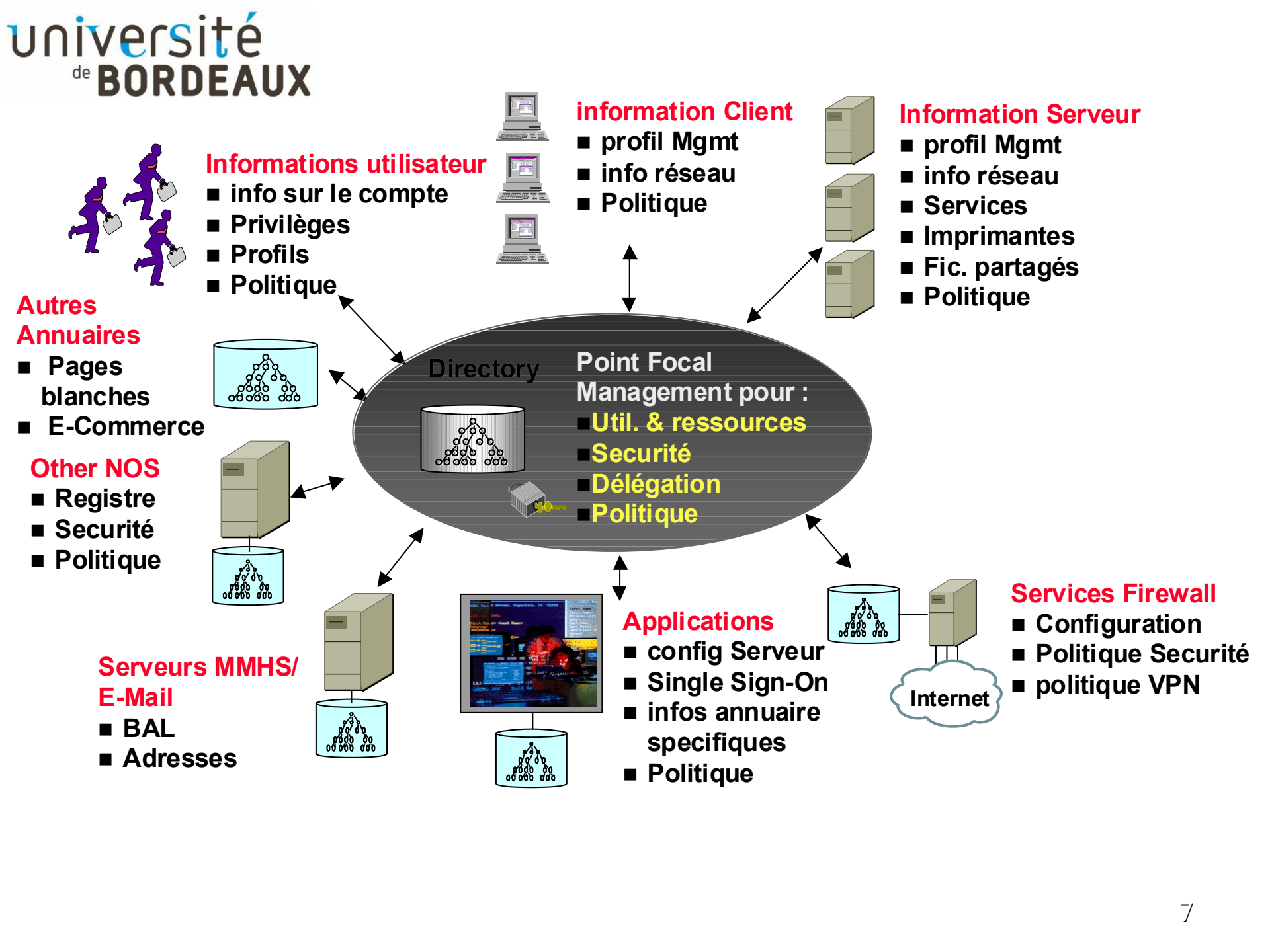
LES ANNUAIRES

Définition

- Un annuaire est un référentiel partagé de personnes et de ressources, dont la vocation est de les localiser à l'aide de fonctions élaborées de navigation et de recherche, et d'offrir des mécanismes de sécurité pour protéger ces informations et y accéder.

- Exemple : - format papier (pages blanches, pages jaunes)
  - format électronique : 3611, carnet d'adresses
  Lotus (Messagerie), Yahoo (Annuaire de sites Web), ...

- Une entrée de l'annuaire n'est pas forcément un individu. Elle peut être une machine, un équipement réseau, ...


université de BORDEAUX
information Client
profil Mgmt
info réseau
Politique
Information Serveur
profil Mgmt
info réseau
Services
Imprimantes
Fic. partagés
Politique
Informations utilisateur
info sur le compte
Privilèges
Profils
Politique
Autres Annuaires
Pages blanches
E-Commerce
Other NOS
Registre
Securité
Politique
Directory
Point Focal Management pour :
Util. & ressources
Securité
Délégation
Politique
Serveurs MMHS/ E-Mail
BAL
Adresses
Applications
config Serveur
Single Sign-On
infos annuaire specifiques
Politique
Internet
Services Firewall
Configuration
Politique Securité
politique VPN

Rôles

Localisation de ressources :

Un annuaire sert avant tout à localiser quelque chose (personnes, entreprises, matériels, ...)

Gérer un parc de ressources :

Un annuaire permet de gérer un parc de ressources (informatiques, ...).

Localiser des applications et des droits d'accès :

Un annuaire permet de localiser des ressources comme des applications, des droits d'accès, ...

Spécificités

- **Les annuaires ne sont pas destinés à gérer des transactions complexes** (les mises à jour des données dans un annuaire se réduisent souvent à la mise à jour de quelques informations concernant une personne ou une ressource).

- **Les annuaires sont plus sollicités en lecture qu'en écriture** (Les annuaires sont destinés à localiser des ressources).

- **Les annuaires offrent un espace de noms homogènes** (Pour localiser rapidement un élément dans un annuaire, il est important de nommer celui-ci de façon homogène).

Spécificités

- **Les annuaires doivent pouvoir communiquer entre eux** (Il ne faut surtout pas centraliser l'ensemble des informations concernant des personnes ou des ressources en un seul endroit).

- **Les informations gérées par un annuaire sont classées de façon hiérarchique** (Pour retrouver une information dans un annuaire, il est plus commode de classer ce qu'elle contient dans une hiérarchie de thèmes).

- **Les annuaires s'appuient sur des bases de données** (stockage des informations).

Spécificités

- **Les annuaires doivent pouvoir gérer les habilitations sur les données de l'annuaire lui-même** (Un annuaire doit permettre de définir des habilitations différentes sur les données en fonction du profil des utilisateurs).

Comparaison

Il est important de souligner les différences entre base de données relationnelles et annuaires. Un annuaire électronique n'a pas pour vocation à stocker uniquement des informations sur des personnes. Il peut être utilisé comme base dans de nombreux types d'applications.

La principale différence est qu'un annuaire électronique est conçu pour être consulté, bien plus que mis à jour. Le rapport lecture sur écriture est donc plus élevé dans les annuaires électroniques que dans les bases de données relationnelles.

L'autre différence est la grande facilité d'extension des annuaires. L'ajout d'attributs, l'équivalent des champs dans les bases de données relationnelles, est très aisé à réaliser. Il ne nécessite pas, par exemple, de reconstruction de la base.

Un exemple

Le D.N.S. est un système d'annuaire hiérarchisé permettant, entre autres, d'établir un lien entre les noms de machines ou de domaines, et les adresses de réseau (adresses IP).

Le fonctionnement du D.N.S. est basé sur une architecture client / Serveur.

Les annuaires de type D.N.S. offrent principalement 3 fonctionnalités :

- obtenir l'adresse I.P. d'une machine dont on connait le nom ;
- obtenir le nom « canonique » d'une machine en ayant son

adresse IP ;

- obtenir des infos sur des domaines (serveur de messagerie).

Un exemple

Les deux implémentations de serveurs D.N.S. les plus utilisées sont **Bind** (Unix) et le serveur **D.N.S. de Microsoft**.

CHAPITRE 2 :

LA NORME LDAP

R.F.C.

Les request for comment (RFC, littéralement demande de commentaires) sont une série de documents et normes concernant l'Internet, commencées en 1969. Peu de RFC sont des standards, mais tous les standards de l'Internet sont enregistrés en tant que RFC.

Les RFC sont rédigées sur l'initiative d'experts techniques, puis sont revues par la communauté Internet dans son ensemble. La première RFC (RFC 1), titrée « Logiciel hôte », a été publiée le 7 avril 1969 par Steve Crocker.

Toute modification apportée à une RFC entraîne l'écriture d'une nouvelle RFC, qui rend obsolète la précédente.

R.F.C. LDAP

La RFC 1777 est apparue en 1993 afin d'alléger le protocole Directory Access Protocol (X500) trop compliqué à mettre en œuvre. Ce document est devenu obsolète avec l'avènement de la RFC 2251 (1997), elle même obsolète aujourd'hui. Depuis, le nombre des RFC sur le sujet ne cessent d'augmenter. L'IETF dénombre au moins 80 RFC sur le sujet. Les plus importantes sont les suivantes :

- Août 2007 : **RFC 5020** The Lightweight Directory Access Protocol (LDAP) entryDN Operational Attribute
- Mai 2007 : **RFC 4876** A Configuration Profile Schema for The Lightweight Directory Access Protocol (LDAP) Based Agents
- Juin 2006 : **RFC 4533** The Lightweight Directory Access Protocol (LDAP) Content Synchronization Operation
- Juin 2006 : **RFC 4532** Lightweight Directory Access Protocol (LDAP) Who am I? Operation
- Juin 2006 : **RFC 4531** Lightweight Directory Access Protocol (LDAP) Turn Operation
- Juin 2006 : **RFC 4530** Lightweight Directory Access Protocol (LDAP) entryUUID Operational Attribute

R.F.C. LDAP

- Juin 2006 : **RFC 4529**	Requesting Attributes by Object Class in the Lightweight Directory Access Protocol
- Juin 2006 : **RFC 4528**	Lightweight Directory Access Protocol (LDAP) Assertion Control
- Juin 2006 : **RFC 4527**	Lightweight Directory Access Protocol (LDAP) Read Entry Controls
- Juin 2006 : **RFC 4526**	Lightweight Directory Access Protocol (LDAP) Absolute True and False Filters
- Juin 2006 : **RFC 4525**	Lightweight Directory Access Protocol (LDAP) Modify-Increment Extension
- Juin 2006 : **RFC 4524**	COSINE LDAP/X.500 Schema
- Juin 2006 : **RFC 4523**	Lightweight Directory Access Protocol (LDAP) Schema Definitions for X.509 Certificates
- Juin 2006 : **RFC 4522**	Lightweight Directory Access Protocol (LDAP): The Binary Encoding Option
- Juin 2006 : **RFC 4521**	Considerations for Lightweight Directory Access Protocol (LDAP) Extensions
- Juin 2006 : **RFC 4520**	Internet Assigned Numbers Authority (IANA) Considerations for the Lightweight Directory Access Protocol (LDAP)
- Juin 2006 : **RFC 4519**	Lightweight Directory Access Protocol (LDAP): Schema for User Applications
- Juin 2006 : **RFC 4518**	Lightweight Directory Access Protocol (LDAP): Internationalized String Preparation
- Juin 2006 : **RFC 4517**	Lightweight Directory Access Protocol (LDAP): Syntaxes and Matching Rules

R.F.C. LDAP

- Juin 2006 : **RFC 4516** Lightweight Directory Access Protocol (LDAP): Uniform Resource Locator
- Juin 2006 : **RFC 4515** Lightweight Directory Access Protocol (LDAP): String Representation of Search Filters
- Juin 2006 : **RFC 4514** Lightweight Directory Access Protocol (LDAP): String Representation of Distinguished Names
- Juin 2006 : **RFC 4513** Lightweight Directory Access Protocol (LDAP): Authentication Methods and Security Mechanisms
- Juin 2006 : **RFC 4512** Lightweight Directory Access Protocol (LDAP): Directory Information Models
- Juin 2006 : **RFC 4511** Lightweight Directory Access Protocol (LDAP): The Protocol
- Juin 2006 : **RFC 4510** Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map
- Février2006 : **RFC 4403** Lightweight Directory Access Protocol (LDAP) Schema for Universal Description, Discovery, and Integration version 3 (UDDIv3)
- Février 2006 : **RFC 4370** Lightweight Directory Access Protocol (LDAP) Proxied Authorization Control
- Janvier 2006 : **RFC 4373** Lightweight Directory Access Protocol (LDAP) Bulk Update/Replication Protocol (LBURP)

...

Normes et standards en vigueur

- la norme civile internationale X500 issue de l'ISO/IEC ***[1]*** ,
- le standard civil international LDAP ***[2]*** issu de l'IETF ***[3]*** ,
- le standard civil français MAIA ***[4]*** issu du Gouvernement ,

[1] ISO/IEC : International Organization for Standardization
[2] LDAP : Lightweight Directory Access Protocol
[3] IETF : Internet Engineering Task Force
[4] MAIA : Meta Annuaire Inter Administration

La norme X500

La première version de cette norme est apparue en 1988.

Revue en 1993, elle est aujourd'hui dans sa troisième version
(1998).

Elle décrit les items suivants :

- X500 : Présentation de la norme

- X501 : Définition des modèles. Cette partie définit sans entrer dans le détail les grands concepts de l'annuaire (domaines d'administration, les objets, les attributs, les contrôles d'accès, l'administration du service, ...). Il permet d'appréhender au mieux la technologie X500.

La norme X500

- X509 : Définition de l'infrastructure d'authentification. Ce document définit la notion de certificat, d'autorité de certification, les listes de révocation, la vérification du chemin de certification.

- X519 : Définition des protocoles de communication.

- X520 : Définition des attributs. Cette partie définit finement les attributs, les syntaxes et les règles de comparaison.

- X521 : Définition des objets. Cette partie décrit les objets dans le service d'annuaire. Elle introduit la notion de règles de recherche, de structure.

La norme X500

- X525 : Définition du modèle de réplication. Cette partie définit le modèle de réplication du service d'annuaire

Ce qui faut réellement penser de la norme X500

La mise en place de la norme X500 apparait comme un modèle de réussite dans la création d'un standard réalisé par des acteurs divers. En effet le critère d'indépendance vis à vis des éditeurs et le critère d'interopérabilité ont été respectés.

Mais rapidement, cette norme a été jugé trop riche et trop complexe à mettre en oeuvre. Des problèmes de performance sont constatés à cause de la modélisation OSI des protocoles réseau. Il apparait que la norme a été décidée et imposée sans tenir compte de la réalité et des besoins du terrain : le déploiement des annuaires X500 a été réalisé selon une démarche inverse au déploiement réussi d'internet. Aussi, seuls les grands organismes publics ont déployé de tels annuaires.

La norme LDAP

Le protocole LDAP est largement diffusé et s'impose comme protocole d'accès aux services d'annuaire. Il sert également de référence dans les nouvelles architectures de systèmes d'information tels que les concepts de méta-annuaire ou de DEN (Directory Enabled Network).

Ce protocole, conçu pour fournir un accès à un service d'annuaire, est vu comme étant un allègement et une adaptation au monde Internet du protocole d'accès à l'annuaire définit dans la norme X.500.

Ainsi, dans sa troisième version, LDAP fournit des services de lecture, écriture et mise à jour. Il propose donc des applications de gestion et de visualisation des informations contenues dans l'annuaire.

La norme LDAP

LDAP V3 définit quatre modèles qui sont les suivants :

- Le ***modèle d'information*** qui reprend en autre le schéma, les OID, les attributs, les classes d'objets,

- Le ***modèle de désignation*** qui s'attache au nommage des entrées, à la hiérarchisation des données, au lien DNS-LDAP,

- Le ***modèle des services*** qui décrit les services offerts par un service d'annuaire LDAP,

- Le ***modèle de sécurité*** qui explore l'authentification, la confidentialité, l'intégrité et la gestion des habilitations.

Le modèle de l'information

Le modèle d’information décrit essentiellement le schéma de l’annuaire :

- Les attributs
- Les syntaxes
- Les classes d'objet
- Les règles de recherche

Les attributs

Les attributs sont les éléments de base du schéma. Ils sont décrits de la manière suivante :

AttributeTypeDescription = "("
numericoid : l'identifiant unique de l'attribut
["NAME"]: le nom de l'attribut
["DESC"] : description
["OBSOLETE"] indication si l'attribut est obsolète
["SUP"] : les classes parentes dont l'attribut hérite
["EQUALITY"] : le nom de la règle de recherche

Les attributs

["ORDERING"] : le nom de la règle d'ordonnancement

["SUBSTR"] : le nom de la règle de recherche dans une sous chaîne de caractère

["SYNTAX"] : la syntaxe de l'attribut

["SINGLE-VALUE"] : un indicateur si une seule valeur est autorisée ou plusieurs

["COLLECTIVE"] : un indicateur si l'attribut est collectif ou non

["NO-USER-MODIFICATION"] : un indicateur si l'attribut ne doit pas être modifié par un utilisateur

Les attributs

["USAGE"] : l’utilisation de l’attribut

")"

Les attributs

```
attributetype ( 2.5.4.16 NAME 'postalAddress'
        DESC 'RFC2256: postal address'
        EQUALITY caseIgnoreListMatch
        SUBSTR caseIgnoreListSubstringsMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.41 )
```

Les attributs

On distingue habituellement deux types d'attributs :

- Les attributs utilisateurs (user attributes) sont les attributs caractérisant l'objet manipulé par les utilisateurs de l'annuaire (nom, prénom, ...)

- Les attributs opérationnels (system attributes) sont des attributs auxquels seul le serveur peut accèder afin de manipuler les données de l'annuaire (dates de modification, ...)

modifiersName : Contient le DN de l'objet utilisé pour s'identifier lors de la modification. Il est présent dans tous les objets modifiés par la commande modify. Son occurrence est unique et il ne peut être modifié par un utilisateur.

OID : Object identifiers

Les OID sont des identifiants universels, représentés sous la forme d'une suite d'entiers. Ils sont organisés sous forme hiérarchique. Ainsi seul l'organisme 1.2.3 peut dire quel est la signifiacation de l'OID 1.2.3.4. Ils ont été définis dans une recommandation de l'International Telecommunication Union. L'IETF a proposé de représenter la suite d'entiers constituant les OID séparés par des points.

L'objectif des OID est d'assure l'interopérabilité entre différents logiciels. Les OID sont utilisés dans le monde LDAP mais aussi dans d'autres domaines, comme par exemple les logiciels SNMP pour identifier des ressources.

OID : Object identifiers

2.5 - fait référence au service X500
2.5.4 - est la définition des types d'attributs
2.5.6 - est la définition des classes d'objets
1.3.6.1.4.1.4203 - OpenLDAP

Les syntaxes

La syntaxe permet d'indiquer quelle règle il faut suivre pour renseigner l'attribut. Par exemple, si l'attribut suit une syntaxe de type DN et que l'on saisit une chaîne de caractère lambda, il ne pourra pas être renseigné. Ci-dessous, quelques exemples de syntaxes :

Certificate
1.3.6.1.4.1.1466.115.121.1.8

Certificate List
1.3.6.1.4.1.1466.115.121.1.9

Certificate Pair
1.3.6.1.4.1.1466.115.121.1.10

Country String
1.3.6.1.4.1.1466.115.121.1.11

DN
1.3.6.1.4.1.1466.115.121.1.12

Les classes d'objet

Les objets représentent une collection d'attributs. Ils sont représentés de la manière suivante :

ObjectClassDescription = "("
numericoid : identifiant unique de l'objet

["NAME"] : nom de l'objet

["DESC"] : description

["OBSOLETE"] ["OBSOLETE"] : indique si la classe d'objet est obsolète

["SUP"] ["SUP"] : les classes parentes dont la classe d'objet hérite

Les classes d'objet

[("ABSTRACT" / : type de classe d’objet
"STRUCTURAL" /
"AUXILIARY")]
["MUST"] : les attributs obligatoires
["MAY"] : les attributs optionnels
")"

Les classes d'objet

(2.5.6.6
NAME 'person'
SUP top
STRUCTURAL
MUST (sn $ cn)
MAY (userPassword $ telephoneNumber $ seeAlso $ description))

Les classes d'objet

(2.5.6.7
NAME 'organizationalPerson'
SUP person
STRUCTURAL
MAY (title $ x121Address $ registeredAddress $
destinationIndicator $ preferredDeliveryMethod $
telexNumber $ teletexTerminalIdentifier $
telephoneNumber $ internationaliSDNNumber $
facsimileTelephoneNumber $ street $ postOfficeBox $
postalCode $ postalAddress $
physicalDeliveryOfficeName $ ou $ st $ l))

Les classes d'objet

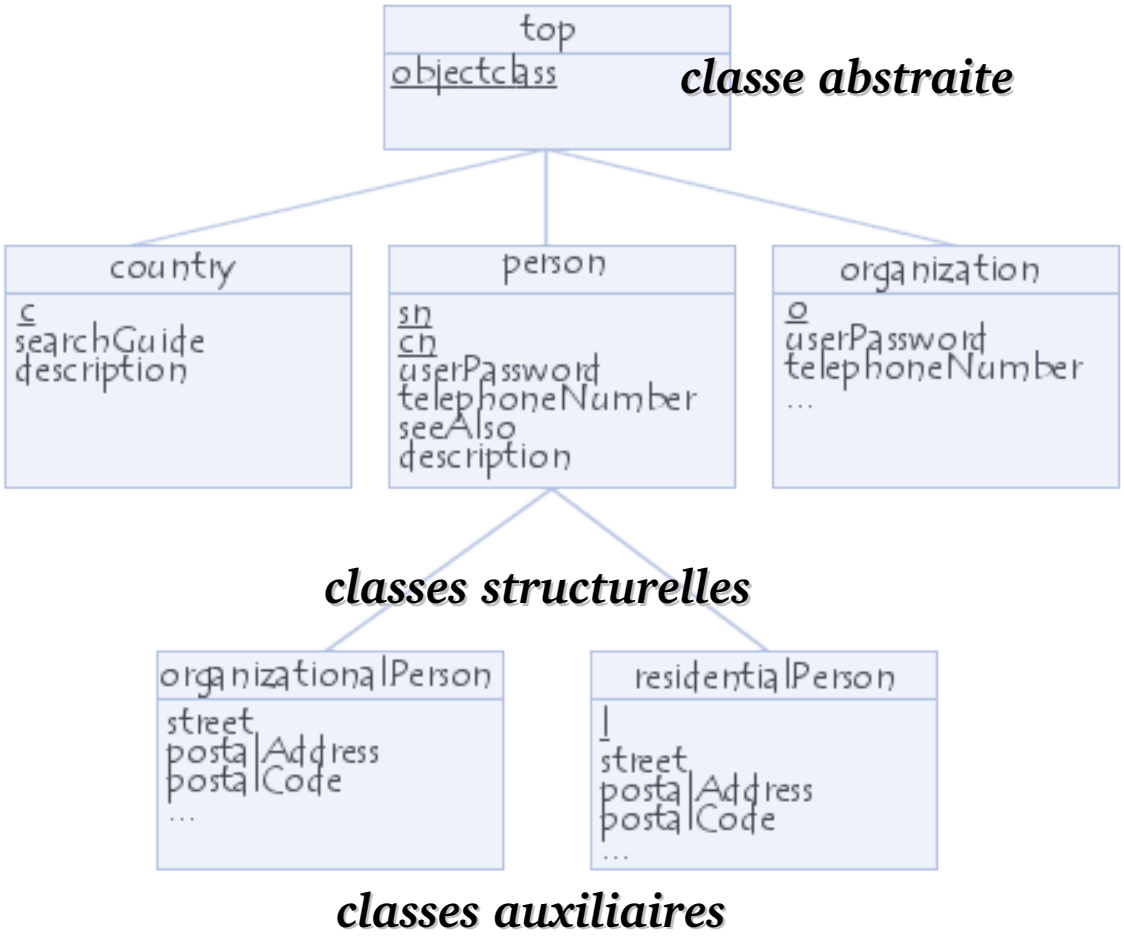
Il existe trois types de classes d'objet :

* Les ***classes abstraites*** sont des classes non instanciables. Il s'agit de classes pouvant être dérivées, c'est-à-dire dont d'autres classes peuvent hériter. La classe d'objet de plus haut niveau étant la classe top dont toute classe d'objet dérive

* Les ***classes structurelles*** sont des classes instanciables. Il est donc possible d'avoir des objets

* Les ***classes auxiliaires*** sont des classes permettant d'ajouter des attributs facultatifs à des classes structurelles.

Les classes d'objet



Les règles de recherche

Les règles de recherche permettent aux serveurs de comparer les valeurs des attributs avec celles demandées dans la requête.

L'exemple ci-dessous présente une règle de recherche sur un entier :

(2.5.13.14 NAME 'integerMatch' SYNTAX 1.3.6.1.4.1.1466.115.121.1.27)

Le modèle de désignation

L'espace de noms homogène

Afin de pouvoir identifier tous les objets dans l'annuaire, il est important de définir les règles de nommage qui seront communes à tout le service d'annuaire mis en place.

Deux notions importantes apparaissent :

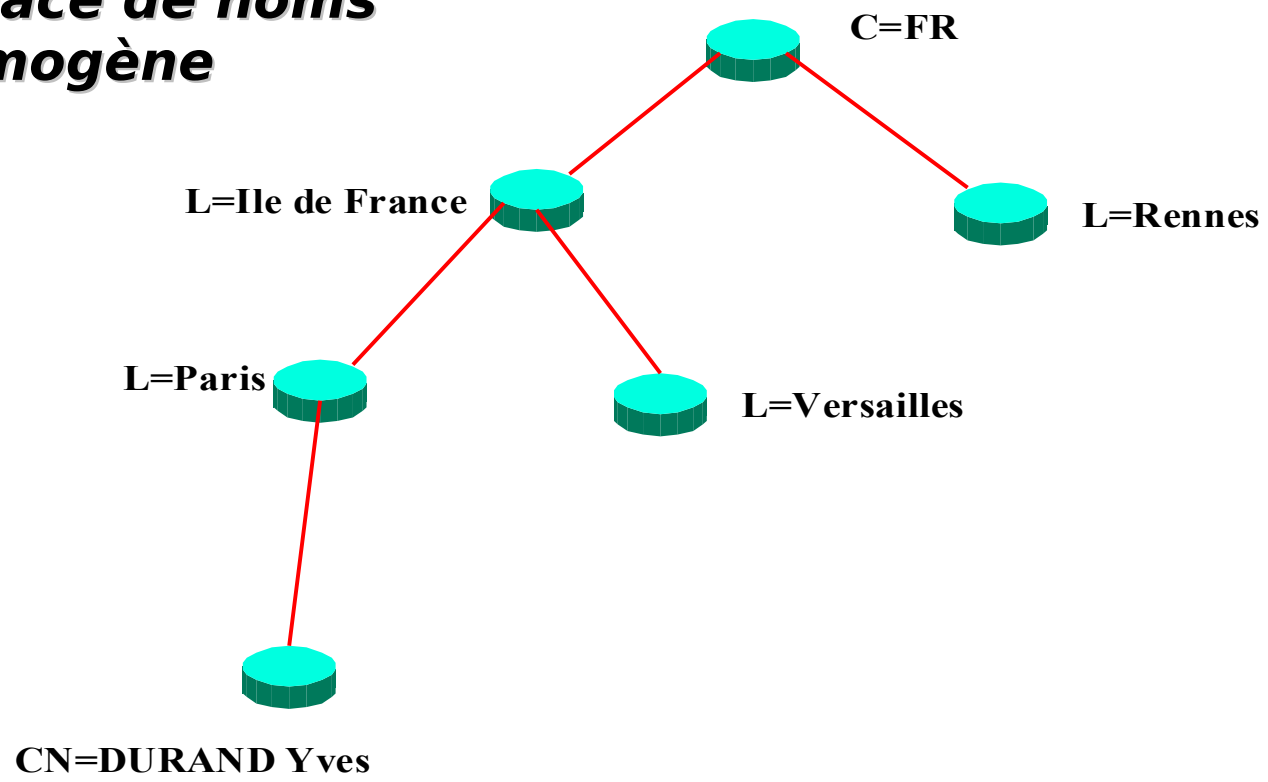
- Le RDN de l'objet : nom unique seulement dans le niveau considéré. (Relative Distinguished Name)

- Le DN de l'objet : nom unique qui permet d'identifier un objet. Le DN est la somme de tous les RDNs.

Ainsi pour avoir un nommage cohérent, il faut s'attacher à définir des règles de nommage pour établir l'unicité de tous les RDNs. Ceci permet de résoudre les problèmes d'homonymie.

Le modèle de désignation

L'espace de noms homogène



Le modèle de désignation

L'espace de noms homogène

RDN = CN=DURAND Yves

CN=DURAND Yves,L=Paris,L=Ile de France,C=fr

L'organisation hiérarchique des données

Un annuaire se présente sous la forme d'arbre (DIT). Il est possible de constituer un arbre de plusieurs manières :

L'arbre peut respecter l'organisation hiérarchique de l'entreprise. Cette technique permet de refléter l'organisation de l' entreprise et de distribuer les données sur les serveurs de manière relativement simple. L'arbre peut respecter l'organisation géographique de l'entreprise. Cette technique permet de concentrer l'administration du service par zone géographique sans se soucier de l'organisation de l'entreprise.

Remarque : les deux premiers modèles peuvent être mixés.

L'organisation hiérarchique des données

L’arbre peut respecter une organisation de l’entreprise, basée sur des critères sémantiques. Cette technique permet de grouper les données qui ont un même sens (Personnes, Applications, Groupes, …). Ceci est représenté dans la figure ci-dessous.

Cette configuration de l’arbre facilite les droits d’accès aux données. Par exemple, la responsable des ressources humaines n’a accès en écriture que sur « OU=Personnes ».

Le modèle des services

Le protocole LDAP est basé sur un modèle client/serveur. Le client transmet une requête décrivant une ou plusieurs opérations à effectuer dans l'annuaire. Le serveur, quant à lui, est responsable de la réalisation de ces opérations dans l'annuaire. Cette responsabilité reste néanmoins limité dans le cas d'un renvoi de référence qui signifie que le serveur est incapable de répondre à la demande du client.

Ainsi, le serveur renvoie au client une réponse contenant soit les résultats demandés, soit une ou plusieurs références à d'autres serveurs susceptibles d'accéder à la demande du client.

Le modèle des services

Toutes les opérations sont encapsulées dans une enveloppe commune : le message LDAP (LDAPMessage).

Chaque requête possède un identifiant. Pour pouvoir le réutiliser, il ne faut pas que :
- La requête ait été abandonnée,
- La requête n'ait pas eu de réponse à l'instant donné.

Quelle que soit la requête, la réponse sera retournée au client sous la forme d'un message LDAPResult.

Le modèle de sécurité

L'accès au service d'annuaire peut se faire par :

- accès anonyme,
- authentification par mot de passe,
- authentification par mot de passe dans une liaison TLS chiffrée,
- authentification forte par certificat,
- authentification forte avec TLS.

Le format LDIF

Le format LDIF est un format de fichier ASCII qui est utilisé pour échanger des données entre DSA. A l'heure actuelle, ce format de fichier pivot est utilisé par tous les constructeurs et permet une interopérabilité entre les produits.

Le format LDIF est utilisé pour faciliter les opérations d'importations et d'exportations massives d'informations d'un annuaire. Un fichier LDIF consiste en une série d'enregistrements séparés par des séparateurs de lignes. Un enregistrement consiste en une séquence de lignes décrivant une entrée de l'annuaire ou, en une séquence de lignes décrivant une série de modifications à effectuer sur une entrée. Un fichier LDIF spécifie soit des entrées, soit des modifications à faire sur des entrées, mais pas les deux.

Le format LDIF

dn: dmdName=Devices,ou=iut_mont-de-marsan, dc=universite, dc=education,dc=gouv,dc=fr

dmdName: Devices

objectClass: dmd

objectClass: top

dn: dmdName=Applications,ou=iut_mont-de-marsan,dc=univer-site,dc=education,dc=gouv,dc=fr

dmdName: Applications

objectClass: dmd

objectClass: top

CHAPITRE 3 :

POLITIQUE ET SCHEMAS D'ANNUAIRE

Le schéma d'annuaire

Le schéma d’un annuaire est un ensemble de définitions et de contraintes concernant la structure du DIT, les façons possibles de nommer les entrées, l’information qui peut être maintenue dans une entrée, les attributs utilisés pour représenter une information et son organisation dans la hiérarchie pour faciliter la recherche et l’ouverture de l’information.

Le schéma d'annuaire

Ainsi, il est possible d'imaginer les branches suivantes :

- Les personnels humains
- Les fonctions
- Les groupes
- Les services
- Les équipements informatiques
- Les applications logicielles
- Les documents
- L’organisation

CHAPITRE 4 :

ARCHITECTURE

La réplication

La réplication est l'opération qui consiste à créer des copies de l'information stockée dans un premier serveur dit **maître** et de la stocker dans un ou plusieurs autres serveurs dits **esclaves**.

D'un point de vue X500 :
Cette technique est normalisée au sein de l'OSI avec le protocole DISP.
La réplication peut être de deux types :
- **primaire** : elle met en jeu un serveur maître et un ou plusieurs serveurs esclaves,
- **secondaire** : elle met en jeu un serveur maître et un ou plusieurs serveurs esclaves, dont l'un est aussi maître pour un autre serveur esclave.

La réplication

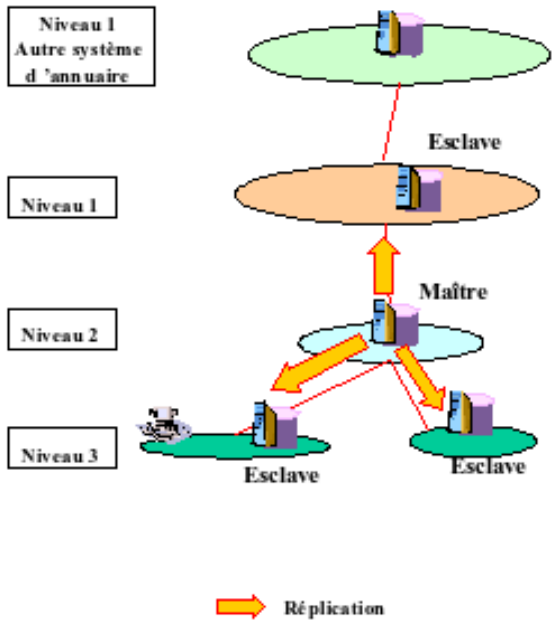
Dans ces deux types de réplication, seules les requêtes de lecture, comparaison et recherche sont permises sur les esclaves. Les requêtes de mise à jour s'effectuent sur le serveur Maître.

D'un point de vue LDAP :
Seule la réplication de type primaire est admise.

Le modèle Maître / Esclave

Le modèle Maître-Esclave consiste à avoir un maître et plusieurs esclaves.

Cette technique peut être utilisée lorsque l'administration des données (de niveaux 2 et 3) est centralisée en un point unique comme le montre le schéma ci-dessous : là, l'administration est centralisée au niveau 2.



Le modèle Maître / Esclave

Le serveur maître se situe au niveau 2 afin de permettre une réplication primaire à la fois vers le niveau 1 et vers le niveau 3.

Le modèle Maître / Esclave

Cette solution a pour avantages :

- d'être simple à mettre en œuvre puisqu'un seul serveur est dédié aux mises à jour,
- d'optimiser les performances d'accès à l'annuaire dans la mesure où les serveurs esclaves sont dédiés à la lecture,
- de permettre le travail en local même si le réseau ne fonctionne plus,
- de permettre la communication entre niveaux 3 même si le service d'annuaire de niveau 2 ne fonctionne plus,
- d'être généralement supportée par la majorité des serveurs d'annuaires du marché,

Le modèle Maître / Esclave

Cette solution a pour inconvénients :

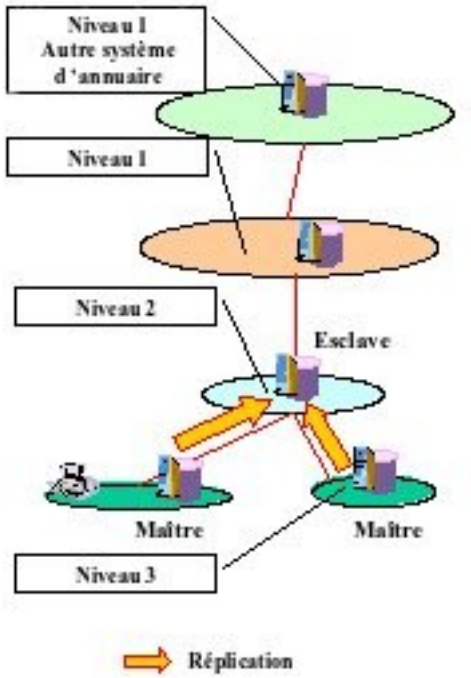
- de ne plus permettre les mises à jour si le serveur maître tombe en panne,
- de ne pas permettre les mises à jour à tout moment si le serveur maître est situé sur un site distant qui n'est pas relié en permanence avec les clients (postes ou applications),

Le modèle MultiMaître / Esclave

Le modèle Multimaître-Esclave consiste à fournir les informations des serveurs maîtres vers un seul serveur esclave ; il s'appuie sur la réplication de type primaire, citée précédemment.

Les serveurs maîtres se situent au niveau 3 et mettent à jour le niveau 2 comme le montre le schéma ci-dessous.

Ce modèle peut être utilisé dans le cas où les produits d'annuaire utilisés sont identiques.



Le modèle MultiMaître / Esclave

Cette solution a pour avantages :

- de positionner les données gérées au plus près des utilisateurs,
- de continuer à pouvoir accéder en lecture aux données des niveaux 3 par l'intermédiaire du niveau 2 si le serveur d'annuaire de niveau 3 ne fonctionne plus,
- de permettre à chaque niveau de gérer ses propres données sur son propre serveur,
- de permettre l'accès en lecture, aux données des niveaux 3, au niveau 2 si un des serveurs maître ne fonctionne plus,
- d'être intéressante dans le cas où les produits d'annuaire utilisés sur les différents niveaux sont identiques.

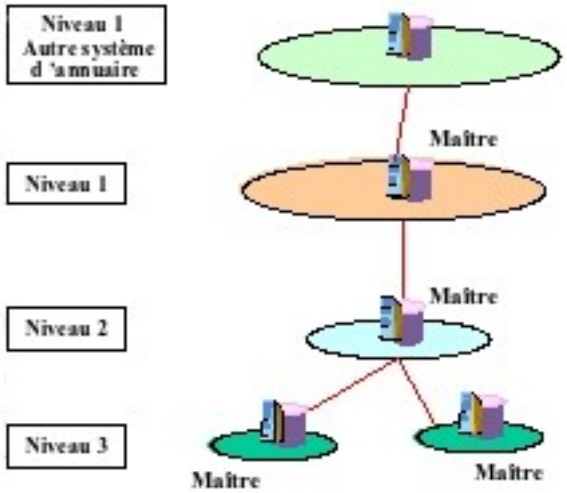
Le modèle MultiMaître / Esclave

Cette solution a pour inconvénients :

- d'être un peu moins simple à mettre en oeuvre puisque deux serveurs sont dédiés aux mises à jour,
- de moins optimiser les performances d'accès à l'annuaire dans la mesure où les serveurs maîtres gèrent à la fois les lectures et les mises à jour.
- de ne plus permettre les mises à jour sur un serveur maître qui tomberait en panne,
- de ne pas être fonctionnelle dans le cas où les produits d'annuaire utilisés sur les différents niveaux sont différents.

Le modèle MultiMaître / Esclave

Le modèle Multimaître consiste à avoir plusieurs maîtres sur lesquels les lectures et les mises à jour peuvent se faire. Il s'appuie au maximum sur les potentialités de la norme LDAP afin de contourner les problèmes des protocoles propriétaires des produits mais il n'est pas normalisé.
Ce modèle est intéressant dans le cas où des produits d'annuaire différents sont utilisés car dans ce cas, les solutions vues précédemment ne fonctionnent pas. Les serveurs maîtres se situent à tous les niveaux, comme le montre le schéma ci-dessous.



Le modèle MultiMaître / Esclave

Cette solution a pour avantages :

- de positionner les données gérées au plus près des utilisateurs,
- de permettre à chaque niveau de gérer ses propres données sur son propre serveur,
- d'être la solution qui se rapproche le plus de la cible visée dans ce document.

Le modèle MultiMaître

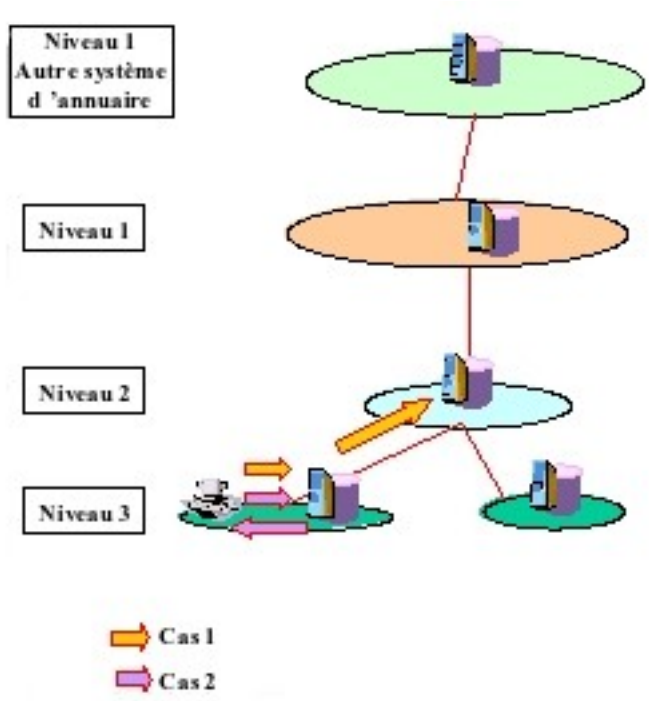
Cette solution a pour inconvénients :

- d'être complexe à mettre en oeuvre puisque tous les serveurs sont maîtres : il est nécessaire de construire le système de réplication entre les serveurs par l'intermédiaire d'un serveur tiers.
- de devoir être bien maîtrisée, ce qui implique de bien positionner les contrôles d'accès et de mettre en place une politique de sécurité homogène au sein du service d'annuaire global.

Le modèle Renvoi de Référence (Referral)

Le service d'annuaire est capable de renvoyer à l'utilisateur une référence pour indiquer où se trouve la donnée recherchée.

Dans ce cas de figure, l'annuaire n'est plus répliqué mais réparti sur les différents annuaires de niveau.



Le modèle Renvoi de Référence (Referral)

Un utilisateur du niveau 3 interroge son serveur d'annuaire. Deux cas se présentent suivant la technologie utilisée :

- Cas 1 (X500) : si le serveur d'annuaire interrogé (de niveau 3) ne possède pas la donnée, le serveur fait une demande au niveau supérieur (niveau 2) pour avoir la référence du serveur qui détient l'information. Le serveur se connecte ensuite au serveur qui détient l'information.
- Cas 2 (LDAP) : si le serveur d'annuaire interrogé (de niveau 3) ne possède pas la donnée, le serveur renvoie une référence au client. Le client se connecte ensuite au serveur qui possède l'information. Ce mécanisme peut être automatique ou non suivant les produits clients.

Le modèle Renvoi de Référence (Referral)

Cette solution a pour avantages :

- de positionner les données gérées au plus près des utilisateurs,
- de permettre à chaque niveau de gérer ses propres données sur son propre serveur,
- d'être généralement supportée par la majorité des serveurs d'annuaires du marché.

Le modèle Renvoi de Référence (Referral)

Cette solution a pour inconvénients :

- de ne plus permettre les consultations et les mises à jour d'un niveau si le serveur de ce niveau ne fonctionne plus,
- de moins optimiser les performances d'accès à l'annuaire dans la mesure où chaque serveur gère à la fois les lectures et les mises à jour et dans le cas où, pour trouver une information, plusieurs serveurs peuvent devoir être consultés.

CHAPITRE 5 :

MISE EN OEUVRE D'UN ANNUAIRE OPENLDAP

OpenLDAP

Il existe de nombreux serveurs LDAP qui sont en général payant. OpenLDAP (téléchargeable depuis le site http://www.openldap.org) est un projet libre de serveur d'annuaire conforme à la norme LDAP 3. Ce serveur, dérivé de l'implémentation mise au point par l'université de Michigan, est développé selon les termes de licence GNU GPL, ce qui signifie qu'il est entièrement gratuit et que les sources de ce logiciel sont disponibles.

OpenLDAP est composé des éléments suivants :

- le serveur LDAP : slapd
- La passerelle LDAP vers X500 : ldapd
- Des outils d'administrations

Version

- OPENLDAP 2.4.23 (Version stable)

- DB 4.8.24 (base de données Berkeley)

- Apache Directory Studio 1.5.3 (Client ldap)

Ce que l'on peut faire avec un annuaire OpenLDAP

- Authentification des utilisateurs à travers le module PAM LDAP (Pluggable Authentication Modules)

- Gestion du D.H.C.P., du D.N.S., ... par l'annuaire Ldap

- Couplage de la messagerie à l'annuaire (gestion des listes de diffusion, liaison personne/fonction ...)

- Stockage des certificats et des listes de révocation de certificats)

- Couplage de Samba à l'annuaire

Base de données Berkeley

Berkeley DB est une base de données embarquée, ce qui rend son déploiement et son administration aisés. Elle est utilisée dans de nombreux environnement, sous Unix, GNU/Linux, Microsoft Windows et dans des systèmes embarqués. Son éditeur revendique 200 millions de déploiements.

Depuis la version 2.0, Berkeley DB est disponible sous deux licences, une libre certifiée par l'OSI (Open Source Initiative) et une licence commerciale. Les versions précédentes étaient sous licence BSD (La licence BSD, Berkeley software distribution license est une licence libre utilisée pour la distribution de logiciels.
Berkeley DB est développée par Sleepycat Software qui a été racheté en février 2006 par Oracle Corporation.

Base de données Berkeley

Ses principales fonctionnalités sont :

* la gestion de transactions
* la possibilité de verrouiller des enregistrements
* une gestion simplifiée des sauvegardes et de la réplication. On peut effectuer des sauvegardes "à chaud", sans arrêter la base.
* la gestion d'un système de cache mémoire interne
* elle supporte de grosse capacité de données (jusqu'à 4 Go par enregistrement et 256 To par base)
* les données peuvent être chiffrées

Installation OpenLDAP

Vous avez le choix entre une installation de type paquet deb (pour ubuntu), facilitée par les paquetages disponible sur les CD d'installation des distributions Linux ou de télécharger une version et de procéder à une installation par tarball (sources).

Installation OpenLDAP

Installation et configuration du serveur

Sur le serveur, les paquets suivants sont nécessaires :

sudo apt-get install slapd ldap-utils

Après avoir installé les paquets, les fichiers suivants sont présents dans / etc / ldap / slapd.d :

/etc/ldap/slapd.d/
/etc/ldap/slapd.d/cn=config
/etc/ldap/slapd.d/cn=config/cn=schema
/
etc/ldap/slapd.d/cn=config/cn=schema/cn={0}core.ldif
/etc/ldap/slapd.d/cn=config/cn=schema.ldif

Installation OpenLDAP

/etc/ldap/slapd.d/cn=config/olcDatabase={-1}frontend.ldif
/etc/ldap/slapd.d/cn=config/olcDatabase={0}config.ldif
/etc/ldap/slapd.d/cn=config.ldif

Les schémas doivent être chargés sur le serveur car, par défaut, ils ne sont pas présents.

sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/cosine.ldif

sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/nis.ldif

Installation OpenLDAP

sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/inetorgperson.ldif

sudo ldapadd -Y EXTERNAL -H ldapi:/// -f /etc/ldap/schema/misc.ldif

Pour créer la base de données réelles qui stocke les entrées dans le répertoire, il faut créer un fichier LDIF.

Dans cet exemple, nous utilisons dc = edu, dc = exemple, dc = org comme suffixe. La base de données est placée sous / var / lib / ldap /.

Installation OpenLDAP

Créer le fichier database.ldif:

```
# Load hdb backend module
dn: cn=module{0},cn=config
objectClass: olcModuleList
cn: module
olcModulepath: /usr/lib/ldap
olcModuleload: {0}back_hdb

# Create the hdb database and place the files under
/var/lib/ldap
dn: olcDatabase={1}hdb,cn=config
objectClass: olcDatabaseConfig
objectClass: olcHdbConfig
olcDatabase: {1}hdb
```

Installation OpenLDAP

olcDbDirectory: /var/lib/ldap
olcSuffix: dc=edu,dc=example,dc=org
olcRootDN:
uid=admin,ou=People,dc=edu,dc=example,dc=org
olcRootPW: example
olcDbConfig: {0}set_cachesize 0 2097152 0
olcDbConfig: {1}set_lk_max_objects 1500
olcDbConfig: {2}set_lk_max_locks 1500
olcDbConfig: {3}set_lk_max_lockers 1500
olcLastMod: TRUE
olcDbCheckpoint: 512 30
olcDbIndex: uid pres,eq
olcDbIndex: cn,sn,mail pres,eq,approx,sub
olcDbIndex: objectClass eq

Installation OpenLDAP

La commande ldapadd est utilisée pour modifier les entrées cn=config :

sudo ldapadd -Y EXTERNAL -H ldapi:/// -f database.ldif

La prochaine étape consiste à créer les entrées suivantes dans l'annuaire : ou=People and ou=Groups. Pour cela, on crée le fichier init_database.ldif.

dn: dc=edu,dc=example,dc=org
objectClass: dcObject
objectclass: organization
o: edu.example.org
dc: edu
description: LDAP root

Installation OpenLDAP

dn: ou=People,dc=edu,dc=example,dc=org
objectClass: top
objectClass: organizationalUnit
ou: People

dn: ou=Groups,dc=edu,dc=example,dc=org
objectClass: top
objectClass: organizationalUnit
ou: Groups

Utiliser la commande ldapadd pour intégrer init_database.ldif:

sudo ldapadd -Y EXTERNAL -H ldapi:/// -f init_database.ldif

Installation OpenLDAP

Enfin, nous modifierons les ACL pour limiter l'accès à la base de données. Ici, nous autorisons l'accès anonyme pour la lecture du répertoire en créant le » fichier acls.ldif :

dn: olcDatabase={1}hdb,cn=config

add: olcAccess

*olcAccess: {0}to attrs=userPassword,shadowLastChange by dn="uid=admin,ou=People,dc=edu,dc=example,dc=org" write by anonymous auth by self write by * none*

*olcAccess: {1}to dn.subtree="" by * read*

*olcAccess: {2}to * by dn="uid=admin,ou=People,dc=edu, dc=example,dc=org" write by * read*

Installation OpenLDAP

Intégrer le fichier :

sudo ldapmodify -Y EXTERNAL -H ldapi:/// -f acls.ldif

Commandes OpenLDAP

Show the current configuration :
sudo ldapsearch -Y EXTERNAL -H ldapi:/// -b cn=config

Show the current data in the directory as anonymous user :
ldapsearch -x -h localhost -b dc=edu,dc=example,dc=org

Dump the database with metadata :
sudo slapcat

L'entrée Root DSE

Dans un annuaire, il existe une entrée spécifique, le “Root DSE” (DSA Specific Entry), qui permet de fournir des informations générales sur l'annuaire.

Cette entrée qui n'a pas de DN, peut être interrogée (requête avec filtre et dont la base_dn n'est pas renseignée).

Elle retournera alors des informations comme par exemple les versions LDAP prises en charge ...

L'entrée Subschema

L'entrée « subschemaSubentry » est une entrée décrivant l'ensemble du schéma d'annuaire utilisé (attribut de la classe d'objet « subschema »).

Celà concerne la description de l'ensemble des syntaxes, des règles de comparaison, des attributs et des classes d'objet.

Les ACcess Lists

```
# Sample access control policy:
#       Root DSE: allow anyone to read it
#       Subschema (sub)entry DSE: allow anyone to read it
#       Other DSEs:
#               Allow self write access
#               Allow authenticated users read access
#               Allow anonymous users to authenticate

#       Directives needed to implement policy:
access to dn.base="" by * read
access to dn.base="cn=Subschema" by * read
access to *
        by self write
        by users read
        by anonymous auth
```

Les ACcess Lists

Les directives access to <quoi> by <qui> <type d'accès> <contrôle> spécifient la configuration du contrôle d'accès à l'annuaire.

<quoi> est un *dn* ou une expression régulière et/ou un/des attribut(s), éventuellement un filtre pour sélectionner certaines valeurs particulières d'attributs. Après avoir défini à quelles entrées s'applique cette règle (à quels objets), on décrit qui est concerné par cette règle et enfin le droit qui est attribué à ces entrées pour ces personnes.

Le **<qui>** peut désigner aussi bien les connexions anonymes (*anonymous*), les utilisateurs authentifiés.

Les ACcess Lists

Le <type d'accès>: dans l'ordre d'importance croissant : *none*, *auth*, *compare*, *search*, *read*, *write*. Chaque niveau implique tous les précédents.

Les <contrôle> sont: *stop*, *continue* ou *break* mais optionnels.

Lors de la liaison avec le serveur LDAP, il va y avoir une vérification des droits de la personne voulant se connecter. Le serveur va chercher la première règle qui s'applique à l'entrée en cours d'inspection (pour modification, suppression, lecture ou recherche).

Les ACcess Lists

S'il n'y a pas d'ACL, l'ensemble des données pourra être lue par tout le monde mais les mises à jour ne seront possibles que par le rootdn.
("access to * by * read")

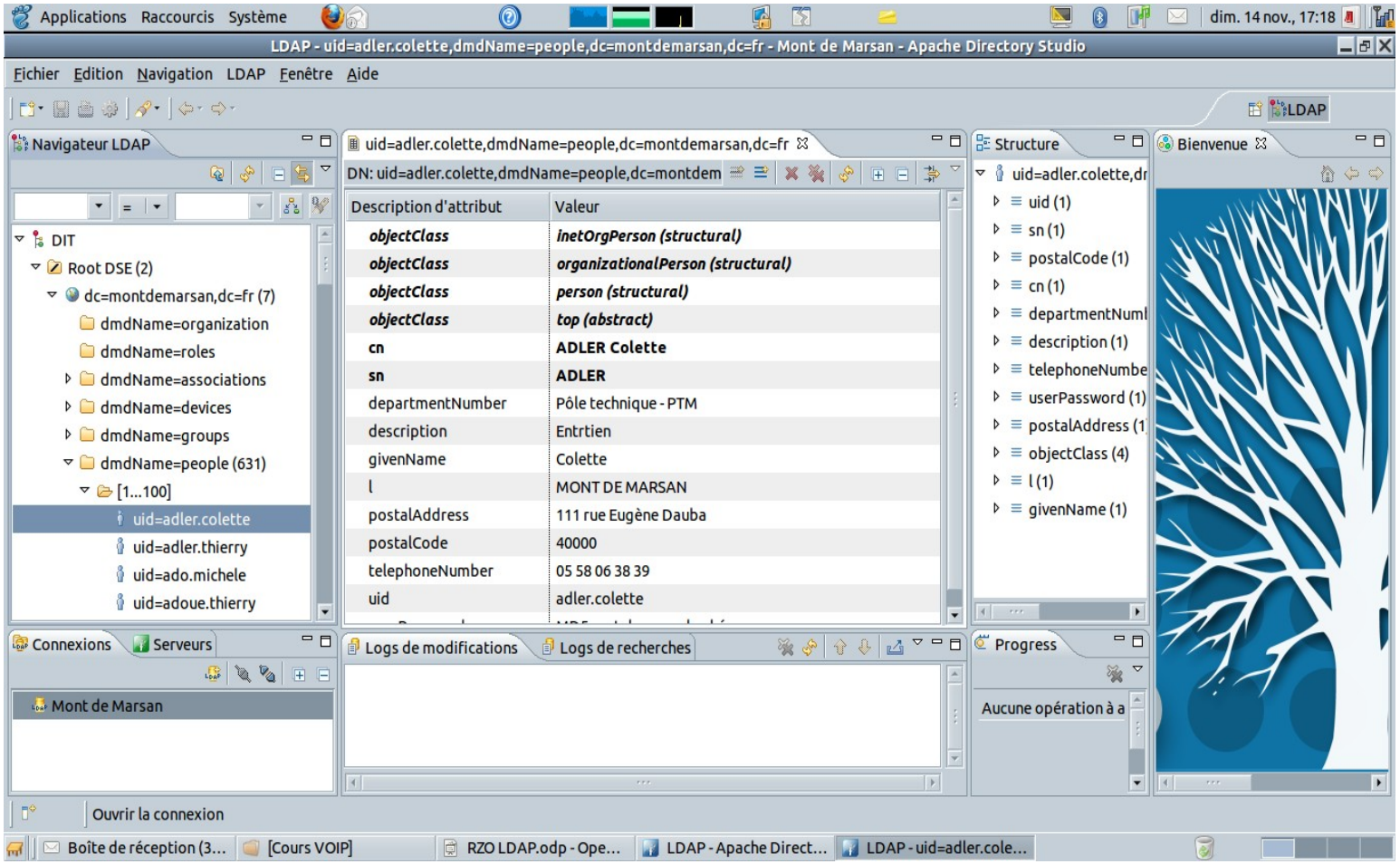
Sinon, c'est la première règle qui comprend l'entrée sur laquelle l'opération se fait qui est retenue. Une fois que la règle est trouvée, c'est dans celle-ci que l'on cherche quels sont les droits d'accès et c'est la première règle qui s'applique.

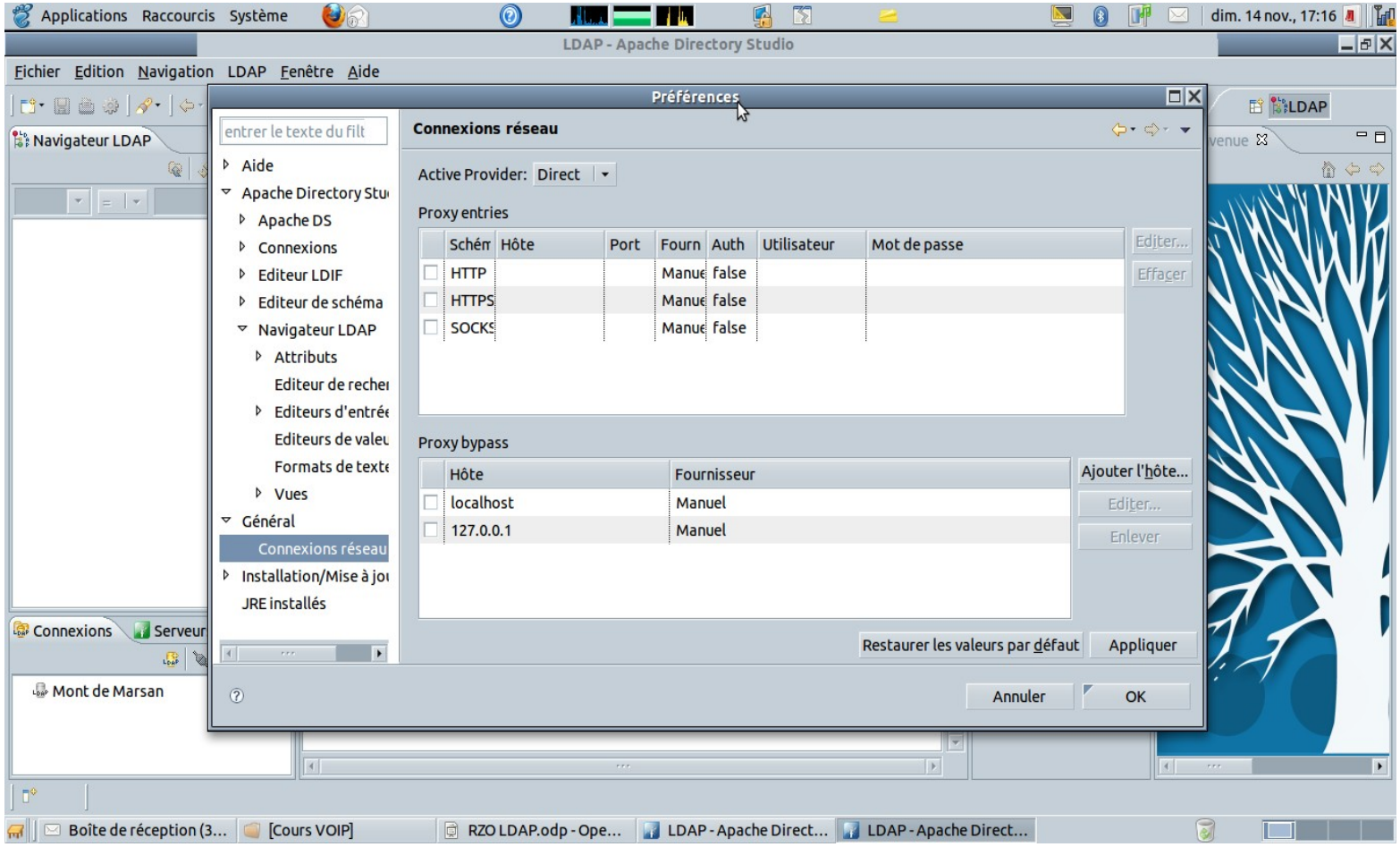
Attention, il faut commencer par les règles les plus fines avant de passer aux plus générales.

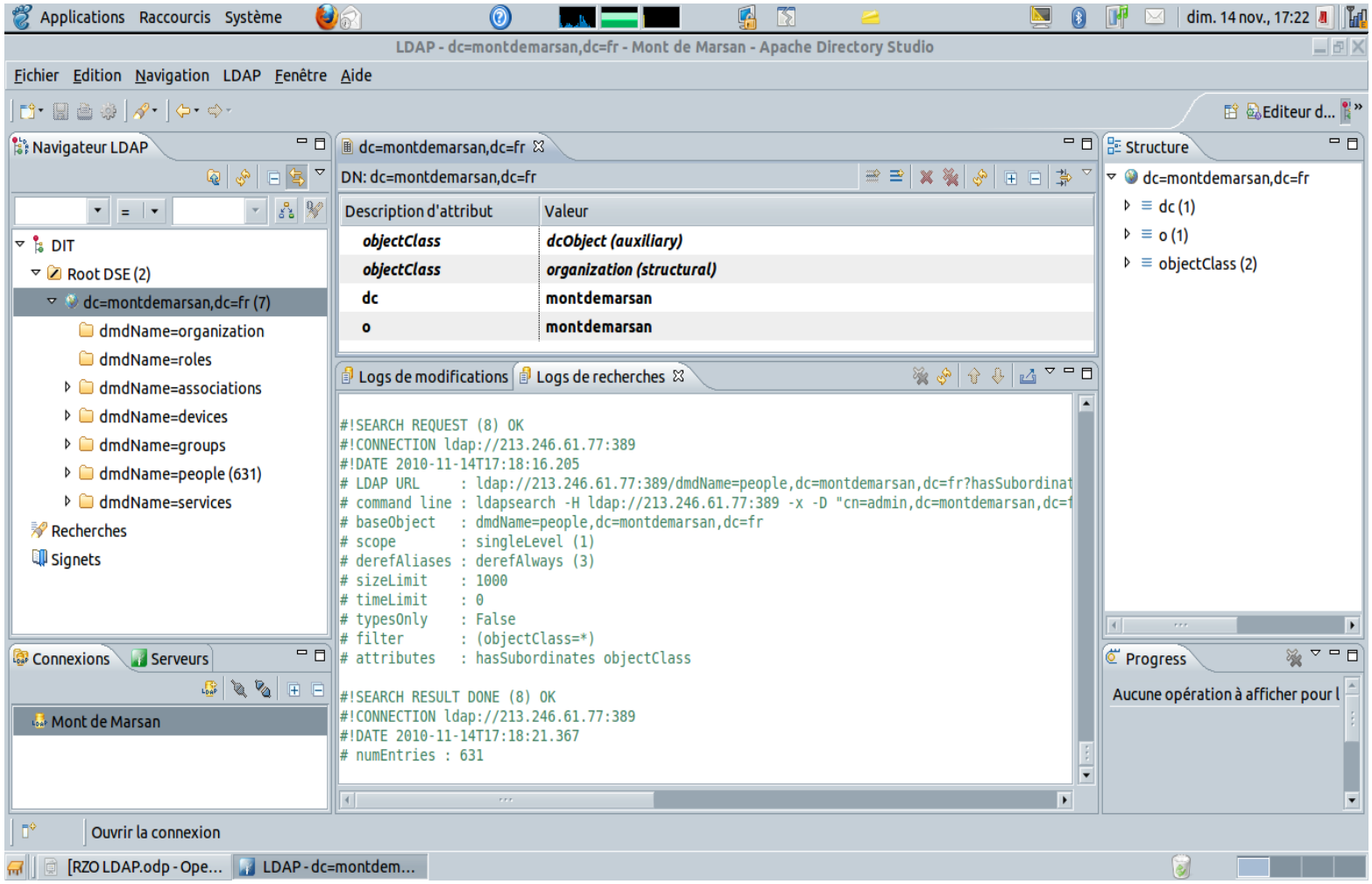
Apache Directory Studio

Apache Directory est un projet libre de la fondation Apache. Son principal composant, Apache Directory Server, est un serveur d'annuaire LDAP « embarquable » écrit en Java. Il a été certifié compatible LDAPv3 par l'Open Group en 2006. Au-delà de LDAP, le serveur prend également en charge d'autres protocoles, et inclut un server Kerberos.

Il existe un sous-projet proposant un outil, basé sur Eclipse, pour gérer un annuaire : Apache Directory Studio. Celui-ci inclut un navigateur et éditeur LD navigateur de schéma, un éditeur LDIF et DSML encore.


Applications Raccourcis Système
dim. 14 nov., 17:18
LDAP - uid=adler.colette,dmdName=people,dc=montdemarsan,dc=fr - Mont de Marsan - Apache Directory Studio
Fichier Edition Navigation LDAP Fenêtre Aide
LDAP
Navigateur LDAP
DIT
Root DSE (2)
dc=montdemarsan,dc=fr (7)
dmdName=organization
dmdName=roles
dmdName=associations
dmdName=devices
dmdName=groups
dmdName=people (631)
[1...100]
uid=adler.colette
uid=adler.thierry
uid=ado.michele
uid=adoue.thierry
uid=adler.colette,dmdName=people,dc=montdemarsan,dc=fr
DN: uid=adler.colette,dmdName=people,dc=montdem
Description d'attribut
Valeur
objectClass
inetOrgPerson (structural)
objectClass
organizationalPerson (structural)
objectClass
person (structural)
objectClass
top (abstract)
cn
ADLER Colette
sn
ADLER
departmentNumber
Pôle technique - PTM
description
Entrtien
givenName
Colette
l
MONT DE MARSAN
postalAddress
111 rue Eugène Dauba
postalCode
40000
telephoneNumber
05 58 06 38 39
uid
adler.colette
Structure
uid=adler.colette,dr
uid (1)
sn (1)
postalCode (1)
cn (1)
departmentNuml
description (1)
telephoneNumbe
userPassword (1)
postalAddress (1
objectClass (4)
l (1)
givenName (1)
Bienvenue
Connexions
Serveurs
Mont de Marsan
Logs de modifications
Logs de recherches
Progress
Aucune opération à a
Ouvrir la connexion
Boîte de réception (3...
[Cours VOIP]
RZO LDAP.odp - Ope...
LDAP - Apache Direct...
LDAP - uid=adler.cole...


Applications Raccourcis Système
dim. 14 nov., 17:16
LDAP - Apache Directory Studio
Fichier Edition Navigation LDAP Fenêtre Aide
Préférences
Navigateur LDAP
entrer le texte du filt
Aide
Apache Directory Stu
Apache DS
Connexions
Editeur LDIF
Editeur de schéma
Navigateur LDAP
Attributs
Editeur de recher
Editeurs d'entrée
Editeurs de valeu
Formats de texte
Vues
Général
Connexions réseau
Installation/Mise à jou
JRE installés
Connexions réseau
Active Provider: Direct
Proxy entries
Schém Hôte Port Fourn Auth Utilisateur Mot de passe
HTTP Manue false
HTTPS Manue false
SOCKS Manue false
Editer...
Effacer
Proxy bypass
Hôte Fournisseur
localhost Manuel
127.0.0.1 Manuel
Ajouter l'hôte...
Editer...
Enlever
Restaurer les valeurs par défaut
Appliquer
Annuler
OK
Connexions
Serveur
Mont de Marsan
LDAP
Boîte de réception (3...
[Cours VOIP]
RZO LDAP.odp - Ope...
LDAP - Apache Direct...
LDAP - Apache Direct...


Applications Raccourcis Système
dim. 14 nov., 17:22
LDAP - dc=montdemarsan,dc=fr - Mont de Marsan - Apache Directory Studio
Fichier Edition Navigation LDAP Fenêtre Aide
Editeur d...
Navigateur LDAP
DIT
Root DSE (2)
dc=montdemarsan,dc=fr (7)
dmdName=organization
dmdName=roles
dmdName=associations
dmdName=devices
dmdName=groups
dmdName=people (631)
dmdName=services
Recherches
Signets
dc=montdemarsan,dc=fr
DN: dc=montdemarsan,dc=fr
Description d'attribut
Valeur
objectClass
dcObject (auxiliary)
objectClass
organization (structural)
dc
montdemarsan
o
montdemarsan
Logs de modifications
Logs de recherches
#!SEARCH REQUEST (8) OK
#!CONNECTION ldap://213.246.61.77:389
#!DATE 2010-11-14T17:18:16.205
LDAP URL : ldap://213.246.61.77:389/dmdName=people,dc=montdemarsan,dc=fr?hasSubordinat
command line : ldapsearch -H ldap://213.246.61.77:389 -x -D "cn=admin,dc=montdemarsan,dc=f
baseObject : dmdName=people,dc=montdemarsan,dc=fr
scope : singleLevel (1)
derefAliases : derefAlways (3)
sizeLimit : 1000
timeLimit : 0
typesOnly : False
filter : (objectClass=*)
attributes : hasSubordinates objectClass
#!SEARCH RESULT DONE (8) OK
#!CONNECTION ldap://213.246.61.77:389
#!DATE 2010-11-14T17:18:21.367
numEntries : 631
Structure
dc=montdemarsan,dc=fr
dc (1)
o (1)
objectClass (2)
Connexions
Serveurs
Mont de Marsan
Progress
Aucune opération à afficher pour l
Ouvrir la connexion
[RZO LDAP.odp - Ope...
LDAP - dc=montdem...

D'autres solutions existent

Il est possible d'administrer son annuaire Ldap avec :

- Jxplorer

- Gq

...

Volumétrie

Test N°2 : Saisie du mot de passe Admin samba en ligne de commande

Test N°3 : Inscription d'une machine sur le domaine

Test N°4 : Inscription d'un utilisateur sur le domaine

Test N°5 : Création d'un groupe d'unité et affectation des utilisateurs à celui-ci

Test N°6 : Inscription d'une fonction sur le domaine

Test N°7 : Inscription d'une imprimante réseau sur le domaine

Volumétrie

N°Test		N°essai	Traffic Entrant Fédérateur		Traffic Sortant Fédérateur	
			Packets	Bytes	Packets	Bytes
2		1	15	1665	19	3894
		2	15	1665	19	3894
3	3.1.1	1	44	4555	43	19244
		2	45	4625	43	19244
	3.1.2	1	84	8603	69	11940
		2	80	8056	63	10996
	3.2	1	12	1299	12	2946
		2	12	1457	12	2946
4		1	23	2994	24	3565
		2	22	2140	23	3495
5	5.1	1	190	18718	162	37929
		2	189	18648	156	37539
	5.2	1	129	12511	105	16541
		2	129	12388	102	15597
6		1	22	2138	24	3559
		2	22	2138	24	3559
7		1	0	0	0	0
		2	0	0	0	0

T.P. : Programme

Durée : 3 heures

- **Rédaction d'une politique d'annuaire et construction d'un fichier ldif** : 1 heure

- **Installation d'un serveur OpenLDAP** : 2 heures

Fin

Bibliographie

- http://ldapbook.labs.libre-entreprise.org/ : Michaël Parienti Maire

- Réflexion sur les politique d'annuaire : CELAR

- Ldap Administration système : Gerald Carter – O Reilly

- http://www.commentcamarche.net/

- Encyclopédie Wikipedia