

## Windows 2000

### ACTIVE DIRECTORY ET ADSI

#### UNE EXPÉRIENCE DANS UN DÉPARTEMENT

LAURENT.KLING@epfl.ch, DÉPARTEMENT DE GÉNIE MÉCANIQUE



#### PRÉAMBULE

Une des préoccupations actuelles est la gestion des usagers et des ressources informatiques qui sont liées. La mise en place de méta-annuaire offrant un lieu commun pour accéder aux éléments doit résoudre cette problématique dans un fonctionnement quotidien.

Un des premiers exemples de cette stratégie est celui de Novell qui intégra dans un protocole réseau spécifique une fonctionnalité d'annuaire. Une démarche similaire a été mise en place par Sun avec l'introduction de NIS sous Unix (Solaris) qui est devenu un standard supporté par d'autres fournisseurs. Le prolongement naturel de cette démarche est la re-création d'un méta-annuaire sous le système d'exploitation majoritaire actuel dans la micro-informatique, Windows 2000.

#### PRINCIPES

Je ne désire pas dans cet article expliquer l'ensemble du fonctionnement de Windows 2000 mais mettre en évidence certains éléments clés de sa mise en œuvre. Windows 2000 est, malgré son pa-

tronyme, un système d'exploitation différent de ses prédécesseurs (comme Windows NT 4 ou Windows 98).

Deux éléments clés sont à la base de son fonctionnement:

- Active Directory: annuaire distribué d'information basé sur X500 et LDAP.

- Active DNS: annuaire DNS lié à Active Directory.

De ce fait, il faut changer nos habitudes de travail.

Pour pouvoir approfondir la connaissance de ce système, je recommande de suivre les cours organisés par le SIC et vous propose de prendre comme livre de chevet l'ouvrage suivant:

**Windows 2000 Active Directory**,  
Alistair G. Lowe-Norris, Editeur  
O'Reilly, ISBN 1-56592-638-2.

Cet ouvrage, écrit en janvier 2000, par un informaticien travaillant dans un environnement similaire au nôtre (Université de Leicester) offre une présentation cohérente, depuis la description des fondements de cette technologie jusqu'à sa mise en œuvre pratique avec l'utilisation de scripts, dont je me suis largement

SUITE EN PAGE 5

## SOMMAIRE FI8

- 1 Windows 2000  
Active Directory et ADSI  
une expérience dans un  
département  
*Laurent Kling*
- 2 sic-info
- 2 EPFL-SCR  
Appel pour les articles  
*Trach Minh Tran*
- 3 1,2,3 A,B,C a,b,c\*  
*Laurent Kling*
- 11 Mac OS X, version 10.1  
*François Roulet*
- 12 Programme des cours
- 15 VPN à l'EPFL  
*Jacques Virchaux*
- 15 Offres d'emploi
- 15 Présentation thinkdesign
- 16 Calendrier

## PROCHAINES PARUTIONS

	délaI	RÉDACTION	PARUTION FI
9	01.11.01		20.11.01
10	29.11.01		18.12.01

---

## SUITE DE LA PREMIÈRE PAGE

---

inspiré pour ce travail. Je le recommande vivement à chaque administrateur de Windows 2000.

La complexité de la mise en œuvre de Windows 2000 dépasse largement celle de Windows NT 4 par trois écueils:

- l'organisation hiérarchique de sa structure: contrairement à NT4 qui offre une structure plate facile à appréhender, Windows 2000 est hiérarchique par nature, ce qui nécessite une vision globale de ses fonctionnalités et une vision claire de son implantation;
- l'apparente simplicité de son interface utilisateur; la multitude des formulaires automatisés de nombreuses fonctions, de la plus simple comme la configuration de la file d'attente pour l'impression, à la plus stratégique, comme la création d'Active Directory et la définition de son rôle. Cette apparente facilité est trompeuse car elle peut occulter la nécessaire réflexion de sa mise en place ;
- la mise à jour immédiate\* des modifications, masquant la complexité du protocole de rafraîchissement des informations.

De ces réflexions, on peut mettre en place une charte de fonctionnement.

## CHARTRE DE FONCTIONNEMENT DE WINDOWS 2000

### PERSISTANCE DES INFORMATIONS:

La répercussion immédiate des modifications d'Active Directory entraîne à une vigilance accrue de celui-ci. Chaque changement de sa structure doit être répertorié et documenté.

### HIÉRARCHIE DES DOMAINES:

Chaque entité administrative doit exister sous la forme d'une Unité d'Organisation (OU) ou d'un serveur de domaine. Pour éviter une multiplication des serveurs de domaine, il peut être recommandé de créer des Unités d'Organisation (OU).

### SAUVEGARDE DES ÉTATS:

À chaque changement majeur de la structure d'Active Directory, on doit disposer d'une sauvegarde. Cette sauvegarde doit être documentée. Les différents logs du système sont sauvegardés dans leur intégralité. Toutes modifications des Groups Policy (GPO) doivent être documentées et sauvegardées.

### RESPONSABILITÉ:

Chaque domaine ou Unité d'Organisation (OU) possède un seul administrateur. La délégation de ce rôle peut être temporaire ou permanente. Dans le cas d'une délégation temporaire, une traçabilité des opérations effectuées est obligatoire. Un administrateur doit être une personne qui a suivi ou qui suit une formation adaptée (comme les cours du SIC).

\* En réalité, «après un certain temps» d'après Fernand Reynaud.

L'objectif de Windows 2000 est de fournir l'*outil* de gestion des ressources systèmes et de leur intégration dans un méta-annuaire; cette approche est un peu une illusion car malgré ses points de départ standard (X500, DNS), ils sont mis en pratique dans une culture de quasi-monopole. De plus, son intégration forte avec le système d'exploitation définit un outil n'offrant pas un support multi-plateforme complet (qui devrait dépasser le simple fait de partager des fichiers,...). Un des premiers services d'un méta-annuaire devrait pouvoir fournir un point d'entrée centralisé pour accéder à l'information sur les ressources informatiques par l'intermédiaire d'une interface disponible sur l'ensemble des plates-formes (comme le Web).

Il nous faut attendre les outils issus du projet e-pfl, ou de leur intégration avec Gaspar. Celui-ci fournit déjà une fonctionnalité adaptée à ces besoins, mais devrait disposer des outils (scripts) pour rapatrier ces données dans l'ensemble des environnements utilisés dans l'EPFL (Unix, Linux, MacOS, Windows NT 4 et Windows 2000). Cette démarche ouvrant l'éventail de leur mise en œuvre, d'un laboratoire ou institut jusqu'à une démarche centralisée pour l'EPFL.

Pour mon département, j'ai créé un outil, la logithèque (accessible par un lien depuis les pages Web du département <http://dgmwww.epfl.ch/>).

## LOGITHÈQUE DU DGM

Le but de ce site est de fournir un cheminement pour accéder aux ressources offertes dans le domaine de l'informatique par le Département de Génie Mécanique et son Service Informatique. Il s'adresse donc aux étudiants, corps enseignant et corps administratif du DGM de l'EPFL. Son utilisation demande un accès à Internet et pour certaines ressources, un accès à un courrier électronique.

Le fonctionnement de la logithèque se décompose en 2 parties:

- la première est une présentation des services offerts dans le département:
  - vous entrez dans la logithèque,
  - vous choisissez l'unité ou la ressource et vous visualisez les éléments disponibles: [unité] [ressource] [usagers].
- la seconde est une métaphore de grande surface:
  - vous entrez dans le magasin (lien: **Commande**),
  - vous choisissez la catégorie d'article souhaité,
  - vous choisissez l'article dans l'étalage en l'ajoutant à votre caddie ou chariot,
  - après l'ensemble de vos achats, vous visualisez votre caddie, où vous pouvez supprimer les courses en trop.
- Il ne reste qu'à cliquer sur le lien **Commander**:
  - où vous devez fournir vos coordonnées,
  - après vérification de l'information, votre demande est enregistrée,
  - et je suis averti par un courrier électronique de votre demande.

Cet outil est réalisé avec FileMaker Pro en utilisant la possibilité d'inclure des *tags* spécifiques à la mise en œuvre de cette base de données. On pourrait imaginer de réaliser ce travail avec des outils plus à la mode comme PHP et

MySQL par exemple. J'ai préféré ré-utiliser un travail similaire réalisé pour le Département d'Architecture; FileMaker Pro offre une souplesse de développement non négligeable en intégrant dans un seul outil multi-plateformes, les deux faces de ce genre d'application (base de données et serveur Web). Il peut également offrir des fonctionnalités avancées comme la répartition de charge. Dans un souci de pérennité, son interface utilisateur simple offre une communication adaptée à de nombreux intervenants.

## UTILISATION CONJOINTE DE LA LOGIÈRE ET DE SCRIPTS ADSI

La création des comptes pour les usagers d'un système est généralement une opération fastidieuse. Une manière classique sous Windows NT 4 est la duplication à partir d'un chablon qui demande l'introduction de valeurs dans 3 champs. Cette même opération sous Windows 2000 nécessite de remplir 5 champs dans 3 fenêtres différentes. En plus, le dossier de l'utilisateur pour son dossier de travail (Home Directory) créé automatiquement n'est pas accessible à l'administrateur (en Service Pack 1). Il est difficile de respecter la charte de fonctionnement précédemment citée, en particulier pour les points concernant la documentation et la mise à jour des informations.

Dans la modification des structures administratives en cours dans l'EPFL, en particulier la mise en place des Facultés et des nouveaux instituts, il est difficile de prévoir quelle sera l'organisation finale. Cependant le niveau d'agrégation minimum reste l'unité ou la chaire. L'objectif est de constituer une structure de gestion technique et administrative autour de cette unité, ceci dans le dessein de préparer son intégration dans un ensemble plus vaste.

Un outil parfaitement adapté pour cela est l'unité d'organisation (OU, Organisational Unit). Cet élément offre l'ensemble des caractéristiques requises:

- hiérarchisation possible,
- délégation de gestion, application de stratégie de sécurité (GPO, Global Policy Object).

La mise en œuvre pratique et son automatisation s'effectue par l'intermédiaire d'une boîte à outil, **ADSI** (Active Directory Service Interface), disponible depuis 1997 et qui offre une interface unifiée à des services différents comme:

- Windows 2000 Active Directory
- Windows NT 4.0
- IBM Lotus Note
- Microsoft Exchange Serveur
- Netscape Directory Serveur 1.0
- et quelques autres,...

Cette pierre de Rosette des administrateurs systèmes offre ainsi un outil parfait pour réaliser les travaux répétitifs.

Son intégration avec WSH (Windows Script Host), environnement *ouvert*\* de développement d'outils de gestion, fournit un outil adapté à nos besoins sous Windows 2000; son utilisation est également possible (après installation) sur

\* Il intègre sous le même chapeau, JavaScript et VBScript qui sont des sous-éléments de Java et de Visual Basic, voire Perl pour les aficionados.

Windows 95, Windows 98 et Windows NT 4. Pour les personnes désirant mieux connaître cet outil, je leur recommande la lecture de l'ouvrage suivant:

**Windows Script Host**, *Dino Esposito*, Editeur Wrox, ISBN: 1-861002-65-3.

Ces 2 outils ont été utilisés pour le travail décrit dans le précédent Flash informatique (*message destiné à tous les gestionnaires de salles de PC Windows ouvertes aux étudiants* de Francois Georgy et Christian Zufferey – <http://sic.epfl.ch/publications/FI01/fi-7-1/7-1-page24.html>) similaire à mon propos mais dans un contexte centralisé.

Cette ouverture du système par l'utilisation de scripts présente naturellement un trou de sécurité. Les 2 moteurs de WSH (**cscript.exe** et **wscript.exe** [respectivement en mode commande ou en mode Windows]) existent avec des permissions ouvertes pour tout le monde par défaut, je recommande que vous supprimiez cette option sur vos serveurs pour éviter des événements fâcheux (ces 2 exécutables se situent dans %SystemRoot%\system32). Je vous recommande également de ne pas ouvrir l'accès de l'utilisateur **Guest**, membre du groupe **Domain Users** [et avec celui-ci comme groupe principal !] offrant ainsi un accès à l'ensemble des ressources de votre système,...

## ENVIRONNEMENT DE DÉVELOPPEMENT

La création de scripts est fastidieuse, le cycle d'écriture, exécution, correction est grandement facilité avec des outils adaptés. Le développement d'outils système à partir de scripts est généralement un parent pauvre comparé aux environnements intégrés de développement (IDE) disponibles pour les langages les plus répandus (Java, C, C++, Visual Basic). Pour remédier à ce manque, la combinaison de 2 outils offre un environnement intégré (voir figure 1):

- le débogueur de scripts de Microsoft, disponible au téléchargement, offrant dans une version un peu spartiate cet outil indispensable de test. Il intègre une fenêtre présentant le code source avec la possibilité de positionner des points d'arrêts et une fenêtre d'affichage et/ou modification des variables;
- l'éditeur **PrimalScript 2.2** ([www.sapien.com](http://www.sapien.com)) offre la pierre de voûte du dispositif en intégrant un éditeur comprenant les possibilités classiques (bibliothèque des fonctions avec les arguments, etc.) et une intégration avec le débogueur et le moteur WSH pour un confort accru:
  - possibilité de se retrouver directement à l'endroit de l'erreur de syntaxe dans le code source,
  - re-direction des messages du moteur VSH dans une fenêtre propre à l'éditeur.

L'utilisation de machine virtuelle (VirtualPC, VMware) offre la mise à disposition de plusieurs environnements disponibles simultanément (dans la limite de la mémoire disponible) sur un seul ordinateur. Dans un souci œcuménique, une partie de ce développement a été réalisée avec VirtualPC sur un Macintosh [en fait, je ne possède pas de PC à la maison, mais un Macintosh]. Ces outils raccourcissent le temps de développement en offrant un confort accru et la possibilité de tester son travail.

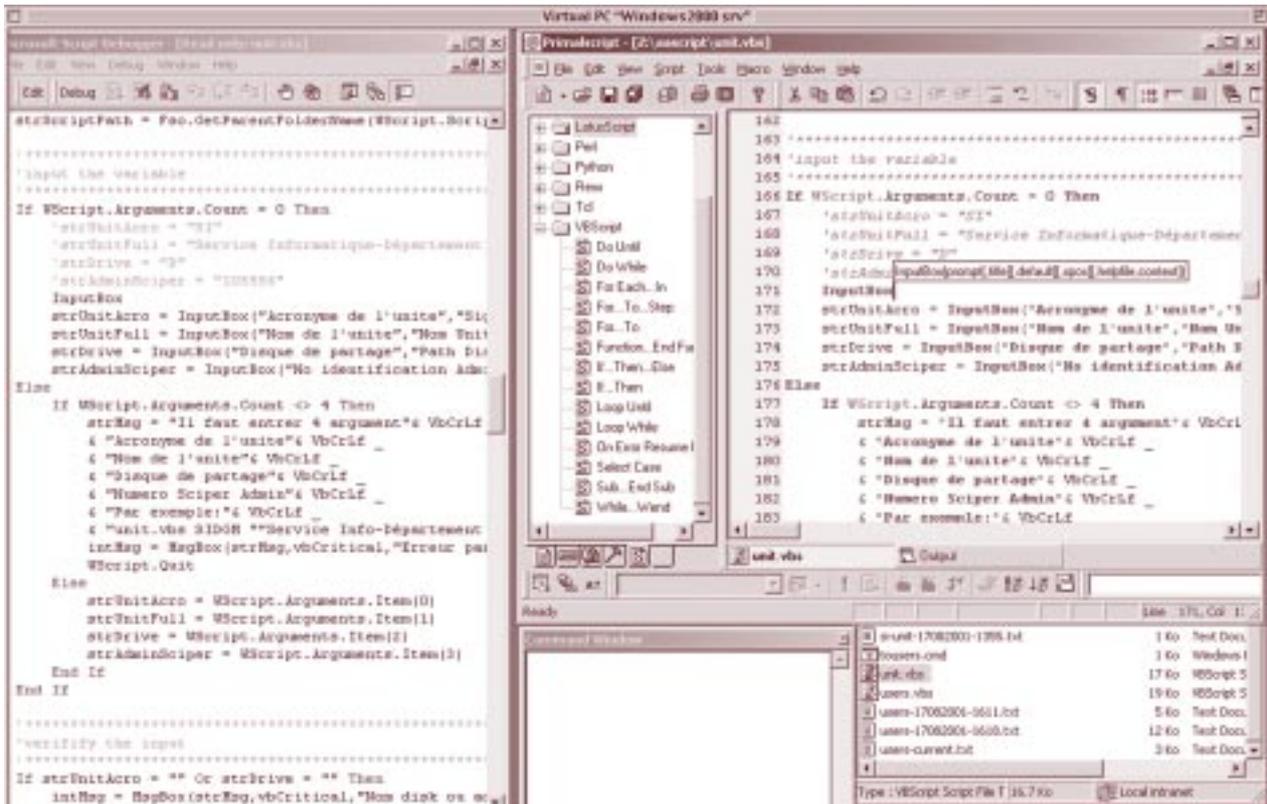


FIGURE 1

## Les scripts...

L'objectif de ce travail est de fournir un outil *clé en main* qui décharge l'administrateur de l'unité de la création d'utilisateurs. Cet outil part du postulat qu'actuellement un grand nombre d'administrateurs de systèmes Windows 2000 ne sont pas engagés dans cette seule activité; beaucoup sont des assistants-doctorants et leur travail sera simplifié si on leur offre un canevas d'administration.

Il nécessite comme point de départ les éléments suivants:

- un serveur Windows 2000;
- un accès en mode administrateur;
- une partition destinée à recueillir les données des utilisateurs de l'unité;
- les informations administratives de l'unité;
- une liste des utilisateurs concernés.

Il se présente sous la forme de deux scripts: <http://sic.epfl.ch/publications/FI01/fi-8-1/8-1-page7a.html> & <http://sic.epfl.ch/publications/FI01/fi-8-1/8-1-page7b.html>.

## 1. CRÉATION DE L'UNITÉ

Le premier script (création de l'unité: **unit.vbs**) est destiné à être exécuté sur le serveur qui va accueillir l'unité avec les droits de l'administrateur. La possibilité existe d'identifier l'accès à Active Directory (LDAP). Cette identification n'est pas nécessaire dans notre cas de figure, car nous héritons des privilèges liés à l'administration permettant d'effectuer des modifications.

Il nécessite 4 arguments (en mode commande, ou par l'intermédiaire de boîtes de dialogue en mode windows)

- acronyme de l'unité (ACRONYME);
- nom de l'unité;
- disque de partage;

- numéro Sciper Administrateur.
- Pour ceux qui ne désirent pas avoir les boîtes de dialogues apparaissant en mode Windows lors du déroulement du script, je propose deux solutions:

- utiliser le mode commande, ...;
- définir le moteur par défaut de l'exécution de WSH (wscript //H:Cscript).

Pour différencier le nom des groupes locaux des groupes globaux, j'ajoute un suffixe **G** aux noms des groupes globaux.

Son exécution entraîne la création des éléments suivants:

- le nom est basé sur = ACRONYME;
- le nom NETBIOS\* est utilisé pour accéder à la ressource;
- l'unité d'organisation (ou OU) = «ACRONYME»;
- le groupe global admin de l'OU = «ACRONYME-AdminG»;
- le groupe global staff de l'OU = «ACRONYME-StaffG»;
- le groupe global student de l'OU = «ACRONYME-StudentG»;
- le groupe global guest de l'OU = «ACRONYME-GuestG»;
- le dossier de l'OU (pour les dossiers des usagers et leur profiles) = «ACRONYME-home»

Ce dossier pour l'OU dans le disque de partage va contenir (créé avec le deuxième script):

- l'utilisateur inclus dans le groupe global staff de l'OU = «ACRONYME-StaffG»;
- son dossier de base se trouvant dans le dossier = \\NETBIOS\ACRONYME\USER;
- le nom du dossier correspond à son nom court = «USER»;
- son dossier de profil se trouve dans le dossier = \\NETBIOS\ACRONYME\USER\profiles

\* accéder par l'UNC = \\NETBIOS\ACRONYME

Il n'est pas nécessaire de fournir à ce script des informations système comme:

- le nom **LDAP** de la hiérarchie Active Directory;
- le nom **NETBIOS** de la machine;
- le nom **WinNT** de partage de fichiers.

Elles sont directement recueillies auprès du serveur concerné.

Dans le log système et dans un fichier, une trace des différentes opérations effectuées est générée:

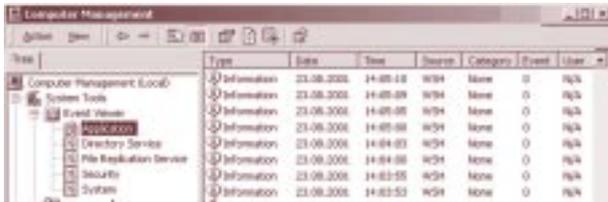


FIGURE 2

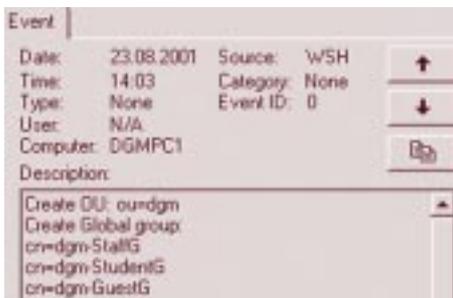


FIGURE 3



FIGURE 4

Et finalement il génère le fichier de configuration nécessaire pour la création des usagers:

```
si:DC=dgm-sg,DC=intranet,DC=epfl,DC=ch:DGM-SG:DGMP1:D:105556:Service Informatique-Département Génie Mécanique
```

Ce script crée directement le point de partage et configure également les droits des groupes globaux inclus dans l'OU (voir figures 5 & 6). La modification physique de l'accès au dossier contenant les données s'effectue par cacls.

## CACLS

Cet outil déjà disponible avec la distribution de ZAK sous NT 4 (voir mon article du numéro du Flash Informatique de septembre 1999: **Installation avec ZAK sur NT** <http://sic.epfl.ch/publications/FI99/ifi-7-99/7-99-page9.html> sur ce sujet) est disponible sous Windows2000, il offre la fonctionnalité de définir les droits d'un élément (dossier et/ou fichier) sur un disque et de lui attribuer des permissions, et cela de manière récursive si nécessaire. Son grand frère, xcacls est disponible au téléchargement chez Microsoft (voir fig. 7).

## 2. CRÉATION DES USAGERS

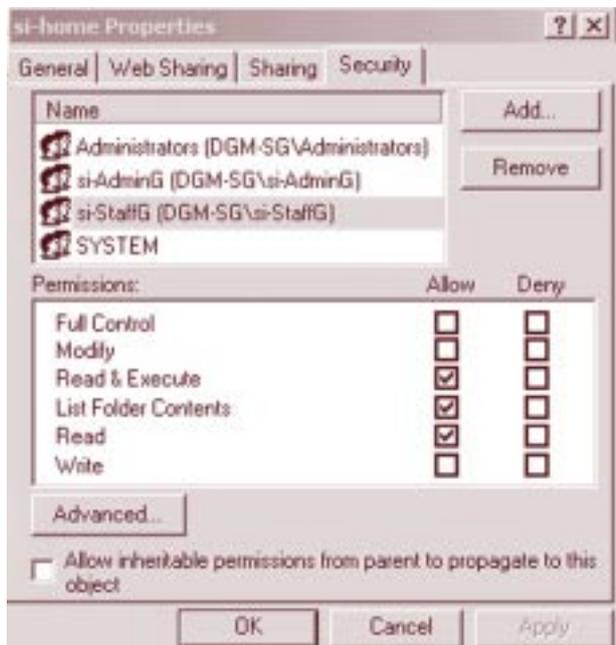
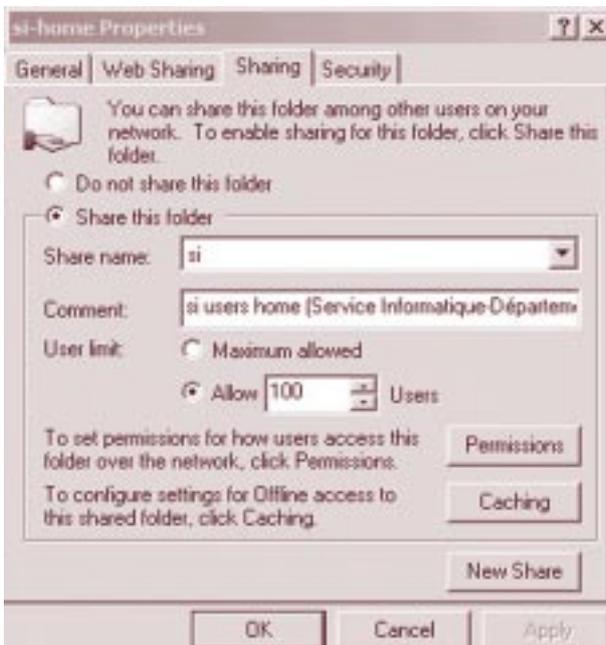
Le second script va créer les usagers (**users.vbs**). Il ne possède pas de paramètres, mais va lire un fichier contenant les informations suivantes sur les usagers (**new-users.txt**):

```
Unité, Institut, Département, Sciper,  
Nom 8 char, Nom e-mail CSO, Fonction,  
Prénom, Nom, Téléphone, Fax, Bureau
```

Par exemple:

```
SI:SG:DGM:129330:bapst:Alexandre.Bapst:  
apprenti:Alexandre.Bapst:3565::
```

Ce fichier peut être automatiquement généré par la logithèque. Et il peut contenir des usagers de différentes unités (pour des mises à jour par exemple).



FIGURES 5 & 6: POINT DE PARTAGE DE L'UNITÉ POUR LES USAGERS ET DROITS AFFÉRENTS

```

cacls.exe
Displays or modifies access control lists (ACLs) of files

CACLs filename [/T] [/E] [/C] [/G user:perm] [/R user [...]]
  [/P user:perm [...]] [/D user [...]]
filename Displays ACLs.
/T Changes ACLs of specified files in
  the current directory and all subdirectories.
/E Edit ACL instead of replacing it.
/C Continue on access denied errors.
/G user:perm Grant specified user access rights.
  Perm can be: R Read
                W Write
                C Change (write)
                F Full control
/R user Revoke specified user's access rights (only valid with /E).
/P user:perm Replace specified user's access rights.
  Perm can be: N None
                R Read
                W Write
                C Change (write)
                F Full control
/D user Deny specified user access.
Wildcards can be used to specify more than one file in a command.
You can specify more than one user in a command.

```

---

```

xcacls.exe
Displays or modifies access control lists (ACLs) of files

XCACLs filename [/T] [/E|/X] [/C] [/G user:perm;spec] [/R user [...]]
  [/P user:perm;spec [...]] [/D user [...]] [/Y]
filename Displays ACLs.
/T Changes ACLs of specified files in
  the current directory and all subdirectories.
/E Edit ACL instead of replacing it.
/X Same as /E except it only affects the ACEs that the
  specified users already own.
/C Continue on access denied errors.
/G user:perm;spec Grant specified user access rights.
  Perm can be: R Read
                C Change (write)
                F Full control
                P Change Permissions (Special access)
                O Take Ownership (Special access)
                X EXecute (Special access)
                E REad (Special access)
                W Write (Special access)
                D Delete (Special access)
  Spec can be the same as perm and will only be
  applied to a directory. In this case, Perm
  will be used for file inheritance in this
  directory. If not omitted: Spec=Perm. Special values
  for Spec only:
  T Not Specified (for file inherit,
  only for dirs valid)
  At least one access right has to follow!
  Entries between ';' and T will be ignored!
/R user Revoke specified user's access rights.
/P user:perm;spec Replace specified user's access rights.
  for access right specification see /G option
/D user Deny specified user access.
/Y Replace user's rights without verify

Wildcards can be used to specify more than one file in a command.
You can specify more than one user in a command.
You can combine access rights.

```

figure 7

Contrairement à une idée répandue, les deux modes d'accès à un domaine Windows 2000, soit:

- l'accès par le nom de domaine NetBios (à la NT 4): par exemple, l'utilisateur *bapst* dans le domaine `\\DGM-SG`
  - ou l'utilisation de la logique DNS: par exemple, l'utilisateur: *Alexandre.Bapst@dgm-sg.intranet.epfl.ch*
- sont strictement équivalents et ne dépendent pas du mode de fonctionnement (mixte ou natif) du domaine.



FIGURE 8

Pour éviter tout problème, et quoique les caractères utilisables dans Windows soient plus nombreux, je propose d'utiliser la norme Internet RFC 1123 qui propose les caractères suivants:

A-Z ; a-z ; 0-9 ; -

Avec le «.» comme séparateur entre le nom et le prénom. Cette manière de faire est identique à celle des adresses logiques du courrier électronique des usagers.

Il existe trois méthodes pour domicilier un usager dans un domaine 2000:

Méthode	Profil	Données	Usage
Uniquement un compte	Stocké sur chaque ordinateur où il travaille	Stockées sur chaque ordinateur où il travaille	Ne pas utiliser de domaine...
Données centralisées	Stocké sur chaque ordinateur où il travaille	Stockées (et sauvegardées sur le serveur)	Correct si l'utilisateur utilise un ordinateur captif
Indépendant	Stocké (et sauvegardé sur le serveur)	Stockées (et sauvegardées sur le serveur)	En cas de grand nomadisme

Il n'existe pas de règle absolue, la seconde ou la troisième solution me semble la plus favorable dans une logique de domaine. La documentation de Microsoft propose dans le troisième cas de créer 2 points de partage (un par type). Du fait de l'utilisation de script, il n'existe pas de problème à intégrer le dossier de profil dans le dossier de travail, offrant ainsi une gestion plus simple de l'utilisateur (en particulier dans son déplacement et dans la sauvegarde et récupération de ses données).

Je me permets de vous renvoyer sur le site Web de l'article déjà précité (<http://pcline.epfl.ch/students/default.asp>) qui décrit de manière très détaillée les conséquences des 2 dernières méthodes.

D'une manière similaire au premier script, dans le log système et dans un fichier, une trace des différentes opérations effectuées est sauvegardée.

```

Create User dans l'OU: si
FirstName - LastName: Alexandre-Bapst
DisplayName : Alexandre.Bapst
Description : Alexandre.Bapst dans l'OU
              si (Service Informatique-Département Génie
              Mécanique)

TelephoneNumber : 3565
EmailAddress : Alexandre.Bapst@epfl.ch
EmployeeID : 129330
Title : apprenti
Department : EPFL-DGM
Division : SG-si
ExpirationDate : 30.10.2003
LoginScript : \\DGM-SG\Loginscripts\si.cmd
    
```

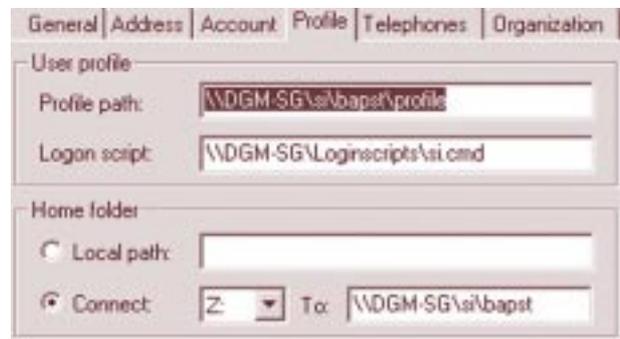
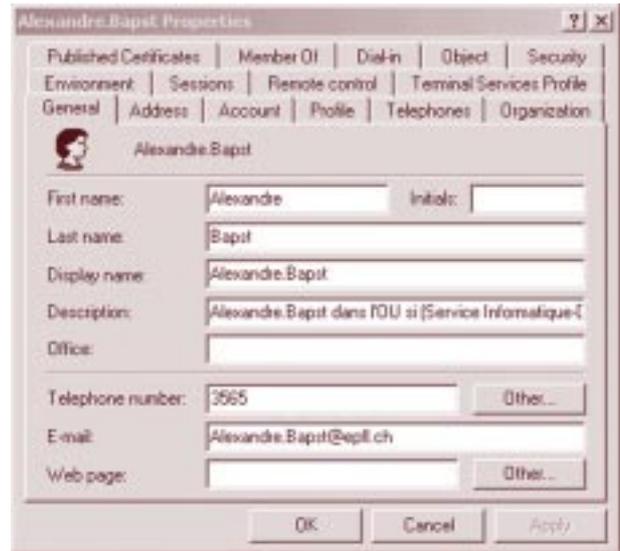
```

Profile : \\DGM-SG\si\bapst\profile
HomeDirectory : \\DGM-SG\si\bapst
    
```

Après l'exécution de ces 2 scripts, on obtient dans active directory:



Et pour un usager:



Les droits de chaque utilisateur sont de pouvoir modifier le contenu de son dossier et du dossier «trf». On peut facilement modifier ces droits dans le script par la commande **cacls** (voir plus haut).

Après son usage, la liste des usagers créée est mise à jour (**users-current.txt**). Pour éviter une re-crédation des usagers, le fichier d'origine est archivé.

Dans la durée de vie d'une unité, il est probable que la partition contenant le dossier de l'unité sera modifiée. Cela ne pose pas de problème car l'ensemble des accès aux fichiers se fait par l'intermédiaire de nom UNC (réseaux). Dans ce cas, il suffit de modifier le fichier de configuration lié à l'unité pour signifier ce changement (valide pour la création des nouveaux usagers).

L'ensemble des noms des fichiers des travaux effectués comporte un identificateur généré à partir de la date et de l'heure de création de l'opération facilitant l'organisation de la documentation des travaux effectués.

Ce travail doit permettre, en attendant la mise en place d'une professionnalisation de l'administration système, d'utiliser un outil permettant à un correspondant informatique de créer des structures informatiques adaptées à l'unité dont il a la charge sans occasionner un travail trop complexe.

Il permet de séparer clairement la gestion physique du serveur qui peut être déléguée à un administrateur système, de la gestion logique des usagers d'une unité ou d'un laboratoire qui doit rester proche des usagers dont il a la charge.

Les éléments de la charte de fonctionnement sont respectés, particulièrement les points concernant la documentation, plus faciles à faire respecter par un programme que par des personnes. ■