

# GESTION DES SALLES DE COURS PC À LA FACULTÉ STI

CLAUDE.MARY@epfl.ch,  
KATHARINA.SCHEIDEGGER@epfl.ch &  
LAURENT.KLING@epfl.ch, FACULTÉ STI



Salle BM2-115



Salle ELD-020



Salle MXF-014



Salle BM2-127

## INFRASTRUCTURE

130 PC, 4 imprimantes, 4 vidéo-projecteurs. L'équipement sert d'appuis pédagogiques et d'exercices pour les 1500 étudiants de la STI

## HORAIRE DES SALLES

Pendant le semestre: du lundi au vendredi de 7h00 à 22h00 et le samedi de 7h00 à 13h00

En fin de semestre: du lundi au vendredi 24h/24h

## INTRODUCTION

Avec la réorganisation de l'EPFL, les départements de microtechnique, électricité, matériaux et de génie mécanique se sont regroupés dans la Faculté STI (Sciences et Techniques de l'Ingénieur).

Pour faire suite à ce mouvement, la gestion indépendante des 4 salles de cours BM2-115, BM2-127, MXF-014, ELD-020 a été remise en question. L'idée d'une gestion uniforme et centralisée s'est rapidement imposée pour répondre aux

besoins de l'enseignement scientifique STI et pour donner aux 1500 étudiants un libre accès simple et contrôlé aux salles de PC. Ceux-ci sont repartis en quatre sections (microtechnique, électricité, matériaux et mécanique) bientôt complétées par la section de génie biomédical.

Les principaux besoins étaient les suivants:

- grande souplesse et réactivité pour la mise à jour des salles PC (système d'exploitation, logiciels d'applications);
- planification horaire (ordinateurs actifs ou inactifs dans des plages horaires variables).

Les problèmes majeurs étaient:

- un parc hétérogène de machines (lié à un historique);
- une dispersion géographique des salles;
- la multiplicité des logiciels en fonction des cours;
- la définition tardive des logiciels nécessaires pour les cours.

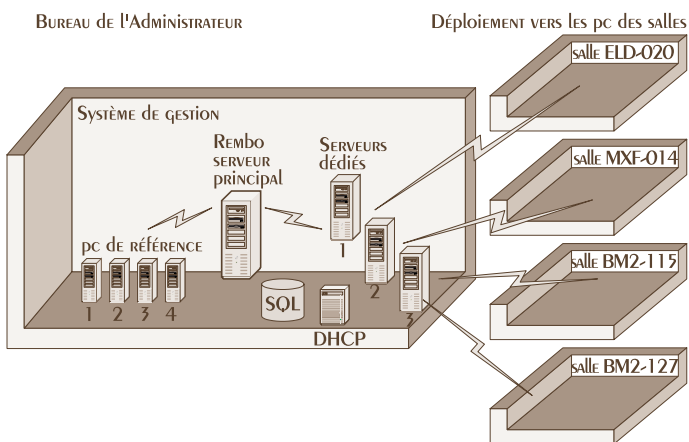
Pour résoudre ces problèmes, nous nous sommes appuyés sur les éléments suivants:

- le protocole PXE
- un outil de déploiement, Rembo Toolkit
- une base de données SQL
- l'intégration dans Active Directory.

L'expérience de la gestion des deux salles de microtechnique avec Rembo a été mise à profit. La migration du système d'exploitation de Windows NT 4 à Windows 2000, avec l'implémentation d'Active Directory (annuaire de données) et l'intégration du domaine STUDENTS offre un service commun à l'ensemble des étudiantes et étudiants de notre Faculté.

## GESTION CENTRALISÉE

En fonction des critères précédents, la solution adoptée est décrite dans le schéma de principe suivant.



La première étape est la génération des images pour une salle particulière sur le PC de référence associé à cette salle de cours. Ensuite, les images sont sauvegardées sur le serveur principal et dupliquées sur les serveurs dédiés correspondants. En dernier lieu, ces images sont transférées vers les PC des différentes salles en fonction de la planification désirée.

Dans la configuration installée, la robustesse du système de déploiement est assurée par plusieurs serveurs. Si le serveur dédié à une salle est en panne avant une demande de distribution, un serveur secondaire répondra à cette requête. Si le serveur dédié tombe pendant une distribution, le poste client peut se connecter à un autre serveur.

## LA PRATIQUE DE LA MISE À JOUR

### ACTUALISATION DU PC DE RÉFÉRENCE

Pour différentes raisons le PC de référence doit être mis à jour régulièrement, en règle générale à la demande d'un enseignant pour l'installation d'un logiciel ou d'une nouvelle version d'un logiciel.

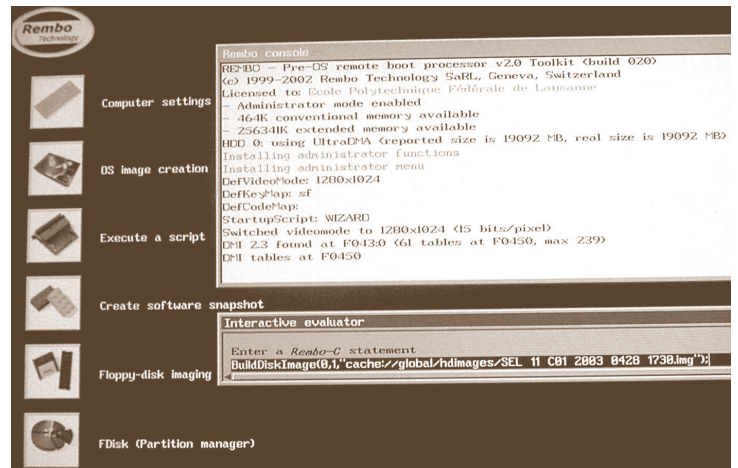
Après l'installation ou la modification du logiciel sur le PC de référence, plusieurs paramètres doivent être réglés:

- la réparation d'éventuelles erreurs sur le disque dur;
- la vérification par l'enseignant du parfait fonctionnement de l'installation;
- l'effacement du fichier *pagefile.sys*;
- la suppression du profil Administrateur local.

### LA MISE À JOUR DES SERVEURS Rembo

À ce stade-là, le PC de référence est prêt à être dupliqué. On redémarre pour cela le système d'exploitation, et sous l'interface graphique de Rembo, on saisit la commande:

```
BuildDiskImage(0,1, "cache://global/hdimages/SEL 11 CO1 2003 0428 1730.img");
```

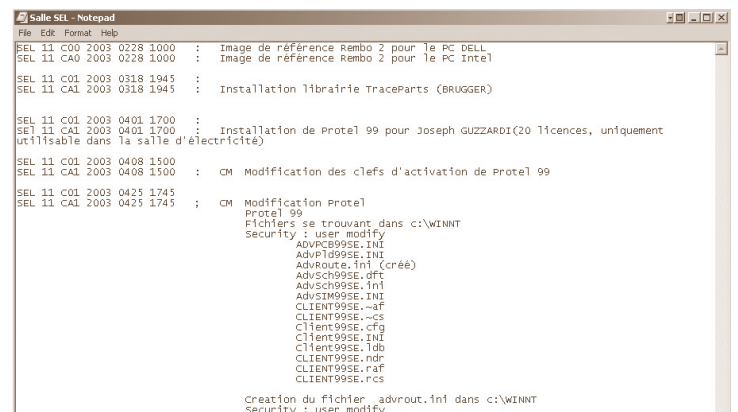


Le système Rembo commence à télécharger les fichiers du PC de référence et les enregistre dans le Serveur Rembo principal sous le répertoire partagé *\Files\global\hdimages*.

L'étape suivante consiste à copier la nouvelle image dans les 3 serveurs dédiés. Ceci nécessite l'arrêt temporaire de ces derniers. Le lancement du script, *Update Rembo from Rembo Server*, exécute la mise à jour. Pendant le transfert le Serveur principal Rembo répond à toutes les tâches. Dès que les transmissions de la nouvelle image sont faites, on passe à la mise en service des serveurs dédiés.

Dans un fichier historique, les descriptions de chaque changement d'image sont enregistrées.

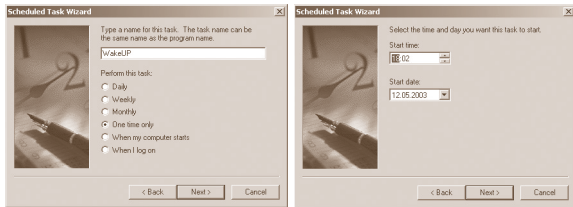
### MISE À JOUR DES SALLES PC



Les 4 serveurs Rembo interagissent et interrogent sans cesse le serveur SQL situé dans une machine dédiée. A cet effet, deux tables sont utilisées, l'une contient la description de l'ordinateur, l'autre contient la planification de la mise à jour.

Deux outils sont intégrés dans ce processus de gestion:

- la fonctionnalité *tâches planifiées* de Windows 2000 permet le réveil ou redémarrage d'une salle à une heure déterminée pour effectuer la mise à jour de celle-ci. Cette mise à jour s'effectue sans intervention humaine, par exemple pendant la nuit pour éviter de perturber le fonctionnement normal de la salle.



- un autre outil permet d'arrêter l'ensemble des PC d'une salle immédiatement, après une minute ou dix minutes avec un message de fermeture pour avertir l'utilisateur de sauvegarder les données. Ces interventions sont parfois nécessaires pour réagir rapidement dans certaines situations.

## PLANIFICATION TEMPORELLE

Il est apparu indispensable de pouvoir planifier le comportement des salles, c'est-à-dire définir quand sont utilisables les ordinateurs et quand sont déployées les images. A cet effet, nous avons exploité les extensions SQL de Rembo.

### L'ORGANISATION DES BASES DE DONNÉES

La première table a la structure suivante: une clef primaire qui est l'adresse IP (ip), le nom de l'ordinateur (name), la grandeur de chacune des trois partitions pouvant être créées (pp1, pp2, pp3), le nom de chacune des images associées aux partitions (img1, img2, img3) et le SID (sid).

ip	name	pp1	pp2	pp3	img1	img2	img3	sid
128.178.47...	dmtscpe9	3600000	2730000	0	SMT 11 C01 2003 0507 1430.img	...	...	D:
128.178.47...	mxpc9	8388608	0	0	SHX 11 C01 2003 0307 1200.img	...	...	8:
128.178.47...	mxpc10	8388608	0	0	SHX 11 C01 2003 0307 1200.img	...	...	4
128.178.47...	mxpc11	8388608	0	0	SHX 11 C01 2003 0307 1200.img	...	...	F:
128.178.47...	mxpc12	9774080	0	0	SHX 11 C01 2003 0307 1200.img	...	...	F
128.178.47...	mxpc13	8388608	0	0	SHX 11 C01 2003 0307 1200.img	...	...	5:
128.178.47...	mxpc14	8388608	0	0	SHX 11 C01 2003 0307 1200.img	...	...	F:
128.178.47...	mxpc15	8388608	0	0	SHX 11 C01 2003 0307 1200.img	...	...	4

la seconde table est propre à chaque ordinateur géré.

planning	commande
590	012513 0000
591	012514 0000
592	012515 0000
593	012516 0000
594	012517 0000
595	012518 0000

Dans celle-ci, la clef primaire *planning* est générée de la manière suivante:

- le mois, le jour et l'heure, ce qui nous donne pour 012513 le 25 janvier à 13 heures;
- la colonne *commande* définit ce que devra faire l'ordinateur entre 13H00 et 14H00.

Dans l'exemple: rien, ou plus exactement, lorsque qu'une personne démarre la station, celle-ci indique qu'elle ne peut pas fonctionner et s'éteint après 10 secondes.

Les autres opérations possibles sont:

- fonctionnement normal;
- installation des images;
- installation de Matlab;
- etc.

Rembo n'est actif qu'au démarrage, il est donc nécessaire d'envoyer une commande de réveil, *WakeUp*, ou de redémarrage, *ShutDown*, selon l'état de l'ordinateur pour qu'il soit activé. Pour automatiser une partie de la gestion des salles, il faut simplement planifier les commandes précédemment citées et définir la planification dans la base de données.

Ce système permet de traiter indépendamment chaque ordinateur à l'heure près.

### UNE GESTION FINE DES DROITS DES ÉTUDIANTS

L'utilisation du domaine STUDENTS offre une identification standardisée à l'EPFL et permet de bénéficier des outils disponibles, par exemple la capacité pour un élève de modifier son mot de passe avec l'interface Gaspar. Pour éviter de dépendre de ressources extérieures dont la mise en service n'était pas assurée à la rentrée 2002, un serveur de fichier propre à la Faculté offre un espace de travail par étudiant de 100 Mo accessible depuis chaque PC.

La problématique de la gestion ne s'arrête pas à l'identification des usagers, mais également à la pertinence de ceux-ci. Il est particulièrement désagréable pour un enseignant dont la salle informatique est réservée pour ses élèves de devoir faire la police pour libérer les postes.

En plus, dans le déroulement d'un enseignement, il est regrettable que les étudiants butinent sur la Toile au lieu d'exécuter les activités demandées.

Cette problématique n'est pas simple à gérer, elle augmente de plusieurs degrés la complexité de la solution actuelle. Deux pistes semblent exister:

- Une relation directe entre les machines et les usagers. Du fait que l'identification s'effectue dans un domaine (STUDENTS) différent de celui où se situe les ordinateurs (STI), il n'est pas possible d'agir directement sur les objets usager dans Active Directory. La possibilité qui nous reste serait de déplacer les ordinateurs dans des Unités d'Organisation (OU) propres à la combinaison Salle-Enseignant-Élèves. Le côté désagréable de cette mise en œuvre serait le redémarrage systématique des ordinateurs à chaque période de cours pour prendre en compte la nouvelle configuration.
- Une utilisation de droits de sécurité par application serait également une piste à suivre et permettrait un suivi plus précis de l'utilisation de celle-ci.

La mise en œuvre de telles solutions est complexe et ne devrait être réalisée que si le savoir-vivre des étudiants ne s'améliore pas...

## DANS LE DÉTAIL

Le point de départ est l'emploi du protocole PXE, qui permet l'intégration de services avant le démarrage effectif du système d'exploitation.

### PREBOOT EXECUTION ENVIRONMENT (PXE)

Cet outil, développé par Intel, permet de charger et d'exécuter un programme de démarrage réseau (NBP) depuis un serveur avant de démarrer sur le disque dur de l'ordinateur. Il procède de la manière suivante:

- obtenir une adresse IP dynamique depuis un serveur DHCP (elle peut être pseudo statique si elle est réservée);
- négocier avec le serveur de démarrage pour rechercher avec TFTP le programme de démarrage réseau;
- exécuter ensuite ce programme pour offrir la fonctionnalité appropriée, en l'occurrence Rembo Toolkit.

En pratique, la machine ne démarre que sur le réseau, exécute l'opération définie avec Rembo et finalement lance l'exécution de Windows 2000. Cette phase préparatoire reste transparente pour le système d'exploitation, car elle intervient au niveau du BIOS.

### Rembo, <http://www.rembo.com>

L'origine de Rembo est un travail issu du centre informatique de l'université de Genève (CUI) commencé en 1996, pour gérer un parc informatique. La fonction principale est de remettre à jour un PC à partir d'un démarrage par le réseau.

La première version consistait en une ROM de démarrage installée physiquement sur la carte réseau. L'apparition du protocole PXE permet d'éviter cette ROM. A partir de ce travail, une version gratuite a été élaborée, elle est disponible à l'URL: <http://www.bpbatch.org/>

À la suite du succès de ce premier logiciel, deux des auteurs, Marc Vuilleumier-Stückelberg et David Clercs, décidèrent d'y ajouter des fonctionnalités plus évoluées et de créer une entreprise autour de ce logiciel. Le 31 mai 2000, Rembo 1.0 fut officiellement mis sur le marché.

La reconnaissance des qualités de ce logiciel fut scellée par la collaboration entre Rembo et Hewlett-Packard. Depuis mars 2001, HP utilise la gamme des produits Rembo comme outils standard pour leur solution de déploiement de PC. Actuellement, il existe 3 produits Rembo:

- Rembo Toolkit
- Rembo auto-deploy
- Rembo auto-backup.

Ces produits permettent le déploiement de partitions aux formats: FAT12, FAT16, BIGDOS, FAT32, EXT2FS, LINUXSWAP, NTFS4 (NT4) et NTFS5 (Win2K/WinXP).

La dernière version, Rembo Toolkit 2, est disponible depuis le mois d'août 2002.

### LE FONCTIONNEMENT DE REMBO

Rembo est basé sur une architecture client-serveur: le serveur fonctionnant sous Windows NT, Windows 2000, Linux, FreeBSD ou Solaris. Le client, un PC équipé d'une carte réseau PXE charge Rembo en mémoire au démarrage,

## EFI, UN LANGAGE COMMUN POUR LE BIOS ?

Dans l'hypothèse qu'on désire connaître les fondements de la technologie PXE, au cœur de ce processus, on observe avec effroi que la version 16 bits n'est plus supportée par Intel !

<http://www.intel.com/labs/manage/wfm/tools/pxepdk20/index.htm>

Notre intérêt se porte naturellement sur son successeur, Extensible Firmware Interface

<http://www.intel.com/technology/efi/index.htm>

Ce *nouveau* concept tend à remplacer le développement de pilotes spécifiques pour les différents composants d'un ordinateur (carte mère, réseau, graphique, etc.) par un langage commun.

Cette initiative ressemble fortement à celle de l'*Open Boot Firmware*, qui devait fournir ce type de fonctionnalités avec le langage Forth.

EFI consiste en une couche d'abstraction intégrée dans la ROM de chaque composant qui sépare les fonctionnalités logiques de l'implémentation physique. Par cette couche intermédiaire entre le matériel avec son firmware et le système d'exploitation, il est possible de simplifier celui-ci, car il ne se connecte qu'avec des composants logiques et permet d'éviter le trop fameux écran du BIOS et les paramètres physiques des composants.

Les fonctionnalités présentes dans EFI sont multiples:

- démarrage de système d'exploitation;
- chargement des protocoles;
- édition de texte;
- accès au réseau;
- démarrage d'applications spécifique;
- utilisation des scripts (lire, écrire et exécuter);
- accès au matériel;
- manipulation de la mémoire.

Le tout avec une implémentation en Langage C dont le code source est disponible.

La syntaxe utilisée est un mélange entre le Shell Unix et celui du monde DOS.

EFI	bash	DOS
PAS SENSIBLE À LA CASE	SENSIBLE À LA CASE	PAS SENSIBLE À LA CASE
\	/	\
^	^	/
fs0 ;	/	C :
ls	ls	Dir
attrib	chmod	attrib
cp	cp	copy
rm	rm	del
mv	mv	move
mkdir	mkdir	md
%var%	\$var	%var%

L'intégration du protocole comme 802.1 X permet une identification complète pour le processus de démarrage. Cela évite le trou de sécurité existant dans BOOTP qui n'identifie la machine que par son adresse MAC.

La première implémentation pratique de cette technologie est prévue pour la plate-forme Itanium (64 bits), mais elle devrait être adoptée également dans la gamme 32 bits.

référence les fichiers et les autres ressources par le biais d'URL. Les fichiers servis par le serveur Rembo sont principalement :

- des scripts Rembo-C
- des bibliothèques de fonction Rembo-C
- des images de partitions.

Voici des exemples d'URL :

*disk:// 0:1/windows/notes.ini* – référence le fichier *notes.ini* du sous-répertoire windows de la partition 1 du disque 0 de la machine où cet URL est invoqué

*floppy://0/autoexec.bat* – référence le fichier *autoexec.bat* dans la racine du premier lecteur de disquette de la machine où cet URL est invoqué

*Net://global/run.rbc* – référence un fichier dans la racine du serveur Rembo à télécharger en Unicast

*Cache://global/run.rbc* – référence un fichier dans la racine du serveur Rembo à télécharger en Multicast et à stocker en cache (voir *Multicast et partition de cache*).

## Rembo-C

L'exécution d'une séquence de démarrages par Rembo est personnalisée par un langage de script dont la syntaxe est inspirée du C. Ces scripts s'exécutent sur le PC client via la machine virtuelle au cœur même de Rembo. Par exemple, le script suivant :

```
SetPrimaryPartitions(0, "FAT32:2097152");
HDClean(0,1);
Synchronize ("cache://global/w98.img",
             "disk://0:1", "b");
HDBoot(0,1);
```

définit une partition primaire FAT32 de 2Gb, la formate, télécharge une image du serveur Rembo, la copie et redémarre l'ordinateur sur celle-ci.

Rembo-C offre de nombreuses fonctionnalités pour personnaliser une image en fonction du type de matériel que Rembo détecte :

- authentification sur un domaine NT ou NIS;
- gestion d'une interface graphique;
- ajout/modification/suppression d'un fichier;
- ajout/modification/suppression d'une clé de registre Windows NT/2000/XP;
- support de SysRep pour Windows 2000 et XP.

Tout cela se déroule AVANT le démarrage du système d'exploitation.

Si une interaction avec l'utilisateur est souhaitée durant la séquence de démarrage, une interface graphique est disponible. Elle utilise le format HTML. Par exemple, le point d'entrée d'une séquence de démarrage est une page Web.

## MULTICAST ET PARTITION DE CACHE

Lorsque le client référence un fichier à télécharger avec un URL de type Cache, le protocole de transfert utilise une technique de Multicast : cela signifie que si plusieurs clients demandent le même fichier, une seule et même session réseau sur le serveur est utilisée. En plus, le fichier ainsi chargé est conservé sur l'espace non partitionné du disque local. Ce principe permet d'éviter de répéter le téléchargement de fichiers déjà existants depuis le serveur.

Il faut, bien entendu, prévoir la structure des partitions sur le disque du client, de façon à laisser un espace non partitionné équivalent au minimum à un tiers de la grandeur de l'image.

## ACCÈS À LA BASE DE REGISTRE DE WINDOWS NT/2K/XP

Rembo peut lire / écrire / ajouter / modifier / supprimer des clés de registre Windows. Du fait de la modification directe de la base de registre, son utilisation demande une connaissance approfondie de celle-ci.

## IMAGES INCRÉMENTALES

Lors de la création d'une image, Rembo :

- recherche sur l'ordinateur de référence, les nouveaux fichiers;
- stocke ceux-ci dans le serveur Rembo;
- génère une nouvelle image contenant la liste de l'ensemble des fichiers.

Lors de la distribution d'une image, Rembo :

- récupère la liste des fichiers contenus dans l'image;
- compare celle-ci avec la liste des fichiers se trouvant dans le cache;
- récupère les fichiers manquants;
- recrée ensuite les structures de fichiers se trouvant dans la ou les partitions.

Rembo utilisant un cache sur le poste client, en conséquence, la distribution des images est incrémentale.

## ACCÈS AUX FICHIERS ET REDONDANCE

Depuis Rembo Toolkit 2, il est possible d'accéder aux fichiers de Rembo via l'OS. En conséquence, un duplicata du serveur peut se réaliser avec l'utilitaire *Robocopy*. Pour augmenter la robustesse du système, il est préférable de multiplier les serveurs. Dans ce but, une variable permet de changer de délai de réponse à un *discovery packet* envoyé lors de la requête DHCP. Ainsi le serveur secondaire répondra dans le délai choisi si le serveur principal est en panne. La variable *Backup* dans Rembo définit le serveur de sauvegarde auquel le client se connectera.

Comme un script peut copier tous les types de fichiers depuis un serveur sur un poste client, la personnalisation des installations se fait sans problème et avec une grande souplesse.

## L'AVENIR DE CETTE CONFIGURATION

La croissance du nombre d'étudiants (plus de 1500) de notre Faculté engendre un ratio nombre de postes disponibles (130) par étudiant actuellement peu favorable.

Le principal avantage de la méthodologie exposée est de permettre une mise à jour très rapide de la configuration, en un jour ou même 3 h. Cette facilité permet de répondre aux nombreuses demandes, même de dernière minute, de modifications de la part des enseignants. La capacité de traiter indépendamment chaque ordinateur à l'heure près est particulièrement souple et permet la mise en place des différents scénarios de gestion.

En plus, la gestion entièrement à distance des salles de cours PC nous permet de nous accommoder d'une répartition géographique décentralisée. Cette prestation centralisée est plus favorable aux étudiants des quatre sections qui composent notre Faculté (matériaux, électricité, microtechnique et génie mécanique) bientôt complétées par une cinquième section en génie biomédical. ■