

*JeDDL*a*J : le teaser*

G rard Milhaud, Fr d ric Bloise a.k.a La F.I.R.M.E
ESIL – Universit  de la M diterran e



La problématique

- Très grand nombre de postes de travail
 - Forte diversité des architectures
 - Configurations logicielles très changeantes dans le temps (formations, intervenants divers, etc.)
 - Plusieurs OS différents, des doubles boot
 - et... indigence des ressources humaines
- => gestion/supervision du parc = 90% de l'activité...

Un début de solution : Rembo Toolkit (1)

- PXE ! Exit disquettes et CD...
 - Images disques stockées AUSSI en cache sur le disque local du client => réinstallation très rapide
 - Répertoire partagé pour les images disques sur le serveur : pour un même OS, un fichier présent dans N images disque n'est stocké qu'une fois => grand nombre d'images facilement stockables
-
-

Un début de solution : Rembo Toolkit (2)

- Langage Rembo-C aux multiples primitives et API pour scripter le comportement d'une machine en mode PXE : tout devient possible (partitionnement, intervention sur le système de fichiers (FAT, NTFS, EXT2/3), accès au réseau pour récupérer des informations, envoi de mails, etc.)
 - Possibilité d'images incrémentales... et donc de paquetages logiciels
 - Multicast : déploiement d'une salle \approx déploiement d'une machine
 - \sim 20€ HT/machine mais **attention** : suite au rachat de Rembo par IBM été 2006, la politique de prix va changer (a priori en mieux...). Pas plus d'infos au jour d'aujourd'hui (Octobre 2006)
-
-

Mais, en vrai, c'est pas encore ça... (1)

- Visibilité du parc matériel (et surtout logiciel) limitée (Rembo-console...)
 - Toute l'information sur le parc est sur le serveur Rembo sans moyen d'interrogation puissant/scriptable
 - Manipulation des variables Rembo assez lourde (Rembo-console et clicofolies)
-
-

Mais, en vrai, c'est pas encore ça... (2)

- Création de paquetages logiciels contraignante (machine dédiée, nécessité de partir d'une image disque non modifiée)
 - Pilotage du parc laborieux : pour ajouter/supprimer un logiciel d'un poste ou d'une salle, manipulations fastidieuses sous Rembo-console
 - Finalement : 2 problèmes principaux
 - Fonctionnalités présentes mais pénibles à mettre en oeuvre pour un usage quotidien
 - Information insuffisamment accessible
-
-

02/2003 : *We have a dream...*

- Dis, tu trouves pas que ça serait terrible si on pouvait :
 - piloter complètement le parc au travers d'une interface web (IW), et ce sans connaissance informatique
 - ajouter/supprimer des logiciels d'une machine ou d'un groupe de machines d'un simple clic dans l'IW
 - avoir une vue synoptique et configurable d'un simple clic de l'ensemble du parc via l'IW, des composants des machines aux logiciels installés en passant par les @ MAC et IP, etc.
 - Se servir de tout ça pour avoir un inventaire matériel et logiciel complet et à jour
-
-

02/2003 : *We have a dream (2)*

- Dis, tu trouves pas que ça serait terrible si on pouvait **AUSSI** :
 - Voir les choses de manière totalement modulaire :
 - 1. on créerait une image disque de base seulement avec l'OS et les réglages voulus qui conviendrait pour toutes les machines d'architecture compatible
 - 2. on créerait, par OS, des paquetages pour chaque logiciel voulu qui conviendrait pour toutes les machines sous cet OS
 - 3. on piocherait par clic via l'IW dans ce réservoir d'images de base et de paquetages logiciels pour configurer toutes les machines du parc
 - Ainsi, la **sainte-quête-du-travail-minimum** serait achevée :
 - **UNE SEULE** installation de l'OS par architecture matérielle
 - **UNE SEULE** installation d'un logiciel par OS
-
-

*09/2003 : We make le dream vrai : JeDDL*a*J is born (1)*

- Pilotage complet du parc par l'IW : plus aucune connaissance Rembo nécessaire
 - Oubliées, les variables Rembo et la Rembo-console : tout est inclus dans une base de données relationnelle MySQL qui décrit entièrement le parc
 - Définition déclarative complète du parc matériel et logiciel via l'IW (les caractéristiques matérielles des machines sont auto-insérées dans la base à leur premier boot PXE)
-
-

*JeDDL*a*J is born (2)*

- La base est consultée/renseignée à la fois par l'IW et les scripts REMBO-C qui composent JeDDL*a*J
 - Création de paquetages logiciels depuis n'importe quelle machine du parc, sans réinstallation
 - Création d'images de base depuis n'importe quelle machine du parc
-
-

JeDDL*a*J is born (3)

- La saine quête est achevée... Après avoir créé une fois pour toutes ses images de base et ses paquetages logiciels, on va, via l'IW, et donc depuis n'importe où :
- 1. choisir une machine ou un groupe de machines
- 2. choisir une image de base parmi les proposées
- 3. choisir les logiciels voulus parmi la liste
- 4. rebooter les machine(s)
- 5. Et voilà. On a déployé dans la joie et du coup...

On part à l'apéro une heure plus tôt !!!

Donc finalement, JeDDLaj, c'est quoi ?

Pas facile à définir en quelques mots... On peut tenter :

- Un outil de pilotage/supervision de parc via le web ?
 - Une base de données MySQL décrivant un parc informatique ?
 - renseignée/consultée via une interface web dynamique
 - exploitée/renseignée par un ensemble de programmes Rembo-C lancés par les machines du parc au boot PXE
 - Un logiciel libre universitaire ?
 - Un front-end web à Rembo Toolkit ?
 - Un moyen puissant d'administrer un parc hétérogène, avec des configurations logicielles fluctuantes ?
 - 9500 lignes de PHP, 3200 de Rembo-C ?
 - Tout ça à la fois ???
-
-

Et qu'est-ce qu'il me faut, à moi admin. réseau, pour installer et utiliser JeDDLàJ ???

- Des cartes réseaux PXE (ou des disquettes...) sur toutes les machines à gérer par JeDDLàJ
 - Déclarer toutes ces machines dans le DNS
 - Déclarer toutes ces machines en attribution d'@IP statique dans le serveur DHCP
 - Un espace WEB avec PHP (4 ou plus) disponible (et EXPECT pour le module non fondamental mais bien pratique de connexion au serveur REMBO depuis l'IW)
 - Un serveur MySQL accessible depuis cet espace web
-
-

*Et qu'est-ce qu'il me faut, à moi admin. réseau, pour installer et utiliser JeDDL*a*J ??? (2)*

- Un serveur REMBO Toolkit (RT)
 - Un pont ODBC entre le serveur RT et le serveur MySQL
 - Lire la doc. à
http://www.esil.univ-mrs.fr/~lafirme/website/rubrique.php3?id_rubrique=7
 - Télécharger la dernière version à cette même adresse
 - ½ journée à 1 journée tranquille pour mettre tout ça en place
 - Un hamac pour la sieste pendant que JeDDL*a*J travaille...
-
-

*JeDDL*a*J : Résumé des features qui vont vous convaincre... (1)*

- Auto-détection matérielle des machines
 - Création de paquetages logiciels/d'images de base sur toute machine du parc
 - Postinstall scripts associables aux OS et/ou aux paquetages logiciels
 - Gestion du multi-boot
 - Gestion des erreurs pendant l'installation
 - Cadeau Linux : postinstall script (pour Debian, mais adaptable) d'ajout à la volée des bons modules kernel en fonction des composants détectés
-
-

*JeDDL*a*J : Résumé des features qui vont vous convaincre... (2)*

- Gestion fine des groupes de machines
 - Action possible sur une machine ou sur un groupe
 - Examen graphique d'une machine, d'un groupe
 - Copie de machine
 - Calcul de l'image de base « la plus adaptée »
 - Échange simple de paquetage ou d'images de base avec d'autres JeDDL*a*Jeux
 - Cadeau Windows : images génériques
-
-

*Ce que JeDDL*a*J ne fait pas... encore*

- Délégation d'administration par authentification de sous-ensembles du parc
 - Interface web d'import/export de packages d'autres JeDDL*a*Jeux
 - Véritable « versionning » des logiciels (en tant que relation dans la BD)
 - Gestion des dépendances entre paquetages logiciels
-
-

Ce que JeDDLaj ne fera peut-être jamais...

- JeDDLaj hérite des limitations de REMBO...
 - Pas de connaissances des systèmes de fichiers autres que FAT, FAT32, EXT[23]FS => images monolithiques pour tout autre OS que Windows et Linux
 - Pas de possibilité de packager des logiciels Linux .deb ou .rpm (problème plus général des fichiers modifiés par plusieurs packages...)
-
-