Guide IDA
de migration
vers l'Open Source

netproject Ltd
124 Middleton Road
Morden, Surrey
SM4 6RW
tél. : +44 (0) 208 715 0072
fax : +44 (0) 208 715 7134
web : www.netproject.com

17 novembre 2003
réf. : OSPL/EEC-01.10 - fr0

0.1 Historique du document

Date	*Version*	*Auteur*	*Modifications*
11-02-2003	0.1	S. Hnizdur	Première édition
13-04-2003	0.2	S. Hnizdur	Ajouts et réorganisation
14-05-2003	0.3	S. Hnizdur	Ajouts et réorganisation
20-05-2003	0.4	S. Hnizdur	Préparation de l'édition pour la réunion PMB
21-05-2003	0.5	S. Hnizdur	Inclusion des commentaires V0.4
31-05-2003	0.6	S. Hnizdur	Inclusion des commentaires V0.5
30-06-2003	0.7	S. Hnizdur	Inclusion des commentaires V0.6 et applications MS serveur
16-07-2003	0.8	S. Hnizdur	Inclusion des commentaires V0.7 et scénario Unix
17-07-2003	0.9	S. Hnizdur	Inclusion des commentaires V0.8 et applications MS-Windows poste de travail
10-08-2003	0.91	S. Hnizdur	Inclusion des commentaires V0.9
27-08-2009	0.92	C. P. Briscoe-Smith	Corrections et inclusion de certains commentaires V0.91
04-09-2003	0.93	C. P. Briscoe-Smith	Corrections
08-09-2003	0.94	C. P. Briscoe-Smith	Autres corrections et inclusion de commentaires
14-09-2003	0.95	S. Hnizdur	Réorganisation après la réunion PMB
25-09-2003	0.96	S. Hnizdur	Corrections
07-10-2003	0.97	S. Hnizdur	Édition finale
10-10-2003	1.0	S. Hnizdur	Sortie de la version 1
17-11-2003	1.0fr0	B. Choppy	Adaptation française

0.2 Distribution

netproject Ltd	x1
Frequentous Consultants Ltd	x1
Commission européenne (pour diffusion aux administrations membres)	x2

0.3 Termes protégés

Les marques ont été utilisées dans ce document dans un strict but d'identification. Les auteurs reconnaissent la propriété de ces marques.

0.4 Copyright

0.5 Table des matières

Annexes

1 Préface

1.1 Abréviations et terminologie

Autant que possible, lors de la première utilisation d'une abréviation, la version étendue sera aussi incluse. Un glossaire des termes est fourni en annexe G. Lorsqu'un terme du glossaire est introduit pour la première fois, il est présenté dans le style suivant : *glossaire*.

Les termes *Logiciel Open Source* et *Logiciel libre* ont chacun leurs inconditionnels. Dans ce rapport, nous utilisons le terme Logiciel Open Source ou OSS dans l'intention d'idiquer que le logiciel décrit présente les caractéristiques implicites dans les deux locutions Logiciel Open Source et Logiciel libre/ Pour plus d'informations sur ces termes, se reporter à http://www.gnu.org/philosophy/catefories.html/ et http://www.opensource.org/.

Les noms de produits seront présentés dans le style suivant : **nom de produit**.

Les termes de systèmes d'exploitation tels que les noms de fichiers seront présentés dans le style suivant : fichier.

Le code programmatique sera présenté dans le style suivant : `Code`.

1.2 Audience

Ce document est conçu pour les D.S.I. des administrations au sein de l'Europe. Le terme *Administration* est utilisé au long de ce guide pour désigner une administration publique européenne et le terme *Administrateur* pour désigner ce groupe de personnes.

1.3 Auteurs

Le rapport a été produit par netproject Ltd en collaboration avec Frequentous Consultants Ltd. Le contenu a été produit par plusieurs consultants dont :

- Steve Hnizdur ;
- Kaith Matthew ;
- Eddie Bleasdale ;
- Alain Williams ;
- Andrew Findlary ;
- Sean Atkinson ;
- Charles Briscoe-Smith.

L'adaptation française a été réalisée par Bernard Choppy.

1.4 Remerciements

netproject souhaite remercier ceux qui suivent pour leur aide :

- en particulier de nombreux membres du l'équipe de gestion de projet pour les vivantes et informatives discussions mensuelles ;
- Thomas Krupp et Haiko Thede (Meklenburg-Vorpommern) ;
- Franck Müller (Chef de la planification du Land de Meklenburg-Vorpommern) ;

- Mru Patel (Sun Microsystems) ;
- Graham Taylor (Open Forum Europe) ;
- Richard Heggs (Conseil municipal de Notthingham) ;
- Andy Lack (Université de la ville de Londres)
- David Amos (groupe BSS) ;
- Steven Pennant (London Borough of Newham) ;
- Andy Trevor (Total Solutions) ;
- Geoff Gunton (Northern Rock).

Première partie : Introduction et sommaire

2 Introduction

L'objectif de ce guide est double :

1. l'aide aux administrateurs dans la décision de migration vers l'OSS ;
2. la description en termes techniques de la démarche d'une telle migration.

Le guide est conçu pour être d'une utilité pratique aux administrateurs et doit ainsi être idoine et fiable tout en restant accessible et compréhensible. Il ne constitue pas un manuel de référence technique détaillé. Sa structure est conçue pour permettre des modifications aisées dans le futur au fur et à mesure que les administration gagneront en expérience et que des modifications des produits seront disponibles.

Pour atteindre de ces objectifs, il est impératif d'en maintenir le contenu à jour et que toutes inexactitudes en soient retirées. À cette fin, les commentaires et contributions des lecteurs sont encouragées sur toute partie de ce guide. Merci de faire parvenir les commentaires à gposs@cec.eu.int[1].

Ce quide ne parle pas de l'OSS en général ni des mérites comparés des différentes licences. Cela, ainsi qu'une grande quantité d'autres informations, peut être trouvé dans les documents IDA suivants :

1. la feuille de faits de l'OSS :
   http://europa.eu.int/ISPO/ida/export/files/en/840.pdf ;
2. le rapport sur l'utilisation de l'OSS :
   http://europa.eu.int/ISPO/ida/export/files/en/837.pdf ;
3. le rapport sur la structure du marché et les éléments sur les relations avec le public :
   http://europa.eu.int/ISPO/ida/export/files/en/835.pdf.

(ces trois documents peuvent être trouvés à http://europa.eu.int/ISPO/ida/jsps/index.jsp?fuseAction=showDocument&parent=crossreference&documentID=333/ dans d'autres formats en sus du PDF).

En France, l'Agence pour le développement de l'administration électronique (A.D.A.E.) a produit un bon guide sur les licences, disponible aussi en anglais à :
http://www.atica.gouv.fr/pages/documents/fiche.php?id=1450&id_chapitre=8&id_theme=55&letype=0.

1 NdT : Les remarques sur l'adaptation française sont encouragée auprès du traducteur à : choppy@imaginet.fr

3 Sommaire

Ce guide est destiné aux D.S.I. et techniciens qui planifient ou effectuent une migration vers le logiciel OpenSource (OSS). Il est fondé sur l'expérience pratique des auteurs et le contenu d'un nombre limité d'études de cas accessibles au public. Il a été validé par la migration vers l'OSS de la Court of Auditors, Schwerin de Mecklenberg Vorpommern.

Pour les administrations, il y a de nombreuses raisons de migrer vers l'OSS. Celles-ci incluent : la nécessité d'utiliser des standard ouverts pour le e-gouvernement, le niveau de sécurité fourni par l'OSS, l'élimination des évolution imposées, le coût de l'OSS. Toutes résultent en un bénéfice par une réduction importante des coûts informatiques et télécommunications (I.T.).

Le guide recommande :

- avant de commencer, avoir un compréhension claire des motifs de la migration ;
- s'assurer qu'il existe un support actif du changement de l'équipe I.T. et des utilisateurs ;
- s'assurer qu'il existe un patron volontaire pour la migration - le plus élevé possible en hiérarchie ;
- construire une expertise et des bonnes relations avec le mouvement OSS ;
- commencer avec des systèmes non critiques ;
- s'assurer que chaque étape de la migration soit gérable.

La migration de systèmes I.T. fournit une opportunité de ré-ingénieurer ceux-ci pour les faire correspondre aux nouvelles exigences qui leur sont demandées. Les questions auxquelles répondre incluent :

- comment assurer l'interopérabilité des systèmes ?
- comment intégrer les utilisateurs mobiles ?
- comment identifier en sécurité les utilisateurs distants ?
- comment construire des systèmes gérables ?

Par-dessus tout : comment garantir que la sécurité soit conçue dès le départ et non pistée après conception ?

Pour les serveurs, l'opportunité de l'OSS est bien comprise et largement déployée. La migration de serveurs peut généralement être réalisée sans effet de bord sur les utilisateurs. C'est en principe un bon point de départ.

Le déploiement d'OSS sur les postes de travail offre les meilleures opportunités d'économies pour de nombreuses organisations. Lors de la migration de ceux-ci, les nouvelles applications OSS devront interopérer avec les applications existantes. En particulier, la manière dont le travail de groupe sera interopérable entre l'OSS et les systèmes propriétaires doit être résolue.

Lors du remplacement de logiciels propriétaires de bureautique, des prototypes doivent être testés pour s'assurer qu'ils produisent des résultats corrects. Les macros doivent être ré-écrites - de préférence sous forme de scripts. Les applications pour lesquelles aucun équivalent OSS n'existe peuvent tourner sur des clients légers. Au cours du temps, les applications de poste de travail peuvent être remplacées progressivement par leurs équivalents OSS.

Bien que le guide suppose un changement complet vers l'OSS, il est vraisemblable qu'un environnement hétérogène soit construit, en particulier parce que la migration de milliers de postes de travail prend du temps. Des mélanges d'applications OSS et propriétaires sont aussi vraisemblables car les applications OSS de remplacement peuvent ne pas être toujours disponibles ou utilisables. Actuellement,

cela est particulièrement vrai pour le remplacement des fonctions de travail de groupe de Microsoft Exchange. Cependant, il existe suffisamment d'applications OSS de qualité pour rendre la migration réalisable.

Il est important de s'assurer que les décisions prises maintenant, même si elles n'ont pas de relation directe avec une migration, n'enferment pas une administration dans des formats de fichiers ni des protocoles propriétaires.

OSS est une technologie de rupture. Il permet un changement fondamental dans la manière dont les organisations fournissent des services I.T. C'est une avancée d'une industrie de produit vers une industrie de service. Le logiciel OSS ne coûte rien à installer. Le point particulier en est de déterminer où trouver le support. Il existe de nombreuses entreprises tierces de maintenance ainsi que des éditeurs de distributions. Cependant si votre attitude vis-à-vis de l'informatique et des télécommunications est : « Qui puis-je attaquer si quelque chose ne va pas ? », alors il est possible que l'OSS ne soit pas pour vous. Une compréhension de la dynamique du mouvement OSS est indispensable. Il est hautement souhaitable de savoir comment communiquer avec la communauté OSS.

4 Méthodologie

Tout exercice de migration consiste généralement en :

1. une phase de collecte d'informations et de définition de projet comportant :
   A) une description de l'ensemble des conditions initiales consistant, par exemple, en :
      a) architecture(s) système,
      b) applications et leurs données associées,
      c) protocoles et standards utilisés,
      d) matériel,
      e) environnement physique telque bande passante réseau et localisation,
      f) pré-requis sociaux tels que les langues et ensembles de particularités des équipes ;
   B) un ensemble de conditions cibles du même niveau de détail,
   C) une description de la démarche permettant le passage des conditions existantes à celles planifiées ;
2. une justification de la migration incluant le calcul des coûts associés ;
3. une ou plusieurs phases pilotes conçues pour tester le plan et la justification. Les données issues de ces pilotes peuvent ensuite être réinjectées dans le modèle de coûts utilisé dans le plan ;
4. déroulement du plan ;
5. suivi et actualisation de l'expérience réelle acquise par rapport au plan.

Le contenu de la première phase ci-dessus définit ce qui sera désigné par le terme *scénario* dans ce guide qui décrit alors comment migrer dans ces circonstances.

Néanmoins, pour que ce guide reste lisible et utile en pratique, il a été nécessaire de procéder à un certain nombre de suppositions simplificatrices, sinon le total des combinaisons possibles serait devenu inexploitable.

Nous avons choisi l'un des nombreux ensembles de conditions cibles possibles et simplifié la description des ensembles des conditions initiales. L'environnement cible est décrit à la section 8.2. Avec l'environnement cible standard supposé, un scénario est défini par référence à l'environnement initial simplifié et le chemin de migration vers la cible.

De nouveaux chapitres pourront être ajoutés à ce document autant que nécessaire dans l'avenir. À partir du chapitre 14, chaque chapitre contient une description raisonnablement détaillée d'un scénario avec son chemin de migration, y compris les cas de migrations partielles. Chaque chapitre devra être mis à jour au fur et à mesure du gain en expérience à partir de migrations réelles.

De plus, une feuille de calcul associée permet de faciliter le calcul des coûts associés cités ci-dessus. Celle-ci permet de comparer les coûts de chaque scénario à ceux de la cible ainsi qu'à ceux de migration.

Les détails disponibles dans les études de cas disponibles sont très limités. Seul un petit nombre de celles-ci (une liste est fournie en annexe A), contenant quelques détails autres qu'un communiqué de presse généraliste fut trouvé. Cela indique que l'essentiel de ce guide est fondé sur l'expérience de netproject et de ses consultants ainsi que sur leurs échanges avec les gens ayant réalisé une migration sans publier leurs résultats.

Le très grand nombre de combinaisons de conditions initiales et cibles, ainsi que les nombreuses manières différentes de progresser des unes vers les autres rend impossible de couvrir l'ensemble des possibilités. Ce guide doit donc être considéré comme indicatif de ce qui peut être réalisé plutôt que prescriptif de ce qui doit être fait. Il peut être utilisé comme point de départ dans le processus de migration. Il ne faut pas en attendre une réponse dans toutes les circonstances.

Il est supposé que la migration a une cible intégralement OSS dans la mesure du possible et du sensé, cependant de nombreuses raisons peuvent imposer la conservation de systèmes propriétaires. Les possibilités de migrations partielles sont aussi abordées.

Deuxième partie : Règles de gestion

5 Vue générale de la migration

Une grande part de ce qui doit être réalisé lors d'une migration d'un environnement propriétaire vers l'OSS est identique dans tous les cas - par exemple depuis MS-Windows NT vers MS-Windows 2000. Même pour ce genre de changements mono-éditeur il ne peut être présupposé que les formats de fichiers, par exemple, soient portables et des tests adaptés sont nécessaires avant de réaliser tout opération d'envergure. Toutes les migrations doivent être fondées sur une planification précautionneuse.

Ce guide n'est pas conçu pour être un manuel de gestion de projet et il est supposé que l'administration dispose de compétences suffisantes pour être capable de gérer correctement la migration. La description ci-dessous est conçue pour mettre en évidence les points saillants d'une migration vers l'OSS.

Idéalement, le processus de migration doit consister en les parties qui suivent. Certaines d'entre elles peuvent être menées en parallèle, notamment 2, 3 et 4.

On peut parfois s'apercevoir que des modifications seront nécessaires sur l'environnement existant avant qu'une migration vers l'OSS puisse être concevable. C'est pourquoi il est recommandé que même les administrations qui n'ont pas de plan immédiat de migration mais souhaitent conserver cette option dans le champ des possibles, exigent exclusivement des standards multi-éditeurs et fondent leur infrastructure sur ceux-ci (voir aussi le chapitre 7.3ci-dessous).

1. monter une équipe avec les compétences nécessaires et un appui managérial. Cet appui est important pour éviter les résistances au changement depuis la norme des systèmes propriétaires. Il devra être suffisant pour permettre au moins la mise en place de pilotes représentatifs, ainsi un cas concret de base sera-t-il sans doute nécessaire avec peut-être un autre plus détaillé par la suite lorsque plus d'informations seront disponibles.
2. comprendre l'environnement cible, aussi bien le logiciel OSS que l'architecture de base (cf. chapitre 8), ainsi que les options et choix possibles. Cela implique la formation de l'équipe existante, du recrutement ou le recours à des consultants. Cela entraîne un certain coût initial et nécessite donc un appui managérial suffisant. Parfois l'on s'attend à ce qu'un logiciel gratuit puisse être compris et utilisé à coût zéro. **Ce n'est pas le cas**.
3. la migration est une opportunité pour re-visiter l'architecture de base aussi bien que les logiciels applicatifs. L'architecture préconisée au chapitre 26 s'appuie sur un contrôle centralisé et dispose d'un certain nombre d'avantages détaillés dans ledit chapitre. Il peut y avoir des implications financières de ces changements qui doivent être prises en compte.
4. il est très important que l'OSS soit compris. Un certain nombre de points doivent être totalement maîtrisés avant toute prise de décision :
   A) les implications des licences OSS doivent être clairement perçues en particulier si l'administration peut être amenée à distribuer des modifications aux logiciels. Se reporter aux documents indiqués dans l'introduction pour les détails.
   B) lorsqu'il existe plusieurs choix pour une seule fonction - il existe par exemple au moins trois tableurs de qualité en OSS - les administrateurs doivent peser le pour et le contre de chaque produit.
   C) il faut étudier les différences entre les distributions. Certaines sont appuyées par des entreprises commerciales qui fournissent un support et des mises à jour. Certaines ont des caractéristiques spécifiques (par exemple, Gentoo propose une distribution fondée sur le ***code source*** qui permet à l'administration d'adapter aisément le logiciel pour le faire coller au mieux à ses besoins spécifiques). Toutes ces différences doivent être étudiées avant qu'un choix soit effectué.
   D) les administrateurs doivent déterminer le niveau de support nécessaire. Un support payant peut être obtenu dans certains cas auprès des développeurs de l'application ou de la distribution. Dans le cas contraire, des entreprises tierces peuvent fournir un support en raison de la disponibilité du code source et de nombreuses entreprises proposent un tel support.
   Il y a une différence marquante par rapport au marché du logiciel propriétaire où un support détaillé ne peut être fourni que par les entreprises qui ont le privilège d'avoir accès aux sources. Cela devient important si l'éditeur propriétaire cesse son activité sans fournir le code source.
   Si tout le reste échoue, de nombreuses applications disposent de listes de diffusion actives sur

lesquelles les questions et demandes d'assistances reçoivent des réponses de personnes qui s'intéressent à celles-ci. La présence d'une liste de diffusion active et d'une communauté d'utilisateurs est souvent l'un des critères initiaux de sélection de composants logiciels.

5. analyser les systèmes existants. Cette donnée ne sera pas utile seulement pour réaliser la migration proprement dite mais elle sera aussi nécessaire pour bâtir un modèle de coût total de possession (TCO - Total cost of ownership) pour un cas concret détaillé.
   Il faut compiler l'inventaire de :
   A) pour chaque application utilisée :
      a) nom de l'application, numéro de version et point de contact,
      b) nombre d'utilisateurs,
      c) système d'exploitation, liste des systèmes d'exploitations possibles y compris les environnements de type *Citrix*,
      d) interdépendance avec d'autres applications aussi bien sur le client que sur le serveur,
      e) matériel nécessaire, en particulier matériels non standard ou spécifiques,
      f) protocoles de communications utilisés,
      g) formats de fichiers nécessaires,
      h) internationalisation et localisation nécessaires (il peut être nécessaire de gérer plusieurs langues et devises par exemple) ;
   B) contraintes sur les données (à interpréter au sens large y compris, par exemple, textes et tableaux, données sonores/vocales et visuelles ainsi que les bases de données) ; plus généralement, tout ce qui est destiné à être traité ou stocké par un ordinateur :
      a) contraintes d'interfaçage avec des systèmes ou utilisateurs hors du contrôle de l'administration,
      b) contraintes d'exploitabilité future des données (NdT : obligations de conservation à 30 ans par exemple), existence d'un répertoire des données de référence à accepter, nécessité d'applications spécifiques pour l'exploitation de celui-ci... Les données peuvent être divisées comme suit :
         i. données jetables - les jeter,
         ii. données à conserver existant dans un format ouvert tel que PDF ou PostScript ou pouvant aisément être transcrites dans l'un de ces formats. Le coût de transcription doit être évalué,
         iii. données à conserver existant dans un format propriétaire difficile à transcrire en format ouvert. Ces données peuvent nécessiter le maintien d'applications propriétaires spécifiques ; le coût de celles-ci doit être évalué, ainsi que le nombre de copies nécessaires sur la base de la fréquence d'utilisation (par exemple, si l'information est rarement utilisée, une seule copie sur une machine centralisée peut suffire). Il peut également être nécessaire de maintenir un matériel spécifique pour utiliser ces applications ;
   C) contraintes de sécurité
      a) quel est le système actuel d'assignation des noms d'utilisateurs et mots de passe ? Existe-t-il une structuration des noms d'utilisateurs ? Laquelle ? Quelle est la politique d'expiration des mots de passe ?
      b) certains systèmes nécessitent-ils une authentification supérieure à un couple utilisateur/mot de passe ?
      c) Quelles sont les règles administratives et gouvernementales d'utilisation des ordinateurs (par exemple, existe-t-il des restrictions quant à l'utilisation d'Internet et du courriel) ?
      d) Y a-t-il des dispositions sécuritaires qui nécessitent l'utilisation de matériel ou de logiciel spécifique ?
6. Créer un cas concret de migration détaillé ; celui-ci sera fondé sur les données rassemblées ci-dessus et comportera notamment :
   A) le TCO de l'environnement existant pour une période de temps raisonnable (5 ans par exemple),
   B) le TCO d'environnements différents et le coût de migration vers chacun pour la même période,
   C) les forces et faiblesses de l'environnement existant et de chaque alternative (la feuille de calcul associée peut aider à effectuer la comparaison) ;
7. consulter les utilisateurs. Leur expliquer le raisonnement sous-jacent à la migration et les effets qui leur seront perceptibles. Prendre sérieusement en compte leurs besoins et leur permettre de « jouer » avec les nouveautés aussi tôt que possible. Plus tôt les utilisateurs sont impliqués, mieux la migration

est acceptée. Il peut y avoir des nécessités légales dans certains pays mais cela doit être fait dans tous les cas pour faciliter l'introduction de ce qui peut constituer une modification significative dans le travail quotidien.
Créer un pôle d'assistance qui puisse répondre aux besoins des utilisateurs. Plus tard, lorsque la migration est en cours, celui-ci pourra résoudre les problèmes et devenir un centre d'excellence et de bonnes pratiques. Créer un site intranet avec une section « astuces et guides » qui puisse être mise à jour par les utilisateurs eux-mêmes. Il est important que les utilisateurs se sentent impliqués et le site donnera aussi au pôle d'assistance une idée du genre de problèmes auxquels sont confrontés les utilisateurs.

8. en supposant réalisé le cas concret, commencer avec des projets pilotes à petite échelle, de préférence dans un environnement isolé avec un petit nombre d'utilisateurs. Ceux-ci fourniront notamment
   A) des données pour affiner les modèles de TCO,
   B) des réactions d'utilisateurs qui permettront de faciliter l'introduction d'autres systèmes,
   C) la validation ou la modification de l'architecture cible et du cas concret ;
9. décider de la vitesse du processus de migration ; il est vraisemblable que les deux systèmes doivent cohabiter côte à côte durant un certain temps. Il est important d'avoir une stratégie de transition permettant à ceux-ci de fonctionner simultanément afin que les activités de production puissent continuer correctement durant la période de migration. Le remplacement de la dernière machine peut prendre beaucoup de temps (ou ne jamais arriver) donc la faisabilité de la coexistence peut être très importante. Les principales options sont :
   A) **Big Bang** : tous les utilisateurs passent au nouveau système le même jour. Dans la pratique, il est vraisemblable que la migration soit planifiée sur une fin de semaine ou durant les vacances. L'avantage réside dans l'absence de structures de double accès et de double maintenance par les équipes techniques. Les inconvénients résident dans un niveau de risque très élevé et dans la très forte mobilisation de ressources durant la migration. Ce schéma semble ne pouvoir être appliqué que dans les petites administrations.
   Dans la mesure du possible, **ÉVITEZ LA MIGRATION BIG BANG**. Ce type de migrations nécessite le contrôle de tant de variables qu'elles échouent en général. Si c'est le cas, ce n'est pas, en général, du fait d'une défaillance de l'OSS mais de celle du management.
   B) **Transition progressive par groupes** : les utilisateurs migrent par groupes. On fait migrer en général des groupes fonctionnels complets afin de minimiser les partages de données et les difficultés de travail de groupe. Les risques peuvent être limités et les ressources gérées par l'adéquation de la taille des groupes. Il peut être possible de réaliser un remplacement du matériel par roulement dans le même temps, en effectuant une mise à jour des postes de travail retirés d'un groupe avant de les attribuer au groupe suivant.
   C) **Transition individuelle** : similaire à la transition progressive par groupes mais avec une taille de groupe d'une personne. Cette méthode au goutte-à-goutte mobilise peu de ressources mais elle est peu efficace et sans doute peu adaptée aux grandes administrations. Elle peut cependant être appropriée aux projet pilotes.
10. Réaliser la migration de l'ensemble de l'administration. Cela implique une formation ultérieure des utilisateurs et de l'équipe technique.
11. Suivre les retours des utilisateurs et résoudre tous les problèmes qui surviennent. Certains besoins utilisateurs peuvent être suffisamment obscurs pour n'avoir pas été prévus à l'avance ni découvert durant les phases pilotes. S'assurer que des ressources suffisantes restent disponibles pour répondre à ce type de besoins après la transition.

À tout instant, il peut être découvert une impossibilité de migration. Cela peut arriver par exemple en raison de l'existence d'applications critiques qui ne fonctionnent pas de manière satisfaisante dans un environnement OSS et dont le coût de réécriture serait trop élevé.

6 Critères humains

Ce guide n'est pas destiné à servir de guide de gestion des ressources humaines et les administrations ont d'ores et déjà été confrontées à de nombreux de ces aspects. Elles disposent en interne de compétences considérables pour les gérer en sympathie et donc l'équipe des ressources humaines devrait être impliquée dès les étapes préliminaires. L'intention ici est simplement de mettre en exergue le genre de problèmes qui ont été rencontrés dans d'autres sites lors de migrations vers l'OSS.

Il est très important que toute l'équipe soit consultée et informée des évolutions. Une manière de réaliser cela est de créer un intranet dont la mise à jour soit facile et qui dispose d'une section pour les retours d'utilisateurs.

L'accès à la formation est très important. Certain sites ont permis à leurs utilisateurs de décider eux-mêmes de leur participation alors que d'autres l'ont imposée. Le choix dépend de la culture d'entreprise et du contenu des cursus. Les manuels et la documentation générale sont souvent seulement disponibles en anglais et cela peut poser un problème avec certaines équipes. La traduction peut être considérée comme un coût de migration mais il restera le problème de la maintenance des traductions.

L'interface utilisateur OSS et notamment *Gnome* et *KDE* proposent un choix de langues mais la traduction peut ne pas être complète pour certaines options de menus et certains écrans d'aide qui restent en anglais. *Gnome* en particulier a de bonnes fonctions d'accessibilité pour les déficients visuels. Toutes les applications n'ont pas non plus de support complet d'internationalisation. Cela change rapidement cependant et la structure que permet l'utilisation de langues autres que l'anglais est en place si l'administration souhaite l'utiliser.

- **Peur de l'inconnu**

  L'utilisation d'OSS sera une nouveauté absolue pour de nombreux utilisateurs et administrateurs système. La peur de l'inconnu, naturelle, les fera résister à l'introduction de l'OSS en raison de sa nouveauté.

  Certains utilisateurs, naturellement plus curieux, pourront être très heureux d'essayer les nouveautés et ce sont ceux à qui présenter l'OSS en premier. L'expérience acquise indique qu'une fois sortis de leur réserve, les gens trouvent que l'OSS n'est pas significativement différent à utiliser que le logiciel propriétaire et sont raisonnablement satisfaits de l'utiliser. Il est ainsi vraisemblable que ce groupe initial d'utilisateurs passeront à l'OSS avec enthousiasme. Dans tous les cas ce sont ceux qui fourniront sans doute les retours les plus utiles.

  Ce premier groupe d'utilisateurs peut être utilisé dans les essais pilotes et une fois expérimentés, il peuvent contribuer à accompagner et éduquer leurs collègues. Dans tous les cas, dans la seconde phase, les utilisateurs qui resteront plus réservés nécessiteront un support plus avancé sous la forme de pôles d'assistance, intranets et utilisateurs locaux expérimentés.

  Le même processus peut être utilisé pour les administrateurs système mais le niveau de formation sera sans doute significatif si l'environnement propriétaire existant n'est pas de type UNIX. L'équipe système en particulier doit voir ses peurs évacuées à une étape précoce. Ils deviendront un point focal pour tous les problèmes qui surviendront et s'ils ne croient pas en le projet, ils ne seront pas capable d'encourager les utilisateurs de manière positive.

- **Effet de dilution de CV**

  Aussi bien l'équipe système que les utilisateurs peuvent penser que la non-utilisation de logiciel « standard » risque d'altérer leur capacité de développement de leur carrière. C'est un problème complexe qui nécessite une gestion précautionneuse. L'administration ne souhaitera pas apparaître sur-outillée dans cette approche mais jusqu'à une adoption massive de l'OSS ce problème peut survenir assez souvent.

- **Connaissance = pouvoir**

Les gens qui connaissent les systèmes existants et leur paramétrage disposent d'un certain pouvoir et peuvent être très réfractaires à son abandon si l'environnement OSS est très différent de l'existant. Cela aussi nécessite une gestion précautionneuse car ceux-ci jouent un rôle critique dans l'exploitation des systèmes existants. Il peut être nécessaire de les intégrer aux premières formations afin de leur permettre de maintenir leur position dans l'organisation.

7 Faciliter les choses

Un certain nombre de considérations peuvent faciliter l'introduction de l'OSS.

7.1 Introduire les nouvelles applications dans un environnement familier

De nombreuses applications OSS fonctionnent sur des systèmes d'exploitation propriétaires et cela donne une opportunité d'introduction de celles-ci sans réaliser une modification intégrale de l'environnement. Par exemple, *OpenOffice.org*, *Mozilla* et *Apache* fonctionnent sous *MS-Windows* et peuvent ainsi être utilisées en remplacement de *Ms-Office*, *MS-Internet Explorer* et *MS-IIS* respectivement. En plus d'être moins intrusive, cette approche permet de jauger les réactions des utilisateurs à une échelle réduite et la planification de la formation peut être fondée sur l'expérience réelle. De plus, des problèmes tels que la conversion de formats de fichiers, macros et modèles peuvent être simplifiés si l'ancienne application reste disponible un moment.

Cette approche implique que le choix de l'application dans l'environnement cible soit limité à celles qui fonctionnent dans l'environnement existant. Par exemple le navigateur cible peut être *Galeon* mais *Mozilla* est le seul qui fonctionne à la fois sous *MS-Windows* et *GNU/Linux*.

7.2 Commencer par le plus simple

Effectuer tout d'abord des modifications qui ne perturbent pas la population utilisatrice, c'est-à-dire d'abord sur le serveur. Celles-ci fourniront ainsi une plate-forme pour l'introduction ultérieure des modifications des clients. De nombreuses modifications serveur seront compatibles avec l'environnement existant, ainsi l'effet perturbant sera minimisé.

Par exemple, les serveurs de noms DNS, les serveurs DHCP et les serveurs de bases de données avec des moteurs propriétaires tels qu'Oracle sont tous candidats à leur remplacement par leur équivalent OSS et interopèrent avec le reste du système existant comme auparavant. Le détail de cela est discuté plus loin.

Certaines applications comme *Samba* ne seront pas utilisées dans un environnement purement OSS mais permettent la coexistence de l'ancien environnement propriétaire et de l'OSS. L'utilisation précoce de celles-ci peut être très efficace pour diviser l'environnement en blocs gérables.

7.3 Penser plus loin

Réduire dès maintenant tout ce qui pourra rendre une migration ultérieure plus difficile. Par exemple :

1. imposer que le développement web maison ou par des prestataires produise un contenu visualisable sur tous les navigateurs actuels, en particulier les navigateurs OSS. Cela doit être une bonne pratique dans tous les cas car l'administration ne doit pas nécessiter de logiciel spécifique pour accéder au contenu en ligne. Des outils tels que *weblint* facilitent le contrôle de compatibilité des pages web ;
2. décourager la prolifération incontrôlée des macros et scripts dans les documents et feuilles de calculs ; chercher d'autres moyens de réaliser la fonctionnalité. Cela aussi doit être une bonne pratique car l'utilisation de ceux-ci est un biais classique d'infection par les virus. De même, les macros peuvent facilement être utilisées pour dérober des informations et détourner des documents (par exemple, elles permettent que le document affiche des éléments différents à l'écran et à l'impression) ;
3. imposer l'utilisation des formats de fichiers ouverts standard, par exemple PostScript et PDF.

   Il existe un certain débat sur le caractère ouvert des standards PostScript et PDF. Il s'agit plus d'un débat sur la définition stricte et en particulier sur le «qui contrôle le standard ? ». En réalité, il s'agit des seuls formats standard largement utilisés actuellement, dont les définitions soient publiques et qui puissent être utilisés sans restriction.

Des tentatives pour créer de véritables standards ouverts fondés sur XML sont apparues et *OpenOffice.org* en est une implantation. Cependant le seul critère technique XML ne rend pas le format ouvert.

En particulier, il faut éviter d'utiliser des formats de fichiers propriétaires pour les fichiers qui ne sont prévus que pour la lecture et non la modification par le destinataire. À nouveau, il s'agit d'une bonne pratique car ceux-ci sont un vecteur de diffusion des virus. L'utilisation de tels formats verrouille l'administration avec un éditeur pour une durée considérable. De plus, ces formats peuvent contenir des quantités considérables de **méta-données** y compris, en particulier, du texte supprimé préalablement qui peut se révéler embarrassant pour l'administration s'il est vu par des tiers ; il n'est pas difficile d'accéder à ces méta-données.

4. lors de la rédaction de documents à plusieurs, utiliser le format le plus petit dénominateur commun. Par exemple, utiliser le format *MS-Word 97* de préférence au format *MS-Windows 2000*. Cela augmente la probabilité de participation par des applications OSS.
5. utiliser des protocoles standard. Les protocoles standard ouverts sont ceux définis comme libres de brevets et disposant d'une implantation OSS. Différents ensembles de standards nationaux existent tels que E-gif au Royaume-Uni, OSOSS aux Pays Bas et SAGA en Allemagne. L'objectif et le contenu de ces différents cadres est assez variable mais en général ils sont imposés.
6. développer des systèmes fondés au minimum sur un modèle 3-tiers (voir section 8.1) dans lequel le code applicatif est indépendant de l'interface homme-machine et des méthodes d'accès aux données. Par exemple, si possible, ayez une interface par navigateur qui fonctionne avec un navigateur OSS. Le développement d'applications selon ce principe modulaire en facilite la migration élément par élément. Cela ne réduira pas seulement l'échelle de toutes les phases de migration mais aussi le risque de défaillance. Les applications à client monolithique traditionnel sont notoirement difficiles à maîtriser.
7. imposer que tout nouveau développement soit portable. Cela implique l'utilisation de langages portables standardisés tels que le C ANSI, Java, Python et Perl ainsi que l'utilisation exclusive de bibliothèques et boîtes à outils I.H.M. multiplates-formes telles que *wxWindows* (http://www.wxwindows.org/) et *FOX toolkit* (http://www.fox-toolkit.org/). Éviter les langages et API spécifiques. Éviter de développer des applications nécessitant la présence d'autres applications propriétaires.
8. éloigner les utilisateurs des outils de courriel qui utilisent des formats de stockage de courriel propriétaires et communiquent avec les serveurs à l'aide de protocoles propriétaires. De nombreuses applications stockent le courriel à l'aide d'IMAP. Si possible, trouver un moyen de stocker les carnets d'adresses et les informations calendaires dans un format ouvert.

Troisième partie : guide technique

8 Architecture de référence

8.1 Architectures génériques

Une manière pratique de décrire une architecture informatique est celle appelée modèle 3 tiers. Ce modèle sépare trois fonctions principales qu'une application réalise en général lorsqu'elle est utilisée par un humain (i.e. lorsqu'il ne s'agit pas d'une application serveur pure ou batch). Elle est présentée dans le diagramme suivant (plus d'informations sur http://www.corba.ch/e/3tier.html/).

Les flèches représentent le passage d'informations entre les trois parties. Ces flux doivent idéalement reposer sur des standards ouverts bien définis. Ainsi une application n'a à gérer que l'exécution de la logique métier en laissant les deux autres fonctions à des composants standard. Le bénéfice net réside dans une simplification du code de l'application et dans son exécutabilité dans différents environnements en raison de la réduction de sa dépendance à des accès machine spécifiques.

Ce modèle 3 tiers a été généralisé au modèle n tiers, dans lequel les composants sont encore plus affinés et réalisé habituellement à l'aide de technologies objets ou composants.

De nombreuses applications client-serveur ont malheureusement été développées sur un modèle 2 tiers dans lequel le code applicatif et l'interface homme-machine (I.H.M.) sont imbriqués. En conséquence, la migration de telles applications et souvent considérablement plus difficile que celle des applications 3 tiers. La raison en est que l'I.H.M. a des chances de nécessiter des modifications et que le code de celle-ci dans les applications 2 tiers est souvent imbriquée dans le flux de la logique métier.

La communication entre les trois parties dans un modèle 3 tiers s'appuie en principe sur des protocoles qui permettent à chaque partie, lorsque c'est nécessaire, de s'exécuter sur des machines différentes de celles des deux autres. Parfois, ces parties peuvent elles-mêmes être séparés sur différentes machines. Le choix de l'emplacement de chacune de ces parties donne accès à différentes architectures génériques.

Du point de vue du poste de travail, sur lequel s'exécute au minimum une partie du code de l'I.H.M., les extrêmes sont :

1. Client ultra-léger

   C'est lorsque le poste de travail n'exécute que du code I.H.M. Typiquement, il ne dispose d'aucune mémoire long terme telle que disque dur ou disquette. Le code applicatif et l'accès aux données sont exécutés à distance. Un terminal X, un terminal texte de type VT100 ou un navigateur embarqué en sont de bons exemples.

2. Client ultra-lourd

   C'est lorsque le code et les données sont placés sur le client sans connexion réseau.

   Les termes de client léger et lourd situent des configurations entre ces deux extrêmes.

On trouve une variante de ces architectures lorsque le code applicatif est stocké sur un serveur puis téléchargé à la demande pour exécution sur le client. C'est la manière dont fonctionnent les appliquettes Java par exemple. Une autre méthode implique le stockage du code applicatif sur un serveur avec un accès par le client comme s'il était stocké en local. Cela nécessite l'utilisation d'un système de fichiers réseau tel que NFS et impose aussi à tous les postes de travail la même architecture processeur.

Le choix d'une architecture pour une application particulière dépend de :

1. **la bande passante du réseau vers les serveurs et ce que celle-ci doit transporter**. Si le poste de travail n'est pas ultra-lourd, le réseau devra transporter les contrôles d'I.H.M., des données ou du code téléchargé. Dans certaines circonstances, la charge engendrée par un seul ou plusieurs postes de travail peut excéder la capacité du réseau ;
2. **la latence acceptable pour l'utilisation de l'application**. Lorsqu'un humain interagit avec le poste de travail en appuyant sur des touches ou en déplaçant la souris, le temps de réaction et de rafraîchissement à l'écran s'appelle la latence. Pour certaines applications telles que la simple saisie, des latences élevées peuvent être acceptables mais pour des applications hautement interactives telles que le dessin, les latences doivent être faibles. Celle-ci dépend de la capacité de tous les segments réseau entre l'I.H.M. et l'application ainsi que de la capacité de la machine exécutant le code applicatif. Pour obtenir les latences les plus faibles, l'application doit être exécutés sur la même machine que celle de l'I.H.M. et cette machine doit être suffisamment performante pour que l'application s'exécute correctement ;
3. **la politique de sécurité**. Si les données résident sur des postes de travail disséminés dans l'administration, cela signifie qu'un vol ou un accès non autorisé à une machine dans un environnement non sécurisé peut aboutir à une perte ou à une divulgation à des tiers ou à des personnes non autorisées. Cela peut n'être pas un problème pour des informations de niveau faible correctement sauvegardées mais sinon, cela peut contrarier la politique de sécurité de l'administration concernant le droit d'accès aux données. À l'inverse, faire transiter des données sur un réseau non chiffré peut produire le même problème ;
4. **la politique de sauvegarde**. Si les données résident sur des postes de travail disséminés dans l'administration, il faut, soit implanter un mécanisme de sauvegarde centralisé, soit distribuer la responsabilité des sauvegardes à de nombreuses personnes, probablement les utilisateurs eux-mêmes. Un schéma de sauvegarde centralisé sera complexe et nécessitera une forte bande passante et la coopération des utilisateurs (qui doivent, par exemple, se rappeler de ne pas éteindre leur machine lors des périodes de sauvegarde planifiées) ;
5. **la structure de l'application**. Si l'application comporte du code d'I.H.M., elle doit, soit s'exécuter sur le poste de travail ou sur un serveur si ledit code est divisé entre le serveur et le poste de travail. Par exemple, le code d'I.H.M. est situé sur le serveur pour un terminal IBM 3270 ou DEC VT100 ou un navigateur embarqué. *Citrix*, *MS-Windows Terminal Server* et le *système X Window* divisent le code d'I.H.M. entre le serveur et le client ;
6. **la capacité du poste de travail à exécuter le code**. Plus le poste de travail est chargé, plus il doit être puissant (et donc, cher) ;
7. **la capacité du poste de travail à stocker les données**. Certaines applications nécessitent l'accès à de grandes quantités de données qui ne peuvent être gérées que par des serveurs spécialisés.
8. **la performance des serveurs**. Si l'application est exécutée sur un serveur au lieu du poste de travail, ce serveur doit être suffisamment puissant pour exécuter toutes les instances de l'application lorsque tous les utilisateurs sont connectés. Cela peut nécessiter un dimensionnement largement supérieur pour assurer le fonctionnement en conditions extrêmes. De plus, le nombre de postes de travail supporté par un nombre donné de serveurs tend à présenter des « sauts » ; cela veut dire que l'ajout de quelques nouveaux postes de travail peut conduire à l'acquisition d'un serveur beaucoup plus gros ;
9. **le coût total de l'implantation**. Comme pour tout problème d'ingénierie, aucune solution n'est une panacée et un poste de travail physique peut fonctionner d'une manière pour une application et d'une autre pour une autre application.

8.2 Architecture de base de référence

L'architecture de base de référence (A.B.R.) utilisée dans ce guide a été choisie pour être représentative de la majorité des cas. Elle peut être étendue pour être allégée ou alourdie pour des applications spécifiques si nécessaire.

En réalité, l'architecture utilisée par une administration sera vraisemblablement une combinaison de plusieurs architectures choisies en fonction des applications spécifiques.

l'A.B.R. peut être caractérisée par un « poste de travail sans état » dans lequel :

1. toutes les applications sont stockées et s'exécutent sur le poste si possible ;
2. aucune donnée persistante n'est stockée sur le poste ;
3. toute l'authentification et les autorisations sont contrôlées par des serveurs centraux ;
4. l'administration système est centralisée ;
5. l'objectif est que les postes soient « plug and play » et ne nécessitent aucun support local.

Les applications s'exécutent localement pour simplifier les problèmes de latences créées par une exécution centralisée et l'A.B.R. suppose que la bande passante soit suffisante pour conserver les données centralisées. De plus, il est supposé que les postes soient globalement similaires, permettant à quiconque de se connecter sur toute machine à laquelle il a légitimement accès. Il doit y avoir une organisation de l'administration système forte pour maintenir la synchronisation de l'installation des logiciels sur les postes.

L'A.B.R. est configurée et administrée en central pour simplifier l'administration système, concentrer toutes les données importantes sur des serveurs centraux pour en faciliter la sauvegarde et la gestion et rendre le poste client « jetable » afin de réduire l'impact d'une défaillance d'un tel poste.

La conservation de données en local impose une identification des machines avec leur utilisateur. Cela pose des problèmes lorsqu'un utilisateur se déplace ou quitte l'organisation. Cela rend aussi l'emplacement du poste spécifique à l'utilisateur, ce qui rend difficile l'implantation de concepts tels que le « hot-desking ». La centralisation des données résout ces difficultés et rend plus souple l'utilisation des postes. Elle permet aussi la limitation de la taille du stockage local sur le poste.

La configuration « plug and play » des postes en simplifie l'installation et réduit ainsi le coût du support.

De nombreuses administrations utilisent déjà une variante d'une telle architecture pour toutes les raisons énumérées plus haut, ainsi l'A.B.R. semble être un choix raisonnable.

l'A.B.R. n'est pas adaptée pour les portables ni les postes sans connexion permanente au réseau local. Ces postes devront, soit être ultra-lourds, soit disposer d'un système de fichiers distribué permettant le fonctionnement déconnecté. Parmi les systèmes de fichiers OSS de ce genre on peut citer *CODA*, *OpenAFS* et *InterMezzo*, mais ils n'ont pas été testés par **netproject**.

9 Groupes fonctionnels

Le modèle de référence est fondé sur les groupes fonctionnels, définissant les principaux types d'activité informatique non spécialisés dans une administration. Des activités telles que la gestion de projet ou les systèmes d'informations géographiques ne sont donc pas abordées. Les activités laissées à l'écart devraient être en général celles utilisées par de faibles proportions de la population utilisatrice.

Les groupes fonctionnels sont divisés en groupes principaux et secondaires. Les groupes principaux représentent la fonctionnalité en termes de processus métier. Les groupes secondaires fournissent des services de support aux groupes principaux et ne sont donc en principe pas implantés seuls.

9.1 Groupes principaux

9.1.1 Bureautique

Il s'agit de la création, de la modification et de l'impression de fichiers contenant des données en format commercial standard telles que les lettres et rapports, les feuilles de calcul et les présentations. Il faut des outils pour le traitement de ces fichiers. Les formats Microsoft de facto ***.doc**, ***.xls** et ***.ppt** doivent pouvoir être lus et écrits de manière très fiable, ainsi que les formats ouverts tels que PDF. Les langues et particularités locales européennes telles que devise et alphabet doivent être supportées.

9.1.2 Courriel

C'est la création, la réception et l'affichage de courrier électronique, incluant le support du courriel sécurisé tel que S/MIME.

9.1.3 Agendas et groupes de travail

C'est la création et le suivi d'agendas et carnets d'adresses personnels et de groupes. Les agendas doivent permettre les réunions et la réservation de salles. Les carnets d'adresses doivent s'intégrer avec les autres groupes fonctionnels.

9.1.4 Accès et services web

C'est la possibilité d'accéder aux protocoles Internet et d'en afficher le résultat (cette fonction est normalement effectuée depuis un navigateur) ainsi que la capacité de créer du contenu et de le publier en interne comme en externe.

9.1.5 Gestion documentaire

C'est le stockage centralisé de documents avec un mécanisme d'accès performant.

9.1.6 Base de données

C'est la manipulation de données structurées dans des bases centrales et individuelles.

9.2 Groupes secondaires

Ces groupes sont généralement définis comme des services techniques et ne sont normalement pas implantés seuls. Ils incluent :

- systèmes d'exploitation ;
- serveurs de fichiers ;
- gestion des utilisateurs, authentification et autorisations ;

- détection de virus et courriel non sollicité ;
- sauvegarde et restauration ;
- gestion d'impression.

La liste complète est placée en chapitre 13.

9.3 Considérations générales

Les administrations ont des besoins spécifiques par rapport à ceux des entreprises. Certains sont imposés par la législation locale, nationale ou européenne. En particulier :

- elles doivent accepter des fichiers dans les formats publics habituellement utilisés (en réalité, cela inclut au minimum les formats Microsoft, mais peut aussi s'étendre à ceux de *WordPerfect* ou *Lotus Notes)* ;
- certaines applications échangent des informations entre le public et l'administration, ce qui doit être réalisé de manière sécurisée avec notification de distribution.

10 Le modèle de référence - sommaire

La grande quantité d'OSS disponible a pour conséquence la disponibilité de multiples applications pour de nombreuses fonctions. Le choix de l'application n'est pas toujours simple et parfois le choix final est déterminé par des préférences personnelles du décideur.

Le modèle de référence utilisé dans ce guide doit donc être considéré comme un exemple d'un système dont le fonctionnement est validé, plutôt que comme une recommandation générale d'un système.

Le guide aborde les points que les décideurs doivent prendre en compte et il peut arriver que ceux-ci mènent à des conclusions différentes, mais également valables. Dans tous les cas, les contraintes locales de l'administrateur peuvent conduire au choix d'un modèle différent.

Les choix possibles sont abordés au chapitre 12 pour les groupes principaux et au chapitre 13 pour les groupes secondaires.

Des sites de référence utiles présentent des listes d'applications OSS disponibles ainsi que leur capacité de remplacement d'applications propriétaires (http://linuxshop.ru/linuxbegin/win-lin-soft-en/ par exemple).

http://www.osafoundation.org/desktop-linux-overview.pdf détaille beaucoup des applications dont il est question plus bas.

Une des forces de l'OSS réside dans sa modularité et sa capacité d'adaptation à des besoins spécifiques. La modularité est la conséquence de la conformité aux interfaces ouvertes publiques. Malheureusement, cette souplesse peut parfois être son défaut, en noyant les administrateurs sous les choix possibles. De nombreuses organisations peuvent fournir aide et support, exactement comme dans le marché propriétaire.

Tous les groupes fonctionnels ne présentent pas de choix de référence, soit par manque d'études de cas adaptées, soit parce que netproject n'a pu valider correctement les produits correspondants dans le cadre de cette étude.

Les choix de référence détaillés sont placés en annexe D pour le poste de travail et en annexe E pour le serveur. De plus, netproject présente en annexe F un mode opératoire qui rend l'installation de postes de travail très simple.

10.1 Le poste de travail

Le système d'exploitation est GNU/Linux dans sa distribution RedHat 8.0. L'interface utilisateur est fondée sur Gnome (inclus dans la distribution) mais le bureau virtuel Ximian *XD2* (fondé sur Gnome) mérite un coup d'oeil. RedHat, dans cette distribution, a tenté de réunir les interfaces utilisateur de KDE et de Gnome.

Les systèmes de fichiers contenant les binaires (comme **`/usr`**) sont montés en lecture seul pour empêcher les utilisateurs de modifier leur contenu et les autres sont montés en « noexec » pour empêcher l'exécution de code depuis ceux-ci. Pour renforcer cela, l'interface ne doit pas permettre aux utilisateurs d'exécuter des programmes autres que par les interfaces prédéfinies. Cela implique que l'accès à la ligne de commande ou l'accès aux modifications des options de menu ou icones soit supprimés. Les machines ne doivent avoir de lecteurs ni de disquettes ni de CD/DVD afin de limiter (mains non empêcher) l'attachement local d'autres systèmes de fichiers.

Les systèmes de fichiers contenant des informations utilisateur à caractère volatil sont montés depuis un serveur NFS central. L'authentification est réalisée depuis une base de données LDAP centrale.

Les serveurs DNS centraux assurent la résolution des adresses IP et des noms et un serveur DHCP fournit la configuration réseau des postes au démarrage.

Les fonctions principales des postes de travail sont assurées come suit :

10.1.1 Bureautique

OpenOffice.org est choisi car :

- si l'administration migre depuis un environnement *MS-Windows*, *OpenOffice.org* peut s'exécuter dans cet environnement, ce qui donne aux utilisateurs un contact initial avec le nouveau logiciel dans un environnement familier ;
- il dispose de l'une des meilleures interprétations des formats de fichiers Microsoft ;
- il devient progressivement l'alternative de fait à *MS-Office*.

10.1.2 Courriel

Evolution est choisi en raison de son interface très similaire à celle de *Outlook* et donc de sa facilité d'apprentissage. Il dispose aussi de fonctionnalités très utiles telles que les dossiers virtuels. Cependant, *Evolution* n'interopère pas avec *Exchange* version 5.5 (bien que cela soit apparemment prévu), donc si l'administration utilise cette version, une solution de travail de groupe OSS web sera nécessaire pendant un moment (sauf si une solution propriétaire est choisie). *Kmail* de son côté supporte S/MIME (au contraire de la version actuelle d'*Evolution*) et a récemment évolué pour devenir un client du serveur de groupes de travail *Kgroupware*. Le choix est donc à évaluer finement et dépend des besoins immédiats et de la configuration existante.

10.1.3 Agenda et groupes de travail

Evolution est le choix pour les agendas personnels et la gestion de contacts. Le travail de groupe est difficile actuellement avec l'OSS puisque seules des solutions web sont réellement disponibles, bien que récemment le projet *Kgroupware* ait produit une solution dont le client est *Kmail*. Ainsi pour une solution intégralement OSS, un navigateur web sera utilisé pour le travail de groupe.

10.1.4 Accès web

Galeon est choisi car c'est un navigateur rapide mono-fonction dont l'interface utilisateur est agréable. *Mozilla* est une alternative si une solution complète incluant lecteur de courriel et carnet d'adresses est nécessaire. *Mozilla* est aussi le choix si l'administration utilise des postes sous *MS-Windows* et que le nouveau navigateur doive fonctionner dans l'environnement existant pour donner aux utilisateurs un contact initial avec le nouveau logiciel dans un environnement familier.

10.1.5 Gestion documentaire

Un gestionnaire de contenu web tel que *Aswad* serait le choix. Cependant, le projet *Aswad* semble arrêté, donc un nouveau choix est maintenant nécessaire.

10.1.6 Bases de données

Les bases de données personnelles sont fondées soit sur *MySQL*, soit sur un produit de travail de groupe web tel que *phpGroupWare*.

10.2 Les serveurs

Le système d'exploitation est GNU/Linux dans sa distribution RedHat 8.0. Ce choix serait différent pour des machines hautement sécurisées telles que les pare-feux où OpenBSD serait utilisé en conjonction avec GNU/Linux.

Les principales fonctions serveur sont assurées par :

10.2.1 Courriel

L'agent de transport de courriel (MTA - Mail Transport Agent) est *Exim* car c'est un produit complet comparable en fonctionnalités à *Sendmail* mais plus simple à maintenir. Il comprend aussi les options de *Sendmail* et peut donc être utilisé en remplacement de celui-ci. *Postfix* serait une alternative acceptable.

L'agent d'accès au courriel (MAA - Mail Access Agent) est *Courier-IMAP* qui est perçu comme plus simple que *Cyrus* en raison de son stockage plus simple. Cependant, *Cyrus* serait un bon choix.

10.2.2 Agenda et travail de groupe

phpGroupWare ou *the Horde* devraient constituer de très bonnes solutions web. Le nouveau *Kgroupware* n'a pas été évalué.

10.2.3 Services web

Apache est choisi en raison de sa position prédominante sur le marché et de sa vaste gamme d'outils associés et de support. D'autres serveurs peuvent être utilisés pour des tâches spécifiques ; *Zope* par exemple (voir 11.4.2 plus bas) peut être utilisé pour la gestion de contenu.

10.2.4 Gestion documentaire

Maintenant que *Aswad* (voir section 42) semble être arrêté, aucune solution de référence ne se détache. La discussion en section 11.5 indique qu'un certain choix perdure ici.

10.2.5 Bases de données

Pour les grandes bases de données à accès principal en lecture seule, *MySQL* ; pour d'autres sortes de bases de données, *PostgreSQL*.

11 Applications - groupes principaux

11.1 Bureautique

Le standard de fait est *MS-Office* qui inclut *MS-Word*, *MS-Excel*, *MS-PowerPoint* et *MS-Outlook* avec leurs formats de fichiers associés ***.doc**, ***.xls** et ***.ppt**. Ces formats ne sont pas ouverts et changent entre les versions de *MS-Office*. Même les propres produits de Microsoft ne peuvent garantir la capacité de lecture et d'écriture avec une fiabilité de 100 % sauf si les fichiers ont été créés avec la même version de leurs produits.

Les applications OSS sont désormais capables de lire ces formats avec une fiabilité suffisante pour que les problèmes rencontrés ne soient pas dissemblables de ceux apparaissant avec les différentes versions des propres produits de Microsoft. Plus le format est ancien, mieux les applications OSS savent le gérer. Celles-ci tendent à être meilleures en lecture qu'en écriture des formats Microsoft. En général, les applications OSS peuvent donc être utilisées avec confiance. Par exemple, Ximian avec son dernier bureau virtuel a configuré le format par défaut des fichiers de sa version de *OpenOffice.org* à ceux de Microsoft.

L'exception intervient lorsque le travail collaboratif est nécessaire et qu'au moins l'une des parties impose l'utilisation d'un format propriétaire. Le cycle lecture-modification-enregistrement des fichiers dans ces formats peut introduire des anomalies qui n'apparaîtront pas avec l'utilisation d'une seule application propriétaire. Néanmoins il faut conserver à l'esprit que cette sorte de dégradation intervient aussi lorsque différentes versions du logiciel propriétaire sont utilisées.

Pour les fichiers qui ne doivent qu'être lus et non modifiés, le format PDF doit être utilisé, ainsi qu'il est indiqué au chapitre 7.3 ci-dessus.

On peut penser que l'interface utilisateur doive être aussi similaire que possible à celle des logiciels Microsoft pour minimiser les coûts de re-formation.

Les modèles et macros *Visual Basic* (VB) sont courants dans de nombreuses administrations. Ils ont de même une format propriétaire fermé et devront être ré-écrits.

Trois suites bureautiques OSS coexistent : *OpenOffice.org*, *Koffice* et *Gnome Office* ; toutes trois doivent être évaluées.

Une étude pilote comparant la capacité des différentes suites bureautiques OSS à utiliser les fichiers *MS-Office* est disponible sur le web :

http://www.acmqueue.com/modules.php?name=Content&pa=showpage&pid=55

11.1.1 *OpenOffice.org* et *StarOffice*

OpenOffice.org est une suite bureautique OSS fondée sur *StarOffice*, produit d'une entreprise allemande appelée StarDivision. Sun Microsystems a acquis StarDivision et transmis le code à la communauté OSS. Elle continue à commercialiser une version d'*OpenOffice.org*, toujours appelée *StarOffice* à un prix très inférieur à ceux des paquetages propriétaires comparables.

StarOffice et *OpenOffice.org* sont globalement identique aux exceptions suivantes près :

- Sun Microsystems fournit un support commercial pour *StarOffice* ;
- *StarOffice* inclut une base de données intégrée (*Adabas*) ;
- *StarOffice* dispose de certains filtres supplémentaires de conversion depuis/vers d'autres paquetages bureautiques (cependant le filtre *WordPerfect* est maintenant disponible sur GNU/Linux pour des raisons de licences) ;
- *StarOffice* dispose de quelques polices propriétaires ;

- *StarOffice* n'est disponible que dans un ensemble restreint de langues (hollandais, français, italien, anglais, allemand, espagnol et suédois) ;
- *OpenOffice.org* est mis à jour plus fréquemment que *StarOffice*.

Les deux applications sont comparables à *MS-Office* mais ne comportent pas de client de courriel. Les deux manipulent les fichiers *MS-Office* jusqu'à *MS-Office XP* bien que la compatibilité décroisse à partir des versions supérieures à *MS-Office 97*. Les fichiers protégés par mot de passe ne sont pas gérés (à l'exception des protections de niveau feuille dans les feuilles de calcul) et ont quelques problèmes avec les objets graphiques intégrés par OLE.

Elles disposent de leur propre implantation de BASIC et ne peuvent interpréter les macros VB qui doivent êtres traduites manuellement. Bien que la traduction représente un coût de migration, l'absence du support des macros bloque la transmission des virus macro. Sun Microsystems établit des liens avec des entreprises pour la traduction des macros et modèles Microsoft vers une forme compatible avec *StarOffice*.

Elles offrent aussi une interface *Java* mais ne reconnaissent pour l'instant que le JDK Sun Microsystems ; Sun Microsystems a annoncé un projet de développement d'un traducteur *Visual Basic for Applications* (VBA) vers *Java*.

La version téléchargeable d'*OpenOffice.org* ne contient que quelques dictionnaires mais d'autres sont disponibles pour la plupart des langues européennes. Il existe d'ores et déjà des versions pré-compilées pour 25 langues différentes.

Bien qu'*OpenOffice.org* ne propose actuellement pas de paquetage de base de données, elle dispose des interfaces ODBC et JDBC vers de nombreux SGBD y compris les principaux SGBD OSS. Elle ne dispose pas non plus de filtres avec *WordPerfect* mais cela est prévu pour une version future.

Les deux s'exécutent sur une gamme de systèmes d'exploitation incluant *MS-Windows* et GNU/Linux.

Les journalistes commencent à reconnaître *OpenOffice.org* à la fois pour ses fonctionnalités et sa stabilité - voir :

http://www.raycomm.coom/techwhirl/magazine/technical/openofficewriter.html

Le site suivant fournit un bon manuel *OpenOffice.org* :

http://www.taming-openoffice-org.com/

11.1.2 Koffice

C'est le composant bureautique du bureau virtuel KDE. C'est un paquetage intégré qui fournit des outils de traitement de texte, tableur, grapheur, présentation, illustration, génération de rapports et organigrammes avec un bureau virtuel optionnel appelé Workspace.

Les filtres Microsoft ne sont pas aussi aboutis que ceux fournis avec *OpenOffice.org*. Elle ne dispose pas de langage de macros mais il est possible de réaliser des scripts.

Koffice fonctionne bien et son interface est esthétique et intuitive.

11.1.3 Gnome Office

Il s'agit d'une collection de programmes conformes aux standards Gnome et donc qui s'intègrent les uns aux autres, disposent d'une interface utilisateur similaire et devraient être capables de s'imbriquer les uns dans les autres.

OpenOffice.org est désormais considéré comme partie intégrante de *Gnome Office* même si elle ne se conforme pas aux standards Gnome. En particulier, Ximian travaille sur la convergence d'*OpenOffice.org* avec Gnome et en inclut sa propre version dans son dernier bureau virtuel, *XD2* (voir http://www.gnome.org/gnome-office/ pour plus de détails).

Gnome Office inclut de nombreux composants y compris *AbiWord* (traitement de texte), *Gnumeric* (tableur), *Sodipodi* et *Sketch* (outils de dessin vectoriel), *Gimp* (édition d'images), *Eye of Gnome* (afficheur d'images), *Dia* (graphiques vectoriels, similaire à *Visio*) et *Agnubis* (présentation).

Ces composants ont un niveau d'utilisabilité variable ; *AbiWord* par exemple est très bon pour le traitement de texte de base mais a des problèmes avec les tableaux et *Agnubis* est très limité, tandis que *Gnumeric* est un tableur très capable. Le développement, toujours très actif, progresse quasiment sur tous les composants.

Gnumeric en particulier a pour but de développer un tableur qui puisse effectuer tout ce que *MS-Excel* effectue et s'en rapproche beaucoup. La dernière version bêta (1.1.20 à la date de rédaction) en supporte toutes les fonction de tableur de la version US et une version de production de ce niveau est attendue pour septembre 2003. Les développeurs viennent du monde de la finance et ont inclus nombre de fonctions qui rendent *Gnumeric* particulièrement adapté à l'utilisation financière. C'est dans ce domaine qu'il semble que *Gnumeric* dépasse *MS-Excel*.

Gnumeric utilise le format ***.xls** en natif, tandis que *OpenOffice.org* convertit les feuilles de calcul dans un format XML.

La gamme de produits disponibles est intéressante et propose nombre de solutions différentes. Néanmoins si une suite intégrée est nécessaire, *OpenOffice.org* est la seule solution réelle.

11.2 Courriel

Le courriel est un domaine complexe avec différents composants logiques qui dispose d'une myriade d'applications OSS dont certaines apportent des fonctionnalités redondantes. Il est aussi finement lié à d'autres points tels que le contrôle des virus et du courriel non sollicité (« spam »).

Le choix d'une application appropriée est complexe et une étude détaillée des différents points se trouve en annexe C ainsi qu'une définition de tous les termes utilisés ici.

11.2.1 Agents de transport

Les principaux agents de transport (MTA) sont *Sendmail*, *Exim*, *Courier-MTA*, et *Postfix*. Il en existe de nombreux autres mais ceux-ci sont considérés comme les principaux dans le cadre d'une utilisation à grande échelle.

Traditionnellement, *Sendmail* était le MTA des sites Unix et OSS ; malheureusement, son niveau de sécurité est pauvre et il est aussi notoirement difficile à configurer.

Tous les autres ont de bonnes réputations et leur niveau technique similaire entre eux. Il existe cependant une différence significative dans le niveau de la documentation disponible, puisque *Postfix* est mieux documenté au niveau débutant tandis que c'est au niveau expert pour *Exim*.

Les MTA *Exim* et *Postfix* sont inclus dans différentes distributions OSS mais peuvent ne pas être installés par défaut.

Courier-MTA est un élément d'une famille incluant un MTA, un MDA (agent de distribution du courriel), un MAA et un paquetage de courriel web (*sqwebmail*). Chaque élément peut être utilisé séparément ou non.

Qmail est un MTA souvent considéré à tort comme OSS. Bien que le code source en soit disponible, sa licence ne permet que la distribution de versions strictement identiques à celles distribuées par l'auteur; il est donc un logiciel propriétaire. Sa licence restrictive le rend extrêmement difficile à intégrer avec

d'autres logiciels et c'est pourquoi il n'est pas intégré dans les distributions OSS. Il a cependant un bon niveau de sécurité et il est très utilisé.

Un autre produit intéressant est le serveur de courriel d'Apache *James*. Il est conçu comme un MTA complet écrit en *Java*. Il lui manque encore quelques fonctionnalités mais il mérite de n'être pas perdu de vue dans l'avenir.

Le choix de référence est *Exim* car il est aussi capable que *Sendmail* tout en étant plus facile à configurer et probablement plus sécurisé. Les deux autres sont peut-être moins aptes à absorber de grands volumes de messages. Le choix n'est pas absolu et les administrateurs feront leur propre choix en fonction de leurs contraintes locales.

11.2.2 Agents de stockage

De nombreuses administrations souhaitent un stockage centralisé des messages de préférence au téléchargement sur les postes locaux. Pour cette raison, nous préconisons fortement l'utilisation d'IMAP.

Il existe trois serveurs IMAP connus : *UW-IMAP* (parfois appelé simplement IMAP), *Courier-IMAP* et *Cyrus*.

En termes de sécurité, l'histoire de *UW-IMAP* n'est pas rose et nous ne le recommandons pas.

Des deux autres, *Courier-IMAP* est largement considéré comme plus simple à configurer. Il nécessite moins de ressources et fonctionne bien avec *Postfix* et *Courier-MTA* (c'est la partie MAA de la famille Courier). Il nécessite un format de stockage maildir et a pour réputation d'avoir des difficultés de traitement de certains messages S/MIME, déplaçant les en-têtes d'une manière qui en invalide la signature.

Cyrus utilise son propre format de stockage similaire à maildir et nécessite son propre MDA pour alimenter celui-ci.

Courier-IMAP et *Cyrus* supportent le standard d'authentification et de confidentialité ***TLS***.

Il existe de nombreux MDA tels que *procmail*, *maildrop* de Courier et *deliver* de Cyrus. Les MDA ont aussi la capacité de filtrer le courriel selon des règles sophistiquées, ce qui est utile si le MUA ne dispose pas de moyens de filtrage.

Le choix de référence est *Courier-IMAP* sans MDA ; ce dernier n'est pas nécessaire car *Exim* est capable d'écrire directement dans des structures maildir et qu'*Evolution* dispose lui-même de filtres très efficaces.

11.2.3 Agents utilisateur

De nombreux MUA existent dans le domaine OSS, aussi bien en mode texte que graphiques.

Si l'on souhaite conserver quelque chose de similaire à *Outlook* ou *Outlook Express*, le choix évident est *Evolution*. Il ne s'agit pas seulement d'un client de courriel mais aussi d'un gestionnaire d'informations personnel (PIM - Personal Information Manager). Il s'intègre à LDAP et peut ainsi accéder à l'annuaire de l'administration dès lors que ce dernier est conforme au schéma utilisé par *Evolution*. Ximian qui l'a développé très activement en a fait sa tête de gamme qui comporte aussi *Connector* (qui est un produit propriétaire payant), un produit qui permet à *Evolution* de se connecter à *MS-Exchange* (sauf pour la version 5.5).

Malheureusement, *Connector* ne supporte pas intégralement le mode IMAP déconnecté ; il copie seulement certains courriels sur le poste. Il offre cependant une fonctionnalité très utile appelée « dossiers virtuels » qui permet à l'utilisateur de définir des règles d'affichage des messages sous différents aspects sans dupliquer ceux-ci.

Les autres MUA sont *Kmail* et *Sylpheed*, tous deux excellents et s'intégrant dans les environnements OSS principaux. *Kmail* sera utilisé avec le bureau virtuel KDE et *sylpheed* avec Gnome.

Evolution supporte *GPG* mais non S/MIME bien que cela soit en principe implanté.

Mozilla supporte S/MIME mais ni *GPG* ni *PGP* bien que cela soit en principe implanté.

Kmail supporte S/MIME, *GPG* et *PGP* avec le projet égyptien fondé par le gouvernement allemand.

De nombreux paquetages de travail de groupe incluent des clients compatibles IMAP ou POP3. En général, ceux-ci ne sont pas aussi bons qu'*Evolution* mais peuvent suffire pour une intégration avec les autres fonctions.

Dans certains cas, il peut être meilleur de faire migrer certaines catégories d'utilisateurs vers une interface web. *SquirrelMail* en est une particulièrement valable (on peut la trouver à http://www.squirrelmail.org/).

L'Open Systems Applications Foundation édite un produit appelé *Chandler* qui est encore embryonnaire mais peut devenir un compétiteur d'*Evolution*.

La majorité des utilisateurs pouvant préférer l'aspect de *MS-Outlook*, *Evolution* est sans doute le mieux placé et donc le choix de référence. Cependant, si S/MIME est nécessaire immédiatement, *Kmail* doit être utilisé, ce qui implique de choisir *KDE* au lieu de *Gnome*.

11.2.4 Anti-virus

Si les systèmes OSS sont correctement configurés, les virus ont des effets limités. Cependant le problème de transmission à d'autres sites (en particulier ceux utilisant les produits Microsoft) perdure.

Il existe un anti-virus OSS, *ClamAV*, mais celui-ci présente quelques problèmes. Dès lors, ce sont des produits propriétaires qui sont recommandés actuellement.

Le meilleur emplacement pour ces produits est dans le MTA. *Postfix* et *Exim* permettent l'incorporation de filtres de ce genre.

Plusieurs produits propriétaires ont bonne réputation (*Sophos*, *RAV* et *Vexira*), *Trend* est efficace pour le contrôle de virus mais nécessite un accès super-utilisateur ce qui en fait un problème de sécurité potentiel plus grave s'il est compromis.

Aucun choix n'est fait pour le modèle de référence car **netproject** n'a pu tester intégralement les produits.

11.2.5 Autres outils

Il existe de nombreux outils anti-spam ainsi que d'autres qui interdisent le téléchargement d'exécutables avec le courriel.

SpamAssassin est probablement le plus connu.

Anomy Sanitizer est un outil configurable capable de supprimer les pièces jointes des messages, mais doit être utilisé avec précaution car une telle action peut invalider la signature d'un document.

MailScanner est un cadre général d'analyse de contenu, incluant des mesures anti-virus et anti-spam. Il peut appeler différents produits anti-virus propriétaires et peut utiliser *SpamAssassin* aussi bien qu'appliquer ses propres règles.

Fetchmail collecte le courriel depuis des emplacements distants pour le stocker ou le transmettre à un MTA. Puisque c'est un outil tirant (le transfert est initié par la machine de réception), il est utile lorsque, pour des raisons de sécurité, les administrateurs ne souhaitent pas ouvrir de port d'entrée sur leur passerelle Internet, ce qui est le modèle SMTP normal.

OfflineImap est un outil de synchronisation de stockages de courriel entre un serveur et un client ; il fonctionne par connexion régulière du client par IMAP. La structure locale est au format maildir. Il peut être très utile pour assurer le mode IMAP déconnecté si le MUA ne le permet pas totalement.

Whoson permet l'authentification d'utilisateurs distants par la méthode POP-avant-SMTP. Son utilisation est nécessaire si les utilisateurs peuvent émettre à distance du courriel via les serveurs de l'administration, que le SMTP authentifié n'est pas disponible et que le MTA de l'administration est ouvert à des connexions IP hors de son contrôle.

11.2.6 Problèmes rencontrés

Stocker des informations dans un serveur LDAP nécessite le choix d'un schéma. Celui-ci doit être compatible avec tous les clients qui peuvent nécessiter d'accéder aux informations. Heureusement, certains paquetages contiennent un schéma susceptible de satisfaire aussi bien les besoins de plusieurs autres.

Courier-IMAP dispose d'un schéma (contrairement à *Exim*) qui, à l'expérience, convient aussi à *Exim*. Nous ne savons pas si toutes les fonctionnalités d'*Exim* sont supportées.

Différents problèmes furent découverts dans le fichier de configuration LDAP de *Courier*. Les correctifs ont été renvoyés au développeur mais n'ont pas encore été inclus dans une version.

L'utilisation de *Courier* avec *Whoson* nécessite quelques modifications de *Courier* dont certaines, disponibles sur le site de *Whoson*, étaient assez anciennes et nécessitaient des modifications significatives avec notre version de *Courier*.

11.3 Agenda et travail de groupe

L'agenda est un sujet mal défini dans l'OSS en raison de l'absence de standards de communication entre les clients et le serveur central. Ainsi les produits actuellement développés s'appuient sur une interface web, ce qui peut dérouter les gens habitués à *Exchange* et *Outlook*. Ce domaine est une faiblesse significative dans le portefeuille OSS.

Sauf mention contraire, les produits listés dans la table ont une interface web. Tous font partie de suites de travail de groupe dont les fonctionnalités sont très variées.

De nombreux produits sont écrits en PHP ou Perl et peuvent donc être personnalisés. Une intéressante intégration de fonctionnalités a été développée dans ceux-ci.

phpGroupWare (http://www.phpgroupware.org/) a bonne réputation.

Tableau 1 :Détail des produits de travail de groupe

Produit	***Courriel***	***Agenda***	***Gestion documentaire***	***Discussion (chat)***	***Liste de tâches***	***Gestion de contacts***	***Base de données***	***Feuille de temps***	***Planification***	***Autres fonctions***	***Remarques***
NullLogic	O	O	O	[1]	O		O				
Twiki		O	O	O			O				Plus un cadre qu'un produit
phpGroupWare	O	O	O		O	O	O		O		Difficile de trouver des informations précises sur le site web
phProject	O	A, B	O		[2]	O		O		Gestion de projets, signets, rappels, système de recherche, système de vote, traceur de requêtes	
Tutos	O	B	O			O		O			
Twiggi	O	O	O		[2]	O			O	Choses à faire, signets	

Notes :

[1] forum au standard IM plus (BBS)

[2] petite fonctionnalité « Post-It »

A projets dynamiques

B ressources (salles, projecteurs...) et enregistrement d'événements (passés et futurs)

The Horde est un cadre d'exécution d'autres applications (par exemple *Imp*, un serveur de courriel web, *Turba*, un gestionnaire de contacts et *Kronolith*, un agenda). Voir http://www.horde.org/.

NullLogic semble n'être disponible qu'en anglais contrairement à *phProject*, *Tutos*, *Twiggi* et *Twiki* qui parlent plusieurs langues.

OpenGroupware est un produit OSS très récent de http://www.opengroupware.org/. Il s'agit de l'application *SKYRiX*, précédemment propriétaire, qui a été rendue OSS. Il est destiné à devenir un remplaçant pour *MS-Exchange*. Nous avons manqué de temps pour l'évaluer en entier mais les premiers éléments font penser qu'il deviendra très influent.

Un autre produit récent est *Kgroupware* (voir http://kolab.kroupware.org/). Ce produit dont le client est fondé sur *Kmail* est à étudier, en particulier si *KDE* est choisi comme interface utilisateur ou *Kmail* comme MUE (pour son support de S/MIME par exemple).

11.3.1 Agendas personnels

Sauf mention contraire, tous les produits savent gérer des agendas personnels et des listes de tâches.

11.3.2 Agendas de groupe

Tutos, *Twiggi*, et *NullLogic* le permettent tous. *Tutos* permet le contrôle par niveaux depuis l'individu jusqu'à tout le monde en passant par le groupe de travail et le groupe de projet.

Dans *NullLogic*, les agendas ne peuvent être cachés aux autres membres du groupe, contrairement aux tâches.

11.3.3 Organisation de réunions

Beaucoup des produits permettent la planification de ressources que l'on peut utiliser pour planifier

des réunions.

Tutos permet l'attribution automatique des gens ainsi que la notification automatique de ceux qui ne sont pas dans l'agenda partagé (comme ceux d'autres organisations). Il conserve la trace des acceptations et envoie des rappels par courriels si nécessaire. *phProject* est similaire et permet aussi les notifications par texto (SMS).

NullLogic permet toutes les fonctions susdites, sauf la réservation de salle.

11.3.4 Synchronisation d'organiseur

phProject permet la synchronisation avec les organiseurs sous PalmOS par un complément. La synchronisation des organiseurs est aussi supportée dans Gnome et *Evolution*. La plupart des organiseurs répandus peut être synchronisée.

11.4 Services web

11.4.1 Navigateur

Les principaux navigateurs OSS sont *Mozilla*, *Galeon* et *Konqueror*. D'autres existent, tels que *Lynx* en mode texte, souvent utilisé comme base de navigateur pour les handicapés et *Firebird* de Mozilla (précédemment connu sous le nom de *Phoenix*), une version allégée de *Mozilla*. Un navigateur propriétaire, *Opera* dispose d'une version GNU/Linux. *Netscape* est fondé sur *Mozilla* et s'exécute sur des plates-formes OSS, mais inclut du code propriétaire.

Mozilla est le projet OSS majeur (fondé sur le code publié par Netscape) qui constitue la base de *Netscape 7*. Il inclut les composants courriel et forums ainsi qu'un carnet d'adresses et un outil de composition web. Beaucoup du code de *Mozilla* est utilisé par d'autres projets, y compris *Galeon* et *OpenOffice.org*.

Galeon est uniquement un navigateur conçu pour être petit et rapide. Il est fondé sur *Gecko* (le moteur de rendu sur lequel est fondé le projet *Mozilla)* et sur une I.H.M. Gnome.

Galeon et *Mozilla* supportent tous les standards ouverts Internet du web et peuvent exécuter les codes Java et Javascript correctement écrits.

Certains contenus nécessitent un composant disponible uniquement pour *MS-Windows*, tel que *Shockwave Director*. Le produit propriétaire de CodeWeavers *CrossOver Plugin* permet aux composants conçus pour *MS-Windows* de s'exécuter sous GNU/Linux.

Konqueror est le navigateur du bureau virtuel KDE ainsi que son gestionnaire de fichier glisser-déposer. Il est fondé sur le moteur de rendu KHTML (avec *Gecko*, celui de *Mozilla*, en option) avec une I.H.M. KDE.

11.4.2 Serveurs web

Le serveur web OSS le plus répandu est *Apache* qui couvre, selon l'estimation de Netcraft (http://www.netcraft.com), plus de 60 % du marché et sa part est en progression. Une combinaison de produits de plus en plus répandue est connue sous le nom de LAMP : *Linux*, *Apache*, *MySQL* et PHP. Elle fournit un cadre d'exploitation de sites utilisant des bases SQL à l'aide du langage PHP. Tous ses composants sont OSS.

Le projet Apache contient nombre de sous-projets, dont l'un, *Jakarta*, couvre l'utilisation côté serveur de Java. *Jakarta* lui-même consiste en sous-projets, dont deux sont *Tomcat* et *Slide*. *Tomcat* fournit un produit de servlets Java conforme au standard JSP et la capacité d'utiliser des technologies telles que le *Websphere* d'IBM. *Slide* est une implantation Java de WebDAV qui permet la gestion de contenu. Se reporter à http://www.apache.org/ pour tous les détails.

Les autres serveurs OSS à considérer sont *Zope* et *Tux*.

Zope (http://www.zope.org/) est conçu pour fournir le support du contenu web dynamique et fondé sur un modèle orienté objet. C'est un paquetage intéressant qui combine un système de gestion de contenu, un serveur web et un système de modèles. *Zope* supporte aussi les composants modulaires (appelés produits) et il est réalisé en langage orienté objets Python. Il est courant de trouver *Zope* implanté « derrière » *Apache* dans une configuration multi-serveurs dans laquelle *Apache* sert le contenu statique et agit comme un accélérateur cache pour les parties du site gérées par *Zope*.

Tux est un développement RedHat désormais dénommé *The Red Hat Content Accelerator*. Il utilise un noyau spécifique et devrait fournir des réponses très rapides pour les pages statiques.

Roxen est une autre combinaison serveur web-gestion de contenu, mais dont la pleine capacité fonctionnelle n'est atteinte que par des composants propriétaires.

Parmi tous, *Apache* est de loin le plus répandu. Il anime actuellement 63 % des sites web publics du globe et sa part de marché progresse régulièrement contre celle de *IIS*, il existe donc une très grande expérience à utiliser lors de la planification d'une migration. *Apache* est un serveur modulaire avec une large sélection de modules pour des usages spécifiques autour d'un moteur de protocole de base.

11.4.3 Portail / Contenu

Zope, comme d'autres composants OSS, est un élément du projet à fonds européens *ASWAD* (http://www.aswad-project.org/) conçu la gestion de contenu. netproject espérait être en mesure de le tester mais il fut difficile d'obtenir des détails à jour. plone est un projet intéressant fondé sur *Zope* (http://www.plone.org/).

JBoss (http://www.jboss.org/) est un serveur d'application en Java de bonne réputation et activement développé.

De nombreux produits de gestion de contenu OSS sont maintenant disponible, comme le montrera une visite à http://www.oscom.org/matrix/index.html. netproject n'a pu trouver aucune information détaillée dans les études de cas publiées et fut dans l'incapacité d'investiguer à un niveau de détail nécessaire pour choisir un candidat pour le modèle de référence.

11.5 Gestion documentaire

11.5.1 Enregistrement et extraction

La gestion documentaire peut et même doit être vue comme une forme de gestion de contenu et de flux. C'est le genre de fonctionnalités qu'*ASWAD* (voir 11.4.3 ci-dessus) est prévu pour couvrir. netproject recommande donc d'adopter une solution fondée sur le produit de gestion de contenu choisi. En particulier, celles qui s'appuient sur WebDAV devraient fournir les solutions les plus utilisables.

Un standard allemand appelé DOMEA (Disposition and Archiving of Electronic Records), peu utilisé en-dehors de l'Allemagne, a cependant été adopté par IBM en conjonction avec SAP. La plupart des documents relatifs à DOMEA (en particulier, ceux trouvés par une recherche Google) sont écrits en allemand. Une entreprise appelée FabSoft fournit un support pour DOMEA sous GNU/Linux sur les grands systèmes IBM. Il ne semble y avoir aucun produit OSS supportant DOMEA.

Certains des produits de travail de groupe fournissent un support pour la gestion documentaire :

- *Tutos* inclut un système de gestion documentaire assurant aussi la gestion des versions ;
- *NullLogic* inclut une capacité de stockage, d'indexation et de téléchargement de fichiers simple ; il ne semble pas offrir de système de gestion des modifications ; il dispose d'un mécanisme de requêtes généralisé qui peut être utilisé pour offrir une indexation sensible.

11.5.2 Travail collaboratif

Cette fonction peut être implantée correctement par simple échange de documents entre les utilisateurs. Cet échange peut être réalisé par pièce jointe ou par des mécanismes tels que ceux utilisés par CIRCA.

La collaboration nécessite que les parties se mettent d'accord sur le format du document et actuellement, utilisent en général le format de Microsoft ***.doc** par défaut. Ce choix nécessite une confiance mutuelle entre les parties en raison de l'efficacité de ces formats en tant que vecteurs de virus. De plus, un standard ***.doc** n'est pas idéal car il est en constante évolution : le format utilisé par *MS-Office 2000* n'est pas identique à celui de *MS-Office 97*. Cela implique que les parties fixent aussi la version logicielle à utiliser.

La pression augmente en faveur de l'adoption de formats de documents à base standard, en particulier ceux fondés sur XML. *OpenOffice.org* propose un standard documentaire XML ouvert qui peut être utilisé comme base de collaboration.

Une approche plus structurée serait d'adopter une solution de gestion de contenu/flux comme décrit plus haut.

Le produit de travail de groupe *Tutos* permet aux documents d'être contrôlés par une seule personne ou par toutes celles d'un groupe défini. *NullLogic* et *Twiki* aussi dispose de contrôles fins.

Comme pour la gestion de contenu ci-dessus, aucun candidat n'est retenu pour le modèle de référence.

11.6 Bases de données

11.6.1 Bases centrales pour applications

Les SGBD OSS disponibles sont *MySQL*, *PostgreSQL* et *Firebird*. Leurs caractéristiques et domaine d'utilisation sont significativement différents.

MySQL est un SGBDR SQL léger à préconiser pour les serveurs web et applications similaires. Il est approprié dans les situations dans lesquelles la lecture est prédominante sur l'écriture. Il ne permet pas les procédures (support prévu pour la version 5) ni les sous-requêtes complexes.

PostgreSQL est un SGBDR complet comparable à *Oracle* et à *DB2*, mais sans les fonctionnalités avancées nécessaires à la manipulation de volumes de données très importants.

Firebird est une version de *Interbase* de Borland publiée sous licence OSS. Beaucoup du code est commun avec *Interbase* et doit donc être considéré comme mature. Un projet, encore embryonnaire, prévoit d'ajouter des capacités de bases de données à *OpenOffice.org* en utilisant *Firebird*.

11.6.2 Bases de données personnelles centrales ou locales

Les bases de données personnelles ne sont pas très pratiques à mettre en oeuvre avec l'OSS. Il n'existe pas d'équivalent direct de *MS-Access* ni en développement.

Certains paquetages de travail de groupe offrent une certaine possibilité dans ce domaine à partir des différents SGBD SQL OSS. Dans certains cas (comme pour *NullLogic*), les utilisateurs ne peuvent qu'exécuter des requêtes prédéfinies. Certains permettent la définition de formulaires permettant le stockage et l'accès aux données.

11.6.3 Connectivité

De nombreux produits SGBD supportent des API directes en langage C. Certains supportent aussi C++ nativement.

Tous offrent la connectivité ODBC et JDBC ; certains offrent aussi la connectivité .NET.

Un produit appelé *Unix-ODBC* fournit une connectivité façon ODBC aux programmes Unix et GNU/Linux.

11.6.4 Performance

La performance d'un SGBD est hautement dépendant de la taille des tables mises en jeu et de la complexité des requêtes.

Aucune des offres OSS n'est en au niveau des exigences que peuvent traiter les SGBD propriétaires tels qu'*Oracle*. C'est particulièrement le cas d'applications telles que les entrepôts de données, en partie parce qu'aucune n'offre jusqu'à présent de possibilités de bases distribuées.

Les produits propriétaires *Oracle*, *DB2*, *Informix*, *Progress*, *Mimer* et *Sybase* sont disponibles sous GNU/Linux et doivent être considérés comme les options préférées pour des applications à lourde charge de bases de données pour lesquels les produits OSS ne sont pas encore adaptés. Les outils de développement *Oracle* sont supportés sous GNU/Linux.

12 Applications - groupe secondaire

12.1 Système d'exploitation

Il existe différents systèmes d'exploitation OSS, dont *OpenBSD*, *FreeBSD*, *NetBSD* et différentes « distributions » (explication plus bas) de GNU/Linux bien que, parmi elles, de nombreuses personnes n'aient entendu parler que de GNU/Linux et ce, généralement sous le nom Linux.

Un système d'exploitation consiste en un noyau qui s'exécute en mode superviseur, tout en permettant l'exécution de programmes en mode utilisateur sous son contrôle. *Linux* est un noyau qui nécessite des chargeurs, compilateurs, pilotes, etc. Beaucoup de ces programmes sont issus du projet GNU de la Free Software Foundation et donc la totalité devrait être appelée GNU/Linux, le terme utilisé dans ce rapport.

Le noyau *Linux* est fourni groupé en distributions avec un ensemble de programmes et applications-support par différentes entreprises telles que RedHat, SuSE et Mandrake. Les contenus d'une distribution doivent interopérer et le noyau peut être spécifiquement adapté par des modifications non disponibles avec d'autres distributions. Le choix d'une distribution doit donc être étudié, chacune ayant ses points forts et faiblesses.

Il existe d'autres distributions telles que *Debian* et *Gentoo*, préparées par des organisations non commerciales, ce qui a des implications sur le support fourni. Celui-ci est disponible, soit auprès de fournisseurs tiers, soit par accès aux listes de diffusion sur Internet. Les deux peuvent fournir un niveau de couverture acceptable.

Debian a une réputation de solidité et sa section « stable » contient du code profondément testé par de nombreux utilisateurs dans le monde. Deux autres sections proposent des logiciels dans des versions plus récentes. La branche stable a aussi la réputation d'être obsolète. C'est une réputation assez injuste car de nombreux utilisateurs commerciaux sont essentiellement intéressés par la stabilité et l'absence de bogues et non par le support du dernier périphérique.

Gentoo est une distribution exclusivement en code source, ce qui permet à l'administration de bâtir facilement ses propres exécutables, en ajustant la distribution à son environnement et à son matériel. La génération d'une telle distribution à partir de rien est consommateur de temps, mais une fois les exécutables engendrés, ceux-ci sont disponibles sans limite. C'est une nouvelle distribution qui vaut l'évaluation. Puisque beaucoup d'autres distributions sont fournies avec le code source, il est possible d'ajuster celles-ci de la même manière ; G*entoo* cependant peut être plus adaptée à un tel traitement.

Les distributions commerciales sont proposées selon différents paquetages avec différents niveaux de support. La distribution disponible par Internet est invariablement supportée seulement durant un an ; au-delà, les utilisateurs sont censés avoir effectué une mise à jour. De nombreuses entreprises fournissent une version « Enterprise » dont le support est garanti 5 ans ou plus, fondée sur des versions stables. Un contrat de support est associé à celles-ci, parfois appelé licence, même si le code est couvert par les licences GPL ou LGPL et qu'aucune autre licence ne peut leur être appliquée. C'est la disponibilité de distributions ainsi stables et avec support que souhaiteront de nombreux administrateurs. Effectivement, une des raisons du mouvement vers l'OSS est l'absence de pression pour de constantes et inutiles mises à jour. Les entreprises s'engagent à intégrer les corrections de bogues. RedHat par exemple a une gamme Enterprise consistant en trois produits, deux pour les serveurs et un pour les postes de travail techniques, chacun fondé sur la version 7.3 de *RedHat Linux*, même si la version téléchargeable est actuellement numérotée 9.

Notre opinion est que GNU/Linux est la plate-forme à préférer pour les postes de travail car il offre de meilleurs outils de configuration et une variété de paquetages plus adaptés à l'usage de poste. De plus, certains produits de poste de travail ne fonctionnent pas sur d'autres noyaux (le navigateur web *Mozilla* par exemple ne fonctionne actuellement pas sur *OpenBSD*).

Pour les serveurs, la situation est nettement moins claire. *OpenBSD* est de loin le plus sécurisé de tous les systèmes d'exploitation OSS, capable d'annoncer la découverte d'une seule vulnérabilité exploitable à distance en 6 ans. Il doit être préféré pour tout ce qui nécessite un niveau de sécurité élevé (tels que les

pare-feux et les serveurs de DMZ-zone démilitarisée).

Les applications serveur fonctionnent généralement bien sur toutes les plates-formes BSD et GNU/Linux, bien que beaucoup aient été écrites pour GNU/Linux puis portées. Les logiciels propriétaires ne sont souvent disponibles que pour GNU/Linux.

12.2 Interface utilisateur

12.2.1 Gestionnaire de bureau - apparence

Il y a plusieurs choix, du gestionnaire de fenêtres le plus simple comme *icewm* jusqu'aux gestionnaires de sessions complets tels que ceux inclus dans *Gnome* et *KDE*. Le choix dépend de l'utilisation prévue.

Parmi les gestionnaires de session, *KDE* est le plus mature, mais *Gnome* se rapproche rapidement. *Gnome* est supporté par Sun Microsystems et les membres de la Gnome Foundation. netproject considère que son architecture est meilleure et pense que son avenir est plus prometteur.

XD2, un bureau virtuel *Gnome* a récemment été publié par Ximian. Il fonctionne sur nombre de différentes distributions de base y compris *RedHat* et *SuSE*. Ximian a fait particulièrement attention à l'intégration des différentes applications pour qu'elles fonctionnent de manière similaire. Cela a impliqué l'inclusion de versions spécifiques de certains produits tels qu'*OpenOffice.org*. Il est trop tôt pour émettre un commentaire complet sur ce gestionnaire mais les impressions initiales sont prometteuses.

Le choix pour toute administration sera sans doute l'expression d'une préférence personnelle ; les deux environnements sont très capables. Les applications conçues pour fonctionner dans un environnement fonctionneront dans l'autre mais des capacités plus spécifiques telles que la gestion de sessions peuvent ne pas fonctionner correctement.

12.2.2 Langues

Les gestionnaires de bureaux virtuels offrent de nombreuses langues européennes mais leur support linguistique peut être laborieux.

12.3 Sécurité

Tous les groupes fonctionnels doivent être configurés avec le souci de la sécurité. Celle-ci ne peut fonctionner, au niveau logiciel, que si elle s'intègre à un cadre plus large de gestion de sécurité. netproject n'a pas étudié en détail toutes les fonctions détaillées dans ce guide.

12.3.1 Chiffrement de données

12.3.1.1 Données en transit

Les données confidentielles d'un intranet doivent être chiffrées le plus tôt possibles. Les informations sensibles transmises par Internet doivent toujours être chiffrées. Cela peut être réalisé par des tunnels chiffrants avec des produits comme *ssh* et *stunnel*.

12.3.1.2 Données stockées

Les données confidentielles portées par des équipements mobiles doivent être chiffrées sur le disque. L'idéal est que toutes les données soient chiffrées mais cela impose une surcharge importante qui n'est pas toujours acceptable. Il existe plusieurs systèmes de fichiers sécurisés et le prochain noyau Linux fournit un meilleur support de ceux-ci. Par exemple, http://koeln.ccc.de/archiv/drt/crypto/linux-disk.html présente une discussion des différentes méthodes disponibles.

12.3.2 Authentification

Les méthodes pour identifier sans ambiguïté une personne ou une machine sont une partie de la communication avec d'autres personnes ou machines. Cela inclut des infrastructures de signature et de gestion de clefs (I.G.C. ou PKI-Public key Infrastructures). Aucune I.G.C. n'a été testée pour ce guide. Toute l'identification a été réalisée depuis une base LDAP avec le couple utilisateur/mot de passe.

12.3.3 Autorisation

Une fois identifiée, l'autorisation détermine ce qu'une personne ou une machine peut réaliser et dans quelles circonstances. C'est normalement une partie du système d'exploitation ou du code applicatif. Le contrôle d'accès par rôles (RBAC - Role Based Access Control) a été défini par le NIST aux États-Unis et il est disponible pour Linux (voir http://csrc.nist.gov/rbac/).

12.3.4 Contrôle anti-virus

Le contrôle anti-virus est essentiellement nécessaire pour empêcher la transmission de virus vers d'autres sites non-OSS. Bien que le courriel soit un des principaux vecteurs de transmission de ceux-ci, ce n'est pas le seul, ainsi un contrôle généralisé des fichiers est nécessaire pour empêcher la transmission par d'autres biais.

Malheureusement, nous ne connaissons aucun produit OSS qui assure un tel contrôle. Cependant, pour configurer correctement les systèmes de fichiers sur les serveurs et les postes de travail, il est possible de garantir que les fichiers exécutables soient seulement ceux installés par les administrateurs système. Il est donc important que ceux-ci s'assurent de l'innocuité des fichiers qu'ils installent, par exemple en contrôlant la signature placée sur les fichiers par les éditeurs de distributions.

12.3.5 Serveur mandataire (proxy)

Une gamme de serveurs mandataires OSS intelligents ou semi-intelligents est disponible.

Parmi les mandataires web, *squid* est le plus répandu. Un produit associé (*squidguard*) empêche l'accès à une liste de sites interdits.

12.3.6 Pare-feux

Tous les systèmes d'exploitation OSS actuels disposent de pare-feux par filtrage de paquets internes, dont la majorité sont avec états. Les pare-feux à états sont ceux qui maintiennent l'information sur les connexions en cours et les flux qui transitent à travers eux et permettent le passage aux paquets associés aux connexions tout en filtrant ceux qui ne le sont pas. Les pare-feux sans états examinent chaque paquet individuellement, sans conserver d'enregistrement des paquets précédents. Des composants spécialisés sont disponibles pour des protocoles tels que ftp et la téléphonie H.323 qui utilisent des formes de connexions non standard.

iptables, actuellement inclus dans GNU/Linux et *ipfilter*, inclus dans *FreeBSD*, sont deux bons produits pare-feux. *Packetfilter*, maintenant inclus dans *OpenBSD*, a aussi bonne réputation. Les bonnes pratiques pour les pare-feux avec l'extérieur consistent à en implanter deux différents entre la connexion au réseau public et les serveurs internes. Nous ne recommandons aucun exemple.

12.3.7 Réseaux privés virtuels (VPN)

12.3.7.1 OpenVPN

Disponible pour de nombreuses versions d'Unix, c'est une offre mature et puissante. Parmi les fonctionnalités on trouve le chiffrement à clef publique, la compression dynamique pour la gestion de bande passante et la capacité d'utiliser la translation d'adresses NAT (voir aussi http://openvpn.sourceforge.net/ pour plus d'informations.

12.3.7.2 FreeSWAN

Il s'agit d'une implantation GNU/Linux des standards IPSEC et IKE, ce qui le rend interopérable avec les équipements compatibles, y compris les routeurs spéciaux et d'autres systèmes d'exploitation. Puisque IPV6 supporte nativement IPSEC, *FreeSWAN* peut être préférable si ce nouveau standard est utilisé. Pour bénéficier de l'extension unique de *FreeSWAN* « chiffrement opportuniste » qui peut automatiser la sécurité, les enregistrements DNS doivent être mis à jour, ce qui peut constituer une limitation. netproject comprend qu'IPSEC peut aussi poser problème avec NAT. Se reporter à http://www.freeswan.org/ pour plus d'informations.

12.3.7.3 CIPE

Celui-ci est moins mature que les deux autres et le support des clefs publiques y est encore expérimental. Il peut cependant fonctionner avec NAT, est disponible pour *MS-Windows* et fait partie de la distribution *RedHat Linux* (il peut même être configuré à l'aide de l'outil de contrôle de périphériques réseau). Plus d'information est disponible à http://sites.inka.de/~W1011/devel/cipe.html.

12.4 Gestion

Le site http://www.infrastructures.org/ contient de nombreuses informations sur la gestion d'un réseau de machines (serveurs et postes) et propose une quantité d'outils OSS pour toute une gamme de tâches de maintenance. Il est maintenu par quelqu'un qui a une expérience de celles-ci de nombreuses années. netproject est en accord avec la quasi-totalité du contenu à l'exception de la section sur la gestion des imprimantes.

Le site montre que la gestion des systèmes Unix (et par extension, GNU/Linux) tend à être réalisée par des outils agrégés depuis des unités mono-fonctionnelles. Cette approche modulaire est extrêmement puissante et c'est ce qui permet aux administrateurs Unix et GNU/Linux d'être très efficaces. Elle indique aussi que le marché des boîtes à outils est petit, car chaque administrateur a tendance à se construire son propre ensemble.

Des outils propriétaires tels que *Tivoli* d'IBM et *CA-Unicenter* de Computer Associates sont aussi disponibles.

12.4.1 Gestion des utilisateurs

C'est la gestion des utilisateurs et des groupes, y compris la gestion des mots de passe. Des produits tels que *Directory Administrator* et *gq* permettent la maintenance de bases LDAP.

12.4.2 Gestion de configuration

Bien qu'un client bien conçu, géré en central, ne doive nécessiter qu'une maintenance locale minimale, la mise à jour de sa configuration sans réinstallation complète reste souhaitable pour les réseaux vastes en fonctionnement depuis un certain temps. Par exemple, si un service central de base est modifié, les clients peuvent nécessiter une reconfiguration pour utiliser celui-ci.

12.4.2.1 Maintenance manuelle

Les administrateurs peuvent réaliser manuellement les mises à jour de configurations comme ils assurent celles des logiciels. Cependant les mêmes problèmes de synchronisation s'appliquent. La modification manuelle de fichiers de configuration, souvent stockés au format texte simple, est particulièrement sujette aux erreurs.

12.4.2.2 Cfengine

Le moteur de configuration GNU (GNU Configuration Engine, http://www.cfengine.org/) automatise la configuration à distance des clients réseau. Il supporte une grande variété de systèmes Unix et

son puissant concept de classes permet la gestion de différents groupes de clients avec un travail minimal. Des agents autonomes sur les clients peuvent mettre à jour des fichiers textes, des interfaces réseau, des liens de fichiers et des permissions, le stockage temporaire et les systèmes de fichiers montés.

Certaines des primitives qui peuvent être automatisées à l'aide de *cfengine* sont :

- contrôle et configuration de l'interface réseau ;
- édition de fichiers texte ;
- création et mise à jour de liens symboliques, voire de plusieurs liens avec une seule commande ;
- contrôle et positionnement de permissions et propriété de fichiers ;
- suppression de fichiers inutiles qui encombrent le système ;
- montage automatisé systématique de systèmes de fichiers (sous Unix) ;
- contrôle de présence de fichiers et systèmes de fichiers importants ;
- exécution contrôlée de scripts et commandes shell.

Cfengine suit une structure décisionnelle de classes.

12.4.2.3 System Configurator

System Configurator (http://sisuite.org/systemconfig/) est un élément la la suite d'installation de systèmes (*System Installation Suite*) utilisé par *System Installer*. Il peut configurer et maintenir de nombreux composants d'une installation GNU/Linux (tels que réseau, stockage, heure locale et démarrage) à travers de nombreuses distributions.

12.4.3 Gestion logicielle

Cette section couvre la maintenance système des clients depuis la configuration initiale sur un matériel nouveau jusqu'aux mises à jour des logiciels et de la configuration et différentes technologies disponibles permettent d'en faciliter la gestion.

12.4.3.1 Installation système

L'installation système est la configuration initiale des logiciels ainsi que celle nécessaire pour le fonctionnement d'une machine. Les machines sortant d'usine peuvent n'avoir aucun système d'exploitation ou arriver pré-installées. Les machines plus anciennes porteuses de logiciels inutiles peuvent aussi être réutilisées en installant un nouveau système.

La première tâche d'installation est le démarrage de la machine. Pour supporter des machines qui ne démarrent pas telles que celles sortant d'usine avec un disque dur non initialisé, le BIOS doit permettre au moins une méthode de démarrage différente que celle du disque dur. La méthode la plus ancienne est celle de la disquette et (bien que celui-ci soit largement disponible) elle suppose qu'un lecteur de disquette soit présent. Cela est le passé. Les disquettes sont lentes, non fiables et offrent une espace très limité pour le logiciel d'installation système des standards modernes. De nombreuses machines (construites depuis 1997) supportent le démarrage depuis le CD-ROM par émulation du secteur de démarrage d'une disquette. Si un lecteur de CD est présent, c'est une méthode plus rapide et qui offre plus d'espace pour le logiciel de démarrage ainsi que pour les autres logiciels nécessaires ensuite. La méthode de démarrage la plus sophistiquée s'effectue par le réseau. Tous les BIOS et toutes les cartes réseau ne supportent pas cette nouvelle fonctionnalité. L'environnement de pré-exécution (PXE) est défini dans le standard industriel Wired for Management (WfM) et permet à de nombreuses machines (acquises depuis 1998) de démarrer depuis le réseau local.

Le programme d'installation doit accéder au support d'installation approprié contenant le logiciel de

niveau supérieur à exécuter après le démarrage. Typiquement, celui-ci sera stocké sur un CD-ROM local ou un serveur de fichiers réseau. Un seul CD peut stocker une image logicielle et la capacité d'un CD-ROM doit suffire pour l'administration d'un poste de base (avec une compression de fichiers standard). Cette image statique peut convenir si le logiciel n'est pas sujet à modifications ou si une installation de base est seulement nécessaire pour l'ajout d'autres logiciels. En général, une installation réseau est plus souple, peut être plus rapide, offre une capacité supérieure et échelonne mieux pour des installations multiples et parallèles en partageant les disques d'installation entre les clients.

L'installeur système transfère le logiciel du support sélectionné vers le disque dur de la machine cible et le prépare au démarrage. Cela implique une détection du matériel, le contrôle de la capacité du disque et la configuration des détails du réseau.

Quelques-unes des méthodes possibles d'installation sont étudiées ci-dessous.

1 Installation manuelle

L'installation la plus basique est effectuée par un administrateur système. Les logiciels sont en principe placés sur des CD, y compris un disque d'installation amorçable. Quelques indices automatiques peuvent guider l'administrateur mais finalement, toute la configuration est manuelle. Puisque tous les détails de sélection de paquetages, partitionnement disque, configuration matérielle et détails réseau doivent être entrés à la main, ce processus est consommateur de temps et sujet aux erreurs humaines. De nombreuses distributions disposent de leur propre programme d'installation, comme par exemple *anaconda* pour RedHat et *YAST2* pour SuSE.

2 Duplication d'image

Si des clones quasi-identiques sont adéquats, un « client en or » peut être installé manuellement puis répliqué. Des distributions exécutables telles que *Knoppix* (qui exécute un environnement GNU/Linux complet depuis un seul CD-ROM - voir http://www.knopper.net/knoppix/) et d'autres disques de secours peuvent être utilisés pour copier des images des systèmes de fichiers du client en or vers d'autres machines. La configuration peut être réalisée par des scripts exécutés avant ou après l'installation. Puisque des systèmes de fichiers entiers peuvent être copiés plutôt que les fichiers qu'ils contiennent, cette méthode est la plus rapide ; cependant, la configuration de clones non identiques est moins efficace et nécessite des compétences d'expert.

3 Installation entièrement automatique

FAI (http://www.informatik.uni-koeln.de/fai/) installe la distribution Debian automatiquement. On accède aux paquetages logiciels sur un site Debian qui peut être un miroir local pour des raisons de rapidité ou de configuration. Le noyau d'installation fourni peut être démarré depuis le réseau ou une disquette, mais le démarrage par CD n'est encore qu'en développement. Bien que FAI soit conçu pour une réplication à l'identique de grappes de machines, le logiciel cfengine décrit plus haut est utilisé pour la configuration système et autorise toute la souplesse nécessaire.

4 System imager

System Imager (http://www.systemimager.org/) automatise l'installation système, la configuration et la maintenance pour de grands réseaux de machines (de préférence avec des matériels similaires) à travers différents distributions. Il peut être amorcé par une disquette, un CD-ROM ou des serveurs réseau PXE. Debian et RedHat ont été testées mais le logiciel System Configurator utilisé devrait supporter toutes les distributions GNU/Linux.

Un « client en or » est installé et configuré à la main. Ses systèmes de fichiers sont dupliqués sur un serveur d'image depuis lequel les machines cibles sont installées. Si le « client

en or » est mis à jour, ces modifications sont propagées vers les clients répliqués par *rsync*. Bien que *rsync* transfère les différences au minimum sur le réseau, cela peut nécessiter une mémoire significative pour effectuer cela. Puisque les modifications sont relatives au « client en or », *System Imager* est plus adapté aux clients dont le matériel est très similaire, voire identique.

5 Kickstart de RedHat

Kickstart (http://www.tldp.org/HOWTO/KickStart-HOWTO.html) est le logiciel d'installation automatisé de RedHat. Il installe les distributions RedHat depuis un CD-ROM, un disque dur ou le réseau et s'amorce depuis le réseau, un CD ou une disquette. Le programme d'installation anaconda offre les deux interfaces graphique et texte et peut être interactif ou totalement automatisé par un fichier de configuration. Le logiciel de détection du matériel kudzu reconnaît automatiquement un gran nombre de périphériques. Les options générales d'installation peuvent être configurées dans le fichier de configuration et les extensions ajoutées par les scripts de pré- et post-installation.

Avec son logiciel intelligent de configuration et de détection, *kickstart* peut être utilisé pour automatiser des installations similaires sur une variété de cibles matérielles. La sélection de paquetages depuis la distribution standard *RedHat* s'effectue d'un trait mais les mises à jour ou extensions peuvent aussi être incluses par adaptation du processus *kickstart*.

12.4.3.2 Maintenance logicielle

Les installations de logiciels ne restent pas statiques au cours de leur existence. Des mises à jours de sécurité ou des corrections de bogues sont publiées après l'installation initiale. De plus, l'ajout ou la suppression de paquetages sera nécessaire pour la gestion logicielle sans réinstallation du système entier.

Autant que possible, les mises à jour doivent être réalisées selon des techniques tirantes plutôt que poussantes. La décision de télécharger des mises à jour doit être prise par une machine, serveur ou cliente, après que celle-ci se soit auto-vérifié par rapport à un serveur maître. Les mises à jour ne doivent pas être sous le contrôle des utilisateurs. Ainsi, les machines peuvent être maintenues au même niveau de versions.

1 Maintenance logicielle manuelle

Les administrateurs système peuvent effectuer la maintenance logicielle à la main. Cela peut impliquer la connexion à distance sur le client cible, la copie des paquetages mis à jour puis leur installation à l'aide du gestionnaire de paquetages natif de la distribution. Cependant, bien qu'elle offre un contrôle fin à l'administrateur, cette méthode est sujette à erreurs et rend la difficile la synchronisation de parcs importants. Certaines distributions offrent des outils de mise à jour permettant la maintenance de leurs paquetages standard mais nécessitent en principe toujours une intervention manuelle et peuvent ne pas convenir pour des extensions à la distribution de base.

2 Ximian Red Carpet

Red Carpet (http://www.ximian.com/products/redcarpet/) est une suite de mise à jour logicielle de Ximian librement disponible. Elle a débuté comme un gestionnaire graphique de paquetages pour le bureau virtuel de Ximian mais offre maintenant un accès distant sécurisé en ligne de commande et plus de canaux logiciels incluant les mises à jour de distributions. Mandrake, SuSE et RedHat sont actuellement supportées. Elle offre une administration distante et une automatisation aisées, ainsi de grands nombres de clients peuvent êtres maintenus en central. Cependant, quelques scories, issues de sa conception initiale, subsistent. Elle ne permet pas les mises à jour de noyau ni celles optimisées par architecture. Un produit serveur propriétaire, Red Carpet Enterprise peut être utilisé pour faciliter la gestion de grands parcs logiciels.

L'interface graphique ne doit pas être utilisée car elle permet aux utilisateurs de contrôler

les mises à jour. L'interface en ligne de commande doit être incorporée dans des scripts qui mettent à jour la machine automatiquement.

3 Red Hat Enterprise Network

RedHat offre une gamme de services de mise à jour logicielle dans leur produit propriétaire Enterprise Network (http://www.redhat.com/rhen/software_delivery/). Le plus puissant est *Satellite Server* qui permet une adaptation complète des mises à jours et correctifs. Tous les serveurs supportent leur client standard *Update Agent* pour la distribution. Les mêmes commentaires que ceux indiqués plus haut pour *Red* Carpet contre l'utilisation de l'interface graphique s'appliquent.

4 Debian APT

APT est une suite d'outils fournis avec la distribution Debian GNU/Linux qui permet les mises à jour automatiques des logiciels installés sur une machine. Elle est capable de contrôler les dépendances entre les paquetages installés et ceux disponibles sur les entrepôts pour lesquels elle a été configurée, ainsi que de télécharger et installer les mises à jour correspondantes disponibles sur un entrepôt. Les organisations peuvent configurer et maintenir leurs propres entrepôts de logiciels à installer sur les clients (Debian inclut les outils de configuration et de maintenance de ces entrepôts), utiliser les entrepôts fournis par Debian et d'autres, ou utiliser toute combinaison de ces sources de logiciels mis à jour. APT a été adapté pour fonctionner sur des systèmes fondés sur RPM tels que RedHat et Mandrake pour lesquels elle fournit des fonctionnalités similaires, et d'une certaine manière, par comparaison, améliorées, par rapport à Red Carpet.

12.4.4 Gestion matérielle et surveillance système

Le matériel peut être surveillé contre les défaillances avérées et potentielles, par exemple par l'utilisation de disques compatibles SMART et des systèmes de contrôle de santé du matériel. Les systèmes matériels et logiciels doivent être surveillés contre les défaillances avérées, potentielles, absence de service et limite de capacité.

12.4.4.1 MRTG et Snmpd

MRTG (Multi-Router trafic Grapher, http://people.ee.ethz.ch/~oetiker/webtools/mrtg/) est un outil de surveillance initialement conçu pour suivre et tracer un graphique de l'utilisation de la capacité de liens réseau. Cependant, il a évolué pour devenir un outil susceptible de tracer virtuellement toute quantité variable et peut être utilisé pour surveiller des variables telles que l'utilisation du processeur, de la mémoire, de l'espace disque, de services réseau (en incluant des statistiques sur le volume de courriel traité, pages web servies, etc.), température système et vitesses de ventilateurs et autres.

Snmpd (Simple Network Management Protocol Daemon, http://net-snmp.sourceforge.net/) est un serveur de gestion système qui peut s'exécuter sur tout poste d'une organisation. Il fournit des informations de gestion système à ses clients (typiquement, un client SNMP central qui agrège les statistiques de plusieurs machines). *MRTG* peut agir en tant que client SNMP et exécuter cette fonction, en fournissant une représentation graphique de l'état d'un grand nombre de machines clientes.

12.4.4.2 Nagios

Nagios (précédemment appelé *NetSaint*, http://www.nagios.org/) est un système de gestion configurable de machines, services et réseau. Il est capable de surveiller les services réseau et d'exécuter diverses procédures de reprise s'il découvre qu'un service est indisponible ou rencontre des problèmes, notamment par l'invocation de scripts de reprise automatique et l'alerte des administrateurs système. *Nagios* peut aussi fournir des rapports et représentations des états passés et présent des services qu'il surveille.

12.4.4.3 smartd

Le jeu d'outils *SmartMonTools* (http://smartmontools.sourceforge.net/) comporte un daemon appelé *smartd* conçu pour surveiller les fonctions SMART (Self-Monitoring, Analysis and Reporting Technology) des disques durs modernes. Puisque ces périphériques sont les plus sujets à défaillance dans un ordinateur moderne, SMART est prévu pour surveiller les paramètres du disque et alerter un administrateur système de défaillances potentielles avant que celles-ci arrivent. *smartd* est conçu pour recevoir ces alertes et effectuer des actions, typiquement en alertant un administrateur système.

12.4.5 Gestion d'impression

12.4.5.1 LPRng

LPRng (http://www.lprng.com/) est une implantation activement développée de l'ancien système standard BSD lpr/lpd. Il contient nombre d'améliorations qui le rendent beaucoup plus robuste et aisé à gérer que les produits d'origine. L'auteur est particulièrement courtois de s'assurer de la sécurité de *LPRng*. Jusqu'à récemment, c'était probablement le choix de gestion d'impression, mais *CUPS* a récemment fait des progrès et la situation est maintenant moins claire.

12.4.5.2 Common Unix Printing System

Le système commun d'impression Unix *CUPS* (http://www.cups.org/) est conçu pour être un système d'impression Unix de niveau entreprise. Il est fondé sur le protocole d'impression standard Internet IPP et incorpore une fonction de navigation qui permet de distribuer automatiquement sur le réseau les détails de noms et caractéristiques des imprimantes. *CUPS* comporte aussi une interface utilisateur web pour l'administration et la configuration des imprimantes. Des pilotes sont disponibles pour les imprimantes les plus courantes.

12.4.5.3 Kprint et GnomePrint

KDE et *Gnome* incluent leurs propres sous-systèmes d'impression, capables d'interfacer les applications utilisateur avec la plupart des systèmes de files d'impression, y compris *LPRng* et *CUPS*.

12.5 Sauvegarde et restauration

Toutes les données utilisateurs et de l'administration sont censées se trouver sur un ou plusieurs serveurs. Il est nécessaire d'être à même de réaliser des sauvegardes incrémentales, de trouver celles contenant des fichiers particuliers et de restaurer des fichiers individuels ou des systèmes de fichiers complets. La sauvegarde de données utilisateurs tend à être plus aisée avec Unix et les systèmes OSS qu'avec *MS-Windows* car les ceux-ci ainsi que leur configuration sont habituellement placés dans un seul répertoire. C'est un autre domaine dans lequel des produits propriétaires tels que *Legato* peuvent être nécessaires pour obtenir les fonctionnalités et le contrôle fin indispensables à un site de grandes dimensions.

12.5.1 Dump et Restore

Ces deux programmes sont inclus dans de nombreuses distributions et sont parfois utilisées en avec *tar* et *cpio* dans des scripts personnalisés pour sauvegarder et restaurer des machines individuelles.

12.5.2 Amanda

Amanda (voir http://www.amanda.org/) est un produit client-serveur conçu pour la sauvegarde de machines multiples sur un seul périphérique. Il est aussi capable de sauvegarder une machine *MS-Windows* via *Samba*.

12.6 Autres services

12.6.1 Serveurs de date

Il est essentiel, dans un environnement hautement réseau, que toutes les machines (serveurs et postes de travail) aient la même notion de l'heure. Un ou plusieurs serveurs sont désignés comme serveurs maîtres et obtiennent l'heure depuis une horloge attachée ou depuis des serveurs externes d'Internet. Toutes les autres machines sont des clients synchronisés sur ces maîtres.

La synchronisation de l'heure peut être effectuée par *ntp* (http://www.ntp.org/) qui peut maintenir un réseau de machines à moins d'une seconde les unes des autres.

Chrony (http://go.to/chrony/) est une alternative à *ntp*. Il a quelques fonctionnalités qui le rendent plus adapté que *ntp* aux noeuds NTP de haut niveau, bien que *ntp* soit meilleur pour les noeuds de bas niveau qui peuvent s'interfacer directement avec des équipements tels que récepteurs GPS et horloges atomiques. Un produit OSS pour *MS-Windows*, tel que *Automachron* ou *nettime* est aussi utile en environnement hétérogène - http://go.to/chrony/ fournit des détails sur les deux.

12.6.2 Serveurs d'infrastructure réseau

Ces services sont nécessaires à l'exploitation d'un réseau TCP/IP.

12.6.2.1 Routage

Les routeurs permettent de découper un grand réseau en petits segments interconnectés. Ils ont la tâche de diriger les paquets d'un sous-réseau vers un autre pour leur permettre de rejoindre leur destination prévue. La mise en place de routeurs nécessite une bonne compréhension des protocoles de base et de nombreuses administrations préféreront sans doute acquérir des routeurs propriétaires dédiés.

Néanmoins, pour ceux qui souhaitent bâtir leurs routeurs, deux produits existent : *Bird* (http://bird.network.cz/) et *GNU Zebra* (http://www.zebra.org/).

12.6.2.2 DNS

Un réseau TCP/IP nécessite un moyen de traduire les adresses IP en noms de domaines humainement compréhensibles et vice-versa. DNS rassemble un protocole et des serveurs inter-communicants dont chacun conserve des données. DNS est à la base du fonctionnement d'Internet. Il existe de nombreux programmes pour bâtir des serveurs DNS, notamment *BIND* (http://www.isc.org/products/BIND/), *MyDNS* (http://mydns.boy.net/) et *MaraDNS* (http://www.maradns.org/). *BIND* est le plus répandu.

12.6.2.3 DHCP

DHCP est un protocole décrit à http://www.dhcp.org/ qui permet à des machines d'obtenir leurs détails réseau à l'amorçage depuis un ou plusieurs serveurs. DHCP permet l'utilisation raisonnée d'adresses IP de plus en plus rares et réalloue celles-ci dès que possible. Il permet l'administration centralisée de nombreuses adresses globales telles que passerelles et serveurs de noms. Le produit phare est à http://www.isc.org/products/DHCP/ et consiste en une application client-serveur. Le client doit s'exécuter sur toutes les machines clientes participantes. Ces produits sont standard dans de nombreuses distributions.

12.6.3 Serveurs de fichiers

Les serveurs de fichiers permettent à des machines connectées en réseau d'accéder à des volumes de stockage sur une machine distante comme si ceux-ci étaient locaux.

12.6.3.1 NFS

C'est le standard de fait qui est utilisé depuis de nombreuses années. Le sous-ensemble

habituellement implanté n'assure pas une sécurité forte, bien qu'une variante sécurisée soit définie et implantée dans certains Unix commerciaux.

NFS consiste en un serveur qui exporte des fichiers de la machine sur laquelle il s'exécute vers les clients présents sur d'autres machines du réseau. On peut contrôler quelles machines peuvent importer ces fichiers, mais une fois l'attachement établi, le trafic sur le réseau s'effectue en clair. Il existe une authentification minimale des utilisateurs sur la version Unix.

L'autre problème avec tout système de fichiers réseau est qu'en cas de défaillance du réseau, l'accès aux fichiers est stoppé. Pour pallier cela, il est nécessaire d'utiliser un système de fichiers distribué (voir plus bas).

NFS est standard dans de nombreuses distributions.

12.6.3.2 Samba

Samba est un produit qui implante le protocole SMB de Microsoft (voir 14.5.1 pour une description plus détaillée). Il est indispensable à l'intégration de systèmes OSS et *MS-Windows* et se trouve dans la plupart des distributions standard. Son utilisation est décrite avec un certain détail au chapitre 14.

12.6.3.3 Netatalk

Pour ceux qui disposent de machines Apple Macintosh, *netatalk* fournit l'implantation du protocole AppleTalk (voir http://netatalk.sourceforge.net/).

12.6.3.4 OpenAFS, CODA et Intermezzo

Ces produits implantent, à des degrés divers, un système de fichiers distribué. Avec un tel système, l'accès aux fichiers peut continuer lorsque le réseau tombe car le cache local donne l'apparence du maintien de la connexion. C'est un problème non trivial et les produits le résolvent de différentes manières. Ce genre de systèmes de fichiers est réellement nécessaire avec les portables et machines attachées par une connexion non permanente. L'autre manière de fournir la même fonctionnalité est que le stockage local soit synchronisé périodiquement avec un serveur central (voir http://openafs.org/, http://www.coda.cs.cmu.edu/ et http://www.inter-mezzo.org/ pour les détails de chaque produit).

http://www.inter-mezzo.org/docs/bottelnecks.pdf contient une discussion détaillée des caractéristiques des produits précédents.

12.6.4 Services de répertoires

Ces services permettent le contrôle rapide de noms et adresses ainsi que de données associées.

Le standard le plus répandu pour les services de répertoires est LDAP. C'est un protocole ouvert implanté dans de nombreux produits (par exemple *Evolution* et *OpenOffice.org*). LDAP fonctionne avec des définitions de données appelées schémas et il est possible aux administrations de développer leurs propres schémas personnalisés. Malheureusement, les schémas utilisés par les applications ne sont pas toujours compatibles entre eux ce qui implique, par exemple, qu'il est difficile pour *OpenOffice.org* de lire les informations de *Evolution* et vice-versa.

L'application OSS *OpenLDAP* est conforme au standard LDAPv3 et les versions 2.1 et ultérieures peuvent être configurées avec plusieurs bases de données de stockage (comme les fichiers plats, SQL ou même spécifiques).

De nombreuses suites de travail de groupe fournissent une certaine forme de service de répertoire mais peut sont compatibles LDAP. Sauf à user des techniques de copier/coller, il est difficile d'utiliser leur base de contacts dans des agents externes de courriel. Beaucoup d'entre elles offrent leur propre agent de courriel mais ne sont pas très performantes en termes de niveau d'intégration disponible dans le gestionnaire de contacts intégré.

OpenOffice.org, *Evolution* et *Mozilla* fournissent des fonctions intégrales de carnets d'adresses. Cependant, les formats de stockage utilisés ne sont pas interchangeables. Pour permettre les échanges, une certaine adaptation du site est nécessaire.

12.6.5 Services de base

12.6.5.1 Émulation de terminal

L'utilisation de *xterm* avec une configuration appropriée de la variable d'environnement TERM permet d'émuler de nombreux types de terminaux texte, tels que VT220 et VT100. Une émulation spécifique 3270 s'appelle *x3270*. Les émulations des terminaux pages se trouvent dans des produits propriétaires.

12.6.5.2 Affichage distant

Voir la discussion en section 13.3 plus bas.

12.6.5.3 Émulation

Voir la discussion en section 13.4 plus bas.

13 Migration d'applications - vue générale

Une fois la liste des applications réalisée, celles-ci peuvent être placées dans l'une des catégories suivantes :

13.1 Applications propriétaires dont un équivalent OSS existe

Certaines applications, par exemple *MS-Office*, *Lotus SmartSuite*, *WordPerfect*, *Framemaker*, *Quark Express* et *Photoshop* ont leurs équivalents natifs en OSS, notamment *OpenOffice.org*, *Gnumeric*, *Evolution* et *The GIMP*. Dans ce cas, il faut évaluer le produit OSS pour s'assurer qu'il fournisse les fonctionnalités nécessaires.

13.2 Applications propriétaires qui fonctionnent en environnement OSS

Certaines applications telles que *Acrobat Reader* existent en version native pour environnement OSS. S'il n'existe pas d'alternative OSS à celle-ci, la seule contrainte est de s'assurer que toutes les fonctionnalités nécessaires soient implantées dans la version OSS. S'il existe une alternative OSS et qu'une migration partielle est acceptable, il faut fonder son choix sur les fonctionnalités offertes par les deux applications.

13.3 Logiciel pouvant être exécuté depuis un affichage déporté

Une autre approche est d'exécuter l'application sur un serveur et de déporter l'affichage jusqu'au poste ; c'est l'approche client léger. Des produits tels que *Windows Terminal Server*, *Citrix* et *Graphon* permettent à des applications de s'exécuter sur un serveur *MS-Windows* multi-utilisateurs. Cela impose que l'application, écrite pour s'exécuter en mode mono-utilisateur sur un poste, doive être modifiée pour s'exécuter avec l'un de ces produits. Cela n'est pas possible sans le code source et des éditeurs tiers peuvent ne pas être coopératifs.

Le plus sophistiqué de ces produits, *Citrix*, utilise son propre protocole de communication, « ICA », qui s'avère excellent, en particulier avec des connexions à faible bande passante. Il permet l'équilibrage de charge sur une ferme de serveurs en plus d'autres fonctionnalités utiles. Des clients ICA libres existent pour GNU/Linux.

Tous ces produits s'appuient sur du logiciel propriétaire à source fermée et *Citrix* est particulièrement cher. Il nécessite une licence serveur *MS-Windows*, une licence *Citrix* et une licence *MS-Windows Terminal Server* si un client non *MS-Windows* est utilisé. De plus, une licence d'accès client est nécessaire pour chaque poste utilisant le logiciel. La licence *Citrix* est calculée sur le nombre d'utilisateurs simultanés, ainsi cette approche peut être plus économique si de nombreux utilisateurs doivent avoir accès à une application mais qu'il y a peu d'accès concurrents. Des études de cas documentées se trouvent à http://www.citrix.com/press/news/profiles/, qui montrent que des clients légers « jetables » suffisent à justifier le rapatriement d'applications sur un serveur. *Citrix* dispose aussi de produits permettant un déplacement similaire d'applications par ICA et affichées sur un poste client léger.

MS-Windows Terminal Server fournit des fonctionnalités similaires à celles de *Citrix* avec un protocole différent, RDP. Le client GNU/Linux pour RDP, *Rdesktop*, est bon mais encore considéré comme expérimental par certains. RDP était très peu efficace en comparaison de ICA mais cette différence est maintenant faible, voire négligeable. *Citrix* a des fonctionnalités telles que l'équilibrage de charge qui en font un meilleur choix pour des installations de grande envergure lorsque le coût supplémentaire peut être justifié.

Citrix et *MS-Windows Terminal Server* peuvent introduire des latences dans l'application si les serveurs ne sont pas dimensionnés correctement ou que le réseau n'est pas suffisamment rapide.

Tarantella (http://www.tarantella.com/) réside sur un serveur placé entre le poste et les serveurs d'applications. Il agrège les flux de *Citrix* sur *MS-Windows* et ceux d'autres applications Unix et grands systèmes IBM et envoie le résultat sur le navigateur du poste. Il utilise son propre protocole propriétaire,

AIP, qui semble raisonnable avec des bandes passantes faibles. Cependant, il augmente la latence en raison de son positionnement entre l'utilisateur et l'application qui ralentit la connexion entre les deux.

Comme mentionné ci-dessus, CodeWeavers produit maintenant une version serveur de son produit *CrossOver Office*. Il fonctionne par une connexion sécurisée du client au serveur central auquel il sert une session X. Cela implique que la communication vers le serveur central est chiffrée et compressée mais nécessite une bande passante suffisante pour supporter un flux X. Les tests de bande passante minimale n'ont pas été réalisés mais il est vraisemblable que celle-ci soit supérieure à celles nécessaires pour ICA (*Citrix*) ou AIP (*Tarantella*).

VNC est un produit OSS développé par AT&T conçu pour afficher une session utilisateur exécutée sur une autre machine. Il consiste en un serveur et un client, tous deux disponibles pour *MS-Windows*, Unix et GNU/Linux. *VNC* permet aux applications de s'exécuter sur un environnement et l'affichage d'être déporté sur un autre. Il utilise son propre protocole, RFB, sur TCP/IP, qui n'est pas aussi efficace que ICA (*Citrix*) ou AIP (*Tarantella*) et nécessite donc des bandes passantes réseau élevées (autour de 100Mbps) pour bien fonctionner. Malheureusement, le serveur *MS-Windows VNC* n'est pas aussi efficace que la version Unix et peut nécessiter plus de puissance processeur que prévu. *VNC* peut être très utile pour une utilisation d'administration occasionnelle, en permettant ^la prise de contrôle d'un poste par un opérateur central. Dans ces circonstances, une latence élevée peut être acceptable.

13.4 Logiciels fonctionnant avec un émulateur

Si aucune des solutions précédentes ne permet l'exécution d'une application ou d'un substitut, il peut être possible de l'exécuter en natif, mais en émulant son environnement d'exploitation normal par-dessus un système d'exploitation OSS. Une bonne discussion des points relatifs à cette approche peut être trouvée sur http://linuxmednews.com/linuxmednews/967526746/index.html. Toutes ces techniques ont des implications en termes de licences car elles peuvent impliquer l'exécution de plusieurs copies de l'application et/ou du système d'exploitation propriétaires.

Cette section a de grandes chances d'être utilisée pour des applications *MS-Windows* mais puisque ces techniques peuvent s'appliquer à d'autres scénarios, elles sont étudiées ici plutôt qu'au chapitre 14.

Deux types d'émulation coexistent :

13.4.1 Émulation matérielle

Des produits tels que *VMware* ou *Win4lin* permettent l'émulation matérielle. Elles permettent à un système d'exploitation de PC ordinaire de s'exécuter comme des applications de niveau utilisateur en imitant le matériel Intel PC par des interfaces logicielles et donc en créant une machine virtuelle. Cela permet à un système d'exploitation natif et à ses applications de s'exécuter sur une plate-forme OSS.

VMware n'est pas un émulateur au sens strict - il permet à beaucoup d'instructions d'être passées directement au processeur, ce qui ne le rend utilisable que sur une machine à architecture x86. C'est l'offre la plus complète mais elle est propriétaire et peut consommer beaucoup de ressources machine.

Win4lin est similaire à *VMware* et tout aussi propriétaire, mais moins cher. Il peut représenter une bonne solution pour les cas simples - par exemple pour exécuter juste des applications de bureautique. C'est un composant du produit *Lindows* disponible pour des utilisations personnelles sur des matériels économiques (en raison de l'absence d'utilisation de comptes non privilégiés, *Lindows* lui-même ne peut être recommandé aux administrations sans une étude attentive de ses implications en termes de sécurité).

Puisque l'approche par émulation matérielle nécessite des licences complètes du système d'exploitation et des applications propriétaires à exécuter, en plus du coût de l'émulateur, elle doit être vue comme une manière d'exécuter un petit nombre d'applications officielles difficiles à migrer.

Il existe des produits serveurs pour *VMware* et *Win4lin* qui peuvent réduire les coûts de licences si le produit propriétaire dispose de licences ***concurrentes*** plutôt que ***par utilisateur potentiel***.

Des applications OSS émulent intégralement un environnement Intel IA-32, *Bochs* par exemple,

mais celles-ci ne sont probablement pas encore prêtes pour une utilisation par l'administration.

13.4.2 Émulation logicielle

L'émulation logicielle permet à des programmes écrits pour un environnement propriétaire de s'exécuter directement sur le système d'exploitation OSS. Tous les appels système réalisés par celles-ci sont traduites en leur interface OSS équivalente. La conséquence est que le système d'exploitation propriétaire n'est plus nécessaire.

Wine permet à des applications écrites pour *MS-Windows* de s'exécuter sur GNU/Linux par émulation logicielle. *Wine* est décrit en détail en annexe B. Le principal problème de *Wine* réside dans le nombre important d'appels système *Ms-Windows* (y compris les bogues) qu'il doit émuler.

Le code OSS de *Wine* est disponible sur http://www.winehq.org/ ou chez CodeWeavers à http://www.codeweavers.com/technology/wine/download.php.

CodeWeavers édite deux produits propriétaires, *CrossOver Office* et *CrossOver Plugin* fondés sur *Wine* et conçus pour supporter des applications *MS-Windows* spécifiques. Bien que ces produits soient propriétaires, des modifications de code sont périodiquement restituées à la version OSS de *Wine*.

CrossOver Office est conçu pour permettre l'exécution native d'applications telles que *MS-Office* et Lotus *Notes* sur GNU/Linux. Certains points sont importants mais le produit est en développement actif. Cependant, cette approche peut être appropriée pour certains utilisateurs en fonction de leurs besoins. *CrossOver Office* est maintenant disponible en version serveur ce qui supprime la nécessité d'une installation complète sur le poste et permet des fonctionnalités similaires à celles de *Citrix*.

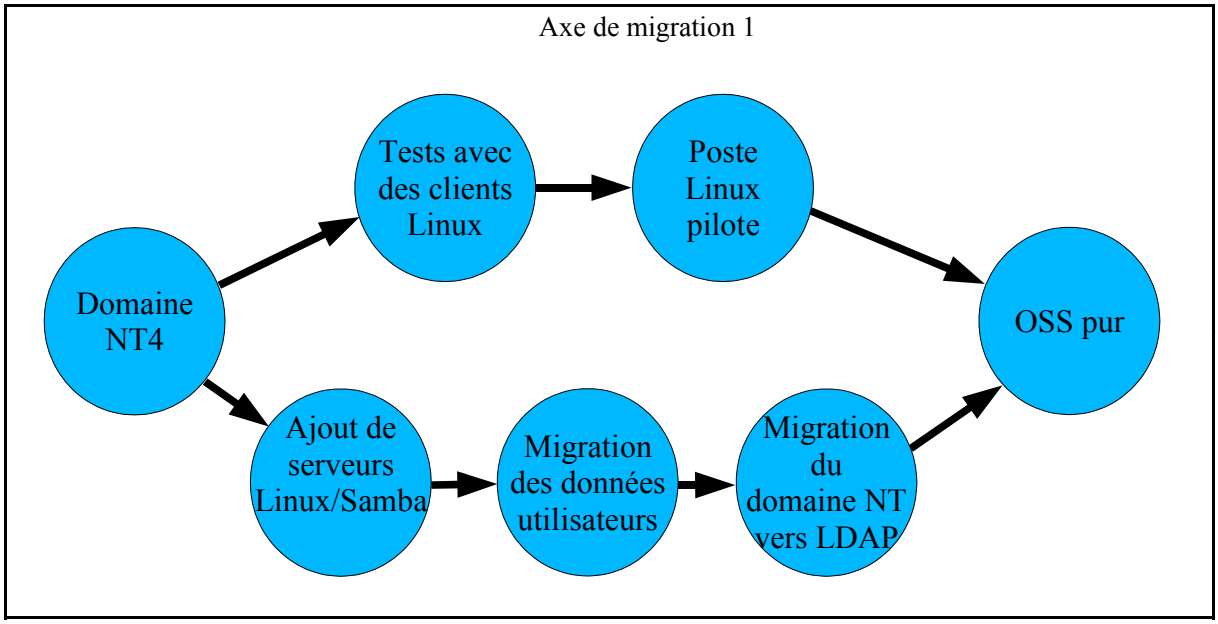
CrossOver Plugin est conçu pour permettre aux composants de navigateurs - qui ne fonctionnent normalement que sous *MS-Windows* - de s'exécuter avec *Netscape*, *Mozilla* et *Galeon* sous GNU/Linux. Ce produit est disponible depuis plus longtemps que *CrossOver Office* et fonctionne très bien.

L'utilisation de ces techniques supprime le coût des licences *MS-Windows* mais pas celui des applications. La licence applicative doit être analysée pour s'assurer qu'elle n'interdit pas l'exécution de l'application sans *MS-Windows*. Cette restriction est utilisée dans certaines nouvelles applications Microsoft comme technique de verrouillage, bien que sa légitimité légale soit discutable.

13.5 Logiciel pouvant être recompilé sous système OSS

Pour les applications maison ou écrites sous le contrôle de l'administration et pour lesquelles le code source est disponible, le logiciel peut être porté vers une plate-forme OSS. En général, le problème du portage d'un code source n'est pas la compilation mais l'utilisation que celui-ci fait des bibliothèques système, à la fois de l'environnement graphique et du système d'exploitation. Cela peut impliquer une quantité significative d'intervention manuelle pour la migration. De plus, toutes les suppositions faites sur l'environnement sous-jacent, tel que la dénomination des fichiers, peut rendre nécessaire soit des modification du code source, soit la réplication de l'environnement et ce, quel que soit le langage utilisé.

1. **Java**. Si le logiciel Java a été écrit en respect des spécifications Java, celui-ci doit fonctionner sans problème. Cependant, si des extensions propriétaires ont été utilisées, il faudra modifier le code pour lui faire utiliser les modules standard ;
2. **Visual Basic**. Un produit propriétaire appelé *DeLux* (http://www.deluxsoftware.com/) peut être utilisé pour convertir du code *Visual Basic* vers *Kylix* (voir item 4 ci-dessous) qui peut être exécuté nativement sous GNU/Linux. netproject n'a pas été en mesure de tester ce produit. Des outils de développement Microsoft permettent de convertir du code *Visual Basic* vers ***.NET*** et produire du code ***CIL***. Le projet OSS Mono permet à ce code de s'exécuter sous GNU/Linux. Mono est actuellement en développement très rapide et les applications peuvent fonctionner ou non en fonction de la manière dont elles interagissent avec les bibliothèques telles que l'affichage écran ;
3. **C#**. Ce langage est de plus en plus supporté sous GNU/Linux et Ximian a produit un compilateur pour le projet Mono, ajoutant des correspondances C# aux composants cruciaux du bureau virtuel


Axe de migration 1
Domaine NT4
Tests avec des clients Linux
Poste Linux pilote
OSS pur
Ajout de serveurs Linux/Samba
Migration des données utilisateurs
Migration du domaine NT vers LDAP

14 Scénario 1 - MS-Windows

L'administration dispose d'un ou plusieurs groupes de travail *MS-Windows*, des contrôleurs de domaine (PDC/BDC) *MS-Windows NT* ou des domaines *MS-Windows 2000 Active Directory*. Tous les utilisateurs ont des postes *MS-Windows*. Toutes les applications centrales s'exécutent sur des serveurs *MS-Windows*.

Tout au long de ce chapitre, le terme *MS-Windows* représente une version de *Microsoft Windows*. Lorsque la version est importante, elle est indiquée. Les exemples de code sont fondés sur un système *RedHat Linux* ; il peut y avoir de subtiles différences pour d'autres distributions.

Le contenu de ce scénario doit être lu en conjonction avec les commentaires généraux des chapitres précédents.

14.1 Planifier la migration

Pour récapituler le contenu du chapitre 5, la planification de la phase de transition est très importante ; le succès d'un projet OSS est jugé autant sur la fluidité de la transition que sur la qualité de service finale. Il est vraisemblable que toute transition d'un système vers un autre prenne place sur une période de plusieurs mois, voire années. Durant ce temps, les données doivent être déplacées, le personnel formé, les logiciels installés et le travail de l'administration doit se perpétuer sans interruption.

Une planification rigoureuse est nécessaire et les grandes administrations doivent passer par une phase pilote pour tester le plan avant de le mettre en oeuvre à grande échelle.

14.2 Domaines

Ce scénario peut être divisé comme suit :

14.2.1 « Groupe de travail » MS-Windows

Un groupe d'ordinateurs *MS-Windows* en relation faible par un nom sur le réseau constituent un « groupe de travail ». Le groupe de travail n'a aucun aspect sécurité - c'est simplement une manière commode de grouper des machines dans les listes de navigation.

Les utilisateurs qui souhaitent partager des fichiers avec d'autres peuvent créer des « partages » - parties de leur hiérarchie de répertoires - en accès libre ou protégés par mot de passe.

Aucune coordination des noms ni des mots de passe n'existe dans ce modèle. En fait, avec certaines versions de *MS-Windows*, il n'existe aucun concept réel de propriétaire.

La migration d'un modèle de groupe de travail vers un autre implique la collecte manuelle des fichiers importants, une machine à la fois.

14.2.1.1 Domaine MS-Windows NT

Dans ce modèle, un ou plusieurs ordinateurs agissent comme contrôleurs de domaine pour coordonner les noms et mots de passe. Une de ces machines serveur est désignée comme contrôleur primaire ou PDC et toutes les modifications sont assurées par celle-ci. Il peut aussi y avoir un ou plusieurs contrôleurs de secours ou BDC qui permettent la redondance et le partage de charge.

Les domaines *MS-Windows NT* incluent en général un ou plusieurs serveurs de fichiers (qui peuvent être confondus avec les PDC et BDC). Ceux-ci fournissent le stockage des profils (bureau virtuel, documents et configuration) et peuvent aussi héberger des espaces de « répertoires personnels », volumes partagés et des services de file d'impression.

Dans un domaine bien géré, les utilisateurs sont priés de conserver tous leurs fichiers dans leur bureau virtuel ou leur répertoire personnel, ainsi aucune donnée importante n'est conservée sur les PC. La

migration de données depuis des environnements ainsi bien gérés vers de nouveaux systèmes est relativement simple, puisque les administrateurs savent où trouver les fichiers importants.

14.2.2 Domaine MS-Windows 2000 Active Directory

Le modèle *MS-Windows NT* devient très difficile à gérer pour de nombreux utilisateurs et *MS-Windows 2000* a introduit un modèle de domaine hiérarchique. Celui-ci est connu sous le nom d'Active Directory ou AD et reprend des idées à la fois du DNS Internet et de LDAP.

Comme dans les domaines *MS-Windows NT*, AD fournit habituellement des serveurs de fichiers pour le stockage des profils et répertoires personnels et il devrait être simple de trouver les fichiers importants lors de la planification d'un processus de migration.

Puisque AD permet l'accès LDAP, un site AD permet plus d'options de migration : par exemple, il doit être possible d'utiliser les serveurs AD pour stocker le nom et le mot de passe des serveurs et clients OSS, ce qui devrait être utile lorsqu'une petite partie des utilisateurs doit migrer vers l'OSS, puisque le processus de gestion des utilisateurs peut rester quasi-identique.

14.3 Vue générale des principaux axes de migration

Les deux axes principaux considérés ici sont :

1. Ajouter des machines OSS aux domaines *MS-Windows* existant et déplacer progressivement les données et les utilisateurs, puis retirer les anciens serveurs propriétaires ; il est possible de migrer indépendamment les clients et les serveurs.
   L'ajout de serveurs au domaine *MS-Windows* est une des voies les plus rapides de tirer bénéfice de l'OSS. Par exemple, la combinaison GNU/Linux et *Samba* constitue un serveur de fichiers et d'impression performant et économique qui peut être utilisé en lieu et place d'un système *MS-Windows* sans aucune modification de l'environnement client.
   Des clients OSS d'un domaine *MS-Windows* constituent une forme de coexistence à faible risque, puisqu'aucune modification n'est nécessaire sur les serveurs. Cela peut être réalisé lorsqu'un petit nombre de personnes vont utiliser des postes OSS dans un environnement qui reste purement *MS-Windows* par ailleurs.
2. Bâtir une infrastructure OSS parallèle et migrer les utilisateurs et leurs données par groupes, avec une interaction minimale entre l'ancien et le nouveau systèmes.
   C'est beaucoup plus simple que l'option de coexistence *MS-Windows*/OSS mais rend nettement plus difficile la coopération entre les utilisateurs respectifs des deux systèmes.

Ces deux axes sont résumés dans le diagramme ci-dessous. Le premier axe permet une intégration plus fine entre les deux systèmes durant la transition mais nécessite significativement plus d'efforts de planification et d'implantation.

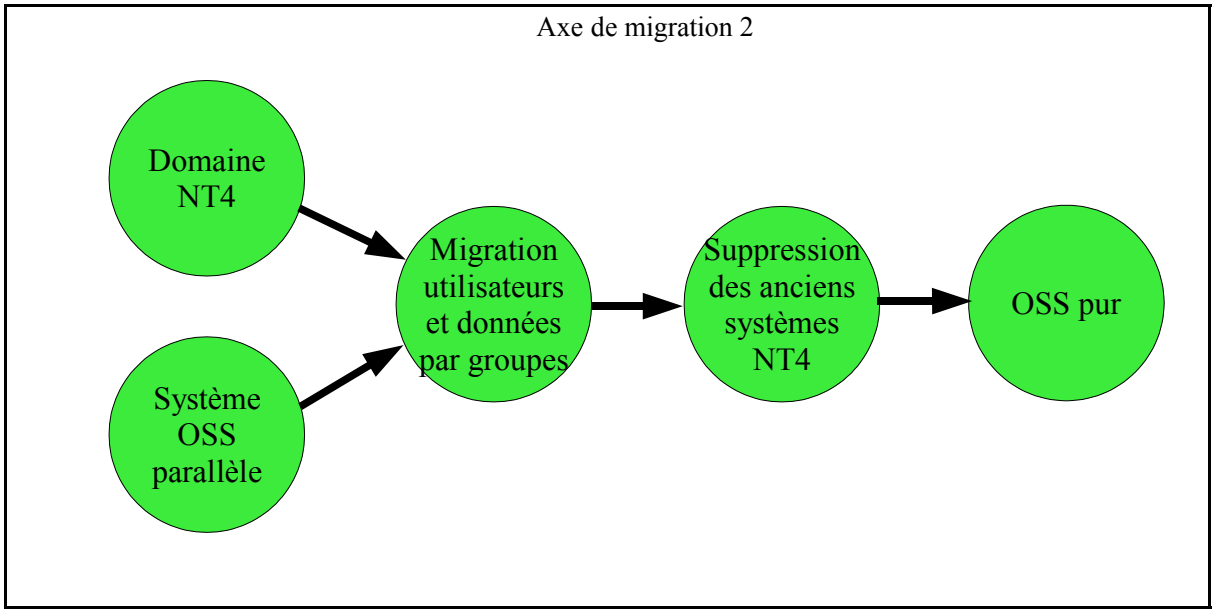
Une contrainte sur le choix de l'axe est la manière dont est organisée l'administration et comment cela s'imbrique dans la structure logique et physique de l'installation informatique.

Les premières étapes de nombreux axes de migration comportent une phase de coexistence durant laquelle les systèmes *MS-Windows* et OSS en place accèdent souvent aux mêmes données. Cela peut être un modèle particulièrement utile lorsqu'une migration partielle est planifiée, certains groupes migrant vers l'OSS tandis que d'autres restent à l'ancien système.

Les détails techniques de réalisation de ces modifications se trouvent en section 14.6 plus bas. Mais tout d'abord seront abordés les pré-requis techniques et outils nécessaires.

14.4 Points généraux

Il existe de nombreuses similarités entre les systèmes propriétaires actuels et les systèmes



OpenSource qui peuvent être amenés à les remplacer. En particulier, les interfaces graphiques (I.H.M.) tendent à converger vers un aspect assez standard qui réduit les difficultés des utilisateurs migrant d'un système vers un autre. La formation des utilisateurs finals reste indispensable afin d'aider le personnel à travailler avec les éléments différents et leur permettre de tirer le meilleur du nouveau système.

Derrière l'apparence similaire des I.H.M., il existe quelques différences importantes entre *MS-Windows* et les systèmes OSS. Celles-ci sont particulièrement apparentes au niveau de l'administration système. C'est là que la plus grande quantité de formation et de planification sera nécessaire. Des systèmes OSS tels que GNU/Linux ont des I.H.M. d'administration mais les installations importantes sont habituellement gérées avec des outils en ligne de commande qui eux-mêmes tendent vers l'écriture de scripts, l'automatisation de processus, l'administration à distance et le contrôle avancé. C'est la possibilité d'automatiser des tâches qui rend les administrateurs Unix et OSS si productifs.

En plus de ces différences dans les processus d'administration, il existe aussi d'importantes différences de service fourni. Il est nécessaire de planifier celles-ci et d'en tenir compte durant la transition.

14.4.1 Noms et mots de passe

Les utilisateurs d'ordinateurs s'identifient à l'aide de noms et mots de passe. Dans certaines administrations, on utilise aussi de cartes à puce ou autres dispositifs cryptographiques améliorant l'identification.

14.4.1.1 Problèmes de noms

Certaines administrations peuvent utiliser des noms « structurés » qui encodent des informations sur l'utilisateur (par exemple, le nom gcg*27* peut désigner la 27ème personne ayant rejoint le groupe comptable général) tandis que d'autres autorisent leur personnel à choisir lui-même son nom et son mot de passe ou utilisent simplement le nom réel. Les schémas de noms structurés peuvent en principe être utilisés sans modification avec l'OSS ; les noms ne peuvent cependant y débuter par un chiffre, ce qui peut causer des problèmes avec les noms structurés dont la structure initiale peut être numérique.

Certains points affectent les systèmes plus spécialisés. Les noms sous *MS-Windows* sont généralement insensibles à la casse (majuscules ou minuscules) tout en préservant celle-ci ; cela veut dire que le titulaire du nom « Marie » peut se connecter indifféremment avec « marie », « MARIE » ou même « mArIe » ; de même, lorsque le système affiche un nom (tel que celui du propriétaire d'un fichier), il présente la forme d'origine de celui-ci (dans ce cas par exemple, « Marie »).

À l'inverse, les noms Unix et OSS sont sensibles à la casse ; l'utilisateur doit entrer son nom dans la forme exacte d'enregistrement initial. Par convention, les noms sont constitués exclusivement de minuscules et chiffres sans autre signe et ne dépassent pas huit caractères.

Ces restrictions ont été largement assouplies durant les dernières années et les systèmes modernes permettent des noms beaucoup plus longs avec un jeu de caractères plus étendu. Certains schémas d'authentification et d'autorisation implantent désormais l'insensibilité à la casse : le schéma LDAP proposé dans le présent document en est un, ainsi les noms comme « Marie » et « DirecteurFinancier » sont possibles. Il faut cependant faire attention car certains paquetages anciens supposent l'application des anciennes règles. En particulier, il serait très imprudent de permettre l'utilisation d'espaces ou de certains autres caractères de ponctuation dans les noms.

Une bonne pratique consiste à limiter les noms aux caractères possibles pour les adresses courriel afin que le nom de connexion puisse être aussi utilisé comme nom de courriel.

14.4.1.2 Problèmes de mots de passe

Les systèmes OSS modernes permettent des mots de passe de toute longueur utilisant un large jeu de caractères ; il est de bonne pratique d'encourager l'utilisation de mots de passe longs (10 caractères ou plus) mélangeant lettres, chiffres, ponctuations et casse variable. Les utilitaires de modification de mots de passe refusent généralement des combinaisons trop faibles à moins d'y être forcés par un administrateur et de nombreux sites décident de configurer des règles encore plus fortes.

Certaines variantes Unix commerciales tronquent toujours les mots de passe à huit caractères ; si donc un environnement hétérogène est prévu, cela doit être pris en compte.

La migration de mots de passe depuis des systèmes propriétaires existant vers de nouveaux systèmes OSS n'est pas toujours possible, car ceux-ci sont généralement stockés dans une forme chiffrée et hachée. Le plan de transition peut inclure la redistribution de mots de passe à tous les utilisateurs ou éventuellement une phase de recueil et de synchronisation des mots de passe.

14.4.2 Services d'authentification

Tout réseau de plus d'une poignée d'ordinateurs nécessite un service de dénomination et d'authentification. Dans *MS-Windows NT*, cela s'appelle le contrôleur de domaine. Dans les systèmes *MS-Windows* plus récents, c'est Active Directory. Le NDS de Novell est aussi largement installé et d'autres systèmes propriétaires disposent de leurs propres systèmes.

La plupart des Unix et systèmes OSS peuvent interopérer avec quasiment tous les services de dénomination et d'authentification. GNU/Linux est particulièrement robuste sur cet aspect. Le service proposé dans ce document est fondé sur LDAP mais il est aussi possible d'utiliser plusieurs systèmes simultanément, ce qui peut être utile durant la phase de transition.

14.4.3 Fichiers

Une partie très importante dans tout plan de transition concerne la migration des données de l'ancien système vers le nouveau. Si une migration « big bang » est planifiée, ce sera une opération « coup de poing » mais si le fonctionnement parallèle plus vraisemblable est envisagé, des accès inter-plates-formes seront nécessaires. On doit faire très attention d'éviter les pertes de données ainsi que la confusion qui peut résulter de l'existence de copies distinctes modifiables d'un même fichier.

14.4.3.1 Contenu et format

C'est le point le plus délicat de la migration et il est décrit en détail en section 14.8 ci-dessous. L'approche normale est l'utilisation d'applications OSS qui puissent lire les fichiers écrits par l'application propriétaire qu'elles remplacent, quoique dans certains cas il puisse être approprié de planifier une conversion de format massive dans le processus de migration.

Les données spéciales telles que macros et scripts nécessiteront vraisemblablement l'attention de programmeurs expérimentés.

14.4.3.2 Noms de fichiers

Comme pour les noms d'utilisateurs, les noms de fichiers *MS-Windows* sont insensibles à la casse et préservent (en partie) celle-ci. Certaines applications *MS-Windows* tendent à capitaliser l'initiale des noms de fichiers ainsi qu'effectuer d'autres modifications dont l'utilisateur n'est pas informé. L'environnement *MS-Windows* traîne aussi l'héritage du format de noms MS-DOS « 8.3 » qui apparaît toujours dans certains utilitaires. Les noms de fichiers *MS-Windows* contiennent souvent des espaces et utilisent normalement le jeu de caractères Unicode ; le séparateur de répertoires est « \ ».

Bien que cela soit moins sensible pour les utilisateurs d'I.H.M., les noms de fichiers absolus de *MS-Windows* doivent comporter une « lettre de lecteur » indiquant le périphérique physique de stockage du fichier ou le nom réel du serveur si le fichier est sur un « disque réseau ». Ces restrictions peuvent poser des problèmes aux administrateurs de grands systèmes *MS-Windows* qui tentent de fournir un service stable devant des modifications matérielles.

Les autres systèmes propriétaires traitent les noms de fichiers différemment (*VMS* par exemple a des noms de fichiers insensibles à la casse qui comportent usuellement un point et peuvent comporter un numéro de version après un point-virgule).

Les noms de fichiers Unix et OSS suivent des règles différentes : ici, les noms de fichiers sont totalement sensibles à la casse et le système n'effectue aucune modification des noms fournis par l'utilisateur. Les noms utilisent un jeu de caractères 8 bits défini par la « locale » utilisée (dans l'essentiel de l'Europe, le jeu de caractères est ISO 8859-15). Les seuls caractères interdits par GNU/Linux dans les noms de fichiers sont le séparateur de répertoires « / » et le caractère « NULL ». Cependant en pratique, il est imprudent d'inclure des caractères non imprimables (par exemple, le système de fichiers *MS-Windows* FAT32 ne peut pas stocker les 32 premiers codes ASCII ni aucun des caractères ", *, ;, <, >, ?, ni |) ; les espaces sont permis, quoique leur présence nécessite plus de prudence pour les utilisateurs de la ligne de commande.

Les systèmes Unix et OSS n'utilisent pas de lettre de lecteur ni n'imposent le nom réel du serveur de fichiers dans le nom de fichier absolu. En lieu et place, le système présente tous les fichiers dans une seule hiérarchie continue. Avec l'utilisation des liens symboliques dans le système de fichiers et des auto-monteurs pilotés par les données, cela donne une grande souplesse pour les administrateurs en séparant le nom absolu d'un fichier de son emplacement physique de stockage.

À peu près tous les noms de fichiers *MS-Windows* peuvent être utilisés directement par des serveurs OSS. La seule exception qui puisse être rencontrée en pratique est pour les noms de fichiers contenant le caractère « / », ce qui devra être modifié durant la transition. Les utilisateurs d'outils I.H.M. ne s'apercevront probablement jamais que les noms de fichiers sont devenus sensibles à la casse car ils ne saisissent ceux-ci que lors de leur création initiale.

14.4.3.3 Accès mixte

Dans de nombreux plans de migration on trouve une période d'exploitation parallèle durant laquelle certains utilisent des systèmes OSS tandis que d'autres restent sur les anciens systèmes propriétaires. Lorsque des fichiers doivent être accessibles par des membres des deux groupes, des précautions spéciales peuvent être nécessaires.

Le partage de fichiers dans les systèmes *MS-Windows* s'appuie sur le protocole SMB (Server

Message Block) qui est une technologie très complexe avec plusieurs niveaux de compatibilité descendante. Il est utilisé sur les serveurs de fichiers dédiés aussi bien qu'en mode «partage » dans lequel des postes rendent accessibles par le réseau des portions de leur système de fichiers. Les environnements administratifs bien gérés sont en principe fondés sur des serveurs dédiés plutôt que sur le partage ponctuel.

Les fichiers utilisateurs non partagés d'un environnement *MS-Windows* peuvent être stockés à différents endroits :

1. sur un disque local du poste ou du portable - par exemple celui appelé « disque C » ;
2. dans le « profil itinérant » de l'utilisateur - cela inclut de nombreux réglages personnels ainsi que le contenu du bureau *MS-Windows* ainsi que (habituellement) le dossier « Mes documents ». Le profil itinérant est stocké sur tout PC utilisé activement par l'utilisateur et synchronisé sur un stockage central lors de la déconnexion. Cela fournit un système de sauvegarde pratique mais peut avoir de sérieuses implications en performances avec des utilisateurs signalant des déconnexions très longues ;
3. dans un « répertoire personnel » d'un serveur de fichiers centralisé ; c'est une option commune dans les grands réseaux de postes car elle permet de gérer correctement les sauvegardes.

Il est peu utile de tenter de fournir un accès mixte aux fichiers stockés sur les postes ou portables individuels, ainsi tous les fichiers des disques locaux ou profils itinérants doivent être déplacés assez tôt sur des serveurs de fichiers gérés.

Le principal mécanisme réseau d'accès aux fichiers pour les systèmes Unix et OSS est NFS. C'est un protocole beaucoup plus simple que SMB et ses spécifications ont toujours été ouvertement disponibles.

Les options d'implantation d'accès mixte se résument à deux grandes catégories : ajouter un support bi-protocole aux serveurs ou ajouter celui-ci aux clients. Toutes choses égales par ailleurs, il est normalement plus simple de modifier des serveurs que des clients et à peu près toujours plus aisé d'ajuster des systèmes OSS que des systèmes propriétaires. Les options sont résumées dans la table :

	Serveurs *MS-Windows*	**Serveurs OSS ou Unix**
Clients *MS-Windows*	L'accès SMB est standard	Les serveurs acceptent SMB par le paquetage Samba. C'est un logiciel mature d'excellentes performances.
Clients OSS	Les clients GNU/Linux peuvent accéder aux partages SMB. C'est assez simple si les clients ne servent qu'un utilisateur mais devient plus complexe si les machines sont utilisées en temps partagé. Les variantes commerciales d'Unix n'ont en principe pas de capacités clientes SMB. Il est possible d'ajouter le service NFS à des serveurs *MS-Windows* mais cela peut devenir très cher.	L'accès NFS est standard. Les clients GNU/Linux peuvent utiliser SMB si cela est souhaitable en fonction du plan de migration, mais cela est moins efficace.

14.5 Outils

Cette section aborde quelques-uns des composants-clefs de l'OSS utilisés lors de la migration depuis des systèmes propriétaires.

14.5.1 Samba

Samba est un paquetage serveur de fichiers et d'impression pour les systèmes OSS. Il implante le protocole SMB de Microsoft et peut dans de nombreux cas remplacer les fonctions d'un serveur *MS-Windows*. *Samba* peut aussi agir comme contrôleur de domaine *MS-Windows NT* et il est capable de stocker des données de gestion de domaines dans un répertoire auquel il accède par LDAP.

Samba contient aussi des outils client permettant la réalisation de scripts, ce qui est très utile pour diagnostiquer des problèmes avec les réseaux SMB comme lors de migrations de masse depuis des serveurs

MS-Windows.

Samba est maintenu par un groupe de base d'environ 30 volontaires très actifs autour du globe. On peut trouver plus d'informations à http://www.samba.org/.

14.5.2 OpenLDAP

OpenLDAP est une implantation du protocole LDAP. Il inclut un serveur de répertoire, un ensemble d'outils d'accès et d'administration ainsi qu'un jeu de bibliothèques destinées au support LDAP dans d'autres applications.

14.5.3 NSS et PAM

Name Service Switch (NSS - litt. : aiguilleur de services de noms) est une technologie utilisée par GNU/Linux et certaines variantes commerciales d'Unix pour permettre l'utilisation de différents services de noms lors de la recherche de machines, utilisateurs, groupes, etc. De nombreux modules sont disponibles dont les plus utiles pour un projet de migration sont :

1. fichiers : recherches simples dans des fichiers textes locaux ;
2. DNS : recherche de machines avec le protocole DNS ;
3. LDAP : recherches LDAP - essentiellement pour les noms d'utilisateurs et de groupes mais aussi utilisable pour de nombreuses autres utilisations ;
4. SMB : recherches par le protocole SMB de *MS-Windows* (voir 14.5.5 ci-dessous).

Le système PAM (Pluggable Authentication Module - litt : module d'authentification enfichable), comme NSS, est habituel sur GNU/Linux et certains dérivés commerciaux. Il est aussi disponible pour *FreeBSD*. Il permet une grande souplesse de configuration du processus d'authentification et d'autorisation. Les modules concernés sont :

1. LDAP : utilisation de LDAP pour vérifier les droits d'un utilisateur ;
2. SMB : utilisation d'opérations *MS-Windows NT* pour vérifier les droits d'un utilisateur ;
3. Access : restriction d'accès à des services réseau ;
4. Cracklib : renforcement des contrôles de qualité des nouveaux mots de passe.

14.5.4 Accès aux fichiers par SMBFS

Samba permet à un système OSS de fournir un service de fichiers à des clients *MS-Windows*. SMBFS fonctionne à l'opposé : il permet à un système OSS d'accéder à des fichiers stockés sur des serveurs *MS-Windows*. SMBFS est fourni avec les distributions GNU/Linux les plus récentes mais ne se trouve pas habituellement dans les systèmes Unix commerciaux.

Le modèle de contrôle d'accès utilisé par les systèmes de fichiers *MS-Windows* est différent de celui utilisé par les systèmes OSS et il existe quelques limitations à ce qui peut être réalisé avec SMBFS.

14.5.5 Winbind

Un autre produit de l'équipe Samba, *Winbind*, permet à des machines GNU/Linux de s'attacher à des domaines *MS-Windows NT*. Il maintient une correspondance entre les identificateurs (SID) *MS-Windows NT* et les UID et GID de style Unix. *Winbind* peut effectuer de nombreuses autres choses qui réduisent la charge des administrateurs système, tel que configurer des environnements de style Unix lors de la première connexion d'un utilisateur.

Le désavantage de *Winbind* dans les réseaux de grande ampleur est que chaque client construit sa propre table de correspondance entre les identificateurs *MS-Windows* et Unix. Cela peut poser problème dans les dernières étapes de migration, lorsque des serveurs de fichiers OSS sont introduits.

Lors de l'utilisation de *Winbind*, les noms d'utilisateurs et de groupes utilisés par GNU/Linux sont

formés par la concaténation du nom de domaine avec l'identifiant *MS-Windows NT* pour former une chaîne unique. Cela peut prêter à une certaine confusion car de nombreux utilitaires Unix n'affichent que les huit premiers caractères des identifiants. Les chaînes plus longues engendrées par *Winbind* sont tronquées à l'affichage.

14.6 Migrer l'environnement système d'exploitation

14.6.1 Ajout de serveurs GNU/Linux dans un domaine MS-Windows NT existant

La configuration est extrêmement simple :

1. installer un serveur GNU/Linux en lui donnant une adresse IP fixe ;
2. s'assurer de l'installation des paquetages *Samba* (typiquement *samba*, *samba-common* et *samba-client* sont nécessaires) - ceux-ci sont normalement inclus dans une installation « serveur » ;
3. éditer **/etc/samba/smb.conf**, configurer le mode de sécurité **domain** et définir le nom de domaine (groupe de travail), puis lister le PDC et tous les BDC comme serveurs de mots de passe et définir les partages que la machine doit servir ;
4. créer les répertoires qui seront partagés et placer les propriétés et permissions adaptées ;
5. insérer la machine dans le domaine *MS-Window NT* à l'aide du mot de passe de l'administrateur de domaine (ou tout couple utilisateur/mot de passe ayant les droits pour cela) :
   ```
   smbpasswd -j DOMAINNAME -r PDCNAME -U Administrator
   ```
6. démarrer *samba* et le configurer pour un démarrage automatique à l'amorçage :
   ```
   /etc/init.d/smb start
   chkconfig smb on
   ```

Le serveur doit alors apparaître dans les listes de navigation et peut être utilisé exactement comme un serveur *MS-Windows NT*.

14.6.2 Utiliser des postes GNU/Linux dans des domaines *MS-Windows NT*

14.6.2.1 Configuration simple pour un petit nombre de machines

Dans les premiers temps d'évaluation des outils OSS, il est très utile d'avoir quelques machines GNU/Linux avec des configurations très simples. Il est possible depuis celles-ci d'avoir accès aux fichiers des serveurs *MS-Windows* pour des tests de compatibilité et de migration avec la commande **smbmount**.

Le montage est le terme Unix/OSS qui désigne l'attachement d'un disque ou d'un système de fichiers distant dans la hiérarchie locale. Le processus est habituellement réalisé automatiquement lors de l'amorçage sous le contrôle du fichier **/etc/fstab** mais peut aussi être réalisé interactivement. Par exemple, la commande qui installe un CD-ROM standard ISO sous **/mnt/cdrom** serait :

```
mount /dev/cdrom /mnt/cdrom
```

La commande **mount** est normalement réservée à l'usage du super-utilisateur *root* pour des raisons de sécurité. Cela n'est pas un problème lorsque la machine est utilisée par un administrateur système mais peut devenir embarrassant si un utilisateur non-technicien est concerné. GNU/Linux fournit différents contournements :

1. l'utilisation d'une entrée spéciale dans **/etc/fstab** permet à des utilisateurs ordinaires de monter certains objets pré-définis (c'est la méthode habituelle de montage des CD-ROM et disquettes à la demande) ; les fichiers du périphérique monté apparaissent habituellement comme la propriété de l'utilisateur qui a monté ce périphérique ;
2. l'utilisation du programme **setuid-root** pour effectuer les opérations privilégiées après en avoir contrôlé l'innocuité ; c'est la méthode la plus simple de réaliser le montage de partages distants *MS-Windows* ;
3. l'utilisation d'un **automonteur** qui réalise le montage des systèmes de fichiers lorsque l'on y accède

et le démontage de ceux-ci lorsqu'ils ne sont plus utilisés ; l'automonteur est un daemon habituellement piloté par des données de configuration de la dimension du réseau. Cela nécessite un effort plus grand à configurer que les autres méthodes, mais c'est extrêmement utile pour les grands réseaux.

Dans ce schéma, nous utilisons les commandes **smbmount** et **smbumount** pour faire apparaître un partage *MS-Windows* existant dans notre hiérarchie locale GNU/Linux. Dans *RedHat Linux* celles-ci font partie du paquetage *samba-client*, il suffit donc que celui-ci soit installé ainsi que *samba-common*. Ces programmes sont conçus de manière à ce que les parties critiques puissent recevoir les privilèges *root* bien qu'ils ne soient normalement pas installés ainsi par défaut ; quelques commandes devront donc être exécutées par *root* avant l'utilisation :

```
chmod u+s /usr/bin/smbmnt /usr/bin/smbumount
```

Notez que la commande modifie **smbmnt** et non **smbmount**. Cela est important car **smbmnt** encapsule uniquement les fonctions de **smbmount** qui nécessitent les privilèges *root*. Ainsi, tout utilisateur peut utiliser **smbmount** et **smbumount** qui s'exécuteront avec les privilèges *root* nécessaires.

Désormais, tout utilisateur peut attacher un partage *MS-Windows* disponible dans sa hiérarchie locale en le montant dans un répertoire dont il est propriétaire ; tous les fichiers présents dans le répertoire en question seront invisibles pendant toute la durée du montage correspondant.

Par exemple, supposons que l'utilisateur *charles* veuille accéder aux fichiers d'un serveur *MS-Windows NT* appelé NT4FINANCE du domaine MINISTERE, partagé sous le nom CDUPONT et appartenant à l'utilisateur *MS-Windows* CDUPONT ; *charles* commence par créer un nouveau répertoire où monter le partage *MS-Windows* :

```
mkdir ~/ntfinance
```

Cela ne doit être effectué qu'une fois (la notation « ~/ » signifie « dans mon répertoire personnel »). Ensuite, pour monter le partage distant :

```
smbmount //nt4finance/cdupont ~/ntfinance \
-o username=cdupont -o workgroup=ministere
```

La commande (qui doit être entrée sur une seule ligne ou scindée par le caractère de continuation « \ » indiqué ci-dessus) demandera le mot de passe de CHARLES sur le serveur puis réalisera le montage du partage *MS-Windows* sur le répertoire *ntfinance* dans le répertoire personnel de *charles*. Pour éviter la frappe de cette ligne à chaque connexion, celle-ci peut être placée dans un fichier script ou même être intégrée dans le processus de connexion de *charles*.

Les partages montés se comportent désormais quasiment comme s'ils étaient sur le disque local : les fichiers peuvent être créés, supprimés et modifiés. Il existe certains pièges malgré tout ; en particulier, il n'existe pas de correspondance entre le contrôle d'accès Unix et les ACL *MS-Windows NT* donc les commandes de modification de propriété ou de mode des fichiers et répertoires du partage monté seront sans effet.

Avant la déconnexion, il pourrait être opportun de démonter le partage :

```
smbumount ~/ntfinance
```

De même, cela peut être intégré au processus de déconnexion si nécessaire.

Le processus décrit dans cette section ne crée aucun lien permanent entre des comptes GNU/Linux et d'autres sur des serveurs *MS-Windows NT*, donc les noms et mots de passe doivent être maintenus séparément sur chaque machine. L'effort de gestion induit peut rapidement devenir excessif au fur et à mesure de l'augmentation du parc, donc ce schéma n'est adapté qu'aux petits environnements de tests.

14.6.2.2 Configuration plus avancée pour parcs plus importants

Lorsqu'il devient nécessaire de déployer un pilote OSS de plus grande envergure, il peut rester opportun de conserver les fichiers et les services d'authentification sur des serveurs *MS*-Windows NT

existant. Le daemon *Winbind* de *Samba* fournit un moyen aisé de relier les deux environnements.

Samba et *Winbind* sont des éléments standard de la distribution *RedHat Linux* mais peuvent ne pas être installés par défaut sur des configurations poste de travail. Pour utiliser *Winbind*, il faut installer les paquetages *samba*, *samba-common*, et *samba-client*.

Le fichier **/etc/samba/smb.conf** doit contenir le nom de domaine correct à la ligne **workgroup** et placer le système dans le mode de sécurité **domain**. Les données de configuration *Winbind* se placent aussi dans la section globale de ce fichier, par exemple :

```
# on separe domaines et noms par un '+', comme ceci : DOMAINE+nom
winbind separator = +
# on utilise les uids de 10000 a 20000 pour les utilisateurs du domaine
winbind uid = 10000-20000
# on utilise les gids de 10000 to 20000 pour les groupes du domaine
winbind gid = 10000-20000
# on autorise l'enumeration des utilisateurs et groupes de winbind
winbind enum users = yes
winbind enum groups = yes
# on donne un repertoire personnel aux utilisateurs winbind
template homedir = /home/winnt/%D/%U
# ainsi qu'un shell
template shell = /bin/bash
```

Pour que *Winbind* fonctionne, certains services doivent être lancés ; pour obtenir cela et s'assurer que ceux-ci démarrent à chaque amorçage, les commandes sont :

```
chkconfig smb on
chkconfig winbind on
/etc/init.d/smb start
/etc/init.d/winbind start
```

La machine doit maintenant rejoindre le domaine *MS-Windows NT* ; cela nécessite un couple nom/mot de passe disposant des permissions appropriées (habituellement, *Administrateur*) :

```
smbpasswd -j DOMAINNAME -r PDCNAME -U Administrator
```

Il doit désormais être possible d'accéder aux listes d'utilisateurs et groupes *MS-Windows NT* par la commande *wbinfo* :

```
wbinfo -u
wbinfo -g
```

Pour rendre les données *Winbind* accessibles au système, il est nécessaire d'éditer les fichiers de configuration PAM et NSS ; cela doit être réalisé avec grande prudence car il est possible de se retrouver verrouillé hors du système si ces fichiers sont endommagés. Dans **/etc/nsswitch.conf**, ajouter le mot **winbind** aux lignes **passwd** et **group**. Dans **/etc/pam.d/system-auth**, ajouter une ligne de la forme :

```
auth sufficient /lib/security/pam_winbind.so use_first_pass
```

juste après la ligne **auth** équivalente qui utilise **pam_unix**, ainsi qu'une ligne de la forme :

```
password sufficient /lib/security/pam_winbind.so use_first_pass
```

juste après la ligne **password** équivalente qui utilise **pam_unix**.

Il sera nécessaire de relancer le daemon cache du service de noms à cette étape :

```
/etc/init.d/nscd restart
```

La translation des noms et groupes *MS-Windows* en leur version Unix au format du fichier passwd est alors accessible :

```
getent passwd
getent group
```

Pour automatiser la création des répertoires personnels des utilisateurs lors de la première connexion, on peut ajouter la ligne suivant dans la partie **session** de **/etc/pam.d/system-auth** :

```
session     required     /lib/security/pam_mkhomedir.so     skel=/etc/skel/
umask=0022
```

Cela créera un répertoire personnel Unix séparé pour l'utilisateur sur chaque poste qu'il utilise. Il peut aussi être utile de placer un script dans le répertoire **/etc/skel** pour automatiser le montage de fichiers *MS-Windows NT* à un emplacement standard lors de la connexion.

14.6.3 Utiliser des postes GNU/Linux dans des domaines Active Directory

En principe, les postes GNU/Linux peuvent rejoindre les domaines AD (Active Directory) d'une manière très similaire à celle dont ils peuvent rejoindre les domaines *MS-Windows NT*. En fait, si le domaine AD est exploité en mode de compatibilité NT, on peut utiliser exactement le même processus.

Les domaines AD offrent aussi la possibilité d'utiliser LDAP pour l'authentification et la recherche de données. C'est un schéma identique à celui proposé pour de plus vastes réseaux de systèmes OSS purs et mérite l'attention. En étendant le schéma AD pour y inclure des données Unix, il serait possible de gérer les utilisateurs des postes et serveurs OSS à l'aide des outils d'administration AD. Le stockage central des données est préférable au schéma *Winbind* utilisé par *MS-Windows NT* car il assure sur l'ensemble des machines la correspondance entre les identifiants *MS-Windows NT* et *Unix*.

14.6.4 Remplacer les PDC/BDC MS-Windows NT avec Samba+LDAP

Samba peut assurer le rôle de contrôleur de domaine primaire, permettant ainsi l'élimination de tous les serveurs *MS-Windows*, même si l'on doit conserver quelques clients *MS-Windows*. Il n'est *pas* possible de remplacer seulement le PDC ou un BDC d'un domaine : tous les contrôleurs de domaines doivent utiliser le même système - *MS-Windows* ou *Samba*. Cela est dû en partie au fait que le protocole de réplication PDC-BDC n'a pas été rétro-développé. Aussi, les contrôleurs de domaine *Samba* ont une approche différente de la résilience : ils délèguent celle-ci aux serveurs LDAP qui stockent effectivement les données.

La configuration d'un contrôleur de domaine *Samba*+LDAP est un travail trop important pour le décrire en détail ici mais cela peut être réalisé en une journée environ par une personne expérimentée. La tâche la plus importante est la planification de la migration des noms d'utilisateurs et de groupes d'un domaine existant. Une partie du travail est décrite dans le Samba-LDAP-HOWTO d'IDEALX (voir les références en section 14.12 ci-dessous). On trouvera au même endroit un ensemble de squelettes d'outils de migration qui peuvent constituer une très bonne base de travail.

Sommairement, le processus est :

1. installer le(s) serveur(s) OSS avec *Samba* et *OpenLDAP*. Il peut être nécessaire de compiler *Samba* depuis les sources (la version fournie dans *RedHat Linux 7.3* par exemple n'inclut pas les fonctionnalités LDAP) ;
2. ajouter les définitions de schéma *Samba* au serveur LDAP ;
3. configurer le serveur LDAP avec une structure d'arbre Distinguished Name (DN) et répertoires appropriée (en utilisant éventuellement les outils d'IDEALX pour alimenter l'arbre avec des entrées automatiques) ;
4. démarrer *Samba* et tester le fonctionnement du contrôleur de domaine ;
5. utiliser **pwdump** sur le PDC pour lister toutes les entrées utilisateurs du SAM et transférer le résultat au format texte sur le serveur OSS ;
6. configurer l'outil IDEALX **smbldap-migrate-accounts.pl** en phase avec l'environnement à créer (le nombre d'options à prendre en compte rend cela non trivial) ;
7. exécuter **smbldap-migrate-accounts.pl** sur les données transférées du PDC afin de créer des entrées dans l'arbre LDAP pour les utilisateurs du domaine, configurer leur mot de passe SMB en face de ceux utilisés sous *MS-Windows NT* (sans pour cela créer des comptes Unix ni GNU/Linux, car les algorithmes de hachage de mots de passes sont différents entre *MS-Windows NT* et les systèmes OSS) - l'outil peut créer les répertoires personnels au passage si on le souhaite ;

8. copier les fichiers utilisateurs et les profils itinérants depuis les serveurs *MS-Windows* ou rapprocher les serveurs existants au domaine désormais servi par des contrôleurs de domaine *Samba*.

Il est vraisemblable que les grands réseaux nécessitent des serveurs LDAP multiples avec de la réplication de données pour avoir une certaine résilience. Si un contrôleur de domaine *Samba* est associé à chaque serveur LDAP, il est possible de réaliser un schéma très proche d'une configuration PDC/BDC classique.

De nombreux autres points sont à considérer, tels que :

1. choix des outils pour la gestion des utilisateurs ;
2. correspondance des ACL *MS-Windows NT* avec les groupes et ACL Unix ;
3. utilisation éventuelle d'un nouveau nom de domaine pour le service fondé sur l'OSS ;
4. création de hachages de mots de passe utilisables par les systèmes OSS (ou maintien des hachages *MS-Windows NT* ou *MS-Lanman*, même dans un environnement pur OSS).

14.6.5 Remplacer un Active Directory par LDAP

La base des données stockées par un Active Directory est accessible par LDAP. À première vue, cela devrait rendre aisé le remplacement de serveurs AD par des équivalents OSS. Malheureusement, ce n'est pas le cas : les systèmes *MS-Windows 2000* n'utilisent pas un LDAP pur pour l'accès aux données ainsi qu'une variante non standard de Kerberos pour l'authentification.

Diverses équipes OSS travaillent à la résolution de ce problème mais à la date de rédaction, le seul moyen d'intégrer des clients *MS-Windows 2000* ou *MS-Windows XP* est de les exploiter dans des domaines *MS-Windows NT* comme expliqué ci-dessus.

14.6.6 Activer l'infrastructure GNU/Linux en parallèle et migrer les utilisateurs par groupes

14.6.6.1 Remplacer tous les clients MS-Windows par GNU/Linux

C'est le plus simple de tous les schémas possibles de migration. L'interaction entre *MS-Windows* et les systèmes OSS est limitée à un transfert unique des fichiers utilisateurs. En gros, le processus est :

1. mise en place de l'environnement OSS de base ; cela inclut les serveurs LDAP (stockage des données de configuration et nom d'utilisateurs), les serveurs maîtres d'installation, un ou plusieurs serveurs de fichiers et d'impression et suffisamment de stations clientes pour l'équipe d'administration système ;
2. mise en place de l'environnement de formation avec suffisamment de stations pour permettre d'accueillir des groupes de taille appropriée ; le travail initial de cet environnement est la validation et la configuration fine des stations avant le déploiement principal.
   À cet instant, le processus de construction de stations de travail doit être finalisé de manière à ce que les machines puissent être installées avec un effort humain minimal ; il est très important que tous les postes soient installés de manière *exactement* identique durant la phase de déploiement principal et celui-ci doit donc être testé prudemment ;
3. utilisation de l'environnement de développement et de formation à l'aide d'un échantillon représentatif de la base d'utilisateurs pour engendrer l'enthousiasme pour le projet et rassembler les impressions sur l'interface utilisateur ; réalisation des modifications nécessaires pour obtenir l'« image de déploiement ».
   Validation des besoins de formation et planification ;
4. installation d'un ensemble de nouveaux postes suffisant pour remplacer l'équipement utilisé par le premier groupe devant migrer vers le système OSS ;
5. enregistrement du premier groupe d'utilisateurs dans le nouveau système ;
6. formation du premier groupe d'utilisateurs au nouveau système ;

7. si nécessaire, reconfiguration des environnements modifiés durant la formation afin que chacun démarre avec un environnement connu ;
8. remplacement des postes du premier groupe avec les systèmes OSS pré-configurés.
   En parallèle, copie des fichiers du groupe vers les nouveaux serveurs de fichiers et protection en lecture seule des originaux ;
9. fourniture d'un support actif au premier groupe pendant leur période d'acclimatation ;
10. mise à jour des anciens postes du premier groupe si nécessaire et installation de l'image standard de poste ;
11. répétition du processus avec le groupe suivant ;
12. une fois la migration de tous les utilisateurs réalisée, réalisation de copies d'archives de tous les fichiers des anciens serveurs et déconnexion de ceux-ci.

14.6.6.2 Conserver quelques clients MS-Windows

Lorsqu'il faut conserver quelques clients *MS-Windows* (par exemple pour supporter des fonctions dont la migration n'est pas économiquement souhaitable), deux options principales se présentent :

- conservation d'un petit domaine *MS-Windows* à l'aide d'un ou plusieurs serveur(s) *MS-Windows* ;
- support des clients *MS-Windows* par les serveurs OSS à l'aide de *Samba*.

Le chemin choisi dépend de la manière dont les clients sont conservés ainsi que de leur distribution géographique.

Dans les deux cas, *Samba* sera vraisemblablement nécessaire sur un ou plusieurs nouveau(x) serveur (s) pour assurer le partage de fichiers entre les clients *MS-Windows* et OSS.

14.7 Migrer les applications serveur

14.7.1 Serveurs web : passer de MS-IIS à Apache

Le serveur web *MS-Windows* est *MS-IIS* (Internet Information Server) qui fournit les services HTTP, FTP et Gopher en un seul paquetage. *MS-IIS* a une réputation de problèmes de sécurité et de stabilité qui a conduit de nombreuses organisations à le remplacer par une alternative. En fait, après qu'une série de failles particulièrement sérieuses aient été exploitées en 2001, les analystes de Gartner émirent une recommandation fermement énoncée à leurs clients de non utilisation de *MS-IIS* pour des fonctions critiques avant une réécriture complète par Microsoft.

Il existe un vaste choix de serveurs web pour le remplacement de *MS-IIS*. Beaucoup d'entre eux sont OSS ou sous une licence très libérale. Quelques-uns des serveurs les plus utilisés sont présentés en section 11.4.2 ci-dessus.

Lors de la migration de sites depuis *MS-IIS*, le choix habituel est *Apache* - souvent agrémentés des modules de script PHP ou Perl. *Apache* fonctionne sous GNU/Linux, *FreeBSD*, à peu près toutes les variantes Unix et aussi sous *MS-Windows*. Cela permet un vaste choix d'options de migration.

14.7.1.1 Points particuliers de migration

1 Noms de fichiers et URL

Lors de la migration d'un site web simple depuis *MS-IIS* sous *MS-Windows* vers *Apache* sous GNU/Linux ou Unix, le point principal réside dans le fait que le système de fichiers *MS-Windows* ignore la casse des noms de fichiers, contrairement à la plupart des systèmes de fichiers GNU/linux ou Unix. La hiérarchie des pages web étant normalement représentée directement dans le système de fichiers, cela implique que les URL deviennent sensibles à la casse lors de la migration vers l'environnement Unix ou GNU/Linux (ce problème ne se pose pas si *Apache* est

implanté sur un serveur *MS-Windows*).

Un point moins habituel est qu'il semble que *MS-IIS* accepte indifféremment « / » et « \ » comme séparateur de composants (il doit traduire les « / » en « \ » pour le système de fichiers *MS-Windows* mais l'utilisation de « \ » semble fonctionner nativement) ; ainsi, un fichier peut avoir indifféremment les URL **repertoire\fichier.html** et **repertoire/fichier.html**.

Aucun de ces points n'affecte un site web correctement réalisé et autonome. Malheureusement, les sites bâtis à l'aide de logiciels *MS-Windows* présentent souvent une utilisation variable des capitales et minuscules et incluent parfois le « \ » dans les URL lorsque la structure de fichiers du site contient un sous-répertoire. En fait, le site exemple distribué avec les premières versions de *MS-IIS* présente ces deux anomalies. Il existe dans *Apache* des contournements simples pour les deux problèmes, ce qui est démontré dans l'exemple plus bas dans ce chapitre. En règle générale cependant, il est mieux de corriger ceux-ci dans les données du site web.

2 Cartes d'images côté serveur

Certains sites web anciens utilisaient une cartographie côté serveur en coordonnées x,y dans les images vers les URL cibles. Cette méthode est maintenant obsolète en raison de son inefficacité et de son mauvais fonctionnement avec les navigateurs en mode texte, mais certains sites peuvent continuer à l'utiliser. Dans *MS-IIS*, ces cartographies ont la forme de fichiers d'extension « ***.map** » et leur format n'est pas compatible avec les fichiers *Apache* équivalents.

La meilleure approche réside dans la conversion des cartes côté serveur en cartes côté clients et cela apporte une amélioration pour l'utilisateur. Si cela n'est pas possible, un simple script Perl permet de traduire les fichiers dans un format utilisable par *Apache*.

3 Scripts et connexions SGBD

Les sites complexes ont souvent des pages dynamiques fondées sur des scripts et des accès SGBD. De nombreux sites *MS-IIS* utilisent *MS-ASP* (Active Server Pages) comme cadre de scripts et peuvent utiliser *MS-Access* ou *MS-SQL Server* comme SGBD, en fonction de la taille de l'application.

Il existe de nombreuses manières d'assurer la migration des scripts *MS-ASP*, dont certaines des plus courantes sont :

- le paquetage *Chili!Soft ASP* pour Unix (désormais appelé *Sun ONE Active Server Pages*) ;
- *ASP2PHP* ;
- le module *Apache::ASP* ;
- la conversion manuelle vers un autre langage.

Chili!Soft ASP est un produit propriétaire mais peut dans certains cas constituer un chemin de migration très économique.

ASP2PHP est un convertisseur de scripts autonome qui convertit les fichiers texte écrits en MS-ASP et MS-VBScript en fichiers texte PHP. Le support des scripts MS-ASP par JScript est en cours de développement. PHP est un cadre de développement de scripts très courant avec des ressemblances avec MS-ASP, ce qui simplifie la transition pour les développeurs Pour des projets plus vastes, il est souvent meilleur de mieux séparer la conception de pages et la logique de scripts que ce qui est possible avec les modèles MS-ASP ou PHP. Dans ces cas, une conversion manuelle à l'aide d'un système de modèles peut être un meilleur choix.

Apache::ASP fournit des fonctionnalités similaires à celles de MS-ASP dans le cadre Apache ainsi que les scripts en Perl. MS-VBScript et JScript ne sont pas supportés.

Dans certains cas, une conversion manuelle de *MS-ASP* vers un autre cadre peut être une meilleure approche. Cela apporte une plus grande souplesse et les sites complexes peuvent largement bénéficier d'une migration vers un système de modèles tel que *Template Toolkit* (http://www.tt2.org/).

Tous les systèmes de scripts *Apache* permettent l'accès à une vaste palette de bases de données (SQL, fichiers plats ou indexés, LDAP, NIS, etc.) ; il est ainsi possible de bâtir des sites dynamiques à données de complexité quelconque.

4 Extensions MS-FrontPage

Le paquetage de conception web *MS-FrontPage* introduisit un ensemble d'extensions permettant la gestion à distance du contenu web. Celles-ci ont depuis été utilisées par d'autres paquetages.

Les extensions *MS-FrontPage* sont disponibles pour les systèmes Unix mais ne sont pas universellement appréciées par les administrateurs *Apache* pour diverses raisons dont des problèmes de sécurité et le grand nombre de modifications à apporter à la zone de stockage des pages web.

Une alternative conforme aux standards est maintenant disponible sous la forme du protocole WebDAV (RFC2518). Celui-ci est supporté par beaucoup de serveurs web (y compris *Apache* avec le module **mod_dav**) et le protocole préféré d'administration de sites web. Microsoft supporte WebDAV depuis *MS-Office 2000* et permet aussi l'accès direct depuis *MS-Windows Explorer*, ainsi un serveur GNU/Linux ou Unix *Apache* peut accepter des clients OSS et propriétaires avec le même mécanisme.

14.7.1.2 Migrer un site web statique

Cet exemple présente le processus complet de migration d'un sit web statique simple depuis *MS-IIS* sur *MS-Windows NT* vers *Apache* sur GNU/Linux.

1. préparation du serveur GNU/Linux, connexion au réseau et test d'*Apache*. Celui-ci est fourni pré-configuré dans de nombreuses distributions, ainsi cette étape est normalement sans embûche. Un serveur visible depuis Internet nécessite indubitablement un renforcement de sa sécurité avant sa connexion ;
2. localisation des données du site web sur le serveur *MS-IIS* (habituellement dans **C:\InetPub**) et copie de celles-ci dans un format de transfert (par exemple dans une archive Zip) ;
3. copie (par FTP par exemple) et décompression de l'archive sur la machine GNU/Linux à l'emplacement choisi (ce paramètre est appelé **DocumentRoot** dans le fichier **httpd.conf** d'*Apache* et se trouve habituellement quelque part vers **/var/www/html**) ;
4. ajout de **default.htm** dans la clause **DirectoryIndex** dans **httpd.conf** (par convention, la page par défaut recherchée par *Apache* est nommée **index.html** alors que celle recherchée par *MS-IIS* est **default.htm** - cette action permet l'utilisation indifférente des deux noms) ;
5. à ce stade, le site doit commencer à fonctionner, sauf qu'il faut y accéder par le nom du nouveau serveur au lieu de son URL propre ; des problèmes peuvent survenir si le site utilise indifféremment les capitales et les minuscules ou s'il utilise « \ » dans les URL ;
6. test du site et si possible correction des problèmes par modification des données du site ; cela déterminera les meilleurs performances (il existe des outils de vérification automatiques qui traversent le site pour trouver les liens pointant vers des pages inaccessibles) ; on peut aussi dresser une liste des pages inaccessibles et passer toutes les pages à un vérificateur HTML ;
7. si la correction des données du site n'est pas possible, on peut ajouter ce qui suit à **httpd.**conf (note : cela entraîne un parcours du répertoire et une redirection HTTP pour chaque élément mal épelé ou mal capitalisé d'un URL, donc a des répercussions sur les performances) :

```
LoadModule speling_module modules/mod_speling.so
AddModule mod_speling.c
```

```
CheckSpelling on
```

8. les pages utilisant « \ » dans les URL peuvent être traitées par **mod_rewrite** en ajoutant ce qui suit dans **httpd.**conf (cela remplace le premier « \ » par « / » dans l'URL puis recommence s'il existe plus d'un « \ ») :
```
RewriteEngine on
RewriteRule ^(.*)\\(.*)$ $1/$2 [N]
```
9. recherche des cartes côté serveur par une commande de la forme (s'il n'en existe qu'une ou deux, la modification peut être manuelle, sinon un script peut automatiser le processus) :
```
find /var/www/html -name '*.map' -print
```
10. à ce stade, l'ensemble du site doit fonctionner correctement ; on peut souhaiter activer FTP, *Samba* ou WebDAV pour permettre l'accès en mise à jour des pages ;
11. pour mettre le site en production, on peut soit déconnecter l'ancien serveur et changer l'adresse IP de la machine qui le remplace, soit changer l'entrée DNS du site pour la faire pointer vers le nouveau serveur.

14.7.1.3 Une configuration simple avec WebDAV

WebDAV peut être utilisé pour administrer tout ou partie d'un site web. Dans cet exemple, il est utilisé pour l'ensemble du site, donc aucun autre accès ne doit être autorisé (d'autres systèmes d'administration - tels que FTP ou l'accès direct aux fichiers - gênent le fonctionnement des clients WebDAV dont le système de verrouillage est différent).

1. création d'un répertoire pour les verrous WebDAV ; celui-ci doit appartenir à l'utilisateur et au groupe d'exécution d'*Apache* (voir les options **User** et **Group** dans **httpd.conf**) - **/var/httpd/webdavlocks** est un bon choix ;
2. ajout des lignes suivantes dans la partie principale de **httpd.conf** :
```
Loadmodule dav_module libexec/libdav.so
Addmodule mod_dav.c
DAVLockDB /var/httpd/webdavlocks
```
3. ajout des lignes suivantes dans la partie **Directory** ou **Location** correspondant au site par défaut :
```
DAV On
AllowOverride None
Options Indexes
AuthType Basic
AuthName "Administrateurs du site web seulement"
AuthUserFile /var/httpd/htpasswd
<LimitExcept GET HEAD OPTIONS>
  require valid-user
</LimitExcept>
```
4. contrôle de propriété des fichiers et répertoires associés par l'utilisateur et le groupe d'exécution d'*Apache* à l'aide d'une commande du genre :
```
chown -R apache:apache /var/www/html
```
5. création du fichier de mots de passe :
```
touch /var/httpd/htpasswd
chown root:apache /var/httpd/htpasswd
chmod 640 /var/httpd/htpasswd
```
6. création du mot de passe pour un utilisateur :
```
htpasswd -m /var/httpd/htpasswd admin_web
```
7. redémarrage d'*Apache* ou rechargement de sa configuration - par exemple :
```
/etc/init.d/httpd reload
```
8. il est désormais possible d'administrer l'ensemble du site à l'aide du protocole WebDAV. *MS-Windows 2000* et les clients plus récents peuvent y accéder par les « Favoris réseau » dans *MS-Windows Explorer* et les applications *MS-Office* peuvent sauvegarder directement les données dans le site ; GNU/Linux fournit des fonctions similaires par **davfs** ;
9. le schéma décrit ici ne fournit qu'une sécurité limitée. La lecture des détails sur l'authentification des utilisateurs du manuel *Apache* est nécessaire pour le choix d'un schéma approprié à ses besoins. Il

peut être nécessaire d'utiliser SSL (à l'aide du module **mod_ssl** d'*Apache*) pour sécuriser les transactions.

14.7.2 SGBD : passer de MS-Access ou MS-SQL Server à MySQL ou PostgreSQL

De nombreux petits projets sous *MS-Windows* utilisent *MS-Access*. Ce produit est attirant pour beaucoup car il est simple d'y débuter et que son interface est familière : il a cependant de sévères limitations : il n'a pas été conçu pour une forte charge multi-utilisateurs et ne peut gérer de grands ensembles de données.

Des bases de données plus importantes peuvent utiliser *MS-SQL Server* ou un SGBD connu : *Oracle*, *Sybase*, *IBM-DB2*, etc. Dans le cas de ces systèmes plus importants, il peut être meilleur de conserver le SGBD sur sa plate-forme existante et de ne migrer que les applications clientes. Cela est particulièrement approprié s'il existe des compétences pointues au sein de l'administration et que celle-ci utilise de nombreuses fonctionnalités propriétaires. Il existe des moyens de connexion standard aux SGBD/R par le réseau et donc le choix de la plate-forme peut différer entre la base et les applications clientes. De plus, de nombreuses bases propriétaires non Microsoft sont disponibles sur plates-formes GNU/Linux et Unix ; il est donc possible de changer de système d'exploitations sans avoir à apprendre intégralement un nouveau SGBD.

En revanche, les SGBD propriétaires peuvent être très coûteux et justifient de se demander si des produits OSS pourraient répondre correctement aux besoins.

Les deux SGBD/R OSS les plus connues sont *MySQL* et *PostgreSQL*. Les deux produits sont matures avec des bases installées imposantes et des équipes de développement actives. Les deux ont une bonne conformité au standard SQL et assurent de très bonnes performances.

Il est important de rappeler qu'il n'est pas impératif qu'une base soit relationnelle. Certaines tâches s'intègrent mieux dans d'autres modèles et l'utilisation directe d'un produit OSS tel que *Berkeley DB* de Sleepycat peut être extrêmement efficace. De même, le modèle LDAP de bases réseau hiérarchiques est très adapté à certains types d'applications distribuées.

14.7.2.1 Migration de bases MS-Access

MS-Access n'est disponible que sur plate-forme *MS-Windows*, donc toutes les bases de ce genre doivent être portées si un environnement tout-OSS est prévu. Un scénario intermédiaire intéressant consiste à migrer les données vers une base OSS tout en conservant *MS-Access* comme frontal ; cela a la propriété de supprimer de nombreux restrictions et problèmes des stockages *MS-Access*.

1 Import/export manuel

Il existe différentes manières de transférer les données depuis *MS-Access*. Pour les ensembles simples, la méthode la plus aisée est peut-être d'exporter les tables au format CSV (Comma Separated Values) puis d'importer celles-ci dans le nouveau serveur. Cette méthode nécessite la création manuelle préalable des tables sur le serveur mais ne nécessite aucun logiciel spécifique.

Par exemple, voici les commandes de création d'une base avec une seule table et import d'un fichier CSV dans *MySQL* ; il faut lancer l'interpréteur depuis l'invite par :

```
mysql --user=myusername -p
```

puis saisir ce qui suit :

```
create database mydb;
use mydb;
create table mytable (
firstname char(30),
surname char(30),
postcode char(10)
```

```
);
load data local infile 'exportfile.csv'
into table mytable
fields terminated by ',' enclosed by '"'
lines terminated by '\r\n';
```

2 Import/export par scripts

Différents scripts et programmes permettent l'export intégral d'une base *MS-Access* avec toutes les informations nécessaires à la re-création de celle-ci dans un autre SGBD. Certains produisent des fichier à copier sur la nouvelle plate-forme tandis que d'autres se connectent directement par le réseau et réalisent les modifications immédiatement. Un exemple de script à fichier est **exportsql2.txt** disponible à http://www.cynergi.net/exportsql. Il produit des fichiers contenant les ordres DROP TABLE, CREATE TABLE et INSERT qui permettent la réplications de la base *MS-Access* dans *MySQL*.

D'autres outils de migration sont décrits dans l'article de Paul Dubois *Migrating from Microsoft Access to MySQL* (http://www.kitebird.com/articles/access-migrate.html).

Une fois les données transférées, il est possible de continuer à utiliser *MS-Access* en frontal vers les tables locales supprimées par des liens vers la nouvelle base sur le serveur *MySQL*.

14.7.2.2 Migration de bases SQL Server

Le processus est très similaire à celui décrit ci-dessus ; l'export des données dans un format commun (habituellement CSV) est souvent suffisant pour les bases simples. Des bases plus complexes incluant procédures stockées et déclencheurs (triggers) nécessitent plus d'efforts et la gamme d'outils disponibles (certains sont OSS et d'autres, commerciaux) mérite un coup d'oeil pour aider au processus de migration. Par exemple :

- PGAdmin est libre pour l'administration de bases PostgreSQL et dispose d'utilitaires enfichables pour la migration depuis d'autres moteurs ; plus d'information est disponible à http://www.pgadmin.org/ ;
- SQLPorter de Realsoftstudio est un produit commercial disponible en différentes variantes selon les moteurs source et cible - voir : http://www.realsoftstudio.com/overview.php ;
- SQLWays d'Inspirer est un autre outil commercial qui assure l'administration sur MySQL ainsi que la migration de données depuis les bases compatibles ODBC - voir : http://www.webyog.com/sqlyog ;
- le site *MySQL* liste une vaste ligne d'autres outils de conversion : http://www.mysql.com/portal/software/convertors/index.html.

14.7.2.3 Points particuliers pour la migration de bases de données

La migration des données est parfois la partie la plus aisée du travail, car si l'on accède aux données sous la forme de tables SQL directes, il ne reste pas grand-chose à faire.

Les problèmes peuvent surtout survenir des outils annexes et langages de scripts qui entourent toute base réelle. SQL lui-même est standard, quoique quasiment tous les éditeurs de SGBD l'étendent et encouragent à l'utilisation de leurs extensions non standard. Il existe aussi souvent différentes manières d'obtenir un résultat donné en SQL et le choix de la plus efficace peut varier entre les moteurs.

De nombreuses applications sont engendrées par des générateurs ou des éditeurs de formulaires. Ceux-ci peuvent ne pas fonctionner avec d'autres SGBD que celui avec lequel ils sont vendus.

MySQL et *PostgreSQL* ont énormément évolué ces dernières années et il est donc important de se

fonder sur des articles *récents* avant de choisir s'il faut migrer et vers lequel.

14.7.3 Travail de groupe : abandonner MS-Exchange

MS-Exchange fournit les services de courriel, agenda et carnet d'adresses. Il est en principe utilisé avec le client *MS-Outlook* de *MS-Windows* quoique certains utilisent aussi *MS-Outlook Web Access* (OWA) pour fournir les fonctions de base par une interface web.

Toutes les fonctions de *MS-Exchange* peuvent être remplacées par des paquetages OSS, souvent de manière très efficace. Le problème réside dans la transparence du remplacement pour les clients *MS-Outlook* car le mécanisme de communication *MS-Exchange/MS-Outlook* est propriétaire. *MS-Outlook* est susceptible d'accéder à certains services standard ouverts mais la perception utilisateur est parfois différente de celle fournie avec le protocole propriétaire. En conséquence, il est utile d'étudier l'éventuelle migration vers un paquetage client OSS en parallèle de celle du server, la population utilisatrice percevant dans tous les cas une modification, même en conservant *MS-Outlook*. Le client de remplacement le plus naturel est *Evolution* de Ximian.

14.7.3.1 Points particuliers généraux

Les noms et mots de passe de tous les utilisateurs *MS-Exchange* sont stockés dans le système. Les versions récentes utilisent Active Directory pour cela et dès lors les éléments sur la migration des informations utilisateurs présentes ailleurs dans ce document s'appliquent ici. En gros, les serveurs OSS peuvent accéder aux données par LDAP et on peut, soit utiliser Active Directory depuis les nouveaux serveurs, soit migrer les données vers un entrepôt OSS tel que *OpenLDAP*.

14.7.3.2 Points particuliers pour le courriel

Certains utilisateurs peuvent avoir stocké un volume considérable de courriel, aussi bien personnel que partagé. La conservation de la trace de la totalité du courriel émis et reçu a des implications légales et réglementaires, donc le stockage et l'accès à ces données doivent être étudiés. Les utilisateurs de portables peuvent télécharger tout leur courriel sur celui-ci ou choisir de conserver une copie synchronisée avec un maître conservé sur l'entrepôt central.

Lors de la planification de la migration des services courriel, il est important de localiser l'ensemble des données locales et d'en garantir l'accessibilité post-migration.

MS-Exchange peut utiliser les groupes *MS-Windows* comme listes de distribution - il s'agit des mêmes groupes que ceux que *MS-Windows* utilise pour le contrôle d'accès. Ce n'est pas une méthode habituelle d'administration des listes de diffusion dans un environnement OSS mais celle-ci peut être supportée si nécessaire.

Si *MS-Outlook* est conservé comme client, il sera nécessaire d'en reconfigurer l'accès « natif » aux boîtes en accès IMAP.

MS-Exchange ne dispose d'aucune capacité d'export et la migration des données doit s'effectuer par une connexion cliente.

Pour plus d'informations sur les systèmes de courriel OSS, se reporter à la section 11.2 et à l'annexe C.

14.7.3.3 Points particuliers pour les carnets d'adresses

MS-Outlook bâtit automatiquement un carnet d'adresse personnel au fur et à mesure de la réception et de l'expédition des messages. Les utilisateurs ont éventuellement aussi accès à des carnets d'adresses partagés *MS-Exchange*. Le contenu de ces carnets d'adresses doit être migré vers un format lisible OSS. Les carnets d'adresses personnels peuvent être exportés au format vCard, compris par de nombreux clients de courriel et analysés par des scripts de conversion vers d'autres formats si nécessaire. De même, les carnets d'adresses partagés peuvent être exportés puis chargés dans un entrepôt LDAP.

Les principaux problèmes proviendront sans doute du fait que *MS-Outlook* et *MS-Exchange* tendent à ne pas utiliser en interne le standard RFC822 des adresses courriel, donc les données de carnets d'adresses peuvent ne contenir des adresses inutilisables lors de l'export. Dans ce cas, il sera nécessaire d'effectuer un traitement supplémentaire avec un script accédant à l'entrepôt Active Directory pour traduire les adresses au format RFC822 ; cette traduction pourra être nécessaire même si *MS-Outlook* est conservé comme client courriel car il ne pourra utiliser son format interne pour envoyer du courriel par des protocoles standard tels que SNMP.

14.7.3.4 Points particuliers pour l'agenda

Certaines administrations ont une utilisation considérable des fonctions d'agenda de *MS-outlook* pour organiser des réunions et la réservation des salles. Ces fonctions peuvent être utilisées sans *MS-Exchange* mais avec quelques limitations.

Si une migration parallèle vers des clients OSS est planifiée, les agendas devront être exportés au format vCal puis chargés sur la nouvelle plate-forme de gestion d'agendas.

14.8 Migration de la bureautique vers l'OSS

14.8.1 Office

14.8.1.1 Conversion de documents

OpenOffice.org est capable de lire et écrire remarquablement bien les formats Microsoft ; il n'est donc pas nécessaire de convertir les documents durant le processus de migration. Si une telle conversion est souhaitée, celle-ci peut être automatisée avec la fonction AutoPilote du menu Fichier. Celle-ci fournit un moyen de conversion de masse des documents. La décision de conversion dépend de l'utilisation future du document. Le chapitre 5 aborde en termes généraux les formats de documents et leur conversion. Si ceux-ci sont destinés à des modifications répétées, le format à utiliser est celui de la majorité des éditeurs.

14.8.1.2 Conversion de modèles

OpenOffice.org peut utiliser directement les modèles au format *MS-Word 97* mais il est meilleur en pratique de convertir ceux-ci au format natif et de les stocker dans une zone partagée appropriée. Cela donne l'opportunité de tester chacun et de corriger les erreurs de conversion. *OpenOffice.org* effectue directement l'essentiel du travail de conversion et le processus peut être automatiser pour les grandes quantités de modèles à l'aide de la fonction AutoPilote du menu Fichier.

Les modèles issus d'autres traitements de textes devront sans doute être recréés manuellement.

14.8.1.3 Conversion de macros

OpenOffice.org utilise un langage de macro à la BASIC. Sa structure est très similaire à celle des langages utilisés par *MS-Word* et les dernières versions de *WordPerfect*, cependant les noms d'objets sont différents, ce qui imposera un certain effort de conversion manuel pour chaque macro.

Les macros de documents sont des risques sévères pour la sécurité et ne sont pas nécessaires pour les tâches quotidiennes, il est donc préférable de rechercher les moyens de s'en dispenser. De nombreuses tâches de mise en forme sont mieux réalisées à l'aide des modèles et des styles et la manipulation simple de données peut être réalisée à l'aide de formulaires.

Depuis la version 1.1 d'*OpenOffice.org*, un enregistreur de macros est apparu, simplifiant la création de macros simples si celles-ci sont jugées essentielles.

Il n'existe actuellement aucune méthode automatique pour convertir les macros bien qu'un effort certain soit en cours sur ce point.

14.8.1.4 Traitement de texte

De nombreux traitements de texte existent sous *MS-Windows*. Les organisations bien gérées ont en général homologué un seul paquetage ou peuvent se trouver dans une phase de transition entre deux. Les plus habituels sont :

- *Microsoft Word* ;
- *Microsoft Works* ;
- *WordPerfect* ;
- *Lotus AmiPro* et *Lotus WordPro* ;
- *Lotus Notes* ;
- *IBM DisplayWrite*.

La cible OSS est *OpenOffice.org*.

Les fichiers aux formats *Microsoft Works*, *IBM DisplayWrite* et Lotus ne sont pas directement lisibles par *OpenOffice.org* et nécessitent une conversion. Il est souvent possible d'exporter ces fichiers depuis leur application mère dans un format commun acceptable, sinon un outil de conversion tiers peut être nécessaire.

Les fichiers *WordPerfect* ne sont pas encore directement lisibles mais l'inclusion de ce format dans *OpenOffice.org* est un projet en cours. Un script de conversion est disponible, ce qui peut être utile pour les conversions de masse.

Le site web http://www.raycomm.com/techwhirl/magazine/technical/openofficewriter.html compare de manière très utile les fonctions disponibles sous *MS-Word* et *OpenOffice.org*. L'interface est suffisamment similaire à celle de *MS-Word* pour permettre aux utilisateurs de basculer de l'un vers l'autre sans trop de difficulté (bien qu'une formation appropriés soit toujours souhaitable pour présenter efficacement le nouveau produit).

14.8.1.5 P.A.O.

La production de documents au-delà des capacités des traitements de texte est habituellement réalisée avec des logiciels de P.A.O. (publication assistée par ordinateur). Les plus communs sont :

- *Framemaker* ;
- *Pagemaker* ;
- *QuarkXpress*.

Le produit OSS *Scribus* sur http://web2.altmuehlnet.de/fschmid est censé remplacer ces produits et peut valoir une évaluation.

OpenOffice.org est beaucoup plus puissant que ne l'étaient les traitements de texte lors de l'apparition des premiers logiciels de P.A.O. Des fonctions avancées telles que les documents-maîtres permettent désormais de gérer de vastes projets tels que la production de livres et les fonctions de mise en page sont maintenant très souples.

Des approches différentes incluent l'utilisation de paquetage de post-traitement où le texte est saisi dans un langage à balises similaire à HTML puis converti dans son aspect final imprimable par application de feuilles de style. Ces systèmes sans I.H.M. peuvent être très utiles pour la production de documents à évolution rapide ainsi que pour l'impression à la demande à partir de données de SGBD.

14.8.1.6 Tableurs

Parmi les principaux tableurs sous *MS-Windows* on trouve :

- *Microsoft Excel* ;
- *Lotus 123* et ses dérivés.

MS-Excel est de loin le plus répandu.

La cible OSS est *OpenOffice.org* bien que *Gnumeric* soit aussi à considérer.

Dans de nombreux cas, une migration de *MS-Excel* ou *Lotus 123* vers *OpenOffice.org* ou *gnumeric* ne doit poser que peu de problèmes sauf si la feuille de calcul contient des contrôles ou autres mécanismes nécessitant des macros. Dan ce cas, ces contrôles et macros doivent être réécrits.

14.8.1.7 Présentation

Dans un environnement *MS-Windows*, les présentations sont habituellement créées avec *MS-PowerPoint* ou *Corel Draw*. *MS-PowerPoint* avec son format ***.ppt** est le plus courant.

La cible OSS est *OpenOffice.org*. Il peut relire les présentations *MS-PowerPoint* avec très peu d'erreurs et peut être configuré pour enregistrer au format ***.ppt** si on le souhaite. Comme indiqué à la section 14.8.1.2 ci-dessus, il peut être intéressant de transférer en masse les modèles importants vers le format natif *OpenOffice.org*.

Les utilisateurs devraient pouvoir basculer entre *MS-PowerPoint* et *OpenOffice.org* assez aisément car les concepts et l'aspect de l'écran sont très similaires.

14.8.1.8 Graphiques et manipulation d'images

Les paquetages graphiques se divisent en trois catégories principales :

- les logiciels de présentation, abordés à la section 14.8.1.7 ci-dessus ;
- les logiciels de dessin vectoriel, typiquement des logiciels de CAO/DAO d'entrée de gamme et paquetages de type *Microsoft Visio* ;
- les logiciels de dessin « bitmap », notamment les programmes de type *MS-Paint* et ceux de retouche photo comme *Adobe Photoshop*.

1 Dessin vectoriel

OpenOffice.org inclut un outil de dessin vectoriel.

Dia (http://www.lysator.liu.se/~alla/dia) est un paquetage OSS très similaire à MS-Visio. Il est très utilisé pour la génération de diagrammes documentaires et ses filtres lisent les fichiers des anciennes versions de MS-Vision (pas ceux de la version 2002). Il contient une bibliothèque de symboles pour de nombreuses applications. Kivio effectue un travail similaire est il est conçu pour s'intégrer harmonieusement dans l'environnement KDE mais semble plus axé sur les organigrammes. Sodipodi (http://sodipodi.sourceforge.net/) fonctionne bien avec les SVG (Scalable Vector Graphics).

Les fichiers originaires de *MS-Visio* et similaires peuvent être lus par les logiciels OSS mais cela doit être testé dans chaque cas particulier avant de planifier une migration.

2 Dessin bitmap

Cette catégorie va de programmes simples comme *MS-Paint* jusqu'à des logiciels avancés de retouche comme *Adobe Photoshop*. Le monde OSS a produit au moins autant de programmes

graphiques que le secteur propriétaire avec des variations similaires en fonctionnalités et qualité. Un paquetage cependant émerge, c'est *The Gimp* (http://www.gimp.org/).

The Gimp est susceptible de lire à peu près tous les formats graphiques bitmap connus (y compris le format interne de Adobe Photoshop) et peut en écrire la plupart. Il fournit toutes les fonctionnalités d'un bon programme de dessin avec les couches, les canaux et autres outils avancés familiers des utilisateurs de Photoshop. The Gimp est largement utilisé pour la génération et l'amélioration d'images pour le web et la publication. La seule fonctionnalité majeure encore manquante est le traitement complet de la gestion des couleurs et il peut n'être pas adapté pour le travail de pré-presse de très haute qualité.

14.8.1.9 Génération PDF

Engendrer un fichier PDF sous OSS est beaucoup plus simple que sous *MS-Windows* dans lequel il faut acquérir quelque chose comme *Adobe Acrobat*. Il existe de nombreux outils PostScript et PDF disponible dans les distributions standard pour cela. De plus, *OpenOffice.org* fournit un moyen de produire directement une sortie PDF. La fonction de production de PDF peut être configurée comme un service d'impression, ouvrant ainsi cette méthode aux utilisateurs restant sous *MS-Windows*.

14.8.2 Courriel

Il existe une gamme énorme d'interfaces utilisateur pour le courriel, aussi bien pour les environnements propriétaires qu'OSS. En conséquence, ce document ne peut fournir qu'un schéma général du processus de migration et des points particuliers. Le courriel est aussi abordé en section 11.2 et en annexe C.

Les principaux points côté client sont :

- choix du nouvel agent utilisateur de courriel et donc de son interface utilisateur ;
- migration du courriel existant stocké ;
- migration des entrées de carnets d'adresses existants.

Quel que soit l'agent de courriel choisi, il sera nécessaire de migrer le courriel stocké et les entrées de carnets d'adresses. Si l'ancien agent est configuré pour stocker tous ses dossiers sur un serveur IMAP, très peu de travail est nécessaire et le nouvel agent peut être simplement configuré pour y accéder. Lorsque des fichiers locaux sont utilisés comme dossiers, il sera nécessaire de les identifier puis de les convertir. Par défaut, *MS-Outlook* stocke le courriel dans des fichiers dont l'extension est ***.pst** dans **C:\Documents and Settings**\<utilisateur>**\Local Settings\Application Data\Microsoft\Outlook**.

Parmi les outils de migration utiles :

- http://outport.sourceforge.net/ - exporte les données *MS-Outlook* vers *Evolution* ou autre ; il est en développement actif et, à mi-2003, conserve des limitations importantes dont l'incapacité à exporter les pièces jointes ;
- l'outil d'export de *MS-Outlook* lui-même, capable d'écrire aux formats CSV ou *MS-Excel* ; il souffre aussi de l'impossibilité d'exporter les pièces jointes vers ces formats ;
- http://sourceforge.net/projects/ol2mbox - un outils OSS de conversion des fichiers ***.pst** en formats utilisables par les agents OSS ; il transfère les pièces jointes ;
- *Kmailcvt* - un outils OSS pour convertir quelques formats propriétaires pour une utilisation avec *Kmail*.

14.8.3 Agendas et travail de groupe

Les agendas ainsi que la gestion de contacts et le courriel sont souvent groupés sous le terme

générique d'assistants personnels (PIM - Personal Information Management). Certains paquetages intégrés tels que *MS-Outlook* fournissent les trois fonctions sous une seule interface tandis que d'autres tels que *Act!* privilégient la gestion de contacts et doivent être considérés comme plus proches des systèmes de CRM (Customer Relationship Management).

La cible OSS est *Evolution* qui intègre les fonctions de manière similaire à celle de *MS-Outlook*. *Mozilla* peut aussi être candidat : il inclut un client courriel capable et un module agenda est désormais disponible à http://www.mozilla.org/projects/calendar - fondé sur le standard iCalendar et permettant la publication et le partage d'agendas par les utilisateurs selon le protocole WebDAV.

14.8.3.1 Agendas

Certaines des fonctions les mieux réalisées des agendas OSS se trouvent dans les suites de travail de groupe web et celles-ci peuvent être considérées comme des candidats pour un service global à l'organisation.

Le standard iCalendar (précédemment vCalendar) définit un format d'échange pour les entrées d'agendas (des détails peuvent être trouvés sur http://www.imc.org/pdi ainsi que dans les RFC 2445, 2446 et 2447). De nombreuses suites d'agendas comprennent ce format qui est donc le chemin de prédilection pour la migration ainsi que la méthode normale pour l'administration quotidienne.

Certains des outils de migration mentionnés au 14.8.2 ci-dessus peuvent aussi extraire les données d'agenda depuis les fichiers *MS-Outlook* au format iCalendar.

14.8.3.2 Gestion de contacts

Quasiment chaque paquetage de courriel ayant existé a défini son propre format pour les stockage des données de carnet d'adresses. De nombreux se bornent au simple stockage d'adresses courriel mais les formats récents tendent à inclure toutes sortes d'informations sur les contacts. La diversité des formats rend la migration plus difficile qu'elle ne devrait l'être.

Heureusement, les applications les plus populaires, aussi bien propriétaires qu'OSS, ont tendu ces dernières années à implanter le format d'échanges iCard (précédemment vCard). La spécification de ce format est ouverte et peut être obtenue sur http://www.imc.org/pdi ainsi que dans les RFC 2425 et 2426. Lorsque les contacts doivent être transférés d'une application propriétaire vers une application OSS, c'est le format à privilégier.

Une autre manière d'administrer les informations de contacts consiste à les consolider dans un annuaire général de l'organisation auquel on accède par LDAP. C'est certainement ce qu'il faut faire pour les données largement utilisées telles que l'annuaire interne et les listes de diffusion fréquents dans de nombreuses organisations. Cela ne peut cependant remplacer totalement les carnets d'adresses personnels : un carnet d'adresses doit être petit et ciblé sur les besoins de la personne qui l'utilise, tandis qu'un annuaire doit être compréhensible et (probablement) trop important pour y naviguer avec efficacité.

Certains des outils de migration mentionnés au 14.8.2 ci-dessus peuvent aussi extraire les carnets d'adresses depuis les fichiers *MS-Outlook* au format iCard.

14.8.4 Navigation web

Les utilisateurs de *MS-Windows* ont des chances d'utiliser une version quelconque de *Microsoft Internet Explorer* pour le web. Il est aussi possible de trouver *Netscape*, *Mozilla* ou *Opera*. La cible OSS est *Galeon*, quoique *Mozilla* reste un bon candidat car il fonctionne sous *MS-Windows*.

La migration d'un navigateur vers un autre est assez facile pour les utilisateurs car tous ont des fonctionnalités et I.H.M. similaires (sauf les navigateurs en mode texte tels que *Lynx*, pour des raisons évidentes). Ce qui concernera les utilisateurs sera centré autour de la conversion des favoris ; de nombreux navigateurs OSS peuvent importer ceux de *MS-IE* et de *Netscape* s'ils sont installés sur la même plate-forme, mais si le système d'exploitation est différent, il peut être nécessaire d'exporter préalablement les

favoris au format HTML.

Toute organisation utilisant un intranet doit vérifier que le HTML soit conforme aux standards du W3C afin qu'ils s'affiche correctement sur tous les navigateurs. Les outils disponibles sur http://www.w3c.org/ sont d'une aide certaine.

Toutes les pages dépendant de JavaScript nécessiteront un test particulièrement approfondi car ce dialecte varie selon le navigateur et l'utilisation d'extensions non standard pose des problèmes.

Toutes les pages dépendant de contrôle ActiveX devront être re-développées pour fonctionner différemment car les navigateurs OSS ne supportent pas cette technologie propriétaire. ActiveX a un modèle de sécurité très pauvre et sa dés-activation est toujours une amélioration.

Les formats habituels imbriqués tels que Java, PDF, Flash et RealPlayer sont bien supportés par les navigateurs OSS (souvent avec un composant, non OSS toutefois, qui peut être difficile à trouver sur le site de l'éditeur) ; d'autres formats tels que *Shockwave Director* nécessiteront *CrossOver Plugin* de CodeWeavers.

14.8.5 Bases de données personnelles

Les utilisateurs de données trop volumineuses out trop complexes pour un tableur mais pas suffisamment pour justifier un SGBD commercial complet utilisent souvent *Microsoft Access*. Ce paquetage fournit un stockage de données relationnel simple ainsi que des outils de scripts et d'édition de formulaires.

La migration de données vers les SGBD OSS est couverte à la section 14.7.2.

Il y a des avantages à stocker les bases de données sur des serveurs bien administrés même si la fonction I.T. centrale n'est pas d'administrer les données ni d'assurer le support des applications. Une migration OSS fournit l'opportunité d'offrir un tel service de stockage de données en implantant un serveur sur lequel les utilisateurs puissent bâtir leurs propres applications. Plusieurs paquetages web peuvent être choisis comme base, tels que *PHPmyAdmin* (http://www.phpmyadmin.net/documentation).

Les outils à base d'I.H.M. plus traditionnelle incluent :

- *Kexi* (http://www.koffice.org/kexi) - un frontal de SGBD du projet KDE, ciblé vers un marché similaire à celui de *MS-Axxess* ;
- *DBDesigner* (http://www.fabforce.net/dbdesigner4) - un outil pour utilisateurs plus avancés qui s'intègre avec Gnome et KDE ;
- *Knoda* (http://www.knoda.org/) - un autre frontal simple pour KDE.

Aucun de ces outils ne permet l'accès aux fichiers *MS-Access*.

14.9 Migration des services d'impression vers l'OSS

Dans les petits environnements de bureau, il est courant que les imprimantes soient directement branchées sur les postes de travail. Dans des organisations plus importantes ainsi que dans celles qui réalisent des travaux d'impression de grand volume, on trouve plus souvent des imprimantes réseau - celles-ci peuvent être connectées directement au réseau ou pilotées par un serveur d'imprimante.

Les environnements OSS permettent les deux approches quoiqu'il soit plus courant de trouver des serveurs d'impression avec un petit nombre d'imprimantes de grande capacité.

14.9.1 Le modèle d'impression MS-Windows

L'impression sous *MS-Windows* est quasiment toujours réalisée depuis une option de menu d'une application ; l'utilisation de la ligne de commandes est très rare. Les applications engendrent une sortie

imprimée selon un processus très proche de celui utilisé pour l'affichage. Un ensemble de pilotes spécifiques sont utilisés par le système d'exploitation pour envoyer le flux de données vers l'imprimante. Ces pilotes sont en général fournis par le constructeur de l'imprimante et doivent être installés localement ou sur le serveur d'impression avant toute impression. Dans un environnement réseau, il est en principe meilleur d'installer et configurer les pilotes sur le serveur d'impression afin d'éviter la configuration manuelle des clients (il reste nécessaire de connecter les imprimantes au client, ce qui peut être réalisé à la main ou par un script de connexion).

14.9.2 Le modèle d'impression Unix et GNU/Linux

GNU/Linux utilise un modèle d'impression hérité de l'Unix BSD : les applications engendrent des fichiers ou des flux de données qui sont passés à une file d'impression qui assure le travail d'impression. Les travaux peuvent être mis en file et transmis de manière transparente à d'autres machines du réseau. Les premiers systèmes Unix ne disposaient pas d'interface unifiée pour la génération des données d'impression ce qui imposait à chaque application d'inclure du code pour tout type d'imprimante à piloter. À l'époque de l'impression en mode texte cela n'était pas un problème, mais lorsque les constructeurs ont ajouté des capacités graphiques, ils ont aussi créé chacun un nouveau langage d'impression différent.

Les systèmes d'impression BSD ont toujours eu la capacité de traiter les travaux d'impression par un ensemble de filtres et on a commencé à écrire des filtres de conversion entre différents langages d'impression pour augmenter la gamme des imprimantes supportées. Les meilleures imprimantes utilisées dans les laboratoires de recherche disposant d'interpréteurs PostScript, ce dernier fut choisi comme langage indépendant de l'imprimante.

De nombreux distributeurs GNU/Linux remplacent désormais le système d'impression BSD par un nouveau paquetage appelé *CUPS* (Common Unix Printing System) qui accepte le protocole d'impression Internet (IPP - Internet Printing Protocol) en plus du standard **lpr**. Cela complète la transition vers le nouveau modèle d'impression :

1. les applications engendrent les travaux en PostScript ;
2. lorsque les travaux sont passés au système d'impression, l'application peut demander certaines fonctions spéciales de l'imprimante (impression en duplex, pliage, brochage, perforation, assemblage, etc.). Les demandes ont un format standard mais n'ont évidemment d'effet que si l'imprimante dispose du matériel correspondant. Il existe une méthode standard permettant aux applications de connaître les fonctions permises par une imprimante donnée ;
3. les travaux peuvent entrer dans une file locale ou être passés immédiatement à un serveur d'impression ; il est inutile que l'utilisateur connaisse la méthode utilisée ;
4. le serveur d'impression exécute le travail à travers une suite de filtres qui convertissent progressivement celui-ci au format nécessaire à l'imprimante et contrôle la communication avec celle-ci.

Le fonctionnement de plus de 600 modèles d'imprimantes est actuellement validé avec ce modèle (c.-à-d. les applications GNU/Linux peuvent accéder à toutes les fonctions disponibles par les pilotes *MS-Windows* avec des résultats équivalents, voire meilleurs).

Bien que PostScript soit le format intermédiaire le plus répandu, *CUPS* peut être configuré pour supporter quasiment tout format de fichier pour lequel des filtres sont disponibles. En particulier, il est habituel de permettre l'impression directe des PDF, JPEG et quelques autres formats et certains sites ajoutent des filtres d'impression améliorée automatique du courriel, etc.

CUPS une interface compatible avec la suite **lpr** de BSD ainsi qu'avec le **lp** des Unix système V. Il est ainsi possible de remplacer entièrement les anciens systèmes des machines existantes (*FreeBSD*, *OpenBSD* et de nombreuses variantes Unix commerciales). Un portage vers *MS-Windows* est en cours.

CUPS permet toute une gamme de fonctions, y compris l'auto-découverte de serveurs d'impression, comptage de pages, quotas, etc. Se reporter au site web de *CUPS* indiqué plus bas.

14.9.3 Mise en place d'un service d'impression OSS

Pour les tous petits déploiements, la configuration d'imprimantes connectées à chaque poste est simple. Celles-ci peuvent être partagées sur le réseau et *CUPS* le permet de manière très aisée.

L'utilisation de serveurs d'impression est recommandée dès lors qu'on dépasse une poignée de clients ou que le volume d'impression est substantiel. Un ou plusieurs serveurs doivent être installés avec des noms logiques dans le DNS en plus de leurs noms de machines. Cela permet des configurations avec des noms référence tels que **imprimante.exemple.org** plutôt que **pc35.exemple.org**, ce simplifie considérablement les futures réorganisations. Chaque machine cliente doit pointer vers un serveur pour toutes les demandes d'impression : cela évite toute reconfiguration de poste lors de l'ajout ou de la suppression d'imprimantes.

14.9.4 Impression depuis MS-Windows vers des imprimantes servies par GNU/Linux

Il existe plusieurs manières de servir les postes *MS-Windows* depuis les serveurs d'impression GNU/Linux ; elles varient en quantité de travail initial et en quantité de travail par client.

14.9.4.1 Utiliser le protocole lpr

Cette méthode est appropriée pour de très petits nombres de clients *MS-Windows*.

lpr est un protocole très courant pour les machines Unix. Comme mentionné plus haut, celui-ci est progressivement remplacé par IPP mais il est largement implanté et peut être utilisé par de nombreuses versions de *MS-Windows*[2].

1. vérifier que la machine GNU/Linux accepte les travaux par **lpr** ;
2. obtenir les pilotes *MS-Windows* pour l'imprimante (le pilote idéal devrait être celui générique de *CUPS* qui engendre du PostScript portable mais il est possible d'utiliser des pilotes spécifiques si *CUPS* est configuré pour permettre l'impression directe) ;
3. se connecter comme Administrateur de la machine *MS-Windows* ;
4. ouvrir l'utilitaire Réseau du Panneau de configuration, choisir l'onglet Services et s'assurer de la présence de la ligne « Impression Microsoft TCP/IP » ou l'ajouter si nécessaire (cela nécessite le CD de distribution et un redémarrage) ;
5. configurer l'imprimante comme locale (pas réseau !) et lors de la sélection du port, créer un « Nouveau port LPR » et le faire pointer vers le serveur GNU/Linux.

Le client *MS-Windows* est alors en mesure d'imprimer par le serveur, mais ses outils peuvent n'être pas en mesure de voir ni manipuler les travaux en file d'attente. *CUPS* permet l'administration web donc il faut prévenir les utilisateurs d'utiliser cette voie si nécessaire.

14.9.4.2 Utiliser les partages d'imprimantes

Cette méthode est aussi appropriée pour les petits nombres de clients *MS-Windows* ; elle fonctionne aussi bien avec *MS-Windows 95/98/Me* qu'avec *MS-Windows NT/2000*.

1. installer et configurer *Samba* sur le serveur GNU/Linux en suivant les instructions pour créer des partages d'imprimantes : *Samba* peut aisément créer automatiquement un partage pour chaque imprimante dont il a connaissance ;
2. sur chaque client *MS-Windows*, utiliser l'assistant d'ajout d'imprimantes pour ajouter une imprimante réseau ; il devrait être possible de naviguer parmi une liste de serveurs pour trouver celui à utiliser. Les pilotes d'imprimantes devront être installés localement sur chaque client.

Il est possible d'affiner ce schéma en chargeant les pilotes vers le serveur *Samba* (sous le compte Administrateur à l'aide de l'assistant d'ajout d'imprimantes) afin qu'il ne soit plus nécessaire d'installer ceux-ci individuellement sur les clients.

2 NdT : Il semble que MS-Windows Me ne permette pas cette méthode.

14.9.4.3 Utiliser la configuration Pointer - imprimer

Cette méthode est appropriée pour des installations plus vastes ainsi que celles dont les postes doivent être installés par du personnel moins qualifié ; elle nécessite nettement plus de travail pour sa configuration initiale mais est beaucoup plus simple à utiliser ensuite. Le processus est assez complexe et tous les détails peuvent être trouvés dans la collection Samba-HOWTO.

1. installer et configurer *Samba* (version 3.0 ou supérieure pour l'ensemble des fonctions, mais l'essentiel fonctionne depuis la version 2.2.4), compilé avec le support de *CUPS* ;
2. configurer *CUPS* pour le support des imprimantes *MS-Windows* en ajoutant le paquetage des pilotes *CUPS* ;
3. utiliser **cupsaddsmb** pour installer les pilotes *MS-Windows* de *CUPS* dans *Samba* ;
4. se connecter depuis un client *MS-Windows* (avec un identifiant permettant la modification de la configuration d'imprimantes sur le serveur) pour ajuster les caractéristiques de l'imprimante par défaut (taille de page, etc.) ; c'est une tâche plus ardue qu'il n'y paraît car *MS-Windows* présente deux fenêtres identiques à différents emplacements de l'I.H.M. de configuration et une seule s'applique aux réglages par défaut (les détails sont dans le Samba-HOWTO) ;
5. rechercher le serveur dans les favoris réseau sur chaque poste *MS-Windows*, effectuer un clic droit sur l'imprimante choisie et choisir « Connecter ». L'imprimante apparaît alors dans les imprimantes locales et peut être utilisée sans problème.

14.9.5 Schémas de migration de l'impression

Pour les petits sites avec quelques postes et imprimantes, il est simple de configurer un serveur d'impression GNU/Linux et de reconfigurer chaque client à la main. Si plusieurs imprimantes sont partagées par des postes, il peut être opportun de consolider celles-ci sur un serveur d'impression. Cette tâche peut être facilitée par l'ajout de cartes Ethernet aux imprimantes qui le permettent (cela peut apporter aussi une amélioration substantielle des performances par rapport aux interfaces série ou parallèle). Les imprimantes parallèles peuvent bénéficier de boîtiers d'impression réseau.

Des sites plus importants bénéficieront certainement de l'utilisation d'un ou plusieurs serveurs d'impression ; ces machines peuvent aussi assurer d'autres tâches mais si le volume d'impression est substantiel, il faut se souvenir que la conversion de PostScript vers d'autres formats est consommateur de temps processeur et les machines doivent être dimensionnées en conséquence. La configuration Pointer - imprimer est intéressante s'il faut supporter des clients *MS-Windows* car la migration des postes des anciens serveurs *MS-Windows* vers les nouveaux GNU/Linux peut alors être réalisée par un simple script de connexion.

14.9.6 Problèmes potentiels

Certains problèmes habituels peuvent être évités par une planification rigoureuse :

- chaque imprimante doit être contrôlée par un seul serveur et tous les postes et serveurs doivent envoyer leurs travaux d'impression au serveur de contrôle de l'imprimante choisie (cela est particulièrement important pour les imprimantes réseau) ; dans le cas contraire, une imprimante peut recevoir plusieurs travaux d'impression simultanés avec corruption possible du résultat ;
- dans la mesure du possible, un seul jeu de pilotes doit être utilisé pour formater les documents vers une imprimante ; cela est souvent meilleur sur le serveur de contrôle de l'imprimante mais pas nécessairement - cela dépend dans une certaine mesure du serveur qui dispose du meilleur pilote pour celle-ci. Les autres machines par lesquelles transitent les impressions doivent les traiter comme des données brutes ; dans le cas contraire, un pilote peut tenter de formater un flux qui l'a déjà été, au détriment du résultat. Cela ne peut être un problème que lorsque l'édition contient des données binaires.

14.9.7 Autres informations sur l'impression

Une grande quantité d'informations est disponible sur le web ; les sites suivants sont de bons points de départ :

- http://www.cups.org/ - *CUPS*, the Common Unix Printing System ;
- http://www.linuxprinting.org/ ; le site de l'impression Linux avec une grande quantité d'informations utiles ;
- http://www.linuxprinting.org/kpfeiles/SambaPrintHOWTO - le HOWTO de l'impression *Samba* ; il s'agit du site de la distribution de l'auteur et le document est un travail en cours. Chercher la dernière version.

14.10 Applications natives

Ces applications pour lesquelles n'existe aucune alternative OSS et ne peuvent être recompilées devront être exécutées sur leur système d'exploitation natif ou par un émulateur matériel ou logiciel. Les techniques évoquées au chapitre 13.4 s'appliquent.

14.11 Protection anti-virus

Un paquetage anti-virus à jour est désormais essentiel dans les environnements *MS-Windows* et *Apple Macintosh*. Même les organisations peu concernées par la sécurité prennent des risques en ignorant cette protection.

Par contraste, il existe très peu de virus fiables affectant les systèmes OSS. En conséquence, la protection anti-virus en environnement OSS est souvent limitée à l'analyse du courriel pour éviter le passage de virus vers les utilisateurs *MS-Windows*.

Dans le passé ont existé des attaques automatiques de systèmes OSS dont la plus fameuse était le « ver de Morris ». Une grande attention portée sur la sécurité depuis cet événement a considérablement réduit les risques mais il est toujours possible qu'un virus efficace puisse voir le jour. De bonnes pratiques d'administration système ainsi que l'éducation permanente des utilisateurs sont actuellement une meilleure défense que les logiciels anti-virus.

Deux projets OSS anti-virus sont actuellement connus : *Open Anti Virus* (http://www.openantivirus.org/) et *ClamAV* (qui a apparemment disparu d'Internet). Tous les deux sont dans les premiers stades du développement et ne peuvent être recommandés actuellement. De nombreux produits anti-virus commerciaux disposent de versions qui s'exécutent sur plates-formes OSS. Ces versions ne sont pas totalement équivalentes à leurs équivalents *MS-Windows* - en général plus orientées vers l'analyse de courriel que vers la détection à la volée de virus dans le code exécuté, courant dans ce dernier environnement. Cependant, comme expliqué ci-dessus, cette détection à la volée est, pour l'essentiel, inutile sur les systèmes OSS ; l'analyse de courriel est généralement suffisante.

14.12 Références

- http://www.samba.org/ : le serveur de fichiers/impression/domaine *Samba* ;
- http://www.openldap.org/ : le serveur d'annuaire *OpenLDAP* ;
- http://www.kernel.org/pub/linux/libs/pam/Linux-PAM-html/pam.html : le guide de l'administrateur système Linux-PAM ;
- http://samba.mirror.ac.uk/samba/docs/man/Samba-HOWTO-Collection.html : la collection principale des HOWTO *Samba* ;
- http://www.csn.ul.ie/~airlied/pam_smb : le module PAM pam_smb d'authentification des

utilisateurs GNU/Linux avec SMB ;

- http://samba.idealx.org/index.en.html : les outils et HOWTO IDEALX en relation avec *Samba* (certains sont en français).

15 Scénario 2 - Unix

L'administration dispose de serveurs Unix (« gros Unix » - Solaris, HP/UX, AIX, OSF/1, etc.), utilise des PC avec des applications client/serveur et quelques stations Unix et terminaux X.

La migration des postes PC est similaire à celle du scénario 1 ci-dessus. Les stations et terminaux X exécutent en principe des applications X qui devraient s'exécuter sans problème sur les nouveaux postes OSS. Le principal problème réside dans la migration des serveurs.

La migration d'Unix vers GNU/Linux est similaire au portage entre deux versions d'Unix ; il faut garder en tête que le terme Unix recouvre les bases AT&T, BSD et OSF/1 qui sont des implantations différentes du standard POSIX - tout comme GNU/Linux. Les différences apparaissent lorsqu'un programme utilise des fonctions non-POSIX, typiquement celles d'administration système et d'optimisation de performances. Les programmes écrits en tenant faiblement compte de POSIX auront aussi des problèmes - l'écriture de programmes portables est un art et les productions « maison » nécessiteront sans doute un certain travail (l'élimination des alertes de compilation produites au plus haut niveau d'alerte est un bon début). Cependant, les problèmes seront plus souvent du niveau du détail plutôt que du niveau architectural. Les Unix utilisent des protocoles ouverts tels que TCP/IP ainsi que les services communs tels que DNS et DHCP.

Il est aussi vraisemblable que la configuration soit différente, quoique le format de stockage des données ait peu de chances d'être propriétaire et donc leur manipulation par GNU/Linux devrait être assez aisée. Cela inclut les noms et mots de passe d'utilisateurs bien que des différences subtiles puissent en empêcher le transfert direct.

Si le code source est disponible, la recompilation devrait permettre le portage du code. Il faudra cependant traiter certains points :

1. il n'existe pas de standard pour les emplacements des fichiers et ceux-ci peuvent être codés en dur dans les programmes (comme par exemple **/usr/bin**, **/usr/local/bin** ou **/opt/bin**) ;
2. la valeur des constantes système peut varier, comme par exemple le nombre maximal de fichiers pouvant être ouverts simultanément ;
3. il peut y avoir des différences subtiles dans le langage de programmation, par exemple ksh et pdksh ; les différents compilateurs C sont plus ou moins stricts dans le contrôle syntaxique et un code autorisé sur une machine peut engendrer une erreur sur une autre.
   Le code peut ne pas être portable par exemple en raison de :
   A) l'utilisation non portable de constantes, comme celle d'un nombre au lieu de SIGPIPE (défini dans un en-tête C) ; c'est un exemple d'une programmation dans un système d'exploitation au lieu du standard POSIX ;
   B) présupposés sur la longueur de mot ou l'ordre des octets.

gcc, le compilateur de GNU/Linux, dispose d'options permettant une certaine souplesse dans ces circonstances.

4. chaque Unix peut avoir des fichiers d'en-têtes et des bibliothèques différents, qui peuvent aussi se trouver à différents emplacements. ; les emplacements et noms peuvent être modifiés automatiquement dès lors qu'ils ont été trouvés si cependant ces éléments ont des comportements différents, une intervention manuelle peut être nécessaire, par exemple :
   A) la sémantique de certains appels système est différente :
      a) pour les threads,
      b) exec (le bit setuid des scripts est ignoré),
      c) entrées/sorties asynchrones,
      d) ioctl pour le contrôle des tty ;
   B) valeurs différentes de errno.

   Le code d'origine peut utiliser des applications ou bibliothèques propriétaires non disponibles sous

GNU/Linux. Il peut nécessiter une réécriture pour utiliser ce qui est disponible ; ce peut être notamment le cas si des interfaces matérielles spécifiques sont nécessaires, une carte de télécopie par exemple. Il peut y avoir beaucoup de travail dans ces circonstances.

5. les fichiers makefile qui servent à la compilation peuvent nécessiter une adaptation correspondant aux différentes mentionnées ci-dessus ;
6. les applications peuvent faire des suppositions sur des sous-systèmes spécifiques tels que l'impression et les bases de données ; cela peut impliquer par exemple une réécriture de code SQL ;
7. le portage de tout code vers un nouveau matériel, compilateur ou système d'exploitation peut faire apparaître des anomalies qui étaient depuis toujours dans le programme sans jamais apparaître, par exemple parce que la mémoire est organisée différemment, que les entiers ont une taille différente ou que les octets sont ordonnés différemment.

   Les références qui suivent donnent plus de détails

   - http://www.linuxhq.com/guides/LPG/node136.html ;
   - http://www1.ibm.com/servers/esdd/articles/porting_linux/index.html?t=gr,l=335,p=PostSolaris2Linux ;
   - http://www-106.ibm.com/developerworks/linux/library/l-solar/?open&t=grl,l=921,p=Sol-Lx ;
   - http://www-1.ibm.com/servers/eserver/zseries/library/techpapers/pdf/gm130115.pdf ;
   - http://www.unixporting.com/porting-guides.html.

16 Scénario 3 - grand système

L'administration s'appuie sur un grand système (qui peut être MVS, VM/CMS, AS/400 ou même Unix). De nombreux utilisateurs ont des terminaux en mode texte. Il y a quelques PC, utilisés essentiellement en émulation de terminal mais avec une ou deux applications locales.

Ce scénario est similaire à celui du client léger pour ce qui concerne le poste de travail, en particulier si l'architecture doit perdurer.

Aucune information sur la migration des serveurs n'était disponible.

17 Scénario 4 - client léger

L'administration utilise des postes légers qui accèdent par *Citrix* ou similaire à un mélange d'applications MS-Windows et Unix.

L'utilisation du BRA n'est pas supposée ici car les raisons qui ont poussé à l'adoption du modèle client léger ont des chances d'exister encore. Si cependant une migration vers le BRA est imaginée, la plupart des problèmes du scénario 1 se produiront. La migration de ce scénario dans ces conditions est alors très simple car l'architecture ne doit pas évoluer.

Puisque le client est très léger, tout ce qui est nécessaire est un outil de visualisation OSS pour chaque protocole. Le système de fenêtrage ne nécessite pas de grandes capacités, donc un gestionnaire léger tel que *tvwm* peut être suffisant.

Les protocoles suivants peuvent être supportés (parmi d'autres) :

1. HTTP : n'importe quel navigateur OSS devrait suffire. La capacité d'exécution de JavaScript et Java doit être étudiée ; de plus, les composants supplémentaires nécessaires doivent être supportés directement, via un substitut utilisant le paquetage *plugger* ou le paquetage propriétaire *CrossOver Plugin* de CodeWeavers ;
2. ICA : c'est le protocole propriétaire de *Citrix*. Cette société fournit un afficheur ICA gratuit mais non-OSS qui fonctionne sous GNU/Linux ;
3. RDP : c'est le protocole utilisé par *MS-Windows Terminal Server* ; un afficheur OSS, *rdesktop*, est disponible ;
4. VT220, VT100, etc. : ces protocoles DEC sont tous supportés par *xterm* avec la variable d'environnement TERM appropriée (*xterm* peut émuler de nombreux types de terminaux : par exemple, la valeur **TERM=**prism9 émule le protocole utilisé par les machines PRIME) ; la connexion au serveur s'effectue par telnet (toutes les émulations supposent une connexion telnet ou similaire ainsi qu'un protocole en mode caractère et non en mode page) ;
5. 3270 : le programme *x3270* fournit le support approprié ; la connexion au serveur s'effectue par telnet ;
6. X : c'est le protocole d'affichage natif sous GNU/Linux et ne devrait poser aucun problème.

Il existe des produits propriétaires pour certains des protocoles plus ésotériques.

Le projet de serveur de terminaux Linux (LTSP - http://www.ltsp.org/) fournit nombre de kits pour bâtir des clients légers fondés sur GNU/Linux. Ce projet est extrêmement actif et la qualité des logiciels semble excellente.

Les modifications nécessaires sur les serveurs sont similaires aux considérations abordées dans les autres scénarios.

Annexes

18 Annexe A : Études de cas publiées

18.1 http://www.turku.fi/tieto/liite44.rtf

Test par la ville de Turku de :

- OpenOffice.org ;
- x3270 ;
- IBM Host On-Demand ;
- WRQ Reflection for the Web ;
- Hansa ;
- émulateur AS400 ;
- Netscape Communicator ;
- Mozilla ;
- Konqueror ;
- F-Secure Anti-Virus.

Aucune expérience de migration réelle n'est rapportée. L'intention est de migrer en 2003.

18.2 http://www.m-tech.ab.ca/linux-biz

Il s'agit d'un ensemble de sites utilisant apparemment GNU/Linux. L'information sur chaque site est faible mais contient des liens vers des contacts possibles susceptibles de fournir plus d'informations.

18.3 http://www.washingtonpost.com/ac2/wp-dyn/A59197-2002Nov2?language=printer

Il s'agit d'une description de l'utilisation de l'OSS en Estrémadure (Espagne). Une distribution locale appelée linex (http://www.linex.org/) a été créée. Leur expérience est que les utilisateurs ont peu de problèmes d'utilisation ; ceux-ci nécessitent un accès à *MS-Windows* pour traiter les formats Microsoft dans certaines circonstances. Ils ont annoncé récemment le support de 80 000 utilisateurs.

18.4 http://www.newsforge.com/print.pl?sid=02/12/04/2346215

C'est une histoire sur la ville de Largo qui contient de nombreux liens et étudie le coût de possession.

18.5 http://people.trustcommerce.com/~adam/office.html

C'est une entreprise qui a décrit son expérience de migration vers l'OSS sur les postes comme sur les serveurs (site KDE).

18.6 http://www.business2com/articles/mag/print/0,1643,44531,00.html

C'est une description de Zumiez qui a installé des postes Gnome Ximian, ainsi que d'autres études de cas.

18.7 http://lwn.net/Articles/13301/?format=printable

Ce site présente des expériences au Danemark. Il se réfère à un rapport du Danish Board of Technology malheureusement non entièrement traduit en anglais[3].

3 NdT : Le traducteur non plus ne lit pas le danois :-(

18.8 http://www.siriusit.co.uk/support/casestudies/k_g_case.html

C'est une étude de cas sur une entreprise britannique qui a récemment migré vers l'OSS.

18.9 http://staff.harrisonburg.k12.va.us/~rlineweaver

C'est une expérience d'utilisation de l'OSS dans des écoles.

18.10 http://www.li.org/success

C'est une liste d'études de cas - quelques informations utiles.

18.11 http://statskontoret.se/pressrum/press/2003/press030207english.htm, http://www.statskontoret.se/pdf/200308eng.pdf et http://www.statskontoret.se/pdf/200308engappendix.pdf

C'est une étude de l'OSS dans les administrations publiques en Suède.

18.12 http://www-3.ibm.com/software/success/cssdb.nsf/topstoriesFM?OpenForm&Site=linuxatibm

C'est une liste d'études de cas spécifiques IBM qui ne contiennent aucun détail réel et sont très orientées serveur mais donnent une idée de l'utilisation de GNU/Linux dans le monde réel.

18.13 http://h30046.www3.hp.com/search.php?topiccode=linuxCASESTUDY

C'est une liste d'études de cas spécifiques HP.

18.14 http://openapp.biz/seminar/Tony_Kenny/Tony_Kenny.pdf

L'hôpital Beaumont de Dublin a entièrement migré vers l'OSS sur plusieurs années, non seulement pour l'informatique technique mais aussi pour l'administrative. Le lien donne de nombreux détails. L'hôpital pense économiser par ce changement un ordre de grandeur de 13 millions d'euros sur une période de 5 ans. Il s'est rendu compte que, correctement administrée, la migration a été trouvée acceptable par toutes les équipes et que les systèmes OSS peuvent être utilisés de manière beaucoup plus efficace.

19 Annexe B : Wine

Wine est l'acronyme récursif de « Wine Is Not an Emulator[4] » et tous les détails peuvent être trouvés sur http://www.winehq.com/.

19.1 Histoire

Le développement de *Wine* a commencé autour de 1993 par Bob Amstadt qui utilisait GNU/Linux et *MS-Windows* sur la même machine. Les logiciels GNU/Linux étaient arrivés à un point permettant de satisfaire de nombreux besoins mais il avait certains jeux dont il était fou qui n'étaient disponibles que sous *MS-Windows*.

Fatigué de redémarrer uniquement pour jouer, il commença à travailler sur le moyen d'intercepter les appels système des jeux utilisés pour les faire correspondre à ceux de l'environnement X sous GNU/Linux. D'autres entendirent parler de ce travail et vinrent à la rescousse, jusqu'à ce que *Wine* soit capable d'exécuter les jeux auxquels ils souhaitaient jouer.

Autour de 1995, certains tentèrent d'exécuter d'autres programmes, dont *Quicken* et la suite *MS-Office* et ceux-ci devinrent le centre d'intérêt principal. Un groupe distinct fit sécession, continuant le support des techniques graphiques complexes utilisées dans les jeux mais rarement nécessaires dans les applications bureautique. Le projet prit alors une approche plus formalisée, avec une équipe de développeurs et un chef de projet. Depuis 2000; le projet a été repris de manière plus systématique avec un chef de projet et une équipe support basés aux États-Unis, deux petites équipes de développement au Canada et des développeurs dans la plupart des pays européens. Des éditeurs majeurs contribuent aussi, IBM par exemple.

Des techniques d'identification des appels systèmes par les programmes furent créées. Dans de nombreux cas, le développement d'une petite quantité de code permettait le fonctionnement d'une application. Il fut trouvé que les programmes préparent souvent l'appel à une interface particulière sans réaliser l'appel lui-même. Le code fut alors écrit pour permettre aux programmes de continuer à faire ces appels préparatoires sans déclencher d'erreur intermédiaire, ainsi que le code des appels eux-mêmes.

La première utilisation commerciale de *Wine* fut réalisée par Corel qui réalisa un gros travail de support de *Wine* et l'utilisa pour produire une version native GNU/Linux de *WordPerfect 8*. D'autres entreprises on depuis utilisé *Wine* pour produire une version GNU/Linux de leurs produits avec le minimum d'efforts et de modifications, l'une des dernières étant Xilinx qui réalise des paquetages de CAO spécialisée en électronique. Le projet Ximian *Mono* prévoit d'utiliser *Wine* pour permettre aux applications .NET écrites pour *MS-Windows* de fonctionner sans réécriture (voir http://appde.winehq.com/ pour le détail du niveau de support de diverses applications).

Récemment, une équipe de développeurs *MS-Windows* expérimentés a commencé la réalisation d'une suite de programmes de test systématique des plus de 12 000 appels système actuellement présents dans l'ensemble des bibliothèques *MS-Windows*.

Actuellement, *Wine* se compose d'environ 750 000 lignes de code « C » implantant autour de 90% des appels des spécifications courantes de *MS-Windows* telles que ECMA-234 et Open32. Les appels non publiquement documentés sont plus difficiles à implanter mais des progrès sont en cours.

Certaines entreprises qui travaillent sur *Wine* développent du code pour des fonctions particulières initialement propriétaires. Elles font cela pour se fondre elles-mêmes ainsi que leur travail dans le flux principal du projet. Elles insèrent leur code dans ce flux lorsqu'elles disposent d'une source de revenus alternative suffisante. Les supports d'OLE et de ActiveX entrent dans cette catégorie.

19.2 Ce que *Wine* fait

Wine intercepte tous les appels système *MS-Windows* et *MS-DOS* ainsi que les interruptions BIOS

4 NdT : Le traducteur s'est laissé dire que c'est aussi l'acronyme de WINdows Emulator...

pour tenter de les faire correspondre dans l'environnement X GNU/Linux. Les instructions natives du processeur sont exécutées comme elles l'auraient été dans l'environnement *MS-Windows* et c'est pourquoi *MS-Wine* n'est pas un émulateur intégral.

Wine n'est pas lié à l'architecture Intel x86 - il existe par exemple une version pour Alpha de DEC/Compaq, mais le besoin et l'utilisation sérieuse n'existent que sur x86. Il ne permet pas aux programmes *MS-Windows* x86 de s'exécuter sur une autre architecture telle que PowerPC ni SPARC, quoique *Wine* puisse se compiler et s'exécuter sur les deux.

Toutes les interfaces de l'environnement *MS-Windows* ne peuvent avoir de correspondance dans l'environnement X GNU/Linux. Certaines n'ont simplement pas d'équivalent. Cela veut dire que dans certains cas une quantité significative de code doive être écrite pour permettre la correspondance. Par exemple, il y a des problèmes avec les curseurs les plus complexes utilisés par certains programmes *MS-Windows*. Le système X-Window ne peut pas gérer plus de deux couleurs dans un curseur, ce qui oblige *Wine* à faire des hypothèses sur les couleurs à utiliser, parfois avec un résultat inutilisable.

Wine est en réalité composé de deux produits ; *Wine* lui-même qui permet l'exécution de programmes *MS-Windows* et *Winelib* une bibliothèque de compilation destinée à la production de programmes natifs GNU/Linux (c'est celle-ci que Corel a utilisé pour la version GNU/Linux de *WordPerfect*).

Winelib peut être utilisée pour exécuter sur un matériel non-x86 des programmes dont le code source est disponible, quoique des problèmes spécifiques existent encore pour d'autres architectures (notamment concernant l'ordre des octets).

19.3 Où Wine est bon

Le support des programmes *MS-Windows 3.x/95/98/Me/NT* est disponible (moins complet pour *MS-Windows NT*). De nombreux programmes pour *MS-Windows 2000* fonctionneront, sauf s'ils utilisent de nouvelles interfaces spécialisées introduites par ce dernier. Peu de travail a été fait pour le support des programmes spécifiques *MS-Windows XP* mais il en existe très peu.

Wine supporte l'essentiel des interfaces documentées de *MS-Windows*, cependant pas toujours de manière aussi complète que l'on pourrait le souhaiter. Voir http://www.winehq.com/?page=status pour connaître l'état actuel du support dans *Wine*.

Les programmes qui s'exécutent en isolation ou utilisent uniquement des interfaces de communication externes doivent fonctionner. Chaque programme doit être testé individuellement car l'interaction des interfaces et paramètres utilisés peut poser problème.

Il a été rapporté l'utilisation avec succès de certains compilateurs et environnements de développements.

19.4 Où Wine n'est pas bon

Certaines zones spécifiques sont incomplètes, l'échange dynamique de données (DDE) par exemple ; cependant, de nombreux programmes font des appels DDE sans utiliser ceux-ci en réalité et ainsi fonctionnent très bien. OpenGL et autres logiciels graphiques à hautes performances ont aussi des problèmes. L'implantation des listes de contrôle d'accès (ACL - comme dans *MS-Windows NT*) existe en partie mais n'est pas encore intégrée avec les ACL du système d'exploitation sous-jacent.

La technologie des pilotes VxD, introduite par *MS-Windows 98*, est une zone difficile. Elle nécessite l'accès au matériel et à l'intérieur du noyau d'une manière qu'aucun système multi-utilisateur sérieux ne peut permettre. Des techniques existent pour produire des résultats équivalents mais elles impliquent une quantité de travail et leur fonctionnement n'est pas garanti. Dans certains cas, les éditeurs peuvent être convaincus de produire une version GNU/Linux utilisant les interfaces normales. Ce type d'accès étant abandonné par Microsoft (les architectures de type *MS-Windows NT* ne le permettent pas), cela cessera progressivement d'être un problème.

Certains programmes *MS-Windows* tentent de manipuler directement les périphériques (notamment les ports série). Cela n'est pas autorisé sous GNU/Linux ni sous Unix. Cela ne s'applique en principe qu'aux paquetages de communication tels que *Procomm* ainsi qu'aux programmes issus du monde *MS-DOS* pour lequel il était nécessaire de procéder ainsi.

Le rendu de certaines images graphiques n'est pas encore satisfaisant (en particulier celui des polices TrueType) mais ce but est activement poursuivi.

L'autre zone de difficulté est le logiciel produit par Microsoft lui-même, car celui-ci tend à utiliser des interfaces non documentées. Quoiqu'il soit possible de découvrir ce qui se passe, les développeurs doivent être prudents car les lois sur la rétro-ingénierie sont très strictes dans certains pays. Les États-Unis par exemple interdisent celle-ci dans tous les cas et de nombreux pays occidentaux ne l'autorisent que pour établir une compatibilité. Ainsi, le travail sur cette partie restera toujours assez lent.

L'exécution de logiciels d'installation, en particulier, a été problématique mais de récents travaux ont résolu l'essentiel des difficultés et le travail continue. Certaines de celles-ci sont causées par les développeurs de paquetages qui n'appliquent pas les techniques recommandées. L'accès à la base de registre en est un exemple. *Wine* maintient sa base de registre dans un format différent de celui de *MS-Windows* pour en faciliter la récupération. Tant que les interfaces documentées sont utilisées, cela n'a pas d'importance, mais parfois, les développeurs tentent d'y accéder directement, au risque de corrompre une base de registre réelle sous *MS-Windows*, avec pour résultat un fonctionnement impossible sous *Wine*.

Wine est parfois critiqué pour ses faibles performances mais cela est souvent dû à la quantité de code de déverminage. Il est possible de compiler *Wine* sans celui-ci mais cela doit être réalisé avec prudence car alors les éventuels problèmes ne pourront être diagnostiqués sans une nouvelle recompilation.

19.5 Wine - alternatives commerciales

Comme mentionné plus haut, des versions étendues de *Wine* sont disponibles comme produits commerciaux afin d'aider au développement du tronc principal de celui-ci. Les deux entreprises qui font cela sont Transgaming et CodeWeavers. Transgaming travaille essentiellement sur l'amélioration des interfaces graphiques et sonores et son produit est destiné au marché du jeu. CodeWeavers travaille sur les applications bureautiques principales et édite un produit, *CrossOver Office* qui supporte *MS-Office* et *Lotus Notes*.

19.6 Wine et Visual Basic

Il a été rapporté un fonctionnement de *Visual Basic 3* mais aucun détail n'est disponible.

Visual Basic 6 ne s'installe actuellement pas. Le travail est en cours pour régler cela, mais il est trop tôt pour dire si le résultat sera un succès complet ou non.

Aucune autre version n'a été testée.

19.7 Migration d'application vers Wine

Voici une liste des lignes directrices de gestion du processus de migration d'applications sous GNU/Linux avec *Wine* :

1. contrôler les conditions de licence : certaines entreprises ont réalisé des licences qui interdisent l'exécution de leur application hors du système d'exploitation cible. Oracle par exemple faisait cela et Microsoft commence à le faire pour les composants qui peuvent être téléchargés gratuitement ; faire une liste à part des programmes qui entrent dans cette catégorie ;
2. obtenir une copie de chaque application à migrer ; les licences site peuvent ne pas permettre les copies pour tests ;
3. configurer une machine avec la dernière version de *Wine* ;
4. tester chaque programme de la liste de test, noter tous les problèmes rencontrés ainsi que la phase de

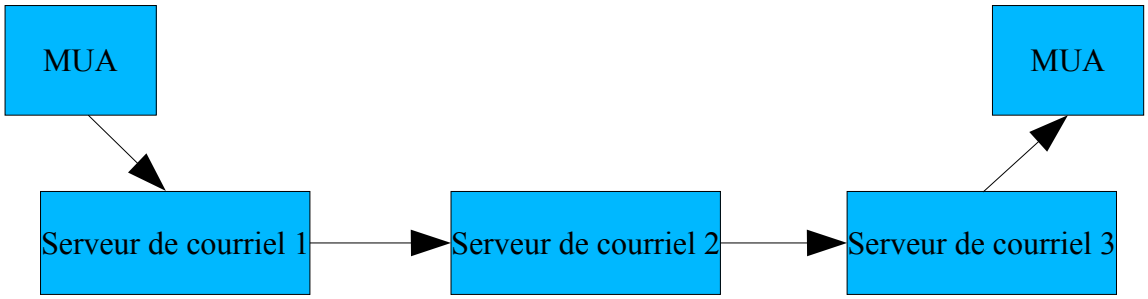
survenance (installation, initialisation ou exécution) et valider leur impact sur une sélection représentative d'utilisateurs finals ; noter aussi les problèmes de performances ;

5. pour chaque programme de la liste de problèmes, contrôler tout d'abord l'existence éventuelle d'une version GNU/Linux et tester alors celle-ci aussitôt que possible ; dans le cas contraire, contacter l'éditeur pour lui suggérer d'en réaliser une, par exemple avec *Winlib* (il faudra négocier séparément avec chaque éditeur) ;
6. si l'éditeur refuse de coopérer, il faudra trouver une application alternative ou abandonner le projet ;
7. utiliser la liste des DLL et appels système manquants nécessaires pour en déterminer un coût d'implantation ; tester à nouveau chaque programme avec les dernières versions de *Wine*/*Winelib* jusqu'à disparition de tous les problèmes (parfois, les correctifs posent des problèmes à des programmes qui fonctionnaient correctement) ;
8. *Wine* est normalement compilé avec la trace de déverminage, ce qui impacte négativement les performances (en particulier pour l'interaction écran) ; tester à nouveau tous les programmes qui fonctionnent avec des problèmes de performances sur une version de *Wine* compilée sans la trace de déverminage ; si les performances restent insuffisantes, un travail de développement sera nécessaire.

20 Annexe C : systèmes de courriel

Cette annexe détaille les systèmes de courriel en général car la gamme de produits OSS peut parfois engendrer une confusion et la terminologie utilisée n'est pas toujours claire.

Le modèle de courriel Internet est fondé sur un certain nombre de composants logiques, chacun en charge d'une tâche spécifique et communicant avec les autres par le biais de protocoles ouverts. Ce modèle est celui utilisé par les systèmes OSS. Il est mieux décrit par quelques diagrammes.



Dessin 1Modèle de courriel Internet

Ce diagramme montre le chemin de remise d'un courriel. Celui-ci est engendré par un agent utilisateur de courriel (MUA - Mail User Agent). Il est ensuite passé à un serveur de courriel qui doit décider s'il peut le remettre localement ou s'il doit être passé à un autre serveur. Le courriel est passé de serveur à serveur jusqu'à ce que l'un d'entre eux décide qu'il est à même de remettre le courriel localement, ce qu'il fait. Une fois la remise complétée, le courriel est disponible pour sa lecture par un MUA. Le MUA final a la responsabilité de relever le courriel ainsi que de passer celui-ci à une interface utilisateur pour l'affichage.

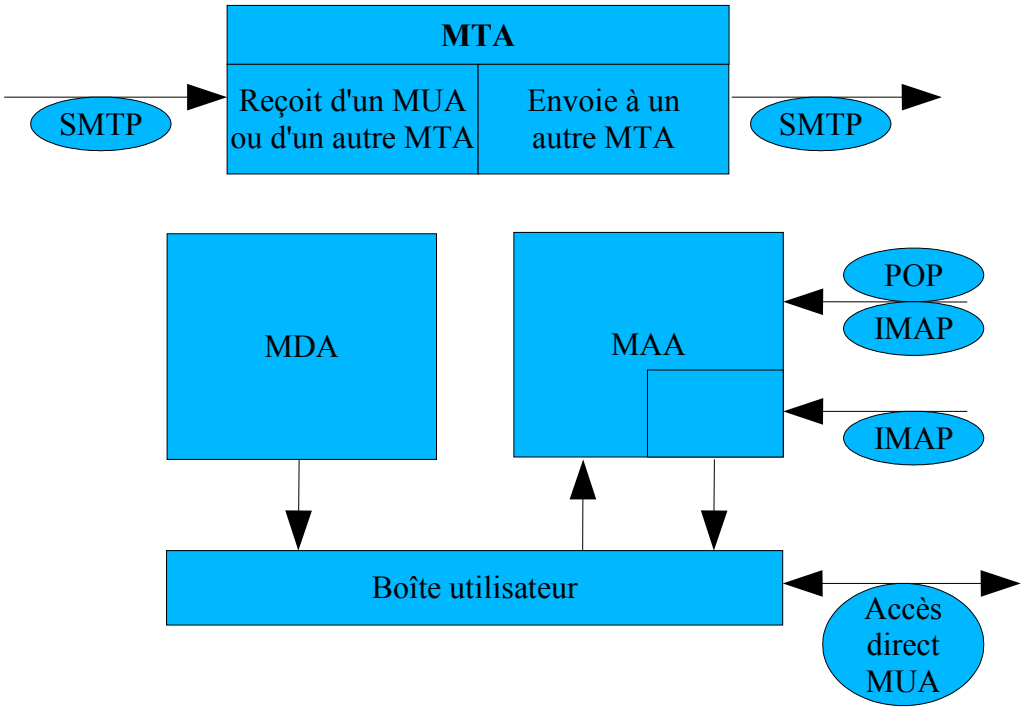
L'algorithme de décision de remise locale ou non pourrait être l'objet d'un autre chapitre. En bref, chaque serveur consulte un ou plusieurs fichiers de configuration locaux ainsi que l'information des serveurs DNS (les enregistrements MX principalement). C'est ce dispositif qui détermine ce qui est considéré comme local. Pour le courriel non local, le serveur utilise ensuite cette information pour déterminer le prochain serveur de courriel à contacter.

La structure générale d'un serveur de courriel est présentée en figure 2.

L'agent de transport de courriel (MTA - Mail Transport Agent) accepte des connexions par les autres serveurs de courriels et les MUA par le protocole simple de transport de courriel (SMTP - Simple Mail Transport Protocol). Si le courriel est destiné à la remise locale, il est passé à un agent de remise de courriel (MDA - Mail Delivery Agent), responsable du stockage dans la boîte du destinataire. La boîte n'est qu'un moyen de stockage de données : un fichier par exemple, un ensemble de fichiers ou même une base de données SQL. La structure de stockage précise est définie par ce que le MDA sait utiliser.

Lorsqu'un utilisateur souhaite consulter son courriel, il utilise un MUA qui, soit relève directement le courriel, soit contacte un composant serveur qui relève le courriel de la boîte pour lui. De tels composants serveurs n'entrent pas dans le modèle traditionnel MTA/MDA/MUA et on peut les appeler des agents d'accès au courriel (MAA - Mail Access Agent)), quoique ce terme ne soit pas d'usage courant.

Le MUA communique avec un MAA à l'aide d'un protocole ouvert, usuellement soit le protocole de bureau de poste (POP - Post Office Protocol) ou le protocole Internet d'accès au courriel (IMAP - Internet Mail Agent Protocol). En principe, le protocole POP supprime le courriel de la boîte lors de la relève tandis qu'IMAP le laisse en place. Ce dernier permet aussi au MUA de modifier la boîte, par exemple en supprimant un courriel ou en le déplaçant d'un répertoire vers un autre.



Dessin 2Composants de serveur de courriel

Le MUA peut stocker le courriel localement sur la machine qui l'héberge. Cela se produit normalement si l'on utilise POP. Ce stockage local permet d'en rendre l'accès futur indépendant du serveur, ce qui est particulièrement utile pour les machines dont la connexion au réseau n'est pas permanente. IMAP, d'un autre côté, fonctionne en principe sans copie locale mais peut aussi fonctionner dans ce qu'on appelle le mode déconnecté, qui conserve une copie locale à laquelle on peut accéder sans connexion réseau. Dans ce mode, les boîtes locale et serveur sont synchronisées lors de l'établissement d'une connexion réseau. Malheureusement, tous les MUA n'assurent pas un support total de l'IMAP déconnecté.

Parfois, un programme tiers relève le courriel et le stocke en local pour que le MUA y accède sans avoir à se connecter lui-même au serveur. Ces types de programmes téléchargent le courriel contrairement aux MTA qui reçoivent celui-ci poussé par un autre MTA. Cela peut être utile si les utilisateurs ne souhaitent pas permettre de connexion vers leur machine depuis Internet ou s'ils opèrent derrière un pare-feu. Un exemple de programme de ce type est *fetchmail*.

La difficulté de ce modèle est que les applications disponibles ne correspondent pas toujours directement car celles-ci assurent souvent plus d'une fonction : un MTA par exemple peut incorporer la fonction MDA et le MTA très répandu *Sendmail* peut même être utilisé comme MUA dans certaines circonstances.

Lorsque le courriel passe du MUA d'origine par les différents serveurs vers le MUA destinataire, un ensemble d'en-têtes est ajouté, détaillant le trajet parcouru et contrôlant le traitement du courriel par les différentes machines. Certains en-têtes sont conformes au standard MIME (Multi-purpose Internet Mail Extension), utilisé pour de nombreuses fonctions de contrôle, dont le support des jeux de caractères non-ASCII, des contenus imbriqués tels que les images ainsi que les pièces jointes. Lorsqu'un MUA joint un fichier, il enregistre son type dans un en-tête MIME et c'est ensuite au MUA destinataire de décoder celui-ci.

Différents éléments de ce modèle sont détaillés ci-dessous.

20.1 Agents de transport de courriel

De nombreux MTA permettent à l'administrateur le contrôle de l'origine des messages acceptés. Celui-ci est souvent réalisé en limitant la plage d'adresses IP depuis lesquelles sont acceptées les

connexions SMTP. Il est extrêmement utile dans la prévention du courriel non sollicité (spam) transmis par le relais du MTA en occupant indûment sa bande passante réseau.

Il existe un ensemble d'environ 20 extensions à SMTP appelé SMTP étendu (ESMTP - Extended SMTP) qui permet notamment aux MTA compatibles un transfert de courriel plus rapide en utilisant l'extension « pipeline ».

Une autre extension active le chiffrement de sécurité de la couche transport (TLS - Transport Layer Extension) et encore une autre, SMTP-AUTH, permet l'authentification d'utilisateurs par différentes techniques. Ces deux extensions sont utiles pour permettre la connexion à un client situé à une adresse IP hors de l'espace autorisé (par exemple pour un ordinateur portable connecté à Internet depuis un site aléatoire). Voir 20.4.2 plus bas.

Le modèle d'origine supposait que le compte de courriel était associé à un compte de connexion sur le serveur de courriel. Cela permettait au MTA d'interroger le fichier local de mots de passe pour l'authentification des utilisateurs. Ce modèle est trop restrictif et les MTA modernes doivent désormais supporter les utilisateurs virtuels dont les caractéristiques sont stockés dans une base de données souvent indépendante de celles de connexion. Cela implique que l'utilisateur peut avoir des mots de passe distincts pour le courriel et pour la connexion.

La base de données peut être au format LDAP, SQL ou un fichier plat. *MySQL* est le serveur SQL préféré en raison de son efficacité et de sa rapidité pour cette application essentiellement en lecture. *PostgreSQL* et *Oracle* peuvent aussi être utilisés.

Une base LDAP est recommandée en raison de son meilleur support pour une structure distribuée.

Les implantations LDAP par défaut utilisent souvent les produits de bases de données Berkeley de Sleepycat Systems.

Parfois, une machine peut se connecter de manière intermittente à un serveur de courriel. Cela peut arriver par exemple dans le cas de télé-travailleurs ou d'utilisateurs de portables, ainsi que dans le cas de petites entreprises pour lesquelles le coût d'une connexion permanente ne peut être justifiée. Dans ces circonstances, le MTA central ne peut pousser le courriel comme il le ferait normalement et doit stocker celui-ci jusqu'à l'établissement d'une connexion. Le cas est similaire pour le MTA (s'il existe) de la machine cliente ou de la passerelle de courriel d'une petite entreprise. Ces MTA doivent être capables de supporter de telles situations et sont souvent appelées hôtes intelligents (Smart Hosts) si c'est le cas.

La remise depuis un hôte intelligent peut être réalisée par SMTP ou POP3.

La remise par SMTP est sans difficulté et la sécurité de la machine destinataire peut être améliorée en restreignant les connexions entrantes au seul hôte intelligent.

La remise par POP3 est réalisée par un MUA ou par l'application *fetchmail*. Ce dernier télécharge le courriel vers une boîte locale comme mentionné ci-dessus ou remet celui-ci à un MTA, par exemple lorsque plusieurs comptes de courriel sont impliqués.

Ces deux méthodes fonctionnent bien mais ont comme inconvénient de ne pas permettre l'utilisation de listes noires pour éviter le spam originaire de relais ouverts et autres sources indésirables. Des outils tels que *SpamAssassin* peuvent éliminer une grande partie du spam mais les coûts de traitement sont beaucoup plus élevés et une bande passante plus élevée est utilisée par la relève du courriel à examiner.

20.2 Agents utilisateur de courriel

Les MUA et MUI ensemble constituent le paquetage perçu par de nombreux utilisateurs comme « application de courriel ». C'est le logiciel client qui s'exécute soit directement sur un serveur web, soit sur le poste de travail pour permettre à ceux-ci l'émission et la réception de courriels. Une certaine forme de stockage est en principe fournie afin que les messages soient triés dans des « dossiers » ou « boîtes locales » pour un usage ultérieur.

Le MUA utilise des protocoles tels que SMTP pour l'émission et IMAP ou POP pour la réception et le tri. Il interprète le format des messages et peut décomposer les messages MIME en leurs différents composants.

Lorsqu'un niveau de sécurité élevé est nécessaire, le MUA est aussi responsable du chiffrement et de la signature des messages. Deux standards concurrents sont en lice : S/MIME, fondé sur les certificats X.509, et PGP/GPG, fondé sur un format de certificats différent avec un modèle « web de confiance » plutôt que « hiérarchie de confiance ».

De nombreux MUA OSS permettent la signature numérique à l'aide de *GNU Privacy Guard* (*GPG*). Peu supportent les signatures S/MIME. Les entités commerciales et gouvernementales ont opté pour le standard S/MIME et son utilisation doit donc être acceptée.

20.3 Stockage de courriel

Les systèmes de courriel des Unix d'origine supposaient que le propriétaire d'un compte de messagerie avait accès à la machine hébergeant le serveur de courriel et pouvait lire un fichier contenant ses messages ou que le courriel était remis à la machine sur laquelle l'utilisateur se connectait habituellement pour travailler. Cela était parfait pour des environnements de peu d'utilisateurs qui devaient avoir aussi un compte de connexion réel sur une machine serveur de courriel, mais n'est généralement ni praticable, ni sûr.

Le format original de stockage de courriel était constitué d'un seul fichier par utilisateur, les nouveaux messages y étant ajoutés à la fin. Ce fichier pouvait devenir très volumineux et sa lecture aléatoire devenait rapidement inefficace. Ce format est souvent désigné par « mbox » et reste utilisé par certains MUA, notamment pour le courriel stocké localement pour l'utilisateur. Une évolution de cela a consisté à stocker chaque élément de courriel dans un fichier séparé d'une structure de répertoire permettant un accès aléatoire nettement plus efficace. Une variante de cette structure est appelée « mh » et une autre, avec certains sous-répertoires et procédures d'accès pré-définis est appelée « maildir ».

Parfois, ces structures étaient conservées sur le serveur de courriel et exportées vers les clients, par exemple par NFS. Cela permettait le stockage central des messages, ce qui en autorisait une sauvegarde correcte mais introduisait des problèmes de verrouillage avec la structure de fichier unique. Cependant, l'utilisation de NFS ne s'est pas avérée populaire, peut-être en raison de l'absence de bons clients NFS sur *MS-Windows*.

Tous les MTA ne permettent pas l'utilisation directe de ces différentes méthodes d'accès, d'où la nécessité des MAA. Un MUA qui ne peut accéder directement au stockage doit utiliser un composant MAA s'appuyant sur POP ou IMAP.

Ces deux protocoles transmettent par défaut les mots de passe en clair. IMAP peut utiliser un algorithme de hachage si le MUA le permet. L'utilisation de liens chiffrés par TLS est possible si le MAA et le MUA le permettent, à conseiller sur les réseaux locaux et doit être indispensable pour l'accès distant.

Parfois, les MTA communiquent avec les MDA par le protocole de transport de courriel local (LMTP - Local Mail Transport Protocol) ; de nombreux MTA et MDA le permettent.

20.4 Utilisateurs itinérants

Le problème, avec les utilisateurs itinérants, est qu'ils peuvent se connecter depuis des adresses IP imprévisibles, ainsi les méthodes habituelles de filtrage du courriel entrant par les MTA leur interdisent l'envoi de message par le serveur de courriel de l'administration. Les MTA doivent interdire l'accès à des clients inconnus pour empêcher leur utilisation comme relais tiers pour le courriel non sollicité.

Trois techniques principales permettent de contourner ce problème.

20.4.1 Réseaux privés virtuels (VPN - Virtual Private Networks)

Dans un VPN, la machine distante peut se voir allouer une adresse incluse dans l'espace d'adressage de confiance. Le problème est que l'accès intégral au réseau interne sera possible à quiconque obtiendra l'accès à la machine distante - un risque significatif avec les portables - sauf si les clefs d'accès sont chiffrées avec un mot de passe entré à chaque établissement de la connexion. Malheureusement, les utilisateurs configurent parfois leur machine pour qu'elle mémorise les mots de passe.

20.4.2 SMTP-AUTH et TLS

L'extension SMTP-AUTH de SMTP permet que la configuration d'un MTA demande un mot de passe pour l'authentification de l'utilisateur distant. Les principales méthodes d'authentification sont PLAIN, LOGIN et CRAM-MD5.

PLAIN nécessite le stockage du mot de passe en clair sur le client mais celui-ci peut être chiffré sur le serveur. Si la connexion SMTP n'est pas chiffrée, le mot de passe est transmis en clair (quoiqu'encodé en base-64) sur le réseau.

LOGIN est moins efficace que PLAIN puisqu'il nécessite trois échanges réseau au lieu d'un et, comme pour PLAIN, l'identifiant et le mot de passe transitent en clair.

CRAM-MD5 chiffre l'identifiant et le mot de passe durant leur transit sur le réseau. Cependant, le mot de passe doit être stocké en clair à la fois sur le client et sur le serveur. Il nécessite deux interactions réseau.

Tous les MUA ne supportent pas SMTP-AUTH et ceux qui le font peuvent ne supporter qu'un nombre limité de méthodes. *Outlook Express* par exemple ne permet que LOGIN.

Comparé à l'utilisation d'un VPN, le seul accès autorisé est l'émission de courriel, ainsi les autres services ne seront pas compromis si la machine distante est volée.

ESMTP permet aussi la négociation d'une session TLS entre le client et le serveur. Cette connexion chiffre les données sur le réseau et peut aussi authentifier la machine cliente. L'authentification nécessite un certificat client correspondant à celui conservé sur le serveur.

20.4.3 POP avant SMTP

Cette méthode tire avantage de ce que POP et IMAP nécessitent une authentification par mot de passe.

Après une connexion réussie par POP ou IMAP, le MAA maintient une base authentifiée des connexions avec l'adresse IP du client, la date et l'heure. Lorsque le client tente d'envoyer du courriel en-dehors du domaine local, le MTA contrôle si l'adresse IP du client se trouve dans son espace d'adressage de confiance. Dans le cas contraire, il recherche son adresse IP dans sa base de connexions. Si aucun enregistrement n'est trouvé ou si la dernière connexion authentifiée est trop ancienne, le MTA refuse le relais du message. Le délai est configurable et sa valeur par défaut est typiquement de 20 minutes. Cette méthode nécessite une coopération entre le MAA et le MTA. Pour cette raison, toutes les combinaisons ne fonctionnent pas.

Cette méthode présente l'inconvénient que les utilisateurs doivent relever leur courriel d'abord. Certains peuvent trouver cela difficile si le MTA ne réalise pas cela automatiquement.

20.5 Performance

En général, un MTA utilise peu de puissance processeur ; les machines dédiées sont habituellement limitées par la bande passante locale ou la performance disque. Les serveurs IMAP et POP nécessitent plus de puissance processeur et IMAP nécessite un peu plus de RAM que POP. Cependant, aucun d'eux ne devrait représenter un problème avec un matériel actuel.

Les analyseurs anti-virus nécessitent beaucoup de RAM et de puissance processeur, particulièrement si les pièces joints MIME sont autorisées.

Même dans ce cas, les limitations de performances proviennent usuellement du trafic plutôt que du nombre de comptes.

Voici quelques exemples de performances issus de rapports de listes de diffusion et de l'expérience des consultants de **netproject**. Ils permettent de donner une idée de ce qui est nécessaire :

- site 1 : 2 Pentium III Xeon 2,4 GHz, 4 Go de RAM, 3x36 Go en SCSI RAID 5 :
  les utilisateurs virtuels sont stockés dans *MySQL,*
  Postfix 2.0.6, *Courier-IMAP* 1.7, *MySQL* 4.12, *RAV-Antivirus*, *Mailman* 2.1, *RedHat Linux* 8.0,
  pas de SSL,
  environ 4 800 utilisateurs ;
- site 2 : Athlon 1 200, 1 Go de RAM, RAID5 :
  Postfix + *Courier-IMAP* (anti-virus sur une machine séparée), pas de SSL,
  8 500 utilisateurs ;
- site 3 : Pentium 133, 40 Mo de RAM, disque IDE :
  Debian GNU/Linux, *Courier-MTA* + *Courier-IMAP* + *SpamAssassin* (ce dernier pour un seul utilisateur),
  typiquement 18 utilisateurs POP3 et 7 utilisateurs IMAP en permanence,
  occupation processeur environ 20% ;
- site 4 : bi-processeur Pentium II 450 Xeon, 256 Mo de RAM :
  MySQL, *Courier-MTA* + *Courier-IMAP*, *sqwebmail*, SSL,
  50 utilisateurs, principalement en POP3 ;
- site 5 : Pentium II 400, 256 Mo de RAM :
  Courier-MTA + *SpamAssassin*, *RedHat Linux* 8.0,
  300 boîtes, environ 4 000 messages par jour ;
- site 6 : Pentium III 677, 512 Mo de RAM, 2 disques IDE :
  FreeBSD 4.7, *Exim* 4.05, *OpenLDAP* 2.1.5, *Cyrus* 2.1.11, *Mailman* 2.1, *Apache* 1.3.26,
  la machine est surtout un serveur web chargé mais traite aussi plusieurs milliers de messages par jour sans charge additionnelle notable.

21 Annexe D : logiciels de référence (poste de travail)

Cette liste montre les paquetages RPM avec leurs numéros de version. Elle est fondée sur *RedHat Linux* version 8.0 avec la mise à jour *Red Carpet* de Ximian et n'inclut que le minimum de dépendances pour l'environnement cible. *Evolution* a été mis à jour par *Red Carpet* vers une version plus récente que celle de *RedHat*.

Nom	*Version*	*Révision*
anacron	2.3	23
aspell	0.33.7.1	ximian.4
atk	1.0.3	1
audiofile	0.2.3	3
basesystem	8	1
bash	2.05b	5
bdflush	1.5	21
bitmap-fonts	0.2	2
bonobo	1.0.21	1.ximian.1
bonobo-activation	1.0.3	2
bonobo-conf	0.16	1.ximian.1
bzip2-libs	1.0.2	5
chkconfig	01/03/06	3
chkfontpath	01/09/06	3
compat-libstdc++	7.3	2.96.110
control-center	2.0.1	8
cpp	3.2	7
cracklib	2.7	18
cracklib-dicts	2.7	18
crontabs	1.1	4
cups-libs	01/01/17	0.2
cyrus-sasl	02/01/10	1
cyrus-sasl-md5	02/01/10	1
db4	4.0.14	14
desktop-background-basic	2	10
desktop-background-extra	2	10
desktop-file-utils	0.3	3
dev	03/03/01	2
dhclient	3.0pl1	26
dialog	0.9b	20020519.1
diffutils	02/08/01	3
docbook-dtds	1	14
e2fsprogs	1.27	9
ed	0.2	28
eel2	2.0.6	1
emacs	21.2	18

Nom	*Version*	*Révision*
eog	1.0.2	5
esound	0.2.28	1
evolution	01/02/02	1.ximian.1
expat	1.95.4	1
fam	02/06/08	4
filesystèem	02/01/06	5
fileutils	04/01/09	11
findutils	04/01/07	7
fontconfig	2	3
fortune-mod	1	24
freetype	02/01/02	7
gail	0.17	2
galeon	01/02/06	0.8.0
gawk	03/01/01	4
Gconf	1.0.9	6
Gconf2	01/02/01	3
gdbm	01/08/00	18
gdk-pixbuf	0.18.0	4
gdk-pixbuf-gnome	0.18.0	4
gdm	2.4.0.7	13
gedit	2.0.2	5
gftp	2.0.13	5
ghostscript	7.05	20
ghostscript-fonts	5.5	7
gimp	01/02/03	9
gimp-print	04/02/01	5
glib	01/02/10	8
glib2	2.0.6	2
glibc	02/03/02	4.80.6
glibc-common	02/03/02	4.80.6
Glide3	20010520	19
gmp	4.1	4
gnome-desktop	2.0.6	4
gnome-libs	1.4.1.2.90	22
gnome-mime-data	2.0.0	9
gnome-panel	2.0.6	9
gnome-session	2.0.5	7
gnome-spell	0.5	1.ximian.3
gnome-terminal	2.0.1	5
gnome-utils	2.0.2	5
gnome-vfs	1.0.5	6.ximian.1

Nom	*Version*	*Révision*
gnome-vfs2	2.0.2	5
gnupg	1.0.7	6
gpm	1.19.3	23
grep	02/05/01	4
grub	0.92	7
gtk+	01/02/10	22
gtk2	2.0.6	8
gtkhtml1.1	01/01/06	1.ximian.1
gzip	01/03/03	5
htmllview	2.0.0	6
hwdata	0.48	1
imlib	01/09/13	9
indexhtml	8	1
info	4.2	5
initscripts	6.95	1
intltool	0.22	3
iproute	02/04/07	5
iputils	20020124	8
kbd	1.06	26
krb5-libs	01/02/05	15
krbafs	01/01/01	6
less	358	28
libacl	2.0.11	2
libart-lgpl	02/03/10	1
libattr	2.0.8	3
libbonobo	2.0.0	4
libbonoboui	2.0.1	2
libcapplet0	1.4.0.1	9
libelf	0.8.2	2
libgal21	0.23	1.80.ximian.1
libgcc	3.2	7
libghttp	1.0.9	5
libglade	0.17	8
libglade2	2.0.0	2
libgnome	2.0.2	5
libgnomecanvas	2.0.2	1
libgnomeprint	1.116.0	2
libgnomeprint15	0.37	2.ximian.1
libgnomeprintui	1.116.0	1
libgnomeui	2.0.3	3
libgtkhtml1.1-3	01/01/08	1.ximian.1

Nom	*Version*	*Révision*
libjpeg	6b	21
libmng	1.0.4	1
libpng10	1.0.13	6
libpng	01/02/02	8
librpm404	4.0.4	8x.27
libsrvg2	2.0.1	1
libstdc++	3.2	7
libtermcap	2.0.8	31
libtiff	03/05/07	7
libungif	04/01/00	13
libuser	0.51.1	2
libwnck	0.17	1
libxml	01/08/17	5
libxml2	02/04/23	1
libxslt	1.0.19	1
linc	0.5.2	2
logrotate	03/06/05	2
losetup	2.11r	10
lvm	1.0.3	9
metacity	2.4.0.92	5
mingetty	1	3
mkinitrd	03/04/28	1
mktemp	1.5	16
modutils	02/04/18	2
mount	2.11r	10
mozilla	1.0.1	26
mozilla-nspr	1.0.1	26
mozilla-nss	1.0.1	26
mozilla-psm	1.0.1	26
nautilus	2.0.6	6
ncurses	5.2	28
net-tools	1.6	7
newt	0.51.0	1
nfs-utils	1.0.1	2
nscd	02/03/02	4.80.6
nss_ldap	198	3
oaf	0.6.10	1.ximian.2
Omni	0.7.0	6
openjade	01/03/01	9
openldap	2.0.27	02/08/00
openoffice	1.0.1	8

Nom	*Version*	*Révision*
openoffice-i18n	1.0.1	8
openoffice-libs	1.0.1	8
openssh	3.4p1	2
openssh-clients	3.4p1	2
openssh-server	3.4p1	2
openssl	0.9.6b	33
ORBit	0.5.13	5
ORBit2	02/04/01	1
pam	0.75	46.8.0
pango	01/01/01	1
passwd	0.67	3
patch	02/05/04	14
pcre	3.9	5
perl	05/08/00	55
perl-Filter	1.28	9
popt	1.7	1.06
portmap	4	46
procps	2.0.7	25
psmisc	20.2	6
pspell	0.12.2	ximian.7
python	02/02/01	17
qt	3.0.5	17
rc	01/02/01	1.ximian.1
rcd	01/02/01	1.ximian.3
readline	4.3	3
redhat-artwork	0.47	3
redhat-logos	01/01/06	2
redhat-menus	0.26	1
redhat-release	8	8
rootfiles	7.2	4
rpm	4.1	1.06
scrollkeeper	0.3.10	7
sed	3.02	13
setup	02/05/20	1
sgml-common	0.6.3	12
shadow-utils	20000902	12.8
sh-utils	2.0.12	3
slang	01/04/05	11
soup	0.7.11	1.ximian.1
switchdesk	03/09/08	9
sysklogd	01/04/01	10

Nom	*Version*	*Révision*
SysVinit	2.84	5
tar	1.13.25	8
termcap	11.0.1	13
tetex	1.0.7	57.1
tetex-fonts	1.0.7	57
textutils	2.0.21	5
tmpwatch	02/08/04	3
urw-fonts	2	26
usermode	1.63	1
utempter	0.5.2	10
util-linux	2.11r	10
Vflib2	2.25.6	8
vixie-cron	3.0.1	69
vnc	3.3.3r2	39.2
vnc-doc	3.3.3r2	39.2
vte	0.8.19	2
which	2.14	1
words	2	20
Xaw3d	1.5	16
Xfree86	04/02/00	72
Xfree86-75dpi-fonts	04/02/00	72
Xfree86-base-fonts	04/02/00	72
Xfree86-font-utils	04/02/00	72
Xfree86-libs	04/02/00	72
Xfree86-Mesa-libGL	04/02/00	72
Xfree86-Mesa-libGLU	04/02/00	72
Xfree86-truetype-fonts	04/02/00	72
Xfree86-xauth	04/02/00	72
Xfree86-xfs	04/02/00	72
Xft	2	1
xinetd	02/03/07	5
xinitrc	3.31	1
xml-common	0.6.3	12
xpdf	1.01	10
xscreensaver	4.05	6
xsri	02/01/00	3
zlib	01/01/04	8.8x

22 Annexe E : logiciels de référence (serveur)

Cette liste montre les paquetages RPM avec leurs numéros de version. Elle est fondée sur *RedHat Linux* version 8.0 avec la mise à jour *Red Carpet* de Ximian. Il ne s'agit pas d'une liste minimale ; certaines applications ont été conservées pour plus de simplicité (par exemple, celles liées à *Gnome* et *X*) car elles permettent l'utilisation d'interfaces d'administration graphiques, mais les interfaces en mode texte pourraient être utilisées en leurs lieu et place.

Nom	*Version*	*Révision*
a2ps	4.13b	24
acl	2.0.11	2
adjtimex	1.13	4
alchemist	1.0.24	4
amanda	2.4.2p2	9
anaconda-help	8	1
anacron	2.3	23
apmd	3.0.2	12
ash	0.3.8	5
at	03/01/08	31
atk	1.0.3	1
attr	2.0.8	3
audiofile	0.2.3	3
authconfig	04/02/12	3
autoconvert	0.3.7	8
autofs	03/01/07	33
autorun	3.3	3
basesystem	8	1
bash	2.05b	5.1
bash-doc	2.05b	5.1
bdflush	1.5	21
beecrypt	02/02/00	6
bg5ps	01/03/00	9
bind	09/02/01	9
bind-utils	09/02/01	9
bitmap-fonts	0.2	2
blas-man	3	18
bonobo-activation	1.0.3	2
booty	0.12	1
bridge-utils	0.9.3	6
bzip2	1.0.2	5
bzip2-libs	1.0.2	5
caching-nameserver	7.2	4
cadaver	0.20.5	2
cdrdao	01/01/05	10
chkconfig	01/03/06	3

Nom	*Version*	*Révision*
chkfontpath	01/09/06	3
cleanfeed	0.95.7b	17
comps-extras	8	3
courier-imap	01/07/01	2.whoson.8.0
courier-imap-ldap	01/07/01	2.whoson.8.0
cpio	02/04/02	28
cracklib	2.7	18
cracklib-dicts	2.7	18
crontabs	1.1	4
cups-libs	01/01/17	0.7
curl	07/09/08	1
cWnn-common	1.11	27
cyrus-sasl	02/01/10	1
cyrus-sasl-gssapi	02/01/10	1
cyrus-sasl-md5	02/01/10	1
cyrus-sasl-plain	02/01/10	1
db4	4.0.14	14
db4-java	4.0.14	14
dev	03/03/01	2
dhclient	3.0pl1	26
dhcp	3.0pl1	26
dialog	0.9b	20020519.1
dictd	01/05/05	3
diffutils	02/08/01	3
diskcheck	1.3	2
docbook-dtds	1	14
dos2unix	3.1	12
dosfstools	2.8	3
dtach	0.5	5
e2fsprogs	1.27	9
ed	0.2	28
eject	2.0.12	7
elinks	0.3.2	1
enscript	01/06/01	22
esound	0.2.28	1
ethtool	1.6	2
exim	4.12	4.rh8x
exim-ldap	4.12	4.rh8x
expat	1.95.4	1
fam	02/06/08	4
fbset	2.1	11

Nom	*Version*	*Révision*
fetchmail	05/09/00	21/08/00
file	3.39	9
filesystem	02/01/06	5
fileutils	04/01/09	11
findutils	04/01/07	7
finger	0.17	14
fontconfig	2	3
freetype	02/01/02	7
FreeWnn-common	1.11	27
FreeWnn-libs	1.11	27
ftp	0.17	15
ftpcopy	0.5.1	1
gail	0.17	2
gawk	03/01/01	4
GConf2	01/02/01	3
gd	01/08/04	9
gdbm	01/08/00	18
genromfs	0.3	12
glib	01/02/10	8
glib2	2.0.6	2
glibc	02/03/02	4.80.6
glibc-common	02/03/02	4.80.6
gmp	4.1	4
gnome-audio-extra	01/04/00	4
gnome-mime-data	2.0.0	9
gnome-python2	1.99.11	8
gnome-python2-bonobo	1.99.11	8
gnome-python2-canvas	1.99.11	8
gnome-python2-gtkhtml2	1.99.11	8
gnome-vfs2	2.0.2	5
gnupg	1.0.7	8
gpg-pubkey	db42a60e	37ea5438
gpm	1.19.3	23
grep	02/05/01	4
groff	1.18	6
groff-perl	1.18	6
grub	0.92	7
gsl	01/01/01	3
gtk+	01/02/10	22
gtk2	2.0.6	8
gtkhtml2	2.0.1	2

Nom	*Version*	*Révision*
gzip	01/03/03	5
h2ps	2.06	6
hdparm	5.2	1
hesiod	3.0.2	21
hotplug	2002_04_01	13
htmlview	2.0.0	6
httpd	2.0.40	11.5
hwdata	0.48	1
imlib	01/09/13	9
indexhtml	8	1
inews	02/03/03	5
info	4.2	5
initscripts	6.95	1
intltool	0.22	3
iproute	02/04/07	5
iptables	1.2.6a	2
iptraf	02/07/00	3
iputils	20020124	8
ipxutils	2.2.0.18	11
isicom	3.05	6
jcode.pl	2.13	6
jfsutils	1.0.17	3
kakasi	02/03/04	8
kakasi-dict	02/03/04	8
kbd	1.06	26
kbdconfig	01/09/16	1
kcc	2.3	14
kdoc	3.0.0	2.cvs20020321.3
kernel	02/04/20	13.8
kernel	02/04/20	18.8
kernel-utils	2.4	8.28
krb5-libs	01/02/05	15
krbafs	01/01/01	6
krbafs-utils	01/01/01	6
ksymoops	02/04/05	1
kudzu	0.99.69	1
lapack-man	3	18
less	358	28
lftp	02/05/02	5
libacl	2.0.11	2
libaio	0.3.13	5

Nom	*Version*	*Révision*
libao	0.8.3	1
libart_lgpl	02/03/10	1
libattr	2.0.8	3
libbonobo	2.0.0	4
libbonoboui	2.0.1	2
libcap	1.1	12
libelf	0.8.2	2
libesmtp	0.8.12	2
libf2c	3.2	7
libgcc	3.2	7
libghttp	1.0.9	5
libglade2	2.0.0	2
libgnome	2.0.2	5
libgnomecanvas	2.0.2	1
libgnomeui	2.0.3	3
libgtop	1.0.12	11
libIDL	0.8.0	3
libjpeg	6b	21
libmng	1.0.4	1
libobjc	3.2	7
libogg	1	1
libole2	0.2.4	4
libpng10	1.0.13	6
libpng	01/02/02	8
librpm404	4.0.4	8x.27
libstdc++	3.2	7
libtermcap	2.0.8	31
libtiff	03/05/07	7
libtool-libs	01/04/02	12
libungif	04/01/00	13
libunicode	0.4	9
libusb	0.1.6	1
libuser	0.51.1	2
libwvstreams	3.7	5
libxml10	1.0.0	11
libxml2	02/04/23	1
libxml2-python	02/04/23	1
libxslt	1.0.19	1
lilo	21/04/04	20
linc	0.5.2	2
lm_sensors	02/06/03	2

Nom	*Version*	*Révision*
lockdev	1.0.0	20
logrotate	03/06/05	2
logwatch	2.6	8
lokkit	0.5	21/08/00
losetup	2.11r	10
LPRng	03/08/09	6.1
lvm	1.0.3	9
m2crypto	0.05_snap4	6
m4	01/04/01	11
macutils	2.0b3	22
mailcap	02/01/12	1
mailman	2.0.13	3
mailx	08/01/01	26
make	3.79.1	14
man	1.5k	0.8x.0
man-pages	1.53	1
mc	04/05/55	12
mdadm	1.0.0	6
mew-common	2.2	6
mingetty	1	3
mkbootdisk	01/04/08	1
mkinitrd	03/04/28	1
mkisofs	1.1	14
mktemp	1.5	16
mod_perl	1.99_05	3
mod_python	3.0.0	10
modutils	02/04/18	2
mount	2.11r	10
mouseconfig	4.26	1
mozilla-nspr	1.0.1	26
mozilla-nss	1.0.1	26
mpage	02/05/02	4
mtools	03/09/08	5
mtr	0.49	7
mt-st	0.7	6
mtx	01/02/16	5
namazu	2.0.10	8
namazu-cgi	2.0.10	8
nc	1.1	16
ncftp	03/01/03	6
ncompress	04/02/04	31

Nom	*Version*	*Révision*
ncpfs	2.2.0.18	11
ncurses4	5	9
ncurses	5.2	28
netconfig	0.8.12	3
netpbm	9.24	9.80.2
net-snmp	5.0.6	8.80.2
net-snmp-utils	5.0.6	8.80.2
net-tools	1.6	7
newt	0.51.0	1
nfs-utils	1.0.1	2
nhpf	1.42	3
nkf	1.92	11
nmap	3	1
nscd	02/03/02	4.80.6
nss_ldap	198	3
ntp	4.1.1a	9
ntsysv	01/03/06	3
nut-cgi	0.45.4	5
open	1.4	16
openjade	01/03/01	9
openldap12	01/02/13	9
openldap	2.0.27	02/08/00
openldap-clients	2.0.27	02/08/00
openldap-servers	2.0.27	02/08/00
openssh	3.4p1	2
openssh-clients	3.4p1	2
openssh-server	3.4p1	2
openssl095a	0.9.5a	21
openssl096	0.9.6	16.8
openssl	0.9.6b	33
openssl-perl	0.9.6b	33
ORBit	0.5.13	5
ORBit2	02/04/01	1
orbit-python	1.99.0	4
pam	0.75	46.8.0
pam_krb5	1.56	1
pam_smb	01/01/06	5
pango	01/01/01	1
parted	01/04/24	6
passwd	0.67	3
patch	02/05/04	14

Nom	*Version*	*Révision*
pciutils	02/01/10	2
pcre	3.9	5
pdksh	05/02/14	19
perl	05/08/00	55
perl-Crypt-SSLeay	0.45	2
perl-DateManip	5.4	27
perl-Digest-SHA1	2.01	6
perl-File-MMagic	1.15	2
perl-Filter	1.28	9
perl-Frontier-RPC	0.06	33
perl-HTML-Parser	3.26	14
perl-HTML-Tagset	3.03	25
perl-libwww-perl	5.65	2
perl-libxml-enno	1.02	25
perl-libxml-perl	0.07	25
perl-Mail-SpamAssassin	2.53	1
perl-NKF	1.71	7
perl-Parse-Yapp	1.05	26
perl-Text-Kakasi	1.05	2
perl-TimeDate	1.13	2
perl-Time-HiRes	1.2	23
perl-URI	1.21	3
perl-XML-Dumper	0.4	22
perl-XML-Encoding	1.01	20
perl-XML-Grove	0.46alpha	21
perl-XML-Parser	2.31	12
pinfo	0.6.4	7
pnm2ppa	1.04	5
popt	01/07/01	1.8x
portmap	4	46
ppp	02/04/01	7
prelink	0.2.0	8
procinfo	18	5
procmail	3.22	7
procps	2.0.7	25
psmisc	20.2	6
pspell	0.12.2	14
psutils	1.17	17
pwlib	01/03/03	5
pygtk2	1.99.12	7
pygtk2-libglade	1.99.12	7

Nom	*Version*	*Révision*
pyOpenSSL	0.5.0.91	1
python	02/02/01	17
python-optik	1.3	2
pyxf86config	0.3.1	2
quota	3.06	5
raidtools	1.00.2	3.3
rc	01/04/01	0.ximian.5.1
rcd	01/04/03	0.ximian.5.1
rdate	1.2	5
rdist	06/01/05	24
readline41	4.1	14
readline	4.3	3
recode	3.6	6
red-carpet	2.0.1	0.ximian.5.1.1
redhat-config-packages	1.0.1	1
redhat-logos	01/01/06	2
redhat-menus	0.26	1
redhat-release	8	8
redhat-switchmail	0.5.14	1
redhat-switch-printer	0.5.12	1
reiserfs-utils	03/06/02	2
rhmask	1.2	2
rhn-applet	2.0.9	0.8.0.1
rhnlib	1	1
rhpl	0.51	1
rootfiles	7.2	4
rpm404-python	4.0.4	8x.27
rpm	04/01/01	1.8x
rpm-python	04/01/01	1.8x
rp-pppoe	3.4	7
rsh	0.17	10
ruby-docs	01/06/07	10
samba	02/02/07	05/08/00
samba-client	02/02/07	05/08/00
samba-common	02/02/07	05/08/00
sane-backends	1.0.8	5
sash	3.4	14
screen	03/09/11	10
scrollkeeper	0.3.10	7
sed	3.02	13
setserial	2.17	9

Nom	*Version*	*Révision*
setup	02/05/20	1
setuptool	1.1	1
sgml-common	0.6.3	12
shadow-utils	20000902	12.8
shapecfg	02/02/12	10
sh-utils	2.0.12	3
slang	01/04/05	11
slocate	2.6	4
spamassassin	2.53	1
spamassassin-tools	2.53	1
stat	3.3	4
statserial	1.1	30
sudo	01/06/06	1
symlinks	1.2	16
sysklogd	01/04/01	10
syslinux	1.75	3
sysstat	4.0.5	3
SysVinit	2.84	5
talk	0.17	17
tar	1.13.25	8
tcp_wrappers	7.6	23
tcsh	6.12	2
telnet	0.17	23
termcap	11.0.1	13
textutils	2.0.21	5
tftp	0.29	3
time	1.7	19
timeconfig	03/02/09	1
tmpwatch	02/08/04	3
traceroute	1.4a12	6
tree	1.2	20
ttcp	1.12	5
ttfprint	0.9	6
unix2dos	2.2	17
up2date	3.0.7	1
up2date-gnome	3.0.7	1
usbutils	0.9	7
usermode	1.63	1
usermode-gtk	1.63	1
utempter	0.5.2	10
util-linux	2.11r	10

Nom	*Version*	*Révision*
vim-common	6.1	18.8x.1
vim-enhanced	6.1	18.8x.1
vim-minimal	6.1	18.8x.1
vixie-cron	3.0.1	69
vlock	1.3	11
vnc-doc	3.3.3r2	39.2
w3c-libwww	05/04/00	1
w3c-libwww-apps	05/04/00	1
w3m-el-common	01/03/01	1
watanabe-vf	1	8
webalizer	2.01_10	9
wget	01/08/02	5
which	2.14	1
whois	1.0.10	4
whoson	2.02a	1
wireless-tools	25	1
wl-common	02/08/01	8
Wnn6-SDK	1	21
words	2	20
wvdial	1.53	7
xferstats	2.16	3
XFree86-base-fonts	04/02/00	72
XFree86-doc	04/02/00	72
XFree86-font-utils	04/02/00	72
XFree86-libs	04/02/00	72
XFree86-Mesa-libGL	04/02/00	72
XFree86-Mesa-libGLU	04/02/00	72
XFree86-xfs	04/02/00	72
Xft	2	1
xinetd	02/03/11	01/08/00
xml-common	0.6.3	12
ypbind	1.11	2
yp-tools	2.7	3
zisofs-tools	1.0.3	5
zlib	01/01/04	8.8x
zsh	4.0.4	8

23 Annexe F : Script d'installation de poste de travail

Ce code est conçu pour la configuration d'un poste en français. Il utilise le programme *kickstart* de RedHat pour l'installation ainsi que *Red Carpet* de Ximian pour la mise à jour. Il est aussi ajusté pour une configuration matérielle spécifique, notamment vidéo et réseau ; des modifications seront nécessaires pour correspondre à l'environnement particulier de l'administration. Ce code est uniquement une preuve de faisabilité et bien qu'il doive fonctionner, il n'est en aucune manière garanti par **netproject**.

```
lang fr_FR
langsupport fr_FR

# D'autres langues peuvent necessiter d'autres claviers
keyboard uk

# Definition de la souris attachee au client.
mouse --emulthree genericps/2

# A remplacer par l'emplacement du client.
timezone --utc Europe/Paris

# Changer le mot de passe ici
rootpw --iscrypted $1$7QNhVztt$2/DrxHONbGs91.D5k4rx21
reboot
install

# Remplacer 192.168.1.1 par l'adresse IP du serveur.
# Remplacer /mnt/space/RedHat-8.0 par l'emplacement de l'entrepot
# logiciel serveur.
nfs --server 192.168.1.1 --dir /mnt/space/RedHat-8.0
bootloader --location=mbr --append vga=0x305
zerombr yes
clearpart --linux --initlabel
part / --fstype ext3 --size 256 --grow --maxsize 512
part /usr --fstype ext3 --size 927 --grow --maxsize 2048
part swap --recommended
network --bootproto dhcp

# Remplacer 192.168.1.1 par l'adresse IP du serveur.
auth --useshadow --enablemd5 --enableldap --enableldapauth --ldapserver
192.168.1.1 --ldapbasedn dc=netproject,dc=com
firewall -disabled

# A modifier pour correspondre au moniteur utilise.
xconfig --depth 16 --resolution 1024x768 --defaultdesktop=GNOME --
startxonboot --card "S3 Savage (generic)" --videoram 16384

%packages --resolvedeps
librpm404
redhat-artwork
gnome-session
XFree86
gdm
openoffice-libs
openoffice-i18n
libglade
gdk-pixbuf
GConf
compat-libstdc++
indexhtml
libcapplet0
gdk-pixbuf-gnome
libghttp
mozilla-nss
metacity
nautilus
gnome-panel
control-center
XFree86-75dpi-font
```

```
gftp
gedit
emacs
desktop-backgrounds-extra

%pre --interpreter /usr/bin/python
import os, fcntl, CDROM
def eject(file):
     try:
          f = os.open(file, os.O_RDONLY | os.O_NONBLOCK)
     except OSError, (errno, strerror):
          print "%s - OS error(%s): %s" % (file, errno, strerror)
          return
     while 1:
          try:
               fcntl.ioctl(f, CDROM.CDROMEJECT);
          except IOError, (errno, strerror):
               print "%s - I/O error(%s): %s" % (file, errno, strerror)
               if (errno == 16):
                    continue
          break
     os.close(f)
eject("/dev/hda")
eject("/dev/hdb")
eject("/dev/hdc")
eject("/dev/hdd")

%post
/bin/sh << "EOF" >> /root/preboot.log 2>&1
MNT=/mnt/tmp

# Remplacer 192.168.1.1 par l'adresse IP du serveur.
HOST=192.168.1.1
date
mkdir -p $MNT
mount $HOST:/kickstart $MNT -t nfs -o ro
cd $MNT
./postinst.sh preboot
EOF
```

Le code ci-dessus appelle un script appelé postinst.sh qui contient ce qui suit :

```
#!/bin/sh
# description: Script d'installation & mise a jour
# chkconfig: 2345 25 03
# © netproject 2003
# Sean Atkinson <sean@netproject.com>, March 2003
# Adaptation française
# Bernard Choppy <choppy@imaginet.fr>, Novembre 2003

SERVICE=postinst
case $1 in
     start)
          exec $0 postboot >> /root/postboot.log 2>&1
     ;;
     stop)
          exit 0
     ;;
     preboot)
          rpm -U *.rpm
          mkdir /kickstart/
          cp -v diffs.patch gdm_bg.png gdm_logo.png /kickstart/
          cp -v grub_splash.xpm.gz /boot/grub/
          cp -v postinst.sh /etc/rc.d/init.d/$SERVICE
          chkconfig --add $SERVICE
          exit 0
     ;;
     postboot)
     ;;
     *)
```

```
          echo "Usage: $0 start|stop|preboot|postboot"
          exit 1
     ;;
esac

# Remplacer 192.168.1.1 par l'adresse IP du serveur.
# Verifier les details d'emplacement et de version
# de la distribution
HOST=192.168.1.1
RCD=http://$HOST/rcd/
REL=redhat-80-i386

date

# Les commandes rpm ci-dessous correspondent a des versions
# specifiques - remplacer si necessaire.
rpm -U $RCD/redcarpet/$REL/rcd-1.2.1-1.ximian.3.i386.rpm
cat << EOF >> /etc/ximian/rcd.conf
[Network]
host=$RCD

[Cache]
enabled=false
EOF
service rcd start
rpm -U $RCD/redcarpet/$REL/rc-1.2.1-1.ximian.1.i386.rpm
until rc ping
     do
     sleep 1
done
rc subscribe $REL redcarpet ximian-evolution
service rcd restart
rpm -e kernel-pcmcia-cs
rpm -e kudzu
rpm -e comps
rpm -e authconfig
until rc ping
     do
     sleep 1
done
rc update --no-confirmation
for PKG in galeon evolution xpdf vnc vnc-doc openoffice
     do
     rc install --no-confirmation $PKG
done

# Cette commande suppose une version specifique de Java
ln -s /usr/java/j2re1.4.1_02/plugin/i386/ns610/libjavaplugin_oji.so \
     /usr/lib/mozilla-1.0.1/plugins/
patch -p0 < /kickstart/diffs.patch
chkconfig --del $SERVICE
chkconfig rcd off
for DIR in /tmp /var/tmp
     do
     rm -rf $DIR
     ln -s /dev/shm $DIR
done
reboot
```

Le code ci-dessus utilise le contenu d'un fichier appelé diffs.patch qui contient :

```
--- gdm.conf 2003-03-25 14:01:20.000000000 +0000
+++ /etc/X11/gdm/gdm.conf 2003-03-25 14:02:38.000000000 +0000
@@ -21,7 +21,7 @@
 Chooser=/usr/bin/gdmchooser
 DefaultPath=/usr/local/bin:/usr/bin:/bin:/usr/X11R6/bin
 DisplayInitDir=/etc/X11/gdm/Init
-Greeter=/usr/bin/gdmgreeter
+Greeter=/usr/bin/gdmlogin
 # Decommenter pour une invite traditionnelle
```

```
 #Greeter=/usr/bin/gdmlogin --disable-sound --disable-crash-dialog
 RemoteGreeter=/usr/bin/gdmlogin
@@ -99,7 +99,7 @@
 GlobalFaceDir=/usr/share/faces/
 Icon=/usr/share/pixmaps/gdm.xpm
 LocaleFile=/etc/X11/gdm/locale.alias
-Logo=

# Cela suppose que le fichier png file existe a cet endroit
# Modifier si necessaire.
+Logo=/kickstart/gdm_logo.png
 ## Beau logo RH pour la ligne precedente : /
usr/share/pixmaps/redhat/shadowman-200.png
 Quiver=true
 SystemMenu=true
@@ -114,8 +114,8 @@
 PositionY=0
 XineramaScreen=0
 # Le type peut etre 0=aucun, 1=image, 2=couleur
-BackgroundType=0
-BackgroundImage=
+BackgroundType=1

# Cela suppose que le fichier png existe a cet endroit
# Modifier si necessaire.
+BackgroundImage=/kickstart/gdm_bg.png
 BackgroundScaleToFit=true
 BackgroundColor=#27408b
 BackgroundRemoteOnlyColor=true
--- grub.conf 2003-03-25 14:01:20.000000000 +0000
+++ /boot/grub/grub.conf 2003-03-25 14:01:20.000000000 +0000
@@ -9,7 +9,7 @@
 #boot=/dev/hda
 default=0
 timeout=10
-splashimage=(hd0,0)/boot/grub/splash.xpm.gz

# Cela suppose que le fichier xpm.gz existe a cet endroit
# Modifier si necessaire.
+splashimage=(hd0,0)/boot/grub/grub_splash.xpm.gz

# Cela suppose une version particuliere du noyau (2.4.18-27.8.0).
# Modifier toutes les references si necessaire.
 title Red Hat Linux (2.4.18-27.8.0)
      root (hd0,0)
      kernel /boot/vmlinuz-2.4.18-27.8.0 ro root=LABEL=vga=0x305
--- fstab 2003-03-27 10:52:12.000000000 +0000
+++ /etc/fstab 2003-03-27 10:54:01.000000000 +0000
@@ -1,8 +1,9 @@
 LABEL=/         /               ext3       defaults        1 1
 none      /dev/pts        devpts     gid=5,mode=620  0 0
 none      /proc proc      defaults                   0 0
-none      /dev/shm        tmpfs      defaults        0 0
-LABEL=/usr     /usr            ext3       defaults        1 2
+none      /dev/shm        tmpfs      defaults,noexec 0 0
+LABEL=/usr     /usr            ext3       defaults,ro     1 2
 /dev/hda3      swap            swap       defaults        0 0
 /dev/cdrom     /mnt/cdrom      iso9660    noauto,owner,kudzu,ro 0 0
 /dev/fd0       /mnt/floppy     auto       noauto,owner,kudzu    0 0

# Remplacer 192.168.1.1 par l'adresse IP du serveur NFS.
+192.168.1.1:/home   /home      nfs        defaults,noexec 0 0
```

24 Annexe G : Glossaire

ACL	Liste de contrôle d'accès (Access Control List) attachée à un objet tel qu'un fichier. Elle est constituée d'expressions de contrôle, chacune autorisant ou interdisant telle ou telle possibilité à un utilisateur ou à un groupe d'utilisateurs.
Administration	Une administration publique européenne.
Administrateur	La gestion informatique et télécommunications d'une administration.
API	Interface programmatique (Application Programming Interface) - méthode spécifique prescrite par un logiciel d'ordinateur par lequel un programmeur peut faire des demandes à celui ci.
Appliquette Java	Mini-logiciel qu'un navigateur Java ou ActiveX télécharge et utilise automatiquement. Il peut ajouter un support sophistiqué des pages web, bien au-delà de la programmation comme DHTML ou JavaScript.
ASP	Active Server Pages - méthode Microsoft d'inclusion dans des pages HTML de scripts traités par le serveur avant expédition à l'utilisateur. D'une certaine manière, une ASP est similaire à une inclusion côté serveur ou une application CGI en ce que ces trois méthodes impliquent des programmes serveurs.
Assistant personnel	Ordinateur portable de poche (PDA).
BDC	Contrôleur de domaine secondaire (Backup Domain Controller) - rôle qui peut être assigné à un serveur dans un réseau d'ordinateurs utilisant le système d'exploitation *MS-Windows NT*. L'idée de domaine sert à gérer l'accès à un ensemble de ressources réseau (applications, imprimantes, etc.) pour un groupe d'utilisateurs. Chaque utilisateur doit seulement se connecter au domaine pour avoir accès à toutes les ressources qui peuvent être situées sur différents serveurs du réseau. Un serveur appelé PDC gère la base d'utilisateurs du domaine. Un ou plusieurs serveurs sont désignés comme BDC. Périodiquement, le PDC transmet des copies de la base aux BDC. Un BDC permet d'équilibrer la charge de travail si le réseau est chargé et peut être promu PDC si celui-ci est défaillant.
Binaire	Le logiciel est habituellement écrit dans un langage appelé code source, aisément compréhensible par des humains. Ce code est converti en une forme compréhensible directement par le processeur de l'ordinateur. Ce code est appelé binaire car il consiste en une série de zéros et de uns. C'est la forme sous laquelle l'essentiel du code propriétaire est fourni et il est très difficile de le reconvertir vers une forme humainement compréhensible. La détention du code source permet la modification du logiciel ainsi que la compréhension de ce qu'il réalise.
Carte à puce	(Smart Card) - Carte en plastique incluant un composant informatique. La carte est utilisée pour effectuer des opérations nécessitant les données qui y sont stockées.
CGI[5]	Common Gateway Interface : standard d'exécution de programmes sur un serveur web.
CIL	Common Intermediate Language : code intermédiaire indépendant du compilateur et de la machine exécuté par un environnement commun de langage (CLR - Common Language Runtime). Ce code peut être obtenu par nombre de langages compilés y compris C# et C. Le CIL et le CLR sont des constituants de l'infrastructure commune de langage (CLI - Common Language Infrastructure).
Code bêta	Durant la phase de réalisation du logiciel, celui-ci passe par différents états avant d'être considéré comme suffisamment fiable pour une

5 Ajout du traducteur.

	utilisation en production. Le premier de ces états est appelé alpha et le second, bêta. Le code bêta est ainsi considéré comme substantiellement correct mais devant être utilisé avec précaution car il peut encore contenir des erreurs significatives.
Daemon	(Disk And Extension MONitor) Programme système qui réside en arrière-plan jusqu'à son invocation par un autre processus ou un événement pour réaliser sa tâche.
DHCP	Protocole de configuration dynamique de machine (Dynamic Host Configuration Protocol) permettant la gestion centralisée et l'automatisation de l'assignation des adresses IP sur un réseau.
Distribution	Pour le logiciel OSS tel que GNU/Linux, des entreprises telles que RedHat se spécialisent dans l'agglomération de composants depuis de nombreuses sources vers un seul paquetage qui peut être distribué simplement aux utilisateurs par téléchargement unique ou sur un ensemble de CD.
DNS	Serveur/service de noms de domaine (Domain Name Server/Service) utilisé pour convertir les noms de machines en adresses numériques et vice-versa.
Équilibrage de charge	Il consiste à diviser la quantité de travail d'un ordinateur entre plusieurs processeurs ou ordinateurs afin d'en réaliser plus dans le même temps et, en général, de servir plus vite les utilisateurs. L'équilibrage de charge peut être implanté avec du matériel, du logiciel ou une combinaison des deux. Typiquement, c'est la principale motivation de la constitution d'une grappe de serveurs.
FTP	Protocole de transfert de fichiers (File Transfer Protocol) indépendant du système entre machines connectées par TCP/IP. Il garantit un transfert correct, même si des erreurs se produisent durant la transmission.
Gestionnaire de fenêtres	Dans un environnement graphique moderne, l'utilisateur dispose d'une série de fenêtres représentant des processus distincts. Ceux-ci peuvent réaliser des tâches très différentes et leur affichage s'effectue simultanément à l'écran. La gestion de ces fenêtres est le rôle de ce composant. Il garde trace de l'ordre d'empilement des fenêtres, permet de basculer entre celles-ci ainsi que d'en créer ou d'en supprimer. Il en contrôle aussi l'aspect et les fonctions de contrôle.
Gestionnaire de session	Lorsqu'un utilisateur se connecte à un ordinateur, il crée une session qui consiste en un environnement rempli d'informations qui lui sont personnelles ainsi que d'une série de processus. Le gestionnaire permet à l'utilisateur de modifier cet environnement et peut aussi sauvegarder celui-ci afin que l'utilisateur puisse le retrouver intact lors de sa prochaine connexion.
Gopher	Système de publication d'informations hypertexte antérieur au web.
GPL	Licence publique générale GNU (GNU General Public licence)[6].
Hachage	Algorithme/identifiant de taille réduite représentant quelque chose de plus compliqué. Les hachages sont produits par des fonctions mathématiques non réversibles. Ils sont utilisés dans les SGBD et dans les systèmes de sécurité et de chiffrement.
HTTP	Protocole de transfert hyper-texte (Hyper-Text Transfer Protocol) constitué d'un ensemble de règles d'échange de fichiers (textes, images, sons, vidéos et autres éléments multimédia) sur le web. Au sein de la pile de protocoles TCP/IP (qui constitue la base de l'échange d'informations sur Internet), HTTP est un protocole de niveau application.

6 NdT : Richard Stallman, auteur de la licence GPL, a interdit toute traduction officielle mais des versions françaises indicatives sont souvent fournies avec les logiciels et manuels traduits.

I.G.C.[7]	Infrastructure de gestion de clefs permettant aux utilisateurs d'un réseau public non sécurisé tel qu'Internet d'échanger en sécurité des informations et des transactions financières par l'utilisation d'une paire de clefs de chiffrement (clefs publique et privée) obtenues et partagées par une autorité de confiance. L'I.G.C. fournit des certificats numériques qui peuvent identifier un individu ou une organisation et peut stocker et si nécessaire révoquer ceux-ci.
I.H.M.	Interface homme-machine (GUI - Graphical User Interface).
JDBC	Java Database Connectivity - Spécification d'interface programmatique (API) pour connecter les programmes écrits en Java aux principaux SGBD. Elle permet d'encoder les requêtes d'accès en SQL qui est transmis au moteur de base de données qui renvoie les résultats par une interface similaire.
LDAP	Protocole léger d'accès aux répertoires (Lightweight Directory Access Protocol) qui permet à quiconque de localiser des organisations, individus ou ressources (fichiers, périphériques) sur un réseau, que celui-ci soit Internet ou un intranet. LDAP est une version « légère » (quantité de code réduite) de DAP (Directory Access Protocol) qui est un constituant de X.500, un standard de services réseau de répertoires.
LGPL	GPL réduite ou GPL de bibliothèques (Lesser/Library GPL).
Utilisateur concurrent	Forme de licence d'une application calculée sur la base du nombre maximal d'utilisateurs simultanés de celle-ci.
Utilisateur potentiel	Forme de licence d'une application calculée sur la base du nombre total d'utilisateurs ayant accès à l'application.
Logiciel libre	La définition se trouve à http://www.gnu.org/philosophy/free-sw.html.
Logiciel Open Source	La définition se trouve à http://www.opensource.org/docs/definition_plain.html.
MAA	Agent d'accès au courriel (Mail Access Agent) - Terme utilisé dans ce document pour décrire le composant serveur qui gère l'accès aux boîtes par les MUA. Les serveurs POP et IMAP en sont deux exemples.
MDA	Agent de remise de courriel (Mail Delivery Agent) - C'est le composant en charge de la décision de remise en local (à un MDA ou vers un stockage direct dans une boîte) ou de transfert à un autre MTA.
MUA	Agent utilisateur de courriel (Mail User Agent) - C'est le composant client de courriel qui relève celui-ci de la boîte et le présente à l'utilisateur. Il permet à ce dernier de créer de nouveaux messages et de les envoyer à un MTA pour transmission. Les MUA est souvent associé à une interface graphique.
.NET	Ensemble de technologies Microsoft de connexion d'informations, personnes, systèmes et périphériques. Il est fondé sur des services web qui sont de petites applications capables de se connecter entre elles ainsi qu'à d'autres plus importantes sur Internet. Le projet OSS Mono est une implantation du cadre de développement .NET.
NFS	Service de fichier réseau (Network File Service) - C'est un protocole couramment utilisé par les systèmes Unix permettant l'accès à des fichiers stockés sur des systèmes distants comme s'ils étaient locaux.
Noyau	Le coeur d'un système d'exploitation qui assure les tâches de base telles que l'allocation de mémoire, les entrées/sorties, l'allocation de processus, la sécurité et l'accès utilisateur.
ODBC	Open Database Connectivity - Standard ouvert d'interface programmatique d'accès à une base de données. L'utilisation de requêtes

7 NdT : Traduction préconisée par l'A.T.I.C.A. du terme PKI.

	ODBC dans un programme permet d'accéder à des enregistrements de nombreuses bases, notamment *MS-Access*, *DB2*, *MS-Excel*, *dBase* et texte. En plus du logiciel ODBC, un module ou pilote spécifique est nécessaire pour chaque type de base.
PDC	Contrôleur de domaine primaire (Primary Domain Controller) - voir BDC.
PHP	Pré-processeur hyper-texte : langage de script et interpréteur librement disponible et utilisé en particulier sur les serveurs web GNU/Linux. PHP est une alternative à la technologie ASP de Microsoft. Comme celle-ci, le script PHP est inclus dans une page web au sein du HTML. Avant transfert de la page à l'utilisateur demandeur, le serveur web appelle PHP pour interpréter et exécuter les opérations décrites par le script PHP.
PKI	Voir I.G.C.
Procédure stockée	Ensemble d'ordres SQL regroupés sous un nom et stocké sous forme compilée dans la base pour permettre son utilisation par plusieurs programmes.
Protocole	Ensemble de règles spécialisé pour la communication au sein d'une connexion. Des protocoles existent à différents niveaux de la connexion, pour chaque couche fonctionnelle de la communication. Les deux extrémités doivent reconnaître et observer le protocole. Les protocoles sont souvent décrits par un standard industriel ou international.
Relais ouvert	Un relais ouvert (parfois appelé relais non sécurisé ou relais tiers) est un serveur de courriel SMTP qui autorise le relais de messages issus de tiers. Le traitement de courriel dont ni l'émetteur, ni le destinataire ne sont locaux rend possible pour un émetteur peu scrupuleux de router de grands volumes de courriel non sollicité (spam). En effet, le propriétaire du serveur - habituellement non averti du problème - offre des ressources réseau et informatiques à l'émetteur. En plus des coûts induits par l'intrusion d'un pirate, une organisation peut aussi souffrir de plantages système, dommages aux équipements et perte d'informations.
Serveur mandataire	(Proxy Server) - Serveur agissant comme intermédiaire entre un utilisateur de poste de travail et Internet de telle manière que l'entreprise puisse assurer la sécurité, le contrôle administratif et un service de cache. Un serveur mandataire est associé à ou inclus dans un serveur passerelle qui sépare le réseau interne du réseau externe et un serveur pare-feu qui protège le réseau interne des intrusions depuis l'extérieur.
Schéma	Organisation ou structure d'une base de données. L'activité de modélisation de données aboutit à un schéma.
Servlet Java	Programme Java qui s'exécute en tant que partie d'un service réseau, typiquement sur un serveur HTTP, et répond aux demandes de clients. L'utilisation la plus courante des servlets est l'extension d'un serveur web par la génération dynamique de contenu. Par exemple, un client peut avoir besoin d'informations stockées dans une base de données ; on peut écrire un servlet qui reçoit la demande, traite les données souhaitées par le client et lui renvoie le résultat.
Session X	Lorsqu'un utilisateur se connecte sur un ordinateur et exécute des programmes selon le protocole X, il crée une session X.
SGBD	Système de gestion de bases de données - programme permettant à des utilisateurs de créer et manipuler des données d'une base de données. Le SGBD interprète les requêtes des utilisateurs et d'autres programmes afin que ceux-ci n'aient pas à connaître l'emplacement physique des données sur le support de stockage ni, sur un système multi-utilisateur, qui d'autre accède aux données.

SMB	Bloc de message serveur (Server message Block) - C'est le protocole utilisé par les réseaux *MS-Windows* pour l'accès aux ressources telles que les fichiers partagés.
SMS	Service de messages courts (Short Message Service) - service d'envoi de messages (les texto) de 160 ou 224 caractères au plus vers le téléphones mobiles GSM (Global System for Mobile - Système global pour mobiles).
Source	Voir Binaire.
SQL	Langage de requêtes structurées (Structured Query Language) - C'est un langage standard interactif et de programmation d'accès et de mise à jour des bases de données. Quoique SQL soit un standard ANSI et ISO, de nombreux SGBD/R ajoutent des extensions propriétaires. Les requêtes prennent la forme d'un langage de commande permettant la sélection, l'insertion, la mise à jour, la recherche d'informations, etc. C'est aussi une interface programmatique.
SSL	Couche de connexions sécurisée (Secure Sockets Layer) - C'est un protocole répandu de sécurisation de la transmission de messages sur Internet. Il a récemment été amélioré par TLS, fondé sur lui. Il utilise une couche de programme située entre HTTP et TCP. Il est inclus dans les navigateurs Microsoft et Netscape ainsi que dans de nombreux produits serveur web.
Terminal X	Terminal spécifiquement conçu pour exécuter un serveur X permettant aux utilisateurs l'affichage selon le protocole X des résultats de programmes qui s'exécutent sur d'autres ordinateurs du réseau.
Trigger	Ensemble d'ordres SQL à déclenchement automatique lors de l'apparition d'un événement spécifique tel que la modification d'une donnée.
TLS	Sécurité de la couche transport (Transport Layer Security) - Couche de services de chiffrement et d'authentification qui peut être négociée durant la phase de démarrage de nombreux protocoles Internet (SMTP, LDAP, IMAP, POP3...). TLS est dérivé de SSL et utilise les mêmes certificats sans nécessiter l'attribution d'un port spécifique pour chaque service.
VMS	Système d'exploitation développé par Digital Equipment Corporation (DEC) pour les mini-ordinateurs VAX, puis porté sur les systèmes 64 bits Alpha. Un des principaux concepteurs de VMS est à l'origine du noyau de *MS-Windows NT*.
WebDAV	Création et gestion de version distribuées du web (Web Distributed Authoring and Versioning) - Standard de l'IETF de création web collaborative ; il s'agit d'un ensemble d'extensions à HTTP qui facilite la modification et la gestion de fichiers collaboratives par Internet entre des utilisateurs distants.
XML	Langage de marquage extensible (Extensible Markup Language) représentant un moyen souple de créer des formats d'information communs ainsi que de partager aussi bien le format que les données sur le web, les intranets et partout ailleurs. XML est une recommandation formelle du W3C similaire au langage des pages web (HTML).