

Table des matières

I	Introduction	4
II	Le Crédit Mutuel Arkéa	6
1	Secteur d'activité	6
2	Historique et chiffres clés	7
3	Structure et produits/services proposés	7
4	L'équipe Systèmes	9
III	Le contexte du stage	11
5	Le début de la virtualisation de postes de travail au CMA	11
6	La gestion du parc de Machines Virtuelles	11
6.1	L'environnement VMware View	11
6.2	Les pools de VMs	12
IV	L'encapsulation d'applications avec VMware ThinApp	15
7	ThinApp	15
7.1	La virtualisation d'application selon Citrix ou VMware	15
7.2	Les fonctionnalités intéressantes de ThinApp	17
7.2.1	AppLink	17
7.2.2	ThinApp Converter	18
8	Création d'une bulle applicative	19
8.1	Prescan	19
8.2	Installation	19
8.3	Postscan	20
8.4	Choix du Format	20
8.5	Packaging	20
8.6	Déploiement	21

9	Phase de test	23
9.1	Capture d'une application simple	23
9.2	Exécution de plusieurs versions du même logiciel	23
9.3	Capture d'un client lourd	24
9.4	Déploiement d'une application sur des VMs	25
9.5	Résultats des tests	25
V	VMware ThinApp Automatisé	26
10	Outils utilisés pour développer le logiciel	26
10.1	Le C++	26
10.2	Qt	27
10.3	Fonctions Windows	27
11	Développement de l'application	28
11.1	Prise en main de Qt	28
11.2	L'interface	28
11.3	Les difficultés rencontrées	30
11.3.1	Utilisation de ThinApp sans l'interface	30
11.3.2	Utilisation des API Windows	30
VI	Conclusion	31
VII	Bibliographie	32
VIII	Annexes	33
12	Bilan Personnel	33
13	WYSE	34

Table des figures

1	L'implantation du Crédit Mutuel Arkéa	6
2	Architecture de marque du Crédit Mutuel Arkéa	8
3	Organigramme de la Direction Technique Informatique	9
4	Schéma du fonctionnement de VMware View	12
5	Schéma d'une architecture utilisant les Clones Liés	13
6	Schéma des interactions entre l'utilisateur et le serveur Citrix	16
7	La bulle ThinApp	17
8	Les étapes permettant la capture d'une application	19
9	Modes de déploiement d'une application ThinApp	21
10	Neuf navigateurs ouverts simultanément	24
11	Schéma du processus de capture	26
12	Ébauche d'IHM	28
13	L'interface graphique finale	29
14	Photo d'un client léger WYSE	34

Première partie

Introduction

Le stage s'est déroulé au crédit mutuel Arkea. Cette entreprise est un groupe de bancassurance composé de trois pôles historiques, le crédit mutuel de Bretagne, le crédit mutuel du Sud Ouest et le crédit mutuel du Massif Central. À ces trois blocs viennent se rajouter une vingtaine de filiales qui ont été soit annexées soit créées par la suite.

L'objectif du stage tient en deux points. Tout d'abord la réalisation d'une étude de solution de virtualisation d'application afin d'envisager ou non l'inclusion de cette solution dans le système d'information. Ensuite la réalisation d'un logiciel permettant d'automatiser le processus de virtualisation d'une application, afin de pouvoir faire en sorte que la capture se fasse en tâche de fond.

Ce rapport est constitué de 4 parties. La première est une présentation de l'entreprise et de l'équipe dans laquelle s'est déroulé le stage. La deuxième présente l'outil de virtualisation de poste de travail VMware View. La troisième partie de ce rapport est consacrée à l'étude de la solution de virtualisation d'application VMware Thinapp. La quatrième partie présente le développement du logiciel.

Remerciements

Je remercie :

LE GOFF Philippe : Pour m'avoir permis de réaliser mon stage au sein de son service et avoir toujours été à l'écoute.

TOMASINO Gilles : Pour m'avoir encadré durant ce stage et m'avoir expliqué les rouages de l'équipe systèmes.

DUBOIS Mikael : Pour m'avoir encadré durant ce stage et m'avoir expliqué le fonctionnement de VMware View.

MORICE Yann : Pour sa patience et son aide précieuse lors de la réalisation du logiciel.

L'équipe Systèmes : Pour m'avoir accueilli pendant la durée du stage.

Deuxième partie

Le Crédit Mutuel Arkéa

1 Secteur d'activité

Arkéa (figure 1) est un groupe coopératif et mutualiste, réunissant les trois fédérations historiques, de Crédit Mutuel de Bretagne (CMB), du Sud- Ouest (CMSO) et du Massif Central (CMMC) autour de la caisse de Crédit Mutuel.

Cet ensemble contrôle la Compagnie Financière du Crédit Mutuel, banque de marché et société-holding rassemblant une vingtaine de filiales spécialisées couvrant les domaines de l'immobilier, du tourisme et du voyage d'affaires pour les particuliers et les professionnels.

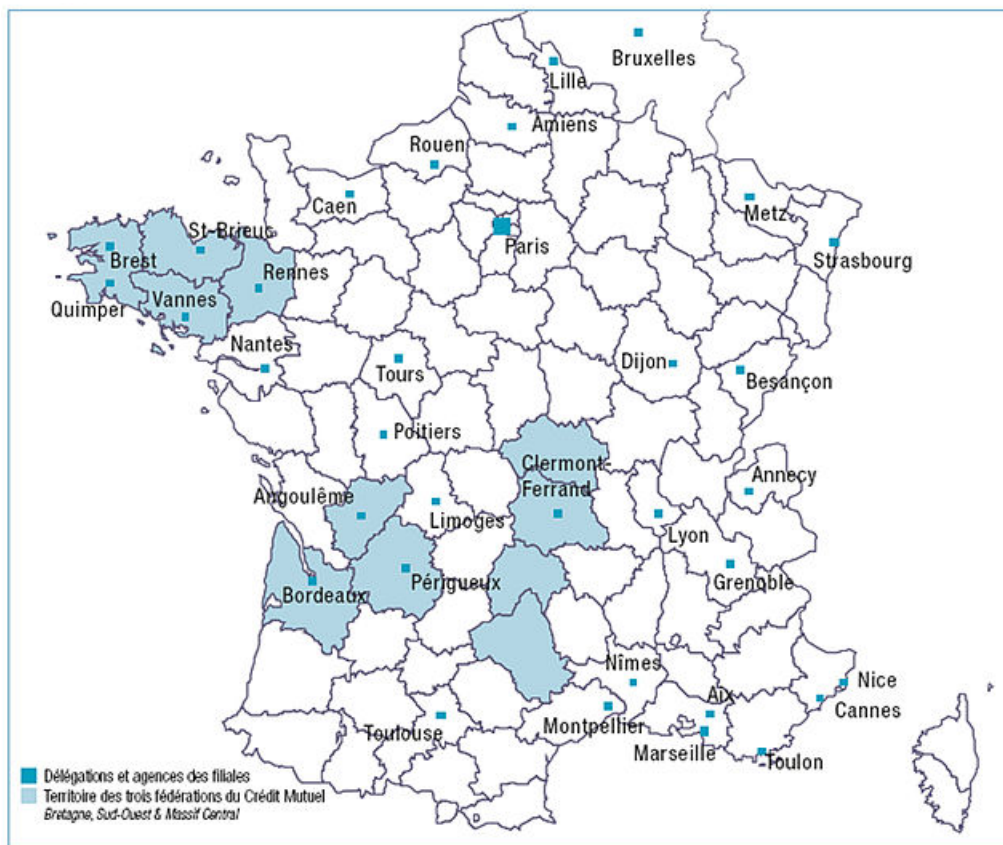


FIGURE 1 – L'implantation du Crédit Mutuel Arkéa

2 Historique et chiffres clés

- **1979** : Cinq caisses bretonnes du Crédit Mutuel se regroupent et forment le Crédit Mutuel de Bretagne.
- **1980** : Création de filiales spécialisées.
- **1996** : La CMB et le CMSO se rassemblent et créent la CICM.
- **2002** : Le CMMC rejoint la CICM, qui devient le Crédit Mutuel Arkéa, et absorbe la Compagnie Financière du Crédit Mutuel (CFCM).

Le Crédit Mutuel Arkéa en chiffre au 31/12/2012, c'est 3,2 millions de sociétaires et clients, 9000 salariés, 91 milliards d'euros de total de bilan (2012), 3700 administrateurs et 595 points d'accueil. Ancré sur ses territoires, le groupe conjugue l'efficacité économique et les valeurs qui font sa force.

3 Structure et produits/services proposés

Banque universelle, ouverte à tous, le Crédit Mutuel Arkéa est à la fois fabricant et distributeur de ses produits et services. Le groupe s'appuie sur une organisation simple, bâtie autour des trois fédérations du Crédit Mutuel vues précédemment formant le pôle Réseaux.

Aux côtés de ces fédérations, quatre « pôles métiers » rassemblent les activités du Crédit Mutuel Arkéa :

- Le pôle **Entreprise et Institutionnels** recouvre l'ensemble des expertises pour le financement et le développement des entreprises.
- Le pôle **Réseaux et services spécialisés** développe les activités de vente de produits et services de banque et d'assurance aux particuliers.
- Le pôle **Produits** a vocation à renforcer la présence du groupe dans le domaine des produits bancaire, financiers et d'assurance, par une offre innovante et compétitive.
- Le pôle **Innovation et Opérations** est chargé de mettre à la disposition des entités du groupe des prestations et d'élargir l'offre de moyens de paiement et les prestations de titres.

Trois fonctions support, **Ressources Humaines**, **Finances** et **Risques et Moyens**, complètent l'organisation du Crédit Mutuel Arkéa et apportent leurs expertises aux pôles métiers.

La figure 2 ci-dessous représente schématiquement La répartition des différents pôles qui constituent le CMA.



FIGURE 2 – Architecture de marque du Crédit Mutuel Arkéa

C'est dans le pôle Innovation et Opérations que se trouve la Direction Technique Informatique (DTI).

La DTI (figure 3) regroupe des départements qui ont pour missions de :

- Accompagner et mettre en œuvre les grands projets d'évolution de la plateforme (Systèmes, BDD, Réseaux, Déploiement logiciel, Automatisation, Gestion des ressources)
- Assister les études informatiques (Équipes méthodes)
- Assister les utilisateurs via l'Assistance de Vannes
- Maintenir le parc matériel du groupe (DataCenters, postes de travail, serveurs, ...)
- Délivrer aux clients internes et externes des filiales des prestations informatiques
- Assurer la continuité, la sécurité, la disponibilité, la performance, la fiabilité, la confidentialité des Systèmes d'Information (SI)

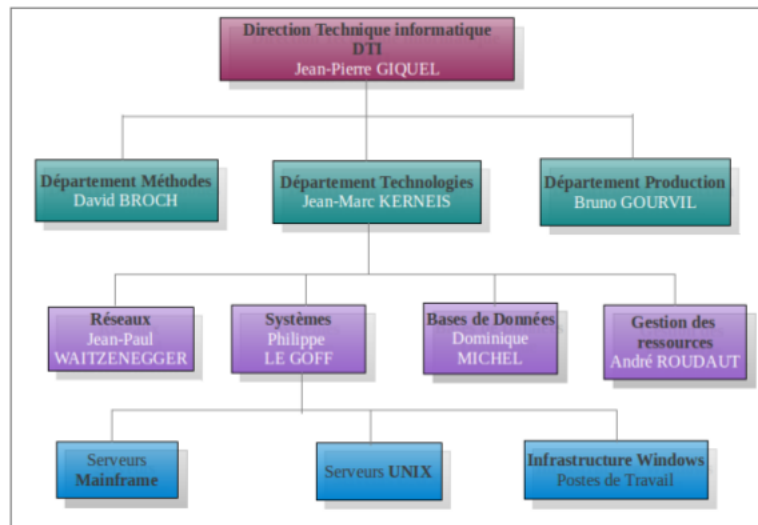


FIGURE 3 – Organigramme de la Direction Technique Informatique

Les équipes fonctionnent en niveaux pour le support utilisateurs :

Niveau 1 : Le front office, il s'agit de l'Assistance de Vannes que tout utilisateur du groupe contacte. Elle effectue un premier diagnostic et tente de résoudre la panne. Si le problème persiste ou qu'il dépasse son niveau de compétence, il contacte une équipe de niveau 2.

Niveau 2 : La back office, il est en charge du diagnostic de la panne. Il entreprend ensuite de dépanner le poste utilisateur ou le serveur qui présente un dysfonctionnement.

Niveau 3 : L'appel à ce niveau se justifie quand il s'agit d'un cas qui requière une expertise technique ou qui concerne un applicatif développé par l'un de ses ingénieurs.

4 L'équipe Systèmes

Le stage s'est déroulé au sein de l'équipe système, celle-ci est divisée en trois sous-équipes :

- **Équipe Poste de Travail :**
 - Gestion de l'architecture de l'Active Directory
 - Gestion des masters des postes de travail
 - Mise en place de solutions de virtualisation des postes

- **Équipe Unix :**
 - Gestion des masters des OS¹ Linux, Solaris et AIX
 - Design de l'architecture serveur et l'architecture stockage
 - Outils de sauvegarde
 - Développement d'outils de supervision du Système d'Information (SI), hors infrastructure réseau
- **Équipe Mainframe :**
 - Gestion de l'OS du mainframe (zOS) ainsi que de ses composants

L'objectif premier du stage était de réaliser l'étude d'une solution de virtualisation d'application, ainsi que de tester concrètement celle-ci afin de déterminer si son intégration dans le SI pourrait s'avérer utile pour l'entreprise. Dans un second temps il a fallu s'intéresser à la réalisation d'un logiciel permettant l'automatisation du processus de création d'une application virtuelle.

1. Operating System

Troisième partie

Le contexte du stage

5 Le début de la virtualisation de postes de travail au CMA

Depuis deux ans le Crédit Mutuel Arkéa s'est lancé dans la virtualisation de poste de travail. Cette décision est née de la volonté de fournir à des sociétés clientes extérieures un ensemble d'outils bancaires développés au sein du CMB. on peut notamment citer le cas de certaines compagnies d'assurance, qui voulant se lancer dans la banque mais n'ayant aucun outils bancaire à leurs disposition, ont choisi de louer les services du CMB afin d'obtenir tout le panel d'outils dont ils avaient besoin.

Les raisons qui ont orienté le choix vers l'utilisation d'une infrastructure virtuelle comme vecteur de déploiement de ces outils bancaire, plutôt qu'une infrastructure physique, sont au nombre de quatre :

- Faciliter l'administration des postes de travail en centralisant le hardware, ce qui permet une résolution plus rapide des problèmes car il n'est alors plus nécessaire d'envoyer quelqu'un sur site.
- Homogénéiser le parc informatique.
- Mieux sécuriser l'accès au réseau interne du CM.
- Faciliter le déploiement des outils car l'utilisation de VMs permet d'être indépendant du matériel client et d'être certain de la compatibilité entre les logiciels et l'OS.

Cette infrastructure regroupe aux alentours de 500 VMs et elle est sous la responsabilité de l'équipe Poste de Travail.

6 La gestion du parc de Machines Virtuelles

6.1 L'environnement VMware View

Le parc de VMs du Crédit Mutuel Arkéa est basé sur une architecture VMware View. Ce système de virtualisation de postes de travail est du type Server Hosted VDI². Cela signifie que à l'inverse d'une virtual box qui ferait, par exemple, fonctionner un système d'exploitation Linux virtuel à partir du système d'exploitation Windows installé par défaut sur une machine, ici il est question de faire fonctionner un OS sur un serveur et d'envoyer les données vers le client qui en fait la demande.

2. Virtual Device Interface

Ce client peut être un PC classique disposant du logiciel Client View, mais aussi un client léger comme un WYSE (cf Annexe p.35), qui agit exactement comme un pc mais ne permet pas d'installer le moindre OS, ou encore un client ultra-léger comme une tablette. C'est ce que montre la figure 4.



FIGURE 4 – Schéma du fonctionnement de VMware View

L'environnement VMware View est une plate-forme permettant de simplifier et d'automatiser la gestion de poste de travail. Il permet notamment d'encapsuler le système d'exploitation, les applications, les profils ainsi que les données personnelles en les isolant en couches distinctes. Ce qui permet une gestion plus efficace des postes de travail et un assemblage de façon dynamique à la demande.

6.2 Les pools de VMs

Au Crédit Mutuel Arkéa, l'architecture VMware View choisie pour la gestion des pools est basée sur l'utilisation de clones liés (ou linked clones en anglais), c'est ce que représente la figure 5.

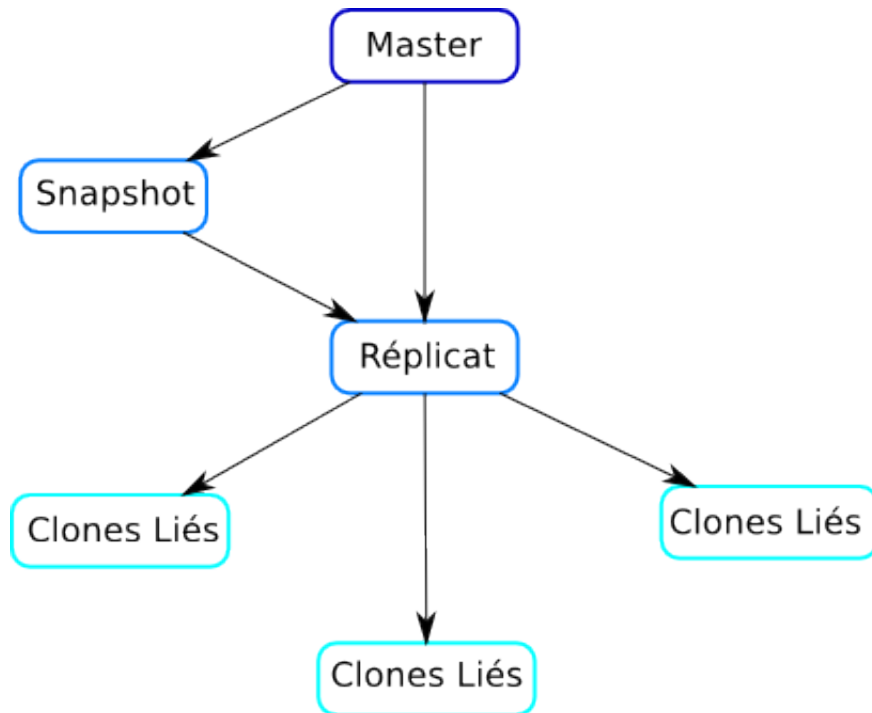


FIGURE 5 – Schéma d’une architecture utilisant les Clones Liés

Le premier élément à mettre en place dans une telle structure est le **master** (également appelé VM parent). Le master est une machine virtuelle créée sur un serveur et dans laquelle on va installer tous les logiciels nécessaires aux futurs utilisateurs du pool. La VM parent ainsi créée sera tout aussi volumineuse qu’une machine physique.

Une fois le master créé on réalise ce que l’on appelle un **snapshot**, celui-ci correspond à un fichier dans lequel seront sauvegardées toutes les modifications apportées au master après la création du snapshot. Il peut avoir plusieurs utilités :

- Permettre de revenir à l’état de la machine avant la création du snapshot et donc, dans le cas présent, effacer toutes les modifications effectuées sur le Master.
- Permettre de modifier le réplicat (nous verrons à quoi cela correspond dans le paragraphe suivant) sans forcément modifier le Master
- Permettre de tester les mises à jour apportées au master avant une véritable mise à jour de celui-ci.

Une fois ces deux éléments créés, il est maintenant possible de créer le **réplicat**. Un réplicat est une copie exacte de la somme du master et de son snapshot. Il prend

donc en théorie au moins autant de place que la VM parent. L'intérêt principale de celui-ci est qu'il permet à l'administrateur de faire des manipulations sur le master sans pour autant que cela ait un impact sur le fonctionnement des VMs utilisateur. Il est important de noter que seul l'administrateur du pool a la possibilité de modifier le Master ou le réplicat (et par extension le snapshot).

Enfin une fois toutes ces étapes réalisées il est possible de créer les **Clones Liés**. Ce sont les VMs que les utilisateur finaux voient affichées sur leurs écrans. En réalité il ne s'agit pas vraiment de machines virtuelles car elle ne sont que des coquilles vides alimentées par le réplicat. En effet seules les données utilisateurs sont conservées dans le disque de ces clones, aucun logiciel n'y est installé au départ. À chaque fois qu'un linked clone utilise un logiciel qui n'a pas été installé après coup par l'utilisateur, c'est au réplicat qu'il s'adresse pour avoir les informations liées à celui-ci. C'est à ce niveau que le gain de place peut être important car si un pool contient 20 VMs, il n'est pas nécessaire d'installer 20 fois la même application. Il n'est même pas nécessaire d'installer windows.

En résumé ce type d'architecture permet un gain de place relativement important car seuls les masters et les réplicats sont aussi volumineux qu'un véritable pc. De plus tant que le nombre de masters reste restreint, la gestion du parc de VM est réalisée sans trop de soucis. Le problème auquel est confronté l'équipe poste de travail aujourd'hui c'est la constante augmentation du nombre de masters alors que le nombre d'utilisateurs finaux ne bouge pas. En effet chaque fois que l'utilisateur d'un pool de VM fait une demande pour obtenir une nouvelle application ou une autre version de la même application, mais que les autres utilisateurs de ce pool n'en ont pas l'utilité il faut créer un nouveau master, un nouveau réplicat et de nouveaux clones. Et ce genre de demande est de plus en plus courant, ce qui rend le parc de plus en plus difficile à gérer pour l'équipe poste de travail. Une solution à ce problème serait l'utilisation du logiciel VMware ThinApp qui devrait permettre, *in fine*, de réduire grandement le nombre de masters. C'est ce sur quoi est basée la première partie du stage.

Quatrième partie

L'encapsulation d'applications avec VMware ThinApp

Le but de la première partie du stage est de savoir si l'intégration de ThinApp dans le SI³ est envisageable et si cette solution peut répondre au différents problèmes auquel doit faire face l'équipe poste de travail aujourd'hui.

7 ThinApp

Qu'est ce que ThinApp ? ThinApp est une solution de virtualisation d'applications qui a pour objectif de séparer l'application de ses dépendances avec l'OS afin de créer un package portable et utilisable sur n'importe quel version d'un même système d'exploitation. Une fois copiée sur un poste de travail l'application doit être capable de fonctionner en complète autarcie, comme dans une bulle. De plus ThinApp étant un produit de VMware il est fait pour permettre un déploiement rapide des applications encapsulées sur les pools ou les VMs voulus.

7.1 La virtualisation d'application selon Citrix ou VMware

Dans le livre de Damien BRULEY sur la virtualisation de poste de travail à l'aide de l'outil VMware View, il est écrit : "L'une des principales problématiques d'un poste de travail traditionnel est l'adhérence entre le matériel, le système d'exploitation, les drivers, les applications et le profil utilisateur". C'est à cette problématique que tente de répondre la virtualisation d'application en dissociant l'application du reste du poste de travail.

Le terme virtualisation d'application regroupe un ensemble de solution bien distinctes, parfois même complémentaires.

Citrix, un concurrent de VMware dans le domaine de la virtualisation, fourni une solution de virtualisation du type server hosted, comme pour les postes de travail virtuel de l'architecture VMware View, mais pour des applications (cf figure 6). Ce qui signifie que les applications sont tout d'abord installées sur des serveurs physique et fonctionnent dessus, mais l'affichage est déporté vers le poste de celui qui utilise cette application. Ceci se fait via un agent Citrix qui permet la communication entre le serveur et l'utilisateur. Cette solution est déjà utilisée par le Crédit Mutuel

3. Système d'Information

Arkéa et permet de résoudre certains problèmes comme les limitations de la vitesse de calcul d'un ordinateur ou encore la possibilité de ne pas avoir à installer une application qui prendrait beaucoup de place en mémoire. Ce qui est très intéressant quand il s'agit d'applications à lancer via une VM. Mais si l'application est bel et bien dissociée du poste de travail, qu'il soit physique ou virtuel, elle n'est pas pour autant dissociée du serveur sur lequel elle fonctionne.

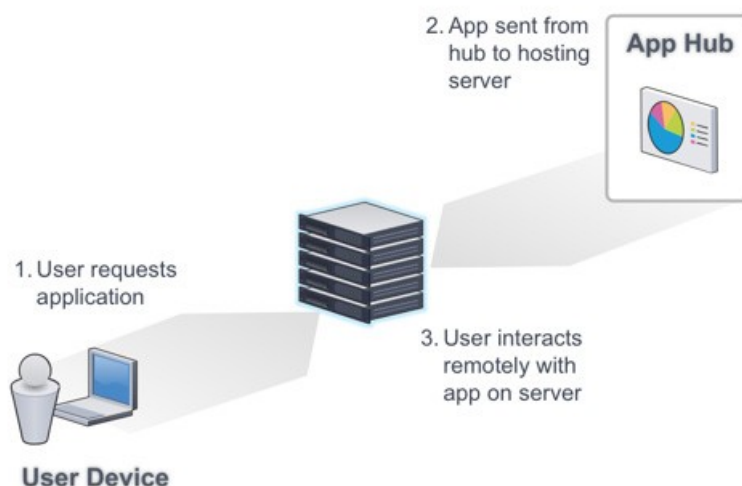


FIGURE 6 – Schéma des interactions entre l'utilisateur et le serveur Citrix

ThinApp, comme il a été vu un peu avant, propose une solution complètement différente de celle de Citrix. Au lieu de faire en sorte d'externaliser la totalité de l'application, le logiciel va packager l'application de manière à ce qu'elle soit complètement autonome. Pour cela il va créer autour de l'application une bulle contenant un système de fichiers virtuels, un registre virtuel ainsi qu'une sandbox dans lequel seront stockés tous les fichiers générés par l'application (cf figure 7). Par exemple si l'on package Word 2010 avec ThinApp, en principe les documents écrits avec cette application seront stockés dans la sandbox. Je dis bien en principe car la structure de la bulle peut légèrement varier suivant le choix du type d'isolation choisi pour la bulle applicative. L'inconvénient de cette solution, par rapport à Citrix, c'est que la bulle applicative créée avec ce logiciel peut vite devenir lourde. Or bien que sur un poste physique la place ne soit en général pas un problème, sur une machine virtuelle cela peut vite devenir problématique. Par contre l'avantage majeur de cette solution est la complète dissociation entre l'application et le reste de la machine. Un autre avantage de ThinApp, c'est la possibilité de faire fonctionner plusieurs versions d'un même logiciel sur la même machine. Il est par exemple possible de faire fonctionner plusieurs versions d'Internet Explorer simultanément, ce qui peut être intéressant pour les webmasters qui souhaitent tester leur site internet sur tous les navigateurs, mais aussi pour les entreprises qui utilisent encore

des applications fonctionnant sur IE 6 et qui voudraient travailler sous de nouvelles versions de Windows.

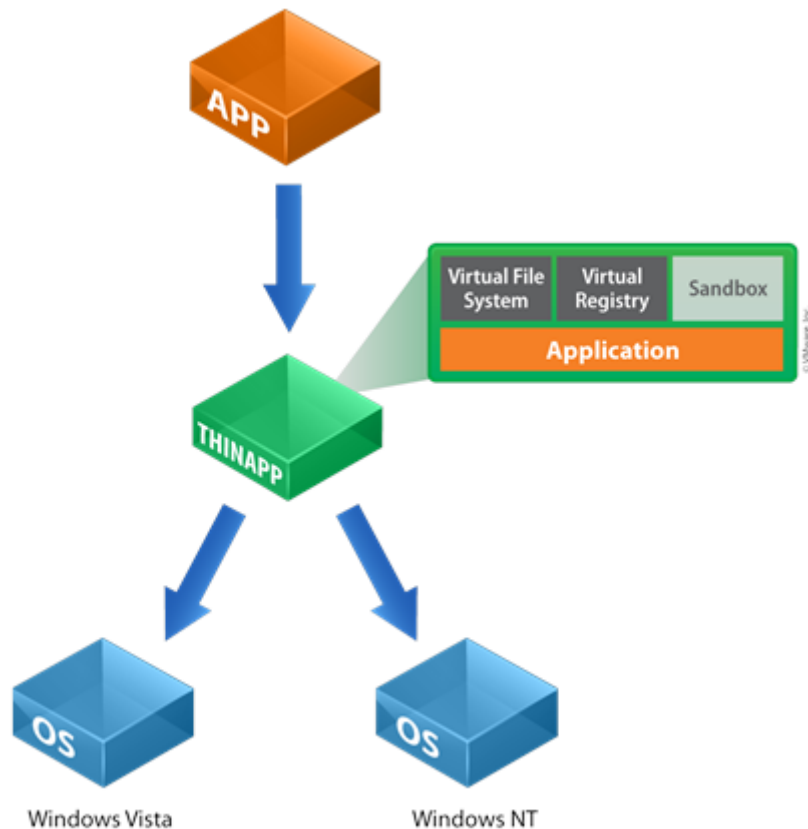


FIGURE 7 – La bulle ThinApp

La virtualisation d'applications permet donc de séparer l'application du reste de la machine physique ou virtuelle. Apportant ainsi une plus grande souplesse dans la gestion des conflits entre les applications dans le support applicatif ainsi que dans les processus de déploiement.

7.2 Les fonctionnalités intéressantes de ThinApp

7.2.1 AppLink

L'un des inconvénients majeurs de la création d'une bulle application est également ce qui fait son charme c'est à dire sa dissociation du reste de la machine et donc du reste des logiciels installés dessus. C'est ce à quoi tente de remédier AppLink. Cette fonctionnalité a pour objectif de lier deux bulles applicatives afin qu'elles puissent communiquer entre elles. Par exemple, si vous souhaitez lire des vidéos sur un Internet Explorer vous aurez probablement besoin d'utiliser flash player. Il suffit alors de

packager flash player et de le lier lors de sa capture à IE. La bulle Internet explorer aura alors connaissance de la présence de la bulle flash player et elle pourra lire les vidéos flash avec.

7.2.2 ThinApp Converter

ThinApp Converter est un exécutable fourni avec ThinApp qui permet de capturer de manière automatique un grand nombre d'applications. Mais son usage requiert que l'installation soit silencieuse pour fonctionner.

8 Création d'une bulle applicative

De la capture au déploiement

L'encapsulation et le déploiement d'une application suivent les étapes présentées sur la figure 8. Nous allons nous intéresser en détail à chacune d'elle pour mieux comprendre le fonctionnement de ThinApp.

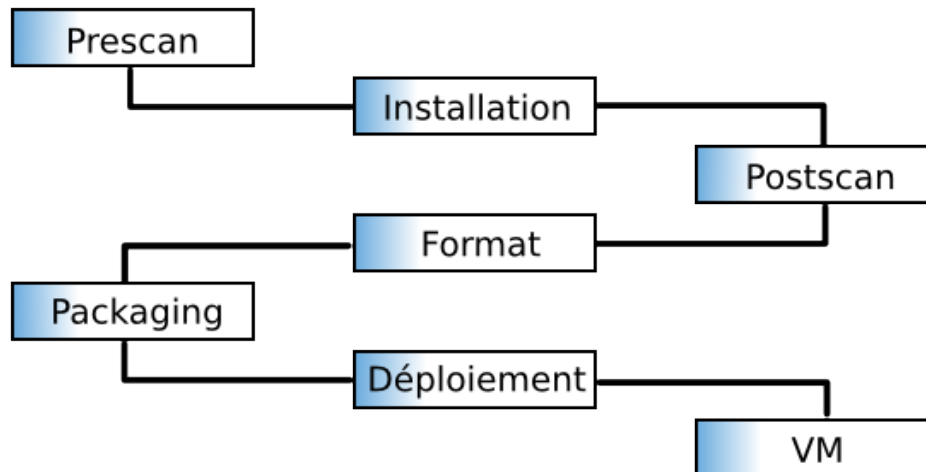


FIGURE 8 – Les étapes permettant la capture d'une application

8.1 Prescan

La première étape de la capture d'une application est le prescan. Lors de cette étape le logiciel "setup capture", qui permet le packaging d'une application en une bulle applicative, commence par réaliser un état des lieux de la machine avant l'installation de l'application que l'on souhaite packager. Pour cela il réalise un scan complet de la machine en vérifiant le ou les disques durs et les registres. Il est possible d'indiquer au logiciel les différents disques et registres qu'il doit scanner.

8.2 Installation

Une fois le prescan réalisé il faut installer l'application puis l'exécuter afin de réaliser toutes les configurations souhaitées.

8.3 Postscan

Comme pour le prescan, le postscan consiste en un état des lieux de la machine après l'installation du logiciel à capturer. Une fois celui-ci terminé, ThinApp va comparer le postscan au prescan et sauvegarder les différences. Ces différences vont permettre de récupérer l'application et ses dépendances pour les encapsuler par la suite.

8.4 Choix du Format

Lors de la capture d'une application il est possible de choisir le type de format de la future bulle applicative :

Exécutable : Il est possible d'encapsuler l'application sous la forme d'un seul exécutable contenant toutes les informations relatives au fonctionnement de l'application. Ce choix peut être intéressant dans le cas d'une application que l'on souhaite rendre portable et ne nécessitant qu'un seul point d'entrée.

Exécutable et fichier data : Une autre possibilité, ressemblant beaucoup à la première, est de créer un fichier data (.dat) contenant toutes les informations nécessaires et un (ou plusieurs) exécutable servant de point d'entrée dans le .dat. Ce choix est intéressant dans le cas d'une application que l'on souhaite portable et qui dispose d'un certain nombre de points d'entrée.

Microsoft Software Installer : Enfin il est également possible de packager l'application sous la forme d'un .msi. Ce format diffère légèrement des autres dans le sens où l'application n'est pas utilisable tel quel. Il faut d'abord exécuter le .msi qui va faire comme si il installait l'application avant de pouvoir accéder à cette dernière. En réalité il ne s'agit pas vraiment d'une installation, au sens propre du terme, puisque le logiciel est bel et bien confiné dans une bulle, mais ceci permet une meilleure organisation des exécutables créés sur l'ordinateur. En effet chaque point d'entrée est placé à l'endroit où il se serait trouvé si l'application avait vraiment été installée, il est même disponible via le menu démarrer. Ce dernier choix est très pratique lorsque l'objectif de la capture est de déployer l'application par la suite sur des postes de travail physiques ou virtuels.

8.5 Packaging

La dernière étape de la capture à proprement parler est le packaging de l'application, c'est la plus longue partie de la capture car il faut que ThinApp récupère, trie toutes les informations et construise la bulle applicative.

8.6 Déploiement

Une fois "ThinAppisée" l'application est déployable de plusieurs manières (cf figure 9).

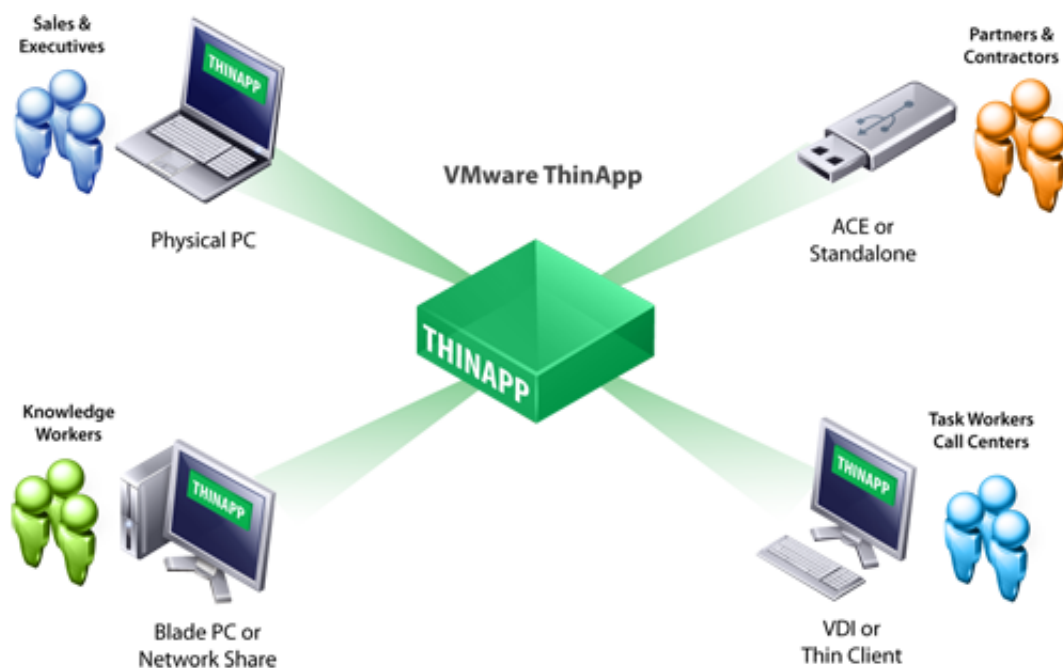


FIGURE 9 – Modes de déploiement d'une application ThinApp

L'application peut être fournie tel quel à un client sur une clef usb, cela est d'autant plus facile que celle-ci est portable. Le client pourra alors lancer l'application sur son ordinateur soit en la copiant dans un dossier soit en la lançant directement à partir de la clef.

Il est également possible de la distribuer en streaming, de manière assez similaire à Citrix. L'application serait alors exécutée sur une autre machine que celle de l'utilisateur et l'affichage serait déporté sur son écran et le client pourra alors interagir avec l'application comme si elle était exécutée sur son propre poste de travail.

Une autre option serait de déployer l'application directement sur un poste physique. L'application serait alors comme toutes les autres application installées nativement sur le pc mais fonctionnerai dans sa bulle.

Et enfin la dernière possibilité et la plus intéressante dans le cas présent est de déployer l'application, à l'aide de l'outil administrateur VMware View (cf annexe ...), sur des pools de VM ou des VMs en quelques clics seulement. Cette fonction

permettrait à l'équipe système de pouvoir limiter grandement le nombre de masters à créer. En effet au lieu de créer un master chaque fois qu'une nouvelle application est nécessaire, il suffirait de déployer une application ThinApp sur les VMs des utilisateurs qui en auraient besoin.

9 Phase de test

Après un certain temps passé à comprendre le fonctionnement de ThinApp sur des forums, de la documentation en ligne et à l'aide d'un livre sur VMware View, la phase de test de l'application a pu débuter. Afin de déterminer si oui ou non ce logiciel est vraiment la solution la mieux adaptée et si elle répondait aux attentes de l'équipe.

9.1 Capture d'une application simple

Mozilla Firefox est la première application qui a été capturée avec ThinApp. Le but de ce test était tout d'abord de me familiariser avec la méthode de capture d'applications mais également de voir à quel point l'application était séparée de son entourage. En effet un navigateur internet permet, comme son nom l'indique, de se connecter à internet et la question qui se pose c'est est-ce que l'application une fois packagée arrive toujours à se connecter et du coup à sortir de sa bulle. Le test fut concluant et a permis de montrer que bien qu'encapsulé, le logiciel était toujours capable de communiquer avec la machine.

9.2 Exécution de plusieurs versions du même logiciel

Une autre facette intéressante de ThinApp, c'est la possibilité de faire fonctionner plusieurs versions de la même application en parallèle. Il a donc fallu capturer toutes les versions d'Internet Explorer de la 6 à la 10, quelques versions de Mozilla Firefox (6, 17 et 22) et une version de Google Chrome. Internet Explorer 10 nécessitant beaucoup de mises à jour et de modifications n'a pas pu être capturé correctement, mais le reste des applications n'ont pas posé problème. Mieux encore il est tout à fait possible de les faire toutes fonctionner en parallèle comme on peut le voir sur la figure 10.

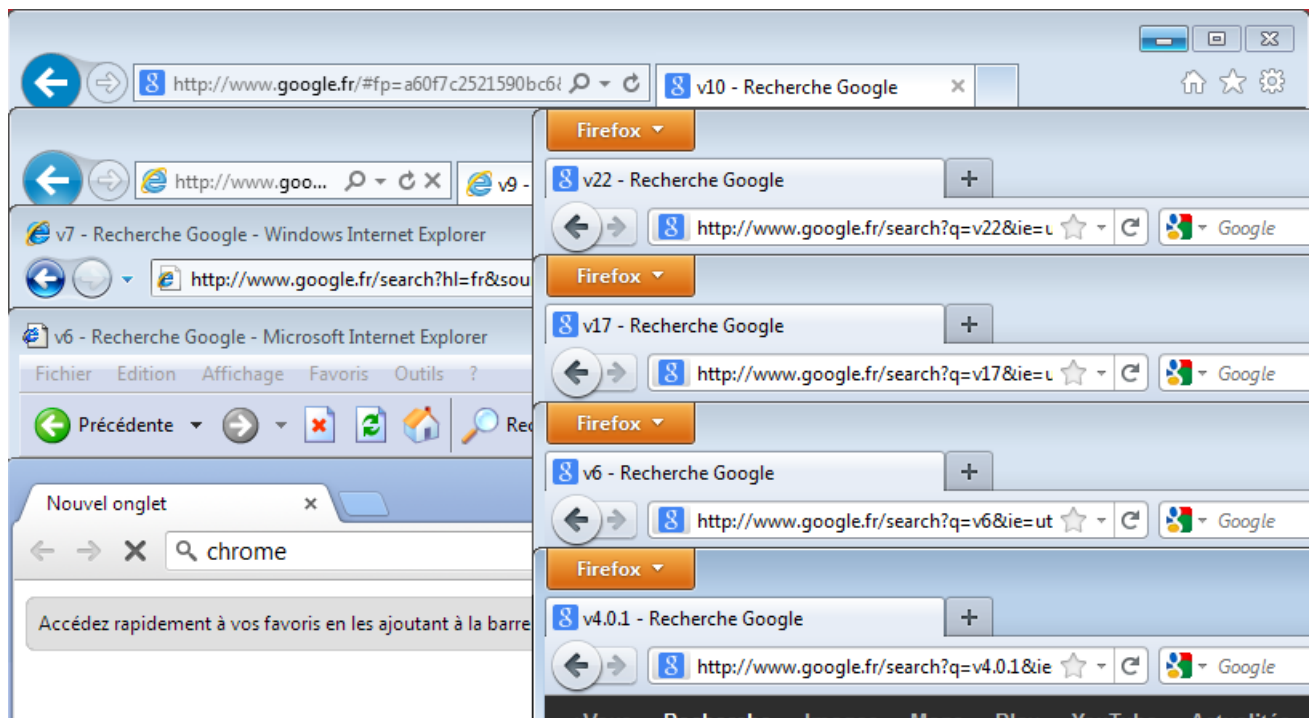


FIGURE 10 – Neuf navigateurs ouverts simultanément

Sur cette capture d'écran la version huit de Internet explorer n'est pas affichée car, bien que fonctionnelle en temps normal, si IE 10 est installé nativement sur le poste de travail il n'est alors plus possible de lancer la version encapsulée de IE 8 car un élément nécessaire au bon fonctionnement de la bulle applicative a du disparaître ou être modifié lors de l'installation de IE 10.

Pouvoir afficher simultanément plusieurs versions d'un même logiciel est intéressant car cela pourrait, par exemple, permettre à des entreprises ayant besoin de tester leur site internet sur plusieurs navigateur de pouvoir le faire sans avoir besoin de le faire sur plusieurs postes de travail.

9.3 Capture d'un client lourd

Après avoir testé la capture d'une application simple comme Mozilla Firefox, il a fallu s'intéresser à un logiciel beaucoup plus imposant : SAS. Cette application permet de réaliser des analyse de marchés et est une aide à la prise de décision dans le secteur commercial. C'est une application volumineuse (environ 20 Go) et qui a donc beaucoup de ramifications dans la machine. Mais cela n'as finalement pas posé de problèmes particuliers. L'application après trois essais a pu être packagée

correctement. Cela prouve que au final pour ThinApp la taille n'a pas d'influence sur le taux de réussite de l'encapsulation d'une application.

9.4 Déploiement d'une application sur des VMs

Une fois le principe de capture compris et testé, il a fallut tester le déploiement de l'application vers des machines virtuelles en utilisant VMware View Administrator. Au final l'opération est aussi simple qu'annoncée dans les documentations sur le sujet. Il suffit d'assigner l'application packagée sous forme de .msi à une VM ou à un pool de VMs et, à la prochaine connexion de l'utilisateur, l'application se déploiera toute seule. Il en va de même lorsqu'il s'agit de désinstaller une application "ThinAppisée". Il suffit d'enlever l'application souhaitée de la liste des applications liées au pool ou à la VM et, à la prochaine connexion de l'utilisateur, le logiciel se désinstallera de lui même.

9.5 Résultats des tests

L'élément le plus important qu'ont permis de mettre en valeur ces tests, c'est la nécessité de réaliser la capture de l'application souhaitée sur un poste de travail immaculé, et donc neuf. C'est très important car si, par exemple, l'objectif est d'encapsuler une application après avoir installer normalement plusieurs autres applications, il y a des risque que l'application ait besoin de la même dll que certains autres logiciels déjà installés au préalable. Par conséquent cette dll ne sera pas intégrée au package final et la bulle applicative ne pourra pas fonctionner.

Il est vrai que réinstaller Windows à chaque fois qu'il faut packager une application peut être chronophage si le nombre de captures à réaliser est important. C'est pour cela que l'utilisation d'un poste de travail virtuel est très utile. Il suffit de créer un snapshot juste après la création du poste et de revenir à ce snapshot après chaque capture d'application.

Cinquième partie

VMware ThinApp Automatisé

Objectif : L'étude de VMware ThinApp a démontré deux aspects négatifs majeurs. Tout d'abord la longueur du temps de capture d'une application, mais malheureusement il n'existe pas de moyen pour accélérer le processus. Et le fait que la virtualisation d'application avec ce logiciel est particulièrement saccadée. En effet lors du processus de nombreuses fenêtres informatives s'ouvrent et il est nécessaire de cliquer sur "continuer" pour poursuivre la capture.

Afin de remédier au deuxième problème, il a fallu développer un logiciel capable de rendre silencieuse la capture d'une application, sans pour autant réduire l'efficacité du logiciel et les choix de l'utilisateur.

Il fallait donc, en utilisant ThinApp, arriver à obtenir une application encapsulée à partir du package d'installation du logiciel souhaité comme le montre la figure 11, et tout cela silencieusement.

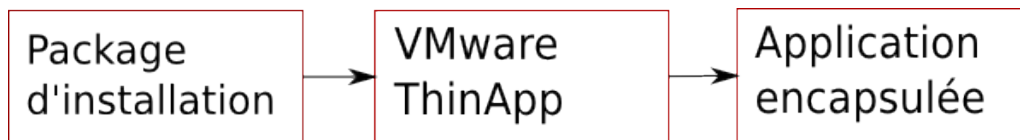
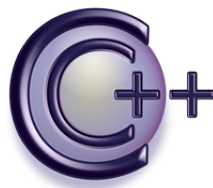


FIGURE 11 – Schéma du processus de capture

10 Outils utilisés pour développer le logiciel

10.1 Le C++



Pour des raisons pratiques, il a été décidé que le logiciel serait développé en C++. Le C++ est un langage de programmation dite "orienté objet". C'est un langage puissant, qui est très largement utilisé par la communauté des développeurs informatiques. Il est notamment utilisé dans le milieu du jeu vidéo.

10.2 Qt



Pour la réalisation de l'interface graphique du logiciel, c'est la librairie Qt, développée par Nokia, qui a été retenue. Notamment car elle est à la fois puissante et gratuite.

10.3 Fonctions Windows

Afin de pouvoir utiliser ThinApp sans passer par l'interface graphique de ce logiciel il a fallu utiliser les fonctions de Windows. En effet il faut, pour utiliser le logiciel, utiliser des fonctions de commande Windows de type bash qui permettent de lancer manuellement les différentes étapes de la capture d'une application.

11 Développement de l'application

11.1 Prise en main de Qt

Avant même d'envisager de commencer à coder le logiciel il a fallu prendre en main la librairie Qt. La création de l'interface graphique a donc débuté par le développement d'une fenêtre affichant un bouton et qui permet, en cliquant sur ce dernier, d'afficher une image (cf figure 12). Cette première étape a pour but de prendre en main la librairie Qt en créant des widgets simples et en les liants à l'aide de « connects » à des actions (ou « slots »). C'est en effet la base de toute Interface Homme Machine (ou IHM).

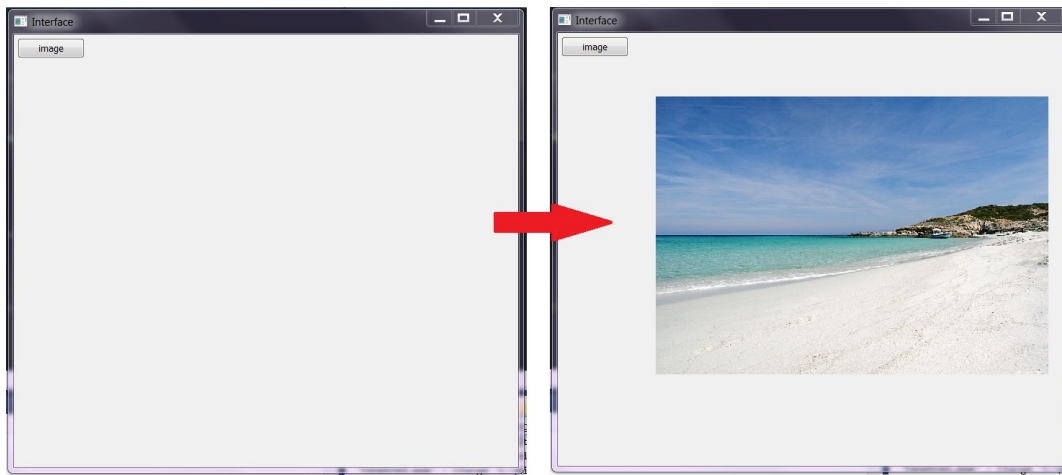


FIGURE 12 – Ébauche d'IHM

Une fois cette première étape d'apprentissage de la librairie Qt, il était possible de commencer à penser à l'interface finale.

11.2 L'interface

Après avoir pris en main Qt la création de l'interface pris peu de temps. Les éléments nécessaires au bon fonctionnement de ThinApp étant peu nombreux.

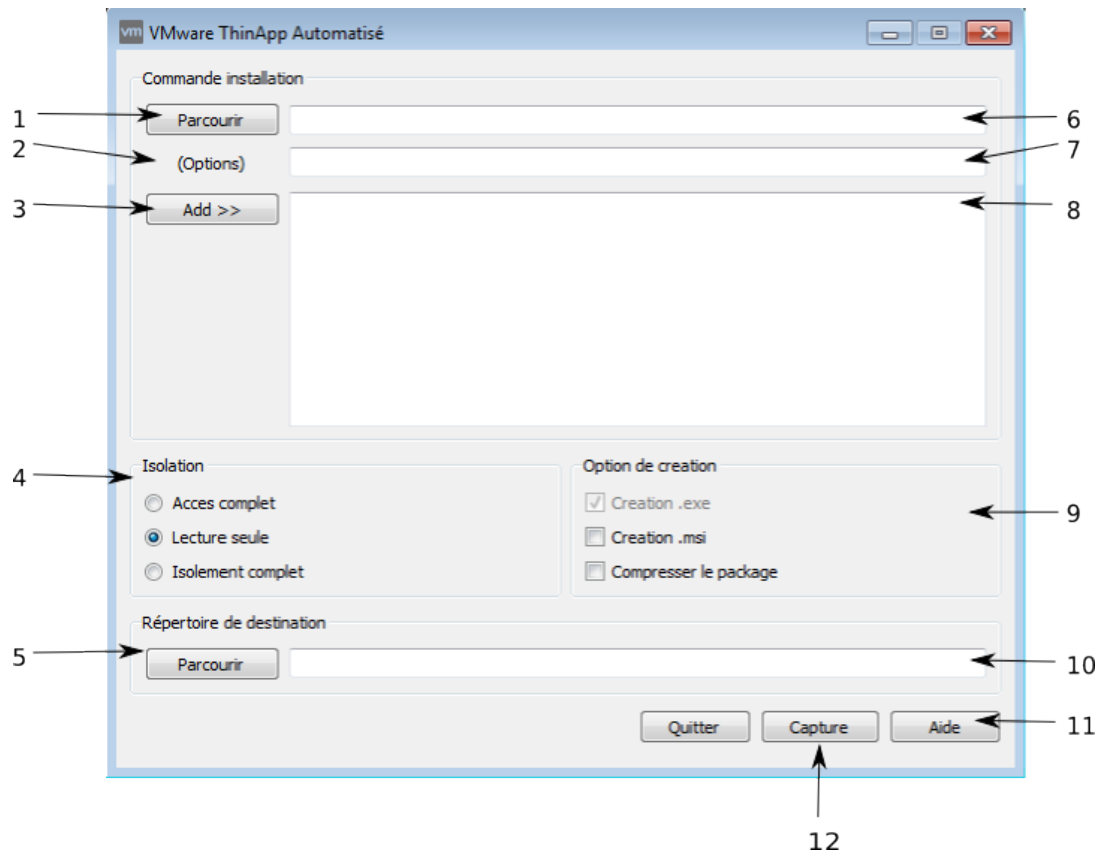


FIGURE 13 – L'interface graphique finale

1. Bouton permettant de parcourir les fichiers afin de retrouver le logiciel à installer.
2. Champ destiné à recevoir les options d'installation du logiciel, comme par exemple /silent pour une installation silencieuse.
3. Bouton permettant d'ajouter une commande d'installation (chemin du fichier + option d'installation).
4. Checkbox permettant de choisir le degré d'encapsulation de l'application "thinappisée".
5. Bouton permettant de naviguer dans les fichiers afin de choisir le répertoire de destination de l'application virtuelle et de l'ensemble des données créées durant le processus de capture.
6. Champ où va apparaître le chemin du logiciel à encapsuler.
7. idem 2
8. Champ affichant la liste des logiciels à installer durant la capture de l'application.

9. Checkbox permettant de choisir le format de l'application encapsulée en sortie de la capture.
10. Champ où va apparaître le chemin du logiciel encapsulé.
11. Bouton permettant d'accéder à la fenêtre d'aide.
12. Bouton permettant de lancer la capture de l'application.

11.3 Les difficultés rencontrées

11.3.1 Utilisation de ThinApp sans l'interface

Une des difficultés rencontrées fut l'utilisation de ThinApp avec des commandes bash. Au départ il avait été envisagé d'utiliser le powershell qui est le nouveau shell de windows, mais après quelques jours de recherche, cette idée a été abandonnée. En effet lors de l'installation de ThinApp, un exécutable du nom de snapshot.exe est installé également. Et cet exécutable permet de reprendre toutes les étapes d'une capture d'application manuellement. Mais cet exécutable utilise non pas le powershell mais le bash. Une fois cela compris, il fut relativement simple d'arriver à comprendre le fonctionnement de ThinApp.

11.3.2 Utilisation des API Windows

Une autre des difficultés rencontrées fut l'intégration dans un code en c++ de fonction Windows permettant de lancer des commandes de type bash. Pour cela il a fallut utiliser des fonctions préconçues par Windows, destiné à envoyer des commandes à partir d'un programme en c++.

Sixième partie

Conclusion

L'étude de VMware ThinApp a prouvé que c'était une solution de virtualisation intéressante. Elle présente trois intérêts majeurs. Tout d'abord, ThinApp permet d'encapsuler les applications et ainsi de les faire fonctionner en autarcie, ce qui évite les conflits entre applications et permet de lancer plusieurs versions du même logiciel en même temps. Ensuite, toujours grâce au fait que ThinApp encapsule l'application, elle permet de prolonger la durée de vie de certaines applications en évitant les conflits entre le logiciel et le système d'exploitation. Et enfin, cette solution permet une grande mobilité des applications. En effet une application encapsulée avec ThinApp peut être déployée simplement car elle est portable. Au Crédit Mutuel Arkea, il existe déjà une autre solution de virtualisation d'application, il s'agit de Citrix. Mais ces deux outils sont utilisés à des fins différentes, ce qui les rends complémentaires. Enfin ce qui peut rendre ThinApp intéressante pour le Crédit Mutuel, c'est que chaque poste de travail virtuel VMware View dispose d'une licence pour ce logiciel et son utilisation ne serait donc pas un coût supplémentaire pour l'entreprise.

En plus de tout cela, à l'aide du logiciel développé au cours du stage, les applications qui ont besoin d'être packagées peuvent l'être de manière silencieuse et donc de manière moins chronophage.

En somme, l'intégration de ThinApp dans le SI serait intéressante pour le Crédit Mutuel Arkéa.

Septième partie

Bibliographie

- Damien BRULEY, *VMware View Virtualisation des postes de travail*, ENI
- Page produits VMware, <http://www.vmware.com/fr/products/thinapp/>
- Portabiliser une application avec ThinApp <http://www.commentcamarche.net/faq/15240-portabiliser-une-application-avec-thinapp>
- Citrix, <http://www.citrix.fr/>
- API Windows, <http://msdn.microsoft.com/fr-fr/library/windows/apps/br211377.aspx>
- Documentation Qt, <http://qt-project.org/doc/qt-4.8/>
- Le site du zéro, <http://fr.openclassrooms.com/>

Huitième partie

Annexes

12 Bilan Personnel

Durant ce stage j'ai énormément appris sur trois sujets :

Le fonctionnement d'une grosse entreprise : Une entreprise de la taille du Crédit Mutuel Arkéa a des fonctionnements bien particuliers, de parts l'architecture des services qui la compose. Ce fut très enrichissant de voir comment fonctionne une entreprise de cette taille, les avantages et les désavantages d'une telle structure, les méthodes de travail employées au sein de celle-ci ou encore l'inertie dont peuvent parfois faire preuve les grosses entreprises.

La virtualisation de postes de travail et d'application : En effet vu que mon stage portait sur la virtualisation d'application, j'ai beaucoup appris sur celle-ci et sur les différentes méthodes qui permettent de virtualiser une application.

La structure d'un poste de travail sous Windows : Lors de la phase de test de ThinApp, nous avons rencontré de nombreux problèmes liés aux éléments qui composent un poste de travail sous Windows. C'est donc tout naturellement que j'ai appris à comprendre globalement comment fonctionne un poste sous Windows.

J'ai également trouvé cette expérience en entreprise particulièrement enrichissante à titre personnel. J'ai pu me rendre compte de ce qu'étais véritablement le travail d'un ingénieur systèmes dans une entreprise tel que le Crédit Mutuel Arkéa, et j'ai pu me confronter à de véritables problèmes techniques.

Enfin j'ai également pu voir comment se passe le déroulement d'un projet en entreprise, avec les blocages, les retours en arrière et bien sur les avancées.

13 WYSE

WYSE Technology inc. est une compagnie américaine spécialisée dans la vente de clients légers (cf figure14)



FIGURE 14 – Photo d'un client léger WYSE

Définition de client léger d'après Wikipedia Au sens matériel, un client léger est un ordinateur qui, dans une architecture client-serveur, n'a presque pas de logique d'application. Il dépend donc surtout du serveur central pour le traitement. C'est sur le presque pas que la nuance se fait. Un client léger peut se contenter d'une machine minimaliste en termes de matériel : Un ordinateur (fort) ancien. Un ordinateur moderne. Un ordinateur créé pour faire du client léger