

Rapport de Stage



VILLE DE
TOURS

Rapport de Stage

Stagiaire au service Responsable d'Application

HUBERT Jules

04 Janvier 2021 au 26 Février

Tuteur de stage : Stéphane PIRONNET

Etablissement : Lycée Paul Louis Courier

Entreprise d'accueil : Mairie de Tours

Sommaire :

I.	Présentation.....	5
1)	L'entreprise.....	5
2)	Le service.....	6
3)	L'environnement matériel et logiciel.....	6
4)	Situation professionnelle.....	7
5)	Mission confiée.....	7
6)	Planification envisagée.....	8
7)	Besoin de formation.....	8
I.	Semaine du 4 janvier au 8 janvier 2021.....	8
1)	Tâches prévues.....	9
2)	Tâches réalisées.....	9
a)	Automatisation via la console WAPT.....	9
b)	Compréhension des scripts déjà créé.....	11
c)	Trouver des réponses aux interrogations de mes collègues vis-à-vis des scripts.....	11
3)	Analyse.....	11
4)	Tâches pour la semaine suivante.....	11
I.	Semaine du 11 janvier au 15 janvier 2021.....	12
1)	Tâches prévues.....	12
2)	Tâches réalisées.....	12
d)	Script d'installation et de désinstallation automatique.....	12
e)	Problèmes rencontrés.....	15
3)	Analyse.....	16
4)	Tâches pour la semaine suivante.....	17
I.	Semaine du 18 janvier au 22 janvier 2021.....	17
1)	Tâches prévues.....	17
2)	Tâches réalisées.....	17
a)	Compte Rendu.....	17
b)	Script de désinstallation et d'installation application WinDev.....	17
c)	Autres applications.....	18
d)	Réunion.....	19
3)	Analyse.....	19
4)	Tâches pour la semaine suivante.....	19
II.	Semaine du janvier 25 au 29 janvier 2021.....	20
5)	Tâches prévues.....	20
6)	Tâches réalisées.....	20

e)	Formalisme et amélioration des données.....	20
f)	Application Monstro.....	20
g)	Applications WinDev.....	21
7)	Analyse.....	21
8)	Tâches pour la semaine suivante.....	21

I. Présentation

1) L'entreprise

Je travaille actuellement dans la mairie de Tours, plus précisément dans le département informatique la DSI. Celle-ci est disposé sur 3 étages et regroupe les services suivant :

Pôle Direction :

- Assistance à maîtrise d'ouvrage (sous-catégorie)

Pôle Aménagement :

- Numérique du territoire

Pôle Ressources :

- Direction
- Marchés et finances
- Assistance
- Juridique

Pôle Fonctionnel :

- Direction
- Responsables Applications
- Coordination de projet
- Chargés de projets
- Projets
- Lab'(Développeurs)

Pôle Technique :

- Direction
- Gestion des demandes
- Coordination technique
- Projets /Expérience utilisateur

- Environnement utilisateur
- Gestion technique
- Infrastructure
- Infrastructure/réseau
- Infrastructure/Système
- Support production

2) Le service

Le service dans lequel je fais mon stage a pour mission d'aider les employés en difficulté avec la mise en œuvre d'une application, c'est le service Responsable d'Application. Ici grâce à un outil de ticketing, les employés peuvent faire une demande de support via une interface graphique leur permettant de préciser quel est leur problème, sur quelle application (car il en existe pas moins de 180) et le niveau d'urgence de celui-ci.

3) L'environnement matériel et logiciel

J'ai à ma disposition un ordinateur tout en un avec un deuxième écran. Le poste fonctionne sous Windows 10 Professionnel x64. Je suis sur le réseau de la mairie (intranet), et j'utilise un compte utilisateur dis « stagiaire » avec les mêmes droits que les personnes de mon service. De plus, j'ai un compte administrateur (comme tous mes collègues à la DSI) en local c'est-à-dire que j'ai des droits administrateurs sur mon poste mais pas sur le réseau. Pour ce qui est de la partie logicielle, j'ai recours à des logiciels d'édition, de Scripting, d'automatisation, de bureautique, mais aussi de communication, tels que :

Edition :

- Notepad++
- GIMP

Scripting :

- Pyscripter(lié à WAPT)

Automatisation :

- Console WAPT
- WAPT Self-Service

Bureautique :

- Word
- Adobe Reader
- Outil de capture
- Mozilla/Qwant

Communication :

- Outlook
- Webex meeting

4) Situation professionnelle

Le projet auquel de participe a pour mission d'automatiser l'installation, la désinstallation et les mises à jour des applications métiers de l'entreprise.

L'idée derrière cette automatisation c'est de faire du Self-Service, c'est-à-dire appliquer la méthode du « do-it yourself » aux employés. Les employés n'étant pas tous formés en informatique et mes collègues de la DSI n'ayant pas un temps infini à consacrer aux problèmes d'installation, il a été décidé d'automatiser l'installation, la désinstallation et les mises à jour des applications. Grâce à cela, les employés peuvent installer l'application à partir du DSI Store en un clic et de la même manière ils peuvent désinstaller le logiciel d'eux-mêmes. C'est ici que l'on voit que l'attribution des droits dans l'AD est très importante, car si une personne venait à avoir des droits trop importants elle risquerait en voulant enlever le logiciel de son poste de désinstaller l'application du Store. C'est pour cela qu'il y a tout un travail en amont de la part de mes collègues du Service Infrastructure Système (Pôle Technique) qui doivent attribuer les droits aux Applications, et aux utilisateurs.

5) Mission confiée

J'ai pour mission de créer une documentation sur la méthode d'automatisation d'une application à partir de la console WAPT en fonction des différences de programmation entre les applications. Typiquement il y a des applications développées sous WinDev, d'autres sous Java et ce paramètre va influencer le script créé avec Pyscripter à partir de la Console WAPT. Pour les mises à jour, c'est encore plus fabuleux car celles-ci se font de manière automatique (grâce au script) à chaque démarrage de l'application. Je dois donc étudier les différentes manières de faire un script, normé (pour que tout le monde comprenne bien qu'est-ce que le script fait) le plus professionnel (lier à la norme on respecte certaines règles internes) et efficace possible (si on peut faire en 10 lignes un programme qui en fait 70). Enfin dans cette documentation je dois regrouper la méthode générale pour automatiser une application de A à Z en plus du script, mais aussi regrouper les différentes erreurs que j'ai pu rencontrer dans mes automatisations.

6) Planification envisagée

Pour l'instant il a été convenu que tous les mercredis on ferait une réunion avec le directeur du Pôle Fonctionnel Monsieur MORCET et mon maître de stage Monsieur PIRONNET. Pour la première semaine j'ai surtout dû me familiariser avec les différents logiciels, je me suis énormément documenté et j'ai pu apprendre beaucoup de choses en très peu de temps. J'ai commencé sur une application qui était déjà en cour d'automatisation avec un début de script, l'objectif trouver une solution pour la désinstallation du logiciel qui n'est actuellement pas fonctionnel pour mercredi.

7) Besoin de formation

J'ai pu utiliser des documentations diverses et varier tels que :

- Documentation de technicien et maître de stage M. PIRONNET

- Documentation sur YouTube notamment avec les vidéos de Tranquil IT (créateur du logiciel WAPT)

- Documentation sur le site officiel de Tranquil IT

- Renseignement auprès de :

 - M. ROHART (Développeur des applications WinDev)

 - M. CHAPPRON (Gestion de demandes)

 - M. GRANDENSCHWILLER (Technicien de mon service)

 - M. PIRONNET (Technicien de mon service et maître de stage)

 - M. MORCET (Directeur du Pôle Fonctionnel)

- Forums

J'ai à la suite de toutes ses documentations et renseignements, commencé à croiser les informations avec les scripts déjà existants et mes connaissances pour pouvoir identifier la meilleure manière de faire un script fonctionnel mais aussi compréhensible pour les applications WinDev.

I. Semaine 1

1) Tâches prévues

- Comprendre la méthode d'automatisation + Apprendre à se servir de la console WAPT
- Créer un compte Webex + Apprendre à se servir de Webex meeting
- Mettre en place mon poste professionnel et personnel pour le télétravail
- Trouver des réponses aux interrogations de mes collègues vis-à-vis des scripts
- Orienter une collègue sur WAPT
- Assister aux réunions de projets

2) Tâches réalisées

a) Automatisation via la console WAPT

=>Premier problème quand je lançais un script la console affichée un message d'erreur « FATAL ERROR READONLY DATABASE exit code : 3 »

Première Hypothèse :

Peut-être l'UAC qui se manifeste (User Account Control) ?

Seconde Hypothèse :

Problème de droit administrateur ?

Solution envisagée : Demandé un compte administrateur

Solution : j'ai obtenu un compte administrateur local me permettant de lancer la console WAPT en mode administrateur Windows et j'ai pu créer un certificat de paquet WAPT.

Conclusion : l'UAC empêche les exécutable de s'exécuter, car en lançant le script sur Pyscripiter on installe directement le logiciel que l'on veut automatiser sur l'ordinateur pour voir si notre script est fonctionnel. L'UAC a été créé comme un filet de sécurité au cas où un utilisateur mal avertis exécuterait un programme malveillant. En étant administrateur celui-ci ne se manifesterait pas donc le programme pourra fonctionner correctement et je pourrais ainsi tester mon programme. Et le manque de certificat nous empêchait de décrypter le paquet.

=>Second problème après l'exécution d'un script j'obtenais ce message d'erreur « ERROR UTF-8 0xbe09 »

Hypothèse : Sûrement un problème de conversion d'un type à un autre et qui créer un conflit.

Solution envisagée : Vérifier tous les types puis exécuter le programme à nouveau.

Solution : Après redémarrage de mon poste, et retour sur une ancienne version du script tout marche correctement.

Conclusion : Soit un simple bug informatique, soit une erreur de PyScripter, soit un problème de conversion de type. A noté que cette erreur a déjà été observé par des techniciens par le passé mais aucune certitude quant à la provenance de celle-ci.

=>Troisième problème après avoir ajouté des modifications à un script une erreur est survenue et elle était inattendu car il n'y avait aucunes erreurs dans le programme d'ailleurs la ligne citée était une ligne de la librairie Pyscripter « SyntaxError: Non-ASCII character '\xc3' »

Hypothèse : Avec l'aide de plusieurs forums, le problème serait une ligne placée juste avant ce commentaire « #-*- coding : utf-8 -*-»

Solution envisagée : Déplacer la ligne qui serait la cause de problème

Solution : La ligne a été remise en première position du code et le script est de nouveau fonctionnel.

Conclusion : Ne jamais placé quoi que ce soit avant la ligne « #-*- coding : utf-8 -*-», les raisons pour lesquelles il ne faut pas le faire m'échappe néanmoins c'est la solution à cette erreur.

=>Quatrième problème quand je fais un test de désinstallation de logiciel celle-ci ne fonctionne pas et renvoie l'information « No package remove »

Première partie du problème : S'il me dit que suite à la désinstallation « aucun paquet n'a été supprimé » c'est un énorme problème... La désinstallation du logiciel doit permettre de l'enlever du poste local mais pas de le supprimer des groupes de paquets de WAPT. Si ça venait à être le cas à chaque désinstallation de la part d'un utilisateur l'application s'enlèverait du poste de celui-ci mais aussi du store ou pire, que du store.

Seconde partie du problème : Pour quelle raison l'exécution du programme se comporte comme ci il ne lisait absolument pas la partie `uninstall()` du programme, ou alors même si il la lis pourquoi il ne l'applique pas ?

Solution envisagée : Remplacer le programme de désinstallation par un exécutable WDUNINST.exe qui a pour mission de désinstaller l'application. Cette méthode a déjà été appliquée avec succès pour la partie installation de certaines applications. De plus, s'il existe un tel fichier, Pyscripter + WAPT peuvent être capable de le trouver.

b) Compréhension des scripts déjà créé

Au début je ne comprenais absolument pas les scripts que j'avais sous les yeux certes c'est du python mais il y a des spécificités liées à Pyscripter. En effet, les pages de codes python sont agrémenté de fonctions propre à l'IDE Pyscripter que je ne connaissais pas et d'ailleurs je pensais même que c'était un logiciel à part entière sauf que non. Et c'est là que mes recherches sur le net avaient donc plus de sens après. J'ai donc pu me documenter correctement sur les fonctions qui étaient utilisées et j'ai donc pu comprendre les scripts que j'avais sous les yeux.

c) Trouver des réponses aux interrogations de mes collègues vis-à-vis des scripts

Comme j'ai dit précédemment je me suis beaucoup documenté mais j'ai aussi compris seul beaucoup de choses et j'ai donc pu expliquer à mes collègues l'utilité de certains termes/fonctions qu'ils n'avaient pas compris. Notamment le termes « `control[' ']` » qui permet de chercher des valeurs de la page control qui doit être renseignée avant de commencer un script. Dans celle-ci on retrouve la version du logiciel, son nom, son nom de paquet WAPT, l'avancement du projet, la clé de cryptage, le nom de la personne qui a créé ce paquet etc...

3) Analyse

Je pensais que chaque mise en place de script serait rapide mais finalement il y a énormément de choses à intégrer et à comprendre. Notamment les normes internes sur l'installation des logiciels, leur emplacement, les droits nécessaire à son ouverture et son édition.

4) Tâches pour la semaine suivante

- Réussir à faire fonctionner la désinstallation de l'application Monstro.
- Continuer la documentation sur la procédure à respecter pour faire un bon script d'automatisation avec un logiciel développé sous WinDev.
- Trouver une solution au problème de l'application IXbus.
- Rediscuter des normes de script qui vont être mis en place avec toute l'équipe et avoir notamment le ressenti du technicien en charge de WAPT.

I. Semaine 2

1) Tâches prévues

- Créer un script de désinstallation fonctionnel pour l'application « Monstro »
- Etudier la question des mises à jour sous WinDev
- Améliorer et alléger le script d'installation déjà créé par d'autres techniciens pour le logiciel « Monstro »
- Améliorer sa compréhension de WAPT
- Appliquer les normes de Script abordées en réunion
- Rediscuter des attentes venant de Monsieur VIALE (technicien en charge de WAPT) et du directeur Monsieur MORCET
- Allègement de la taille du paquet WAPT

2) Tâches réalisées

d) Script d'installation et de désinstallation automatique

i. *Page Control*

package	: tmvl-monstro
version	: 1.4.0.1-8
architecture	: all
section	: base
priority	: optional
name	: Monstro
categories	: Applications Métiers
maintainer	: TM-DSI FONC Resp.Appli. Pironnet S. and Stagiaire Jules
description	: Application servant à la gestion des encombrants pour le service de la collecte (Tour(s) Plus)
depends	:
conflicts	:
maturity	: DEV
locale	:
target_os	:
min_os_version	:
max_os_version	: 1
min_wapt_version	:
sources	:
installed_size	:
impacted_process	: Monstro.exe

Cette page permet de donner des informations à la plateforme WAPT, tel que : la version, le nom, l'auteur, la maturité du projet, la catégorie du projet etc... Ces informations en plus d'informer un développeur WAPT sur la nature du paquet, sont aussi utilisées dans le script grâce à la commande « control[''] ». Il suffit de noter le nom du paramètre, control['name'] et il retournera « Monstro ».

ii. Attribution des variables

```
#Folder
InstallDirectory = r'c:\APPS\MonstroProd'
Install = 'INSTALL.EXE'
Targ=r'c:\APPS\MonstroProd\Monstro.EXE'
TargUdate=r'c:\APPS\MonstroProd\WDUPDATE.NET'
ShortCut = 'Monstro'
```

InstallDirectory : définit le répertoire d'installation du logiciel

Install : définit l'exécutable

Targ : Donne la « cible » du raccourci

TargUpdate : Définit le fichier d'Update à remplacer

ShortCut : Nom du raccourci

iii. Script d'installation

```
def install():
    #Anti-plantage
    killalltasks(control['impacted_process'])
    #On vérifie l'existence du logiciel sinon on vérifie la version
    if os.path.isdir(InstallDirectory):
        #Annonce qu'il est déjà installé et que l'on va vérifier la version
        print(control['name'],('is already installed, verification if need update'))
        #On copie le fichier d'update
        filecopyto('INSTALL\WDUPDATE.NET',TargUdate)
        #Par défaut les appli windew se mettent à jour si le setup d'installation se lance et détecte que l'application est déjà présente
        install_exe_if_needed(Install,silentflags='/SILENT', key='',min_version='')
    else:
        #Annonce de l'installation
        print(control['name'], ('will be installed'))
        #Création du repertoire qui va contenir le logiciel
        mkdirs(InstallDirectory)
        #On installe le logiciel
        install_exe_if_needed(Install,silentflags='/SILENT', key='',min_version='')
        #Création d'un raccourci sur le bureau
        create_desktop_shortcut(ShortCut, target=Targ)
        #Création d'un raccourci dans le menu démarré
        create_programs_menu_shortcut(ShortCut, target =Targ)
        #On copie le fichier d'update
        filecopyto('INSTALL\WDUPDATE.NET',TargUdate)
        #Le fichier est installé
        print(control['name'],('is now installed'))
```

Dans un premier temps on fait appel à l'un des champs de control « impact_process » dans lequel nous avons mis le nom de l'exécutable de l'application. La fonction « killalltasks() » arrête le processus fournit en paramètre. L'intérêt ici est d'empêcher qu'une personne ayant déjà installé l'application et la réinstalle (soit parce qu'elle n'a pas fait exprès soit parce qu'elle a oublié etc...) alors même que l'exécutable du logiciel est en fonctionnement. Si l'utilisateur venait à le faire, il y aurait alors un plantage de l'application qui entraînerait sûrement la création d'un ticket au prêt du support.

Ensuite on crée une condition simple « Si l'application est déjà présente dans le dossier d'installation alors on vérifie qu'il n'y a pas de mise à jour, sinon on l'installe » (Le paramètre et/ou la fonction vont peut-être changer c'est en discussion)

PS : Tous les print() sont là simplement pour informer le développeur dans l'avancé d'exécution de son programme en cas d'erreur notamment.

Ensuite on copie le fichier de mise à jour du paquet WAPT pour remplacer celui en destination locale pour les mises à jour avec la fonction « filecopyto(filename, target) ».

Si la personne a l'application depuis longtemps son fichier « WDUPDATE.NET » n'est peut-être plus en phase avec l'évolution du logiciel faite par les développeurs.

Ensuite on lance le setup d'installation qui va mettre à jour seul l'application, spécificité des applications WinDev développé en interne.

Dans le cas du « else », on rentre dans la phase d'installation. On annonce l'installation, puis on crée le répertoire d'installation s'il n'existe pas déjà avec la fonction « mkdirs() ». On lance le setup d'installation avec la fonction « install_exe_if_needed() ».

On crée un raccourci sur le bureau avec la fonction « create_desktop_shortcut(name, target) » la cible de la fonction va définir le bon fonctionnement ou non du raccourci.

On crée un raccourci dans le menu démarré avec la fonction « create_programs_menu_shortcut(name, target) »

Le fichier de mise à jour du paquet est copié puis collé dans le répertoire d'installation.

Enfin on annonce que l'installation est terminée.

iv. Script de désinstallation

```
def uninstall():
    #Anti-plantage
    killalltasks(control['impacted_process'])
    #Cherche si le fichier existe: si oui il est désinstallé sinon affiche un message
    if os.path.isdir(InstallDirectory): #return boolean
        #Ici on annonce la désinstallation
        print(control['name'],('will be uninstalled'))
        #On supprime le raccourci bureau à partir de son nom
        remove_desktop_shortcut(ShortCut)
        #On supprime le raccourci du menu démarré à partir de son nom
        remove_programs_menu_shortcut(ShortCut)
        #On supprime le Dossier qui contient l'application
        shutil.rmtree(r'c:\APPS\MonstroProd')
        #Si le chemin est introuvable c'est qu'il a été supprimé
        if os.path.isdir(InstallDirectory) is False:
            print(control['name'],('is now uninstalled'))
        else:
            print("ERROR : Uninstall failed")
    else:
        #On déclare qu'il est déjà désinstallé
        print(control['name'],('is already uninstalled'))
```

Comme pour l'installation on empêche une situation critique en arrêtant le processus de l'application en cas d'activation de la désinstallation.

PS : Même si l'idée d'empêcher la désinstallation marche aussi, il ne faut pas oublier que nous sommes dans une démarche de self-service donc tout est simplifié pour l'utilisateur. Il est donc très facile pour l'utilisateur d'installer ou de désinstaller, en cas de désinstallation non voulu il lui suffit de cliquer sur le bouton d'installation et en quelques secondes c'est fait.

On vérifie que le répertoire d'installation est déjà présent si oui on lance la désinstallation sinon c'est que l'application est déjà désinstallée.

On annonce la désinstallation, on supprime le raccourci du bureau et le raccourci du menu démarré avec les fonctions : « remove_desktop_shortcut(name) » et « remove_programs_menu_shortcut(name) »

Ensuite, on supprime le répertoire d'installation et tout son contenu avec la fonction « shutil.rmtree(NameDirectory) ». On vérifie ensuite que la suppression a fonctionné et on met un message suivant la réponse.

e) Problèmes rencontrés

i. Installation incomplète à 98%

Lors de l'installation via le Store, l'application est installée à 2%... Quand on installe au travers du script PyScripter l'installation et tous les processus se font en compte

Administrateur, mais quand on installe avec le Store nous sommes en compte System. Et dans l'installation certains processus sont en lien direct avec des serveurs que seul un administrateur peu voir tel que le stockage des applications qui est souvent mis à jour avec les dernières versions des logiciels. En lançant le processus avec un compte System l'accès au serveur de stockage s'enlève grâce (ou à cause) de la gestion des droits avec l'AD.

Solution envisagée : Attribuer des droits de « passage » pour accéder au serveur de stockage mais seulement pour l'application voulant être installé mais aussi seulement pour les applications que l'utilisateur a le droit d'utiliser. Le réseau reste sécurisé et la communication entre les paquets WAPT et le serveur est possible.

Solution choisie : La solution n'a pas encore été acceptée par l'administration, l'équipe de développeurs, et l'équipe système (qui s'occupe de l'AD).

Démarche et source : Avec l'aide de Monsieur VIALE (technicien travaillant principalement sur WAPT et la gestion du store) et la documentation officielle WAPT nous avons pu identifier le problème.

ii. *Désinstallation défectueuse*

Lors de la désinstallation le programme affichait que la désinstallation est une réussite que ce soit sur PyScripter ou le store, sauf qu'aucun fichier n'était supprimé. Après avoir fait beaucoup de recherche, il s'est avéré que la fonction que j'utilisais pour supprimer le répertoire d'installation s'exécute comme-ci j'avais des droits System alors même que l'application est lancée en compte Administrateur.

Solution envisagée et choisie : remplacer cette fonction « `remove_file()` » par « `shutil.rmtree()` » qui est une fonction qui passe au travers des droits Windows, à manipuler avec précaution.

Démarche : Lecture de différents forums sur le net, le forum officiel de Tranquil IT, M. VIALE.

3) Analyse

La désinstallation fonctionne, techniquement l'installation aussi mais elle est freinée par un problème de droit, et il va falloir faire une réunion avec les principaux concernés. Les normes de script ont été appliqué avec succès. La taille du paquet a été réduit de moitié c'est donc une réussite. Il y a trop régulièrement des problèmes de droits et cela pour tous les techniciens, il faudrait repenser la façon dont on sécurise et on cloisonne le réseau de l'entreprise (constatation partager avec plusieurs de mes collègues)

4) Tâches pour la semaine suivante

- Tenter une installation sur base de test de Monstro afin que l'on puisse vérifier si on peut se passer du groupe GGMONSTRO nécessaires pour permettre un accès en lecture/écriture au Q:\Monstro Prod.

-Si possible passer à une autre application WinDev pour valider les acquis.

-Commencer sur une autre application qui n'a pas été développée en WinDev

I. Semaine 3

1) Tâches prévues

-Tenter une installation sur base de test de Monstro afin que l'on puisse vérifier si on peut se passer du groupe GGMONSTRO nécessaires pour permettre un accès en lecture/écriture au Q:\Monstro Prod.

-Remplacer les anciens scripts de désinstallation.

-Commencer d'autres applications non internes.

-Avancer sur les autres applications WinDev

-Faire un point mercredi après-midi sur l'objectif jusqu'au prochain mercredi

2) Tâches réalisées

a) Compte Rendu

Dans le cadre de mon stage il a été convenu avec mon supérieur Monsieur Morcet que l'on fasse le point tous les mercredis lors d'une réunion. Durant cette réunion, je suis amené à apporter ma pierre à l'édifice en donnant mon point de vue et en le confrontant à celui de mes collègues pour savoir qu'elle est la meilleure décision à prendre. De plus, à la suite de cette réunion que j'ai dirigée pour la première fois je dois faire un compte rendu pour mes collègues pour avoir un fil conducteur de réunion en réunion. Grâce à cela, nous pouvons voir notre avancement et nous pouvons voir ce qui a été fait ou pas, ce qui est à faire etc...

b) Script de désinstallation et d'installation application WinDev

i. Application Monstro

M.Rohart, développeur avec qui je travaille sur l'harmonisation des applications métiers, a créé une nouvelle version de l'application Monstro avec la dernière mise à jour WinDev. Le but de ce test était de voir si après modification de l'installateur nous pouvions nous passer de la communication avec le réseau [Q:\](#). L'installation et la désinstallation fonctionnent et même en compte SYSTEM grâce à une directive donnée à l'UAC dans le code d'installation de l'application. L'intégration du nouveau désinstallateur est une réussite, il reste le problème de la désinstallation silencieuse j'attends le retour de mon collègue là-dessus. Il se trouve que d'autres tests ont été faits mais pas de mon côté sur les droits avec l'AD notamment pour cette application

pour prendre des mesures sur les prochaines applications et la gestion avec les droits des utilisateurs, l'UAC, l'AD, le groupware...

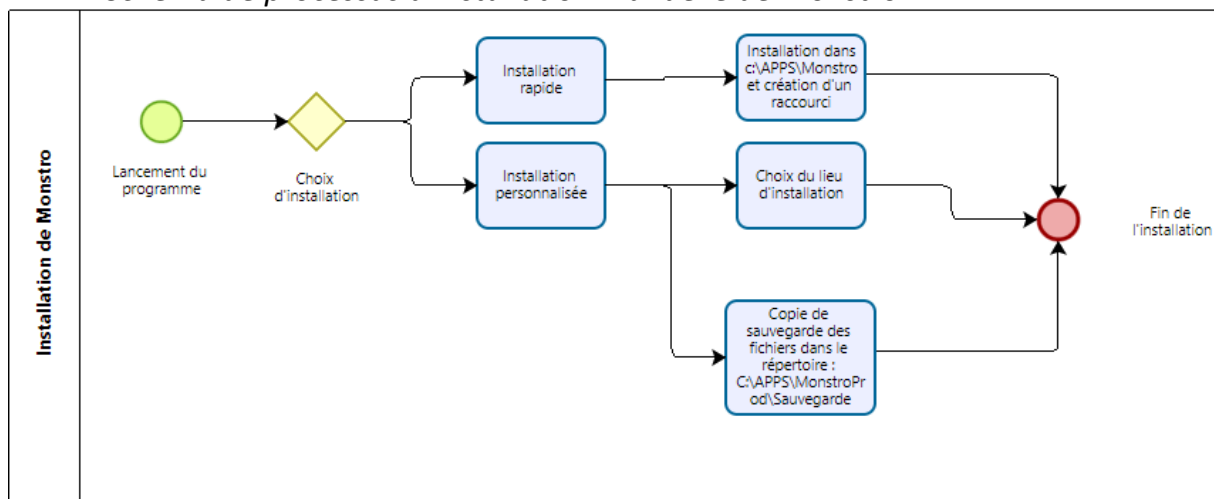
ii. *Application Gestar*

Le script de l'application Monstro a été adapté à l'application Gestar, elles ont un fonctionnement similaire car ce sont des logiciels créés en interne. De plus, elles ont été développées sous WinDev elles ont donc le même fonctionnement au niveau de l'installation.

iii. *Application Omega*

Comme l'application Gestar et Monstro, c'est une application développée sous WinDev. Elle a été plus difficile à mettre en place notamment à cause d'un exécutable mal configuré, elle a donc été plus longue à installer (j'ai dû faire une demande de modification du « .exe »). Elle avait aussi un problème d'encodage au moment d'upload les paquets à cause de la présence d'accents dans la fiche control (Problème inexplicable pour l'instant à part un problème venant de Windows).

iv. *Schéma de processus d'installation manuelle de Monstro:*



Ce schéma représente la manière avec laquelle on doit aborder l'installation d'une application. On doit d'abord comprendre son fonctionnement en version manuelle pour mieux l'adapter en script et faire un véritable Self-Service. C'est donc dans cette démarche que j'ai modélisé le processus d'installation manuelle de l'application Monstro à l'aide du logiciel Bizagi. Dans un futur proche d'autres processus d'installations seront à modéliser, pour créer une base de données des processus d'installation des applications métiers.

c) *Autres applications*

i. *Script de désinstallation*

Beaucoup d'applications métiers sont dépourvues de processus de désinstallation. Ces applications n'ayant pas de processus d'applications exploitable, beaucoup de paquet n'ont donc pas de script de désinstallation. J'ai donc créé un script de désinstallation n'ayant pas besoin d'exécutable de désinstallation. De plus, je l'ai créé

de tel sorte à ce qu'il soit fonctionnel même sur des logiciels qui n'ont pas été développé sous WinDev. Je l'ai implémenté sur les applications : Regicette, Logipieces, Vision+, Numacte et Gestim.

PS : D'autres modifications dans les scripts d'installation ont été faites mais elles sont mineures.

ii. **Application ControleA**

J'ai travaillé sur l'application ControleA, un logiciel qui n'a pas été développé en interne. J'ai passé plusieurs jours sur l'analyse de son installation qui n'est pas intuitive et qui demande en installation manuelle particulière. Après en avoir discuté avec un technicien qui a déjà travaillé dessus il a été décidé que je passe à une autre application. La raison est que pour le moment, le seul moyen d'installer cette application avec un script c'est exclusivement avec du Batch. De plus, l'application n'étant plus mise à jour régulièrement par son créateur, ayant de nombreux bug et étant capricieuse à utiliser, elle va être remplacée.

d) **Réunion**

i. **Problème abordé en réunion**

J'ai pu observer après de nombreuses désinstallations que certaines applications apparaissent dans la liste des programmes installés de Windows. La présence de ces logiciels même après la désinstallation montre une faille de mon script. Celui-ci supprime directement les fichiers du logiciel dans un emplacement précis du disque C:\, ce n'est pas un exécutable qui va chercher des « traces » du logiciel comme le fichier WDUinst.exe qui va chercher le logiciel sur l'ordinateur. De plus, dans l'exécutable il est configuré la suppression des clés de registre et de toutes les informations liées au logiciel dans la base de données de Windows. Pour l'instant aucunes décisions n'ont été prise à voir pour la suite.

3) Analyse

J'ai pu constater durant cette semaine la difficulté de s'adapter à chaque logiciel qui existent et qui ne sont pas développé de la même façon. Il y a aussi la problématique du comportement de Windows qui parfois n'accepte pas certaines méthodes d'encodage etc. Il faut constamment se poser les bonnes questions par rapport à nos acquis, les nouvelles choses que l'on apprend nous permettent d'harmoniser d'anciennes méthodes et façons de faire. Il faut aussi penser aux personnes qui vont nous remplacer et vont devoir comprendre les raisons qui nous ont poussé à faire telle ou telle chose.

4) Tâches pour la semaine suivante

-Modéliser le processus d'installation des applications déjà traités et à traiter dans le futur.

-Voir les applications : Arkheia, Melodi, Spectacle etc.

-Voir le problème de Web Socket Ixbus (nouveau dans le travail)

- Préparer la prochaine réunion mercredi
- Avancer dans la documentation d'automatisation
- Se renseigner sur la marche à suivre pour les applications WinDev
- Commencer à se renseigner sur le parc Informatique

I. Semaine 4

1) Tâches prévues

- Tester la nouvelle version de l'application Monstro
- Modifier les script d'installation des applications WinDev qui ont été mis en ligne sur la plateforme WAPT.
- Modifier les icônes défectueuses
- Préparer la réunion et faire le compte rendu
- Étudier les applications ControleA et Arkheia
- Avancer dans la documentation sur les applications waptisables.

2) Tâches réalisées

e) Formalisme et amélioration des données

Tous les jours mon maître de Stage a des réunions. Dans ces réunions il y a un moment consacrer sur mon travail et des suggestions peuvent-être faites pour améliorer la communication entre les services ou pour mieux transmettre de l'information de mon travail. De plus, il peut y avoir aussi des demandes du Directeur comme la petite DataBase.

- i. *Organisation d'une petite base de données des applications traités*
Un répertoire sur le réseau a été spécialement créé pour y stocker mon travail. C'est donc dans celui-ci que je « range » mes comptes rendu, ma documentation sur l'automatisation des applications WinDev (Décision prise en réunion la documentation ne concerne plus que les applications WinDev), les différents script d'applications et leurs icônes, la liste des applications WinDev, la liste des applications à traiter.
- ii. *Amélioration d'icônes*
A la demande de mon collègue Monsieur Viale, j'ai modifié des icônes qui à cause d'un problème d'affichage étaient visibles en rose fluo dans le Store. Par soucis d'esthétisme j'ai donc modifier ces icônes.

f) Application Monstro

- i. *Test de l'application Monstro version 15.0.0.6*
Suite aux discussions sur Webex, par mail et en réunion, Monsieur Rohart (Développeur) a créé une nouvelle version de l'application Monstro, qui intègre une une désinstallation silencieuse et l'installation se passe au travers de l'UAC et donc devrait résoudre les problèmes de droit. L'exécutable de désinstallation semble être

la meilleure manière de désinstaller un logiciel, cela étant dû au fait que le script de désinstallation que j'ai fait ne prend pas en charge la suppression de clé de registre WinDows. Le problème étant qu'aux « yeux » de Windows si la clé de registre est encore présente alors il considère que l'application est toujours sur le disque dur, sauf qu'il est possible que l'application soit déjà désinstallée. Pour remédier à ce problème il a été décidé dans la semaine de créer la version 15.0.0.7 qui prend en charge dans son désinstalleur la suppression des clés de registre. Si en exécutant manuellement l'exécutable, la désinstallation fonctionne, en script c'est une autre histoire. En effet, la « clé » permettant de lancer la désinstallation silencieuse ne fonctionne pas. La solution envisagée a été de recréer une nouvelle version avec une clé fonctionnelle et une autre manière d'installer l'application.

ii. *Test de l'application Monstro version 15.0.0.7*

Dans cette version l'installation et la désinstallation se font silencieusement et s'exécutent en compte administrateur pour la durée des tests. De plus, dans cette version la désinstallation supprime aussi les clés de registre de Windows pour avoir une désinstallation complète de l'application sur l'ordinateur de l'agent. Suite à la désinstallation complète de l'application, un problème est survenu lors de la réinstallation de celle-ci. En effet, pour installer une nouvelle fois le logiciel il faut redémarrer l'ordinateur ! J'attends un retour de Monsieur Rohart sur la raison de ce redémarrage.

g) Applications WinDev

Certaines applications WinDev ont été mises en ligne sur le Store avec un script qui certes les installent et les désinstallent mais qui ne prend pas en compte les mises à jour du logiciel. L'intérêt de mon script est qu'il prend en compte les mises à jour et est moins lourd. Néanmoins il n'est toujours pas fonctionnel à cause des problèmes de droits.

h) Préparation de la réunion

Chaque semaine je prépare une réunion et durant celle-ci la direction de travail est donné, les applications à traiter, les icônes à refaire, les documentations à avancer etc. Le compte rendu permet aux personnes qui n'ont pas assisté à la réunion d'avoir un bref résumé des idées abordées et de savoir qu'elles décisions ont été prises, cela étant aussi valable pour les personnes présentes.

i) Applications propriétaires

i. *Application ControleA*

Après avoir passé beaucoup de temps sur cette application, j'ai conclu par son architecture et son mode d'installation qu'elle ne rentrait pas dans une démarche de self-service et donc d'automatisation. En effet, celle-ci demande un certain nombre

de manipulations manuelles qui ne sont pas automatisable. De plus, c'est une application peu utilisée.

ii. *Application Arkheia*

Comme pour l'application ControleA, le logiciel Arkheia ne semble pas Waptisable. En effet, elle aussi demande beaucoup de modification manuelle et même plus qui ne me semblent pas automatisable (Documentation d'installation manuelle de 50 pages...)

Conclusion sur les deux applications : Même si pour l'instant il a été convenu avec d'autres collègues que ces applications semblent difficilement automatisable, il est toujours possible de revenir dessus plus tard.

3) Analyse

Parfois on passe beaucoup de temps sur certaines applications, scripts, ou solutions pour au finale en trouver une autre plus simple ou encore passer à une autre application car celle-ci n'est juste pas automatisable.

4) Tâches pour la semaine suivante

- Organiser une réunion inter-service pour trancher sur la marche à suivre lors de l'installation d'une application WinDev et de sa désinstallation, notamment le problème des clés de registre. Mais aussi poser le problème des mises à jour automatique et du lien entre le réseau Q et le fichier « WDUPDATE.NET ».
- Commencer à travailler sur les applications Gescime et Maestro.
- Étudier le problème du Web Socket avec IXBUS.
- Tester les mises à jour via serveur Web
- Continuer la documentation d'automatisation des applications WinDev si il y a du nouveau.
- Continuer de faire les schémas de processus d'installation.

II. Semaine 5

5) Tâches prévues

- Organiser une réunion inter-service pour trancher sur la marche à suivre lors de l'installation d'une application WinDev et de sa désinstallation, notamment le problème des clés de registre. Mais aussi poser le problème des mises à jour automatique et du lien entre le réseau Q et le fichier « WDUPDATE.NET ».
- Commencer à travailler sur les applications Gescime et Maestro.

- Étudier le problème du Web Socket avec IXBUS.
- Tester les mises à jour via serveur Web
- Continuer la documentation d'automatisation des applications WinDev si il y a du nouveau.
- Continuer de faire les schémas de processus d'installation.
- Applications : Logement, A fleur de ville
- Logo A fleur de ville
- Informations actions manuelles d'installation simplifiée de Gescime
- Nouvelle version de Monstro et Tests de l'application.
- Contacter les prestataires des applications ControleA et Arkheia
- Faire un compte rendu de la réunion

6) Tâches réalisées

j) Quelques précisions

Ayant une réunion tous les mercredis de nouvelles suggestions sont faites, de nouvelles applications sont à automatiser etc. Ce qui explique l'écart qu'il y a entre le prévu et le réalisé presque permanent que j'ai. De plus, je dois m'adapter aux demandes faites par mail pour améliorer tels ou tels applications ou pour m'occuper d'une application en particulier.

k) Amélioration du script

Vient le problème suivant :

Problème : Durant mes tests sur l'application Monstro version 15.0.0.13 j'ai pu constaté que mon script passé au travers de mes conditions comme-ci elles étaient null ou ne fonctionnaient pas. J'ai donc chercher

Solution Envisagée : Revoir mon script il n'a pas été conçu pour fonctionner avec un désinstallateur mais bien pour être autonome de tout exécutable.

Solution : Avec la venue de désinstallateurs viable il s'est avéré que ma condition « `if os.path.isdir(InstallRepository):` » n'était plus opérationnelle. En effet, celle-ci vérifie l'existence du dossier par son nom attribué à la variable « `InstallRepository` ». L'exécutable supprime les fichiers de l'application dans son dossier d'installation mais ne supprime pas le dossier en lui même ce qui posera problème dans mon script au moment de la vérification de la présence ou nom de cette application. J'ai donc remplacer cette fonction par celle-ci « `os.path.isfile(Install)` ». Cette fonction retourne aussi un booléen mais ici on ne vérifie la présence d'un dossier mais de l'exécutable de l'application la faisant fonctionner dans le dossier d'installation. Cela me permet de m'adapter parfaitement aux nouvelles décisions. (Une update sera à faire sur tous mes scripts, le but de mon script est qu'il s'adapte le plus possible à tous types d'applications, avec cette nouvelle fonction cela est possible)

I) Applications Traitées

i. *Municipol*

Problème : L'installation se fait parfaitement bien, aussi bien en version Test qu'en version Prod. Toutefois l'application empêche les utilisateurs de se connecter à la base de données.

Hypothèse : J'ai l'impression qu'il a beaucoup de fichiers permettant la création d'une interface, mais ceux-ci ne sont présents que sur le réseau mais pas dans le dossier d'installation créé par l'exécutable d'installation. Je pense qu'il a des fichiers à ajouter soi-même.

Solution envisagée : Vérifier les fichiers de l'application pour trouver la source du problème et ajouter les fichiers manquant et obligatoire à l'installation. Le but n'est pas de copier l'intégralité des fichiers présents sur le réseau [Q:\](#). En effet, le réseau sert aussi bien de base données pour alimenter les applications mais aussi de stockage aux exécutable d'installation et de désinstallation de chaque applications.

Solution : Un collègue qui a travaillé sur l'application m'a aidé et ma notamment expliqué deux problèmes : à la base l'application fonctionne sous serveur avec sa clé de registre intégrée au serveur (lors de ma copie de fichiers d'installation la clé de registre n'est pas intégrée car elle n'est pas visible pour ma part, il me l'a donc fournit), le dossier d'installation fournit les fichiers primordiaux à la connexion à la base de données, mais celui-ci ne fournit pas les fichiers d'interfaces qui permettent la navigation dans l'application (j'ai pu bénéficier du setup donné par le prestataire).

ii. *Gescime*

Problème : Application à copier beaucoup trop lourde(8Go)

Solution : Demande de nettoyage des fichiers de l'application à une collègue

Suite au nettoyage des fichiers de l'application j'ai pu copier les fichiers d'exécution de l'application. J'ai donc intégré la version Test et Prod au paquet WAPT avec la création de deux raccourcis. L'application est maintenant passée en production, c'est à dire que tous les agents ont maintenant accès à l'application.

iii. *Base écoles*

Problème : L'exécutable d'installation ne veut pas installer dans le répertoire d'installation mais dans la racine du C.

Solution envisagée : Forcer l'installation dans le répertoire.

Solution : Faire modifier l'exécutable qui n'est pas métrisable.

J'ai pu discuter avec les développeurs au sujet de cette application et il a été décidé de ne pas l'automatiser car elle va bientôt être remplacée.

iv. *Monstro(version 15.0.0.11)*

Problème : Nouvelle version de l'application qui lors de l'installation crée deux répertoires d'installation : Monstro Test, Monstro Prod

Solution envisagée: Demander un nettoyage du désinstalleur.

Solution : Monsieur Rohart (développeur des applications WinDev) qui m'aide dans la démarche de trouver une solution viable pour les applications WinDev qui doivent être automatisées dans le futur. Dans cette idée il a créé plusieurs versions de l'application Monstro et il a aussi fait plusieurs tests de ces différentes versions. La dernière version qu'il m'a donné est censée régler tous les problèmes aperçu par le passé. En effet, dans cette version certains problèmes des versions précédentes sont corrigés tels que : redémarrage pour réinstaller l'application après une désinstallation, problème des droits SYSTEM lors de l'installation manuelle, nettoyage des fichiers prod des anciennes versions qui parasitaient la version actuelle.

v. *A fleur de ville*

Cette application est une application WinDev j'ai donc adapter mon script pour applications Windev avec les dernières modifications. J'ai aussi modifié son icône pour le Store.

7) Analyse

La gestion des droits et des clés de registre ainsi que leur suppression est très compliqué à gérer parmi toutes les sécurités présentes dans l'entreprise pour contourner les piratages.

8) Tâches pour la semaine suivante

- Continuer les applications : Monstro (version 15.0.0.13), Municipol(finaliser l'installation du paquet), Logement, continuer l'application NutriData
- Mettre à jour tous les scripts avec la nouvelle fonction.
- Se renseigner sur le parc informatique de la mairie
- Continuer la documentation WinDev
- Si les tests sur Monstro sont concluant alors il faut appliquer les modifications de script à toutes les autres applications WinDev

III. Semaine 6

9) Tâches prévues

- Revenir sur l'application Municipol
- Avancer sur l'application Monstro avec les nouvelles décisions prises en réunions
- Réaliser une étude du patrimoine informatique de la mairie Tours pour Mme Ferreira
- Schéma de l'installation de Monstro via la console WAPT et via le Store
- Modification de la documentation sur l'automatisation des applications WinDev
- Compte rendu de la dernière réunion

10) Tâches réalisées

m) Applications Traitées

i. *Municipol*

Problème : Le setup installe l'application à la racine du [C:\](#) et non pas dans APPS.

Solution : Le prestataire devrait être contacté pour adapter son exécutable à nos besoins.

ii. *Monstro*

Suite à une observation d'un collègue sur le problème de droits qui nous empêchaient d'installer une quelconque application WinDev, il a été décidé de contourner les droits. En effet, plutôt que d'exécuter le setup d'installation du logiciel à partir du paquet, on le copie dans [C:\APPS](#) puis on l'exécute. Cela permet de ne pas subir les droits du compte SYSTEM lors de l'installation avec le store, ici le processus d'installation s'exécute en tant qu'administrateur. Grâce à cela, l'application se met à jour seule grâce à son processus de recherche de mise à jour (WDUPDATE.NET).

Premier problème : Après une première installation le setup d'installation reste dans APPS. Si il n'y avait qu'une seule application WinDev à installer cela ne serait pas un problème mais il y en a plusieurs avec leurs propre système d'installation. Pour éviter le plantage d'une installation il a été décidé de supprimer le setup d'installation une fois sa tâche accomplie. Le problème est que le script python n'attend pas d'avoir terminé l'installation qu'il supprime le setup. Une erreur survient, les fichiers de l'application sont incomplets il faut donc recommencer.

Solution envisagée : Mettre un `time.sleep()` permettant de créer un delay entre la suppression et l'installation. Cela permettrait de laisser l'installateur terminer son processus normalement et ainsi de le supprimer après sa tâche effectuée.

Solution : L'utilisation de la fonction `time.sleep(10)` permet à l'installateur de fonctionner correctement durant l'installation et dans un second temps la suppression du setup pour ainsi faire une installation dite « propre ». (Demande faite aux développeurs si cette suppression ne pourrait pas se faire en interne dans les fichiers du setup de l'application)

Deuxième problème : Au niveau du résultat la solution envisagée est bonne. Sauf qu'il ne faut pas oublier à qui est destiné ce type d'applications. En effet, les agents travaillant avec ces applications n'ont pas forcément un poste avec un environnement stable. Si la machine de l'utilisateur est peu puissante l'installateur mettra plus de temps et le `time.sleep()` risque de ne pas être assez long. De plus, si l'agent n'est pas patient il peut penser que l'installation ne fonctionne pas et donc annuler le processus d'installation, et il créera probablement un ticket pour signaler le dysfonctionnement de l'application. Le but de l'automatisation des applications via WAPT est de libérer du temps aux techniciens avec du self-service, si le self-service pousse ses utilisateurs à faire appel au support on perd toute l'utilité de la manœuvre.

Solution envisagée : La première venant d'un collègue serait de faire une boucle while vérifiant si différents processus sont toujours en cour et ainsi retarder la suppression. Cette méthode implique l'utilisation de la librairie « subprocess », et la création d'une méthode permettant de détecter les processus en cour. Après de nombreuses tentatives cette méthode a été écartée. Autre méthode serait de faire une vérification de la présence ou non du processus d'exécution de l'application dans ces fichiers. Si oui, le setup est supprimé si non, il ne fait rien. Cette seconde méthode a aussi été écartée car le script est trop rapide et ignore cette condition.

Solution : Faire une vérification de la présence ou nom du dossier « INSTALLEUR » (nom du dossier comportant le setup d'installation). Si il est trouvé, il est remplacé par celui de l'application en cour d'installation. De plus, à chaque démarrage de l'ordinateur WAPT supprimera tout fichier s'appelant « INSTALLEUR », ça permet de ne pas avoir de conflit d'installateurs.

n) Étude de la gestion du patrimoine informatique

J'ai pu questionner mes collègues et ainsi obtenir des informations sur le parc informatique de la DSI pour pouvoir faire mon étude.

11) Analyse

Après de nombreuses réunion, rapports, compte-rendu, des décisions ont été prises et on voit de l'avancement dans l'automatisation des applications WinDev. Et j'ai pu aussi construire une base de données des applications que j'ai pu traiter, je laisse donc derrière moi une aide pour mes collègues et indirectement futurs collègues.

12) Tâches pour la semaine suivante

- Faire un powerpoint pour présenter mon travail aux directeurs
- Adapter la solution WinDev appliqué à Monstro à toutes les applications WinDev

IV. Semaine 7

13) Tâches prévues

- Faire un powerpoint pour présenter mon travail aux directeurs
- Adapter la solution WinDev appliqué à Monstro à toutes les applications WinDev

14) Tâches réalisées

-J'ai fais un PowerPoint pour présenter mon travail aux directeurs ce qui leur a permis de mieux comprendre ce que j'ai fais durant cette période de stage et ainsi voir tout le travail effectué.

-Création de deux scripts en python 2,7 l'un fonctionne sans désinstallateur l'autre avec. Ils sont tous les deux fonctionnels à l'automatisation d'installation, de désinstallation et de mise à jour de logiciels métiers WinDev.

