

MANUEL D'INSTALLATION ET D'ADMINISTRATION

Environnement technique	:	Systeme d'exploitation : Microsoft Windows Seven (32/64) Navigateur : Sans objet Langages : Python, SQL Base de données : PostgreSQL Serveur Web : Sans objet Bibliothèques : Qt, PyQt Outils : Compilateur C/C++
--------------------------------	---	---

TABLE DES MATIERES

1	AVANT-PROPOS	4
1.1	Aspect sécuritaire.....	4
1.2	Pourquoi ce document ?.....	4
1.3	A qui s'adresse-t-il ?.....	4
1.4	Références.....	4
1.5	32/64 bits.....	4
1.6	Racine des logiciels	4
2	Remarques avant installation.....	4
3	Installation du serveur de base de données "PostgreSQL"	5
3.1	Pré-requis.....	5
3.2	Installation du serveur PostgreSQL sur système d'exploitation Windows	5
3.2.1	Programme d'installation.....	5
3.2.2	Influence sur le système et sa sécurité.....	7
3.2.3	Démarrage et/ou arrêt du serveur	8
3.2.4	Interaction avec les pare feux	8
3.2.5	Paramétrage de l'environnement.....	9
3.2.6	Modification du dossier de stockage des bases de données	11
3.2.7	Installation de l'outil géographique PostGIS	12
3.3	Installation du serveur PostgreSQL sur système d'exploitation Linux.....	13
4	Installation de la librairie Qt.....	14
4.1	Pré-requis.....	14
4.2	Installation de la librairie Qt sur système d'exploitation Windows	14
4.2.1	Installation du compilateur.....	14
4.2.2	Paramétrage de l'environnement.....	15
4.2.3	Installation de la librairie Qt.....	17
4.2.4	Paramétrage de l'environnement.....	19
4.3	Installation de la librairie Qt sur système d'exploitation Linux.....	20
5	Installation de l'interpréteur Python.....	21
5.1	Pré-requis.....	21
5.2	Installation de l'interpréteur Python sur système d'exploitation Windows.....	21
5.2.1	Déroulement du programme d'installation de l'interpréteur de base	22
5.2.2	Paramétrage de l'environnement.....	23
5.2.3	Vérification de la bonne installation de Python	25
5.2.4	Le module Psycopg2	26
5.2.5	Le module Numpy.....	27
5.2.6	Le module sip	28
5.2.7	Le module PyQt.....	29
5.2.8	Vérification de la bonne installation de PyQt.....	30
5.2.9	Influence sur le système et sa sécurité.....	30
5.3	Installation de l'interpréteur Python sur système d'exploitation Linux.....	30
6	Administration de la base de données.....	31
6.1	Bref aperçu des fichiers de configuration PostgreSQL	31
6.1.1	Fichier postgresql.conf	31
6.1.2	Fichier pg_hba.conf.....	31
6.1.3	Fichier pg_ident.conf.....	31
6.1.4	En cas de non démarrage du service PostgreSQL.....	32
6.2	Modification du mot de passe postgres.....	32
6.3	Sauvegarde et restauration d'une base de données	32
6.3.1	Sauvegarde	32

6.3.2 Restauration 35
6.4 pgAdmin III 36
7 GLOSSAIRE 37



1 AVANT-PROPOS

1.1 Aspect sécuritaire

Ce document n'a pas de classification particulière.

Une version numérique de ce document se trouve dans le dossier "doc" situé à la racine du cdrom d'installation.

1.2 Pourquoi ce document ?

Ce document est le manuel d'installation des outils informatiques utilisés pour faire tourner les applications PPQ (PostgreSQL/Python/Qt). Il est donc destiné aux différents administrateurs de ce système.

1.3 A qui s'adresse-t-il ?

Ce manuel est destiné aux différents administrateurs des systèmes ci-dessus.

1.4 Références

Ce manuel ne fait référence à aucun autre

1.5 32/64 bits

Dans un soucis de compatibilité ascendante, l'ensemble des logiciels utilisés est prévu pour fonctionner sur un os 32 bits (ce qui lui permet de fonctionner aussi sur un os 64 bits).

1.6 Racine des logiciels

L'ensemble des logiciels utilisés se trouve dans le dossier "logicielsTiers" se trouvant à la racine du cdrom d'installation. Tout dossier relatif de ce document sera relatif à ce dossier "logicielsTiers".

2 REMARQUES AVANT INSTALLATION

Certains logiciels à installer n'acceptent pas d'espace ni dans le chemin du dossier les contenant, ni dans le chemin du dossier d'installation. Ainsi le dossier "C:\Program Files" habituel de Windows ne convient pas.

C'est pourquoi, dans ce document, l'ensemble des logiciels sera installé dans le dossier "C:\Development". Et si, pour des raisons de rapidité ou autre, vous décidez de copier le contenu du dossier "logicielsTiers" sur disque, veillez à le positionner dans un dossier ne contenant aucun espace dans son chemin complet ("C:\temp" par exemple).

3 INSTALLATION DU SERVEUR DE BASE DE DONNEES "POSTGRESQL"

3.1 Pré-requis.

Ce chapitre ne concerne que l'ordinateur destiné à accueillir un serveur de base de données PostgreSQL et n'en possédant pas déjà un (auquel cas, le serveur existant pourra très bien accueillir aussi les bases de données).

Les différents fichiers d'installation se trouvent dans le dossier "logicielsTiers\PostgreSQL".

La version du moteur PostgreSQL présent dans ce cdrom est la 9.3.05 qui date de novembre 2014. Tout ce chapitre explicatif est basé sur la version 9.3.05 et on retrouvera ces chiffres dans les noms des divers outils PostgreSQL.

Lorsque les versions suivantes de PostgreSQL (disponibles sur le site "<http://www.postgresql.org>") seront disponibles, les numéros ayant évolués, il conviendra, lors de la lecture de ce document, de remplacer dans ce chapitre les chiffres "9.3.05" par ceux correspondants à la version de PostgreSQL en cours.

3.2 Installation du serveur PostgreSQL sur système d'exploitation Windows

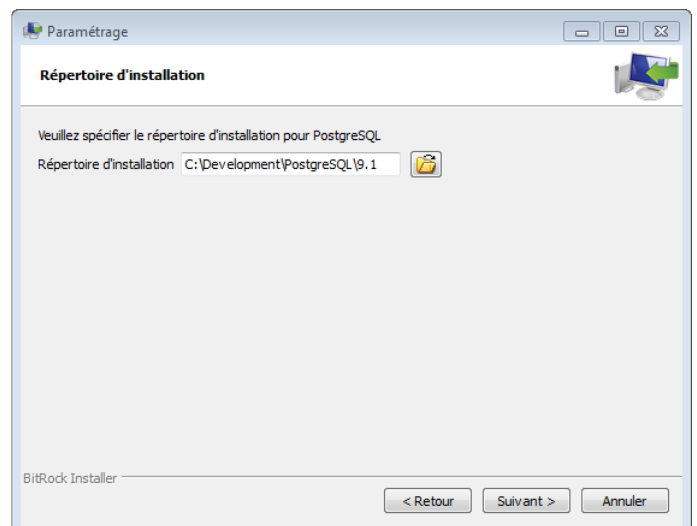
L'installation du serveur PostgreSQL est de la responsabilité de l'administrateur du réseau sur lequel le serveur sera installé. Le programme d'installation se nomme "postgresql-9.3.05-windows.exe" et possède la signature md5 "a3d813b288ee133b7381657c98cec3f1".

3.2.1 Programme d'installation

Exécuter le programme "postgresql-9.3.05-windows.exe"
Une fois les premiers éléments chargés, le programme propose la fenêtre suivante...

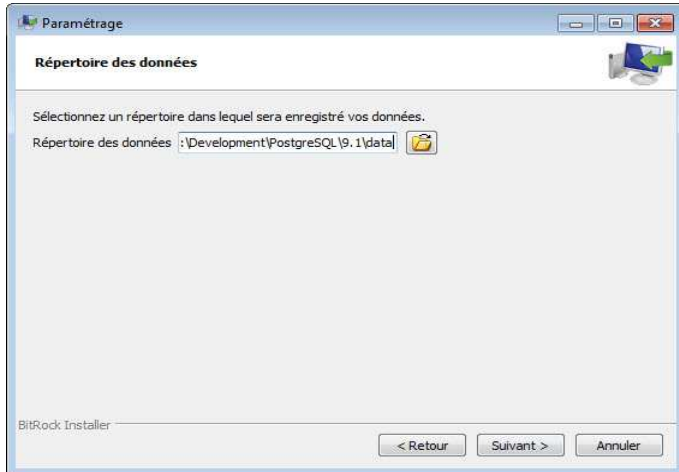


... qui n'a rien de particulier. Il suffit donc de cliquer sur le bouton "Suivant" ce qui amène sur la fenêtre suivante...



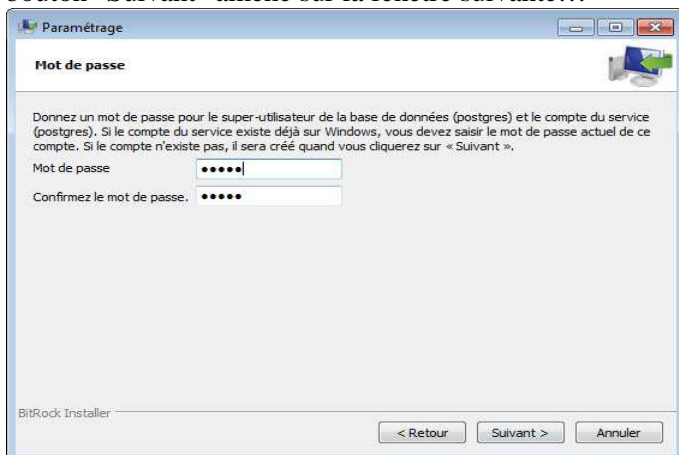
... où il faut alors indiquer dans quel dossier installer le serveur PostgreSQL. Le choix offert par défaut est "C:\Program Files\PostgreSQL\9.3". L'installateur peut cependant utiliser à loisir tout dossier qui lui semble correct; l'emplacement de ce dossier d'installation n'influera pas sur les performances de PostgreSQL. Conformément aux remarques préliminaires concernant le chemin du dossier d'installation des logiciels, nous choisirons d'installer le serveur dans "C:\Development\PostgreSQL\9.3".

Une fois l'emplacement du serveur choisi, l'appui sur le bouton "Suivant" amène sur la fenêtre suivante...



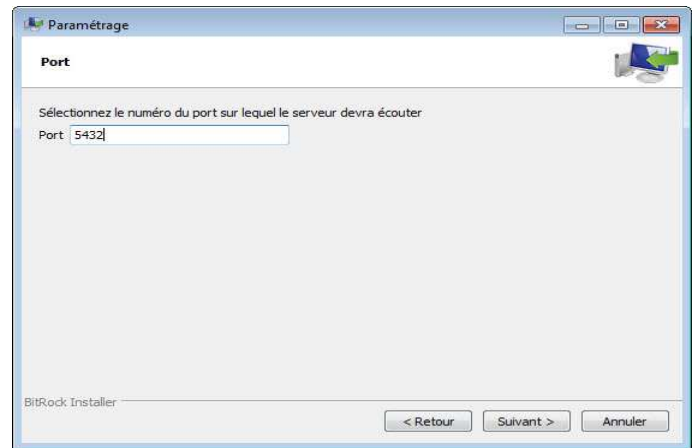
... où il faut indiquer l'emplacement où seront stockées les bases de données qui seront gérées par le serveur. Le choix offert par défaut est un dossier "data" dans le dossier choisi à l'étape précédente, à savoir "C:\\Development\\PostgreSQL\\9.3". Cet emplacement est à priori définitif toutefois il est possible de le modifier par la suite au travers de la base de registre Windows.

Une fois l'emplacement des données choisi, l'appui sur le bouton "Suivant" amène sur la fenêtre suivante...



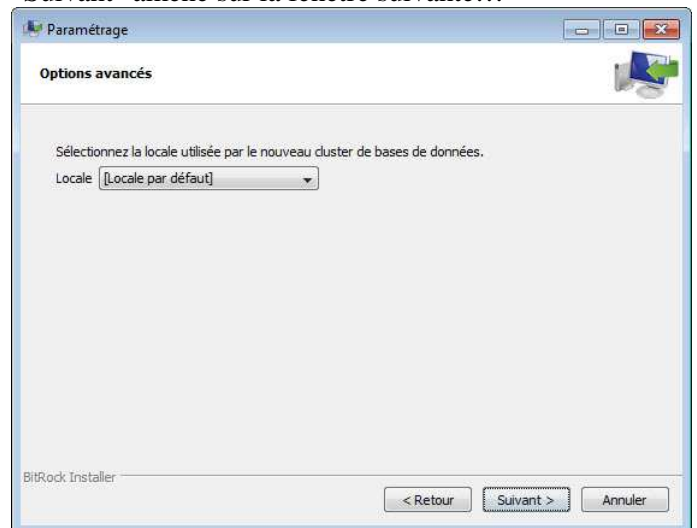
... où il faut indiquer le mot de passe de l'administrateur PostgreSQL. Ce mot de passe est à choisir avec rigueur et intelligence car c'est lui qui permet toute l'administration du serveur. Toutefois il est modifiable ultérieurement. A noter par ailleurs que ce mot de passe doit en plus correspondre aux contraintes éventuelles de complexité du système si celles-ci ont été activées.

Une fois ce mot de passe choisi et enregistré, le bouton "Suivant" amène sur la fenêtre suivante...



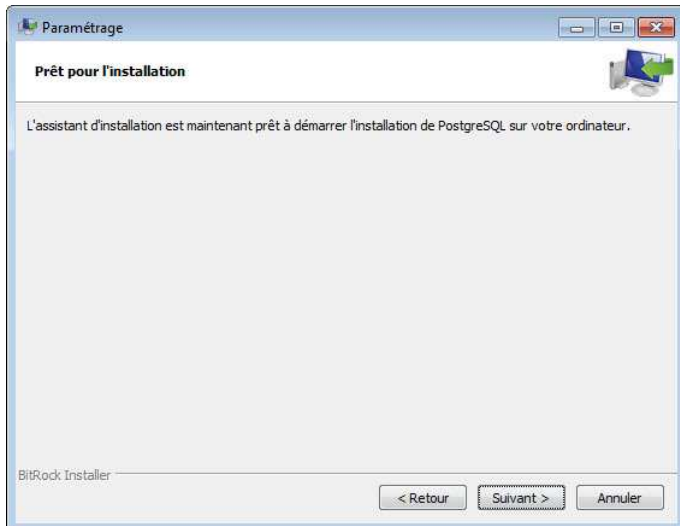
... où il est demandé sur quel port le serveur devra attendre les connections. Le choix conventionné est 5432 mais tout autre numéro de port non utilisé par un autre service réseau peut être utilisé à la place.

Une fois ce n° de port validé, l'appui sur le bouton "Suivant" amène sur la fenêtre suivante...

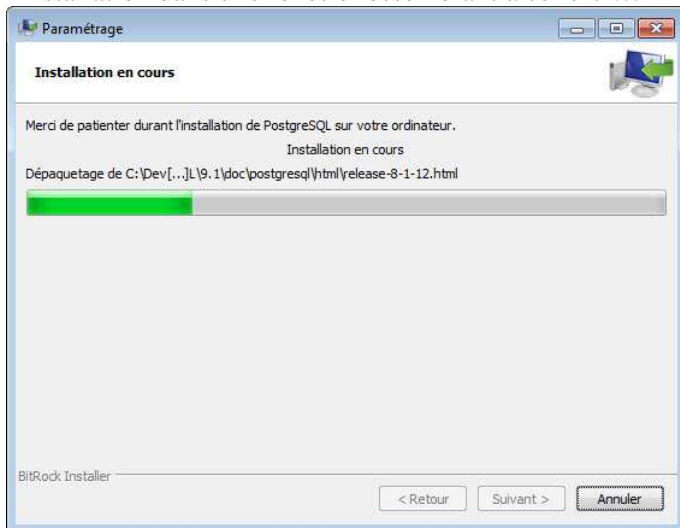


... où on indique dans quel encodage de caractères devra travailler le serveur PostgreSQL. Le choix proposé par défaut "[Default locale]" indique au serveur PostgreSQL d'utiliser l'encodage du système sur lequel il se trouve.

Ce choix sera le dernier et amène sur la fenêtre suivante...



... qui indique que le serveur est maintenant prêt à être installé. L'appui sur le bouton "Suivant" déroule alors l'installation dans une fenêtre ressemblant à celle-ci...



3.2.2 Influence sur le système et sa sécurité

L'installation du serveur PostgreSQL entraîne la création d'un utilisateur Windows nommé "postgres" qui ne fait partie d'aucun groupe (et donc qui n'est pas administrateur). Cet utilisateur est nécessaire car c'est à son nom qu'est lancé le service nécessaire à PostgreSQL.

Comme tout utilisateur Windows, il lui est créé aussi un dossier "C:\Users\postgres" contenant tous les dossiers standard d'un utilisateur lambda (Desktop, Images, Téléchargement, Vidéos, etc...)

... et se terminant par celle-là...

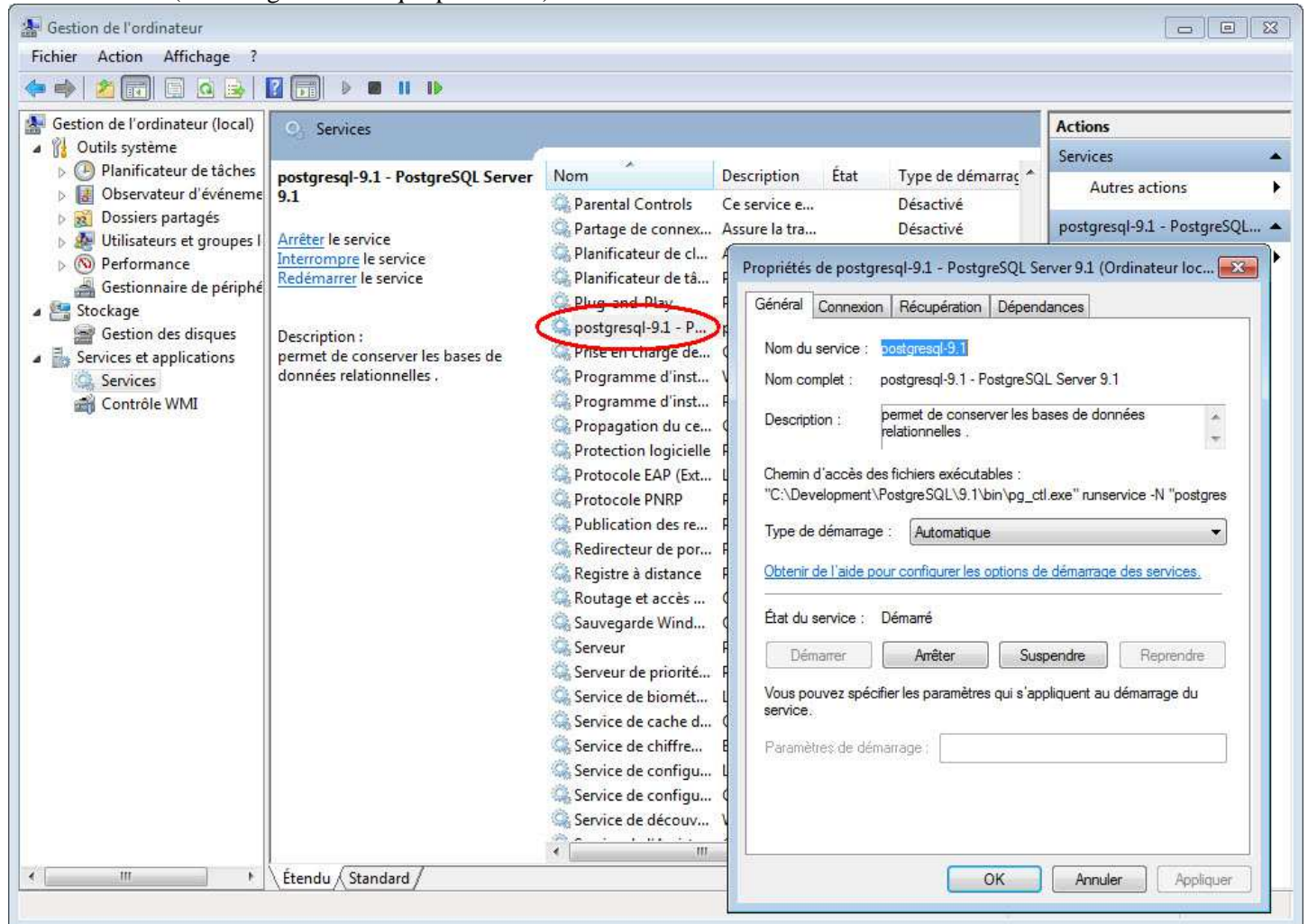


... indiquant que l'installation du serveur est maintenant terminée.

Le programme "Stack Builder" qui est coché par défaut est un programme d'installation d'outils additionnels pour PostgreSQL qui sont alors récupérés sur Internet. Comme ces outils ne sont pas utiles ici et que l'accès à Internet n'est pas garanti, il est donc conseillé de décocher cette option et cliquer ensuite sur "Terminer" pour clore l'installation de PostgreSQL.

3.2.3 Démarrage et/ou arrêt du serveur

Le serveur PostgreSQL est géré par un service Windows nommé "postgresql-9.3" qui permet d'automatiser son démarrage avec Windows (démarrage automatique par défaut).



3.2.4 Interaction avec les pare-feux

Dans le cas où un pare-feu serait installé sur le système, il sera alors nécessaire d'autoriser les programmes pg_ctl, postgres et psql à accéder au réseau.

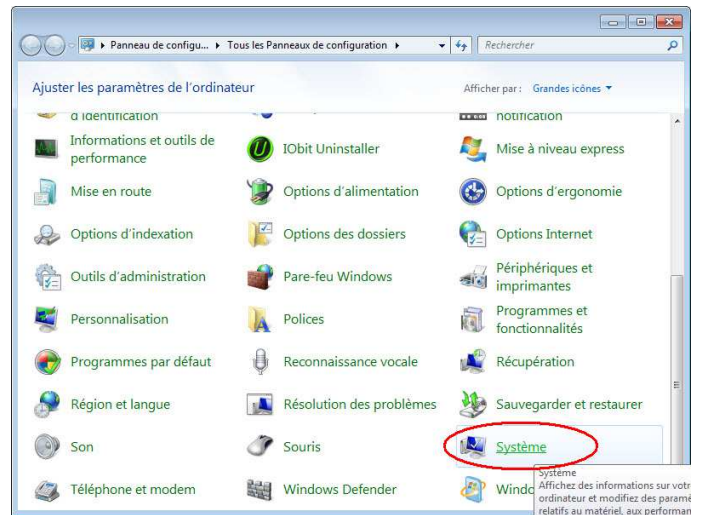
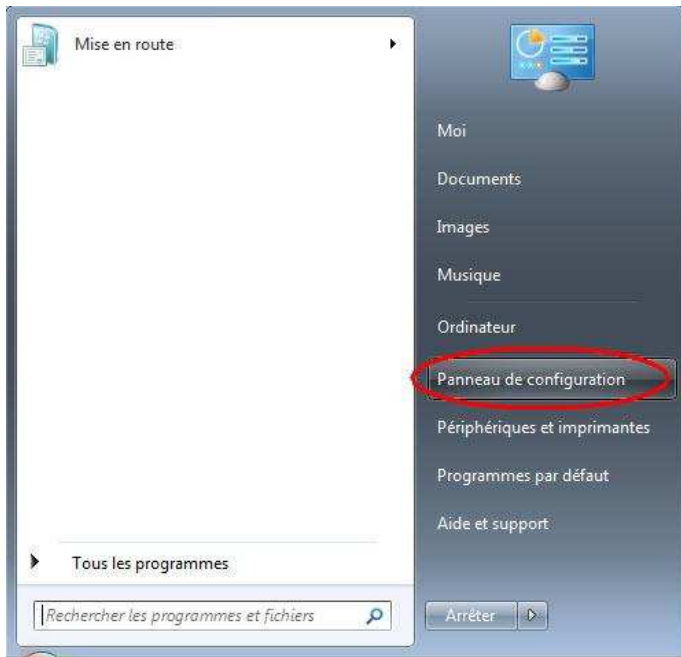
Le programme pg_ctl est le contrôleur de PostgreSQL. Le programme "postgres" est le serveur proprement dit et le programme "psql" est un programme permettant d'interroger le serveur en mode "lignes de commandes". Ces 3 programmes nécessitent un accès ouvert au réseau.

3.2.5 Paramétrage de l'environnement

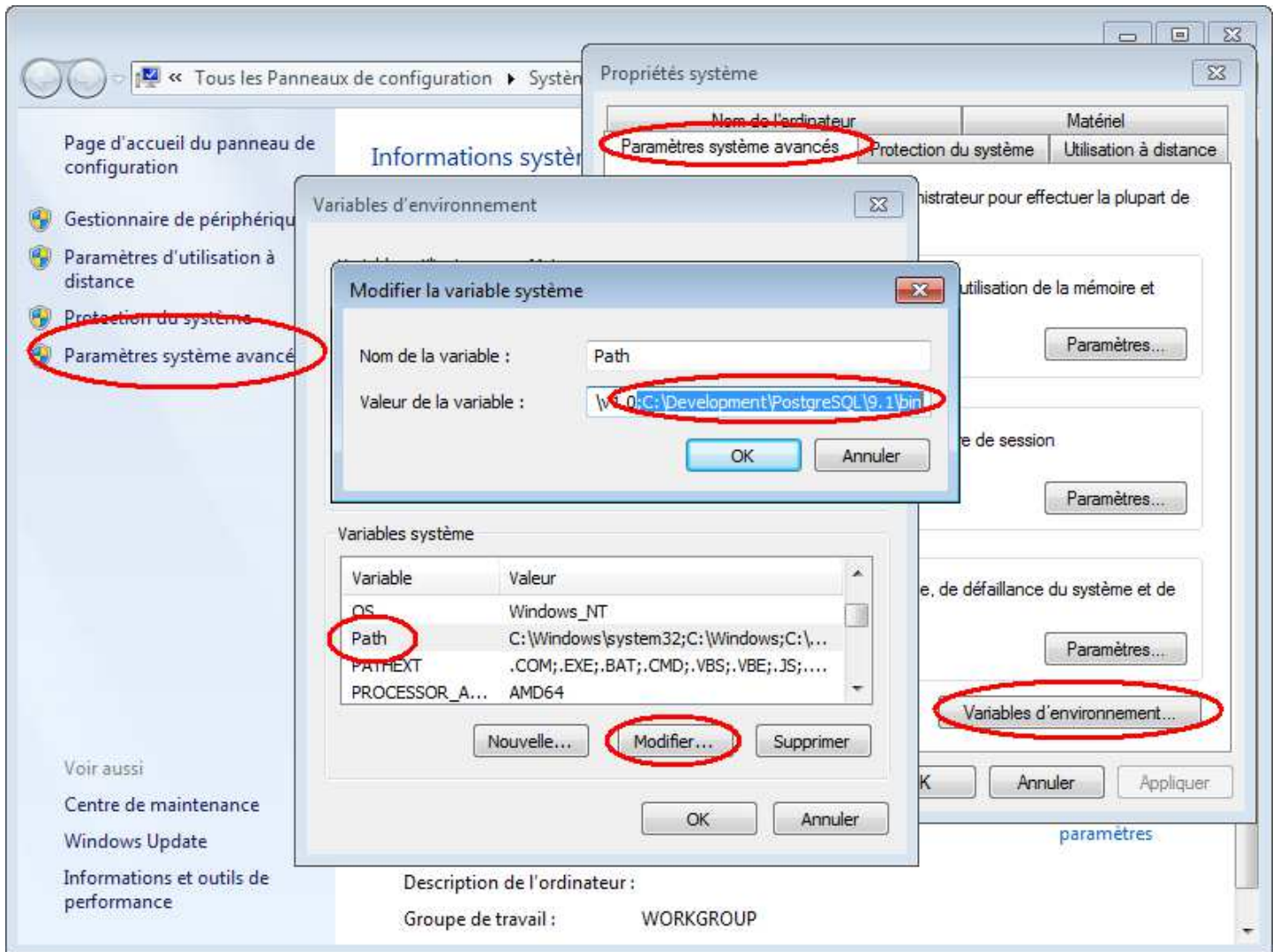
Ce paragraphe permet d'indiquer au système où se trouvent les commandes PostgreSQL que devra utiliser l'administrateur. Cette étape n'est pas obligatoire mais simplifiera grandement les actions futures de l'administrateur en lui évitant de mettre le chemin complet de la commande psql chaque fois qu'il voudra l'utiliser.

Il est nécessaire pour cela de modifier la variable "PATH" en y rajoutant le chemin d'accès aux commandes PostgreSQL qui est, dans ce document, "C:\Development\PostgreSQL\9.3\bin".

Cette variable est accessible par la fenêtre "Propriétés système" qui se lance via le menu "Panneau de configuration\Tous les Panneaux de configuration\Systeme "



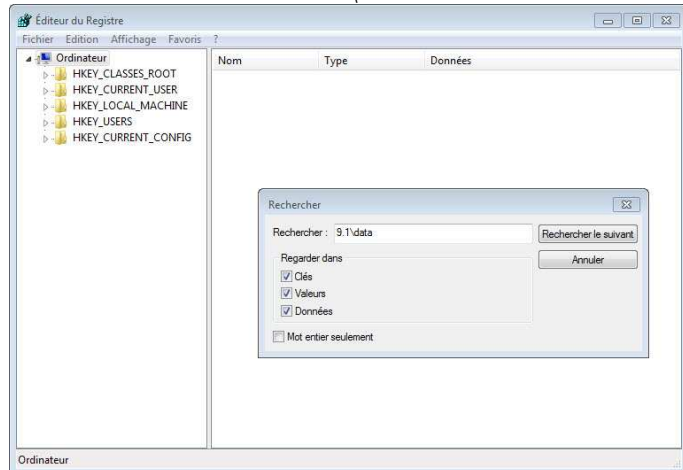
Il est nécessaire de modifier cette variable pour lui rajouter le chemin (en le séparant des autres chemins par un caractère de séparation "point-virgule") d'accès aux commandes de PostgreSQL susceptibles d'être utilisées par la suite.



3.2.6 Modification du dossier de stockage des bases de données

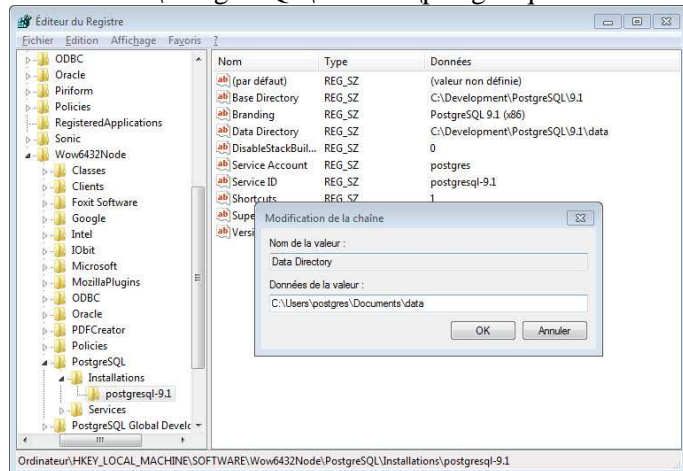
L'emplacement du dossier contenant les bases de données gérées par le serveur PostgreSQL est définitivement fixé lors de l'installation. Toutefois, il est possible de le modifier en appliquant la procédure suivante :

- ✓ Arrêter le serveur PostgreSQL s'il est démarré
- ✓ Copier ou déplacer le dossier "data" (situé dans " C:\Development\PostgreSQL\9.3\data") dans l'emplacement désiré (qui doit être bien entendu accessible à l'utilisateur postgres). Pour cet exemple on le placera dans les documents de l'utilisateur postgres (C:\Users\postgres\Documents\data) puisque, si on réfléchit bien, il s'agit de documents utilisés par cet utilisateur...
- ✓ Ouvrir regedit
- ✓ Rechercher la chaîne "9.3\data"

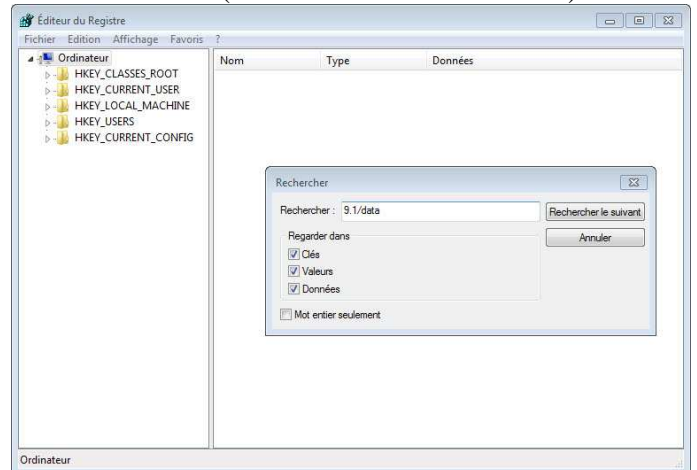


- ✓ Remplacer toutes les occurrences de "C:\Development\PostgreSQL\9.3\data" par "C:\Users\postgres\Documents\data". Théoriquement l'ensemble de ces occurrences se trouve dans :

- ☞ HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\PostgreSQL\Installations\postgresql-9.3
- ☞ HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\PostgreSQL\Services\postgresql-9.3

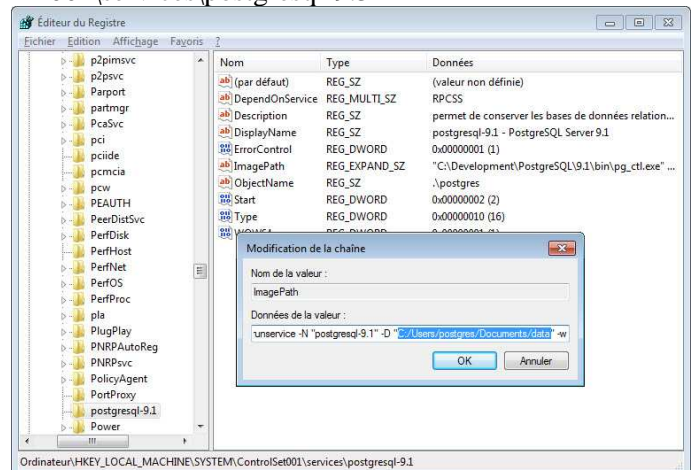


- ✓ Revenir au début de la base de registre et chercher la chaîne "9.3\data" (attention au sens du slash "/")



- ✓ Remplacer toutes les occurrences de "C:/Development/PostgreSQL/9.3/data" par "C:/Users/postgres/Documents/data" (attention au sens du slash "/"). Théoriquement l'ensemble de ces occurrences se trouve dans :

- ☞ HKEY_LOCAL_MACHINE\SYSTEM\ControlSet001\services\postgresql-9.3
- ☞ HKEY_LOCAL_MACHINE\SYSTEM\ControlSet002\services\postgresql-9.3

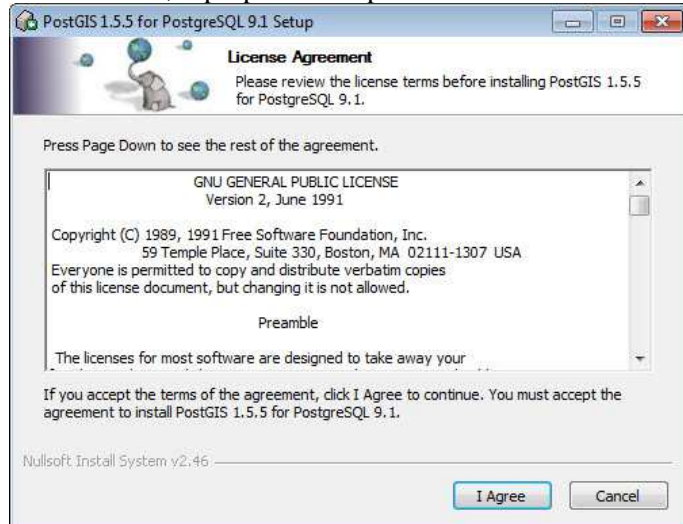


- ✓ Fermer Regedit
- ✓ Redémarrer le serveur PostgreSQL

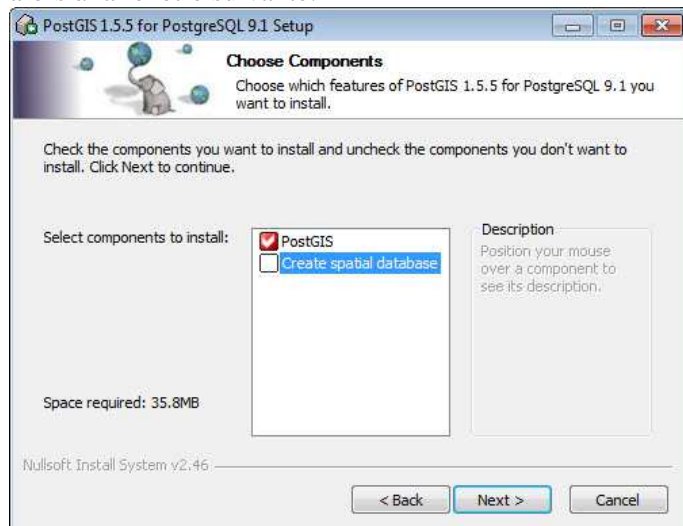
3.2.7 Installation de l'outil géographique PostGIS

Il s'agit d'un module additionnel à PostgreSQL permettant de stocker des objets géographiques dans une base de données. Le programme se trouve dans le dossier "PostgreSQL\AddOn Postgres" et se nomme "postgis-bundle-pg93x32-setup-2.1.3-1.exe". Il possède la signature md5 "efa872a3c23bb93c54b8f3dbb49b0a08".

A l'exécution, il propose cette première fenêtre...

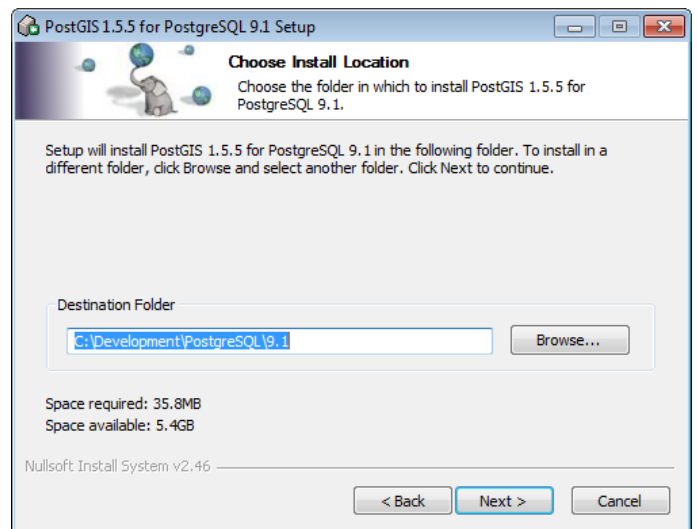


... qui n'a rien de compliqué. Le bouton "I Agree" amène alors à la fenêtre suivante.



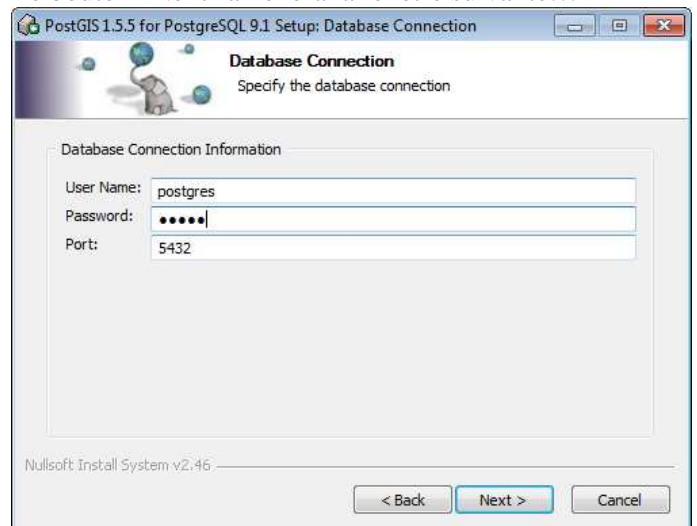
L'option "Create spatial database", non nécessaire, ordonnera en plus la création d'une base de données de démonstration.

Le bouton "Next" amènera la fenêtre suivante...



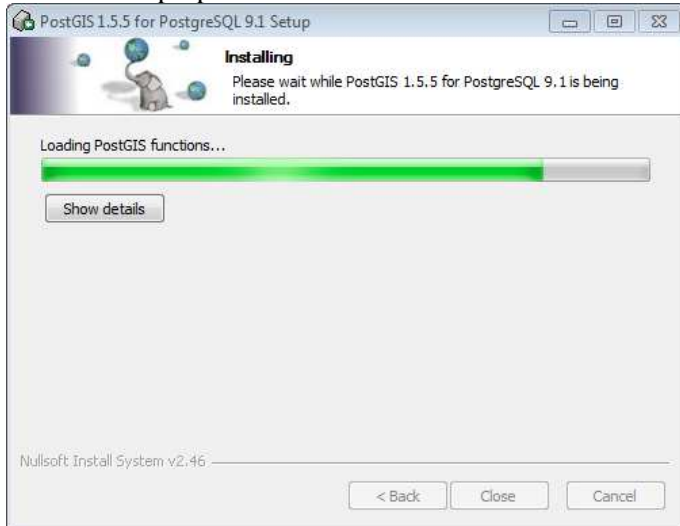
... permettant de choisir le dossier d'installation (par défaut le même que celui de PostgreSQL).

Le bouton "Next" amènera la fenêtre suivante...

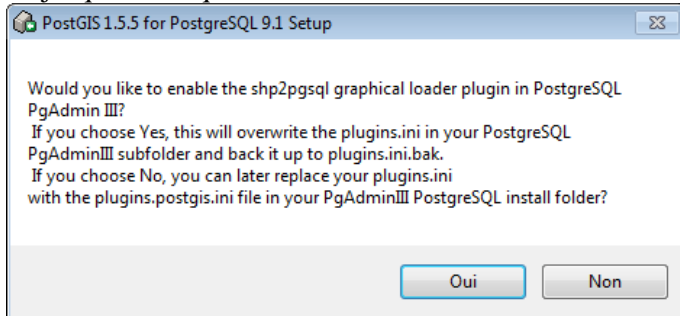


... dans laquelle Postgis demande le mot de passe du serveur PostgreSQL. Il convient de rentrer ici celui qui avait été saisi lors de l'installation de PostgreSQL (ou bien le nouveau si celui-ci a été modifié).

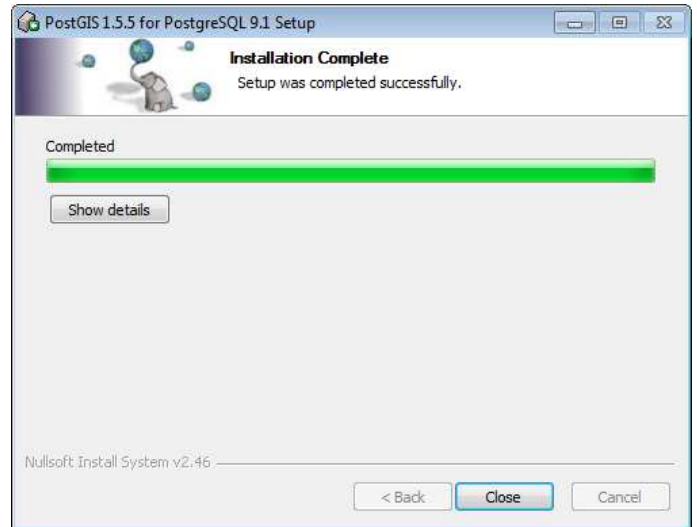
L'appui du bouton "Next provoquera le lancement de l'installation proprement dit...



... jusqu'à cette question...



...demandant si on accepte d'écraser le fichier "plugins.ini" de PostgreSQL par celui de Postgis. Il convient ici de répondre "oui" afin de terminer l'installation.



[3.3 Installation du serveur PostgreSQL sur système d'exploitation Linux](#)

Fera l'objet d'une rédaction ultérieure si nécessaire.

4 INSTALLATION DE LA LIBRAIRIE QT

4.1 Pré-requis.

Ce chapitre ne concerne que les machines ne possédant pas la librairie "Qt". Il s'agit d'une librairie graphique gratuite permettant de programmer des IHM.

Les différents fichiers d'installation se trouvent dans le dossier "logicielsTiers\Qt" situé à la racine du cdrom d'installation. La version de cette librairie est la 4.8.6 qui date de novembre 2014.

4.2 Installation de la librairie Qt sur système d'exploitation Windows

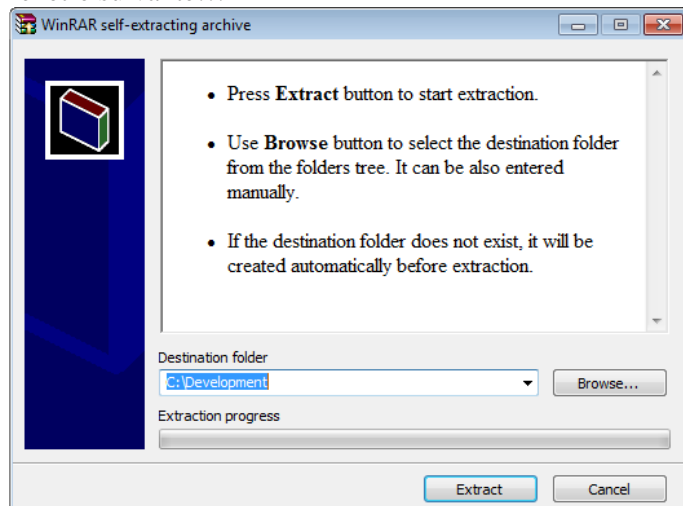
L'installation de la librairie Qt est de la responsabilité de l'administrateur du poste sur lequel l'interpréteur sera installé. Les différents programmes à installer avec leurs signatures md5 associées seront :

✓ MinGW_gcc4814.exe (md5: 2f74d2eb0b9cd3bb2c55833dd396f3a7): Compilateur gratuit C/C++ (stocké ici sous la forme d'une archive WinRAR autoextractible)

✓ qt-win-opensource-4.8.6-mingw.exe (md5: b0f70eb2607b89653c7aafe524352ce3): Librairie Qt proprement dite

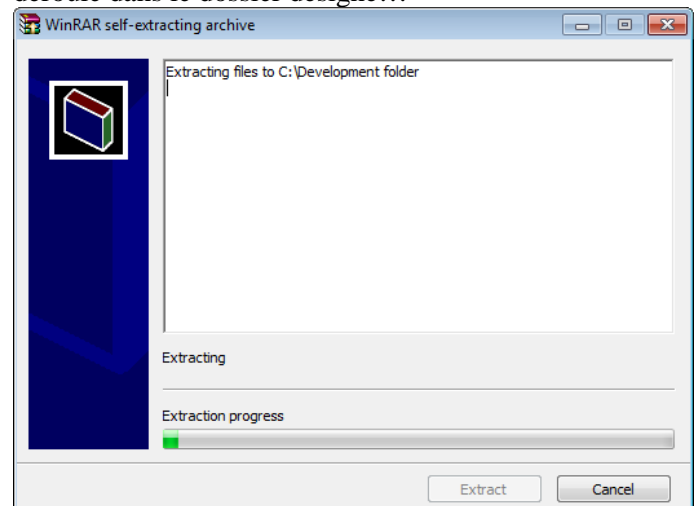
4.2.1 Installation du compilateur

Exécuter tout d'abord le programme "MinGW_gcc4814.exe". Le programme propose la fenêtre suivante...



... demandant où sera installé le compilateur. Comme convenu en introduction, nous indiquerons "C:\Development". Il n'est pas nécessaire de préciser un sous-dossier, celui-ci (MinGW_gcc4814) sera automatiquement créé.

Dès l'appui sur le bouton "Extract", l'extraction se déroule dans le dossier désigné...



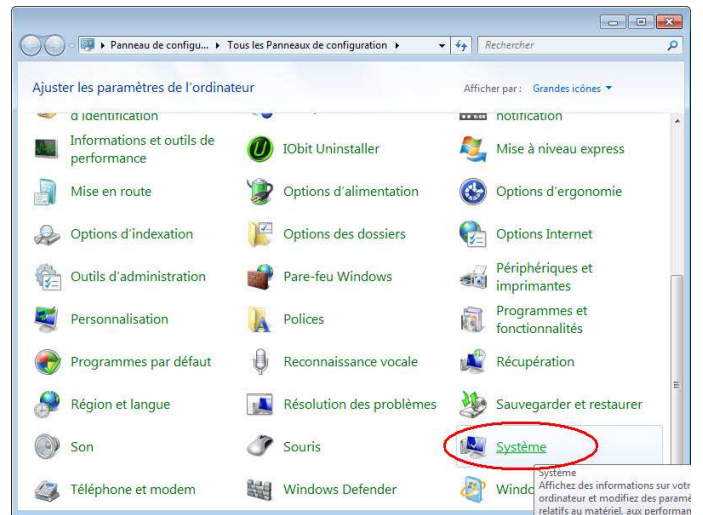
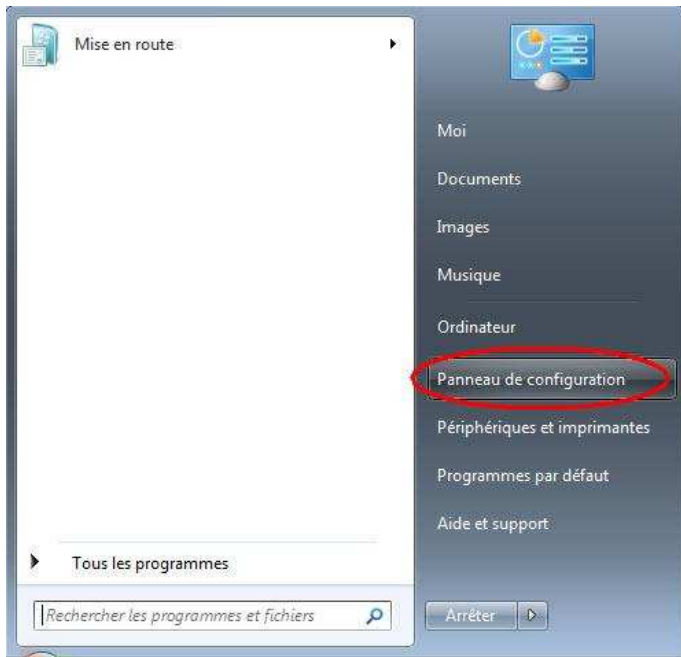
... et une fois terminée, la fenêtre se referme.

4.2.2 Paramétrage de l'environnement

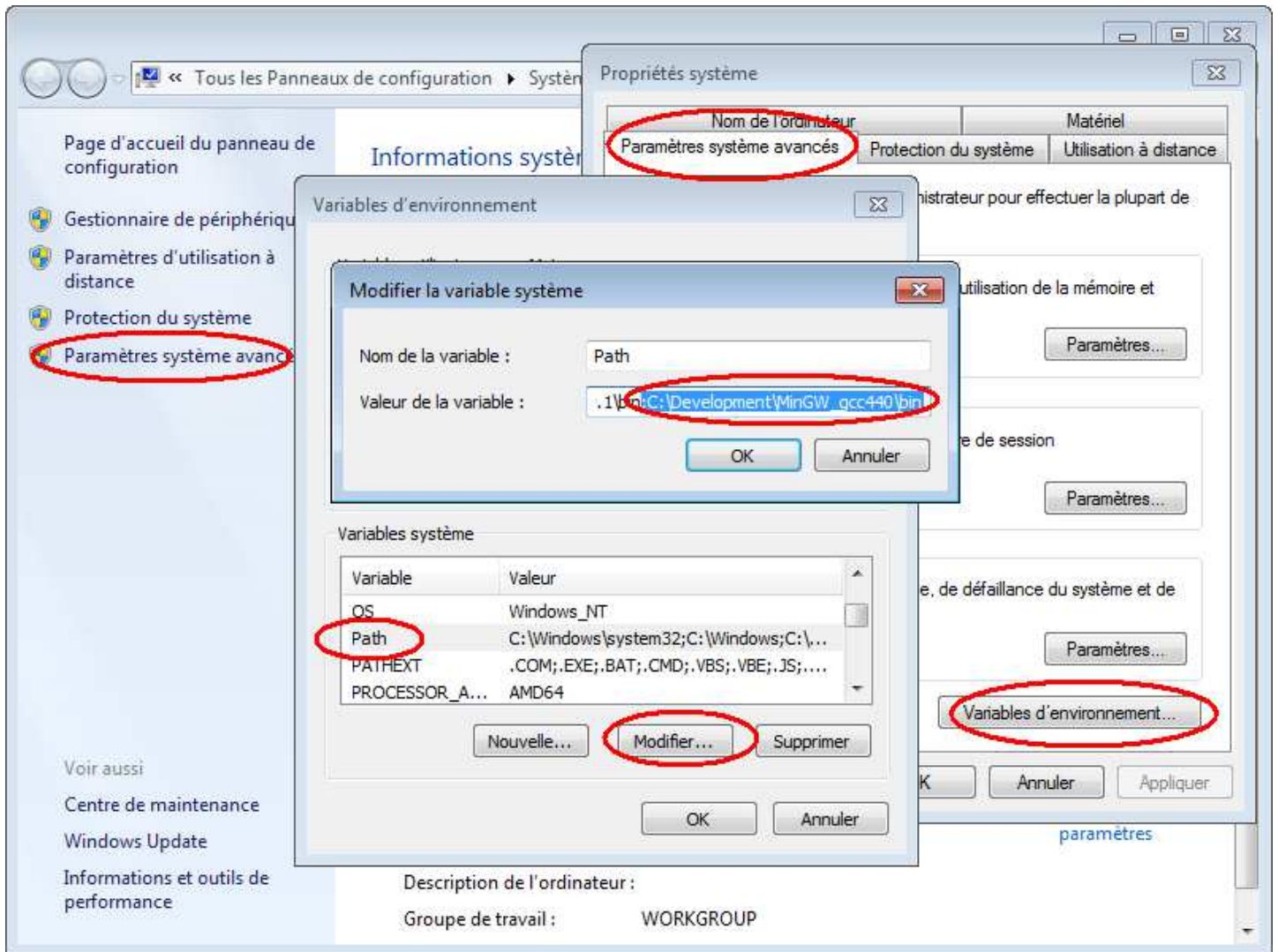
Ce paragraphe permet d'indiquer au système où se trouvent les outils MinGW que devront utiliser les autres logiciels. Cette étape est obligatoire car les autres logiciels utilisent ces outils mais n'ont aucun autre moyen de connaître cet emplacement.

Il est nécessaire pour cela de modifier la variable "PATH" en y rajoutant le chemin d'accès aux programmes MinGW qui est, dans ce document, "C:\Development\MinGW_gcc4814\bin".

Cette variable est accessible par la fenêtre "Propriétés système" qui se lance via le menu "Panneau de configuration\Tous les Panneaux de configuration\Systeme "



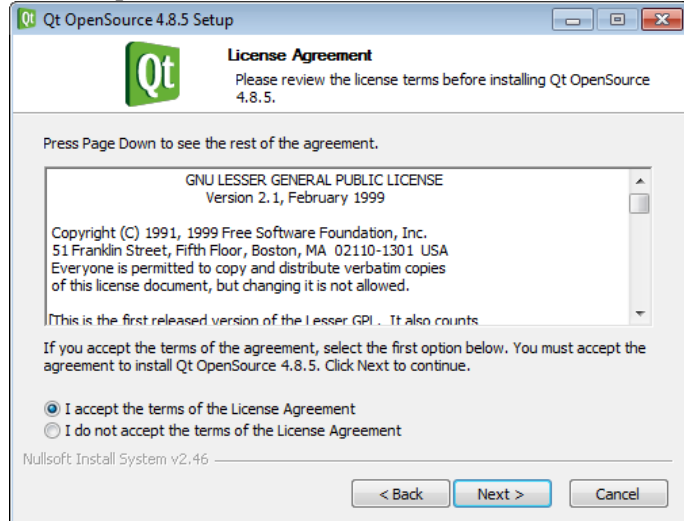
Il est nécessaire de modifier cette variable pour lui rajouter le chemin (en le séparant des autres chemins par un caractère de séparation "point-virgule") d'accès aux programmes de MinGW qui seront utilisés par la suite.



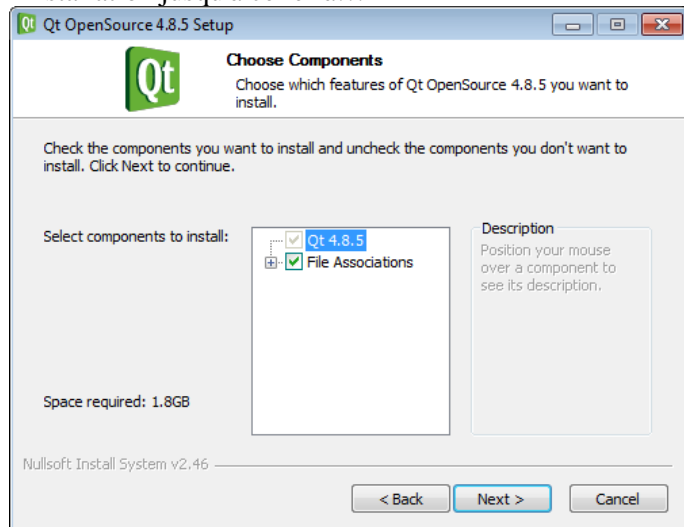
4.2.3 Installation de la librairie Qt

Exécuter maintenant le programme "qt-win-opensource-4.8.6-mingw.exe".

Après avoir désarchivé son contenu et passé sur les deux premières fenêtres ne demandant rien d'autre que d'appuyer à chaque fois sur le bouton "Next", on arrive sur cette première fenêtre...

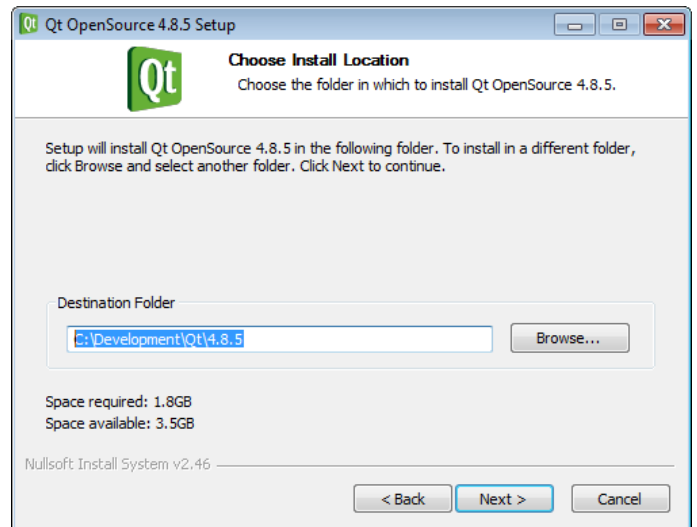


...demandant d'accepter la licence pour pouvoir continuer l'installation jusqu'à celle-là...



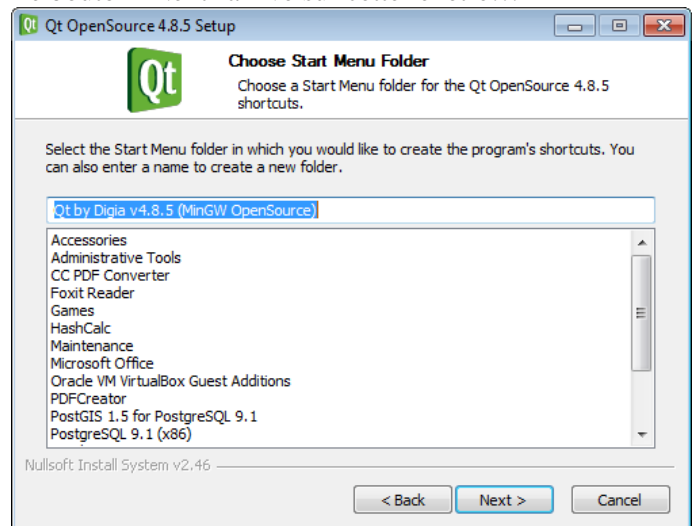
... demandant de choisir les composants (qui ne sont pas nombreux) avant d'arriver sur celle-là...

...demandant où sera installé la librairie elle-même.



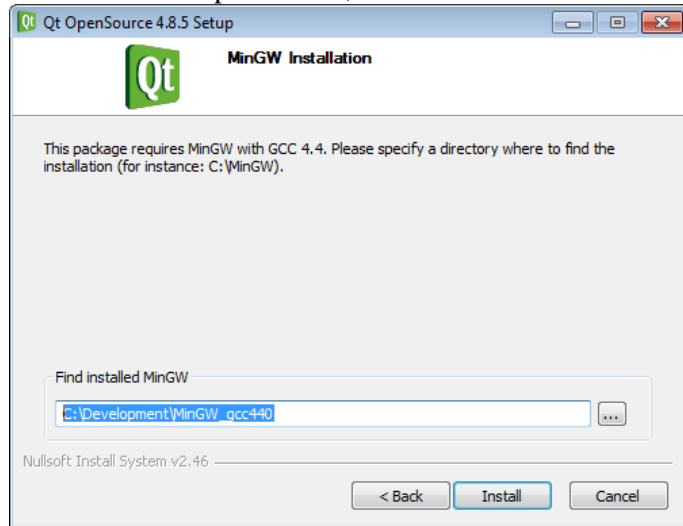
Comme convenu, il est prévu que cette librairie soit installée elle-aussi dans le dossier "C:\Development". Il est toutefois nécessaire ici de préciser le chemin complet. On indique donc la totalité du chemin, à savoir "C:\Development\Qt\4.8.6".

Le bouton "Next" arrive sur cette fenêtre...



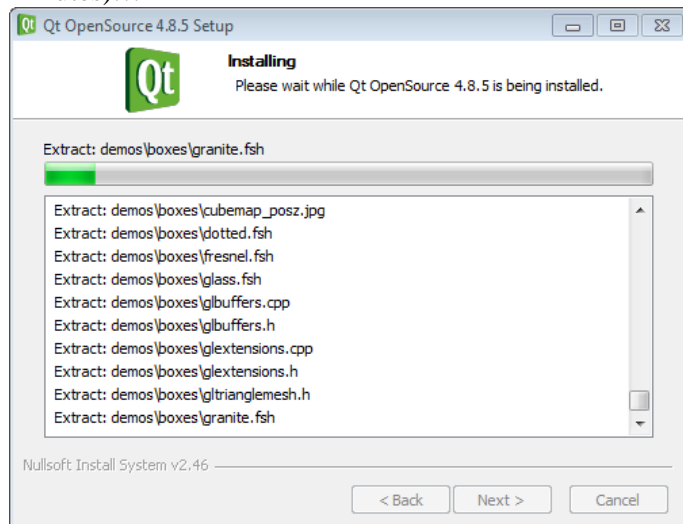
...demandant de définir le menu Windows permettant d'accéder à la librairie (étape obligatoire mais inutile pour nous).

Une fois cette étape franchie, la fenêtre suivante...

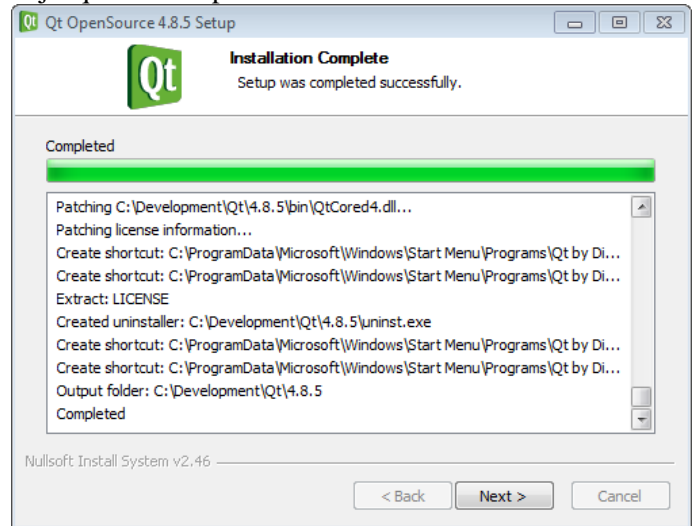


...demande où se trouve le compilateur (précédemment installé). Il est nécessaire alors de préciser le chemin choisi lors de cette installation, à savoir ici "C:\Development\MinGW_gcc4814".

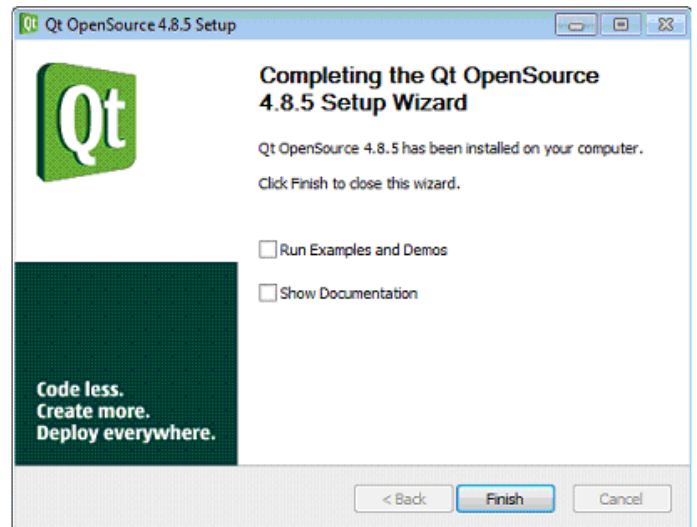
Et enfin le bouton "Installer" installe la librairie proprement dite (étape de l'ordre d'une dizaine de minutes)...



...jusqu'à sa complétion...



... où le bouton "Next" arrive sur cette dernière fenêtre...



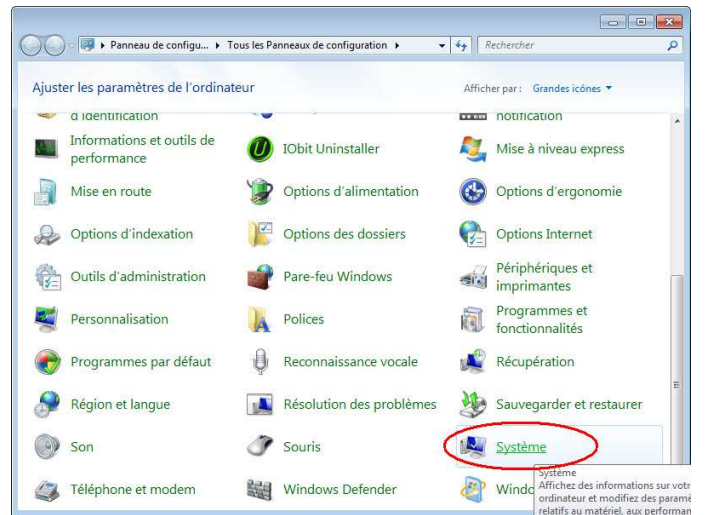
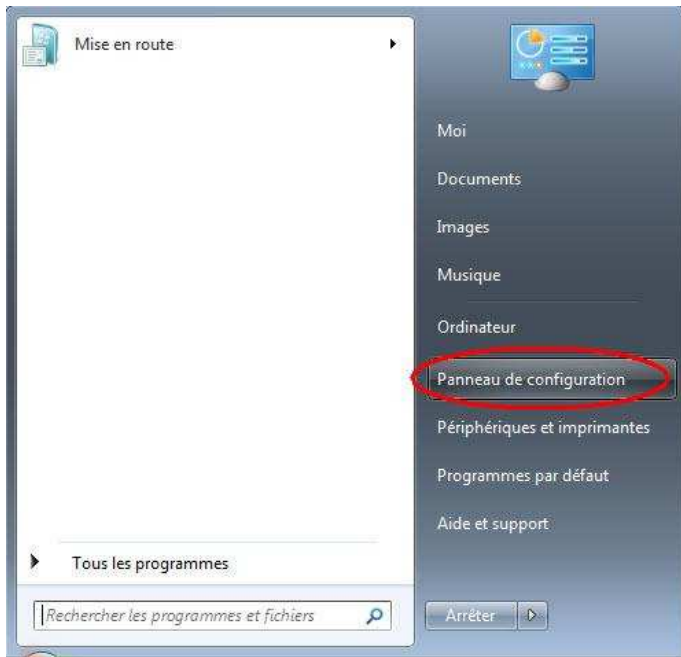
... indiquant que l'installation est maintenant terminée. Celui qui le désire pourra cocher les options demandant le lancement d'exemples et de démonstrations ou bien les lancer plus tard depuis le menu Windows.

4.2.4 Paramétrage de l'environnement

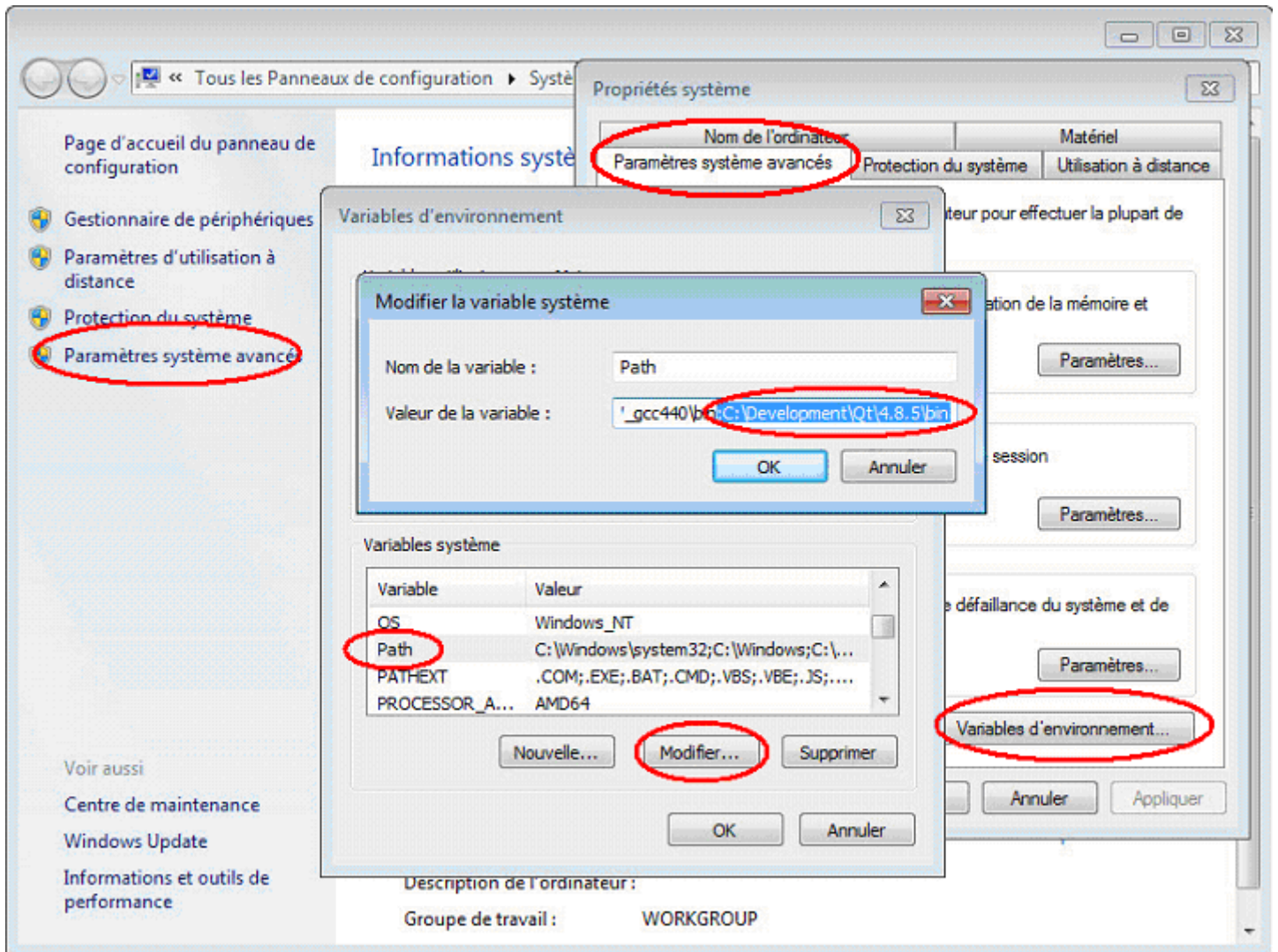
Ce paragraphe permet d'indiquer au système où se trouvent les outils Qt que devront utiliser les autres logiciels. Cette étape est obligatoire car les autres logiciels utilisent ces outils mais n'ont aucun autre moyen de connaître cet emplacement.

Il est nécessaire pour cela de modifier la variable "PATH" en y rajoutant le chemin d'accès aux programmes Qt qui est, dans ce document, "C:\Development\Qt\4.8.6\bin".

Cette variable est accessible par la fenêtre "Propriétés système" qui se lance via le menu "Panneau de configuration\Tous les Panneaux de configuration\Systeme "



Il est nécessaire de modifier cette variable pour lui rajouter le chemin (en le séparant des autres chemins par un caractère de séparation "point-virgule") d'accès aux programmes de Qt qui seront utilisés par la suite.



[4.3 Installation de la librairie Qt sur système d'exploitation Linux](#)

Fera l'objet d'une rédaction ultérieure si nécessaire.

5 INSTALLATION DE L'INTERPRETEUR PYTHON

5.1 Pré-requis.

Ce chapitre ne concerne que les machines ne possédant pas d'interpréteur Python.

Les différents fichiers d'installation se trouvent dans le dossier "logicielsTiers\Python\3.4" situé à la racine du cdrom d'installation.

La version de l'interpréteur Python est la 3.4 qui date de novembre 2014. Tous les outils associés sont basés sur cette version qui ne doit pas être modifiée.

Pour ceux qui voudraient s'intéresser au langage lui-même, deux manuels d'apprentissages au format PDF et nommés "python_notes.pdf" et "diveintopython.pdf" se trouvent dans le dossier "logicielsTiers\Python\doc" du cdrom d'installation.

5.2 Installation de l'interpréteur Python sur système d'exploitation Windows

L'installation de l'interpréteur Python est de la responsabilité de l'administrateur du poste sur lequel l'interpréteur sera installé. Les différents programmes à installer avec leurs signatures md5 associées seront :

- ✓ python-3.4.2.msi (md5: 0aa1a556892d8dc0b60c19bf3102fb3f): Interpréteur de base Python
- ✓ psycopg2-2.5.4.win32-py3.4-pg9.3.5-release.exe (md5: 81370d1691e66082764ef5dca6a39063): Module Python permettant d'accéder à une base de données PostgreSQL
- ✓ numpy-1.9.1-win32-superpack-python3.4.exe (md5: 1402e7689bebbd7b69630bdcdc58a492): Module Python de calculs mathématiques complexes
- ✓ sip-4.16.4.exe (md5: 47985819cbc3a09610798bbe37d4e0f7): Module Python permettant d'appeler des fonctions C (stocké ici sous la forme d'une archive WinRar autoextractible)
- ✓ PyQt-x11-gpl-4.11.3.exe (md5: f8f88d66ba21ec2a74990b339da5403a): Module Python qui intègre les fonctions de la librairie Qt (stocké ici sous la forme d'une archive WinRar autoextractible). Ce module sera compilé par le compilateur MinGW (installé précédemment)

5.2.1 Déroulement du programme d'installation de l'interpréteur de base

Exécuter tout d'abord le programme "python-3.4.2.msi". Le programme propose la fenêtre suivante...



... qui détermine la configuration du menu Démarrer (création dans "All Users" ou dans le menu de l'utilisateur). Ensuite la fenêtre suivante...



... permet de définir l'emplacement de l'interpréteur Python. Le choix par défaut est "C:\Python34" mais conformément à la convention concernant le dossier d'installation, nous l'installerons dans "C:\Development\Python34".

Et enfin cette dernière fenêtre...



... permet de choisir les divers outils Python. Il est nécessaire de demander l'installation totale.

Une fois ce dernier choix validé, l'installation de Python se déroule pour terminer par cette fenêtre finale...



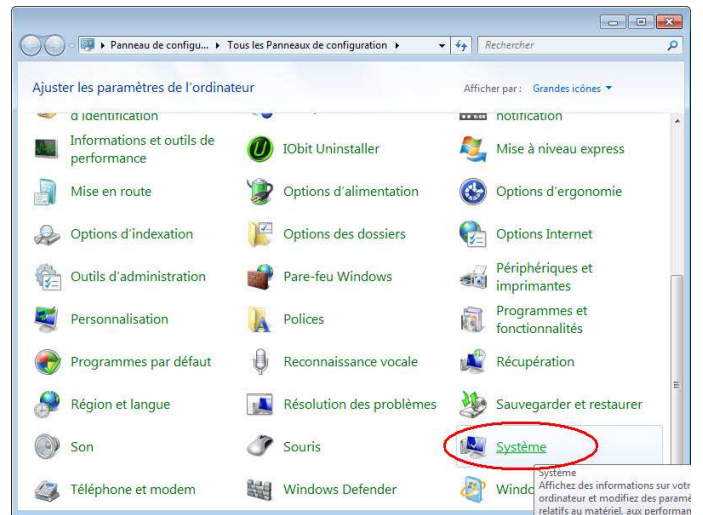
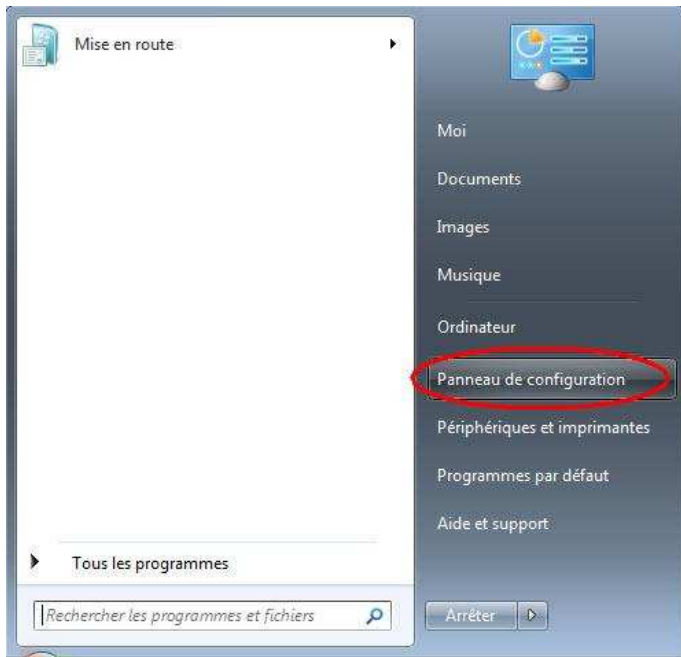
... indiquant que l'installation est terminée.

5.2.2 Paramétrage de l'environnement

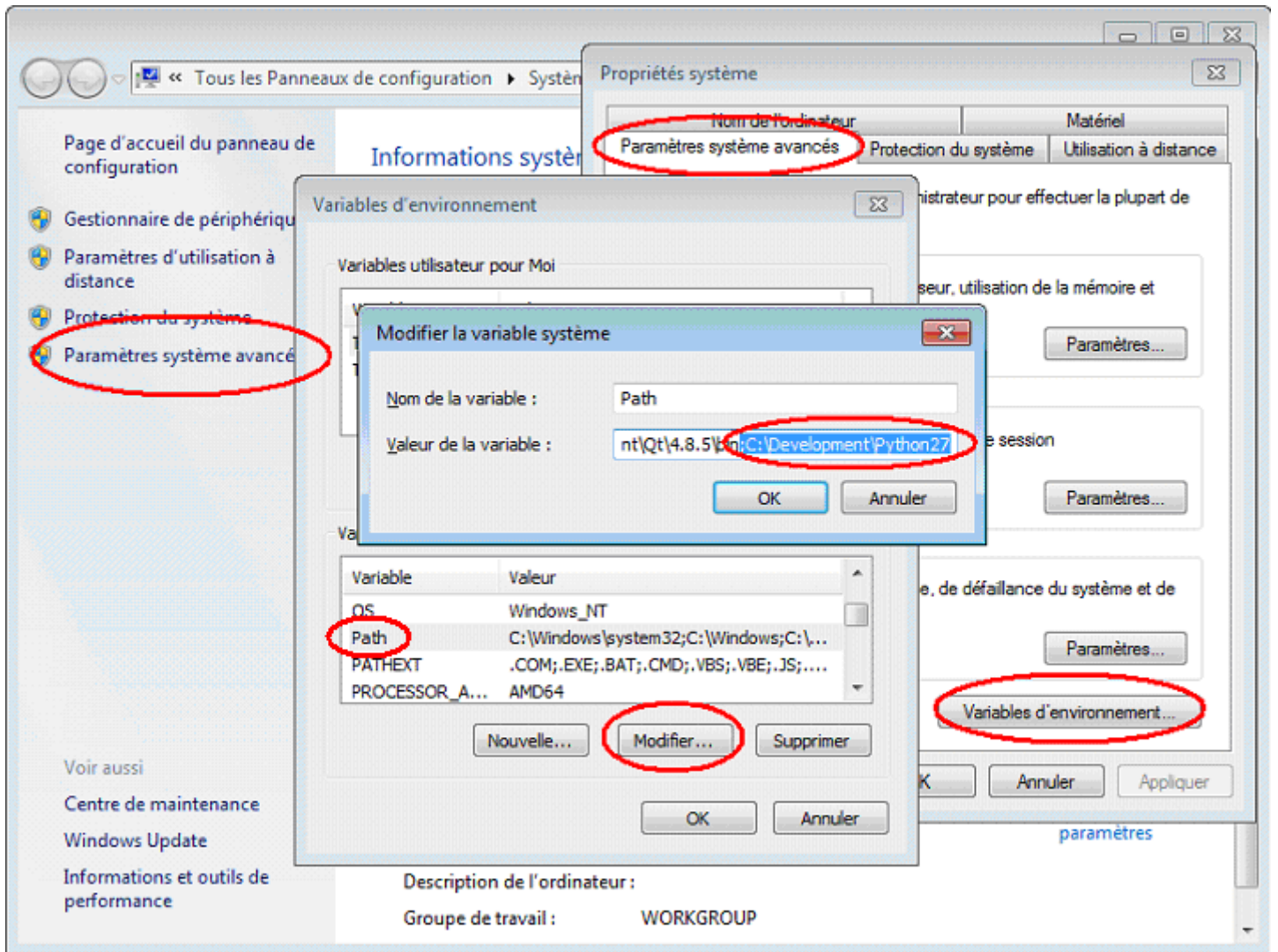
Ce paragraphe permet d'indiquer au système où se trouve le programme Python que devra utiliser les autres logiciels. Cette étape est obligatoire car les autres logiciels utilisent ces outils mais n'ont aucun autre moyen de connaître cet emplacement.

Il est nécessaire pour cela de modifier la variable "PATH" en y rajoutant le chemin d'accès au programme Python qui est, dans ce document, "C:\Development\Python34".

Cette variable est accessible par la fenêtre "Propriétés système" qui se lance via le menu "Panneau de configuration\Tous les Panneaux de configuration\Systeme "



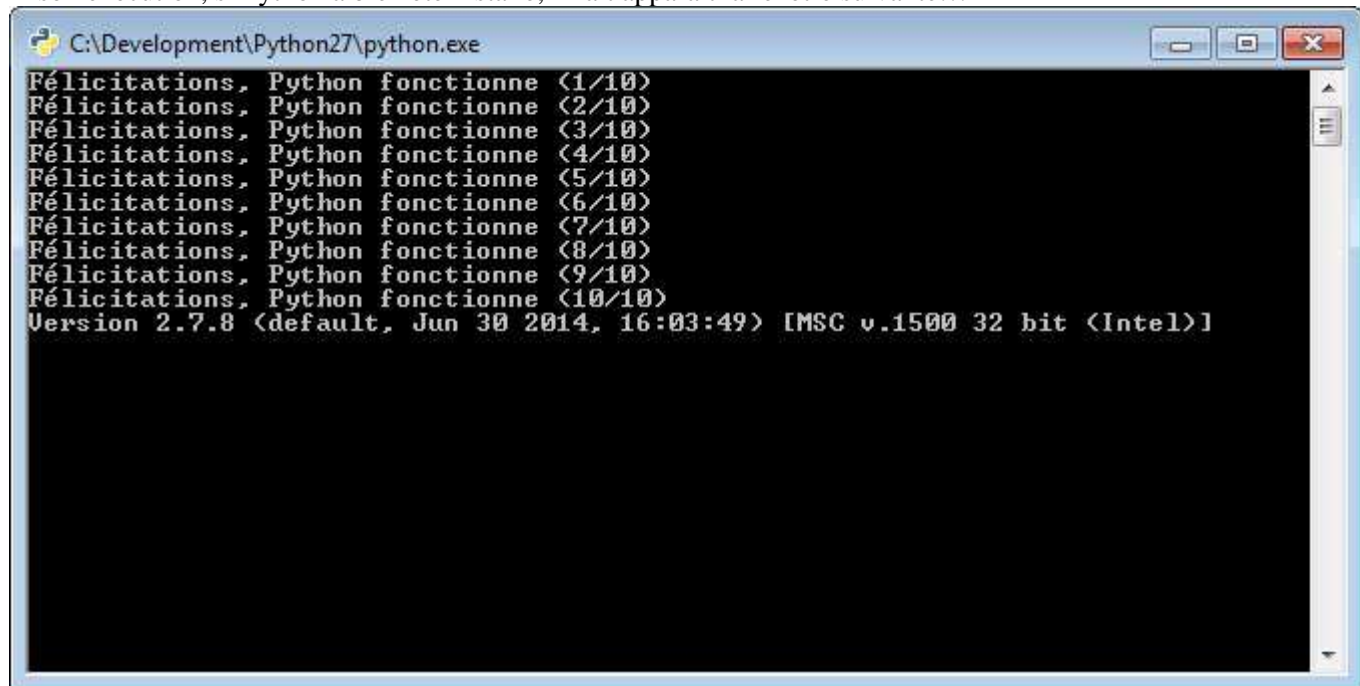
Il est nécessaire de modifier cette variable pour lui rajouter le chemin (en le séparant des autres chemins par un caractère de séparation "point-virgule") d'accès au programme Python qui sera utilisé par la suite.



5.2.3 Vérification de la bonne installation de Python

Un programme nommé "VerifPython.py" permet de vérifier que Python a correctement été installé. Il se trouve dans le dossier "logicielsTiers\Python\testInstallation".

A son exécution, si Python a bien été installé, il fait apparaître la fenêtre suivante...



```
C:\Development\Python27\python.exe
Félicitations, Python fonctionne (1/10)
Félicitations, Python fonctionne (2/10)
Félicitations, Python fonctionne (3/10)
Félicitations, Python fonctionne (4/10)
Félicitations, Python fonctionne (5/10)
Félicitations, Python fonctionne (6/10)
Félicitations, Python fonctionne (7/10)
Félicitations, Python fonctionne (8/10)
Félicitations, Python fonctionne (9/10)
Félicitations, Python fonctionne (10/10)
Version 2.7.8 (default, Jun 30 2014, 16:03:49) [MSC v.1500 32 bit (Intel)]
```

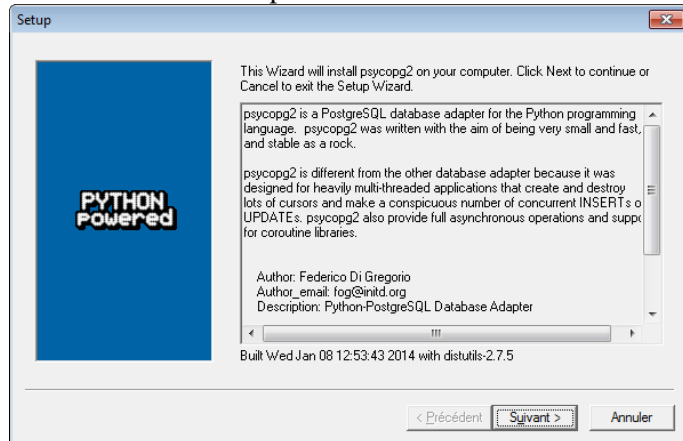
... qui reste ouverte 10 secondes puis qui disparaît.

5.2.4 Le module Psycpg2

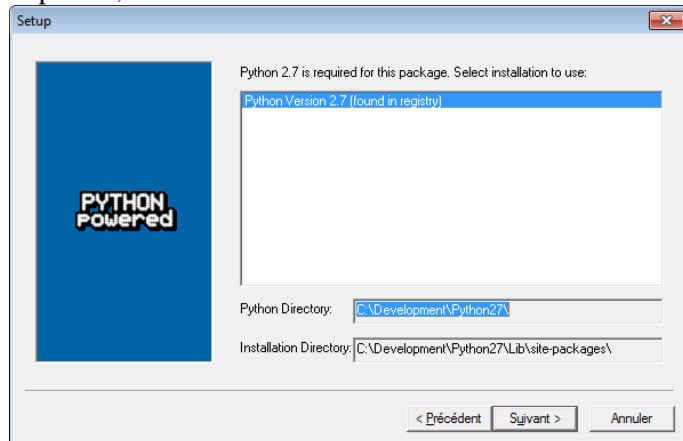
Ce module permet à Python d'interfacer un serveur PostgreSQL afin d'accéder à ses différentes bases de données. Il se trouve dans le dossier "logicielsTiers\Python\AddOn_Python\Postgres".

Il se présente sous la forme d'un programme d'installation à exécuter. Pour ce faire, exécuter simplement le programme "psycpg2-2.5.4.win32-py3.4-pg9.3.5-release.exe".

Une fois la fenêtre de présentation...

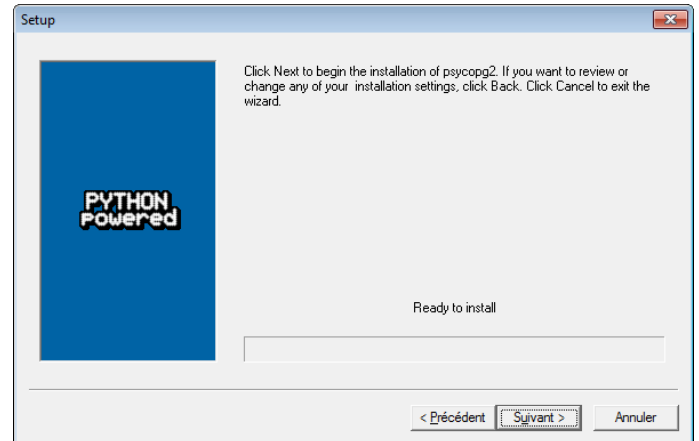


...passée, la fenêtre suivante...



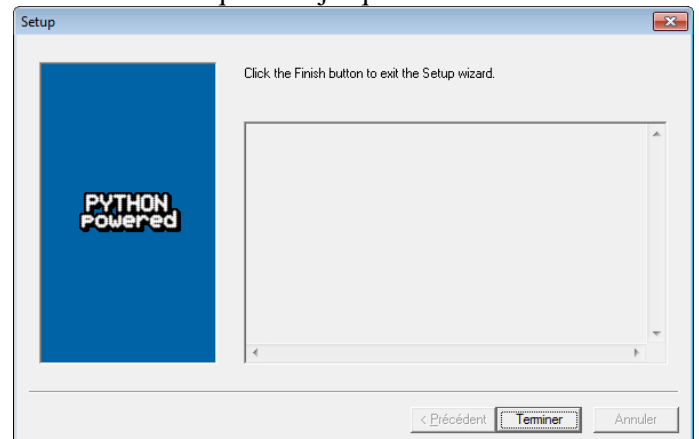
...demande de vérifier la version de Python installée et son emplacement (ici "C:\Development\Python34").

Dès cette vérification faite, le bouton "Suivant" amène la fenêtre suivante...



...indiquant que le module est prêt à être installé.

L'appui du bouton "Suivant" active l'installation qui se déroule automatiquement jusqu'à cette fenêtre...



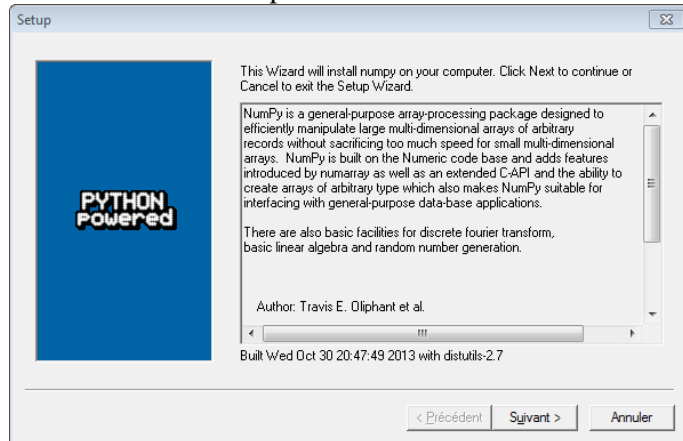
...indiquant que l'installation du module est terminée.

5.2.5 Le module Numpy

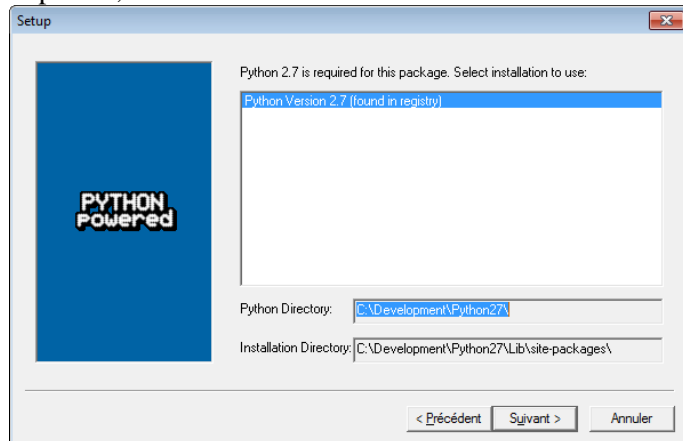
Ce module intègre dans Python des outils mathématiques (tels des matrices, etc). Il se trouve dans le dossier "logicielsTiers\Python\AddOn_Python\Qt".

Il se présente sous la forme d'un programme d'installation à exécuter. Pour ce faire, exécuter simplement le programme "numpy-1.9.1-win32-superpack-python3.4.exe".

Une fois la fenêtre de présentation...

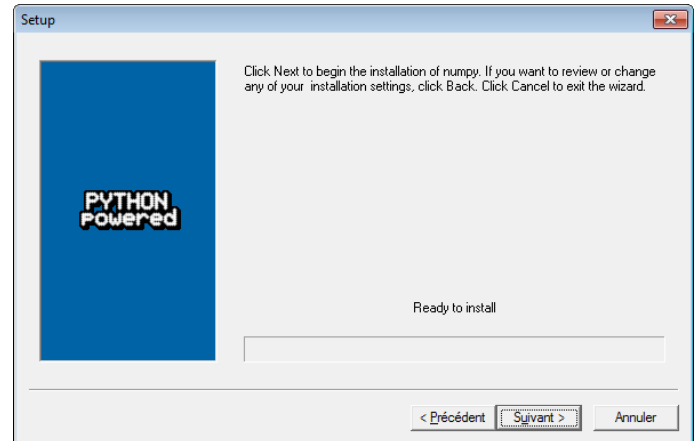


...passée, la fenêtre suivante...



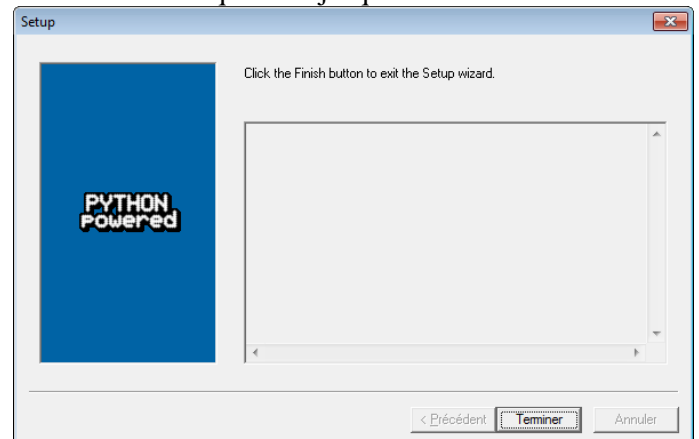
...demande de vérifier la version de Python installée et son emplacement (ici "C:\Development\Python34").

Dès cette vérification faite, le bouton "Suivant" amène la fenêtre suivante...



...indiquant que le module est prêt à être installé.

L'appui du bouton "Suivant" active l'installation qui se déroule automatiquement jusqu'à cette fenêtre...



...indiquant que l'installation du module est terminée.

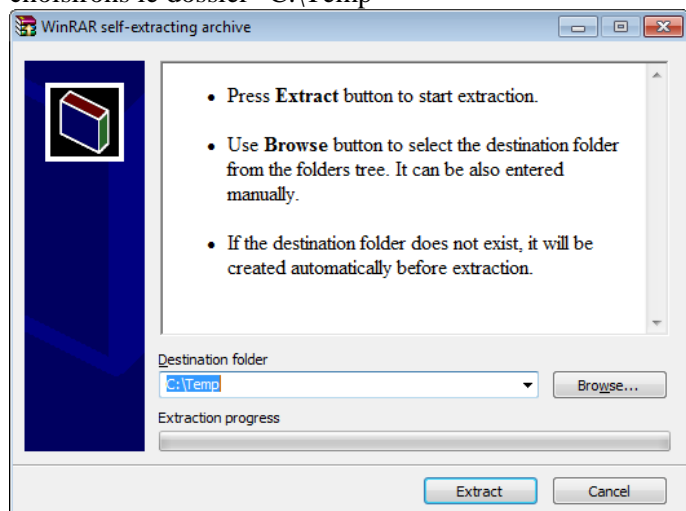
5.2.6 Le module sip

Ce module permet à Python d'interfacer des fonctions C/C++. Il a été stocké sous forme d'archive WinRAR autoextractible et se trouve dans le dossier "logicielsTiers\Python\AddOn_Python\Qt".

L'installation se fait en 2 phases :

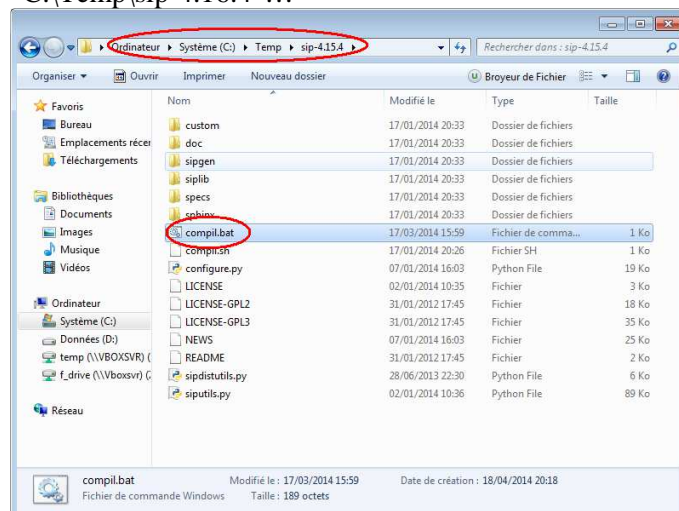
- ✓ Descente de l'archive dans un dossier de travail temporaire (pourra être supprimé ensuite)
- ✓ Extraction et installation des bibliothèques Python

Commencer par exécuter le programme "sip-4.16.4.exe". Celui-ci propose le choix du dossier d'extraction. Le choix par défaut est suffisant et peut rester tel quel sauf si le programme est directement exécuté depuis le cdrom d'installation (non modifiable). Dans ce cas, il sera alors nécessaire de choisir un dossier de travail dont le chemin ne contient pas d'espace. Pour cet exemple nous choisirons le dossier "C:\Temp"



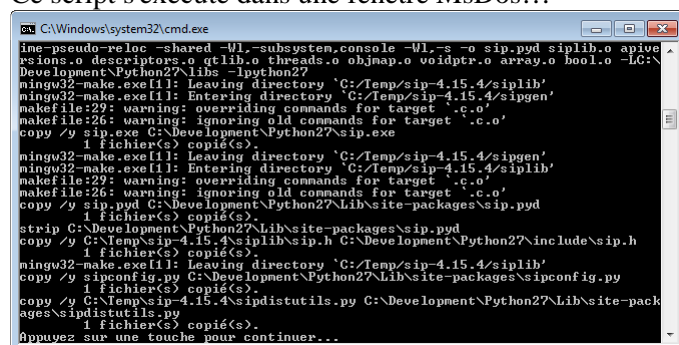
Le bouton "Extract" extrait l'archive (nommée "sip-4.16.4") dans le dossier demandé ("C:\Temp") ce qui termine la première partie.

La seconde partie consiste à se déplacer dans le dossier "C:\Temp\sip-4.16.4"...



...pour y exécuter le script d'installation "compil.bat".

Ce script s'exécute dans une fenêtre MsDos...



...laquelle attend en final qu'on y appuie sur une touche pour se fermer ce qui termine la seconde partie et l'installation du module sip.

Une fois cette seconde partie terminée, on peut alors supprimer le dossier "C:\Temp\sip-4.16.4".

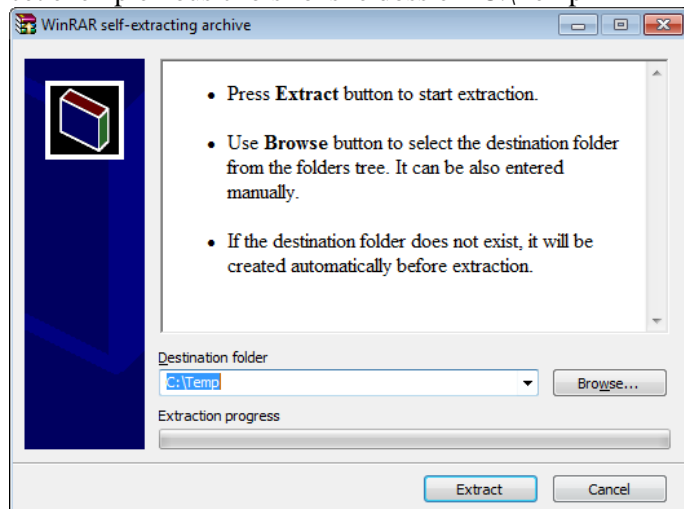
5.2.7 Le module PyQt

Ce module permet à Python d'intégrer les éléments de la librairie Qt. Il a été stocké sous forme d'archive WinRAR autoextractible et se trouve dans le dossier "logicielsTiers\Python\AddOn_Python\Qt".

L'installation se fait en 2 phases :

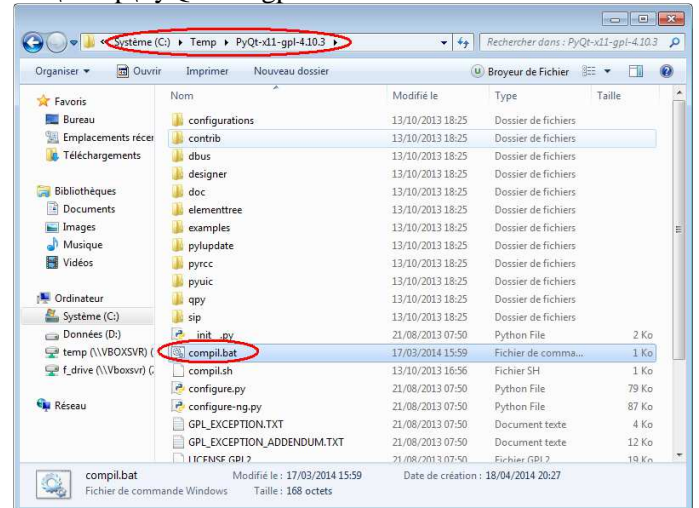
- ✓ Descente de l'archive dans un dossier de travail temporaire (pourra être supprimé ensuite)
- ✓ Extraction et installation des librairies Python

Commencer par exécuter le programme "PyQt-x11-gpl-4.11.3.exe". Celui-ci propose le choix du dossier d'extraction. Le choix par défaut est suffisant et peut rester tel quel sauf si le programme est directement exécuté depuis le cdrom d'installation (non modifiable). Dans ce cas, il sera alors nécessaire de choisir un dossier de travail dont le chemin ne contient pas d'espace. Pour cet exemple nous choisirons le dossier "C:\Temp"



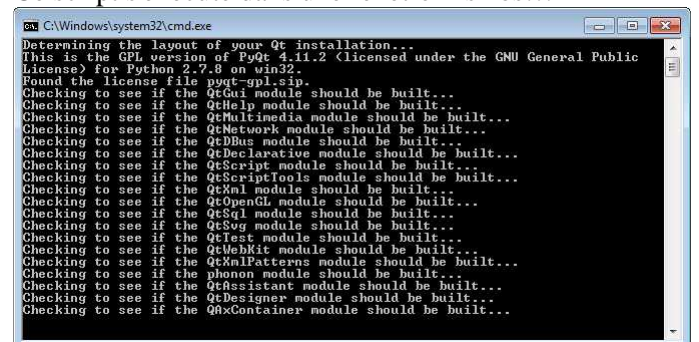
Le bouton "Extract" extrait l'archive (nommée "PyQt-x11-gpl-4.11.3") dans le dossier demandé ("C:\Temp") ce qui termine la première partie.

La seconde partie consiste à se déplacer dans le dossier "C:\Temp\PyQt-x11-gpl-4.11.3"...



...pour y exécuter le script d'installation "compil.bat".

Ce script s'exécute dans une fenêtre MsDos...



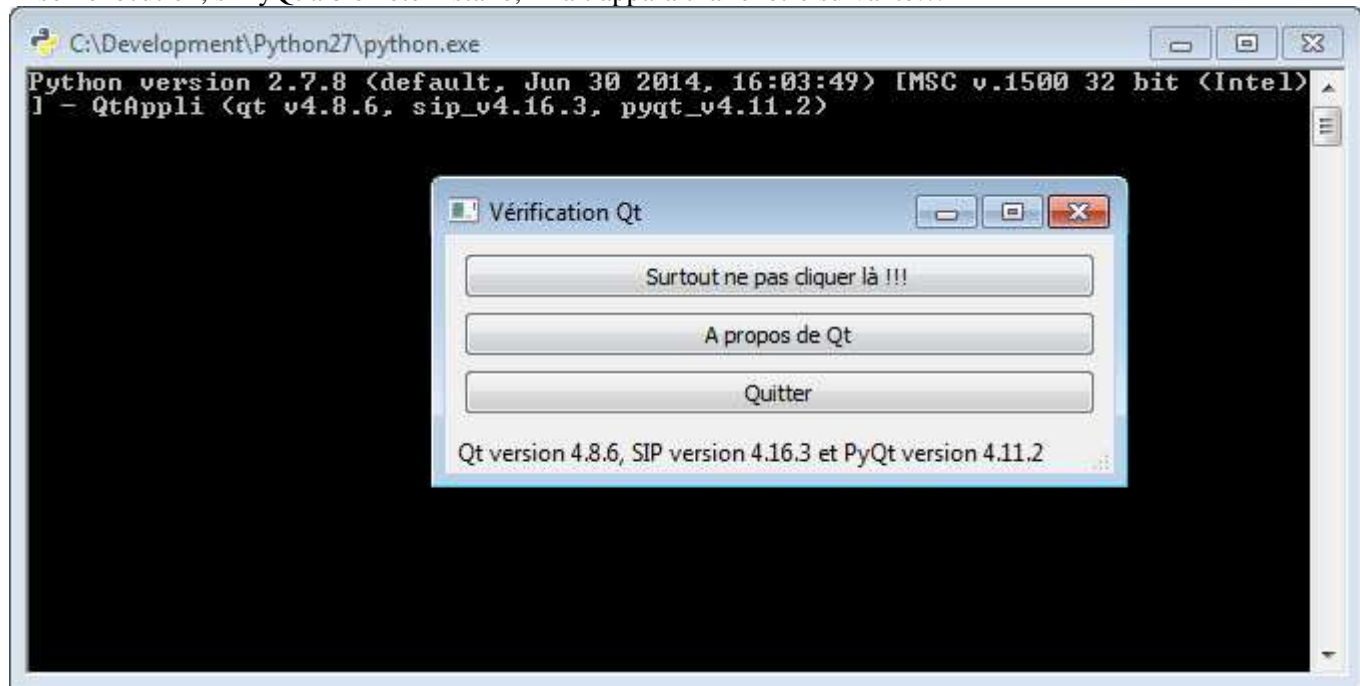
...laquelle attend en final qu'on y appuie sur une touche pour se fermer ce qui termine la seconde partie et l'installation du module PyQt.

Une fois cette seconde partie terminée, on peut alors supprimer le dossier "C:\Temp\PyQt-x11-gpl-4.11.3".

5.2.8 Vérification de la bonne installation de PyQt

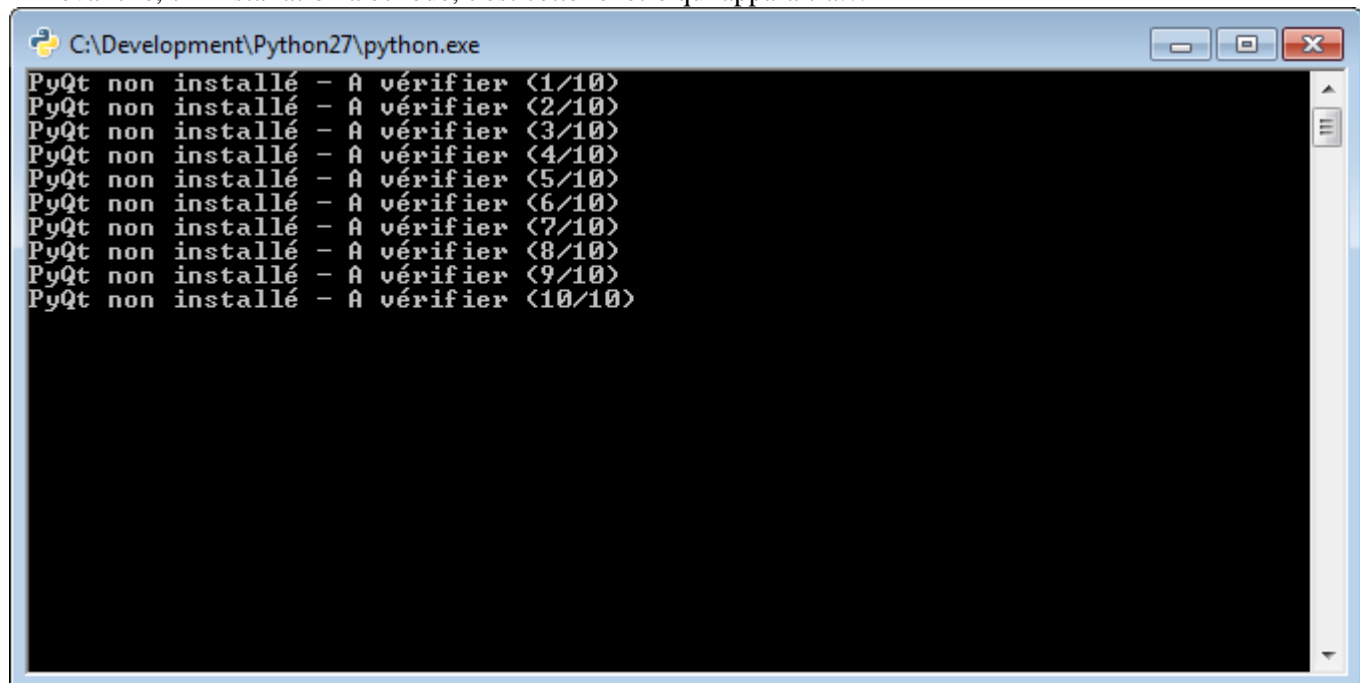
Un programme nommé "VerifQt.py" permet de vérifier que PyQt a correctement été installé. Il se trouve dans le dossier "logicielsTiers\Python\testInstallation".

A son exécution, si PyQt a bien été installé, il fait apparaître la fenêtre suivante...



... offrant une petite ihm Qt de démonstration sur un fond de fenêtre MsDos.

En revanche, si l'installation a échoué, c'est cette fenêtre qui apparaîtra...



... pour rester ouverte 10 secondes puis qui disparaîtra.

5.2.9 Influence sur le système et sa sécurité

L'installation de l'interpréteur Python et des bibliothèques annexes ne compromet en rien l'intégrité et la fiabilité du système.

5.3 Installation de l'interpréteur Python sur système d'exploitation Linux

Fera l'objet d'une rédaction ultérieure si nécessaire.

6 ADMINISTRATION DE LA BASE DE DONNEES

L'administration de la base de données est de la compétence de l'administrateur du réseau sur lequel est installé PostgreSQL. Un manuel d'utilisation et d'administration PostgreSQL au format PDF et nommé "pg91.pdf" se trouve dans le dossier "logicielsTiers\PostgreSQL\doc" du cdrom d'installation.

6.1 Bref aperçu des fichiers de configuration PostgreSQL

Ce paragraphe a pour but de décrire brièvement les différents fichiers de configuration PostgreSQL et donner un exemple de paramétrage type.

Ce paragraphe n'a pas pour but de se substituer aux manuels d'administration PostgreSQL

Les fichiers de configuration se trouvent dans le dossier "data" de PostgreSQL et sont au nombre de 3

- ✓ postgresql.conf : configuration générale PostgreSQL
- ✓ pg_hba.conf : type d'authentification requis par les utilisateurs PostgreSQL
- ✓ pg_ident.conf : translation entre "nom externe" et "nom interne"

Un exemplaire type de ces fichiers se trouve dans le dossier "logicielsTiers/PostgreSQL/config" du cdrom d'installation.

6.1.1 Fichier postgresql.conf

Ce fichier définit la configuration générale PostgreSQL. On trouvera entre autres la mémoire allouée, le port utilisé, le nombre de connexions maximales, etc. Il est déconseillé de modifier les valeurs par défaut de ce fichier sans avoir une bonne connaissance de leurs implications.

Toutefois, une valeur pourra être modifiée sans trop de souci : la valeur "listen_adresses". Cette valeur indique la liste des cartes réseaux écoutées par le serveur ; ces cartes étant représentées par l'adresse IP associée. La valeur par défaut est "*" (étoile) indiquant une autorisation générale. Rien n'interdit de restreindre cette liste en n'y indiquant que les adresses réellement écoutées.

6.1.2 Fichier pg_hba.conf

Ce fichier définit la politique d'authentification des clients PostgreSQL. Cette politique se définit en fin de fichier par des lignes au format texte formatées de la façon suivante :

- ✓ Un mot clef "local" (si le client se connecte au serveur via un socket local) ou "host" (si le client se connecte via réseau tcp/ip).
- ✓ Le nom de la base de données sur laquelle le client est autorisé à se connecter (ou "all" pour toute base de données).
- ✓ Le nom de l'utilisateur qui est autorisé à se connecter (ou "all" pour tout utilisateur).
- ✓ La plage d'adresses ip autorisées à se connecter. Cette plage se définit par une adresse de base suivi d'un / (slash) et du nombre de bits du masque sous-réseau (ex: 190.100.100.0/24 => toute adresse ip qui, masquée par les 24 premiers bits, donnera 190.100.100.0 sera autorisée à se connecter)
- ✓ La méthode d'authentification requise ; à choisir parmi "trust" (pas d'authentification requise), "password" (mot de passe requis mais circulera en clair sur réseau), "md5" (mot de passe chiffré par md5), "ldap" (authentification gérée par un annuaire ldap, "krb5" (authentification gérée par un module kerberos), "ident" (authentification gérée par le fichier pg_ident.conf) ou "reject" (authentification rejetée d'office).

La configuration préconisée pour les postes utilisant les applications est de mettre la ligne suivante

host	all	all	ip/mask (à affiner pour que tous les clients soient visibles)	md5
------	-----	-----	---	-----

6.1.3 Fichier pg_ident.conf

Ce fichier définit un mapping possible entre "nom d'utilisateur logique" et "nom d'utilisateur PostgreSQL". Il permet de créer un utilisateur PostgreSQL et de définir un alias (pour raison quelconque comme par exemple plus de commodité) associé à cet utilisateur.

Le client PostgreSQL ne connaît que son alias et celui-ci est converti en client PostgreSQL connu par le serveur.

Ce fichier n'est pas utilisé par mais rien n'interdit à un administrateur de l'utiliser pour lui-même.

6.1.4 En cas de non démarrage du service PostgreSQL

Trop jouer avec les fichiers de configuration ci-dessus peut provoquer un plantage du service PostgreSQL sous Windows. Il ne veut alors plus démarrer, et ce même après une restauration des fichiers d'origine.

Une solution testée est de

- ✓ Renommer le dossier "data"
- ✓ Lancer une fenêtre Dos en tant qu'utilisateur "postgres" (taper le mot de passe donné lors de l'installation)
- ✓ Taper la commande "initdb -D chemin_du_dossier_data => cette commande va recréer un dossier "data" vierge
- ✓ Copier les fichiers "postgres.conf", "pg_hba.conf" et "pg_ident.conf" du dossier "data" nouvellement créé dans l'ancien dossier "data"
- ✓ Effacer le nouveau dossier "data" créé par la commande "initdb"
- ✓ Renommer l'ancien "data" en "data"
- ✓ Relancer le service PostgreSQL

6.2 Modification du mot de passe postgres

Le compte "postgres" utilisé dans les diverses commandes "psql" vues jusqu'à présent est un compte critique pour le serveur PostgreSQL car il a tout pouvoir sur les bases de données gérées par ce serveur. La modification du mot de passe de cet utilisateur particulier revêt donc une importance sans égale.

La modification du mot de passe se fait par commande psql. Il suffit donc d'ouvrir d'abord une fenêtre "invite de commande" (menu "Démarrer" => "Programme" => "Accessoires") et d'exécuter la commande suivante:

```
echo alter role postgres password 'le nouveau mot de passe'; |psql -U postgres -W -h <nom ou adresse IP du serveur PostgreSQL> -p <n° de port défini lors de l'installation>
```

La commande demandera alors d'entrer le mot de passe postgres courant (sauf si cette restriction n'a pas été demandée dans le fichier "pg_hba.conf") et le remplacera par le nouveau mot de passe choisi.

6.3 Sauvegarde et restauration d'une base de données

6.3.1 Sauvegarde

La sauvegarde d'une base de données Postgres se fait au travers de la commande pg_dump. Charge à l'administrateur de se familiariser avec les options de cette commande.

Ce document propose cependant un petit outil sous forme de deux scripts permettant d'automatiser la sauvegarde. Le premier script écrit en Python peut être utilisé pour sauvegarder une base de données quelconque

```
#!/usr/bin/env python
# -*- coding: utf-8 -*-

# Script de sauvegarde de la base de données de l'application
# Auteur: Frédéric Lang (fr.lang@free.fr)

# Usage: prog [-c] [-d] [-m] [-w] [-y] [-h host] [-p port] [-o fic] [-?] [bdd ...]
# Option -c: Sauvegarde attributs complets
# Option -d: Sauvegarde journalière
# Option -m: Sauvegarde mensuelle
# Option -y: Sauvegarde annuelle
# Option -w: Sauvegarde hebdomadaire
# Option -h host: Sauvegarde à partir du host "host"
# Option -p port: Connexion sur le port "port"
# Option -o fic: Sauvegarde dans un fichier "fic"
# Option -?: Utilisation de ce programme
#

# Compatibilité python versions futures
from __future__ import print_function          # Instruction "print" devient fct
#from __future__ import unicode_literals      # Toutes chaînes en unicode
#from __future__ import absolute_import       # Les imports sont en absolu

# Modules Python
import os                                       # Gestion os
import sys                                     # Paramètres et fonctions systèmes
import time                                    # Gestion temps
import getopt                                  # Gestion options Posix

# Fonction "usage"
def usage():
    print("Usage: %s [-c] [-d] [-m] [-w] [-y] [-h host] [-p port] [-o fic] [-?] [bdd ...]" % os.path.basename(sys.argv[0]))
    print("\tOption -c: attributs complets")
    print("\tOption -d: sauvegarde journalière")
    print("\tOption -m: sauvegarde mensuelle")
    print("\tOption -w: sauvegarde hebdomadaire")
```



```

print("\tOption -y: sauvegarde annuelle")
print("\tOption -h host: Sauvegarde à partir du host 'host'")
print("\tOption -p port: Connexion sur le port 'port'")
print("\tOption -o fic: Sauvegarde dans fichier \"fic.sql\"")
# usage()

# Déclaration des variables
bdd_default="postgres"
periode_default=('y')

# Base de donnée par défaut
# Période par défaut

dir_dump="."
# Répertoire de sauvegarde

user="postgres"
# Utilisateur

# Date du jour
date=time.gmtime(time.time())

# Paramètres de sauvegarde
mois=(
    "janvier", "fevrier", "mars", "avril",
    "mai", "juin", "juillet", "août",
    "septembre", "octobre", "novembre", "decembre"
)
paramSvg={
    "d" : {"type" : "day", "date" : "%04d_%02d_%02d" % (
        date.tm_year,
        date.tm_mon,
        date.tm_mday
    )
    },
    "m" : {
        "type" : "month", "date" : "%04d_%02d(%)s" % (
            date.tm_year,
            date.tm_mon,
            mois[date.tm_mon - 1]
        )
    },
    "w" : {"type" : "week", "date" : "%04d_%02d" % (
        date.tm_year,
        date.tm_yday / 7 + 1
    )
    },
    "y" : {"type" : "year", "date" : "%04d" % date.tm_year},
}

# Gestion des options
optItem={
    "periode" : [],
    "-c" : False,
    "-h" : "/tmp",
    "-o" : "",
    "-p" : 5432,
}
try:
    (optList, args)=getopt.getopt(sys.argv[1:], "cdh:mo:p:wy?")
except getopt.GetoptError, eClass:
    print(eClass)
    usage()
    sys.exit(1)
# try

for (opt, val) in optList:
    if opt == "-c":
        optItem[opt]=True
    elif opt == "-h":
        optItem[opt]=val
    elif opt == "-o":
        optItem[opt]=val
    elif opt == "-p":
        optItem[opt]=int(val)
    elif opt in ("-d", "-m", "-w", "-y"):
        val=opt.replace("-", "")
        if val not in optItem["periode"]:
            optItem["periode"].append(val)
    elif opt == "-?":
        usage()
        sys.exit(0)
    # if
# for

# Suppression des options de la liste des arguments
sys.argv=[sys.argv[0,] + args

```

```

# Remplissage période sauvegarde
epoch=optItem["periode"] if optItem["periode"] != [] else periode_default

# Affichage infos
print("host=%s" % optItem["-h"])
print("port=%d" % optItem["-p"])
print("epoch=%s" % epoch)

# Boucle sur chaque bdd
for bdd in sys.argv[1:] if len(sys.argv) > 1 else (bdd_default,):
    print()

    # Vérification bdd présente
    #print("use $bdd" |psql -U$user 1>/dev/null 2>&1)
    #if test $? -ne 0
    #then
        #print("Bdd '%s' does not exist => abort !" % bdd)
        #continue
    #fi

    print("Sauvegarde bdd '%s'" % bdd)

    # Boucle sur chaque période
    for periode in epoch:
        # Création du nom de la sauvegarde
        if optItem["-o"] == "":
            name_dump="%s%s.%s_%s.sql" % (
                "%s/" % dir_dump if dir_dump != "" else "",
                bdd,
                paramSvg[periode]["type"],
                paramSvg[periode]["date"],
            )
        else:
            name_dump="%s" % optItem["-o"]
        # if

        # Sauvegarde
        print(
            "      type sauvegarde: %s - %s is dumping..." % (
                paramSvg[periode]["type"],
                os.path.basename(name_dump)
            )
        )
        fp=open(name_dump, "w")
        fp.write("\connect \"%s\" \n" % bdd)
        fp.close()
        os.system(
            "pg_dump" \
            + " --host=%s" % optItem["-h"] \
            + " --port=%d" % optItem["-p"] \
            + " --username=%s" % user \
            + " --superuser=%s" % user \
            + " --no-password" \
            + " --format=p" \
            + " --clean" \
            + " --%s" % ("column-inserts" if optItem["-c"] else "inserts") \
            + " %s" % bdd \
            + " >> \"%s\" " % name_dump
        )
        fp=open(name_dump, "a")
        fp.write("vacuum verbose analyze;\n")
        fp.close()

    # for

```

Une fois enregistré sous un nom quelconque (par exemple "dumpBdd.py"), il suffit de le lancer en lui passant en paramètre

- ✓ -d pour qu'il génère un fichier associé à la date courante
- ✓ -w pour qu'il génère un fichier associé à la semaine courante
- ✓ -m pour qu'il génère un fichier associé au mois courant
- ✓ -y pour qu'il génère un fichier associé à l'année courante

Ce script nécessite aussi la création d'un sous dossier nommé "DossierStockage" qui servira à stocker les sauvegardes

Le second script écrit en batch MsDos sert à lancer le premier avec des paramètres prépositionnés et peut être programmé dans le gestionnaire des tâches planifiées de Windows

```

rem Exemple de sauvegarde de base de données
rem Les bdd_X et Y seront sauvegardées tous les jours. La bdd_Z toutes les semaines.
dump.py -d bdd_X bdd_Y

```

```
dump.py -w bdd_Z
```

Ces deux scripts se trouvent dans le dossier "logicielsTiers\PostgreSQL\config\SauvegardeBdd" du cdrom d'installation.

6.3.2 Restauration

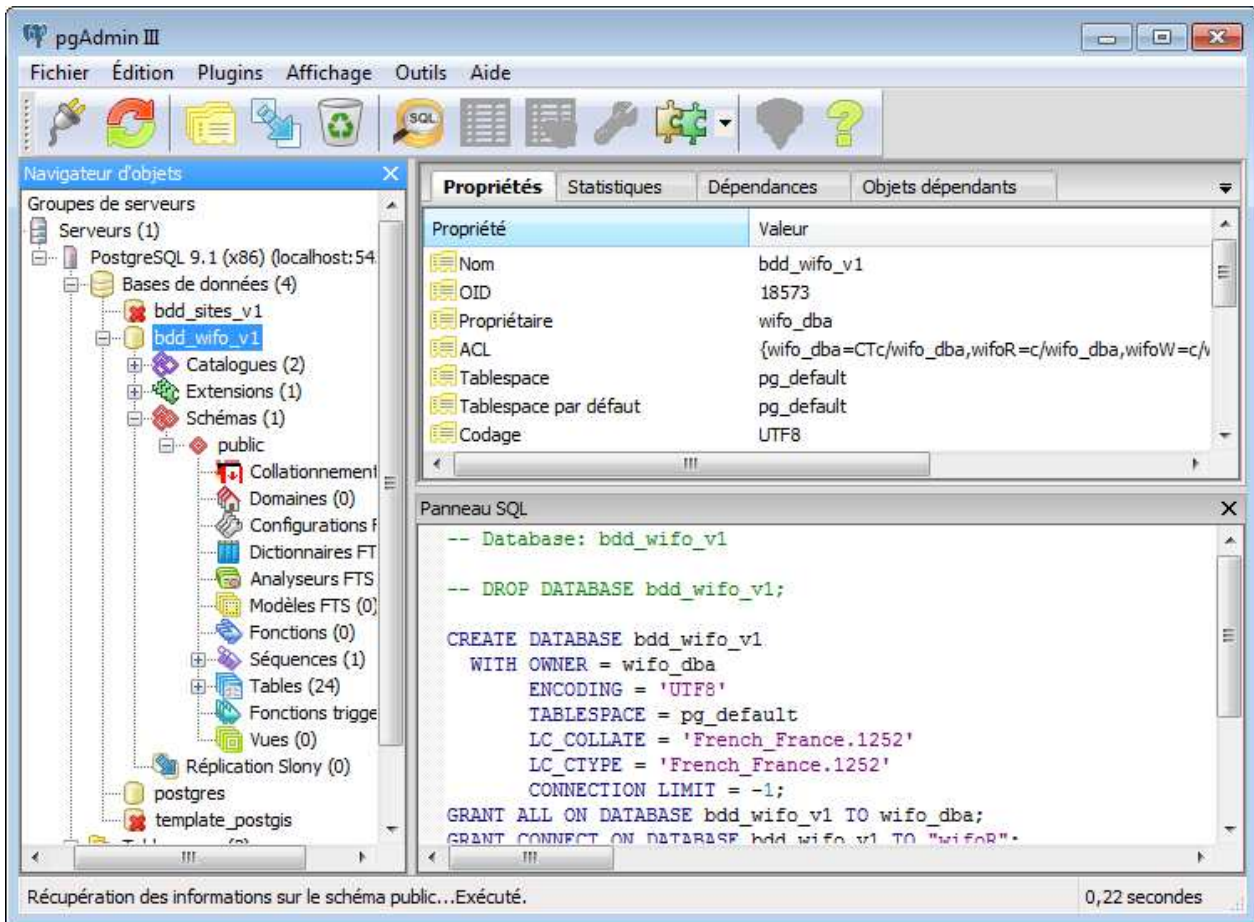
La restauration d'une base de données PostgreSQL se fait à partir d'un fichier créé par la commande `pg_dump`. Il suffit d'aller dans le dossier où se trouve ledit fichier et taper la commande

```
type <nom_du_fichier_de_sauvegarde>.sql |psql -U postgres -W -h <nom ou adresse IP du serveur PostgreSQL> -p <n° de port défini lors de l'installation>
```

Attention, l'exécution de cette commande dure un certain temps qui dépend de la taille du fichier de sauvegarde et de la puissance de l'ordinateur hôte. Compter à peu près trois minutes par tranche de 100Mo. De plus, si la variable `PATH` n'a pas été modifiée pour aller chercher automatiquement les commandes PostgreSQL dans "C:\Development\PostgreSQL\9.3\bin", il sera alors nécessaire de remplacer la commande "psql" par "C:\Development\PostgreSQL\9.3\bin\psql".

6.4 [pgAdmin III](#)

Il s'agit d'un outil graphique permettant d'administrer un serveur PostgreSQL via une interface fenêtrée ayant la forme suivante:



7 GLOSSAIRE

BDD(R) : Base de données (relationnelle).

DBA : « database administrator » : personne qui est en charge de l'administration de base de données

GPL : « Gnu Private Licence » : Licence particulière mise en place par la Free Software Foundation autorisant quiconque à utiliser et redistribuer librement le logiciel qui en est couvert. Connue aussi sous le nom de "licence libre".

IMD : Interface de manipulation des données. Il s'agit d'une interface homme/machine (IHM) qui permet de manipuler les données d'une base au travers d'un client léger (par exemple un navigateur Internet) ou bien un client lourd (un logiciel dédié).

PostgreSQL : Nom de la communauté offrant les différentes versions du moteur de gestion de base de données de même nom, moteur développé sous licence GPL à partir du moteur "Ingres"

PL/SQL : « Programming language/structured query language » langage de programmation du SGBD permettant d'écrire des fonctions et des procédures de bases de données.

SGBD : Système de gestion de base de données (ex.: Oracle, Ingres, PostgreSQL, MySQL)

SQL : « structured query language » : langage structuré de manipulation de base de données

Fin du document