CONCLUSION GENERALE :

Conformément à la demande de la Vice-primature en charge du Développement et de l'Aménagement du Territoire, le cadrage général de l'OAT a été établi et même sa présentation a été déjà effectué et validé par les acteurs de l'Aménagement du Territoire et les observatoires existants à Madagascar ; un prototype du SIL a été conçu et mise en œuvre afin de simuler le fonctionnement de l'activité de l'OAT. Il possède alors son outil de travail afin de lui permettre d'effectuer effectivement ses missions et d'atteindre ses objectifs. Néanmoins, le système n'est pas encore parfait et donc à améliorer en faisant des comparaisons et en recueillant les expériences des autres pays qui ont déjà développé leur système.

Au terme de cette étude de mise en place, l'OAT reste encore une structure potentielle non exploité pour Madagascar, sa base de données n'est pas encore alimentée par des données à jour et fiables, les personnels techniques d'exécution ne sont pas encore à sa disposition, le remplissage de ces conditions assurera son opérationnalisation.

Ces constats généraux nous ont permis de réfléchir à une nouvelle étude qui consiste à mettre en place un Observatoire régional de l'Aménagement du territoire pour acheminer les données à partir de la base, mais également à une organisation d'un deuxième atelier pour montrer aux partenaires potentiels la réalisation du système et de les faire comprendre la nécessité de la synergie d'action pour rendre opérationnel le système.

En conclusion, les études et les recherches effectuées lors de la réalisation de ce travail de mémoire nous ont fourni de solides connaissances sur le SIG et la Géomatique, qui nous a permis de résoudre un problème dans un monde professionnel. Et cela constitue un acquis et une expérience pour notre future carrière professionnelle.

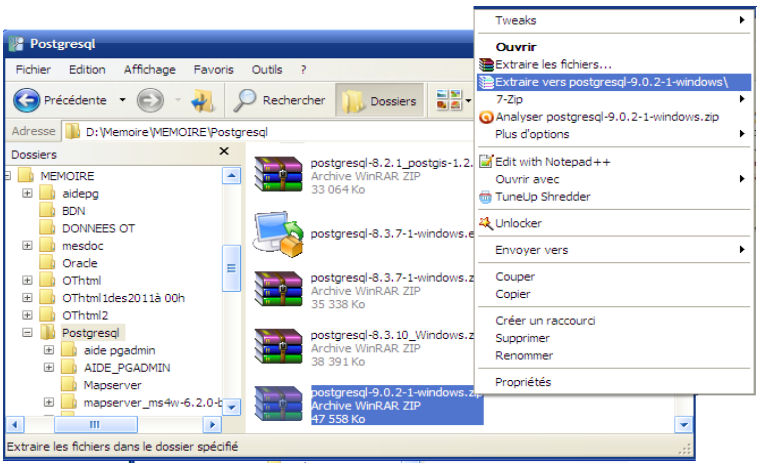
BIBLIOGRAPHIE

Annexe 1 : INSTALLATION DU LOGICIEL POSTGRESQL/POSTGIS

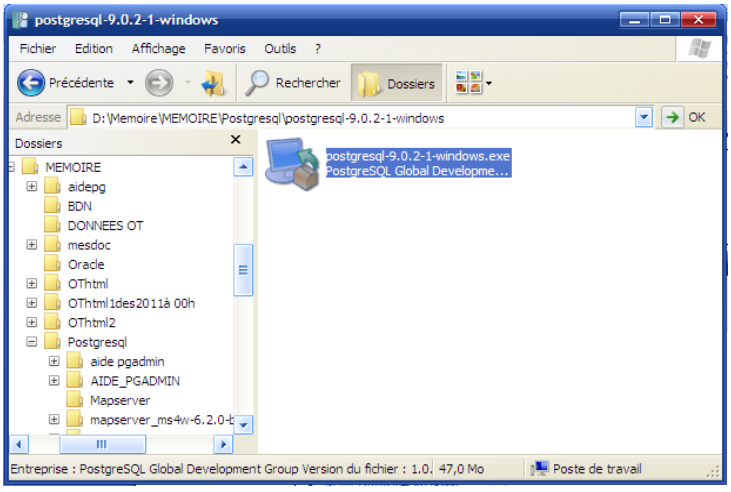
Le logiciel est libre, nous allons mettre en ligne notre ordinateur pour obtenir la source qui est disponible sur le lien : http ://wwwmaster.postgresql.org/download/mirrors-ftp.

Il existe plusieurs versions disponibles pour le logiciel mais nous avons utilisé l'avant dernier version Postgresql 9.0 que nous trouverons le plus stable et à jour.

Le fichier téléchargé est un fichier d'archive *.zip que nous stockons dans le répertoire Postgresql de notre disque dur.



Après extraction du fichier compressé, nous lançons l'application en double cliquant sur le fichier exécutable

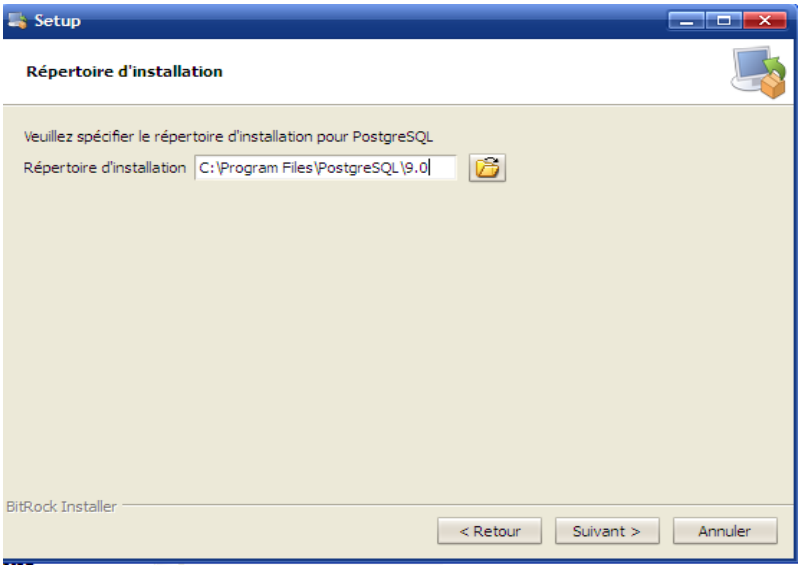


Ainsi, une fenêtre de bienvenu s'ouvre pour indiquer le début d'installation.

On clique tout simplement sur **« suivant »** pour continuer l'installation

Nous acceptons le répertoire par défaut proposé par l'assistant d'installation du logiciel.

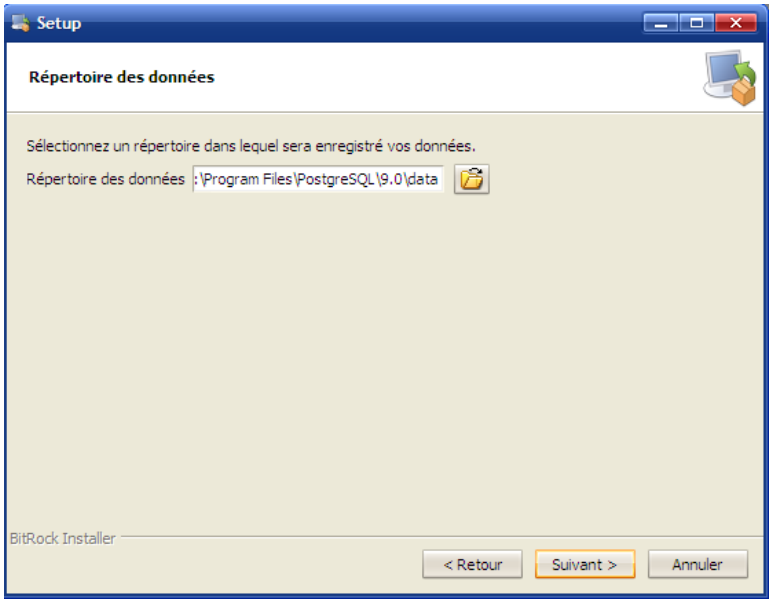
Ici C:\Program Files\PostgreSQL\9.0



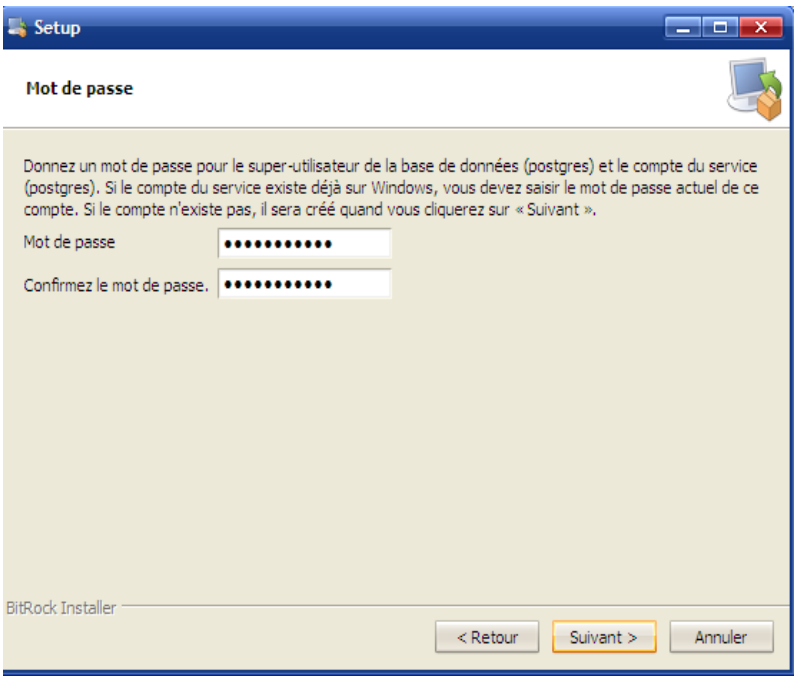
Acceptons également le répertoire par défaut proposé pour l'enregistrement des données.

Ici C:\Program Files\PostgreSQL\9.0\data

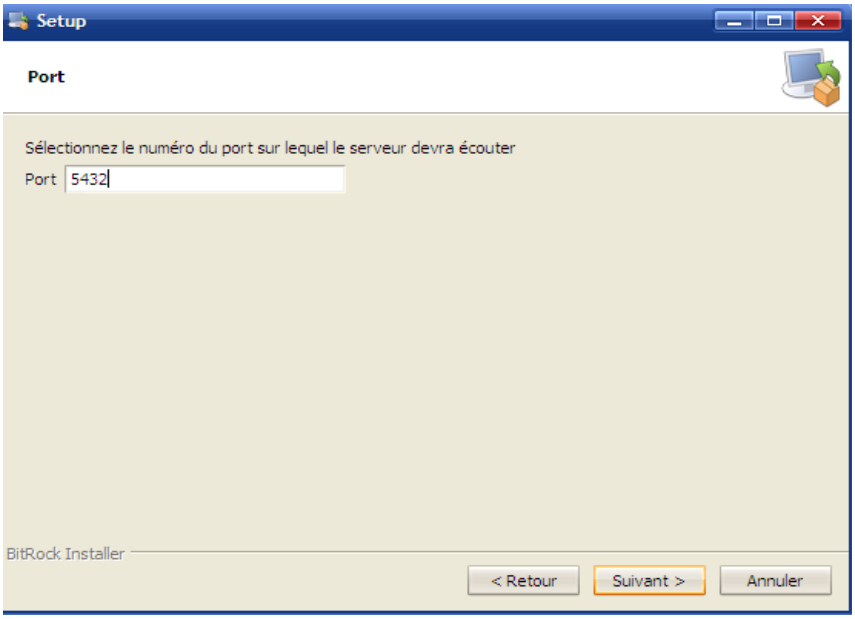
Poursuivons l'installation en cliquant sur « suivant »



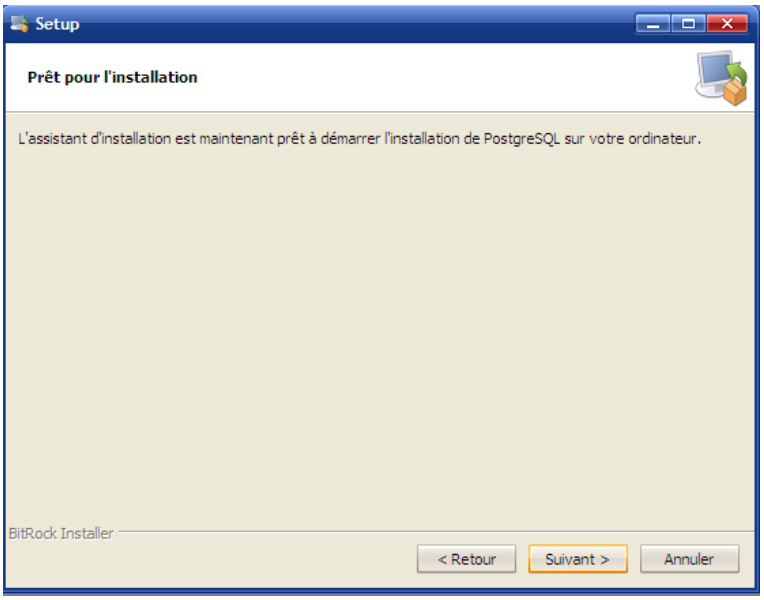
L'installation se poursuivra après avoir tapé un mot de passe qui doit être la même avec celui du compte administrateur et de cliquer sur «suivant »



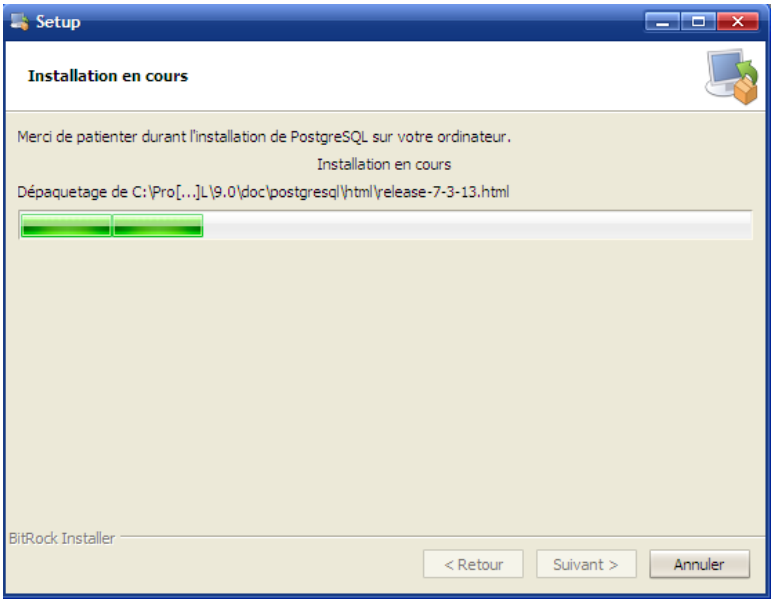
Nous continuons l'installation en acceptant le port d'écoute par défaut et de cliquer sur « suivant »



L'installation démarrera en cliquant sur « suivant »

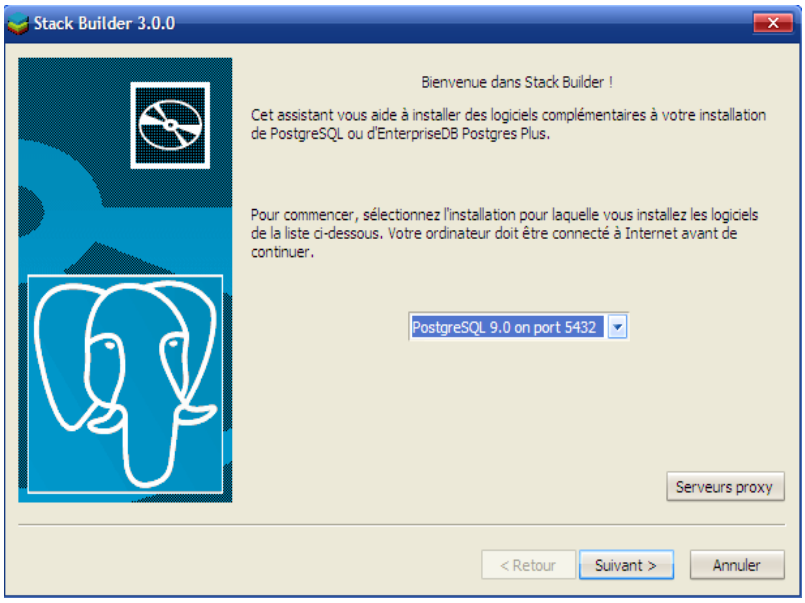


L'installation se fait en quelques minutes, nous devons patienter.

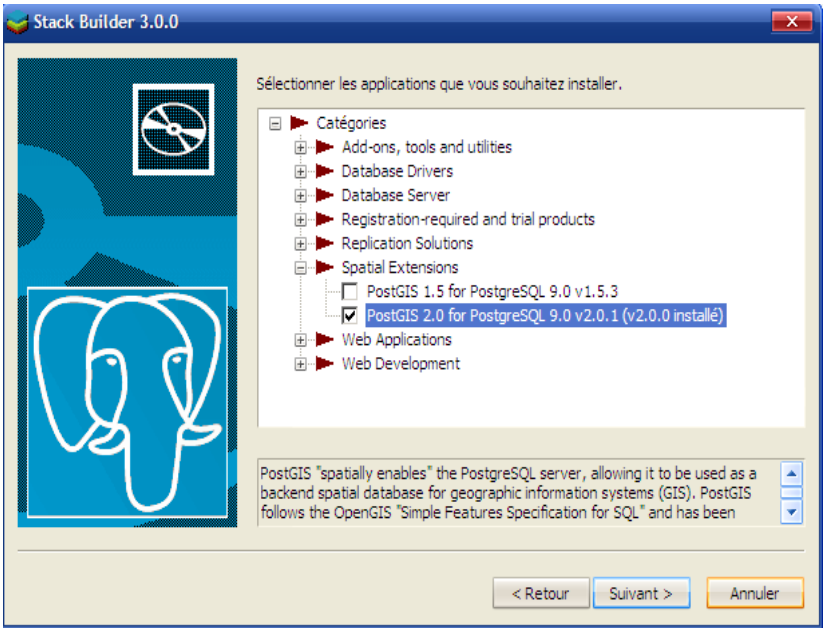


A la fin de l'installation de PostgreSQL, l'assistant d'installation va nous demander si nous lançons l'application « stackBuilder » afin d'ajouter tous les modules compatibles de cette version : cartouche spatiale PostGIS, application web, developpement web, etc. Pour cela, nous laissons l'assistant finir l'installation et de lancer « stackBuilder » en appuyant sur « Terminer » comme l'illustre la figure ci- dessous.

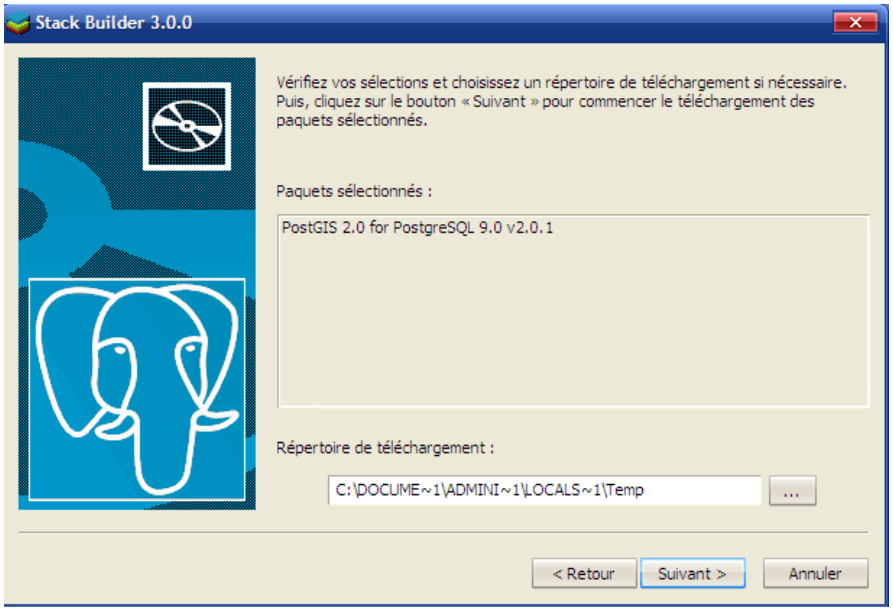
Par la suite nous choisissons l'instance du PostgreSQL et de cliquer sur suivant.



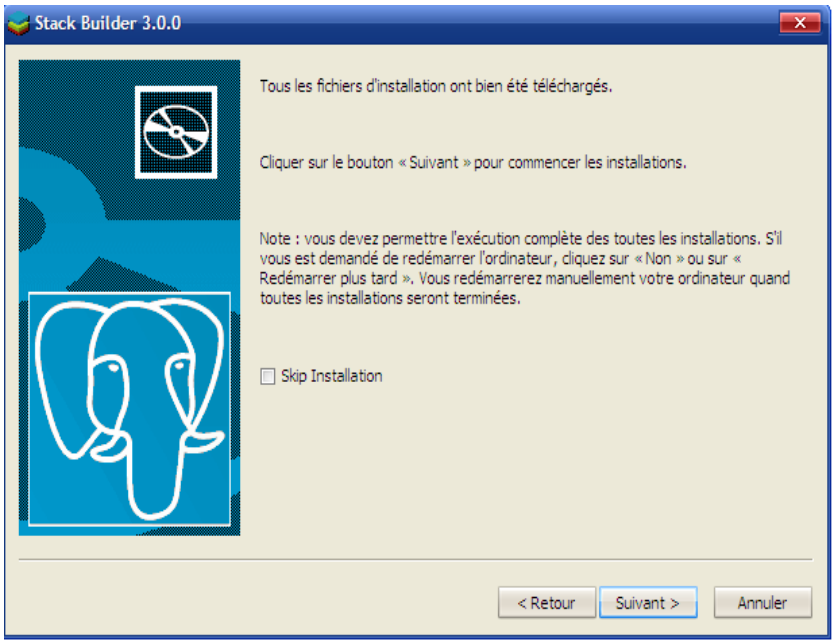
Cocher « PostGIS2.0 » dans la catégorie « Spatial Extensions » (Figure 2) et cliquer sur « Suivant ». Par suite, on suit l'assistant d'installation jusqu'à ce que la cartouche spatiale soit installée.



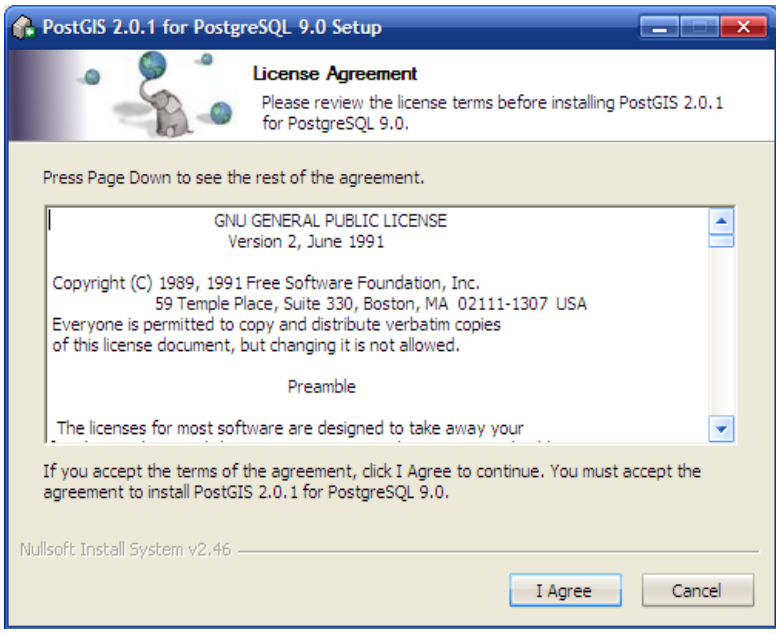
Laissons le répertoire de téléchargement par défaut et cliquer sur « Suivant ». Le fichier est téléchargé



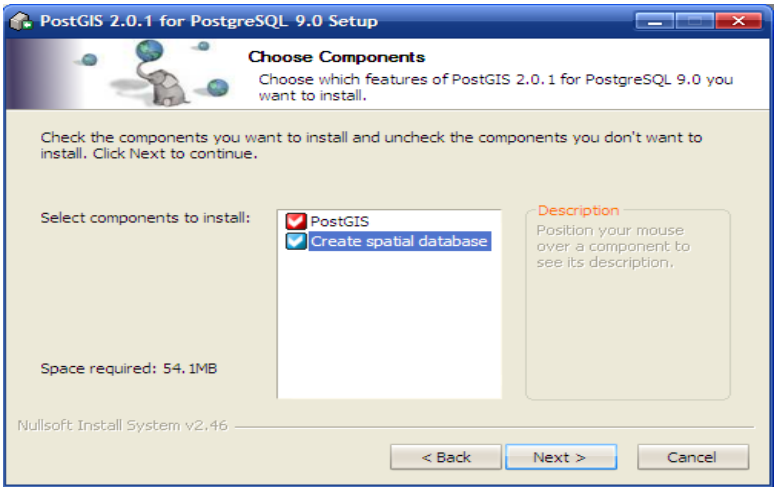
Une fois le fichier téléchargé, ne pas cocher « Skip Installation » et cliquer sur « Suivant ».



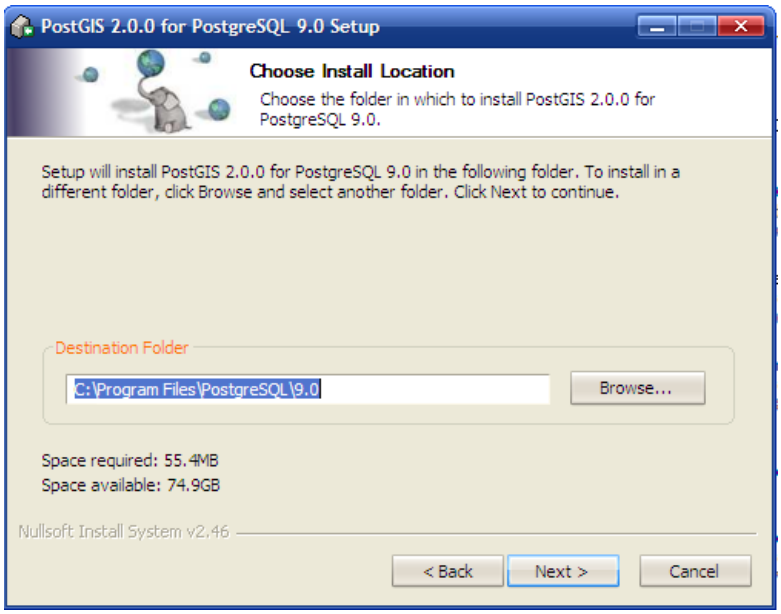
Accepter la licence en cliquant sur « I Agree ».



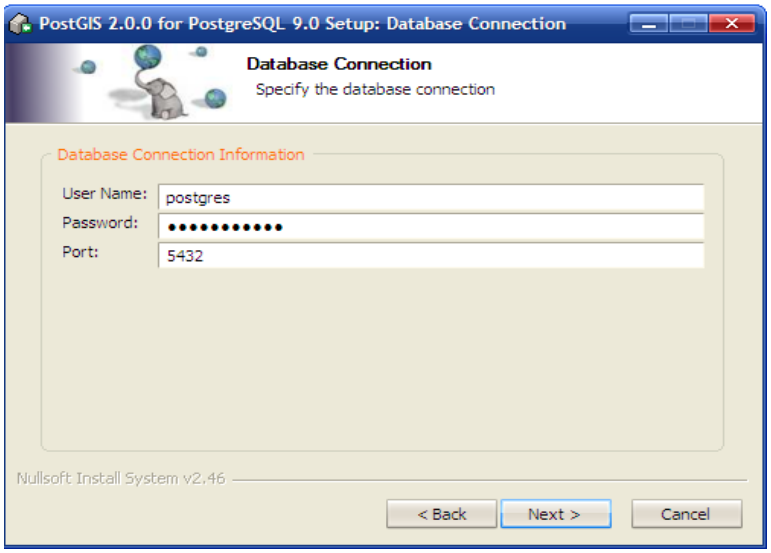
Laisser tous les composants sélectionnés et cliquer sur « Next ».



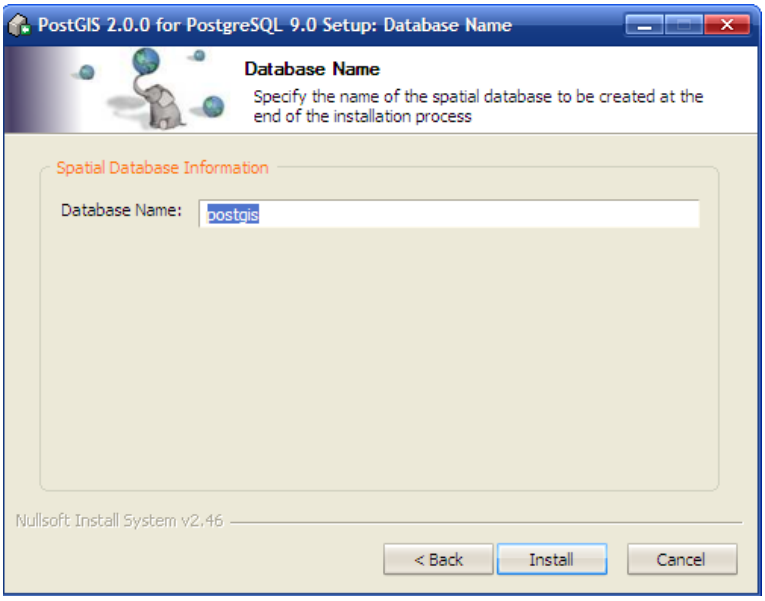
Vérifier que le chemin vers PostgreSQL est correct et cliquer sur « Next ».



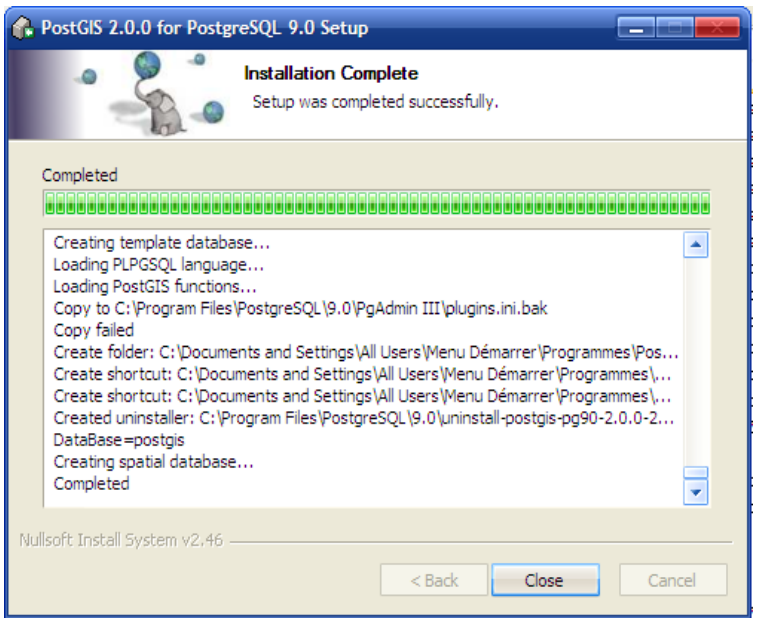
Entrer le mot de passe de l'utilisateur « postgres » et cliquer sur « Next ».



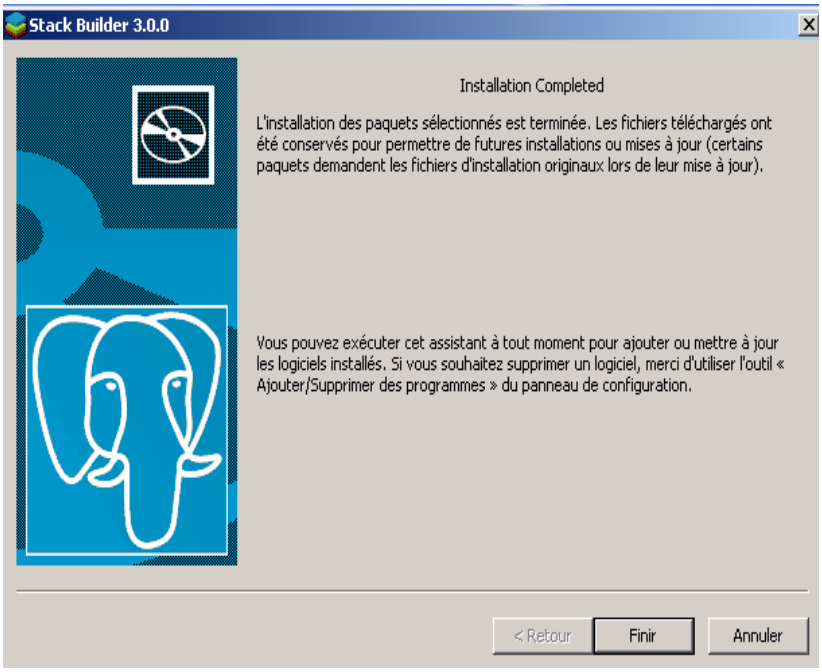
Entrer le nom de la base de données spatiale à créer : « postgis» et cliquer sur « Install ».



Cliquer sur « Close » pour terminer l'installation



Cliquer sur « Finir » pour fermer « StackBuilder »



Annexe 2 : DESCRIPTION DU MAPFILE

❖ **le bloc MAP :**

C'est le bloc racine du mapfile, il va contenir les paramètres concernant la globalité de la carte à générer.

Il contient les paramètres suivants : NAME, UNITS, EXTENT, STATUS, FONTSET et SYMBOLSET, SHAPEPATH, IMAGETYPE, SIZE, RESOLUTION, IMAGECOLOR.

Paramètres généraux :

NAME : c'est le nom de la carte, préfixe ajouté à tous les fichiers images générés par le mapfile, donc à garder de courte taille, et entre guillemets.

UNITS : L'unité de la carte, utilisée pour les calculs et le dessein de la barre de l'échelle.

L'information sur l'unité du fond de carte n'est pas toujours apportée par le bloc « PROJECTION », il faut essayerdonc de le fournir explicitement à chaque fois.

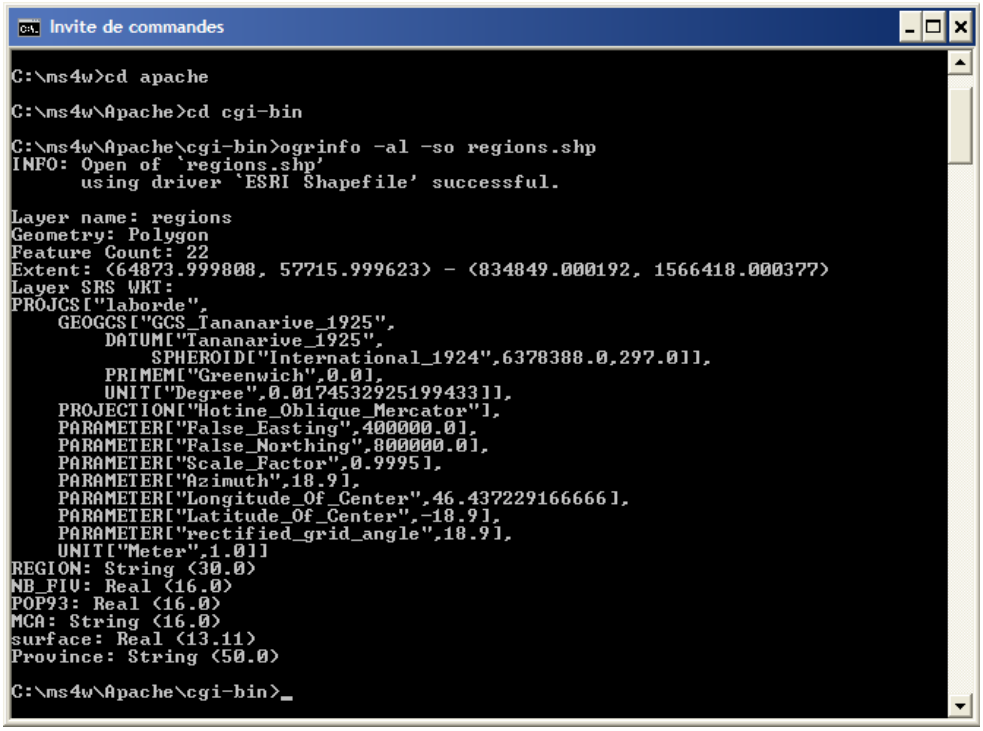
EXTENT : L'extension de la carte, donc coordonnées des extrémités de la carte dessinée dans la carte générée par mapfile.

Ordre : [xmin] [ymin] [xmax] [ymax]

Pour déterminer ces valeurs, il fallait utiliser l'utilitaire « ogrinfo » qui se situe dans le répertoire C:/ms4w/Apache/cgi-bin/.

Avant d'exécuter la commande, les fichiers shapes doivent se trouver dans ce même répertoire.

La figure ci-dessous illustre ce principe :



STATUS : On ou Off, active ou désactive la totalité du mapfile.

FONTSET et SYMBOLSET : chemin et nom de fichier contenant les définitions de polices de caractères et de symboles. On peut aussi définir directement un symbole dans le mapfile, mais c'est utile de pouvoir les rassembler dans un seul fichier qui pourra être appelé par différents mapfiles.

SHAPEPATH : chemin vers les répertoires contenant les fichiers de données (des couches), ou le chemin de base à partir duquel les paramètres DATA iront les chercher.

Paramètres de génération de l'image

IMAGETYPE : type de fichier image à générer. Correspond à l'ensemble inclus par défaut dans le bloc output format. Ce paramètre dépend des options avec a été compilé MapServer.

Valeurs : [gif|png|jpeg|bmp|gtiff|swf|user_defined]

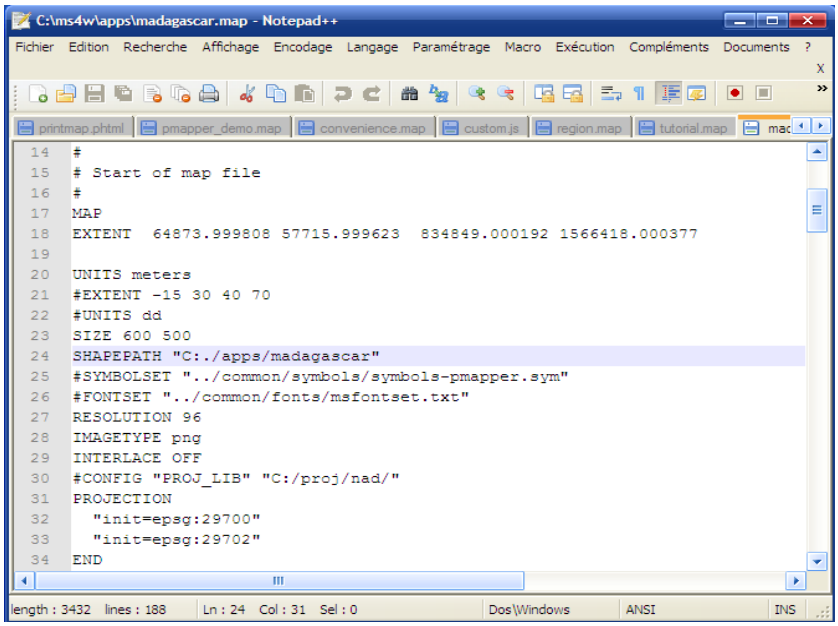
SIZE : dimensions (largeur et hauteur) de l'image à générer, en pixels, séparées par un espace. En général, ce sont des valeurs de la largeur et de la hauteur de l'image.

RESOLUTION : résolution de l'image en dpi, par défaut c'est 72. N'affecte que les calculs d'échelle.

IMAGECOLOR : couleur de fond de l'image de la carte (RGB).

Le bloc MAP va contenir, en plus des paramètres généraux, les blocs de second niveau, qui vont permettre de spécifier le contenu de la carte.

La figure ci-dessous nous montre un exemple concret d'un bloc MAP :



❖ **Le bloc WEB :**

Ce boc va paramétrer le fonctionnement du serveur web autour de la carte générée par MapServer.

EMPTY [url] : URL à renvoyer à l'utilisateur si une requête échoue. Si elle n'est pas définie la valeur du paramètre ERROR est utilisée.

ERROR [url] : URL à renvoyer à l'utilisateur si une erreur arrive. Les vieux messages moches d'erreur de MapServer apparaitront si la valeur n'est pas définie.

FOOTER [nom de fichier] : Modèle à utiliser après que tout ait été envoyé. En mode requête de plusieurs résultats seulement.

HEADER [nom du fichier] : Modèle à utiliser AVANT que tout soit envoyé. En mode requête de plusieurs résultats seulement.

IMAGEPATH [chemin] : Chemin vers le répertoire temporaire pour l'écriture des fichiers et images temporaires. L'écriture doit être possible par l'utilisateur du serveur web. Doit se terminer par un / ou en fonction de la plateforme.

IMAGEURL [chemin] : URL de base pour IMAGEPATH. C'est l'URL qu'utilisera le navigateur web vers IMAGEPATH pour obtenir les images.

```
#
# Start of web interface definition
#
WEB
  TEMPLATE "map.html"
  IMAGEPATH "/ms4w/tmp/ms_tmp/"
  IMAGEURL "ms_tmp/"
  METADATA
    #"MAPFILE_ENCODING" "ISO-8859-1"
    "ows_title"     'MADAGASCAR500'
    "ows_onlineresource" 'http://localhost/cgi-bin/mapserv.exe?map=/ms4w/apps/madagascar.map&mode=map'
    "ows_srs"       'EPSG:29700'
  END  # Metadata
END  # Web
```

❖ **Le bloc REFERENCE**

Trois types de cartes de référence sont gérés. La plus commune serait celle montrant l'étendue d'une carte dans une interface interactive. Il est également possible d'interroger des cartes de référence comme partie d'une requête. Les requêtes ponctuelles génèreront une image avec un marqueur (voir plus bas) placé sur le point de requête. Les requêtes basées sur une région montreront l'étendue de la zone d'intérêt. Enfin, les requêtes basées sur les features afficheront la sélection des features utilisées.

COLOR [r] [g] [b] : Couleur dans laquelle la boîte de référence est dessinée. Définissez tous les composantsà -1 pour avoir une couleur transparente. Rouge par défaut.

EXTENT [minx][miny][maxx][maxy] : L'étendue spatial de l'image de base pour la référence.

IMAGE [nom du fichier] : Nom du fichier complet de l'image de base pour la référence. Ce doit être une image GIF.

MARKER [entier|string] : Définie un symbole (à partir du fichier de symboles) à utiliser lorsque la boîte devient trop petite (voir MINBOXSIZE et MAXBOXSIZE plus bas). Utilise une croix par défaut.

MARKERSIZE [entier] : Définie la taille du symbole à utiliser au lieu de la boîte (voir MARKER plus haut).

MINBOXSIZE [entier] : Si la boîte est plus petite que MINBOXSIZE (utilise la largeur ou la hauteur de la boîte) alors utilise un symbole définie par MARKER et MARKERSIZE.

MAXBOXSIZE [entier] : Si la boîte est plus grande que MAXBOXSIZE (utilise la largeur ou la hauteur de la boîte) alors ne dessine rien (souvent l'ensemble de la carte est couverte lorsque vous zoomez en arrière et il est parfaitement évident de l'endroit où vous êtes).

OUTLINECOLOR [r] [g] [b] : Couleur à utiliser pour le contour de la boîte de référence. Définissez une valeur à -1 pour ne pas avoir de contour.

SIZE [x][y] : Taille, en pixel, de l'image de base de la référence.

STATUS [on|off] : Est-ce que la carte de référence doit être créée ? Off par défaut.

```
#
# Start of Reference map definition
#
REFERENCE
  EXTENT 64873.999808 57715.999623  834849.000192 1566418.000377
  IMAGE "../../images/reference.png"
  SIZE 199 149
  COLOR -1 -1 -1
  OUTLINECOLOR 255 0 0
END  # Reference
```

- **Le bloc LEGEND :**

La taille de l'image de la légende n'est pas connu avant sa création, soyez donc attentif de ne pas coder en dur la largeur et la hauteur dans la balise <img> du fichier modèle

IMAGECOLOR [r] [g] [b] : Couleur d'initialisation de la légende (i.e. l'arrière-plan).

INTERLACE [on|off] : [on] par défaut. Ce mot-clé est maintenant déprécié en faveur de l'utilisation de la ligne FORMATOPTION "INTERLACE=ON" dans la déclaration OUTPUTFORMAT.

LABEL : Signal le début d'un objet LABEL.

OUTLINECOLOR [r] [g] [b] : Couleur à utiliser pour le contour de la boîte des symboles.

POSITION [ul|uc|ur|ll|lc|lr] : Où placer une légende incluse dans la carte. lr par défaut.

KEYSIZE [x][y] : Taille des boîtes de symboles en pixels. 20 par 10 par défaut.

KEYSPACING [x][y] : Espace entre les boîtes de symboles ([y]) et les étiquettes ([x]) en pixels. La valeur par défaut est 5 par 5.

POSTLABELCACHE [true|false] : Demande à MapServer de réaliser le rendu de cette légende après que toutes les étiquettes dans le cache aient été dessinées. Utile pour ajouter des bordures ou des éléments similaires. False par défaut.

STATUS [on|off|embed] : Est-ce que la légende doit être créée ?

TEMPLATE [filename] : Fichier modèle HTML de la légende.

TRANSPARENT [on|off]: Est-ce que la couleur d'arrière-plan doit être transparente. Cette balise est maintenant dépréciée en faveur de la déclaration de la transparence dans les déclarations OUTPUTFORMAT. Off par défaut.

❖ Le bloc PROJECTION :

L'indication d'un géo référencement pour la carte produite par le mapfile nécessite un bloc PROJECTION (LABORDE pour Madagascar). Ce bloc contient les informations relatives au référencement, qui peuvent être de deux formes :

```
PROJECTION
  'proj=omerc'
  'lat_0=-18.9'
  'lonc=44.1'
  'alpha=18.9'
  'k=0.9995000000000001'
  'x_0=400000'
  'y_0=800000'
  'ellps=int1'
  'towgs84=-189,-242,-91,0,0,0,0'
  'pm=2.337229166698544
  'units=m'
  'no_defs'
END
```

ou

```
PROJECTION
  "init=epsg:29700"
  "init=epsg:29702"
END
```

❖ Le bloc SCALEBAR

Mapserver gère les échelles selon une technique assez particulière. En effet, il part du principe que la carte sera une image possédant des dimensions en pixels, qui sera visualisée au moyen d'un écran qui possède une certaine résolution. Par ailleurs, la carte doit être dessinée dans un rectangle d'extension maximale donné par le paramètre EXTENT.

L'échelle finale de la carte doit être donc définie selon ces paramètres. Le paramètre EXTENT prime sur le paramètre SCALE, car c'est lui qui définit plus précisément ce que doit contenir la carte à dessiner.

Pour dessiner une échelle indiquant une certaine longueur terrain, il faut donc déterminer combien de pixels cette longueur va représenter et l'indiquer dans le paramètre size du bloc.

Le bloc SCALEBAR permet de dessiner des barres d'échelle, dans l'image de la carte ou comme une image distincte. Il possède les paramètres suivants :

POSITION : Code à deux lettres définissant l'endroit où sera dessinée l'échelle, ce code est le même que celui utilisé par le paramètre POSITION du bloc LEGEND.

SIZE : dimensions en pixels (largeur espace hauteur) du rectangle contenant la barre d'échelle. Important car détermine la longueur totale de la barre.

INTERVALS : nombre de subdivision à afficher.

STATUS : inclusion (EMBED), dans une image à part (ON) ou annulation (OFF).

STYLE : apparence de la barre, 0 donnant une barre de rectangle plein de couleurs alternées, 1 une barre fine munie de repères (barbules vers le haut).

UNITS : unités pour le calcul de la longueur des intervalles et l'affichage de l'unité de la barre d'échelle. Toutes unités possibles sauf degrés décimaux.

IMAGECOLOR : couleur RGB du rectangle qui contient l'échelle.

BACKGROUNDCOLOR : couleur RGB de la barre d'échelle et de ses libellés.

COLOR : Couleur alternative à BACKGROUNDCOLOR si barre de type 0 et plusieurs intervalles spécifiés.

OUTLINECOLOR : Couleur RGB de la réserve autour de la barre d'échelle (mais pas autour des libellés).

TRANSPARENT : Valeur booléenne (ON/OFF) qui précise si le rectangle contenant l'échelle est transparent.

```
#
# Start of ScaleBar definition
#
SCALEBAR
  STATUS embed
  TRANSPARENT off
  INTERVALS 4
  SIZE 200 3
  UNITS kilometers
  COLOR 250 250 250
  OUTLINECOLOR 0 0 0
  BACKGROUNDCOLOR 100 100 100
  STYLE 0
  POSTLABELCACHE true
  LABEL
    COLOR 0 0 90
    #OUTLINECOLOR 200 200 200
    SIZE small
  END  # Label
END  # SCALEBAR
```

❖ **Le bloc OUTPUTFORMAT :**

Ce bloc permet de définir précisément le format d'image du fichier qui sera généré par MapServer. Le paramètre général IMAGETYPE CORRESPOND en fait à des blocs OUTPUTFORMAT prédéfinis dans MapServer, par défaut. On peut ainsi mieux préciser certains paramètres de la sortie image, par exemple le taux de compression du format de l'image.

```
OUTPUTFORMAT
  # 8 bit PNG
  NAME 'agg_png8'
  DRIVER AGG/PNG
  IMAGEMODE RGB
  FORMATOPTION "QUANTIZE_FORCE=ON"
  FORMATOPTION "QUANTIZE_DITHER=OFF"
  FORMATOPTION "QUANTIZE_COLORS=256"
END
```

```
OUTPUTFORMAT
  NAME GTiff
  DRIVER "GDAL/GTiff"
  MIMETYPE "image/tiff"
  IMAGEMODE RGB
  #FORMATOPTION "TFW=YES"
  #FORMATOPTION "COMPRESS=PACKBITS"
  EXTENSION "tif"
END
```

❖ **Le bloc LAYER** :

Ce bloc va définir les propriétés de création et d'affichage d'une couche de données SIG par mapserver. Les blocs LAYER sont dessinés dans l'ordre du mapfile, c'est-à-dire que le premier bloc du mapfile est dessiné en premier, les suivants viendront par-dessus sur la carte. L'ordre du mapfile est donc l'ordre inverse de la superposition verticale des sources.

Paramètres généraux

NAME : nom de la couche, utilisé comme identifiant dans l'interface web. Il doit être unique dans le mapfile et d'une longueur maximale de 20 caractères. Entre guillemets de préférence(ou apostrophes).

GROUP [name] : Nom d'un groupe auquel la couche appartient. Le nom de groupe peut ensuite être référencé comme un nom normal de couche dans les fichiers de mise en forme ("Template"), ce qui permet de faire certaines opérations comme afficher ou cacher un groupe de plusieurs couches en une seule opération.

METADATA : Ce mot-clé permet d'enregistrer des données arbitraires comme un couple de valeurs. Ceci est utilisé avec OGC WMS pour définir des éléments comme le titre de la couche. Il peut aussi permettre plus de souplesse lors de la création des mises en forme ("Template"), puisque tout ce que vous mettrez ici sera accessible via les balises du Template.

STATUS [on|off|défault] : statut (visibilité) du layer. Doit prendre la valeur « default » pour que le layer soit visible lorsque l'on utilise mapserver en mode map(le statusOn ne suffit pas à le rendre visible, il faut que le layer soit expressément requis).

Définit le statut actuel de la couche. Fréquemment modifié par MapServer lui-même. Default active la couche ("on") en permanence. En mode CGI, les couches avec STATUS DEFAULT ne peuvent pas être désactivées par les mécanismes classiques. Il est recommandé de définir des couches en STATUS DEFAULT lorsque l'on souhaite débuguer un problème puis de les redéfinir à ON/OFF en utilisation normale.

Pour le WMS, les couches dans le mapfile du serveur dont le paramètre STATUS vaut DEFAULT sont toujours envoyées au client.

TYPE [point|line|polygon|circle|annotation|raster|query|chart] :

Définie comme les données doivent être dessinées. Ne doit pas être nécessairement le même que celui du type du shapefile. Par exemple, un shapefile de type polygone peut être dessiné comme une couche point, mais un shapefile de type point ne peut pas être dessiné comme une couche polygone. Règle de bon sens. Annotation signifie qu'un point étiquette sera calculé pour les features, mais la feature en elle-même ne sera pas dessinée bien qu'un marqueur peut être optionnellement dessiné. Cela permet l'étiquetage avancé comme numéroter les autoroutes. Les points sont étiquetés de cette manière. Les polygones sont d'abord étiquetés en utilisant un centroïde, et s'il ne tombe pas dans le polygone une approche de "scan de ligne" est utilisée pour garantir que l'étiquette tombe dans la feature. Les lignes sont étiquetées au milieu de l'arc le plus long dans la portion visible de la ligne. query signifie seulement que la couche peut être interrogée mais pas dessinée.

Dans le but de différencier les POLYGON et les POLYLINE (qui n'existent pas en tant que type), utilisez ou omettez simplement respectivement le mot-clé COLOR lors de la classification. Si vous l'utilisez, c'est un polygone avec une couleur de remplissage, sinon ce sera une polyligne avec seulement un OUTLINECOLOR.

Un cercle doit être défini par une limite rectangulaire minimale. Pour cela, deux points définissent le plus petit carré qui peut y être contenu. Ces deux points sont les deux coins opposés de cette boîte.

MINSCALE [double] – déprécié :

Depuis MapServer 5.0, le bon paramètre à utiliser est MINSCALEDENOM. Le paramètre obsolète MINSCALE était l'échelle minimale à laquelle la couche était dessinée. L'échelle est donnée comme dénominateur de la fraction de l'échelle actuelle. Par exemple pour une carte à l'échelle de 1:24000, il s'agit de 24000.

MAXSCALE [double] – déprécié :

Depuis MapServer 5.0, le bon paramètre à utiliser est MAXSCALEDENOM. Le paramètre obsolète MAXSCALE était l'échelle maximale à laquelle la couche était dessinée. L'échelle est donnée comme dénominateur de la fraction de l'échelle actuelle. Par exemple pour une carte à l'échelle de 1:24000, il s'agit de 24000.

SYMBOLSCALE [double] - déprécié

Depuis MapServer 5.0 le paramètre correct à utiliser est SYMBOLSCALEDENOM. Le paramètre SYMBOLSCALE déprécié est l'échelle à laquelle les symboles et/ou les textes apparaissent à leur taille normale. Cela permet une échelle dynamique des objets en fonction de l'échelle de la carte. S'il n'est pas défini alors cette couche apparaitra toujours à la même taille. La mise à l'échelle a lieu seulement entre les limites comprises entre MINSIZE et MAXSIZE comme décrit plus haut. L'échelle est donnée comme le dénominateur de la fraction de l'échelle, par exemple, pour une carte à l'échelle de 1:24 000 utilisez 24000. Implémenté dans MapServer 5.0, pour remplacer le paramètre déprécié SYMBOLSCALE.

OPACITY [entier|alpha] :

Définit le niveau d'opacité (autrement dit l'incapacité de voir à travers la couche) de tous les pixels classés pour une couche donnée. La valeur peut être soit un entier compris entre 0 et 100 soit le mot-clé nommé "ALPHA". Une valeur de 100 est opaque et une valeur de 0 est complétement transparente. Implémenté dans MapServer 5.0 pour remplacer le paramètre obsolète TRANSPARENCY.

Le mot-clé "ALPHA" indique au moteur de rendu de MapServer de prendre en compte la transparence indexée ou alpha des symboles pixmap utilisés pour styler une couche. C'est uniquement nécessaire en cas de formats de sortie RVB et ne devrait être utilisé que lorsque cela est nécessaire car il est coûteux de rendre transparents des symboles pixmap sur une image de carte RGB.

OFFSITE [r] [g] [b] :

Sélectionne l'index de couleur utilisé pour la transparence des couches raster.

POSTLABELCACHE [true|false] :

Indique à MapServer de dessiner cette couche après que toutes les étiquettes en cache auront été dessinées. Très utile pour ajouter une touche finale et d'autres éléments similaires. Désactivé par défaut.

CLASSITEM [attribut] :

Nom de l'élément dans la table attributaire à utiliser dans les correspondances de classe.

LABELITEM [attribut] :

Nom de l'élément dans la table attributaire à utiliser comme classe d'annotation (par exemple pour l'étiquetage).

TEMPLATE [fichier|url] :

Utilisé comme une alternative globale au TEMPLATE CLASS.

Paramètres de données :

- **Shapefiles**

Les shapefiles sont les sources de données la plus courante pour mapserver, qui au départ était construit uniquement autour de ce format (issu de la bibliothèque shapelib). On indique que la source de données du layer est un shapefile avec le paramètre DATA, qui prend comme valeur le chemin vers le shapefile, relatif au chemin SHAPEPATH .le paramètre FILTER permet d'utiliser les expressions régulières pour filtrer les objets de la source de données au niveau du bloc LAYER, c'est-à-dire avant le traitement des blocs CLASS qui peuvent eux aussi comporte une sélection.

L'utilisation d'une source de données shapefile offre de plus la possibilité de lier (jointure) une table dbf de données attributaires, en utilisant un bloc secondaire JOIN. Ces données deviennent ainsi interrogeables par mapserver en mode Query.

❖ **Couches vectorielles accédées avec OGR**

La bibliothèque de fonctions OGR permet à mapserver de lire un plus grand nombre de formats de SIG vectoriels

Pour indiquer une source de données OGR, il faut utiliser la syntaxe suivante :

CONNECTIONTYPE OGR

CONNECTION [nom de la source]

Ce dernier paramètre dépend du type de source, généralement il s'agit du chemin relatif au SHAPEPATH et du nom de fichier ou du nom de répertoire. Certains types de données vectorielles sont organisés en couches multiples par fichier, il faut donc choisir la couche à utiliser avec le paramètre DATA [numero/nom de la couche]. Mapserver peut aussi récupérer en partie les éventuels styles d'affichage présents dans les couches, avec le paramètre STYLEITEM AUTO. Le paramètre FILTER est utilisable aussi avec ce type de données, comme pour les shapefiles.

❖ ***Sources serveurs de données***

Mapserver peut se connecter à des serveurs SIG directement, et ainsi assembler des couches de diverses origines de façon transparente. Cela permet de construire une organisation des données dispersées sur des serveurs spécialisés. Les paramètres de connexion sont fournis dans une chaîne de caractères passée en valeur au paramètre CONNECTION. Le paramètre FILTER peut contenir une expression de requête SQL c'est-à-dire le texte venant après un élément SQL WHERE.

Syntaxe de connexion entre Mapserver et les serveurs de données

Postgis :

CONNECTIONTYPE postgis

CONNECTION ''user=nom_utilisateurdbname=nom_bdd host=serveur''

DATA 'colonne_géométrique FROM table_ou_requête_SQL'

Oracle Spatial :

CONNECTIONTYPE oraclespatial

CONNECTION 'nom_utilisateur/mot_de_passe@bdd'

DATA 'colonne_géométrique FROM table_ou_requête_SQL'

```
LAYER
        NAME "REGION"
        TYPE POLYGON
        DUMP true
        TEMPLATE fooOnlyForWMSGetFeatureInfo
        EXTENT -64873.999808 57715.999623  834849.000192 1566418.000377
        CONNECTIONTYPE postgis
        CONNECTION "dbname= 'postgis' host=127.0.0.1 port=5432 user=postgres password=oat"
        DATA 'geom FROM "REGION" USING UNIQUE Id_region USING srid=-1'
        FILTER ( GeometryType("geom") IN ('POLYGON','MULTIPOLYGON'))
        METADATA
             'ows_title' 'REGION'
             "description" "REGION"
             "result_fields" "Id_region,nom_region,nb_fiv,pop93,mca,surface,province,geom"
             "result_headers" "Id_region,nom_region,nb_fiv,pop93,mca,surface,province,geom"
             "legendicon" "images/legend/REGION.png"
        END # metadata
        STATUS default
        PROJECTION
        "init=epsg:29700"
        "init=epsg:29702"
        END
        CLASS
             Name 'REGION'
             SYMBOL 0
             SIZE 10.5
             OUTLINECOLOR 170 85 255
             ##COLOR 170 85 255
             END
        END # CLASS
END # LAYER
```

- **Couches raster**

Les données raster géo référencées sont gérées avec le paramètre TYPE raster, le paramètre DATA devant pointer vers le fichier (et son éventuel chemin relatif à SHAPEPATH). Mapserver possède quelques fonctions de traitement des rasters, accessibles avec le paramètre PROCESSING. GDAL est capable de réaliser des traitements de ré échantillonnage, mise à l'échelle, de tramage et de sélection de bande.

Les sources de données raster composées de nombreux fichiers juxtaposés sont gérables en une seule fois par Mapserver, sous la forme d'un tuilage. Ce tuilage consiste en un fichier shape qui contient des objets polygone représentant l'extension de chaque tuile raster, et ayant comme attribut le nom du fichier. Ce type de tuilage peut être produit en utilisant l'utilitaire gdaltindex distribué avec GDAL. Cela permet une gestion beaucoup plus rapide des grandes couvertures raster, nottamment si les bitmaps sont eux aussi optimisés pour être lus à différentes résolutions (index pyramidal dans les tifs par exemple).

- **Le bloc CLASS**

Ce bloc permet de définir des classes thématiques dans la couche, qui vont pouvoir être affichées différemment sur la carte globale. Les blocs CLASS sont traités dans l'ordre du fichier map, selon l'ordre de classement vertical. Ce bloc peut contenir les paramètres suivants :

NAME [string]

Nom à utiliser dans les légendes pour cette CLASS. Si ce n'est pas défini, la CLASS n'apparaitra pas dans une légende.

EXPRESSION **[string] :**

Quatre types d'expressions sont maintenant supportés pour définir l'appartenance à une classe. Les comparaisons de chaînes, les expressions régulières, les expressions logiques simples et les fonctions de chaînes. Si aucune expression n'est donnée, alors toutes les entités sont considérées comme appartenant à cette classe.

- Les comparaisons de chaînes sont sensibles à la casse et sont les plus rapides à analyser. Il n'y a pas de caractère délimiteur à utiliser cependant la chaîne doit être entre quotes si elle contient des caractères spéciaux (Afin de prendre de bonnes habitudes, il est recommandé de mettre entre quotes toutes vos chaînes).
- Les expressions régulières fonctionnent de la même façon que dans les précédentes versions de MapServer. Cependant vous devez dorénavantdélimiter toute expression régulière en utilisant /regex/. Aucune quote ne doit être utilisée.
- Les expressions logiques vous permettent de construire des tests assez complexes basés sur un ou plusieurs attributs et sont donc seulement utilisables avec les fichiers Shape. Les expressions logiques sont délimitées par des parenthèses "(expression)". Les noms des attributs sont délimités par des crochets "[ATTRIBUT]". Ces noms sont sensibles à la casse et doivent correspondre aux objets dans le fichier Shape

COLOR [r] [g] [b] :

Couleur à utiliser pour dessiner les entités.

OUTLINECOLOR [r] [g] [b] :

Couleur à utiliser pour les polygones et certains symboles de marquage. Les symboles de ligne ne gèrent pas ce paramètre.

SYMBOL **[integer|string|filename] :**

Le nom du symbole ou le numéro à utiliser pour toutes les entités si les tables d'attributs ne sont pas utilisées. Le numéro est l'index du symbole dans le fichier de symboles, celui-ci commence à 1, le 5ième symbole dans le fichier est donc le symbole numéro 5. Vous pouvez aussi donner des noms à vos symboles en utilisant le mot-clé NAME dans le fichier de définition des symboles puis l'utiliser pour y faire référence. 0 est la valeur par défaut ce qui produit un simple pixel, une ligne simple fine ou un polygone plein selon le type de la couche.

Vous pouvez aussi spécifier un fichier GIF ou PNG. Le chemin est relatif à l'emplacement du mapfile.

SIZE [integer] :

Hauteur, en pixels, du symbole/motif à utiliser. Seulement utile avec les symboles re-échantillonables. Pour les symboles de type vecteur (et ellipse), la taille par défaut est basée sur la gamme des valeurs de Y dans les POINTS définissant le symbole. Pour les rasters, le défaut est la taille verticale de l'image. La taille par défaut est de 1 pour les symboles TTF

MINSIZE [integer] :

Taille minimale en pixels pour dessiner un symbole. Le défaut est de 0.

MAXSIZE [integer] :

Taille maximale en pixels pour dessiner un symbole. Le défaut est de 50.

TEXT [string] :

Texte fixe pour étiqueter les entités de cette classe. Ceci écrase les valeurs obtenues par LABELITEM. La chaîne de caractères peut être donnée comme une expression délimitée par des parenthèses. Ceci vous permet de concaténer plusieurs attributs en une seule étiquette. Par exemple : ([FIRSTNAME], [LASTNAME])

TEMPLATE **[filename] :**

Fichier template ou URL à utiliser pour présenter les résultats des requêtes à l'utilisateur. Voyez la référence ***Templating*** pour plus d'information.

DEBUG [on|off] :

Active le débogage de l'objet classe. Une sortie verbeuse est générée et envoyée au canal de sortie standard des erreurs (STDERR) ou vers le fichier de trace ("log") de MapServer si celui-ci est précisé dans le paramètre LOG de l'objet *WEB*.

Le bloc CLASS peut contenir des blocs secondaires LABEL et STYLE.

```
CLASS
    Name 'REGION'
    SYMBOL 0
    SIZE 10.5
    OUTLINECOLOR 170 85 255
    ##COLOR 170 85 255
    END
END # CLASS
```

- **Le bloc LABEL :**

Ce bloc permet de configurer l'étiquetage des éléments de la classe, selon un champ attributaire indiqué par le paramètre LABELITEM du bloc LAYER qui contient le bloc LABEL.

Paramètres de base :

TYPE [bitmap|truetype] :

Type de police de caractères à utiliser. Généralement les polices bitmap sont plus rapides à dessiner que les polices TrueType. Cependant les polices TrueType sont redimensionnables et disponibles sous de nombreuses formes. Assurez-vous que le paramètre FONT est activé si vous choisissez "TrueType".

COLOR [r] [g] [b] | [attribute]:

- Couleur à utiliser pour dessiner le texte.
- [*ATTRIBUTE*] a été implémenté dans la version 5.0 pour spécifier le nom de l'attribut dans la table attributaire à utiliser pour les valeurs de couleur. Les crochets [] sont nécessaires

SIZE [integer]|[tiny|small|medium|large|giant]|[attribute]:

- Taille du texte. Utiliser "integer" pour donner la taille en pixels de votre police de caractères TrueType ou utilisez l'un des 5 mots-clés listés pour les polices de caractères bitmap.
- [*Attribute*] a été implémenté dans la version 5.0 pour spécifier le nom de l'attribut dans la table attributaire à utiliser pour les valeurs de taille. Les crochets [] sont nécessaires.

MINSIZE [integer] :

Taille minimale de la police de caractères à utiliser pour redimensionner le texte (en pixels). 4 par défaut.

MAXSIZE [integer] :

Taille maximale de la police de caractères à utiliser pour redimensionner le texte (en pixels). 256 par défaut.

MINFEATURESIZE [integer|auto] :

Taille minimale d'une entité devant être étiquettée, en pixels. Pour les données linéaires, la longueur totale de la ligne affichée est utilisée, pour les entités polygonales la taille du plus petit côté du rectangle englobant est utilisée. Le mot-clé "Auto" indique à MapServer d'étiqueter uniquement les entités plus grandes que leur propre étiquette. Disponible seulement pour les étiquettes en cache.

Paramètres d'effets d'affichage du texte (la présence du paramètre active la fonction) :

ANTIALIAS [true|false] :

Le texte doit-il être antialiasé ? Notez que ceci nécessite un plus grand nombre de couleurs, diminue les performances de rendu et génère des images plus lourdes.

OUTLINECOLOR [r] [g] [b] | [attribute]:

- Couleur utilisée pour dessiner une ligne d'un pixel autour du texte.
- [*ATTRIBUTE*] a été implémenté dans la version 5.0 pour spécifier le nom de l'attribut dans la table attributaire à utiliser pour les valeurs de couleur. Les crochets [ ] sont nécessaires.

OUTLINEWIDTH [integer] :

Largeur du contour si OUTLINECOLOR a été définie. 1 par défaut. Pour l'instant sur le moteur de rendu AGG gère des valeurs supérieures à 1, et rend un contour avec un effet "d'halo" : valeurs recommandées : 3 ou 5.

SHADOWCOLOR [r] [g] [b] :

Couleur de l'ombre.

SHADOWSIZE [x][y] :

Décalage de l'ombre en pixels.

BACKGROUNDCOLOR [r] [g] [b] :

Couleur à utiliser pour dessiner un rectangle d'arrière-plan (panneau d'étiquette ("billboard")). Désactivé par défaut.

Paramètres de positionnement :

POSITION [ul|uc|ur|cl|cc|cr|ll|lc|lr|auto] :

Position de l'étiquette relative au point d'insertion (couches seulement). La première lettre est la position en "Y", la seconde lettre est la position en "X". "Auto" indique à MapServer de calculer une position d'étiquette qui n'interférera pas avec d'autres étiquettes. Avec des points, MapServer sélectionne l'une des 8 positions extérieures (c'est à dire en excluant cc). Avec les polygones, MapServer sélectionne cc (ajouté dans MapServer 5.4), uc, lc, cl et cr comme positions possible. Avec des lignes, il utilise seulement lc ou uc, jusqu'à ce qu'il trouve une position qui n'entraîne pas de chevauchement (collision) avec des étiquettes déjà dessinées. Si toutes les positions entraînent un conflit, alors l'étiquette n'est pas dessinée (sauf si le paramètre d'étiquette FORCEest activé). "Auto" est seulement disponible avec les étiquettes en cache.

ANGLE [double|auto|follow|attribute]:

- Angle, données en degrées, pour dessiner l'étiquette.
- AUTO permet à MapServer de calculer l'angle. Valide uniquement pour les couches de type ligne (LINE).
- FOLLOW a été implémenté dans la version 4.10 et indique à MapServer de calculer une étiquette courbe appropriée aux entités linéaires (cf. la *MS RFC 11: Support for Curved Labels* pour les détails).
- [ATTRIBUTE] a été implémenté dans la version 5.0 pour spécifier le nom de l'attribut dans la table attributaire à utiliser pour les valeurs angulaires. Les crochets [] sont nécessaires

OFFSET [x][y] :

Valeurs de décalage pour les étiquettes, relatives au coin inférieur gauche de l'étiquette ou du point d'insertion de l'étiquette, en pixels. Dans le cas d'un texte tourné (rotation), cette commande spécifie les valeurs comme si toutes les étiquettes étaient horizontales et toute rotation sera compensée.

MINDISTANCE [integer] :

Distance minimale entre des étiquettes en double, en pixels.

BUFFER [integer] :

Remplissage, en pixels, autour des étiquettes. Utile pour maintenir un espace autour du texte pour en améliorer sa lecture. Disponible seulement pour les étiquettes "cachables". 0 par défaut.

FORCE [true|false] :

Force l'affichage des étiquettes d'une classe particulière, sans s'occuper des superpositions (collisions). Disponible seulement pour les étiquettes en cache. Désactivé par défaut.

PARTIALS [true|false] :

Le texte peut-il être caché s'il n'apparait qu'en partie sur la carte ? Activé par défaut.

WRAP [character] :

Caractère qui représente une rupture de ligne dans le texte de l'étiquette et qui entraine la création d'une étiquette multi-ligne. Interagit avec MAXLENGTH pour le retour à la ligne conditionnel après un nombre de caractères donnés

```
LABEL
  POSITION Auto
  COLOR 0 0 150
  BUFFER 2
  TYPE truetype
  ENCODING "UTF-8"
  FONT FreeSans
  SIZE 8
  MAXSIZE 9
  #SIZE small
END #Label
```

Utilisation du Template :

Dans le cadre d'une utilisation sur un serveur web, les auteurs de MapServer ont ajouté une fonctionnalité au logiciel pour lui permettre de générer non seulement une image de carte, mais un document HTML complet pouvant comprendre les divers éléments comme la carte, l'échelle , la légende, les titres et les mentions. Ce fonctionnement utilise un fichier modèle HTML, dans lequel MapServer va modifier des valeurs pour les remplacer par les éléments qu'il va produire. Ce fichier modèle HTML est ce qu'on appelle un ''Template'' dans le vocabulaire de mapserver.

Le fonctionnement des templates est très simple : dans le mapfile on spécifie le chemin vers un fichier template HTML. Ce dernier contient le code HTML de la page dans laquelle vont se trouver les éléments produits par mapserver, ainsi que certaines balises spéciales, qui seront remplacées par le code HTML adéquat par mapserver. La balise la plus importante est la balise [img], qui sera remplacée par mapserver par le chemin vers le fichier image de la carte qu'il génère.

TABLE DE MATIERES

INTRODUCTION……1

PARTIE I : CADRAGE GENERAL DE L'OBSERVATOIRE DE L'AMENAGEMENT DU TERRITOIRE (OAT) .. 2

Chapitre1 : PRESENTATION GENERALE 3

1.1. Présentation de l'Observatoire de l'Aménagement du Territoire 3

1.1.1. Définition générale : 3

1.1.2. Objectifs et missions de l'OAT 3

1.2. Présentation de la mission de conception et d'opérationnalisation du SIL 4

1.2.1. Contexte et justification 4

1.2.2. Résultats attendus 5

1.2.3. Méthodologie de la consultation 5

Chapitre2 : RECUEIL ET ANALYSE DE L'EXISTANT 7

2.1. La situation des données à Madagascar : 7

2.1.1. Incohérence d'action de développement: 7

2.1.2. Hétérogénéité des données : 7

2.1.3. Insuffisance des données : 7

2.1.4. Qualité des données : 7

2.1.5. Dispersion des informations : 7

2.1.6. Rétention d'informations : 8

2.1.7. Identification des sources d'information et des informations existantes : 8

2.2. L'OAT : une nécessité pour Madagascar 8

2.2.1. Une demande sans cesse croissante d'information géographique : 8

2.2.2. Économie d'échelle sur la production d'information géographique : 9

2.2.3. Nécessité de disposer d'un référentiel géographique unique : 9

2.2.4. Nécessité de disposer d'une nomenclature unique pour l'Occupation du Sol : 9

Chapitre3 : RATTACHEMENT INSTITUTIONNEL ET ORGANISATIONNEL DE L'OAT 11

3.1. Forme juridique de l'OAT 11

3.1.1. Les spécificités des scénarii : 11

3.1.2. Le meilleur scénario pour l'OAT : 12

3.1.3. La structure institutionnelle au démarrage de l'OAT : 13

3.2. STRUCTURE ORGANISATIONNELLE DE L'OAT 14

3.2.1. Les acteurs principaux de l'OAT : 14

3.2.2. Les partenaires de l'OAT : 15

3.2.3. Structure fonctionnelle: 17

PARTIE II : CONCEPTION D'UN PROTOTYPE DU SYSTEME D'INFORMATION LOCALISEE(SIL) 18
Chapitre4 : LE SYSTEME D'INFORMATION LOCALISEE (SIL) 19
4.1. Présentation du Système d'Information Localisée : 19
4.1.1. Définition du SIL : 19
4.1.2. Les objectifs du SIL : 19
4.2. Les composantes du SIL : 19
4.2.1. La Base de données nationale : 19
4.2.2. Le Tableau de Bord National : 20
Chapitre5 : CONCEPTION DE LA BASE DE DONNEES NATIONALE 21
5.1. Définitions : 21
5.1.1. La base de données : 21
5.1.2. La base de Données Nationale : 21
5.2. Méthode utilisée : 21
5.2.1. Les méthodes existantes : 21
5.2.2. Choix de la méthode utilisé : 21
5.2.3. Mise en œuvre de la méthode choisie : 22
5.3. Implémentation de la base de données dans le logiciel PostgreSQL : 32
5.3.1. Présentation de PostgreSQL/PostGIS : 32
5.3.2. Modélisation de la BDN dans PostgreSQL : 33
Chapitre6 : CONCEPTION DU TABLEAU DE BORD NATIONAL (TBN) 41
6.1. Présentation du Tableau de Bord National : 41
6.1.1. Définitions : 41
6.1.2. Objectifs : 41
6.1.3. Les produits du Tableau de Bord National : 41
6.2. Identification des indicateurs pertinents : 44
6.2.1. Définition d'un indicateur : 44
6.2.2. Choix des indicateurs : 44
6.2.3. Calculs des indicateurs : 50
PARTIE III : MISE EN ŒUVRE DU SYSTEME D'INFORMATION LOCALISEE (SIL) 52
Chapitre7 : IMPLEMENTATION DU SIL 53
7.1. Architecture informatique du SIL : 53
7.1.1. La topologie du système : 53
7.1.2. Présentation de l'architecture informatique du SIL : 54
7.1.3. Alternative d'implémentation du SIL : 55

7.1.4. Architecture matériel et logiciel du SIL : 57
7.2. Mise en œuvre de la solution adoptée :..... 58
7.2.1. Description de l'architecture multi-tiers : 58
7.2.2. Fonctionnement de l'architecture 3-tiers : 60
7.2.3. Etablissement de la connexion sur tous les niveaux : 65
7.2.4. Génération des cartes statiques ou dynamiques sur internet : 71
Chapitre8 : Simulation du Système d'Information Localisée (SIL) 76
8.1. Conception des interfaces hommes machines (IHM) : 76
8.1.1. Conception de l'interface en ligne du SIL : 76
8.1.2. Conception de l'interface autonome du SIL : 82
8.2. Réalisation avec quelques indicateurs de l'OAT :..... 88
8.2.1. Taux de couverture du territoire en titre privé : 89
8.2.2. Le niveau de contrôle d'ouvrage d'art : 105
8.2.3. Etude budgétaire du projet : 106
CONCLUSION..... 109
BIBLIOGRAPHIE..... 110
ANNEXES..... 114
TABLE DE MATIERES XXVII

Nom: RAJAONARISON
Prénoms : Eli Jean Olivier
Titre du mémoire: Conception et opérationnalisation d'un système d'information localisée de l'Observatoire de l'Aménagement du Territoire.
Nombre de pages : 109
Nombre de figures : 65
Nombre de tableaux : 7

RESUME :

La maîtrise de l'information territoriale est un bon investissement dans tout processus de suivi et d'évaluation de la planification et de la gestion de l'aménagement du territoire. La mise en place d'un observatoire du territoire favorise cette action. Dans ce cas, la VPDAT voudrait opérationnaliser la DOAT mise en place en février 2012 en lui dotant d'un système d'information bien adapté au cas de Madagascar. Le SIL est alors proposé afin de rassembler au sein d'une Base de données nationale toutes les informations territoriales disponibles, de permettre la mise en réseau de tous les acteurs de l'aménagement du territoire et les observatoires existants et enfin de diffuser les indicateurs du TBN de l'OAT pour orienter les décideurs dans leur décision. Malgré les difficultés qu'on devrait affronter au cours de la réalisation d'un tel projet, on arrive à simuler le fonctionnement du système avec un indicateur très pertinent.

ABSTRACT:

The control of territorial information is a good investment in any follow up process, the planning evaluation and the management of the regional planning. The implementation of a territory observatory is improving this action. In this case, the VPDAT would like to operationalize the DOAT set up in February 2012 by offering an adapted system of information to the case of Madagascar. Thus the SIL is proposed in order to gather in a national database all the available territorial information, and to allow the networking of all the actors of the regional planning and the existing observatories and to diffuse the TBN indicators of the OAT at the end in order to direct the leaders in their decision. In spite of the difficulties that we had to face during the realization of this project, we can manage the simulation of the operation of the system with a very pertinent indicator.

Rapporteurs: Monsieur RAJAONARISON Jean Désiré, Enseignant à l'ESPA

ANDRIAMITANTSOA Tolojanahary, Maître de conférences au sein du département Géographie de l'Université d'Antananarivo

Adresse de l'auteur : Lot D 53 bis Tsarafasina Moramanga (514)

Contacts : 0341741362/0337351850**/reliolivier@yahoo.fr**