## Installation PostgreSQL

- 1. extraire les fichiers de postgresql-8.2.3-1.zip dans un répertoire temporaire
- 2. lancer postgresql-8.2.msi et choisissez le Français
- 3. sélectionner le support des langues
- 4. configuration du service : noter le nom du compte de service « postgres » et adjoignez lui un mot de passe à ne pas oublier (utilisateur qui lance le service) !
- 5. notez le numéro de port 5432 ; acceptez les connexions externes (pas seulement localhost; il faudra éditer le fichier pg hba.conf); choisissez une locale France, french ; codage LATIN1 ; superutilisateur : c'est le DBA : il peut être le même que celui qui lance le service...
- 6. installer lo (large objects), autoinc.
- 7. terminer l'installation

#### Démarrage

- -Pour lancer le service, menu Démarrer, PostgreSQL 8.2, lancer le service.
- Lancer pgAdminIII, repérez la bd postgres, créer une nouvelle base de données « test », dans le schéma public, ajouter une table etudiant contenant les colonnes suivantes, ...

#### **CREATE TABLE etudiant**

```
(
 numet serial NOT NULL,
 nom text,
 prenom text
ALTER TABLE etudiant
 ADD CONSTRAINT etudiant_pkey PRIMARY KEY(numet);
```

- Lancer psql sur postgres :
  - select user; -> postgres •
  - select now();
  - select current\_database(); -> postgres •
  - \c test
  - insert into etudiant(nom,prenom) values ('dupont','pierre'),('martin','jacques'); •
- Afficher la table etudiant à l'aide pgAdminIII, notez les nouvelles lignes
- Remarquez les problèmes d'affichage dans la fenêtre dos psql !
- $\$  cmd.exe /c chcp 1252 -> change le code de page de la fenêtre dos
- -Changer la propriété de police de la fenêtre psql afin d'utiliser Lucida
- -Lancer \? et vérifier le bon affichage des accents !

#### **Divers**

- Pour ajouter des données en mode graphique, dans pgAdminIII, afficher les données d'une table, et insérer de nouvelles lignes en bas de la table.
- Pour ajouter de nombreuses lignes à une table depuis un fichier ou depuis l'entrée standard (stdin), utiliser COPY etudiant FROM fic.txt;
- Pour sauvegarder la bd test dans pgAdmin, cliquer droit sur la bd, sauvegarder, donner un nom de fichier « testbackup.sql », PLAIN, commandes insert ;

### Connexion psql à un serveur distant

Sur le serveur :

- 1. Tout d'abord, il faut lancer l'application pare-feu afin d'ouvrir le port TCP 5432 ;
- 2. Il faut ensuite configurer le serveur PostgreSQL à l'aide de 2 fichiers de configuration situés dans « c:/Program Files/PostgreSQL/8.2/data » :
- 3. vérifier dans postgresql.conf qu'il existe une ligne : listen\_addresses = '\*'
- 4. ajouter la ligne suivante dans pg\_hba.conf :

# TYPE DATABASE USER CIDR-ADDRESS METHOD

host all all 192.168.0.0/24 password

5. panneau de config, outils d'administration, services, redémarrer le service postgreSQL pour prendre en compte les changements de configuration ;

Sur le client :

1. psql -h machineServeur -U utilisateur bd

soit psql –h 192.168.0.2 –U postgres test

192.168.0.2 peut être remplacé par le nom de la machine serveur

# Installation et utilisation de PostgreSQL ODBC

- 1. extraire les fichiers de psqlodbc-08\_02\_0400.zip dans un répertoire temporaire puis lancer l'installation du driver psqlODBC et de la documentation sur le client.
- une fois le driver odbc installé sur la machine cliente, il reste à créer un « Data Source Name » associé à une bd particulière : panneau de config, outils d'administration, source de données (ODBC), onglet source de données système pour permettre l'accès à d'autres utilisateurs depuis d'autres machines, ajouter, choisir le driver PostgreSQL Unicode, (le client et le serveur peuvent être la même machine)
  - a. Data Source : choisissez un nom parlant : « connexion test »
  - b. Database : test
  - c. Server : localhost ou le nom du serveur distant 192.168.0.2
  - d. Username : postgres ou un autre utilisateur
  - e. Password : donnez le password associé à l'utilsateur
- 3. Toujours tester avant de sauvegarder.
- 4. Une fois la source de données système installée, on peut interagir sur la bd test depuis une autre application : programme C, autre sgbd comme access, ...
- 5. ouvrir Access, créer une bd, une table, y mettre quelques lignes, puis dans l'onglet table d'Access :
  - a. cliquer droit sur la table, exporter, choisir le format ODBC, choisir un nom de table distante, sélectionner la source de données « connexion test » et lancer la création ;
  - b. vérifier sur le serveur que la table a bien été créée.

Remarquons que depuis la machine cliente, on peut utiliser une source de données système créée localement et associée à un serveur distant, ou bien au moment de sélectionner la source, on peut aller chercher une source de données fichier située à distance sur le réseau !

#### Installation de Wamp

Wamp est un paquetage contenant Apache (serveur http), MySQL (serveur de BD), et PHP (langage côté serveur http). Pour l'installer, cliquer sur wamp5\_1.7.0.exe, et laissez-vous guider par l'installation. Ne cliquez pas sur auto-start sauf si vous voulez lancez le serveur apache automatiquement à chaque ouverture de windows. Laissez la racine de l'arbre web sur

c:\wamp\www, indiquez votre serveur de mail sortant ainsi que votre adresse email. Débloquer Apache depuis votre pare-feu (port 80).

Une fois l'installation terminée, remarquez l'icône et le menu wamp dans la zone de notification : vous pouvez arrêter ou relancer les serveurs Apache ou Mysql et réaliser bien d'autres tâches depuis ce menu :

- 1. cliquez sur localhost : vous ouvrez votre navigateur favori à la racine du serveur Apache <u>http://localhost/</u>, c'est-à-dire sur le fichier **c:/wamp/www/index.php**.
- 2. si vous cliquez sur le lien phpinfo, vous allez voir les options de configuration et d'installation de PHP ;
- 3. si vous cliquez dans le menu wamp sur phpMyAdmin, vous tombez sur la page <u>http://localhost/phpmyadmin/</u> qui vous permet de créer une base de données MySQL et de l'administrer :
  - a. créer une BD essai, y ajouter une table etudiant à 3 colonnes :

) ENGINE = innodb;

- b. insérer quelques étudiants à cette table ; dupont paul , durand jacques, ...
- c. afficher les étudiants ;
- 4. Depuis le menu wamp, lancer la console MySQL, et connectez vous sans mot de passe
- 5. show databases ;
- 6. use essai;
- 7. show tables ;
- 8. select prenom from etudiant;

## Application web liée à une BD Mysql

Nous allons maintenant créer un projet simplissime consistant à afficher sur une page Web les prénoms des étudiants !

- dans le menu wamp, cliquez sur www directory, puis créez un nouveau dossier projet\_etudiant « C:\wamp\www\projet\_etudiant »;
- 2. copier le contenu suivant dans le fichier index.php :

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>>Liste des prénoms des étudiants</title>
</head>
<body>
<h1>Liste des prénoms des étudiants</h1>
<?php
 $urlsgbd='localhost';
 $monnom='root';
 $monpasswd=";
 $mabd="essai";
 $co=mysql_connect($urlsgbd , $monnom, $monpasswd) or
 die('Connexion impossible');
 mysql_select_db($mabd,$co) or die("Sélection de la bd $mabd impossible");
 $req="SELECT prenom FROM etudiant;";
 $res = mysql_query($req, $co) or die("Requête $req impossible");
```

```
echo "\n",' ',"\n"; // début du tableau HTML
while ($ligne=mysql_fetch_array($res)){ // tq il reste des étud
echo ''; // nouvelle ligne
echo "",$ligne['prenom'],"
```

```
3. Allez voir la page http://localhost/projet_etudiant/
```

#### Application web liée à une BD PostgreSQL

Nous allons maintenant créer un projet simplissime consistant à afficher sur une page Web les prénoms des étudiants depuis Postgresql. Pour cela, il faut changer la configuration de php de wamp qui par défaut ne charge pas l'extension postgresql.

- 1. dans le menu wamp, config files, php.ini : éditer le fichier en décommentant la ligne ;extension=php\_pgsql.dll (supprimer le ;)
- 2. redémarrer le serveur apache : dans le menu wamp, apache, restart service ;
- 3. puis créez un nouveau dossier projet\_etudiant « C:\wamp\www\projet\_pg\_etudiant »;
- 4. copier le contenu suivant dans le fichier index.php :

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
<title>>Liste des prénoms des étudiants</title>
</head>
<body>
<h1>Liste des prénoms des étudiants</h1>
<?php
 $urlsgbd='localhost';
 $monnom='meynard';
 $monpasswd='michel';
 $mabd="test";
 $co=pg_connect("host=$urlsgbd user=$monnom password=$monpasswd dbname=$mabd")
 or die('Connexion impossible à la bd $mabd impossible');
 $req="SELECT prenom FROM etudiant;";
 $res = pg_query($co, $req) or die("Requête $req impossible");
 echo "\n", ' ', "\n"; // début du tableau HTML
 while ($ligne=pg_fetch_array($res)){ // tq il reste des étud
  echo '';
                                    // nouvelle ligne
  echo "",$ligne['prenom'],"\n"; //fin de ligne
 }
 echo "\n\n"; // fin du tableau HTML
?>
</body>
</html>
```

 Noter la syntaxe très différente du pg\_connect par rapport au mysql\_connect et l'inversion des paramètres dans pg\_query ! Allez voir la page <u>http://localhost/projet\_pg\_etudiant/</u>