Setting up AircastDB with PostgreSQL

This article explains how to use PostgreSQL as a backend storage for Aircast.

PostgreSQL is an open source database management system, available for Windows, Linux and other operating systems. Being free of charge, fast and networking-enabled, we recommend to use PostgreSQL as the backend for any networked AircastDB installation (where two or more Aircast computers or processes will share the same database).

PostgreSQL server setup

Windows

Binary packages/installers for Windows are available here: http://www.postgresql.org/download/windows/ (for the actual download, you will be taken to a site called EnterpriseDB)

Aircast needs at least PostgreSQL 8.3 but will work fine with newer versions. You should pick the latest stable version available. As of October 2014, this is PostgreSQL 9.3.5.

The Windows installer includes the graphical administration frontend, **pgAdmin III**.

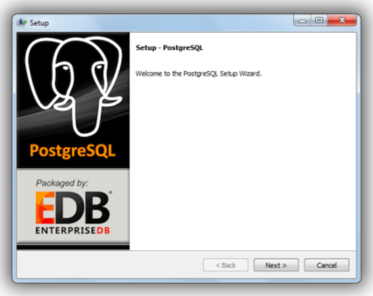
Installation

Download the appropriate setup file for your system (in this example `postgresql-9.3.5-1-windows-x64.exe`) and run it.

The installer will ask you for a couple of things - most can be left untouched:

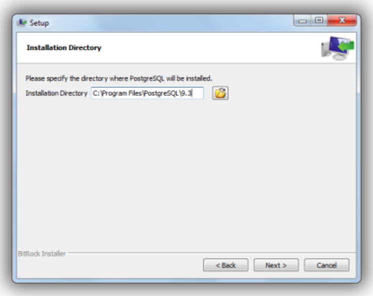
- Installation directory: Where the program files go - leave untouched.
- Data directory: This is where PostgreSQL will store all SQL data. Usually inside the program directory, but it is possible to use a different location, e.g. a larger partition or hard drive.
- Password: This is the password for the superuser `postgres`. Choose a password and write it down. You will need it later to access the database with pgAdmin III.
- Port number: use default 5432
- Locale: leave untouched
- Stack Builder: We don't need that, uncheck this option.

At the end of the installation, you will find pgAdmin III in the Windows Start Menu.



Setup - PostgreSQL

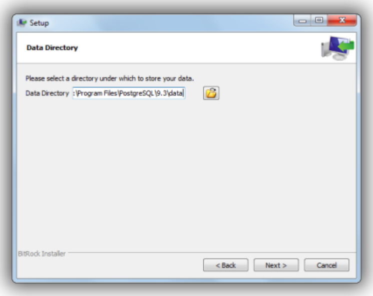
Welcome to the PostgreSQL Setup Wizard.



Installation Directory

Please specify the directory where PostgreSQL will be installed.

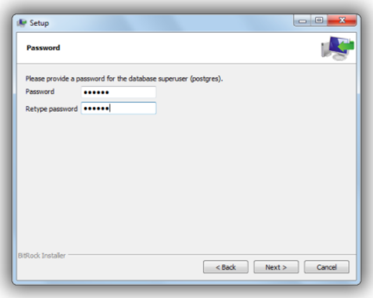
Installation Directory C:\Program Files\PostgreSQL\9.3



Data Directory

Please select a directory under which to store your data.

Data Directory :\Program Files\PostgreSQL\9.3\data

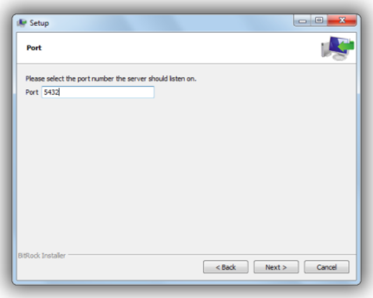


Password

Please provide a password for the database superuser (postgres).

Password ••••••

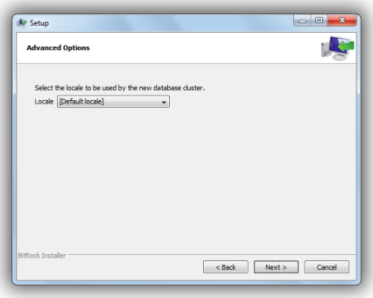
Retype password ••••••



Port

Please select the port number the server should listen on.

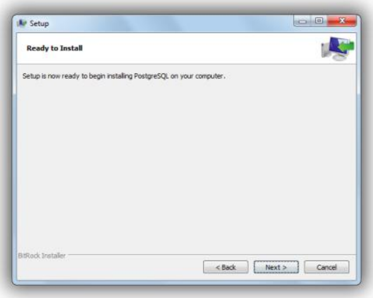
Port 5432



Advanced Options

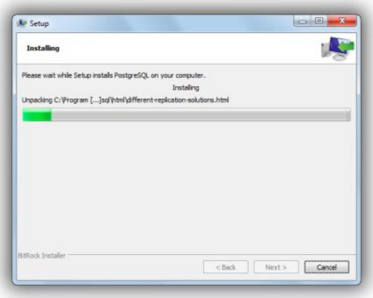
Select the locale to be used by the new database cluster.

Locale [Default locale]



Ready to Install

Setup is now ready to begin installing PostgreSQL on your computer.

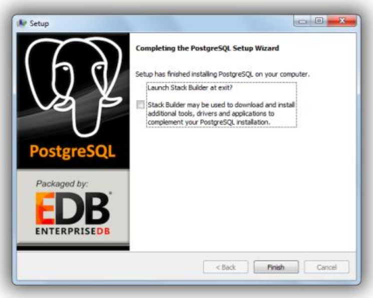


Installing

Please wait while Setup installs PostgreSQL on your computer.

Installing

Unpacking C:\Program [...]sql\html\different-replication-solutions.html



Completing the PostgreSQL Setup Wizard

Setup has finished installing PostgreSQL on your computer.

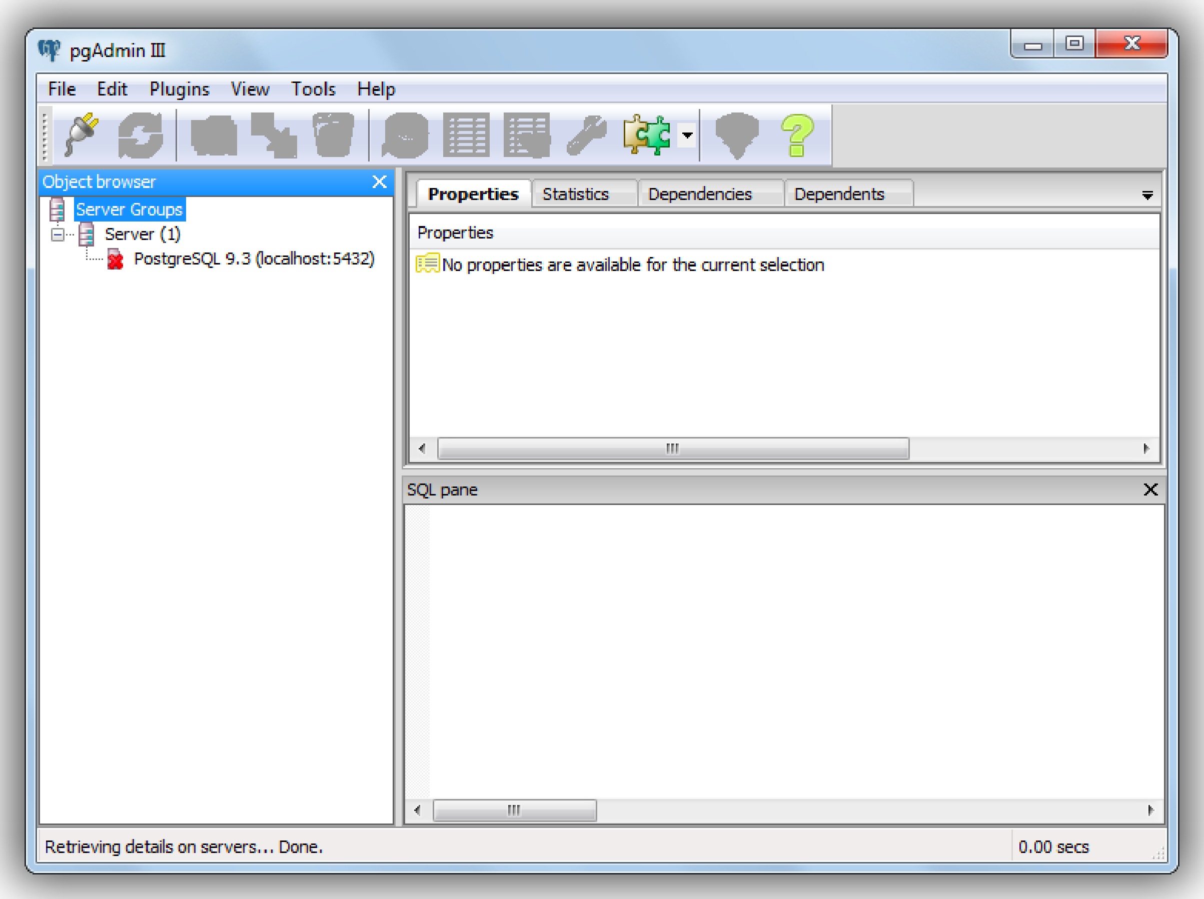
Launch Stack Builder at exit?

□ Stack Builder may be used to download and install additional tools, drivers and applications to complement your PostgreSQL installation.

Create user and database

The next step is to create an SQL user and an empty database for Aircast.

Open pgAdmin III, the main window will show:


pgAdmin III
File Edit Plugins View Tools Help
Object browser
Properties Statistics Dependencies Dependents
SQL pane

Server Groups
Server (1)
PostgreSQL 9.3 (localhost:5432)

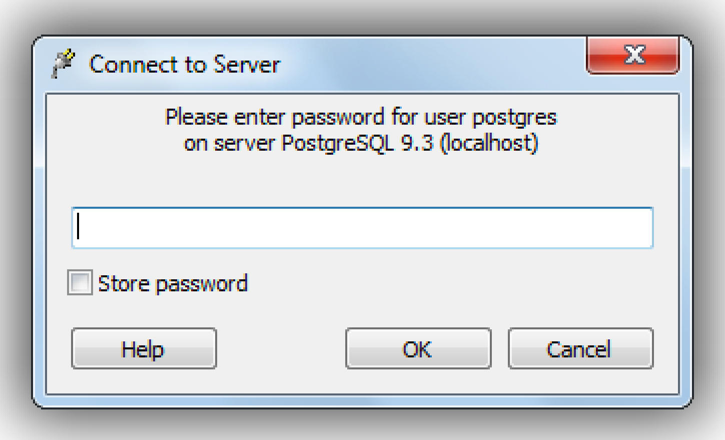
Properties

No properties are available for the current selection

Retrieving details on servers... Done.

0.00 secs

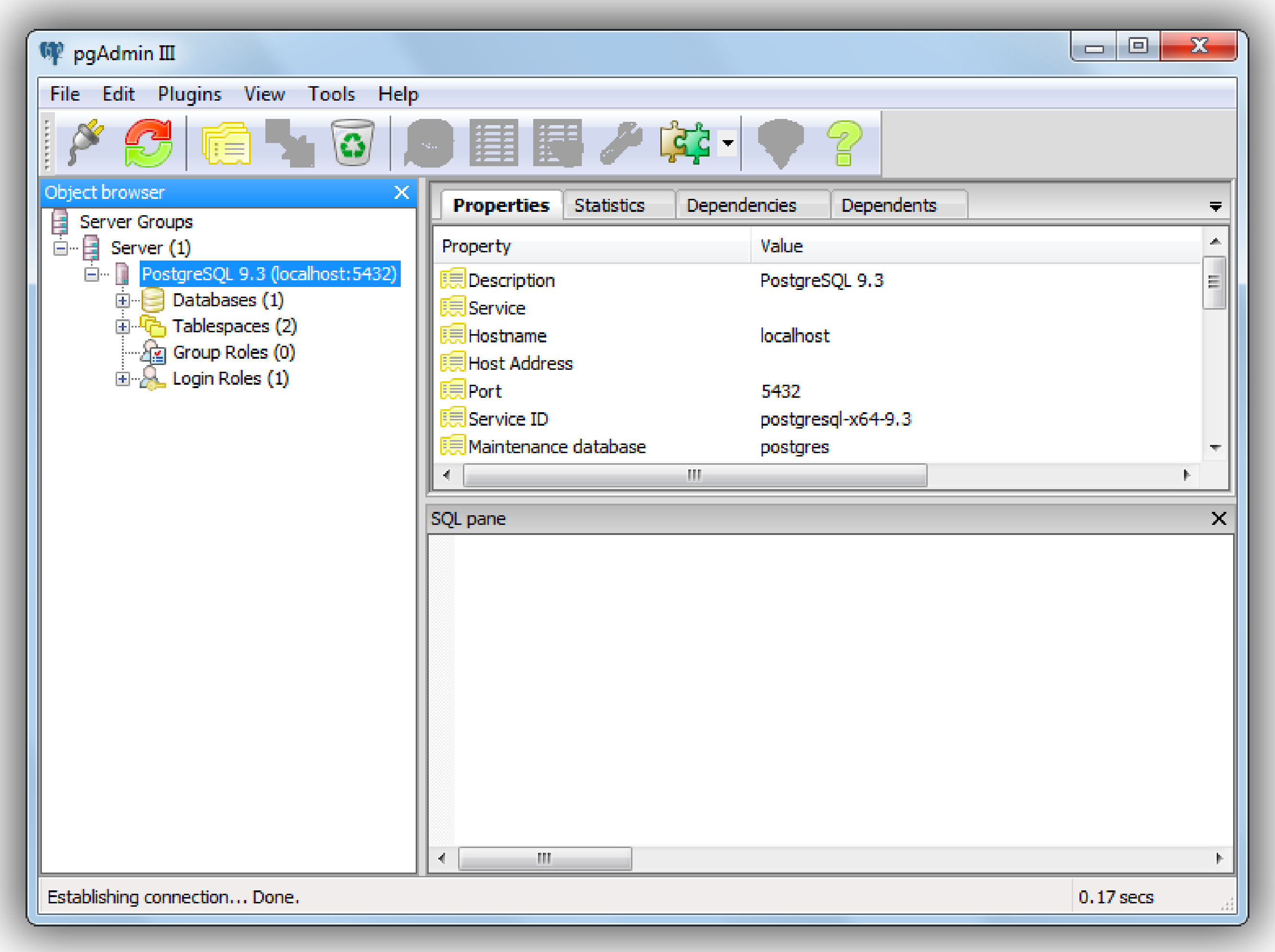
In the tree on the left, you see the “PostgreSQL 9.3 (localhost:5432)” node - this is the server we have just installed. The red x icon means that pgAdmin is not connected to the server yet. Double-click that node, and pgAdmin will prompt for the password of the `postgres` user that you have set during installation:



Please enter password for user postgres
on server PostgreSQL 9.3 (localhost)

□ Store password

Enter the password and click OK. The database tree will expand and show a couple of new nodes: Databases, Tablespaces, Group Roles, Login Roles



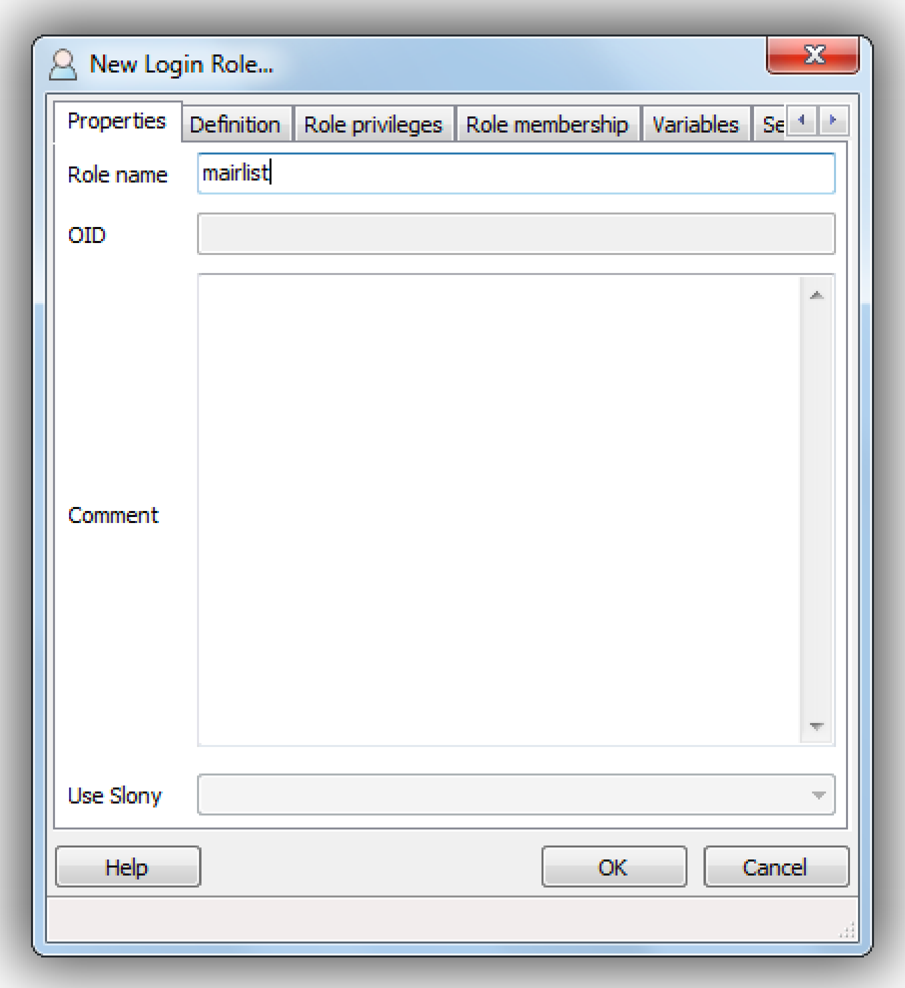
Property	Value
Description	PostgreSQL 9.3
Service	
Hostname	localhost
Host Address	
Port	5432
Service ID	postgresql-x64-9.3
Maintenance database	postgres

Establishing connection... Done.

0.17 secs

First, we have to create a new user (“login role”) under which Aircast will access the database server. Right-click the *Login Role* node, and select *New Login Role*.

In the dialog that appears, first enter the desired name of the role (e.g. *Aircast* - you should use all lower case letters).



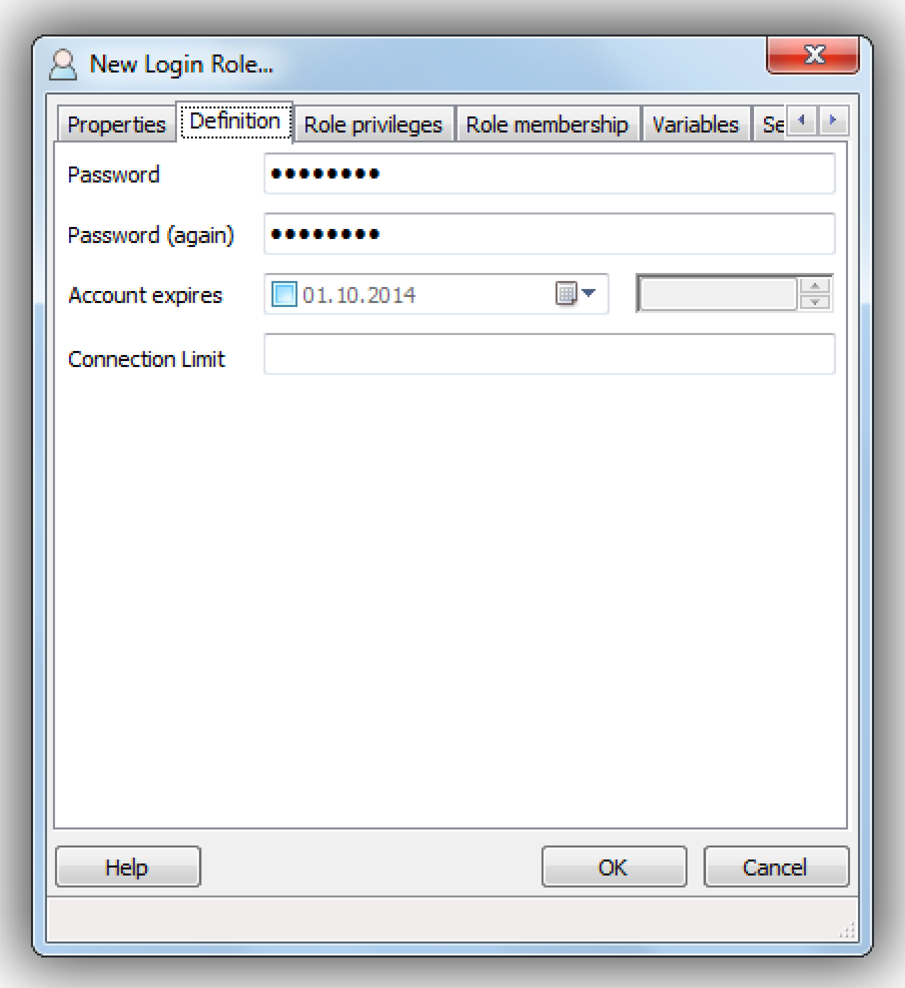
Role name: mairlist

OID

Comment

Use Slony

Then switch to the *Definition* tab and set a password for the user.



New Login Role...

Password ●●●●●●●●●

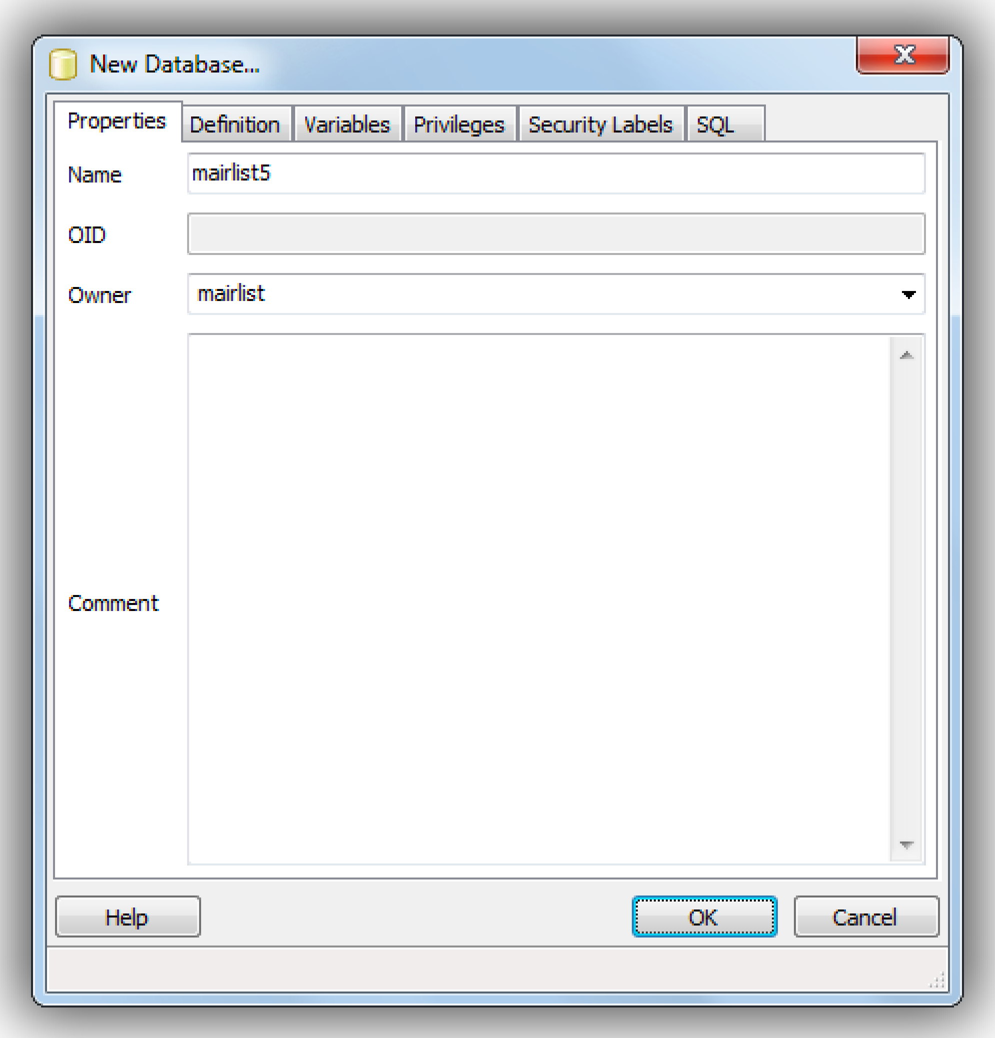
Password (again) ●●●●●●●●●

Account expires □ 01.10.2014

Connection Limit

Click OK, and the login role will be created.

Next, we create an empty database in which Aircast will store its contents later. Right-click the *Databases* node, then select *New Database*. In the dialog that appears, enter a name for the new database, e.g. `Aircast5` (again all lower case!). Don't forget to set the owner to the `Aircast` user we have just created!



Name mairlist5

OID

Owner mairlist

Comment

Leave all other fields untouched. In particular, the encoding of the database (on the *Definition* tab) must be set to UTF-8. Click OK to create the new database.

Enable network access

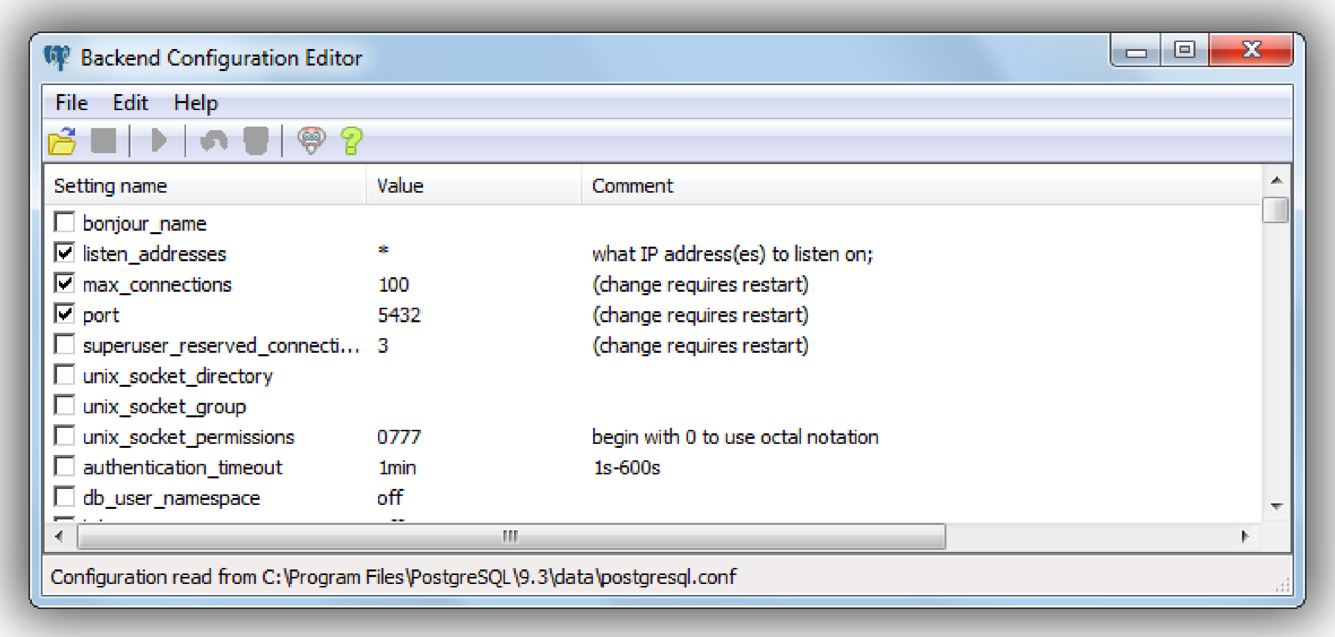
Out of the box, PostgreSQL does not allow any connections from other computers in your LAN. Before we can access the server from another computer, we must edit to configuration files – `postgresql.conf` and `pg_hba.conf` – and then restart the PostgreSQL service.

These configuration files can be found in the PostgreSQL data directory you chose during setup - by default this is `C:\Program Files\PostgreSQL\9.3\data`.

You can open the files with any text editor, or use the editing functions built into pgAdmin III. We will demostrate the latter option here.

Edit postgresql.conf

In pgAdmin, go to *File* → *Open postgresql.conf*, and locate the `postgresql.conf` file in the PostgreSQL data directory. The following dialog will appear:



Setting name	Value	Comment
□ bonjour_name		
☑ listen_addresses	*	what IP address(es) to listen on;
☑ max_connections	100	(change requires restart)
☑ port	5432	(change requires restart)
□ superuser_reserved_connecti...	3	(change requires restart)
□ unix_socket_directory		
□ unix_socket_group		
□ unix_socket_permissions	0777	begin with 0 to use octal notation
□ authentication_timeout	1min	1s-600s
□ db_user_namespace	off	

Configuration read from C:\Program Files\PostgreSQL\9.3\data\postgresql.conf

Look for the setting name *listen_addresses*. It must be enabled (check mark set) and set to *.

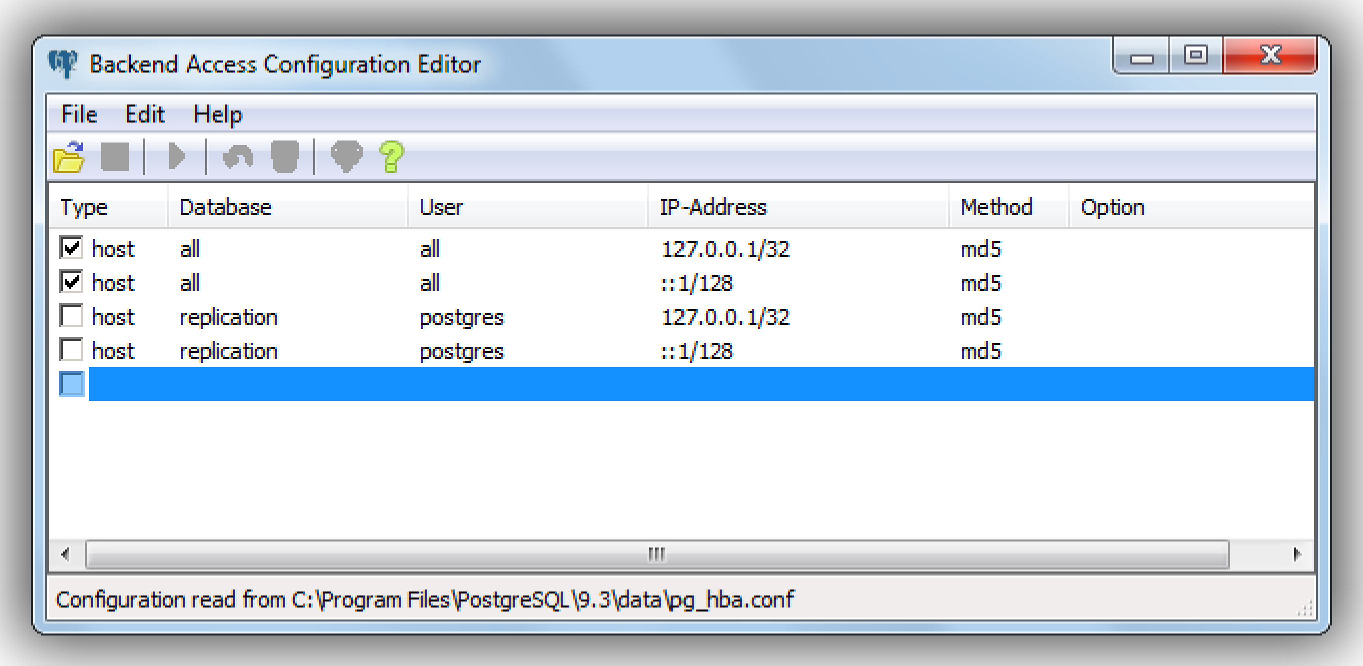
Note: Newer PostgreSQL installations on Windows apparently have this enabled by default. It's a good idea to confirm that the setting is correct though.

Don't forget to click Save if you have made any changes.

Edit pg_hba.conf

The second file, `pg_hba.conf`, defines which databases should be accessible from which hosts and users, and which authentication method should be used.

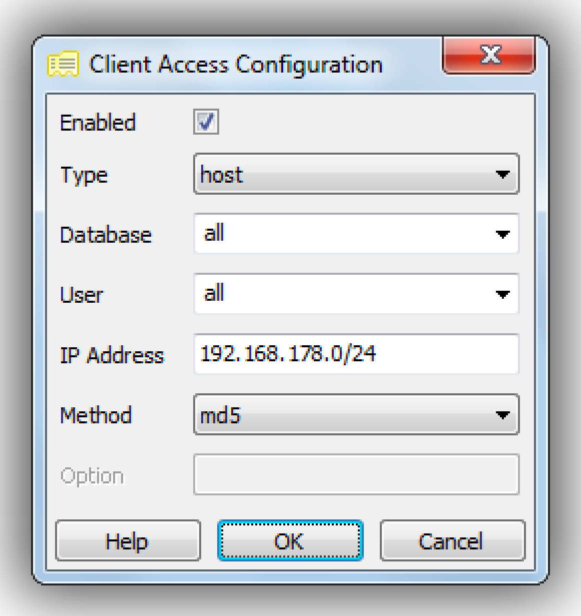
In pgAdmin, click *File* → *Open pg_hba.conf*, and select the `pg_hba.conf` file from the data directory. You will see a window like this:



Type	Database	User	IP-Address	Method	Option
☑ host	all	all	127.0.0.1/32	md5	
☑ host	all	all	::1/128	md5	
□ host	replication	postgres	127.0.0.1/32	md5	
□ host	replication	postgres	::1/128	md5	
□					

Configuration read from C:\Program Files\PostgreSQL\9.3\data\pg_hba.conf

Double-click the empty line (that only has a checkbox) at the bottom of the list, and a dialog will appear where you can add a new entry:



Enabled	☑
Type	host
Database	all
User	all
IP Address	192.168.178.0/24
Method	md5
Option	

In this dialog, enter the following values:

- Enabled: *yes*
- Type: *host*
- Database: *all* (or only *Aircast5* if you want to be very restrictive)

- User: *all* (or only *Aircast*)
- IP address: The address range of your LAN, see comments below
- Method: *md5*

The IP address range must be given in <IP>/<prefix> form. For example, if your network uses the 192.168.178.x addresses with netmask 255.255.255.0 (= prefix /24), enter *192.168.178.0/24*. For more information on subnet prefixes, see here: https://en.wikipedia.org/wiki/Subnetwork

Don't forget to click OK and save the new `pg_hba.conf` file.

Restart PostgreSQL service

If you made any changes to the `postgresql.conf` file, you must now restart the PostgreSQL service. You can either reboot the computer, or go to the Windows Control Panel, services, right-click the PostgreSQL service and restart it.

Changes in the `pg_hba.conf` file do not necessarily require a restart, but you still have to reload the configuration. In the Windows Start Menu, PostgreSQL group, you will find a link *Reload configuration* which does the job.

Adjust Windows Firewall

The final step is to open TCP port 5432 in the firewall so that clients on the LAN can connect to our server.

If you use Windows Firewall (enabled by default on all recent Windows installtions), go to Windows Control Panel, *System and Security → Windows Firewall → Advanced Settings*. The window *Windows Firewall with Advanced Security* will appear.

In that window, right-click the node *Inbound Rules* in the tree on the left, then click *New Rule*. Select *Port* for as the rule type, then, on the next page, *TCP* as the protocol, and *5432* as the local port. Click *Next*, then select *Allow connection*, click *Next*, select all network types, click *Next* again, choose a name (e.g. `PostgreSQL server`), then click *Finish* to save.

Linux

PostgreSQL is available as a package in all major Linux distributions. The graphical frontend pgAdmin III is also available as a package (`pgadmin3`). If you want to use pgAdmin III, install those two packages, and then follow the instructions for creating a user and database, and enabling network access, from the Windows section above.

For the experienced Linux administrator, we briefly summarize the steps required to get PostgreSQL up and running. The following commands are based on Ubuntu 14.04 LTS with PostgreSQL 9.3. We assume that you work as root while issueing these commands.

Install PostgreSQL package:

```
apt-get install postgresql
```

Create a user `Aircast` - you will be promted for a password:

```
sudo -u postgres createuser -P Aircast
```

Create a database `Aircast5` owned by user `Aircast`:

```
sudo -u postgres createdb -O Aircast Aircast5
```

To check if networking is enabled, edit the file `/etc/postgresql/9.3/main/postgresql.conf` and check if it contains the following line:

```
listen_addresses = '*'
```

To grant access to the computers on your LAN (e.g. 192.168.178.x), edit `/etc/postgresql/9.3/main/pg_hba.conf` and add the following line:

```
host    all             all             192.168.178.0/24            md5
```

Then restart the PostgreSQL daemon:

```
service postgresql restart
```

Now everything should be ready.

Client setup

Now that the PostgreSQL server is up and running, we can set up the clients, in other words, configure the database connection in Aircast. The database application is part of the normal setup package, and the connection setup is made in the same configuration app that is also used to configure the playout functions.

First client computer

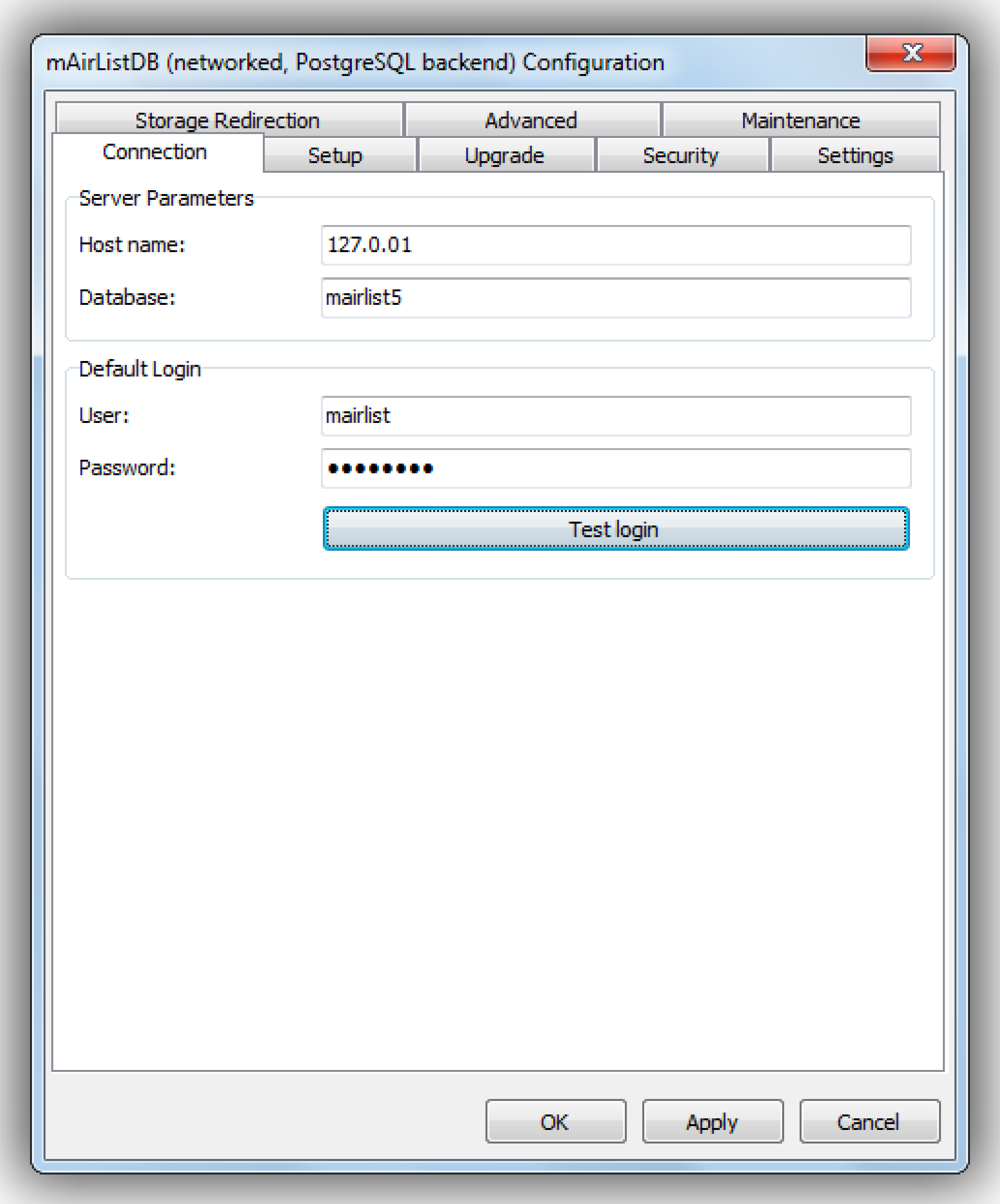
Connection setup

On all client/Aircast computers that you want to access the AircastDB, a database connection must be set up in the config.

Install Aircast, then open *Windows Start Menu → All Programs → Aircast → Configuration*. Go to the *Databases* page, then click *Add* and add a new connection of type *AircastDB (networked, PostgreSQL backend)*.

The AircastDB connection properties dialog will appear:

On the *General* tab, enter the host name or IP of the PostgreSQL server, the database name (*Aircast5* in the example above) and the name and the password of the user that you created during the server setup (*Aircast* in our example):



mAirListDB (networked, PostgreSQL backend) Configuration

Server Parameters

Host name: 127.0.01

Database: mairlist5

Default Login

User: mairlist

Password: ••••••••

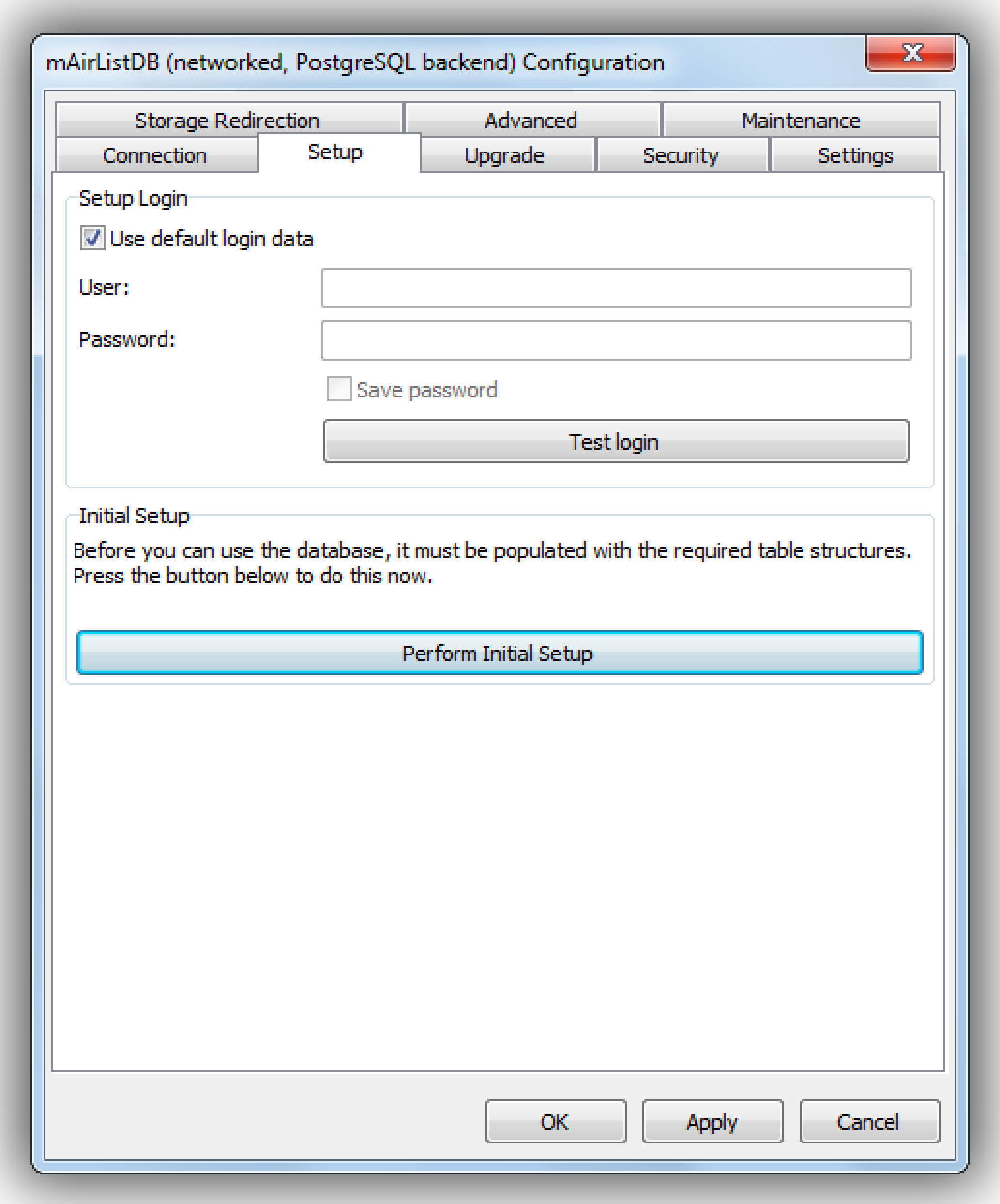
When you click *Test login*, an error message will appear, saying that the “relation config does not exist”. This is OK at this point, as we haven't set up the tables yet (see next section). Any other message, like “connetion timeout”, “missing pg_hba.conf entry” etc. means that something is wrong with the connection config, server or firewall.

Before you close the dialog and repeat the steps on the other client computers, go to the *Setup* tab to perform the initial setup tasks.

Initial table setup

Before Aircast can use the SQL database, you have to create the tables (database schema) in the still-empty SQL database. **This must only be done once, from the first client computer that you set up! Do not repeat this on the other computers!**

Go to the *Setup* tab and click *Perform initial setup*:



☑ Use default login data

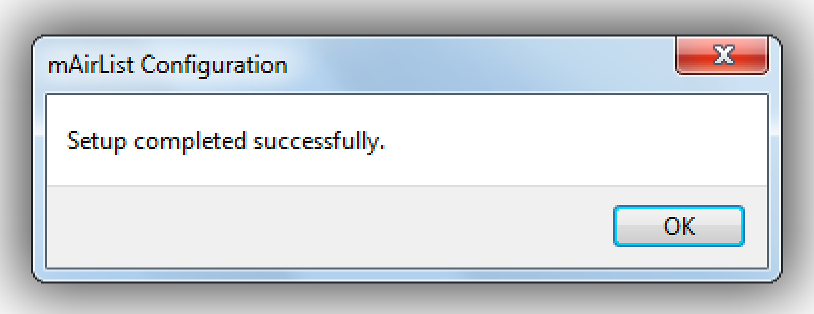
User:

Password:

□ Save password

Before you can use the database, it must be populated with the required table structures. Press the button below to do this now.

Aircast will now create the SQL tables necessary to hold the AircastDB library and scheduling data. This might take a few seconds and will eventually be confirmed with the messsage *Setup completed successfully*:



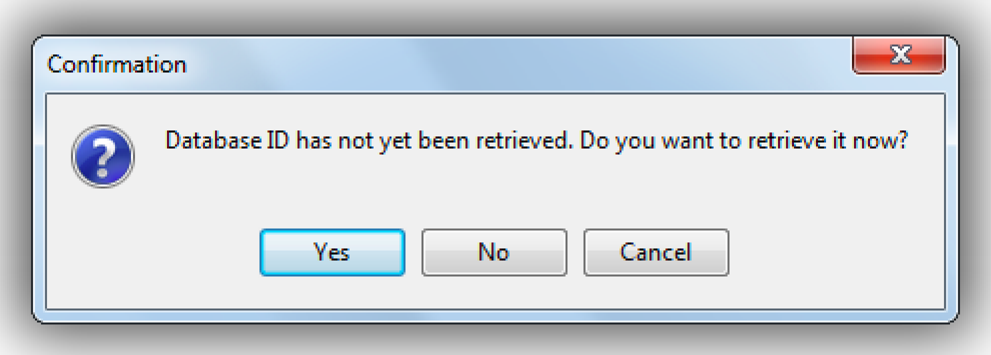
mAirList Configuration

Setup completed successfully.

Additional clients

If you want to connect additional Aircast computers to the database, go to the Aircast configuration app on each computer, and add the connection as described above. Now that we have created the tables, the *Test login* button should give the message: *Login OK*

When you click OK to save the connection, you will see the following message:



Confirmation

Database ID has not yet been retrieved. Do you want to retrieve it now?

Each AircastDB has a unique identifier (made up of several random hexadecimal characters), the "Database ID". The ID is stored in the `config` table inside the SQL server, and also in the `database.ini` file of each client PC. It will help Aircast to identify the correct connection if you happen to maintain connections to more than one AircastDB at a time. Clicking *Yes* in this dialog will retrieve the ID from the SQL table and store it in the local configuration. (You can look it up on the *Advanced* tab in the connection properties dialog if you like.)