

**KEMI-TORNIO UNIVERSITY OF APPLIED SCIENCES
TECHNOLOGY**

Zeng Xi

The Usability of PostgreSQL

The Bachelor's Thesis IT programme
Kemi 2010

PREFACE

I want to thank my supervisor Aalto Teppo who gave me great help and good advice on my thesis modification.

ABSTRACT

Kemi-Tornio University of Applied Sciences, Technology	
Degree Programme	Information Technology
Name	Zeng Xi
Title	Usability of PostgreSQL
Type of Study	Bachelor's Thesis
Date	20 th march, 2011
Pages	29
Instructor	Teppo Aalto
Supervisor	Teppo Aalto

My topic of the thesis is usability of PostgreSQL. It presents basic information about PostgreSQL, such as its history, function, and comparison with one of the most popular databases - MySQL. PostgreSQL has been tested in different environments and with different programming languages. The research is based on the documents from PostgreSQL official website.

The thesis introduces and proves the usability and the compatibility of PostgreSQL. PostgreSQL has been installed on Windows workstation and checked in different programming environments. Those environments which have been chosen are free and open sources, such as Qt ceator, Dev-Cpp, and Eclipse. Whole the tests have been used by different programming languages (C, C++, Java, PHP, Ruby and Python) in the different environments, to check using create, select, insert, and delete commands to control PostgreSQL.

Most of the codes and solutions can be found in documents and FAQ in official websites. Those tests can prove PostgreSQL's compatibility and usability. PostgreSQL is absolutely safe, fast, and technique strong.

Key words: PostgreSQL, programming environment, programming languages, Windows.

TABLE OF CONTENTS

PERFACE.....	I
ABSTRACT.....	II
1. INTRODUCTION.....	1
2. Basic knowledge	
2.1. SQL.....	2
2.2. DBMS.....	2
3. Features of PostgreSQL and comparison with MySQL.....	3
3.1. History of PostgreSQL.....	3
3.2. MySQL	4
3.2.1. History of MySQL.....	5
3.3. PostgreSQL vs MySQL.....	5
3.3.1. Features of PostgreSQL.....	6
3.3.2. Features of MySQL.....	7
4. Install and operate PostgreSQL.....	8
4.1. PostgreSQL installation	8
4.2. Basic operation in PostgreSQL.....	12
5. PostgreSQL in different environment.....	17
5.1. PostgreSQL works with Qt.....	17
5.2. PostgreSQL works with Dev-C++.....	18
5.3. PostgreSQL works with Eclipse.....	19
6. PostgreSQL and web programming languages.....	20
6.1. PHP.....	20
6.2. Python.....	21
6.3. Ruby.....	22
7. Conclusions.....	26
8. Reference.....	28

1. INTRODUCTION

The topic of this thesis was the usability of PostgreSQL. It has been presented PostgreSQL in a three-dimensional way. It started by introducing the basic knowledge of SQL and DBMS, then extended to content, history, development and features of PostgreSQL. For more impressively express the usability of PostgreSQL, it has been compared with one of the most popular databases - MySQL. The comparison has been made concerning cost, license, technique, etc.

There were some tests for checking the usability and compatibility of PostgreSQL. PostgreSQL was installed on Windows XP, and worked in the different programming environments (Qt creator, Dev-C++, Eclipse). It has been checked by using create, select, insert and delete databases commands in those programming environments. It has also been checked by using different programming languages (C, C++, Java, PHP, Python, Ruby) in the different environments. So PostgreSQL has been tested by using different programming languages in the different programming environments and different web programming languages in the same web programming environment.

The thesis displayed the usability of PostgreSQL from both theories and practices. All the tests were very simple and easy to understand, they basically introduced how to connect PostgreSQL in different environments, how to manage and maintain databases with different languages, such as creating new databases, inserting information, deleting databases or tables. The thesis has been displayed with figures to show the installation of PostgreSQL 9.0 and managing PostgreSQL by using pgAdmin III on Windows. The thesis listed main problems when installing and programming softwares.

2. BASIC KNOWLEDGE

2.1. SQL

Before getting started to introduced the usability of PostgreSQL, it is very important to know what SQL is. All the database products are made for managing databases, same does PostgreSQL. To make the work easier, it is necessary understand SQL at beginning.

SQL is short for Structure Query Language, it is a language which supports database query and programming. It is a high-level non-procedural programming language that allows users work on high-level data structure. It neither requires users to specify method of data storage nor requires users to understand details of specification of storage method. This makes it possible that different database systems to use the same language as the SQL data input and management. This feature forces programmers to create better and better database management systems (DBMS)./1/

2.2 DBMS

SQL was developed at IBM by Donald D. Chamberlin and Raymond F. Boyce in the early 1970s. The first Relational Database Management System was RDMS, developed at MIT in the early 1970s, soon followed by Ingres, developed in 1974 at U.C. Berkeley. Ingres implemented a query language known as QUEL, which was later supplanted in the marketplace by SQL. The first commercial database which support SQL language is Oracle. It is one of many DBMS products. DBMS is a large database manipulation and management software, used to create, use, and maintenance database. For database's safety and integrity, it unified database control and management. Users can see data access DBMS and administrators can maintenance database access DBMS. For example in youtube, users can search and watch different kind of videos, after they log in, they can upload videos and update their personal information, as administrators, they can delete those videos and user accounts. Those are usability of DBMS, data storage, data remove, get data and check data etc. /2/

Nowadays, there are many DBMS products in marketing, such as: Oracle, Sybase, Informix, Microsoft SQL Sever, Microsoft Access, PostgreSQL, MySQL and so on. Those products have their own features and use in different cases. In the thesis, it presented the research of the usability of PostgreSQL, and compared PostgreSQL with the most popular one - MySQL.

3. Features of PostgreSQL and comparison with MySQL

On 20.09.2010 when PostgreSQL official website published PostgreSQL 9.0 release, people started to keep eye on it again. Programmers started to ask what will be added and changed in the new version. For beginners, it is necessary to start with PostgreSQL's background, history, development and evaluation. /17/

PostgreSQL is one of the most advanced and the most powerful free database management systems. It is not controlled by any single company, it belongs to a global community of developer and companies develop the system. It is based on ORDBMS(object-relational database management system). As a free open source, PostgreSQL supports most of the SQL standard and offers many other features: such as: complex queries, foreign keys, triggers, views, transactional integrity, multi-version concurrency control. Also, it can be extended in many ways, for example, by adding new data types, functions, operators, aggregate functions, index methods, procedural languages and because of the permit flexible, every user can use any purpose free to charge, modify, and distribute PostgreSQL, no matter for private, commercial or academic use./3/

From the view of technology, PostgreSQL uses classic structure C/S (client/ server) structure, which means a client corresponds to a server-side daemon mode, it queries the request, after data retrieval and formatting, the final output of the results return to the client. In order to facilitate programmers programming, PostgreSQL provides a uniform interface for the client C. All the different client interfaces come from the same interface C, such as ODBC, JDBC, Python, Perl, Tcl, C, C ++, ESQL etc. It points out that PostgreSQL supports almost all types of database client interfaces, and it supports different operating systems, such as Linux, FreeBSD (BSD-derived Unix), MacOS, Solaris (SVR4-derived Unix), and Windows. /3/

3.1 History of PostgreSQL

PostgreSQL started with Ingres, which was a project in the University of California Berkeley in 1982. The project leader, Michael Stonebraker, left Berkeley to make a proprietary version of Ingres. In 1985, he returned to Berkeley and started a new project which named post-Ingres. During the early 1980s, the team had been solved problems about contemporary database system. In the next project - Postgres, it already had the fewest features can completely supported types, which had ability to define types and can fully describe relationships. Postgres used many of the ideas of Ingres, but the totally different code.

From 1986 to 1993, the team had published four different versions. In 1989, the first version had been published, it had a small number of users. After it released the

version 4, it already got great number of users supported and the project ended. It was a big step for the project.

In 1995, Berkeley's graduated students Andrew Yu and Jolly Chen, replaced Postgres' POSTGQUEL query language, with an extended subset of SQL, created Postgres95, and released code in Internet. In 1996, the project renamed to PostgreSQL, reflected it supported SQL. At the same time, it set the version numbers started from 6.0, which backed to original sequence by the Berkeley Postgres project. The first released version PostgreSQL 6.0 had been published in January 1997. Since then, the software has been maintained by a group of database developers and volunteers around the world./4/

3.2. MySQL

MySQL is quite popular and widely used in high profile, large scale World Wide Web products, such as, Wikipedia, Google, Youtube and Facebook. MySQL is also very popular using by small and medium size of web site companies, because of its small, fast, low cost, and especially it's open sources. Those features can reduce companies' total cost of the ownership. MySQL is a popular choice for using in web application, and it is a central component of the LAMP (Linux, Apache, MySQL, PHP/Perl/Python) web application software stack. For commercial use, MySQL has several payable editions for sale./18/

MySQL is written in C and C++, its SQL parser is written in yacc, and a home brewed lexical analyzer named sql_lex.cc. MySQL supports many different system platforms, such as AIX, BSDi, FreeBSD, HP-UX, i5/OS, Linux, Mac OS X, NetBSD, Novell NetWare, OpenSolaris, eComStation, OS/2 Warp, QNX, IRIX, Solaris, Symbian, SunOS, SCO OpenServer, sco UnixWare, Sanos, Tru64 and Microsoft Windows. /5/

3.2.1. History of MySQL

MySQL was developed by Swedish company MySQL AB in 1995. The first version had been released in Internet on 23rd May 1995. On 8th January 1998, it had the version for Windows 95 and Windows NT. The difference between MySQL and PostgreSQL, MySQL always has beta versions before release productions. Version 3.23, beta had been published in June 2000, the formal production had been released in January 2001. Version 4.0, beta had been published in August 2002, the formal production had been released in March 2003. Version 4.01, beta had been published in August 2003, and in this year, Jyoti had adopted MySQL for database tracking. Version 4.1 beta had been published in June 2004, the formal production had been released in October 2004. Version 5.0 beta had been published in March 2005, the formal production had been released in October 2005. On 26th February 2008, Sun Microsystems acquired MySQL AB. On 27th November 2008, version 5.1 had been

released, which had solved 20 crashing problems. On 27th January 2010, Oracle had acquired Sun Microsystems. The version 5.5 is currently generally available, the default storage engine is InnoDB, which supports transaction and referential integrity constraints, SIGNAL and RESIGNAL statement in compliance with the SQL standard.

/6/ /7/

3.3. PostgreSQL vs MySQL

There are plenty of articles can be found in Internet about comparison between PostgreSQL and MySQL. They both are technique strong, open sources database and many supporters. They both have large numbers of users, their supporters have listed their benefits against each other. It is impossible to choose the absolutely winner, but from their features and differences, users can choose the fit one according to the situations.

It is not necessary to choose which you prefer, but after understanding the advantages and disadvantages of each products, it is important to make right choice in the different cases. MySQL has many famous users, such as Facebook, Youtube, Google, etc. But it doesn't matter for programmers to make their own choice, PostgreSQL definitely has a strong technique compare to MySQL. /22/

PostgreSQL has an older history, larger scale and faster growing community. It started 1985, Berkeley leaded the whole project. From now on, there are thousands volunteers and members. The advantage of this large community is that users and developers can closer communicate, users even can work directly involved in the new features' design. This model is imitated by other open sources communities. /23/

MySQL is used by business world, as the high-speed database, MySQL AB has been chosen MySQL as an open source supported by commercial companies. MySQL becomes one of the most popular databases in the short time. Because of that, Sun Microsystems has acquired it with \$ 1,000,000,000. Also the same reason, Oracle has acquired Sun Microsystems on 27th January 2010. /5/

Table 1. Comparison between PostgreSQL and MySQL /24/ /25/

Databases	PostgreSQL	MySQL
Developer	PostgreSQL Global Development Group	MySQL AB (A subsidiary of Oracle)
Stable release	9.0.3/ 8.4.7/ 8.3.14/ 8.2.20	5.5.11 GA/ 5.1.56 GA/ 5.6.2 m5
Written in	C language	C, C++
Operating system	Cross-platform	Cross-platform
Type	ORDBMS	RDBMS

License	PostgreSQL license	GUN General Public License
Website	www.postgresql.org	www.mysql.com dev.mysql.com

Table 1 shows the basic information of PostgreSQL and MySQL, which compares their developer, stable release, written in, operating system, type, and license. They have the same operating system, but they are written in different languages, different licences and different types.

3.3.1. Features of PostgreSQL

Though decades of development, PostgreSQL becomes one of the most advanced open source databases. As full-featured open source relational database management system, PostgreSQL provides many features to support high-transaction, mission-critical applications. The most competitive feature is the security of data protection. It goes through the authentication mechanisms of enterprises. For example, LDAP or Kerberos control the connection to database, all the transaction to the database can take the SSL connection after authenticate, it increases the security of protection.

PostgreSQL enforces users to use user-defined constraints when adding or modifying data, to ensure data quality follows the limit of rules. PostgreSQL has a simple online backup tool - PITR, which means Point-In-Time Recovery mechanism, that administrators can fast recovery or restore from a time in the past. Users can check from simple range or complex foreign key, they can backup and restore data at anytime if they have saved data in the disk./19/

In the the core architecture of PostgreSQL, it allows other community groups through the form of add-on modules to create more advanced features into PostgreSQL. The good example of this is PostgreSQL's geospatial support. This function comes from a module called PostGIS which is a simple extension to PostgreSQL that makes it the strongest spatial, open source or commercial database. Another feature of PostgreSQL, is that has many different types of stored process languages, developers can use the best language for their needs./8/

There is similar between PostgreSQL and MySQL, there are many "roles" in the database, different roles visit different data. There is some limit in the different roles, for example, administrators and users have different rights in a web site. Users use commands like "create role", "alter role", "drop role" to control the database. But the difference is, "roles" in the PostgreSQL can be mapped and bound to system users, which means it can use different forms of system to certify: ident server authentication, LDAP server authentication, PAM and Kerberos. For local connections, users also can use the system permission to determine who can access the Unix domain, and its location. /20/

3.3.2. Features of MySQL

MySQL has the reputation for the most popular open source database, because of its legacy of performance and simplicity. At the beginning, MySQL has been designed to be a fast indexed sequential access method (ISAM) data store for web sites. The type of work, which with many small queries, has been led to features (such as query cache) improved MySQL's performance. This concentration on performance has inspired features such as MySQL Cluster, which allows the database to scale beyond a single physical server. /8/

PostgreSQL is not the only database that allows to add modules which has external extension functions on database. One of the greatest features of MySQL is its pluggable storage engine - MyISAM, which is the default storage engine of MySQL, it provides the performance for read mostly environment, and the InnoDB storage engine provides the transaction robustness necessary for write intensive applications. /8/

PostgreSQL and MySQL have distinct reputations for supporting specific types of applications. They are frequently used by many applications. They support many different programming languages, operating systems, programming environments and be supported by strong teams. MySQL is a strong database for web applications, it also can be used by some applications which requires to handle large range of use. People chooses MySQL because of its simple operation, but operation of PostgreSQL is not hard as well. After understand the operation rules of PostgreSQL, people will find out they are pretty similar. They are both open sources, but PostgreSQL is totally free and it is not belongs to any company or organization, it attracts people using, developing and contributing. /8/

4 . Install and operate PostgreSQL

Previously, it introduced basic information of PostgreSQL and comparison between PostgreSQL and MySQL. In this chapter, it will present installation of PostgreSQL 9.0 on Windows and some basic operations by using PostgreSQL tool - pgAdmin III, such as creating/deleting database/table and inserting values, etc.

4.1 PostgreSQL installation

PostgreSQL supports many different of operating systems, such as, Linux, FreeBSD (BSD-derived Unix), MacOS, Solaris (SVR4-derived Unix) and Windows. It has different installers for those different operating systems. In this chapter, it will list how to install the last version of PostgreSQL in the most used operate system - Windows.

1. Open <http://www.postgresql.org/download/windows>

Windows package has been chosen for installation in the thesis. The fastest way to get up and run in Windows is the one click installer. The Windows installer includes the PostgreSQL server, pgAdmin III, a graphical tool, and StackBuilder. Version 9.0.2-1 Win x86-32 which is the newest version at the moment has been chosen for installing in the thesis. Versions for supporting Windows are PostgreSQL 8.2 and above.

2. Double clicking the file named "postgresql-9.0.2-1-windows.exe" which is just downloaded. Installation is started. It is no different with another softwares, it only needs to read carefully and click "Next" until finish the installation. There is one thing needs to be careful which is choosing directory for PosgreSQL. As in the Figure 1:

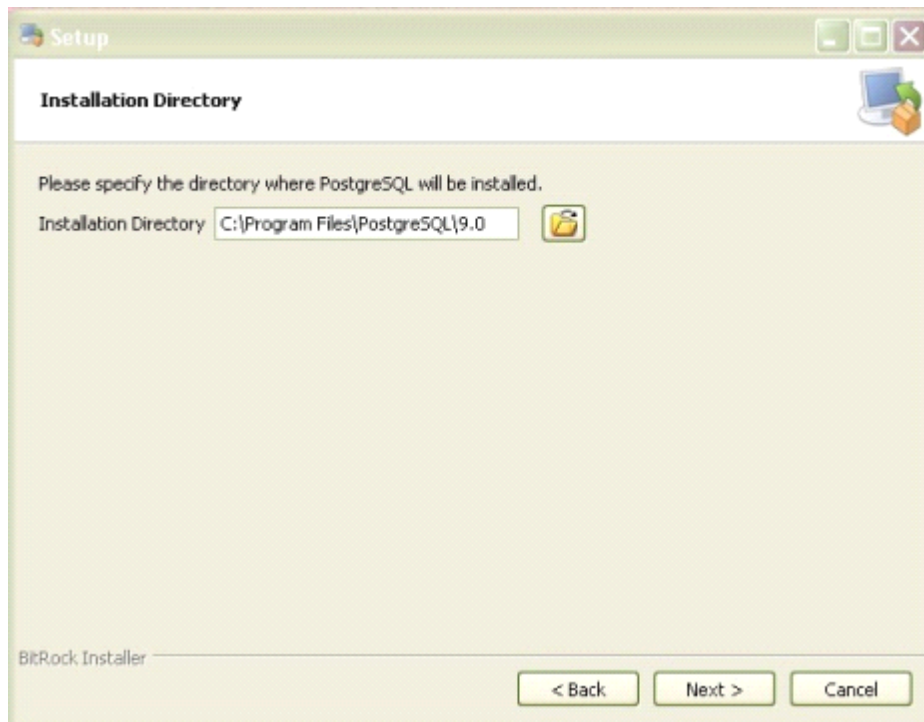


Figure 1. Installation Directory.

The Figure 1 shows specifying PostgreSQL 9.0 directory. The directory for installing PostgreSQL can be in any disks or folders. It is better to install it in a directory which without space between folders' name. For example, it is better to install in "C:\PostgreSQL\9.0" than in "C:\Program Files\PostgreSQL\9.0". Because of "Program Files" has one space between Program and Files. A directory which has spaces in names that may lead to DOS can't recognize the directory, same does in the command prompt. After users choose a right directory and click "Next", then will see a window like the Figure 2:

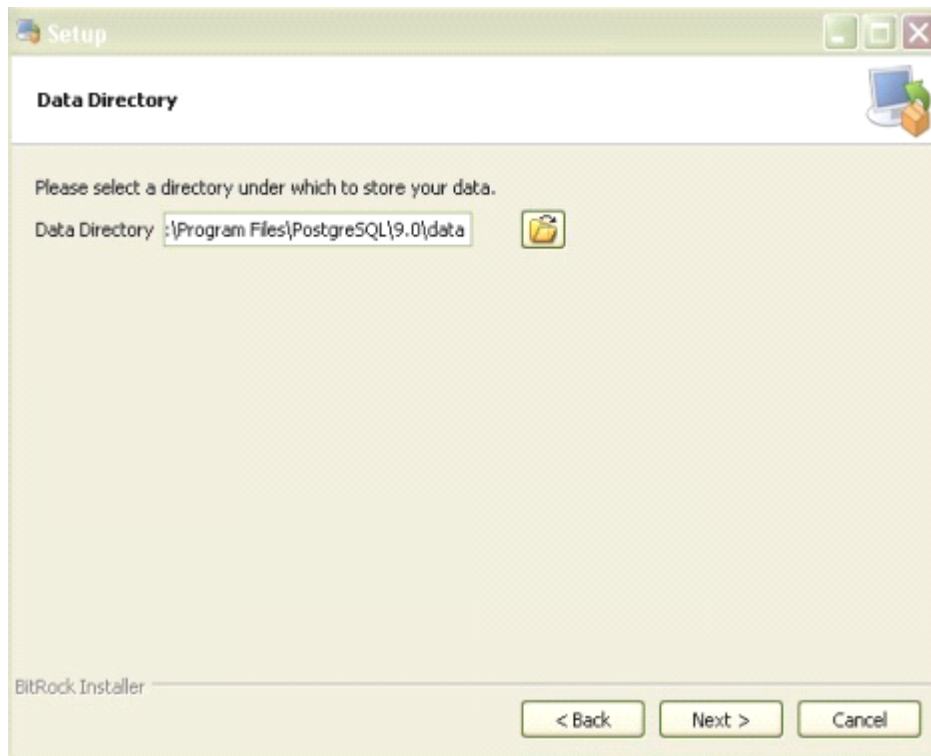


Figure 2. Data Directory.

The Figure 2 shows selecting a directory for data. The directory for storing data should be under the directory of PostgreSQL 9.0. If data directory has been changed to another place from where PostgreSQL 9.0 was originally installed, the system will give warning until user change it back.

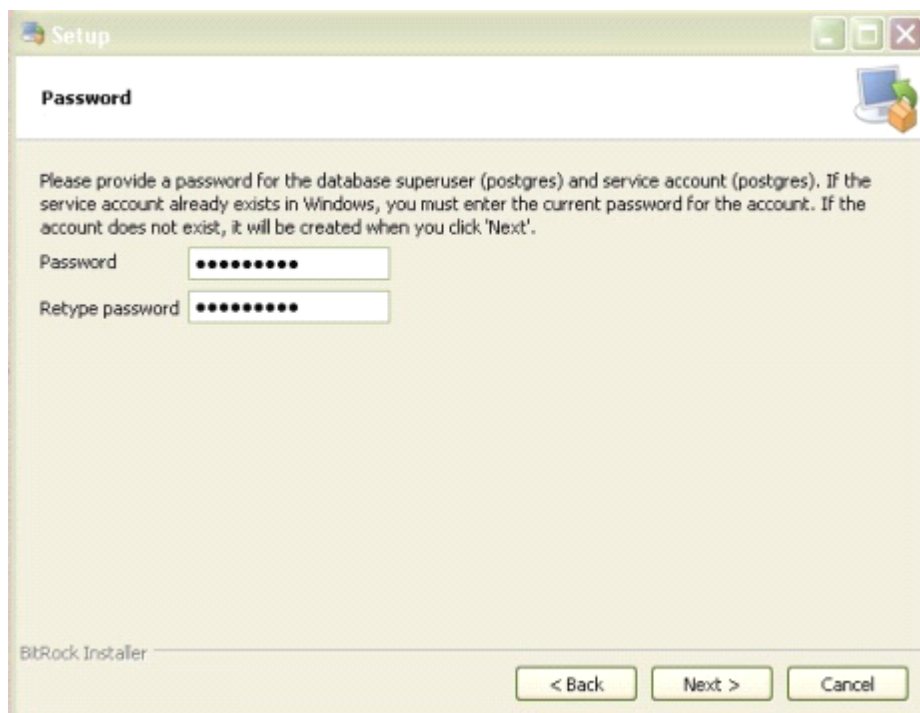


Figure 3. Set password for database.

The Figure 3 shows password window for the database superuser (postgres) and service account (postgres). In this step, if user uses a computer which has been installed PostgreSQL before, even it has been uninstalled. The system will still show an error which is the specific password incorrect. The problem won't solve if user tries to type different passwords. Because of the previous Postgres account still in the computer. It needs few steps to solve the problem. User needs to find "administrative tools" in the "control panel". Opening "computer management", to find "local users and groups", and delete the postgres account. Then user can continue the installation. If user uses a computer which has never installed PostgreSQL before, it won't have the problem. Remember to delete previous account if it is needed to install PostgreSQL again.

3. Setup and be ready to install. In this step, it needs to choose the port and locale for database cluster. The original port is 5432, users can choose any number which is easy to remember. "English, United States" has been chosen as the locale in the thesis. Finishing the setup, it is ready for installing PostgreSQL. It needs to wait a few minutes to complete the installation.

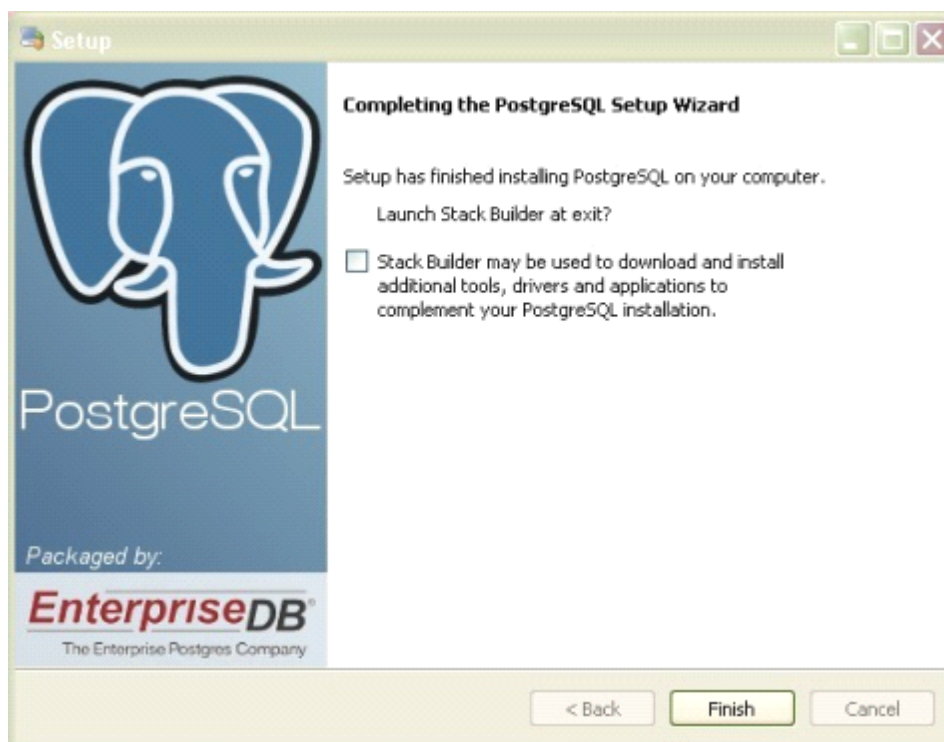


Figure 4. Complete the PostgreSQL setup.

The Figure 4 shows the final setup. It means installation complete, and ask user if she/he wants to run Launch Stack Builder at exit. If user wants install additional tools then just need to click "Finish". It will show a window which can choose additional tools and drivers. Users can download and install add-ones depend on situations. It takes some time to complete downloading and installation. Figure 4 shows the mark has been removed which is asking Launch Stack Builder at exit. Additional tools are not necessary at this moment. After clicking "Finish", the whole installation is completed.

4.2. Basic operation in PostgreSQL

Installation is done. It is the time to check how does PostgreSQL 9.0 work, and is it easy to use? People who has used MySQL before, probably know the tool for maintenance and management of database - MySQLAdmin. PostgreSQL has same kind of tool too - pgAdmin III. It supports different operating systems and it helps users manage the database easily. The tool is with Windows installer. It can be found in Start menu when complete installation. Opening pgAdmin III then can see an interface like the Figure 5:

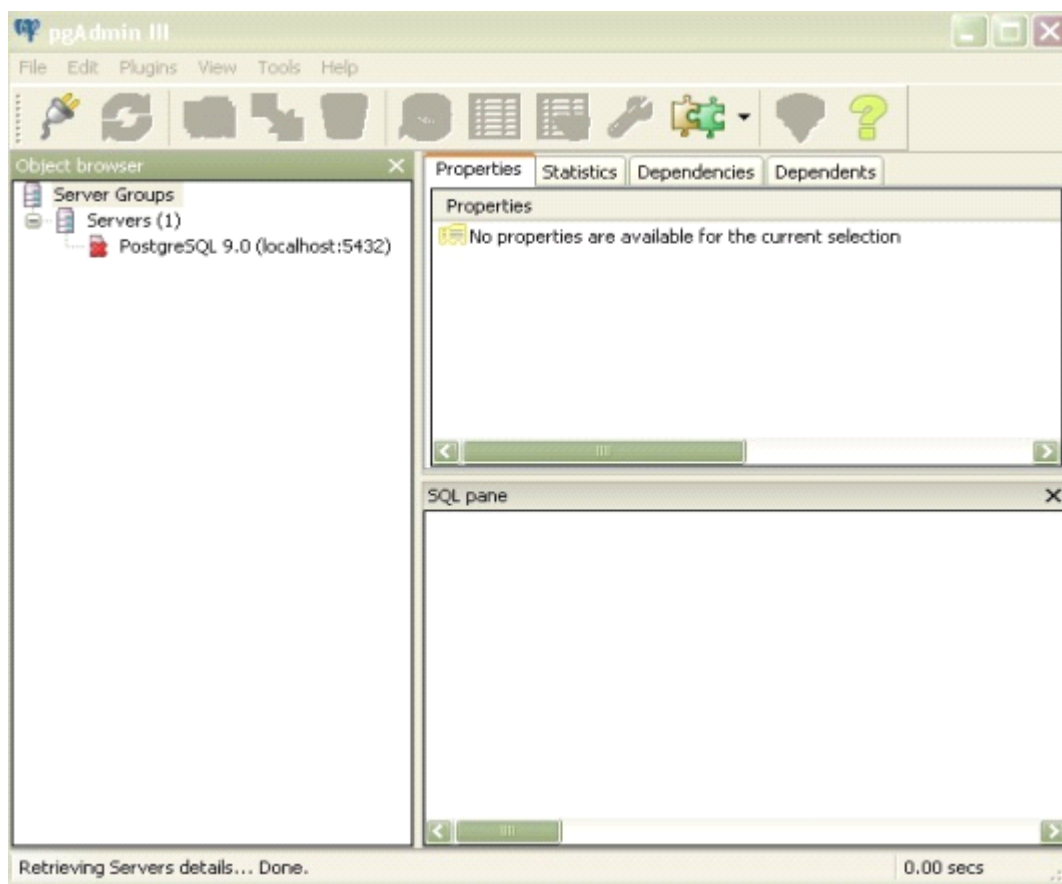


Figure 5. pgAdmin III window.

The Figure 5 shows the pgAdmin III interface which is the first view that users can see when they manage PostgreSQL 9.0. pgAdmin III has a simple and clear interface displays "File", "Edit", "Plugins", "View", "Tools", and "Help". It has three modules: "Object browser", "Properties/Statistics/Dependencies/Dependents" and "SQL pane". In the Figure 5, the most of functions are not available, because of the servers haven't been connected yet. If user wants to connect the current database as in the Figure 5, she/he should click "postgreSQL 9.0 (localhost:5432)" which is in the Object browser on the left side, then click right button and choose "connect database". It will show a window that ask user's password which has been set in installation. In this step users can choose if they want to store the password or not. After the password is matched, the original database will be connected.

The database is connected. Users can operate and check all the information now in the current (original) database. Also it means pgAdmin III is available now, users can maintenance and manage database (or create new database) at right now. Users can clearly see the available pgAdmin III interface as the Figure 6:

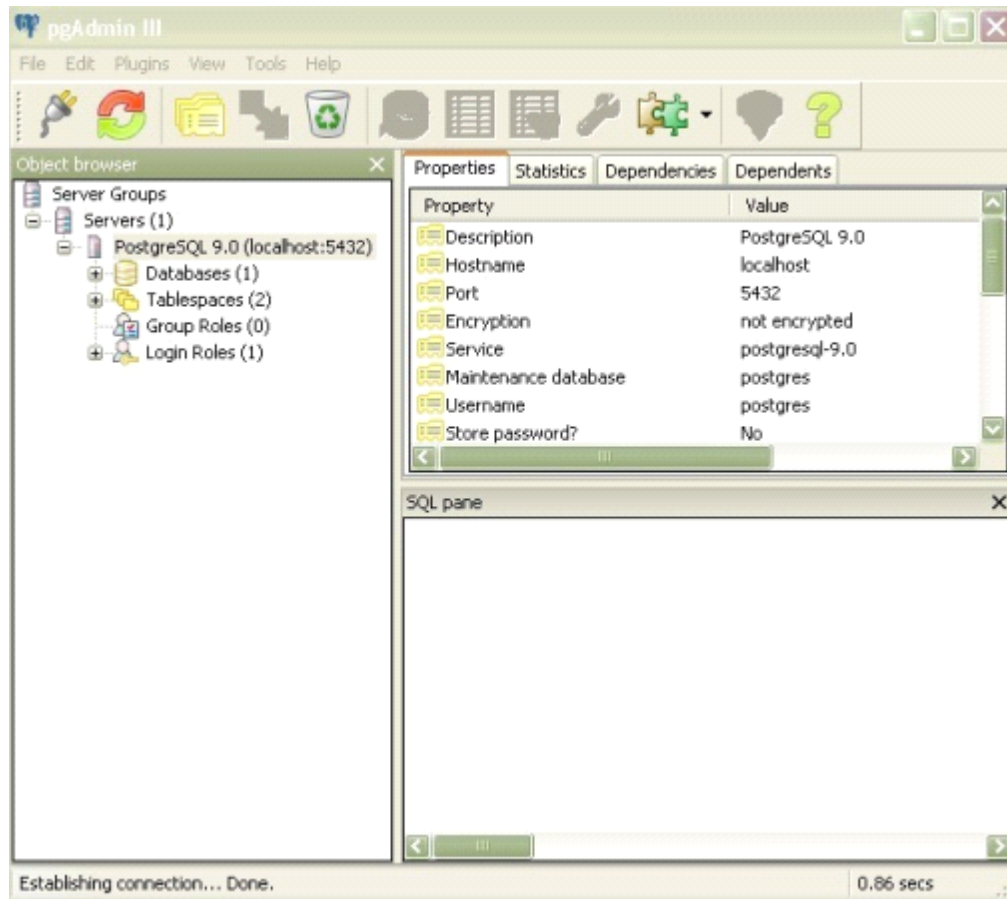


Figure 6. Available pgAdmin III.

The Figure 6 shows interface of pgAdmin III after database is connected. In the Object browser window, there is "Server Groups". Under the Server Groups, there is one server in the "Servers", one original database in the "Databases", two system tablespaces in the "Tablespaces" and one login role - Postgres. All the information are available now. Users can check everyone in the "Object browser", they can click one and that one's information will show in the right side. Users can check everyone from its property, statistics, dependency, and dependent. For example, in the Figure 6, user clicks "PostgreSQL 9.0 (localhost:5432)", then can check its information from "Properties", "Statistics", "Dependencies" and "Dependents". In the "Properties", it shows properties and their values in the "PostgreSQL 9.0 (localhost:5432)" such as "Description: PostgreSQL 9.0", "Hostname: localhost", "Port: 5432", "Encryption: not encrypted", etc. Those are original setting after installation.

In that original database, users can insert values, also can delete or add databases. In the thesis, it will display using pgAdmin III to create a new database, new table, and insert information into the new table. pgAdmin III is definitely helps users to manage PostgreSQL very easily.

Users can create new database by choosing "Databases" in the "Object browser", then clicking the right button to choose creating "New database". The window for creating new database has "Properties", "Variables", "privileges" and "SQL" for making different databases depend on users' requirements.

The new database is named "test". Double clicking the name of the database in the Object browser window on the left side. The database is connected. User can create a new schema in the new database. Clicking "test" which is the name of the new database, then click right button to choose "New object" and choose " New Schema". The window for "create new schema" has the same structure for "create new database". It has "properties", "privileges", "default privileges" and "SQL" for creating different schemas depend on user's requirements.

The new schema is named "test-1". Users can refresh window if can't see the new schema. User can create a new table in the new schema. Clicking "test-1" which is the name of the new schema, and choose "New Object" which can be found from a list when user click the right button. Choosing "new table" from "New Object"'s list, then user can get a window for creating table.

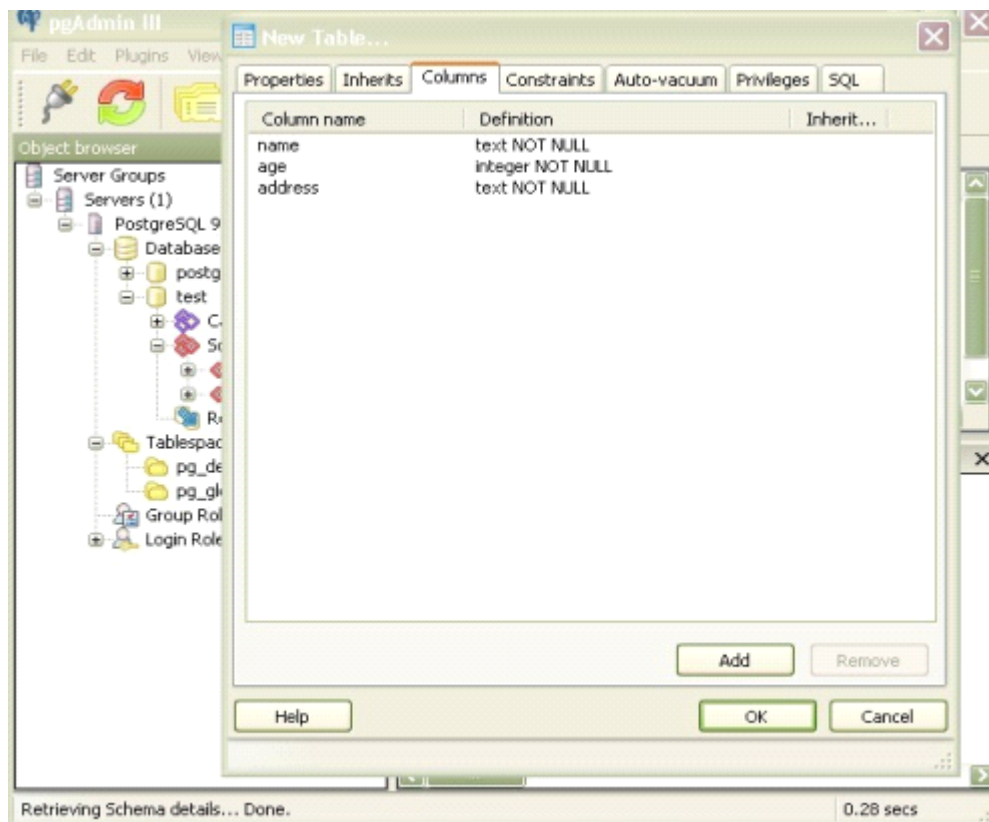


Figure 7. Window for creating new table.

Figure 7 shows a window for creating a new table. It has "Properties", "Inherits", "Columns", "Constraints", "Auto-vacuum", "Privileges", and "SQL" for setting values. In the "Columns", it already has three columns in the new table.

Users can name their tables in the "Properties" of creating table window. The new table is named "user". The Figure 7 shows there are "name", "age", and "address" in the "Columns" of the table. In the "Constraints" window, user can add primary key by clicking "Add" button in the window. After clicking "Add" button, it will show a window named "New Primary Key". It has "Properties" and "Columns" for setting. The new primary key has been named "Primary_Key", and the new primary key can be chosen from the "Columns" window, by adding one of columns in the table. Here "name" has been chosen as the new primary key. Clicking "OK" then the new table is available.

The next step is inserting values to the new table. Clicking the name of table which can be found in the Object browser window, then clicking the right button, in the list of the "Script" choosing "INSERT script". Then user can see a window like the Figure 8:

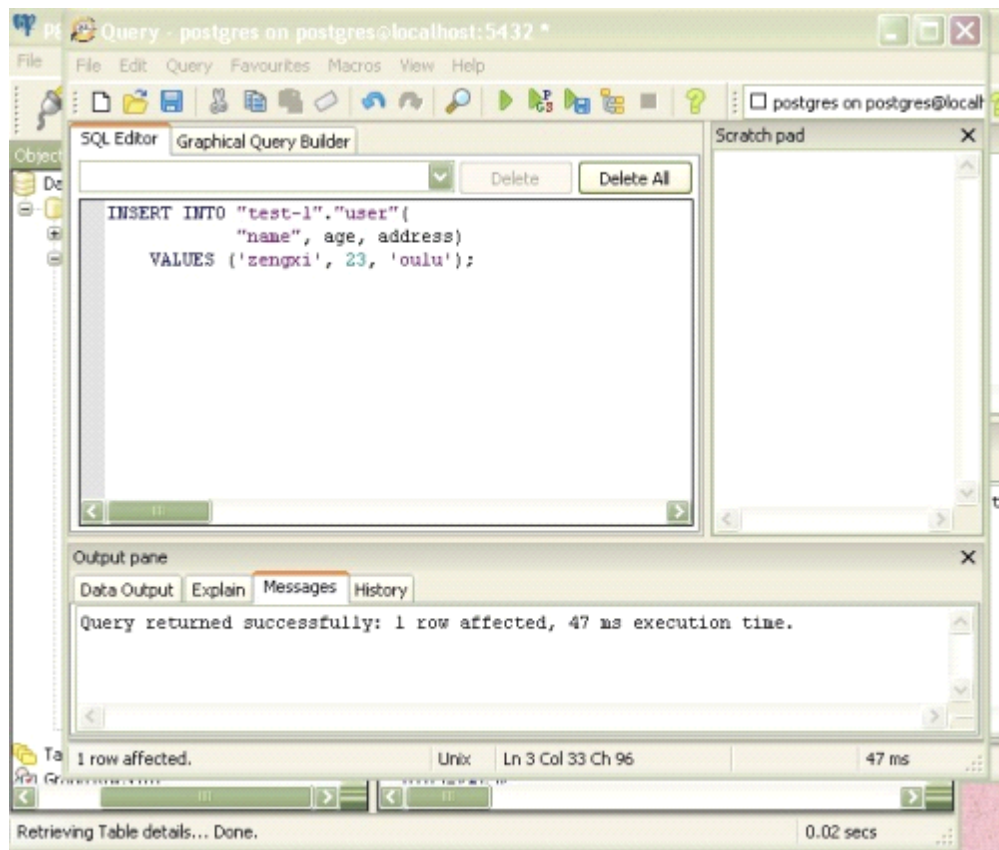


Figure 8 . Query window, when users insert script for table.

The Figure 8 shows a window which for querying the new table. Users can edit SQL by choosing "INSERT script". In the "SQL Editor" of the Query window, it already gives all the values which has been inserted in the table: schema's name, table's name and table' values. Users can insert script and run it, then users can get a result like the Figure 9.

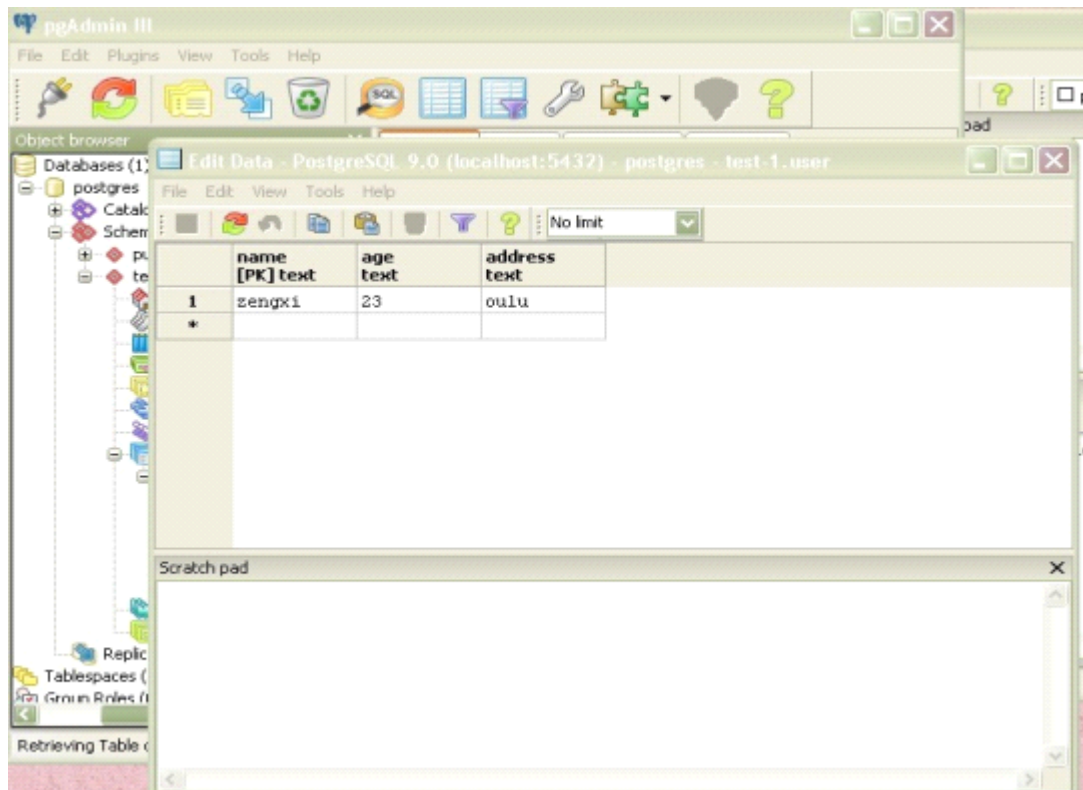


Figure 9. Edit data window.

The Figure 9 shows the information of the table after inserting. The window can be found by clicking the right button and choose "View all the data" in the list. In this window, users can edit values such as deleting and adding values. This is quite easy way to edit values in the tables.

From the installation to the operation, PostgreSQL is no difference with another databases, easy installation, simple operation, and clearly interface for every step. There is also the documentation of pgAdmin III for solving users' different problems and questions.

5. PostgreSQL in different programming environments

Previously, it presented the PostgreSQL installation and basic operation in Windows. In this chapter, it will display the usability of PostgreSQL. It will present PostgreSQL works in the different programming environments by using different programming languages.

The compatibility of PostgreSQL has been checked by using different programming languages (C, C++, Java). And also has been checked with different integrated development environments (IDE). In the chapter 3, it presented that PostgreSQL supports many different programming languages and programming environments. In this chapter, it will lead us to the truth. Does its compatibility as strong as it introduced?

5.1. PostgreSQL and Qt

Qt SDK (open source) has been chosen as the first test. Qt has been in widely use, for 15 years, Qt has helped hundreds of thousands of developers and companies. It has been used in different kind of environments such as mobile, home media, aerospace&defense, automative infotainment, medical, IP communication, oil&gas, visual effects, MIDs, E-books, and Linux distors. Qt SDK includes Qt creator, Qt libraries, and tools. The installation of Qt SDK v2010.05 is simple, users need to install it in a directory which doesn't has space between folders' name. It's the same reason as installation of PostgreSQL. After Qt SDK is installed, user can create a project. If the system shows a warning that can't find path and has an error in Makefile when create a project, then it is needed to download and install Qt 4.7. Problem will be solved after install Qt SDK and Qt 4.7 in same directory. It won't have any warning or error when create a new project. After installation, clicking the Qt creator, uses can create a project in the welcome page. Here chooseing "Qt C++ project", and "Qt console Application" in the thesis. The new project named "test". Two files are created with the new project - "test.pro" and "main.cpp". In the "test. pro" file, it needs to add "QT += sql" in the code, which means add sql to Qt project. Before programming, it is necessary to know some basic knowledge about PRO file, such as "qmake", "namke", "-I", "-D", "dll" etc. It would help a lot to understand the original code in the "test.pro" and also can add/delete some codes during programming./21/

To understand "qmake" is very important for using Qt. It is computer program that automates the generation of Makefiles. Makefiles are used by the program Make to compile software from source code, therefore qmake is a make-makefile tool, or makemake for short. Also the basic knowledge about system variables is needed, such as: TARGET, CONFIG, SOURCES, INTERFACES, etc. How to use variables and key words, such as "qmake", "Makefile" are very important for managing Qt creator. /9/

There is an online reference documentation in the Qt official website. It has documentations for every different editions, such as Qt, tools, and addons. There are examples, tutorials and demos for different editions. It won't be a problem if it is the first time to use Qt. It is easy to find the code for connection with PostgreSQL by using C++. There are many tutorials about how to manage PostgreSQL with Qt, such as deleting, adding, updating and maintenance.

In the Qt reference documentation, there is an example about connecting PostgreSQL in QtSql. User can copy and paste the code to the "main.cpp", debug and run the project. The result shows, "QPSQL is not loaded". There are many articles can be found in Internet to teach how to solve this problem, such as adding path to "Environment variables", changing the sequence of installation, downloading drivers' libraries, changing code in PRO file, using different codes to connect PostgreSQL, and copying DLL files from PostgreSQL's bin to Qt's library. All those ways have led to the same result in this case. Opening both folders of PostgreSQL and Qt, it is not hard for user to find out that in the Qt folder there is QPSQL plug-in missing which is in the plugins folder of the sqldrivers folder. So it means the next step is building the QPSQL plug-in.

First it needs to download "make tools", and update environmental variable PATH for the "make" directory. It also needs to download "mingw-utils", user gets "reimp tool" after extracting the "mingw-utils". Users need to copy and paste "make tools" and "reimp tools" to the system32 folder which is under the WINDOWS folder. After preparation, it needs to open Qt Command Prompt which can be found in Start menu. The basic commands about DOS is needed, there are many documents about DOS commands can be found in Internet. Using "cd" which is DOS command to go where PostgreSQL is installed. Finding sub-directory "lib" which is in the PostgreSQL folder, and run "reimp libpq.lib" (it is a library file in the PostgreSQL's folder) to produce the "liblibpq.a" and "libpq.def" files. Then use DOS commands to go where is needed to add the plug-in. User can use "qmake" and "Makefile" to produce new DLL file. After that, user needs to clear the cache, debug and run the project. Now the result shows QPSQL is available./10/

After QPSQL is available, user can use Qt to manage PostgreSQL, for example, create a new database, or add/delete new information in the current database. For new users, it is not a problem to use Qt. In the Qt Centre, there are many documents, FAQ, blogs for programmers' questions and problems. In this test, it proves PostgreSQL and Qt are compatible, and PostgreSQL supports C++.

5.2. PostgreSQL and Dev-C++

Dev-C++ has been chosen as the second test. It is a very popular IDE for programming C and C++. It is bundled with MinGW, a free compiler. And it is very easy to download in Internet, and very simple to install. After installation, user can open Dev-C++ and choose to create a new project. The console application and C project has been chosen in this project. The project name is "Test". For people who maybe don't

know how to connect PostgreSQL and Dev-C++ by using C language, there is an example from PostgreSQL official website:

```

#include <stdio.h> /* stdio header file */
#include <stdlib.h> /*stdlib header file */
#include "libpq-fe.h" /* libpq header file */
int main() /* main function starts*/
{
    char    state_code[3]; /* holds user state code */
    char    query_string[256]; /* holds constructed SQL query */
    PGconn  *conn; /* holds database connection */
    PGresult *res; /* holds query result */
    int     i;
    conn = PQconnectdb("dbname=test"); /* connect to the database */
    if (PQstatus(conn) == CONNECTION_BAD)
    {
        fprintf(stderr, "Connection to database failed.\n");
        fprintf(stderr, "%s", PQerrorMessage(conn));
        exit(1);
    }
    printf("Enter a state code: "); /* prompt user for a state code */
    scanf("%2s", state_code);

    sprintf(query_string, /* create an SQL query string */
            "SELECT name \
            FROM statename \
            WHERE code = '%s'", state_code);

    res = PQexec(conn, query_string); /* send the query */

    if (PQresultStatus(res) != PGRES_TUPLES_OK) /* did the query fail? */
    {
        fprintf(stderr, "SELECT query failed.\n");
        PQclear(res);
        PQfinish(conn);
        exit(1);
    }

    for (i = 0; i < PQntuples(res); i++) /* loop through all rows returned */
        printf("%s\n", PQgetvalue(res, i, 0)); /* print the value returned */

    PQclear(res); /* free result */

    PQfinish(conn); /* disconnect from the database */

    return 0;
}

```

Program 1. Connect to PostgreSQL in C language. /11/

The Program 1 shows using C++ to connect PostgreSQL and Dev-C++. There are explanations in the steps. There are few things needed before to compile the project. Firstly, it needs to add library folder of PostgreSQL to the project. Finding the project "test" which is just created. Clicking "project options", and selecte the "Parameter Tab", then user needs to click the "Add Library or Object" button to find the "libpq.a" file which is in the PostgreSQL's library folder. The library directory of the new project is: "C:\PostgreSQL\9.0\lib\libpq.a". After that, user needs to find the "Directories Tab" in the project options window, and switch it to the "Include Directories" window. In the Include Directories window, it needs to add include folder's directory where is under the PostgreSQL folder. The include directory of the project is:"C:\PostgreSQL\9.0\include".

After those preparation, users can copy an example from the official PostgreSQL's website, and paste it to Dev-C++. It needs to change some details information of database, then user an run the project. There is one thing should be notice, if the result likes flash and disappear quickly, then user should to add "getchar();" after "return 0" in the code.

5.3. PostgreSQL and Eclipse

Eclipse has been chosen as the third test. It is free and open source development environment which is comprising an IDE and an extensible plug-in system. Eclipse supports Java, Ada, C, C++, COBOL, Perl, PHP, Python, Ruby, Scala, Clojure and Scheme programming languages./26/

Firstly, if users want to connect Eclipse and PostgreSQL by using Java , it needs to install the Java driver (JDBC) which can be downloaded from official website of PostgreSQL. Extracting it after downloaded, then user can see the "jar" file. Downloading Eclipse from Eclipse's official website and extract it then users can use it by double clicking the "Eclipse.exe". It is needed to add jar file's path into the Eclipse's "classpath" setting, if users do not add the path, it will show an error like class is not found when run the project. Opening Eclipse, users can see the interface for Eclipse when close the welcome page. User needs to click "create a new project", and choose "generate project". The new project is created and named"test". There are many examples can be found when google "Java, PostgreSQL" or "Eclipse, SQL". Those examples will give users ideas how to connect and control PostgreSQL by using Java, such as inserting, deleting and creating. It is important step to install JDBC if users want to connect PostgreSQL with Eclipse by using Java. Eclipse also supports PHP, Perl, Rudy and C++,etc. It would be interesting to see managing PostgreSQL in Eclipse by using different programming languages. Here is code for using Java:

```
import java.sql.DriverManager;           /*import java.sql*/  
import java.sql.Connection;  
import java.sql.SQLException;
```



```
public class Example1 {
public static void main(String[] argv)
{
System.out.println("Checking if Driver is registered with DriverManager.");/*output*/

try {
    /* new function for connection fails*/
    Class.forName("org.postgresql.Driver");
}
    catch (ClassNotFoundException cnfe)
    {
        System.out.println("Couldn't find the driver!");
        System.out.println("Let's print a stack trace, and exit.");
        cnfe.printStackTrace();
        System.exit(1);
    }

System.out.println("Registered the driver ok, so let's make a connection.");

Connection c = null;

try {
    /* The second and third arguments are the username and password, */
    /* respectively. They should be whatever is necessary to connect */
    /* to the database. */
    c = DriverManager.getConnection("jdbc:postgresql://localhost/booktown",
        "username", "password");
}
    catch (SQLException se)
    {
        System.out.println("Couldn't connect: print out a stack trace and exit.");
        se.printStackTrace();
        System.exit(1);
    }

if (c != null)                /* if functions for database connection description*/
    System.out.println("Database connected!");
else
    System.out.println("Fail to connect to database.");
}
}
```

Program 2. Connect to PostgreSQL in Java. /12/

The Program 2 shows using Java to connect PostgreSQL with Eclipse. It needs to ensure that the Driver gets registered within the code. The first three lines of the code makes available for the required classes to register with the Driver Manager object, to create a connection object and to use SQLException object.

6. PostgreSQL and web programming languages

After tested PostgreSQL in different environments by using different programming languages. It is interesting to find out how does it work with different web programming languages. As a database management system, it is important to compatible with different web programming languages. There are many web programming languages have been widely used, such as PHP, Ruby, Python, HTML, JavaScript, etc. In this chapter, PostgreSQL will be tested with three of the most popular web programming languages - PHP, Python and Ruby.

6.1. PHP

PHP is one of the most popular web programming languages. It is very special web programming language, it unique mix of C, Java, Perl. It can display dynamic web pages faster than CGI and Perl. I have used PHP many times to connect with MySQL. In this test, PHP will be used to connecte and manage with PostgreSQL./27/

It is needed to make sure that computer has everything to support using PHP to connect PostgreSQL. It needs to download PHP and Apache. The newest version PHP 5.3.6 and Apache 2.2 has been installed. It is the time to start programming. The result has been chosen to display in the HTML page, so it needs to use HTML and PHP to connect database. Here's the code:

```
<html>
<body>
<?                                /*php starts*/
$host = "localhost";
$user = "postgres";
$password = "jeshi1987";
$db = "test";                    /* open a connection to the database server*/
$conn = pg_connect ("host = $host dbname = $db user = $user password =
$password");
                                /* database connection*/
if (!$conn)
{
die ("could not open connection to database server");
}

/* control the database, here it will generate and execute a query*/
$query = "SELECT * FROM user";
$result = pg_query($conn, $query) or die ("Error in query: $query.
".pg_last_error($conn));

$rows = pg_num_rows($result);
```

```

echo "There are $rows users in the database.";      /*close database connection*/
pg_close($connection);
?>
</body>
</html>

```

Program 3. Connect to PostgreSQL in PHP.

The program 3 shows using PHP to connect and control PostgreSQL. The result of this project will show: "There are 2 users in the database". The code above can summary like this: using PHP to get data from PostgreSQL database has several steps. The first step, connect with PostgreSQL database. After connection is done, then it goes to next step. In the second step, it needs to manage the database, users can edit database, query database, delete and update the database. In the third step, it needs to close the database connection. That's all about how to use PHP to manage PostgreSQL, it doesn't have so big difference with MySQL, isn't it?

6.2. Python

Python has been chosen as the second test. It needs Apache, install it if there is no Apache in the computer. Django is needed as well. It is an open source web framework written by Python, it can reduce the workload, and provide interface for PostgreSQL extract and storage. Django supports PostgreSQL, MySQL and Oracle. Comparing with another programs installation, Django is different. Because there is no such an ".exe" file can directly finish installation, it has to install in DOS. Firstly, it needs to install Python and extract Django. Secondly, it needs to copy and paste the extracted Django folder to C disk. It can copy and paste to any folder which can easily be found by using DOS. Thirdly, using DOS to go where the Django folder is. Fourthly, typing "setup.py install" and wait it complete the installation . There is a way to check if Django is installed on the computer or not. Finding Python in the "Start" menu and choose the command line. Typing command "import django" and "django VERSION", then it shows the version installed. The result shows in the test: <1.2.5 final,0>. /28/

Python is very powerful, comprehensive, stable, and simple language. It supports most of operating systems, and has large amount of users. When google Python, it shows 104,000,000,000 results less than 1 second. So it is not a problem for beginners to study it.

PostgreSQL has adapters for Python, the most popular one is Psycopg. Windows version can be found from its official website, and choose the right version depends on the version of Python users have been installed. In this test Python2.7 has been installed. "Python Shell" can be used to test that if "Psycopg 2.0" has been installed or not. Typing "import psycopg2" in the Python Shell, and then type the command "psycopg2.apilevel". The result shows: "2.0", it means Psycopg 2.0 is already installed.

For beginners, it is impossible to handle a new language, they can check examples in Internet. There are lot of examples about using Python control PostgreSQL in the

PostgreSQL official website. It is easy to understand by checking every step's meaning. Here is an example for connecting PostgreSQL:

```
DATABASE={
'default':{
'ENGINE': 'django.db.backends.postgresql_psycopg2 /* connection with database*/
#add 'postgresql_psycopg2','postgresql' /* add files*/
'NAME': 'postgres',
'USER': 'postgres', /*database information*/
'PASSWORD': 'jeshi1987',
'HOST': 'localhost',
'PORT': '5432',
}
}
```

Program 4. Connect to PostgreSQL in Python. /13/

The Program 4 shows using Python to connect PostgreSQL. The code is very simple and easy to understand. It needs to make sure that all the information in the database are correct. After studying more Python, then it is not hard to use Python to manage the PostgreSQL, such as deleting, inserting, creating, and updating, etc.

6.3. Ruby

Ruby is very young object-oriented programming language. Since it has been published in December 1995, Ruby has drawn devoted coders worldwide. Perl's pronunciation likes the birthstone of June - pearl, and Ruby has same name as the birthstone of July - ruby. /14/

Ruby is fully object-oriented programming language, everything can be an object in Ruby, include basic data types in other language, such as integers. And everything has value, no matter mathematic, logical expression or sentence. Before of those special points make many programmers start to learn this language. Ruby is also totally free. Not only free of charge, but also free to use, copy, modify, and distribute. Those features are reasons for choosing it as test to connect with PostgreSQL./14/

When discuss about Ruby, Ruby on Rails will come with the discussion. It is a popular web application framework which is written by Ruby. People use Rails to reduce the workload and provide interface for databases extract and storage. It is needed to install Ruby on Windows. It can be downloaded from Ruby official website.

Before get starting, it is needed to configure a database. Every Rail application will interact with a database. The database to use is specified in a configuration file "config/database.yml". :

```
development:
adapter: postgresql
encoding: unicode
database: blog_development
```

```
pool: 5  
username:  
password:
```

Program 5. Configuring PostgreSQL database. /15/

The Program shows configuring PostgreSQL database. The default database configuration use the SQLite3. It is needed to change SQLite3 to PostgreSQL. The "config/database.yml" file will be customized like the Program 5.

It has been used "console" in the project. The console is a command-line tool that execute Ruby code in the context of application: "\$ rails console". And it can create connection with PostgreSQL:

```
require 'postgres'  
conn = PGconn.connect("", 5432, "", "", "test") /16/
```

In the guide of Ruby on Rails, there are many examples can be found as reference. There is no such an example exactly shows how to use Ruby to manage PostgreSQL, but users can get some tips from those examples. It is a long way to study one language, and it is impossible for beginners to handle Ruby in short days. Ruby is very young programming language, there is not many tutorials can be found in Internet if compares it with another languages. It still needs programmers to keep working and sharing the resources.

7. Conclusions

Before started to write this thesis, I have visited many big libraries in China to try to find a Chinese book of PostgreSQL, but I didn't find. "PostgreSQL" only appears when authors make comparison with another database management systems. But when I started researching, I found out that there're many useful information can be found in Internet, such as, English books, tutorials, guides, documents, etc. PostgreSQL is more popular than I thought, it has large numbers of users and it is supported by a huge independent community. On the other hand, we can see, there is not enough Chinese document of PostgreSQL. It needs people keep working that translate English PostgreSQL documents to Chinese, and publish them in Internet. It would attract more Chinese users.

In the last few years, MySQL has been used in different project works, such as Programming project and Distribute system project. Now I get a chance to study PostgreSQL and compare it with MySQL by myself. It's not only read articles in Internet, but also test its usability by using different languages in the different environments. PostgreSQL is really good database management system. It is fast, safe, reliable and definitely technique strong.

I was confuse when I read my topic of thesis in the beginning. There were many parts I can't understand. For example: "install PostgreSQL in different programming environments"; if image programming environment as an IDE which supports programs to run there, then how can install one software into an IDE? But it didn't take long time for me to solve the problem. "Install PostgreSQL in different programming environments" also can translate to test PostgreSQL's compatibility in the different environments. It is abstract but easy to understand if imaging PostgreSQL is cocoa, programming environment is milk, and programming language is stick. If we want a cup of hot cocoa, we need to add cocoa into a hot milk then stir them with stick. The thesis is about using same cocoa to put in different brands of milk and stir with different sticks to get different flavors of hot cocoas. This thesis is exactly proving that how strong PostgreSQL's compatibility is. PostgreSQL supports many languages and environments. In the thesis, PostgreSQL has been tested with different programming environments, such as Qt creator, Dev-C++, and Eclipse by using C, C++, Java, PHP, HTML, Ruby and Python.

The first and the hardest problem I have met was to solve the connection between PostgreSQL and Qt creator. I have downloaded Qt SDK, it includes Qt creator, tools and Qt libraries. It showed the result that QPSQL is not loaded when I tried to connect PostgreSQL with Qt by using C++. No matter how I changed code, added different paths, or copied different libraries to different files, it always showed the same result. I almost gave up that tried to use Ubuntu to connect PostgreSQL and Qt. But luckily I found a article from the Qt Centre, it presented how to build QPSQL plug-in on Windows to solve the problem which is QPSQL can't loaded. I tried to build plug-in step by step following the article. Finally I solved the problem. Even it took almost two weeks to find the right solution, but I think it was worth.

I found out a problem when I prepared web programming environment for PostgreSQL that there are many softwares still do not support PostgreSQL. They are like WAMP (Windows, Apache, MySQL, PHP), bundle MySQL with PHP, Apache and Windows. For example, I planed to use EsayPHP to get development of PHP and PostgreSQL on the localhost, but EasyPHP doesn't support PostgreSQL. So does Joomla, I planed to check how it works with PostgreSQL, but it doesn't have a version for supporting PostgreSQL. As an open source, PostgreSQL is totally free, it doesn't belongs to any company or organization. MySQL is different, Oracle owns it, it may some day face to cost totally. So it is a big marketing for WAMP software to create a version which can compatible with PostgreSQL. It would be a great step for both programs, and I believe that will attract more users.

I have learned a lot of professional skills when write the thesis. At the beginning of the preparation, I have read lot of documents. It is not only about PostgreSQL, but also includes standard of SQL, development and history of MySQL, evaluation of both databases and meaning of technical words. If I don't understand a single word in a article, to make sure I won't make mistake when I write the thesis that I will keep studying until I understand it. Those documents which has been searched included many information that I don't have written in the thesis. I need to make sure I am not confuse when I start writing. When I test PostgreSQL by using web programming languages, I have learned frameworks which written in those languages. PostgreSQL has been installed in Windows workstation, I have learned how to install it in Linux also. I have study many programming environments and languages that I have never use before. And the most important thing is that I have learned the usability of PostgreSQL.

At the end of the thesis, I want to thanks those people who has contributed for open sources documentations. They are very helpful, especially for beginners who study a new language or a new environment. Those documents are like lights in the dark road to lead people go to right place. They have helped me to solve all kind of problems I have met when I was researching, installing and programming. I really appreciate those programmers who contribute for open sources. I wish I can join the team some day too.

8. REFERENCES

List of the sources:

- /1/W3schools, [WWW-document],
<http://www.w3schools.com/sql/sql_intro.asp > 29.03.2011
- /2/ RoseIndia, [WWW-document],
<<http://www.roseindia.net/databases/sql.shtml> > 02.04.2007
- /3/ PostgreSQL, [WWW-document],
<<http://www.postgresql.org/about/> > 18.12.2010
- /4/PostgreSQL, [WWW-document],
<<http://www.postgresql.org/about/history> > 11.03.2011
- /5/Acs, [WWW-document],
<<http://www.itsabacus.com/mysql.html> > 30.03.2011
- /6/MySQL, [WWW-document],
<<http://www.mysql.com/news-and-events/sun-to-acquire-mysql.html> > 16.01.2008
- /7/ORACLE, [WWW-document],
<<http://www.oracle.com/us/corporate/press/044428> > 27.01.2010
- /8/eWeek, [WWW-document],
<<http://www.eweek.com/c/a/Linux-and-Open-Source/PostgreSQL-vs-MySQL-How-to-Select-the-Right-OpenSource-Database/> > 21.10.2010
- /9/ Google, [WWW-document],
<<http://www.google.com/search?hl=en&defl=en&q=define:Qmake&sa=X&ei=t8dsTcGNA4mWOpvE8YAD&sqi=2&ved=0CBMQkAE> >
- /10/ QtCentre, [WWW-document],
<http://www.qtcentre.org/wiki/index.php?title=Building_the_QPSQL_plugin_on_Windows_using_MinGW >
- /11/PostgreSQL, [WWW-document],
<http://www.postgresql.org/files/documentation/books/aw_pgsql/writing_apps/node3.html > 31.01.2010
- /12/Practise PostgreSQL, [WWW-document],
<<http://www.faqs.org/docs/ppbook/x20856.htm> >

- /13/ WangFan, [WWW-document],
<<http://wenku.baidu.com/view/8e32d10c6c85ec3a87c2c500.html>> 13.01.2011
- /14/ Ruby, [WWW-document],
<<http://www.ruby-lang.org/en/about/>>
- /15/ Rails guide, [WWW-document],
<http://guides.rubyonrails.org/getting_started.html> 30.08.2010
- /16/osChina, [WWW-document],
<http://www.oschina.net/code/snippet_12_493> 18.10.2010
- /17/ PostgreSQL, [WWW-document],
<<http://www.postgresql.org/about/news.1235>>
20.09.2010
- /18/MySQL, [WWW-document],
<http://www.mysql.com/why-mysql/>
- /19/PostgreSQL, [WWW-document],
<<http://www.postgresql.org/docs/8.4/interactive/continuous-archiving.html>>
- /20Linuxeden, [WWW-document],
<http://www.linuxeden.com/html/news/20100317/76110.html> 17.03.2010
- /21/ Qt, [WWW-document],
<<http://qt.nokia.com/qt-in-use>>
- /22/ MySQL, [WWW-document],
<<http://www.mysql.com/customers/industry/?id=85>>
- /23/ eWeek, [WWW-document],
<<http://www.eweek.com/c/a/Linux-and-Open-Source/PostgreSQL-vs-MySQL-How-to-Select-the-Right-OpenSource-Database/>> 21.10.2010
- /24/ PostgreSQL, [WWW-document],
<www.postgresql.org>
- /25/ MySQL, [WWW-document],
<www.mysql.com>
- /26/ Eclipse, [WWW-document],
<<http://www.eclipse.org/org/>>
- /27/ PHP, [WWW-document],
<<http://www.php.net/manual/en/index.php>>

/28/ Django, [WWW-document],

<<http://docs.djangoproject.com/en/dev/faq/general/#what-does-django-mean-and-how-do-you-pronounce-it> >