



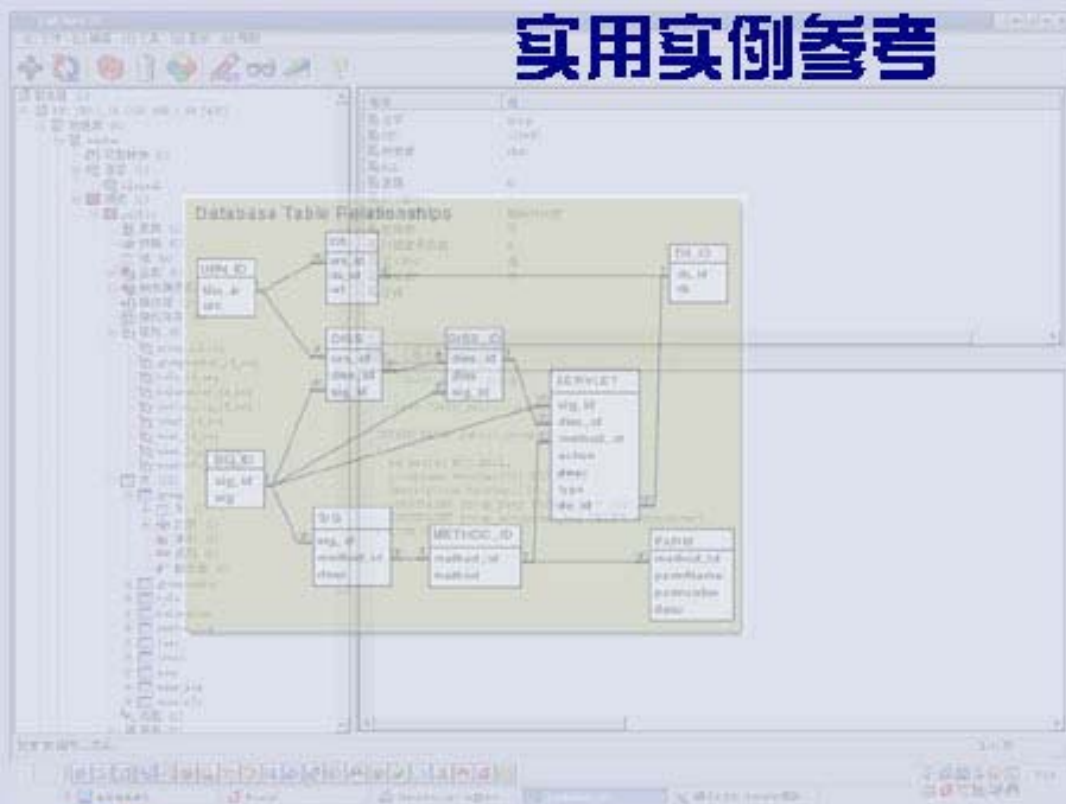
# PostgreSQL



[Download](#) • [Mirrors](#) • [Lists](#) • [Users](#) • [Developers](#) • [Docs](#) • [Search](#)

# PostgreSQL

## 实用实例参考



PostgreSQL has won the 2003 Linux Journal Editors' Choice Award for the best DBMS!

深圳市福田区梅林一村

# PostgreSQL 实例参考

(netkiller)

PostgreSQL  
200  
2  
PostgreSQL  
OpenLDAP  
PHP + Corba + Python  
JBuilder +  
Weblogic + PostgreSQL  
EJB  
netkiller(at)9812(dot)net  
300  
HTML  
PDF PS PostScript  
Microsoft Word  
300  
doc  
XML  
make  
docbook / latex  
Page Maker  
docbook  
UNIX  
Page Maker  
Adobe  
PDF  
" "

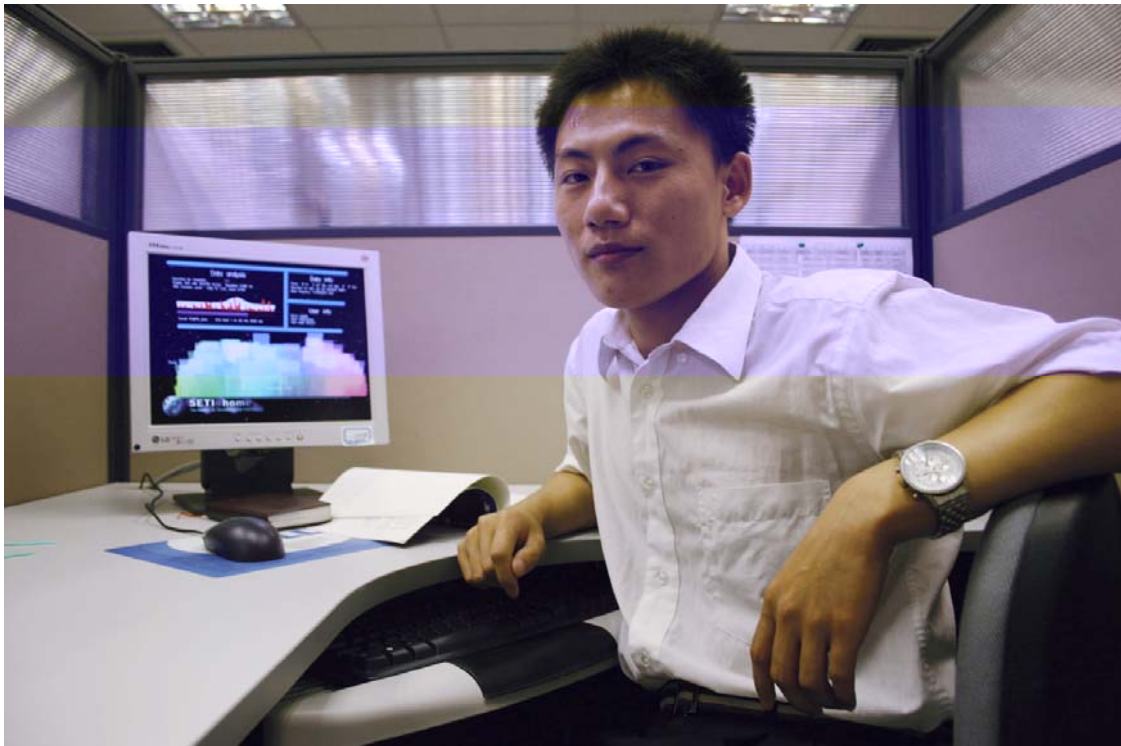
## 1.1 本文档的读者对象

create  
select insert update

## 1.2 本文档主要内容

PostgreSQL  
SQL

## 1.3 怎样使用本文档



netkiller, UNIX like ,  
LDAP J2EE Corba

<http://www.9812.net/>

ICQ:101888222

Yahoo:snetkiller

AIM:xnetkiller

openunix@163.com

E-Mail: [netkiller@9812.net](mailto:netkiller@9812.net)

Email

Newsgroup

<news://news.cdut.edu.cn/cn.lang.java>

<news://news.cdut.edu.cn/cn.lang.python>

Web Newsgroup:

<http://202.103.190.130:8080/news>

**BBS**

<http://www.pgsqldb.org>

<http://www.chinaunix.com>

<http://www.linuxforum.net>

	.....	2
1.1	.....	2
1.2	.....	2
1.3	.....	3
	.....	3
	.....	5
	PostgreSQL .....	10
1	.....	10
1.4	.....	10
1.5	pgsql .....	10
1.6	PostgreSQL SQL99 .....	11
2	PostgreSQL .....	12
2.1	PostgreSQL .....	12
2.2	RPM .....	13
2.3	APT .....	17
2.4	PostgreSQL 8.0 beta for windows .....	19
	2.4.1    pgAdmin III.....	20
	2.4.2    psql .....	20
	2.4.3    ODBC.....	21
	2.4.4    Unix/Linux    Windows .....	23
	2.4.5    Windows    Unix/Linux .....	24
2.5	.....	24
	2.5.1 .....	24
	2.5.2 .....	25
	2.5.3 .....	26
	-PGP/GPG .....	26
2.6	.....	41
	2.6.1 .....	41
	2.6.2 .....	42
2.7	.....	42
2.8	.....	43
	2.8.1 .....	43
	2.8.2 .....	44
	2.8.3    vacuumdb .....	50
	2.8.4 .....	50
	2.8.5 .....	51
	2.8.6 .....	53
2.9	TCP/IP .....	56
	2.9.1    SSL    TCP/IP .....	56
	2.9.2    SSH    TCP/IP .....	69
2.10	ipv6 .....	71

3	DDL	74
3.1		74
3.1.1		74
3.1.2		74
3.1.3		75
3.1.4		75
3.1.5		75
3.1.6		76
3.1.7	date_part	76
3.2		77
3.3	" "	79
3.3.1	text to varchar	79
3.4		81
3.4.1		81
3.4.2	"1 2 3 4 5 6 7 8 9..."	82
3.4.3	"1 3 5 7 9..."	83
3.4.4	"2 4 6 8 10..."	84
3.4.5	n <sub>1</sub> +n <sub>2</sub>	85
3.5		85
3.6		85
3.7		87
3.8		87
3.8.1		87
3.8.2		87
3.8.3		89
3.9	/	91
3.9.1		91
3.9.2		92
3.9.3	PostgreSQL 7.3.x	92
3.9.4	-	93
3.9.5		101
3.10		101
3.10.1		101
3.10.2		101
3.10.3		101
4	Entity-Relation	103
4.1	E-R Entity-Relation	104
4.2		105
4.3		107
4.4		109
4.5		110
5	SQL DML	111
5.1		111
5.2	substring()	113

5.3	sum()	.....	115
5.4	n	.....	116
6		.....	119
6.1	VIEW	.....	119
6.2	HTML VIEW	.....	120
6.3	view	.....	124
6.4	20 " ..."	.....	125
6.5		.....	126
7		.....	127
7.1		.....	127
7.2	Select Into	.....	128
7.3	integer	.....	130
7.4	void	.....	130
7.5	record	.....	131
7.6		.....	132
7.7	shell	.....	133
8		.....	134
8.1		.....	134
9		.....	137
9.1		.....	137
9.2		.....	137
9.3		.....	140
9.3.1		.....	141
9.3.2		.....	141
9.4		.....	142
10		.....	145
10.1		.....	146
10.2		.....	147
11		.....	148
11.1		.....	148
11.1.1	- 1	.....	148
11.2	- 2	.....	149
12		.....	150
12.1.1		.....	150
12.1.1.1		.....	150
12.1.1.2		.....	151
12.1.2		.....	151
12.1.2.1		.....	151
12.1.2.2		.....	152
12.1.2.3		.....	152
12.1.3		.....	153
12.1.4		.....	153
12.1.4.1		.....	153
12.1.4.2	IP	.....	154

12.1.5	.....	154
12.1.6	.....	155
13	.....	155
	.....	155
13.1	.....	155
13.2	JDBC.....	155
13.2.1	Jsp/Java.....	156
13.2.2	toChinese() .....	156
13.2.3	Unicode (UTF-8) .....	156
13.2.3.1	setCharacterEncoding() .....	156
13.2.3.2	Web.xml Filter .....	159
13.2.3.3	Jdbc url charSet .....	165
13.3	Tomcat JNDI Datasource .....	165
13.4	JDBC SSL .....	168
13.5	.....	168
13.5.1	Create Java Entity Bean (not EJB CMP).....	168
13.5.2	.....	169
13.5.3	SQL .....	180
13.5.4	HTML .....	185
13.5.5	.....	213
14	PHP .....	215
14.1	PHP PostgreSQL .....	215
14.2	set CLIENT_ENCODING TO 'GB18030'; .....	215
14.3	convert() .....	225
14.4	PHP iconv() .....	225
14.5	I/O Linux iconv .....	226
15	/ .....	231
15.1	Macromedia Dreamweaver MX 2004 JSP .....	231
15.2	Jcreator .....	240
15.3	Eclipse .....	241
15.4	JBuilder + Weblogic + PostgreSQL .....	242
15.5	GUI .....	273
15.5.1	phpPgAdmin.....	273
15.5.2	PgAdmin III.....	275
15.5.3	Case Studio 2.....	276
15.5.4	PostgreSQL Manager .....	284
15.5.5	DeZign for Databases.....	289
15.5.6	GUI .....	291
16	FAQ.....	291
16.1	" " null.....	291
16.2	Postgresql .....	291
16.3	Putty .....	294
16.3.1	Putty .....	294
16.3.2	.....	299



16.4	.....	303
16.5	PostgreSQL RPM 5432 .....	303
16.6	PostgreSQL 7.4.2 rhel3( , AS3).....	306
16.7	Pureftpd pgsq .....	306
16.8	Vsftpd pgsq .....	306
16.9	OpenLDAP-PostgreSQL HOWTO .....	307
16.10	PostgreSQL .....	307
17	.....	307
17.1	.....	307
17.2	.....	315
17.3	.....	337
17.3.1	setenv.sh .....	337
17.3.2	install.sh.....	338
17.4	.....	342
17.5	.....	342
18	.....	343
19	.....	343

# PostgreSQL

## PostgreSQL Wins Linux Journal Editors Choice Award

Posted on 2004-08-02

Posted by [press at postgresql.org](#)

PostgreSQL has won the 2004 Linux Journal Editors' Choice Award for the best DBMS!

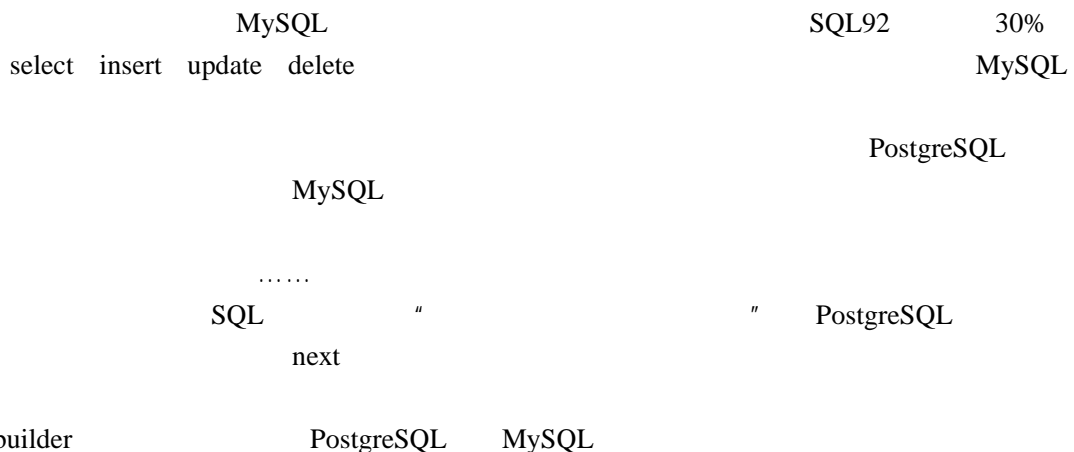
Linux Journal's Editors' Choice Awards are well-known as the premiere forum recognizing outstanding product developments and achievements in the Linux market, and winners of the sixth annual awards are featured in the August 2004 issue of Linux Journal.

Check out [the full article](#).

## 1

PostgreSQL	2000	PostgreSQL	2003	2000	5.x	6.x
		MS Sql Server 7/2000	Oracle 8	Oracle 8		
	MS Sql Server 7/2000	Oracle 8		Free Database		

## 1.4 关于性能



## 1.5 为什么说 postgresql 是最先进的开源数据库?

1.



--	--

32 TB

1 GB

32K

<http://www.pgsql.org/postgres-faq.html>4.5

## 2 PostgreSQL

### 2.1 PostgreSQL 分区

PostgreSQL

/boot	100	MB
/	1000	MB
/home	120000	MB
/usr	5000	MB
/var	5000	MB
/usr/local/mysql	10000	MB
/var/lib/pgsql	100000	MB
swap	2000	MB

240000 MB

243100 MB

```
[chen@linux chen]$ df
```

```
Filesystem      1K-blocks      Used Available Use% Mounted on
/dev/sda9        1004024        99892    853128   11% /
/dev/sda1         101089         9498     86372   10% /boot
/dev/sda2       120952116     7648124 107159936  7% /home
none              515400          0    515400   0% /dev/shm
/dev/sda10       2522048        33260   2360672   2% /tmp
/dev/sda7        5036284       2238244  2542208  47% /usr
/dev/sda6        5036284       1919140  2861312  41% /var
/dev/sda5        40313964        99444  38166636   1% /var/lib/pgsql
/dev/sda3        60476068       212532  57191508   1% /cvsroot
```

```
[chen@linux chen]$
```

```
[chen@linux chen]$ df -m
```

```
Filesystem      1M-blocks      Used Available Use% Mounted on
/dev/sda9         980           98     833   11% /
/dev/sda1          99            10     84   10% /boot
/dev/sda2       118117         7469   104648  7% /home
none              503            0     503   0% /dev/shm
```

```

/dev/sda10          2463      33      2305    2% /tmp
/dev/sda7           4918     2186     2482   47% /usr
/dev/sda6           4918     1875     2794   41% /var
/dev/sda5          39369      98     37272    1% /var/lib/pgsql
/dev/sda3          59059     208     55851    1% /cvsroot
[chen@linux chen]$

```

## 2.2 RPM 包安装

```

[root@linux software]# ls -l
postgresql-7.3.4-1PGDG.i386.rpm
postgresql-contrib-7.3.4-1PGDG.i386.rpm
postgresql-debuginfo-7.3.4-1PGDG.i386.rpm
postgresql-devel-7.3.4-1PGDG.i386.rpm
postgresql-docs-7.3.4-1PGDG.i386.rpm
postgresql-jdbc-7.3.4-1PGDG.i386.rpm
postgresql-libs-7.3.4-1PGDG.i386.rpm
postgresql-pl-7.3.4-1PGDG.i386.rpm
postgresql-python-7.3.4-1PGDG.i386.rpm
postgresql-server-7.3.4-1PGDG.i386.rpm
postgresql-tcl-7.3.4-1PGDG.i386.rpm
postgresql-test-7.3.4-1PGDG.i386.rpm
[root@linux software]# rpm -Uvh --nodeps `ls -l`
Preparing... ##### [100%]
 1:postgresql-test ##### [ 8%]
 2:postgresql ##### [ 17%]
 3:postgresql-contrib ##### [ 25%]
 4:postgresql-debuginfo ##### [ 33%]
 5:postgresql-devel ##### [ 42%]
 6:postgresql-docs ##### [ 50%]
 7:postgresql-jdbc ##### [ 58%]
 8:postgresql-libs ##### [ 67%]
 9:postgresql-pl ##### [ 75%]
10:postgresql-python ##### [ 83%]
11:postgresql-server ##### [ 92%]
12:postgresql-tcl ##### [100%]
[root@linux software]# rpm -qa|grep postgre
postgresql-devel-7.3.4-1PGDG
postgresql-7.3.4-1PGDG
postgresql-python-7.3.4-1PGDG
postgresql-contrib-7.3.4-1PGDG
postgresql-jdbc-7.3.4-1PGDG
postgresql-server-7.3.4-1PGDG
postgresql-debuginfo-7.3.4-1PGDG

```

```
postgresql-libs-7.3.4-1PGDG
postgresql-tcl-7.3.4-1PGDG
postgresql-test-7.3.4-1PGDG
postgresql-pl-7.3.4-1PGDG
postgresql-docs-7.3.4-1PGDG
[root@linux software]#
[root@linux software]# service postgresql start
Starting postgresql service:                [ OK ]
[root@linux software]# su postgres
bash-2.05b$ createdb
CREATE DATABASE
bash-2.05b$ psql
Welcome to psql 7.3.4, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms
       \h for help with SQL commands
       \? for help on internal slash commands
       \g or terminate with semicolon to execute query
       \q to quit

postgres=# \q
bash-2.05b$
bash-2.05b$ vi /var/lib/pgsql/data/postgresql.conf
#-----

#
#           Connection Parameters
#
#tcpip_socket = false
tcpip_socket = true
#ssl = false

#max_connections = 32
max_connections = 128
#superuser_reserved_connections = 2

#port = 5432
#hostname_lookup = false
#show_source_port = false

#unix_socket_directory = "
#unix_socket_group = "
#unix_socket_permissions = 0777 # octal
```

```
#virtual_host = "
```

```
#krb_server_keyfile = "
```

```
#
```

```
# Shared Memory Size
```

```
#
```

```
#shared_buffers = 64 # min max_connections*2 or 16, 8KB each
```

```
shared_buffers = 256 # min max_connections*2 or 16, 8KB each
```

```
#max_fsm_relations = 1000 # min 10, fsm is free space map, ~40 bytes
```

```
#max_fsm_pages = 10000 # min 1000, fsm is free space map, ~6 bytes
```

```
#max_locks_per_transaction = 64 # min 10
```

```
#wal_buffers = 8 # min 4, typically 8KB each
```

```
bash-2.05b$ vi /var/lib/pgsql/data/pg_hba.conf
```

```
host all all 127.0.0.1 255.255.255.255 md5
```

```
bash-2.05b$ psql
```

```
Welcome to psql 7.3.4, the PostgreSQL interactive terminal.
```

```
Type: \copyright for distribution terms
```

```
\h for help with SQL commands
```

```
\? for help on internal slash commands
```

```
\g or terminate with semicolon to execute query
```

```
\q to quit
```

```
postgres=# CREATE USER netkiller WITH PASSWORD 'chen';
```

```
CREATE USER
```

```
postgres=# CREATE DATABASE netkiller WITH OWNER = netkiller TEMPLATE = template0  
ENCODING = 'UNICODE';
```

```
CREATE DATABASE
```

```
postgres=# \du
```

```
List of database users
```

```
User name | User ID | Attributes
```

```
-----+-----+-----
```

```
netkiller | 100 |
```

```
postgres | 1 | superuser, create database
```

```
(2 rows)
```

```
postgres=# \l
```

```
List of databases
```

```
Name | Owner | Encoding
```





```

drwx----- 2 root root 4096 Nov 5 19:14 kde-root
drwx----- 2 root root 16384 Nov 5 18:46 lost+found
drwxr-xr-x 2 root root 4096 Nov 5 18:55 .mozilla
drwx----- 2 root root 4096 Nov 5 11:38 orbit-root
drwxr-xr-x 2 root root 4096 Nov 5 19:14 .qt
-rw----- 1 root root 1024 Nov 5 18:52 .rnd
srwxrwxrwx 1 postgres postgres 0 Nov 11 16:29 .s.PGSQL.5432
-rw----- 1 postgres postgres 26 Nov 11 16:29 .s.PGSQL.5432.lock
-r--r--r-- 1 root root 11 Nov 5 14:49 .X0-lock
drwxrwxrwt 2 root root 4096 Nov 5 14:49 .X11-unix
[root@linux software]# file /tmp/.s.PGSQL.5432
/tmp/.s.PGSQL.5432: socket

```

```

file /tmp/.s.PGSQL.5432 /tmp/.s.PGSQL.5432:
socket /tmp/.s.PGSQL.5432:

```

```

[root@linux software]# psql -h127.0.0.1 -Upostgres db
Password:
psql: FATAL: Password authentication failed for user "postgres"

```

```

[root@linux software]# psql -h127.0.0.1 -Upostgres netkiller
Password:
Password:
Password:
Password:
Password:
Password:
Password:
psql: FATAL: Password authentication failed for user "postgres"

```

3.

```

[root@linux software]# psql -h127.0.0.1 -Unetkiller netkiller
psql: FATAL: No pg_hba.conf entry for host 127.0.0.1, user netkiller, database netkiller
/var/lib/pgsql/data/pg_hba.conf
host all all 127.0.0.1 255.255.255.255 md5

```

## 2.3 APT 安装

Apt Debian Linux

APT

<http://www.rpmfind.net/>

apt

<http://www.rpmfind.net/linux/rpm2html/search.php?query=apt>

```

[root@linux root]# cd /usr/local/src/

```

```
[root@linux src]# wget http://194.199.20.114/linux/freshrpms/redhat/9/apt/apt-0.5.5cnc6-fr1.i386.rpm
[root@linux src]# rpm -ivh apt-0.5.5cnc6-fr1.i386.rpm
warning: apt-0.5.5cnc6-fr1.i386.rpm: V3 DSA signature: NOKEY, key ID e42d547b
Preparing... ##### [100%]
   1:apt ##### [100%]
[root@linux src]# apt-get update
Get:1 http://ayo.freshrpms.net redhat/9/i386 release [1171B]
Fetched 1171B in 6s (170B/s)
Get:1 http://ayo.freshrpms.net redhat/9/i386/os pkglist [1357kB]
Get:2 http://ayo.freshrpms.net redhat/9/i386/os release [140B]
Get:3 http://ayo.freshrpms.net redhat/9/i386/updates pkglist [487kB]
Get:4 http://ayo.freshrpms.net redhat/9/i386/updates release [153B]
Get:5 http://ayo.freshrpms.net redhat/9/i386/freshrpms pkglist [151kB]
Get:6 http://ayo.freshrpms.net redhat/9/i386/freshrpms release [157B]
Fetched 1995kB in 2m41s (12.3kB/s)
Reading Package Lists... Done
Building Dependency Tree... Done
[root@linux src]#
[root@linux src]# apt-get check
Reading Package Lists... Done
Building Dependency Tree... Done
You might want to run `apt-get -f install' to correct these.
The following packages have unmet dependencies:
  postgresql-python: Depends: mx but it is not installed
E: Unmet dependencies. Try using -f.
[root@linux src]#
[root@linux src]# apt-get -f install
Reading Package Lists... Done
Building Dependency Tree... Done
Correcting dependencies... Done
The following extra packages will be installed:
  mx
The following NEW packages will be installed:
  mx
0 packages upgraded, 1 newly installed, 0 removed and 146 not upgraded.
Need to get 609kB of archives.
After unpacking 2266kB of additional disk space will be used.
Do you want to continue? [Y/n]
Get:1 http://ayo.freshrpms.net redhat/9/i386/os mx 2.0.3-8 [609kB]
Fetched 609kB in 2m49s (3596B/s)
Executing RPM (-Uvh)...
Preparing... ##### [100%]
   1:mx ##### [100%]
[root@linux src]#
```

```
[root@linux src]# apt-get install postgresql-server
```

Debian Linux

```
[root@linux src]# apt-get install postgresql-server
```

## 2.4 PostgreSQL 8.0 beta for windows 版本安装

PostgreSQL 8.0 beta windows

<http://laser.dyndns.zhengmai.net.cn/download/postgresql-8.0-beta1.msi>

postgresql

D:\PostgreSQL\8.0-beta1\data\pg\_hba.conf

UltraEdit

host	all	all	192.168.0.1	255.255.255.255	md5
------	-----	-----	-------------	-----------------	-----

Windows XP SP2

```
PostgreSQL 8.0      :  
                    PostgreSQL 8.0-beta1  Start service  
PostgreSQL 8.0      :  
                    PostgreSQL 8.0-beta1  Stop service  
psql                :  
                    PostgreSQL 8.0-beta1  psql to template1
```



```
psql to template1  
Password:  
Welcome to psql.exe 8.0.0beta1, the PostgreSQL interactive terminal.  
Type:  \copyright for distribution terms  
       \h for help with SQL commands  
       \? for help with psql commands  
       \g or terminate with semicolon to execute query  
       \q to quit  
template1=#
```



Password:

Welcome to psql.exe 8.0.0beta1, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms  
      \h for help with SQL commands  
      \? for help with psql commands  
      \g or terminate with semicolon to execute query  
      \q to quit

template1=# \l

      List of databases

Name	Owner	Encoding
netkiller	chen	UNICODE
template0	postgres	UNICODE
template1	postgres	UNICODE

(3 rows)

template1=# \c netkiller chen

Password:

You are now connected to database "netkiller" as user "chen".

netkiller=#

## 2.4.3 ODBC

1.

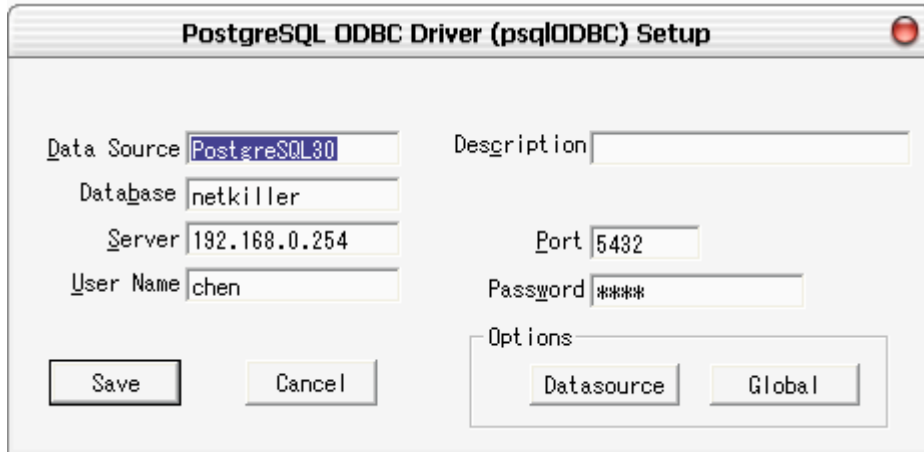
(ODBC)



2. " "



3. PostgreSQL " "



4. Save



5. " "

## 2.4.4 Unix/Linux Windows

```
[root@linux root]# psql -h 192.168.0.254 -U chen netkiller
Password:
Welcome to psql 7.3.4, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms
      \h for help with SQL commands
      \? for help on internal slash commands
      \g or terminate with semicolon to execute query
      \q to quit
```

```
netkiller=#
```

## 2.4.5 Windows                      Unix/Linux

```
D:\PostgreSQL\8.0-beta1\bin>psql.exe -h 192.168.0.1 -U netkiller netkiller
```

```
Password:
```

```
Welcome to psql.exe 8.0.0beta1, the PostgreSQL interactive terminal.
```

```
Type:  \copyright for distribution terms
        \h for help with SQL commands
        \? for help with psql commands
        \g or terminate with semicolon to execute query
        \q to quit
```

```
netkiller=>
```

## 2.5 数据库备份方案

### 2.5.1

tar.gz

```
[root@linux root]# cat backup.sh
#!/bin/bash

FTPHOST=ftp.9812.net
USER=netkiller
PASSWD=xxx

echo "Starting Backup PostgreSQL ... "
#big5 gb2312 gb18030 ...
export PGCLIENTENCODING=gb18030
su - postgres -c pg_dumpall > pgsql-backup.`date +%Y-%m-%d.%H:%M:%S`.dmp
tar zcvf pgsql-backup.`date +%Y-%m-%d`.tar.gz *.dmp

echo "Upload File ..."
ftp -n ${FTPHOST} <<!
user ${USER} ${PASSWD}
binary
prompt
mkdir backup
```



```
cd backup
mput *.tar.gz
close
bye
!
echo "Remove temp file ..."
rm -rf postgresql-backup.*.dmp
rm -rf postgresql-backup.????-??-??*.tar.gz
[root@linux root]#
```

## IP

```
[root@linux root]# cat backup.sh
#!/bin/bash

echo "Starting Backup PostgreSQL ... "
su - postgres -c pg_dumpall > postgresql-backup.`date +%Y-%m-%d.%H:%M:%S`.dmp
tar zcvf postgresql-backup.`date +%Y-%m-%d`.tar.gz *.dmp

echo "Remove temp file ..."
rm -rf postgresql-backup.*.dmp
[root@linux root]#
```

## 2.5.2

```
[root@linux root]# cat getbackup.sh
#!/bin/bash

FTPHOST=ftp.9812.net
USER=netkiller
PASSWD=xxx

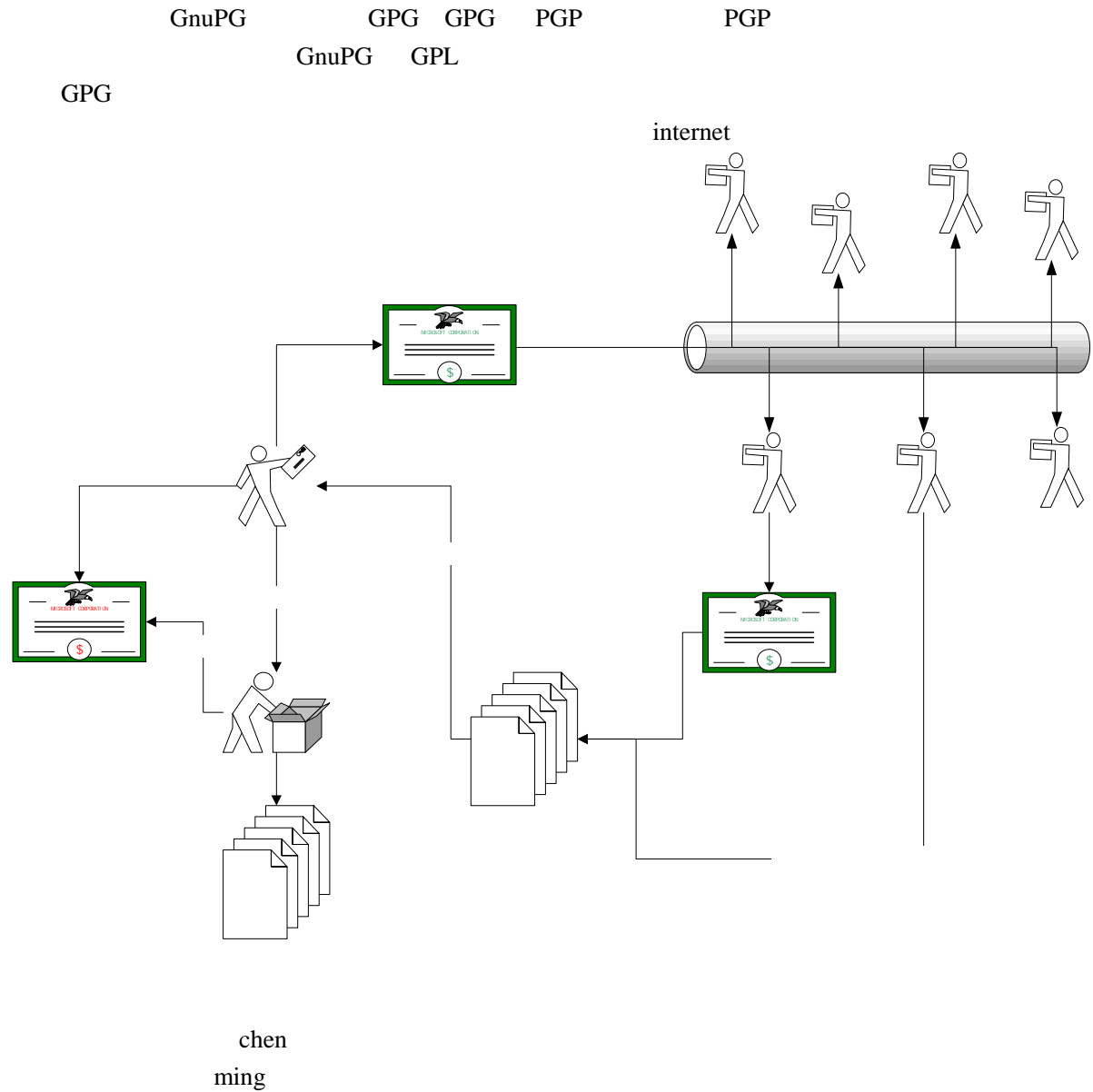
wget ftp://${USER}:${PASSWD}@${FTPHOST}/backup/*

ftp -n ${FTPHOST} <<!
user ${USER} ${PASSWD}
binary
prompt
cd backup
mdelete *
close
bye
!
```

```
[root@linux root]#
```

## 2.5.3 -PGP/GPG

DML) SQL (xxxx.sql DDL



**chen**

1.

```
[chen@linux chen]$ ls -la
total 56
drwx----- 4 chen  chen  4096 Dec 12 20:38 .
```

```

drwxr-xr-x   7 root    root      4096 Nov 12 11:47 ..
-rw-----   1 chen    chen      4953 Dec 10 14:05 .bash_history
-rw-r--r--   1 chen    chen        24 Feb 11  2003 .bash_logout
-rw-r--r--   1 chen    chen      191 Feb 11  2003 .bash_profile
-rw-r--r--   1 chen    chen      124 Feb 11  2003 .bashrc
-rw-r--r--   1 chen    chen     5531 Feb  4  2003 .canna
-rw-r--r--   1 chen    chen      847 Feb 20  2003 .emacs
-rw-r--r--   1 chen    chen      120 Feb 27  2003 .gtkrc
drwxr-xr-x   3 chen    chen      4096 Aug 12  2002 .kde
-rw-----   1 chen    chen      594 Dec 10 09:38 .viminfo
drwxr-xr-x   2 chen    chen      4096 Nov  5 19:16 .xemacs
[chen@linux chen]$

```

2.

## GPG

```
# gpg --gen-key
```

```

[chen@linux chen]$ gpg --gen-key
gpg (GnuPG) 1.2.1; Copyright (C) 2002 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
gpg: /home/chen/.gnupg: directory created
gpg: new configuration file `/home/chen/.gnupg/gpg.conf' created
gpg: keyblock resource `/home/chen/.gnupg/secring.gpg': file open error
gpg: keyring `/home/chen/.gnupg/pubring.gpg' created
Please select what kind of key you want:
  (1) DSA and ElGamal (default)
  (2) DSA (sign only)
  (5) RSA (sign only)
Your selection? █
DSA keypair will have 1024 bits.
About to generate a new ELG-E keypair.
           minimum keysize is  768 bits
           default keysize is 1024 bits
           highest suggested keysize is 2048 bits
What keysize do you want? (1024) █
Requested keysize is 1024 bits
Please specify how long the key should be valid.
  0 = key does not expire
  <n>  = key expires in n days
  <n>w = key expires in n weeks

```



```
[chen@linux chen]$
```

3.

```
[chen@linux chen]$ ls -la
total 52
drwx-----  5 chen  chen  4096 Dec 12 20:47 .
drwxr-xr-x   7 root  root  4096 Dec 12 20:44 ..
-rw-r--r--   1 chen  chen   24 Dec 12 20:44 .bash_logout
-rw-r--r--   1 chen  chen  191 Dec 12 20:44 .bash_profile
-rw-r--r--   1 chen  chen  124 Dec 12 20:44 .bashrc
-rw-r--r--   1 chen  chen 5531 Dec 12 20:44 .canna
-rw-r--r--   1 chen  chen  847 Dec 12 20:44 .emacs
drwx-----  2 chen  chen  4096 Dec 12 20:52 .gnupg
-rw-r--r--   1 chen  chen  120 Dec 12 20:44 .gtkrc
drwxr-xr-x   3 chen  chen  4096 Dec 12 20:44 .kde
-rw-----   1 chen  chen   61 Dec 12 20:45 .Xauthority
drwxr-xr-x   2 chen  chen  4096 Dec 12 20:44 .xemacs
[chen@linux chen]$ ls .gnupg/
gpg.conf  pubring.gpg  pubring.gpg~  random_seed  secring.gpg  trustdb.gpg
```

4.

```
gpg --gen-key
```

```
gpg --output revoke.asc --gen-revoke netkiller netkiller
```

Real name:

```
gpg --output revoke.asc --gen-revoke netkiller@9812.net
```

```
[chen@linux chen]$ gpg --output revoke.asc --gen-revoke netkiller
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information

sec 1024D/B00847C5 2003-12-12 netkiller ( ) <netkiller@9812.net>

Create a revocation certificate for this key? y
Please select the reason for the revocation:
 0 = No reason specified
 1 = Key has been compromised
 2 = Key is superseded
 3 = Key is no longer used
 Q = Cancel
(Probably you want to select 1 here)
Your decision?
Enter an optional description; end it with an empty line:
> :( cancel
>
```

Reason for revocation: Key has been compromised

:( cancel

Is this okay?

Please select the reason for the revocation:

0 = No reason specified

1 = Key has been compromised

2 = Key is superseded

3 = Key is no longer used

Q = Cancel

(Probably you want to select 1 here)

Enter an optional description; end it with an empty line:

> :( cancel

>

Is this okay? **y**

You need a passphrase to unlock the secret key for

user: "netkiller ( ) <netkiller@9812.net>"

1024-bit DSA key, ID B00847C5, created 2003-12-12

ASCII armored output forced.

Revocation certificate created.

Please move it to a medium which you can hide away; if Mallory gets access to this certificate he can use it to make your key unusable.

It is smart to print this certificate and store it away, just in case your media become unreadable. But have some caution: The print system of your machine might store the data and make it available to others!

[chen@linux chen]\$ ls

revoke.asc

[chen@linux chen]\$ cat revoke.asc

-----BEGIN PGP PUBLIC KEY BLOCK-----

Version: GnuPG v1.2.1 (GNU/Linux)

Comment: A revocation certificate should follow

iFIEIBECABIFAj/Zv08LHQI6KCBjYW5jZWwACgkQrAfIc7AIR8X3agCcDBjqRkFx

QUzcZ/1Gyf1/jjFis04An2rYQz2XrCode08Y78Fj63RVNKD9

=ovDh

-----END PGP PUBLIC KEY BLOCK-----

[chen@linux chen]\$

5.

gpg --list-key

[chen@linux chen]\$ gpg --list-key

gpg: WARNING: using insecure memory!

```
gpg: please see http://www.gnupg.org/faq.html for more information
/home/chen/.gnupg/pubring.gpg
```

```
-----
pub 1024D/B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
sub 1024g/0B70F0CB 2003-12-12
```

```
[chen@linux chen]$
```

6.

```
ASCII                               gpg --output netkiller.gpg --armor --export netkiller
gpg --output netkiller.gpg --export netkiller
```

```
ASCII                               BASE64
```

```
[chen@linux chen]$ gpg --output netkiller.gpg --armor --export netkiller
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
[chen@linux chen]$ ls
netkiller.gpg  revoke.asc
[chen@linux chen]$ cat netkiller.gpg
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.2.1 (GNU/Linux)

mQGIBD/ZuhgRBACBRuWYRtJ8+8VmnYUgNy7TS/nVl0sHrsGD2kgIWWUuZyYgKSUoM
vT4MUHWdd52yesovAV61qsVCfUz+O76ovhQRUzv4jp+bkIOKcc7E07Z2MZmc1BqR
+Gavb3gsJM6DmOLcRiU0m3fqod1KCGFf8K6ZLQUhRjYWDI80KEgJqliG4wCgo2xn
5WS1CIGnvGDFUiGY6VhdamsD/jdiqSIcwFt2x6VMjzeWkHHM5wNYHuBJnp9DPd9g
rn3uEq+tSex8ZXRyZHGj+N4SKezhEYal1D762kDxjGYltk5Xce5dXQBn9fulEDhD
OzOp78GvIvJ/m33D/J6xECbXUz8XsFFhxJ6QnVh/RURY+EvHE1Tmz/FRG69Rc1Uc
JBqCA/0faHEkyDv+FWEsmFKjflDNqN5NHtdWzJZQZKD1Vb64oJ5CK6r2l+vmxbBr
fVpfk5OVXnfMSpLKc7aGA9X+mUMuNrGRNzzzsmVK6urWQovL/BfeukMgDBZXkLd8
fO7aA53XeBhmVC49atFPH8hsOemdd0mombrzcvcKczjMp0ThP9rQzbnV0a2lsbGVy
ICjpmYjmma/lS7DnmoTlr4bpbkqUpIDxuZXRraWxsZXJAOTgxMi5uZXQ+iFkEExEC
ABkFAj/ZuhgECwcDAgMVAgMDFgIBAh4BAheAAAoJEKwHyHOwCEfFBqMAN0HoK9Xc
zvzVkfODVZPWUskzwAhqAJ4rbgYEjSN1/CrdUBzTMtecGu9P+7kBDQQ/2boaEAQA
zhoIDY866/GWUUpuarpVKcN1ijn+5M1Pr42vm2Z42ns4PZW3cagHJeIOuJ5R2Aw1
6V4zZwP5PcBScYxQpM0m0bVmTGp/suZmZ6/u3+ADgvJYSxAXdpzP0cL9rVRKqaPa
MKh+HOanAJ9tWcSy6KW83JKG2NS/0U6OSGGDSonLEIMAawUD/iGBjPfXD5jsepg+
Z9J1RefM5/R1nnBEeOROnWyaczIU1okswlyluAthi+2+ijpEULaqSQ+ZjtuBjcMp
kE5UKKq16yBAk2CqJMvkVLIDbPFqbidkAqGp5riKWkc487jR6iZjIAhHvXL0xPIQ
erBmEpi4UT7RlaCAmYwvZ1nxGP3eiEYEGBECAAYFAj/ZuhoACgkQrAfIc7AIR8U0
xACfT5pZ+0YjSp9z0/9jPwDfhw7J1bcAnjqxP+uKfkzDHnXRYfErTN+7iHE
=CII0
-----END PGP PUBLIC KEY BLOCK-----
[chen@linux chen]$
```

```
[chen@linux chen]$ gpg --output bin-netkiller.gpg --export netkiller
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
[chen@linux chen]$ ls
bin-netkiller.gpg  netkiller.gpg  revoke.asc
[chen@linux chen]$
```

## 7. file

```
[chen@linux chen]$ ls
bin-netkiller.gpg  netkiller.gpg  revoke.asc
[chen@linux chen]$ file bin-netkiller.gpg
bin-netkiller.gpg: data
[chen@linux chen]$ file netkiller.gpg
netkiller.gpg: PGP armored data public key block
[chen@linux chen]$ file revoke.asc
revoke.asc: PGP armored data public key block
[chen@linux chen]$
```

## 8.

### mail

```
[chen@linux chen]$ pine
PINE 4.44  MAIN MENU  Folder: INBOX  No Messages

?  HELP          -  Get help using Pine

C  COMPOSE MESSAGE -  Compose and send a message

I  MESSAGE INDEX -  View messages in current folder

L  FOLDER LIST   -  Select a folder to view

A  ADDRESS BOOK  -  Update address book

S  SETUP         -  Configure Pine Options

Q  QUIT          -  Leave the Pine program

Copyright 1989-2002.  PINE is a trademark of the University of Washington.
[Folder "INBOX" opened with 0 messages]
? Help          P PrevCmd      R RelNotes
```



PINE 4.44 COMPOSE MESSAGE

Folder: INBOX No Messages

To : netkiller@9812.net

Cc :

Attchmnt:

Subject :

----- Message Text -----

Attchmnt

^G Get Help ^X Send ^R Rich Hdr ^Y PrvPg/Top ^K Cut Line ^O Postpone  
 ^C Cancel ^D Del Char ^J Attach ^V NxtPg/End ^U UnDel Line ^T To Files

Attchmnt: ^J -&gt; ^T

File to attach:

^G Get Help ^T To Files

^C Cancel TAB Complete

PINE 4.44

BROWSER

Dir: /home/chen

..	(parent dir)	.gnupg	(dir)
.kde	(dir)	mail	(dir)
.xemacs	(dir)	.addressbook	0 B
.addressbook.lu	2.3 KB	.bash_logout	24 B
.bash_profile	191 B	.bashrc	124 B
bin-netkiller.gpg	909 B	.canna	5.5 KB
.emacs	847 B	.gtkrc	120 B
netkiller.gpg	1.3 KB	.pinerc	14 KB
revoke.asc	275 B	.Xauthority	61 B

[ Searched to end of directory ]

? Get Help    E Exit Brwsr                    -    Prev Pg    D Delete            C Copy  
                  S [Select]    W Where is    Spc Next Pg    R Rename            A Add

netkiller.gpg

Attachment comment: my netkiller.gpg file

^G Get Help

^C Cancel

PINE 4.44    COMPOSE MESSAGE

Folder: INBOX    No Messages

To            : netkiller@9812.net

Cc            :

Attchmnt: 1. /home/chen/netkiller.gpg (1.3 KB) "my netkiller.gpg file"

Subject : my netkiller.gpg file

----- Message Text -----

<http://linux.9812.net>

email:netkiller@9812.net

[File /home/chen/netkiller.gpg attached as type TEXT/PLAIN]

^G Get Help    ^X Send            ^R Rich Hdr    ^Y PrvPg/Top    ^K Cut Line    ^O Postpone

^C Cancel      ^D Del Char    ^J Attach      ^V NxtPg/End    ^U UnDel Line ^T To Files

Send message?y

? Help            Y [Yes]

^C Cancel        N No

y

PINE 4.44    MAIN MENU

Folder: INBOX    No Messages

PINE 4.44    COMPOSE MESSAGE

Folder: INBOX    No Messages

To            : netkiller@9812.net

Cc            :

Attchmnt: 1. /home/chen/netkiller.gpg (1.3 KB) "my netkiller.gpg file"

Subject :

----- Message Text -----

Attchmnt

[Sending mail | 0% ]]

PINE 4.44 MAIN MENU

Folder: INBOX No Messages

- ? HELP - Get help using Pine
- C COMPOSE MESSAGE - Compose and send a message
- I MESSAGE INDEX - View messages in current folder
- L FOLDER LIST - Select a folder to view
- A ADDRESS BOOK - Update address book
- S SETUP - Configure Pine Options
- Q QUIT - Leave the Pine program

Copyright 1989-2002. PINE is a trademark of the University of Washington.

[Message sent and copied to "sent-mail".]

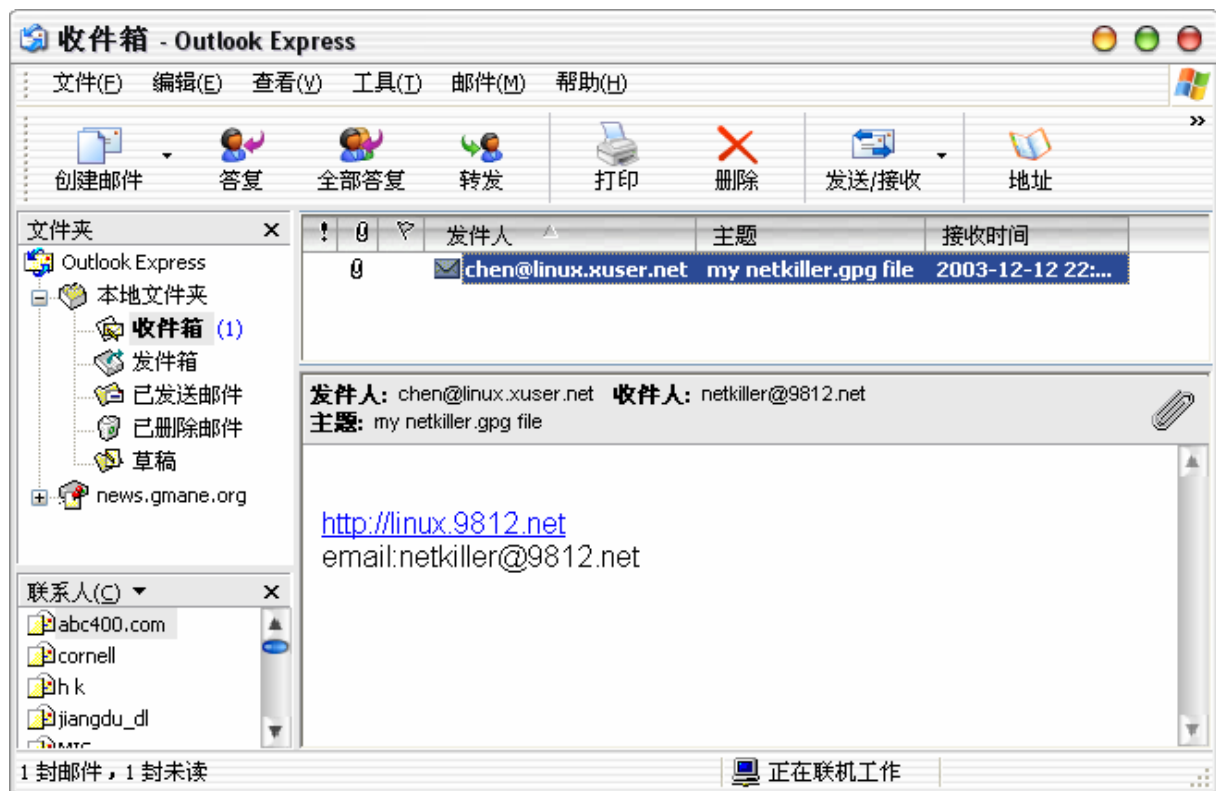
? Help

P PrevCmd

R RelNotes

O OTHER CMDS > [ListFldrs] N NextCmd

K KBLock



9.

```
[chen@linux chen]$ cp netkiller.gpg /tmp
```

## ming

10.

```
[ming@linux ming]$ cp /tmp/netkiller.gpg .
[ming@linux ming]$ ls
netkiller.gpg
[ming@linux ming]$
```

11.

```
[ming@linux ming]$ gpg --import netkiller.gpg
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
gpg: /home/ming/.gnupg: directory created
gpg: new configuration file `/home/ming/.gnupg/gpg.conf' created
gpg: keyblock resource `/home/ming/.gnupg/secring.gpg': file open error
gpg: keyring `/home/ming/.gnupg/pubring.gpg' created
gpg: /home/ming/.gnupg/trustdb.gpg: trustdb created
gpg: key B00847C5: public key "netkiller (䄀x9\x8䄀x9䄀 䄀x9\x8䄀x8䄀x9䄀) <netkiller@9812.net>"
imported
gpg: Total number processed: 1
```

```
gpg: imported: 1
[ming@linux ming]$ gpg --list-key
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
/home/ming/.gnupg/pubring.gpg
-----
pub 1024D/B00847C5 2003-12-12 netkiller ( ) <netkiller@9812.net>
sub 1024g/0B70F0CB 2003-12-12

[ming@linux ming]$
```

12.

```
[ming@linux ming]$ gpg --fingerprint netkiller
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
pub 1024D/B00847C5 2003-12-12 netkiller ( ) <netkiller@9812.net>
    Key fingerprint = 0058 5847 7598 556F AAFD 81A5 AC07 C873 B008 47C5
sub 1024g/0B70F0CB 2003-12-12

[ming@linux ming]$
```

13.

(gpg --sign-key netkiller)

```
[ming@linux ming]$ gpg --gen-key
gpg (GnuPG) 1.2.1; Copyright (C) 2002 Free Software Foundation, Inc.
This program comes with ABSOLUTELY NO WARRANTY.
This is free software, and you are welcome to redistribute it
under certain conditions. See the file COPYING for details.

gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
Please select what kind of key you want:
  (1) DSA and ElGamal (default)
  (2) DSA (sign only)
  (5) RSA (sign only)
Your selection?
DSA keypair will have 1024 bits.
About to generate a new ELG-E keypair.
    minimum keysize is 768 bits
    default keysize is 1024 bits
    highest suggested keysize is 2048 bits
What keysize do you want? (1024)
```



```
[ming@linux ming]$ gpg --sign-key netkiller
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information

pub 1024D/B00847C5  created: 2003-12-12 expires: never      trust: -/-
sub 1024g/0B70F0CB  created: 2003-12-12 expires: never
(1). netkiller (          ) <netkiller@9812.net>

pub 1024D/B00847C5  created: 2003-12-12 expires: never      trust: -/-
Primary key fingerprint: 0058 5847 7598 556F AAFD  81A5 AC07 C873 B008 47C5

netkiller (          ) <netkiller@9812.net>
```

How carefully have you verified the key you are about to sign actually belongs to the person named above? If you don't know what to answer, enter "0".

- (0) I will not answer. (default)
- (1) I have not checked at all.
- (2) I have done casual checking.
- (3) I have done very careful checking.

Your selection? 3

Are you really sure that you want to sign this key with your key: "mings (I am ming) <mings@9812.net>"

I have checked this key very carefully.

Really sign? y

You need a passphrase to unlock the secret key for user: "mings (I am ming) <mings@9812.net>"  
1024-bit DSA key, ID 3D9CE6DF, created 2003-12-12

Enter passphrase: mings

```
[ming@linux ming]$
[ming@linux ming]$ gpg --list-key
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
/home/ming/.gnupg/pubring.gpg
-----
pub 1024D/B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
sub 1024g/0B70F0CB 2003-12-12
```

```
pub 1024D/3D9CE6DF 2003-12-12 mings (I am ming) <mings@9812.net>
sub 1024g/510C2A18 2003-12-12
```

```
[ming@linux ming]$
```

14.

```
[chen@linux chen]$ gpg --check-sigs netkiller
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
pub 1024D/B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
sig!3      B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
sub 1024g/0B70F0CB 2003-12-12
sig!      B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
[ming@linux ming]$ gpg --check-sigs netkiller
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
pub 1024D/B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
sig!3      B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
sig!      79F1102B 2003-12-12 mings (I am mings) <mings@9812.net>
sub 1024g/0B70F0CB 2003-12-12
sig!      B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
```

```
[ming@linux ming]$
```

```
[chen@linux chen]$
```

```
[ming@linux ming]$ gpg --check-sigs netkiller
gpg: WARNING: using insecure memory!
gpg: please see http://www.gnupg.org/faq.html for more information
pub 1024D/B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
sig!3      B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
sub 1024g/0B70F0CB 2003-12-12
sig!      B00847C5 2003-12-12 netkiller (          ) <netkiller@9812.net>
```

```
[ming@linux ming]$
```

15.

```
[ming@linux ming]$ pg_dump -Unetkiller -h127.0.0.1 >pgsql-dump.sql
```

```
Password:
```

```
[ming@linux ming]$
```

```
[ming@linux ming]$ gpg -sear netkiller pgsql-dump.sql
```



```
gpg: WARNING: using insecure memory!  
gpg: please see http://www.gnupg.org/faq.html for more information
```

```
You need a passphrase to unlock the secret key for  
user: "mings (I am mings) <mings@9812.net>"  
1024-bit DSA key, ID 79F1102B, created 2003-12-12
```

```
gpg: checking the trustdb  
gpg: checking at depth 0 signed=1 ot(-/q/n/m/f/u)=0/0/0/0/1  
gpg: checking at depth 1 signed=0 ot(-/q/n/m/f/u)=1/0/0/0/0  
[ming@linux ming]$
```

```
[ming@linux ming]$ gpg -ear netkiller pgsq-dump.sql  
gpg: WARNING: using insecure memory!  
gpg: please see http://www.gnupg.org/faq.html for more information  
[ming@linux ming]$ ls  
netkiller.gpg  pgsq-dump.sql  pgsq-dump.sql.asc  
[ming@linux ming]$
```

```
pgsq-dump.sql.asc  chen ( WEB/FTP  
                  )  
chen
```

```
[chen@linux chen]$ gpg -d pgsq-dump.sql.asc > pgsq-dump.sql  
gpg: WARNING: using insecure memory!  
gpg: please see http://www.gnupg.org/faq.html for more information  
  
You need a passphrase to unlock the secret key for  
user: "netkiller ( ) <netkiller@9812.net>"  
1024-bit ELG-E key, ID 0B70F0CB, created 2003-12-12 (main key ID B00847C5)  
  
Enter passphrase:  
gpg: encrypted with 1024-bit ELG-E key, ID 0B70F0CB, created 2003-12-12  
"netkiller ( ) <netkiller@9812.net>"  
[chen@linux chen]$
```

## 2.6 备份计划

### 2.6.1

```
[root@linux etc]# cat crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
0 3 * * * root /usr/local/backup/backup.sh
```

## 2.6.2

4 00

4 00

3:00

4 00

```
[root@linux etc]# cat crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
0 4 * * * root /usr/local/backup/getbackup.sh
```

## 2.7 数据恢复

```
[root@linux root]# su postgres
bash-2.05b$ psql member -f pgsq1-backup.xxxx-xx-xx.xx:xx:xx.dmp
```

## 2.8 性能提升

### 2.8.1

2.2	SHMMAX	SHMALL	32 MB	proc
		128 MB		

1

```
# echo 134217728 >/proc/sys/kernel/shmall
# echo 134217728 >/proc/sys/kernel/shmmax
```

```
[root@linux root]# cat /proc/sys/kernel/shmall
2097152
[root@linux root]# cat /proc/sys/kernel/shmmax
33554432
[root@linux root]# echo 134217728 >/proc/sys/kernel/shmall
[root@linux root]# echo 134217728 >/proc/sys/kernel/shmmax
[root@linux root]# cat /proc/sys/kernel/shmall
134217728
[root@linux root]# cat /proc/sys/kernel/shmmax
134217728
```

rc.local

```
[root@linux root]# cat /etc/rc.d/rc.local
#!/bin/sh
#
# This script will be executed *after* all the other init scripts.
# You can put your own initialization stuff in here if you don't
# want to do the full Sys V style init stuff.

touch /var/lock/subsys/local
/usr/local/jakarta-tomcat/bin/startup.sh
/usr/local/apache/bin/apachectl start
echo 134217728 >/proc/sys/kernel/shmall
echo 134217728 >/proc/sys/kernel/shmmax
```

2 sysctl

```
[root@linux root]# sysctl -w kernel.shmall=134217728
kernel.shmall = 134217728
[root@linux root]# sysctl -w kernel.shmmax=134217728
kernel.shmmax = 134217728
[root@linux root]#
```

3 /etc/sysctl.conf

```
kernel.shmall = 134217728
```

```
kernel.shmmax = 134217728
```

```
[root@linux root]# cat /etc/sysctl.conf
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled.  See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

# Controls the System Request debugging functionality of the kernel
kernel.sysrq = 0

# Controls whether core dumps will append the PID to the core filename.
# Useful for debugging multi-threaded applications.
kernel.core_uses_pid = 1
kernel.shmall = 134217728
kernel.shmmax = 134217728
```

sysctl

## 2.8.2

32

```
max_connections = 100
shared_buffers = 200
shared_buffers = max_connections*2
```

```
[root@linux data]# cat postgresql.conf
#
# PostgreSQL configuration file
# -----
#
# This file consists of lines of the form:
#
#   name = value
#
# (The '=' is optional.) White space may be used. Comments are introduced
# with '#' anywhere on a line. The complete list of option names and
# allowed values can be found in the PostgreSQL documentation. The
# commented-out settings shown in this file represent the default values.
#
# Any option can also be given as a command line switch to the
```

```
# postmaster, e.g. 'postmaster -c log_connections=on'. Some options
# can be changed at run-time with the 'SET' SQL command.
#
# This file is read on postmaster startup and when the postmaster
# receives a SIGHUP. If you edit the file on a running system, you have
# to SIGHUP the postmaster for the changes to take effect, or use
# "pg_ctl reload".

#=====

#
#           Connection Parameters
#
#tcpip_socket = false

tcpip_socket = true

#ssl = false
#ssl = true

#max_connections = 32
max_connections = 100
#superuser_reserved_connections = 2

#port = 5432
#hostname_lookup = false
#show_source_port = false

#unix_socket_directory = "
#unix_socket_group = "
#unix_socket_permissions = 0777 # octal

#virtual_host = "

#krb_server_keyfile = "

#
#           Shared Memory Size
#
#shared_buffers = 64           # min max_connections*2 or 16, 8KB each
shared_buffers = 200         # min max_connections*2 or 16, 8KB each
```

```
#max_fsm_relations = 1000      # min 10, fsm is free space map, ~40 bytes
#max_fsm_pages = 10000        # min 1000, fsm is free space map, ~6 bytes
#max_locks_per_transaction = 64 # min 10
#wal_buffers = 8              # min 4, typically 8KB each
```

```
#
```

```
# Non-sharef(_re)14.4(i2.7(8)4. 2.7(o)-8.7(ry(8)4.95(Siz0 -1.4829 Tc(# )Tj)TjT**0.0004 Tc02451 Tw[(#w)4.2(
```

```
#default_statistics_target = 10 # range 1-1000

#
#       GEQO Optimizer Parameters
#
#geqo = true
#geqo_selection_bias = 2.0      # range 1.5-2.0
#geqo_threshold = 11
#geqo_pool_size = 0            # default based on tables in statement,
                                # range 128-1024
#geqo_effort = 1
#geqo_generations = 0
#geqo_random_seed = -1        # auto-compute seed

#
#       Message display
#
#server_min_messages = notice # Values, in order of decreasing detail:
                                #   debug5, debug4, debug3, debug2, debug1,
                                #   info, notice, warning, error, log, fatal,
                                #   panic
#client_min_messages = notice # Values, in order of decreasing detail:
                                #   debug5, debug4, debug3, debug2, debug1,
                                #   log, info, notice, warning, error

#silent_mode = false

#log_connections = false
#log_pid = false
#log_statement = false
#log_duration = false
#log_timestamp = false

#log_min_error_statement = panic # Values in order of increasing severity:
                                #   debug5, debug4, debug3, debug2, debug1,
                                #   info, notice, warning, error, panic(off)

#debug_print_parse = false
#debug_print_rewritten = false
#debug_print_plan = false
#debug_pretty_print = false

#explain_pretty_print = true
```

```
# requires USE_ASSERT_CHECKING
#debug_assertions = true

#
#       Syslog
#
#syslog = 0                # range 0-2
#syslog_facility = 'LOCAL0'
#syslog_ident = 'postgres'

#
#       Statistics
#
#show_parser_stats = false
#show_planner_stats = false
#show_executor_stats = false
#show_statement_stats = false

# requires BTREE_BUILD_STATS
#show_btree_build_stats = false

#
#       Access statistics collection
#
#stats_start_collector = true
#stats_reset_on_server_start = true
#stats_command_string = false
#stats_row_level = false
#stats_block_level = false

#
#       Lock Tracing
#
#trace_notify = false

# requires LOCK_DEBUG
#trace_locks = false
#trace_userlocks = false
#trace_lwlocks = false
#debug_deadlocks = false
```



```
#trace_lock_oidmin = 16384
#trace_lock_table = 0

#
#      Misc
#
#autocommit = true
#dynamic_library_path = '$libdir'
#search_path = '$user,public'
#datestyle = 'iso, us'
#timezone = unknown          # actually, defaults to TZ environment setting
#austalian_timezones = false
#client_encoding = sql_ascii # actually, defaults to database encoding
#authentication_timeout = 60 # 1-600, in seconds
#deadlock_timeout = 1000    # in milliseconds
#default_transaction_isolation = 'read committed'
#max_expr_depth = 10000     # min 10
#max_files_per_process = 1000 # min 25
#password_encryption = true
#sql_inheritance = true
#transform_null_equals = false
#statement_timeout = 0      # 0 is disabled, in milliseconds
#db_user_namespace = false

#
#      Locale settings
#
# (initialized by initdb -- may be changed)
LC_MESSAGES = 'en_US.UTF-8'
LC_MONETARY = 'en_US.UTF-8'
LC_NUMERIC = 'en_US.UTF-8'
LC_TIME = 'en_US.UTF-8'
```

```
[root@linux data]# service postgresql restart
```

```
[ OK ]
```

```
Starting postgresql service:
```

```
[ OK ]
```

```
[root@linux root]# psql -Uchen member
```

Welcome to psql 7.3.3, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms  
      \h for help with SQL commands  
      \? for help on internal slash commands  
      \g or terminate with semicolon to execute query  
      \q to quit

```
member=> show max_connections;
max_connections
```

```
-----
100
(1 row)
```

## 2.8.3 vacuumdb

2 00      3 00      4 00

```
[root@linux etc]# cat crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/

# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
0 2 * * * root /usr/local/pgsql/optimize.sh
```

vacuumdb

```
[root@linux pgsql]# cat optimize.sh
#!/bin/bash
vacuumdb -hlocalhost -p5432 -Upostgres -a -f -z
[root@linux pgsql]#
```

## 2.8.4

1.

Insert

10000

10000

	begin; insert into .....
--	-----------------------------

	<pre> insert into ..... insert into ..... ..... 1000  insert into ..... insert into ..... commit; </pre>
	<pre> begin; insert into ..... ..... 1000  insert into ..... insert into ..... commit; </pre>
	<pre> begin; insert into ..... ..... 1000  insert into ..... insert into ..... commit; </pre>

2. copy from

<pre> pg_copy_from (PHP 4 &gt;= 4.2.0)  pg_copy_from --  bool pg_copy_from ( resource connection, string table_name, array rows [, string delimiter [, string null_as]]) pg_copy_from()      rows          COPY FROM SQL                     TRUE          FALSE pg_copy_to() </pre>
--

3.

`vacuumdb -hlocalhost -p5432 -Upostgres -a -f -z`

## 2.8.5

1.

PC	SATA		
	ATA		
	ATA 66 (5400rpm)	RAID 0	ATA100(7200rpm)
RAID		RAID	RAID0

2.

S

hotswap

RAID5

1

RAID	RAID 0	RAID 1	RAID 3	RAID 5
		50%		
		50%	n-1 /n n	n-1 /n n

2

RAID	RAID 10	RAID 30	RAID 50
2	4	6,8,10,12,14 16	6,8,10,12,14 16
3	6	9,12 15	9,12 15
4	8	12 16	12 16
	50%		

	50%	n-2/2	n-2/n
		n	n

3.

## 2.8.6

hdparm

ATA

```
[root@linux root]# hdparm
```

```
hdparm - get/set hard disk parameters - version v5.2
```

```
Usage: hdparm [options] [device] ..
```

Options:

- a get/set fs readahead
- A set drive read-lookahead flag (0/1)
- b get/set bus state (0 == off, 1 == on, 2 == tristate)
- B set Advanced Power Management setting (1-255)
- c get/set IDE 32-bit IO setting
- C check IDE power mode status
- d get/set using\_dma flag
- D enable/disable drive defect-mgmt
- E set cd-rom drive speed
- f flush buffer cache for device on exit
- g display drive geometry
- h display terse usage information
- i display drive identification
- I detailed/current information directly from drive
- Istdin similar to -I, but wants /proc/ide/\*/hd?/identify as input
- k get/set keep\_settings\_over\_reset flag (0/1)
- K set drive keep\_features\_over\_reset flag (0/1)
- L set drive doorlock (0/1) (removable harddisks only)
- M get/set acoustic management (0-254, 128: quiet, 254: fast) (EXPERIMENTAL)
- m get/set multiple sector count
- n get/set ignore-write-errors flag (0/1)
- p set PIO mode on IDE interface chipset (0,1,2,3,4,...)
- P set drive prefetch count
- q change next setting quietly
- Q get/set DMA tagged-queuing depth (if supported)
- r get/set readonly flag (DANGEROUS to set)
- R register an IDE interface (DANGEROUS)
- S set standby (spindown) timeout

- t perform device read timings
- T perform cache read timings
- u get/set unmaskirq flag (0/1)
- U un-register an IDE interface (DANGEROUS)
- v defaults; same as -mcudkrag for IDE drives
- V display program version and exit immediately
- w perform device reset (DANGEROUS)
- W set drive write-caching flag (0/1) (DANGEROUS)
- x tristate device for hotswap (0/1) (DANGEROUS)
- X set IDE xfer mode (DANGEROUS)
- y put IDE drive in standby mode
- Y put IDE drive to sleep
- Z disable Seagate auto-powersaving mode
- z re-read partition table

/dev/hda                  cache

```
[root@linux root]# hdparm -Tt /dev/hda
```

```
/dev/hda:
```

```
Timing buffer-cache reads: 128 MB in 0.26 seconds =492.31 MB/sec
```

```
Timing buffered disk reads: 64 MB in 2.28 seconds = 28.07 MB/sec
```

```
[root@linux root]#
```

DMA

```
[root@linux root]# hdparm -d0 /dev/hda
```

```
/dev/hda:
```

```
setting using_dma to 0 (off)
```

```
using_dma = 0 (off)
```

```
[root@linux root]#
```

DMA                  DMA 66                  32

```
[root@linux root]# hdparm -d1 -c3 -X66 /dev/hda
```

```
/dev/hda:
```

```
setting 32-bit IO_support flag to 3
```

```
setting using_dma to 1 (on)
```

```
setting xfermode to 66 (UltraDMA mode2)
```

```
IO_support = 3 (32-bit w/sync)
```

```
using_dma = 1 (on)
```

```
[root@linux root]#
```

```
[root@linux root]# hdparm -d1 -c3 -X100 /dev/hda
```

```
/dev/hda:
setting 32-bit IO_support flag to 3
setting using_dma to 1 (on)
setting xfermode to 100 (unknown, probably not valid)
HDIO_DRIVE_CMD(setxfermode) failed: Input/output error
IO_support   = 3 (32-bit w/sync)
using_dma    = 1 (on)
[root@linux root]#
```

```
PC      A
CPU P4 2.6G
      512MB
      70GB  ATA133
```

```
[root@linux root]# hdparm -d1 -Tt -c3 /dev/hda
/dev/hda:
setting 32-bit IO_support flag to 3
setting using_dma to 1 (on)
IO_support   = 3 (32-bit w/sync)
using_dma    = 1 (on)
Timing buffer-cache reads: 128 MB in 0.25 seconds =512.00 MB/sec
Timing buffered disk reads: 64 MB in 2.28 seconds = 28.07 MB/sec
[root@linux root]#
```

```
PC      B
CPU P4 2.4G
      1GB
ATA RAID 0 120G ATA100 *2
```

```
[root@linux root]# hdparm -d1 -Tt -c3 /dev/sda

/dev/sda:
operation not supported on SCSI disks
[root@linux root]#

[root@linux root]# hdparm -Tt /dev/sda

/dev/sda:
Timing buffer-cache reads: 128 MB in 0.33 seconds =392.46 MB/sec
Timing buffered disk reads: 64 MB in 1.00 seconds = 63.87 MB/sec
[root@linux root]# hdparm -Tt /dev/sda

/dev/sda:
Timing buffer-cache reads: 128 MB in 0.33 seconds =392.46 MB/sec
Timing buffered disk reads: 64 MB in 1.02 seconds = 62.77 MB/sec
[root@linux root]#
```

PC            A            B                            ATA100    RAID0  
ATA133                            4            RAID 0                            A

## 2.9 安全的 TCP/IP 联接

### 2.9.1        SSL                            TCP/IP

1.

```
[root@linux8 root]# su - postgres
-bash-2.05b$ ls
data  initdb.i18n
-bash-2.05b$ cd data/
-bash-2.05b$ ls
base      pg_clog      pg_ident.conf  pg_xlog      postmaster.opts
global   pg_hba.conf  PG_VERSION     postgresql.conf  postmaster.pid
-bash-2.05b$ openssl req -new -text -out server.req
Using configuration from /usr/share/ssl/openssl.cnf
Generating a 1024 bit RSA private key
....++++++
.....++++++
writing new private key to 'privkey.pem'
Enter PEM pass phrase:
Verifying password - Enter PEM pass phrase:
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [GB]:CN
State or Province Name (full name) [Berkshire]:Guang Zhou
Locality Name (eg, city) [Newbury]:Shen Zhen
Organization Name (eg, company) [My Company Ltd]:Open Source Organization
Organizational Unit Name (eg, section) []:technical
Common Name (eg, your name or your server's hostname) []:www.9812.net
Email Address []:netkiller@9812.net

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:chen
An optional company name []:netkiller
```



```
-bash-2.05b$ ls
base    pg_clog      pg_ident.conf  pg_xlog          postmaster.opts  privkey.pem
global  pg_hba.conf  PG_VERSION     postgresql.conf  postmaster.pid   server.req
-bash-2.05b$
```

server.req

```
-bash-2.05b$ cat server.req
Certificate Request:
  Data:
    Version: 0 (0x0)
    Subject: C=CN, ST=Guang Zhou, L=Shen Zhen, O=Open Source Organization, OU=technical,
CN=www.9812.net/Email=netkiller@9812.net
    Subject Public Key Info:
      Public Key Algorithm: rsaEncryption
      RSA Public Key: (1024 bit)
        Modulus (1024 bit):
          00:a5:30:9a:ef:75:9f:40:40:ee:90:4e:06:f7:f7:
          0b:de:97:d0:1a:2e:48:ef:4c:7b:c2:cd:f2:f4:30:
          1b:f4:c7:9d:65:7a:53:d7:d7:7c:ea:25:8f:be:b0:
          57:f5:89:91:2e:80:4c:ff:f1:96:1e:42:06:01:64:
          9f:98:69:24:c1:7f:e6:0c:a5:ae:b9:9c:4c:29:db:
          a3:a3:3d:76:da:89:c0:33:29:c5:a5:8b:7a:e1:e5:
          f4:3b:f3:7d:54:d4:65:fa:c8:c0:1c:11:07:1c:24:
          03:8e:f0:61:d9:70:cf:fa:dd:e2:04:4a:31:c2:63:
          2a:5f:44:ec:48:68:30:44:8d
        Exponent: 65537 (0x10001)
    Attributes:
      challengePassword      :chen
      unstructuredName      :netkiller
    Signature Algorithm: md5WithRSAEncryption
      09:4a:1c:e5:87:7a:9c:6f:69:ed:cd:11:8d:b6:bc:da:e0:4a:
      f5:7a:33:70:0d:5f:28:63:82:79:39:6b:a5:ae:02:7b:87:cb:
      86:74:2e:2b:eb:ec:23:3b:dc:02:25:29:02:74:e7:92:76:ed:
      34:e1:63:e9:ef:dc:12:33:31:84:31:ce:b3:d4:f2:49:92:a5:
      2c:5e:0a:3d:73:f8:1f:95:8f:71:f9:2d:ee:eb:4a:9c:8c:13:
      a5:26:a2:d2:49:c3:7e:69:c7:1b:73:bb:59:8d:9c:bf:dd:ac:
      4b:c4:41:02:b1:3c:a6:c9:c9:eb:00:b3:75:2d:e2:ab:29:b3:
      85:75
-----BEGIN CERTIFICATE REQUEST-----
MIICFzCCAYACAQAwgacxCzAJBgNVBAYTAkNOMRMwEQYDVQQIEWpHdWVuzYBaaG91
MRIwEAYDVQQHEwITaGVuIFpoZW4xITAfBgNVBAoTGE9wZW4gU291cmNIIe9yZ2Fu
aXphdGlvbjESMBAGA1UECXMJdGVjaG5pY2FsMRUwEwYDVQQDEwx3d3cuOTgxMi5u
ZXQxITAfBgkqhkiG9w0BCQEWEm5ldGtpbGxlckA5ODEyLm5ldDCBnzANBgkqhkiG
9w0BAQEFAAOBjQAwgYkCgYEApTCa73WfQEDuke4G9/cL3pfQGi5I70x7ws3y9DAb
```

```
9MedZXpT19d86iWPvrBX9YmRLoBM//GWHkIGAWSfmGkkwX/mDKWuuZxMKdujoz12
2onAMynFpYt64eX00/N9VNRI+sjAHBEHHCQDjvBh2XDP+t3iBEoxwmMqX0TsSGgw
RIOCAwEAAaAvMBMGCSqGSIb3DQEJBzEGEWrjaGVuMBGCSqGSIb3DQEJAjELEwlu
ZXRraWxsZXIwDQYJKoZIhvcNAQEEBQADgYEACUoc5Yd6nG9p7c0Rjba82uBK9Xoz
cA1fKGOCeTlrpa4Ce4fLhnQuK+vsIzvcAiUpAnTnknbtNOFj6e/cEjMxhDHOs9Ty
SZKILF4KPXP4H5WPcfkt7utKnIwTpSai0knDfmmHG3O7WY2cv92sS8RBArE8psnJ
6wCzdS3iqymzhXU=
-----END CERTIFICATE REQUEST-----
```

2.

```
-bash-2.05b$ openssl rsa -in privkey.pem -out server.key
read RSA key
Enter PEM pass phrase:
writing RSA key
-bash-2.05b$ ls
base      pg_clog      pg_ident.conf  pg_xlog          postmaster.opts  privkey.pem  server.req
global    pg_hba.conf  PG_VERSION     postgresql.conf  postmaster.pid   server.key
-bash-2.05b$
```

privkey.pem server.key

```
-bash-2.05b$ cat privkey.pem
-----BEGIN RSA PRIVATE KEY-----
Proc-Type: 4,ENCRYPTED
DEK-Info: DES-EDE3-CBC,EE59B06E786A2FCA

C8RnlMX5tF7CRdx/jxHk/2D4SUu+PVNfphwDbsytmUJIx5qMQAHxCy+NdIDZX9L/
AWIwaShdwFOaP6CMwrzBav54DW1/IibF688X3DA6xUY1+ZvV4RU4t1O6EhEPINth
1KBqgtSw8lnu6HQa6aIFvZ4f/Wqluk04ylGe4CLLW1xPQ36ntw3tFXPm3eIFl3wQ
INxYjNTxSjA9x5IBzyJpaJk27f+/WJARdkFKOwUn9J71IPC5yYybv6IG65xFg6/
kpLqfzx/wAaJxReB/EP95jLVkEmzyi6rqzsbLLgAl6mxGGN5kT34lfK4v9xuRWRz
J2AIBJnloq8NTE48N2g7N1UqHI0r3nNkLdfYEeq6th7d12hiSACGECvSfhlWirsx
sFYcrAhBGCK+4OXjn717AYeAYw+/JPrX0ZuDVFogVKNB9x/S15+y8yh2AgIUjpJ/
BOZ3LCxXyFznu4yBvxNoTOJT2xWuAXV5k5AI3UftOfBAvRZdayAwh6LdoNG77ead1
hNwIAvS5LUIlG8KeAbQHlJuh51YCpmEBCsTqrZybMNoEAiCg0Gn/5tE5cfVmH3Ei
LjhCTtRJ6oGx6dsYaY4A1Jt1+B8DMNnRTez8NN5D2+4wasr4dTYwsRXRyqMCPZJH
+z8m6zautVoHlhGQhRxO4ZcBunyJrdW5XQBGFacUbp1xORCvqP+SW8Z4wDyu2Sk+
MlxPL1T4P3xEANG7hOlsabBiQ2kyCq1iiJCHBlfXxIm86c1ffRYTrdB+PoFyyaII
ErS68kMbv+Y5Tr+X3MI1AMNEEU5YAn/O1wSoL5Cz0nIpKeknKAl/vA==
-----END RSA PRIVATE KEY-----
```

```
-bash-2.05b$ cat server.key
-----BEGIN RSA PRIVATE KEY-----
MIICXAIBAAKBgQCIMJrvdZ9AQO6QTgb39wvel9AaLkqvTHvCzfL0MBv0x51leIPX
13zqJY++sFflIZEugEz/8ZYeQgYBZJ+YaSTBf+YMpa65nEwp26OjPXbaicAzKcWl
```



00:a5:30:9a:ef:75:9f:40:40:ee:90:4e:06:f7:f7:  
0b:de:97:d0:1a:2e:48:ef:4c:7b:c2:cd:f2:f4:30:  
1b:f4:c7:9d:65:7a:53:d7:d7:7c:ea:25:8f:be:b0:  
57:f5:89:91:2e:80:4c:ff:f1:96:1e:42:06:01:64:  
9f:98:69:24:c1:7f:e6:0c:a5:ae:b9:9c:4c:29:db:  
a3:a3:3d:76:da:89:c0:33:29:c5:a5:8b:7a:e1:e5:  
f4:3b:f3:7d:54:d4:65:fa:c8:c0:1c:11:07:1c:24:  
03:8e:f0:61:d9:70:cf:fa:dd:e2:04:4a:31:c2:63:  
2a:5f:44:ec:48:68:30:44:8d

Exponent: 65537 (0x10001)

X509v3 extensions:

X509v3 Subject Key Identifier:

93:4F:D5:41:4A:CA:A2:83:19:C3:5D:BE:58:E6:45:70:7E:95:A5:0A

X509v3 Authority Key Identifier:

keyid:93:4F:D5:41:4A:CA:A2:83:19:C3:5D:BE:58:E6:45:70:7E:95:A5:0A

DirName:/C=CN/ST=Guang Zhou/L=Shen Zhen/O=Open Source

Organization/OU=technical/CN=www.9812.net/Email=netkiller@9812.net  
serial:00

X509v3 Basic Constraints:

CA:TRUE

Signature Algorithm: md5WithRSAEncryption

3f:f1:99:89:37:ec:1b:80:e2:c6:3a:8e:ed:e8:94:b8:70:10:  
34:1c:9a:ef:f7:be:b7:05:51:f4:a2:cb:03:4e:f4:dd:6f:73:  
51:49:d2:91:fc:eb:40:3c:30:54:b6:f0:aa:a1:e8:d4:33:b2:  
9b:d0:0e:0d:b4:4b:65:c5:ae:bf:ed:fa:ff:1c:e6:1d:aa:41:  
4f:da:76:7a:57:7d:8d:f5:1b:17:65:fc:63:4f:db:dd:45:33:  
3d:e7:c9:dd:e8:d6:8d:6f:a5:d7:97:da:7f:cf:09:15:ab:2f:  
0a:f3:70:e0:d0:d3:50:90:05:78:92:ac:8a:17:78:23:b7:66:  
c6:55

-----BEGIN CERTIFICATE-----

MIID0DCCAzmgAwIBAgIBADANBgkqhkiG9w0BAQQFADCBpzelMAkGA1UEBhMCQ04x  
EzARBgNVBAgTCKd1YW5nIFpob3UxEjAQBgNVBAcTCVNoZW4gWmhlbjEhMB8GA1UE  
ChMYT3BlbiBTb3VyY2UgT3JnYW5pemF0aW9uMRIwEAYDVQQLEwl0ZWNoYmVjY2Vw  
FTATBgNVBAMTDHd3dy45ODEyLm5ldDEhMB8GCSqGSIb3DQEJARYSbmV0a2lsbGVy  
QDk4MTIubmV0MB4XDTAzMTAyNTAxMTMwNVoXDTAzMTEyNDExMTMwNVowgacxZzAJ  
BgNVBAYTAkNOMRMwEQYDVQQIEWpHdWVzYyZ2FuaXphdGlvbjESMBAGA1UECXMJ  
ZW4xITAFBgNVBAoTGE9wZW4gU291cmNlIE9yZ2FuaXphdGlvbjESMBAGA1UECXMJ  
dGVjaG5pY2FsMRUwEwYDVQQDEwx3d3cuOTgxMi5uZXQxITAFBgkqhkiG9w0BCQEW  
Em5ldGtpbGxlcA5ODEyLm5ldDCBnzANBgkqhkiG9w0BAQEFAAOBjQAwgYkCgYEA  
pTCa73WfQEDukE4G9/cL3pfQGi5I70x7ws3y9DAb9MedZXpT19d86iWPvrBX9YmR  
LoBM//GWHkIGAWsfmGkkwX/mDKWuuZxMKdujoz122onAMynFpYt64eX00/N9VNR1  
+sjAHBEHHCQDjvBh2XDP+t3iBEoxwmMqX0TsSGgwRI0CAwEAAaOCAQgwggeEMBOG  
A1UdDgQWBSTT9VBSsqjxnDXb5Y5kVwfpWICjCB1AYDVR0jBIHMMIHJgBSTT9VB

```
SsqigxnDXb5Y5kVwfpWICqGBraSBqjCBpzELMAkGA1UEBhMCQ04xEzARBgNVBAgT
Ckd1YW5nIFpob3UxEjAQBgNVBAcTCVNoZW4gWmhlbjEhMB8GA1UEChMYT3BlbiBT
b3VyY2UgT3JnYW5pemF0aW9uMRIwEAYDVQQLEwl0ZWNoZmlyYW5FTATBgNVBAMT
DHd3dy45ODEyLm5ldDEhMB8GCSqGSIB3DQEJARYSbmV0a2lsbGVyQDk4MTIubmV0
ggEAMAwGA1UdEwQFMAMBAf8wDQYJKoZIhvcNAQEEBQADgYEAP/GZiTfsG4DixjqO
7eiUuHAQNBya7/e+twVR9KLLA0703W9zUUUnSkfzrQDwwVLbwqqHo1DOym9AODbRL
ZcWuv+36/xzmHapBT9p2eld9jfUbF2X8Y0/b3UUzPefJ3ejWjW+115faf88JFasv
CvNw4NDTUJAFeJKsihd4I7dmxIU=
-----END CERTIFICATE-----
-bash-2.05b$
```

4.

```
rm privkey.pem      server.key      600
```

```
-bash-2.05b$ rm privkey.pem
-bash-2.05b$ chmod og-rwx server.key
-bash-2.05b$ ls -l
total 56
drwx----- 10 postgres postgres 4096 Oct 23 12:03 base
drwx-----  2 postgres postgres 4096 Oct 25 08:35 global
drwx-----  2 postgres postgres 4096 Jul  8 17:01 pg_clog
-rw-----  1 postgres postgres 2714 Jul  8 17:57 pg_hba.conf
-rw-----  1 postgres postgres 1441 Jul  8 17:01 pg_ident.conf
-rw-----  1 postgres postgres   4 Jul  8 17:01 PG_VERSION
drwx-----  2 postgres postgres 4096 Oct 15 01:03 pg_xlog
-rw-----  1 postgres postgres 5336 Oct 24 17:01 postgresql.conf
-rw-----  1 postgres postgres  32 Oct 25 08:35 postmaster.opts
-rw-----  1 postgres postgres  44 Oct 25 08:35 postmaster.pid
-rw-r--r--  1 postgres postgres 3670 Oct 25 09:13 server.crt
-rw-----  1 postgres postgres  87 Oct 25 09:04 server.key
-rw-r--r--  1 postgres postgres 2377 Oct 25 08:59 server.req
-bash-2.05b$
```

5.

```
postgresql.conf
SSL      #ssl = false      ssl = true
```

```
-bash-2.05b$ vi postgresql.conf
ssl = true
```

```
postgresql.conf
```

```
-bash-2.05b$ cat postgresql.conf
#
# PostgreSQL configuration file
# -----
#
# This file consists of lines of the form:
```

```
#
#   name = value
#
# (The '=' is optional.) White space may be used. Comments are introduced
# with '#' anywhere on a line. The complete list of option names and
# allowed values can be found in the PostgreSQL documentation. The
# commented-out settings shown in this file represent the default values.
#
# Any option can also be given as a command line switch to the
# postmaster, e.g. 'postmaster -c log_connections=on'. Some options
# can be changed at run-time with the 'SET' SQL command.
#
# This file is read on postmaster startup and when the postmaster
# receives a SIGHUP. If you edit the file on a running system, you have
# to SIGHUP the postmaster for the changes to take effect, or use
# "pg_ctl reload".

#=====

#
#           Connection Parameters
#
#tcpip_socket = false

tcpip_socket = true

#ssl = false
ssl = true

#max_connections = 32
max_connections = 100
#superuser_reserved_connections = 2

#port = 5432
#hostname_lookup = false
#show_source_port = false

#unix_socket_directory = "
#unix_socket_group = "
#unix_socket_permissions = 0777 # octal

#virtual_host = "
```

```

#krb_server_keyfile = "

#
#      Shared Memory Size
#
#shared_buffers = 64          # min max_connections*2 or 16, 8KB each
shared_buffers = 200         # min max_connections*2 or 16, 8KB each
#max_fsm_relations = 1000    # min 10, fsm is free space map, ~40 bytes
#max_fsm_pages = 10000      # min 1000, fsm is free space map, ~6 bytes
#max_locks_per_transaction = 64 # min 10
#wal_buffers = 8            # min 4, typically 8KB each

#
#      Non-shared Memory Sizes
#
#sort_mem = 1024             # min 64, size in KB
#vacuum_mem = 8192          # min 1024, size in KB

#
#      Write-ahead log (WAL)
#
#checkpoint_segments = 3    # in logfile segments, min 1, 16MB each
#checkpoint_timeout = 300   # range 30-3600, in seconds
#
#commit_delay = 0           # range 0-100000, in microseconds
#commit_siblings = 5        # range 1-1000
#
#fsync = true
#wal_sync_method = fsync    # the default varies across platforms:
#                            # fsync, fdatasync, open_sync, or open_datasync
#wal_debug = 0              # range 0-16

#
#      Optimizer Parameters
#
#enable_seqscan = true
#enable_indexscan = true
#enable_tidscan = true
#enable_sort = true
#enable_nestloop = true

```

```

#enable_mergejoin = true
#enable_hashjoin = true

#effective_cache_size = 1000    # typically 8KB each
#random_page_cost = 4           # units are one sequential page fetch cost
#cpu_tuple_cost = 0.01          # (same)
#cpu_index_tuple_cost = 0.001   # (same)
#cpu_operator_cost = 0.0025     # (same)

#default_statistics_target = 10 # range 1-1000

#
#       GEQO Optimizer Parameters
#
#geqo = true
#geqo_selection_bias = 2.0       # range 1.5-2.0
#geqo_threshold = 11
#geqo_pool_size = 0              # default based on tables in statement,
                                # range 128-1024
#geqo_effort = 1
#geqo_generations = 0
#geqo_random_seed = -1          # auto-compute seed

#
#       Message display
#
#server_min_messages = notice    # Values, in order of decreasing detail:
                                #   debug5, debug4, debug3, debug2, debug1,
                                #   info, notice, warning, error, log, fatal,
                                #   panic
#client_min_messages = notice    # Values, in order of decreasing detail:
                                #   debug5, debug4, debug3, debug2, debug1,
                                #   log, info, notice, warning, error
#silent_mode = false

#log_connections = false
#log_pid = false
#log_statement = false
#log_duration = false
#log_timestamp = false

#log_min_error_statement = panic # Values in order of increasing severity:
                                #   debug5, debug4, debug3, debug2, debug1,

```



```

# info, notice, warning, error, panic(off)

#debug_print_parse = false
#debug_print_rewritten = false
#debug_print_plan = false
#debug_pretty_print = false

#explain_pretty_print = true

# requires USE_ASSERT_CHECKING
#debug_assertions = true

#
# Syslog
#
#syslog = 0 # range 0-2
#syslog_facility = 'LOCAL0'
#syslog_ident = 'postgres'

#
# Statistics
#
#show_parser_stats = false
#show_planner_stats = false
#show_executor_stats = false
#show_statement_stats = false

# requires BTREE_BUILD_STATS
#show_btree_build_stats = false

#
# Access statistics collection
#
#stats_start_collector = true
#stats_reset_on_server_start = true
#stats_command_string = false
#stats_row_level = false
#stats_block_level = false

#
```

```

#      Lock Tracing
#
#trace_notify = false

# requires LOCK_DEBUG
#trace_locks = false
#trace_userlocks = false
#trace_lwlocks = false
#debug_deadlocks = false
#trace_lock_oidmin = 16384
#trace_lock_table = 0

#
#      Misc
#
#autocommit = true
#dynamic_library_path = '$libdir'
#search_path = '$user,public'
#datestyle = 'iso, us'
#timezone = unknown          # actually, defaults to TZ environment setting
#australian_timezones = false
#client_encoding = sql_ascii # actually, defaults to database encoding
#authentication_timeout = 60 # 1-600, in seconds
#deadlock_timeout = 1000    # in milliseconds
#default_transaction_isolation = 'read committed'
#max_expr_depth = 10000     # min 10
#max_files_per_process = 1000 # min 25
#password_encryption = true
#sql_inheritance = true
#transform_null_equals = false
#statement_timeout = 0      # 0 is disabled, in milliseconds
#db_user_namespace = false

#
#      Locale settings
#
# (initialized by initdb -- may be changed)
LC_MESSAGES = 'en_US.UTF-8'
LC_MONETARY = 'en_US.UTF-8'
LC_NUMERIC = 'en_US.UTF-8'
LC_TIME = 'en_US.UTF-8'

```

## 6. SSL

```
[root@linux root]# psql -h 127.0.0.1 -Uchen member
Welcome to psql 7.3.3, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms
      \h for help with SQL commands
      \? for help on internal slash commands
      \g or terminate with semicolon to execute query
      \q to quit

member=> \q
[root@linux root]# service postgresql restart
[ OK ]
Starting postgresql service:
[ OK ]
[root@linux root]
[root@linux root]# psql -h 127.0.0.1 -Uchen member
Welcome to psql 7.3.3, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms
      \h for help with SQL commands
      \? for help on internal slash commands
      \g or terminate with semicolon to execute query
      \q to quit

SSL connection (cipher: EDH-RSA-DES-CBC3-SHA, bits: 168)

member=>
```

SSL connection (cipher: EDH-RSA-DES-CBC3-SHA, bits: 168)

TCP

SSL

SSL

[Chapter 19](#)

SSL

<http://www.pgsqldb.org/pgsqldoc-cvs/ssl-tcp.html>

## 7. pg\_hba.conf SSL

```
-bash-2.05b$ vi pg_hba.conf
hostssl all all 127.0.0.1 255.255.255.255 md5
```

## 8.

```
[root@linux8 root]# service postgresql restart
[ OK ]
Starting postgresql service:
[ OK ]
[root@linux8 root]# psql -h 127.0.0.1 -Uchen member
```

Welcome to psql 7.3.3, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms  
      \h for help with SQL commands  
      \? for help on internal slash commands  
      \g or terminate with semicolon to execute query  
      \q to quit

SSL connection (cipher: EDH-RSA-DES-CBC3-SHA, bits: 168)

member=>

pg\_hba.conf

```
-bash-2.05b$ cat pg_hba.conf
# PostgreSQL Client Authentication Configuration File
# =====
#
# Refer to the PostgreSQL Administrator's Guide, chapter "Client
# Authentication" for a complete description.  A short synopsis
# follows.
#
# This file controls: which hosts are allowed to connect, how clients
# are authenticated, which PostgreSQL user names they can use, which
# databases they can access.  Records take one of three forms:
#
# local    DATABASE  USER  METHOD  [OPTION]
# host     DATABASE  USER  IP-ADDRESS  IP-MASK  METHOD  [OPTION]
# hostssl  DATABASE  USER  IP-ADDRESS  IP-MASK  METHOD  [OPTION]
#
# (The uppercase quantities should be replaced by actual values.)
# DATABASE can be "all", "sameuser", "samegroup", a database name (or
# a comma-separated list thereof), or a file name prefixed with "@".
# USER can be "all", an actual user name or a group name prefixed with
# "+" or a list containing either.  IP-ADDRESS and IP-MASK specify the
# set of hosts the record matches.  METHOD can be "trust", "reject",
# "md5", "crypt", "password", "krb4", "krb5", "ident", or "pam".  Note
# that "password" uses clear-text passwords; "md5" is preferred for
# encrypted passwords.  OPTION is the ident map or the name of the PAM
# service.
#
# This file is read on server startup and when the postmaster receives
# a SIGHUP signal.  If you edit the file on a running system, you have
# to SIGHUP the postmaster for the changes to take effect, or use
# "pg_ctl reload".
```

```

# Put your actual configuration here
# -----
#
# CAUTION: The default configuration allows any local user to connect
# using any PostgreSQL user name, including the superuser, over either
# Unix-domain sockets or TCP/IP.  If you are on a multiple-user
# machine, the default configuration is probably too liberal for you.
# Change it to use something other than "trust" authentication.
#
# If you want to allow non-local connections, you need to add more
# "host" records.  Also, remember TCP/IP connections are only enabled
# if you enable "tcpip_socket" in postgresql.conf.

# TYPE  DATABASE  USER          IP-ADDRESS    IP-MASK       METHOD

local  all        all           trust
host   all        all           127.0.0.1     255.255.255.255  trust

# Using sockets credentials for improved security. Not available everywhere,
# but works on Linux, *BSD (and probably some others)

#local  all        all           ident  sameuser
#host   all        all           127.0.0.1     255.255.255.255  md5
#local  all        all           trust
#host   all        all           0.0.0.0       0.0.0.0        md5
hostssl all        all           127.0.0.1     255.255.255.255  md5

```

9.

1. server.key server.crt data postgresql.conf
2. server.key 600 postgres
3. privkey.pem

## 2.9.2 SSH TCP/IP

-L -R

```

[chen@linux chen]$ ssh --help
Usage: ssh [options] host [command]
Options:
  -l user      Log in using this user name.
  -n          Redirect input from /dev/null.
  -F config   Config file (default: ~/.ssh/config).
  -A          Enable authentication agent forwarding.

```

-a Disable authentication agent forwarding (default).  
 -X Enable X11 connection forwarding.  
 -x Disable X11 connection forwarding (default).  
 -i file Identity for public key authentication (default: ~/.ssh/identity)  
 -t Tty; allocate a tty even if command is given.  
 -T Do not allocate a tty.  
 -v Verbose; display verbose debugging messages.  
 Multiple -v increases verbosity.  
 -V Display version number only.  
 -P Don't allocate a privileged port.  
 -q Quiet; don't display any warning messages.  
 -f Fork into background after authentication.  
 -e char Set escape character; ``none" = disable (default: ~).  
 -c cipher Select encryption algorithm  
 -m macs Specify MAC algorithms for protocol version 2.  
 -p port Connect to this port. Server must be on the same port.  
 -L listen-port:host:port Forward local port to remote address  
 -R listen-port:host:port Forward remote port to local address  
 These cause ssh to listen for connections on a port, and  
 forward them to the other side by connecting to host:port.  
 -D port Enable dynamic application-level port forwarding.  
 -C Enable compression.  
 -N Do not execute a shell or command.  
 -g Allow remote hosts to connect to forwarded ports.  
 -1 Force protocol version 1.  
 -2 Force protocol version 2.  
 -4 Use IPv4 only.  
 -6 Use IPv6 only.  
 -o 'option' Process the option as if it was read from a configuration file.  
 -s Invoke command (mandatory) as SSH2 subsystem.  
 -b addr Local IP address.

-L listen-port:host:port

-R listen-port:host:port

ssh -L : PostgreSQL host:5432 @

```

[root@linux8 root]# ssh -L 3333:localhost:5432 root@127.0.0.1
root@127.0.0.1's password:
Last login: Sat Oct 25 09:27:30 2003 from 192.168.1.2
[root@linux8 root]# psql -p3333 -Uchen
psql: could not connect to server: No such file or directory
        Is the server running locally and accepting
        Connections on Unix domain socket "/tmp/.s.PGSQL.3333"?
  
```

```

[root@linux8 root]# psql -p3333 -Uchen member
psql: could not connect to server: No such file or directory
        Is the server running locally and accepting
        Connections on Unix domain socket "/tmp/.s.PGSQL.3333"?
[root@linux8 root]# psql -h 127.0.0.1 -p3333 -Uchen member
Welcome to psql 7.3.3, the PostgreSQL interactive terminal.

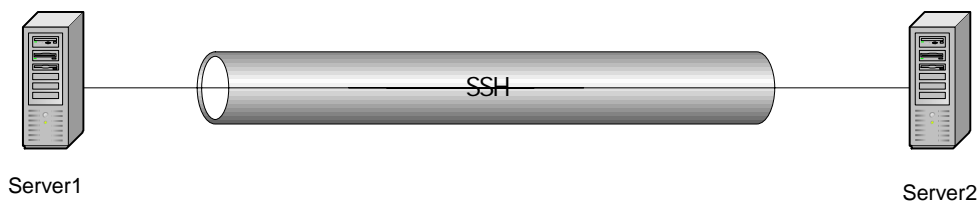
Type: \copyright for distribution terms
       \h for help with SQL commands
       \? For help on internal slash commands
       \g or terminate with semicolon to execute query
       \q to quit

SSL connection (cipher: EDH-RSA-DES-CBC3-SHA, bits: 168)

Member=>

```

SSH+SSL                      SSH    SSL-----Fast Ethernet----- SSL    SSH



```

server1                      server2                      server1
SSH                      server2
server1
IP 192.168.0.1
   client.9812.net
server2
IP 192.168.0.2
   server.9812.net
SSH
server1
server1$ ssh -f -n -L 8000:server.9812.net:5432 server.9812.net
server1$ psql -h 127.0.0.1 -p 8000 -Unetkiller mydb

```

## 2.10 连接 ipv6 主机

1

```

login as: root
Sent username "root"
root@192.168.1.4's password:

```

Last login: Sun Dec 28 13:07:40 2003 from 192.168.1.3

[root@linux9 root]# ifconfig

```
eth0      Link encap:Ethernet  HWaddr 00:0C:29:09:0B:60
          inet addr:192.168.1.4  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: fe80::20c:29ff:fe09:b60/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:34364 errors:0 dropped:0 overruns:0 frame:0
          TX packets:29074 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:22502007 (21.4 Mb)  TX bytes:3561608 (3.3 Mb)
          Interrupt:10 Base address:0x10e0
```

```
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:114 errors:0 dropped:0 overruns:0 frame:0
          TX packets:114 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:10918 (10.6 Kb)  TX bytes:10918 (10.6 Kb)
```

[root@linux9 root]# ifconfig eth0 inet6 add 3ffe:ffff:0:f101::1/64

[root@linux9 root]# ifconfig

```
eth0      Link encap:Ethernet  HWaddr 00:0C:29:09:0B:60
          inet addr:192.168.1.4  Bcast:192.168.1.255  Mask:255.255.255.0
          inet6 addr: 3ffe:ffff:0:f101::1/64 Scope:Global
          inet6 addr: fe80::20c:29ff:fe09:b60/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:34392 errors:0 dropped:0 overruns:0 frame:0
          TX packets:29106 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:22503903 (21.4 Mb)  TX bytes:3565060 (3.3 Mb)
          Interrupt:10 Base address:0x10e0
```

```
lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:114 errors:0 dropped:0 overruns:0 frame:0
          TX packets:114 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:10918 (10.6 Kb)  TX bytes:10918 (10.6 Kb)
```

[root@linux9 root]#



```
[root@linux9 root]# ping6 3ffe:ffff:0:f101::1
PING 3ffe:ffff:0:f101::1(3ffe:ffff:0:f101::1) 56 data bytes
64 bytes from 3ffe:ffff:0:f101::1: icmp_seq=1 ttl=64 time=0.863 ms
64 bytes from 3ffe:ffff:0:f101::1: icmp_seq=2 ttl=64 time=0.099 ms
64 bytes from 3ffe:ffff:0:f101::1: icmp_seq=3 ttl=64 time=0.086 ms
64 bytes from 3ffe:ffff:0:f101::1: icmp_seq=4 ttl=64 time=0.080 ms
64 bytes from 3ffe:ffff:0:f101::1: icmp_seq=5 ttl=64 time=0.124 ms

--- 3ffe:ffff:0:f101::1 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 401ms
rtt min/avg/max/mdev = 0.080/0.250/0.863/0.307 ms
[root@linux9 root]#
```

```
[root@linux classes]# ssh -6 chen@::1
chen@::1's password:
[chen@linux chen]$
```

2

```
[root@linux9 root]# service postgresql start
Initializing database:                [ OK ]
Starting postgresql service:          [ OK ]
[root@linux9 root]# su postgres
bash-2.05b$ createdb -E unicode
CREATE DATABASE
bash-2.05b$ psql -l
      List of databases
  Name      | Owner   | Encoding
-----+-----+-----
 postgres  | postgres | UNICODE
 template0 | postgres | SQL_ASCII
 template1 | postgres | SQL_ASCII
(3 rows)

bash-2.05b$ psql
Welcome to psql 7.3.2, the PostgreSQL interactive terminal.

Type:  \copyright for distribution terms
       \h for help with SQL commands
       \? for help on internal slash commands
       \g or terminate with semicolon to execute query
       \q to quit
```

```
postgres=# \q
```

```
pg_hba.conf
```

```
bash-2.05b$ vi /var/lib/pgsql/data/pg_hba.conf
```

```
local all all ident sameuser  
host all all 3ffe:ffff:0:f101::1/64 trust
```

## 3 DDL

### 3.1 日期时间常量

#### 3.1.1

```
current_date
```

```
netkiller=> select current_date;
```

```
date
```

```
-----  
2003-11-28
```

```
(1 row)
```

```
netkiller=>
```

#### 3.1.2

```
current_time
```

```
netkiller=> select current_time;
```

```
timetz
```

```
-----  
19:38:47.270235+08
```

```
(1 row)
```

```
netkiller=>
```

### 3.1.3

current\_timestamp

```
netkiller=> select current_timestamp;
              timestampz
-----
2003-11-28 19:39:25.548505+08
(1 row)

netkiller=>
```

### 3.1.4

1. current\_timestamp::timestamp (0)
2. current\_timestamp::timestamp (0) without time zone;

```
netkiller=> select current_timestamp::timestamp (0);
              timestamp
-----
2003-11-28 19:44:33
(1 row)

netkiller=>
netkiller=> select current_timestamp::timestamp (0) without time zone;
              timestamp
-----
2003-11-28 19:40:10
(1 row)

netkiller=>
```

### 3.1.5

```
netkiller=> select to_date('2003-12-2','YYYY-MM-DD')-to_date('2003-12-1','YYYY-MM-DD');
?column?
-----
1
(1 row)

netkiller=>
netkiller=> select to_date('2003-12-2','YYYY-MM-DD')-to_date('2003-10-2','YYYY-MM-DD');
?column?
-----
```

```
61
(1 row)
```

### 3.1.6

```
netkiller=> select to_date('2003-12-6','yyyy-mm-dd')+12 ;
?column?
-----
2003-12-18
(1 row)
```

```
netkiller=> select to_date('2003-12-6','yyyy-mm-dd')+20 ;
?column?
-----
2003-12-26
(1 row)
```

### 3.1.7 date\_part

```
netkiller=> select date_part('epoch', '2003-12-3 10:20:30' - timestamp '2003-12-1 02:00:00') ;
date_part
-----
202830
(1 row)
```

```
netkiller=> select date_part('day', '2003-12-3 10:20:30' - timestamp '2003-12-1 02:00:00') ;
date_part
-----
2
(1 row)
```

```
netkiller=> select date_part('hour', '2003-12-3 10:20:30' - timestamp '2003-12-1 02:00:00') ;
date_part
-----
8
(1 row)
```

```
netkiller=>
```

<http://www.pgsql.org>

## 3.2 汉字做字段名

```
PostgreSQL " " " " "
UNICODE EUC_CN
```

```
Create table " "(
  " " Serial NOT NULL UNIQUE,
  " " Varchar(20) NOT NULL,
  " " Varchar(255),
  UNIQUE (" "),
  PRIMARY KEY (" ")
);
```

```
member=> Create table " "(
member(> " " Serial NOT NULL UNIQUE,
member(> " " Varchar(20) NOT NULL,
member(> " " Varchar(255),
member(> UNIQUE (" "),
member(> PRIMARY KEY (" ")
member(> );
NOTICE: CREATE TABLE will create implicit sequence ' _ _seq' for SERIAL column ' . '
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index ' _pkey' for table ' '
NOTICE: CREATE TABLE / UNIQUE will create implicit index ' _ _key' for table ' '
CREATE TABLE
member=> \d
```

### List of relations

Schema	Name	Type	Owner
public	group	table	chen
public	group_id_seq	sequence	chen
public	groupmember	table	chen
public	groupmember_id_seq	sequence	chen
public	role	table	chen
public	role_id_seq	sequence	chen
public	rolemember	table	chen
public	rolemember_id_seq	sequence	chen
public	system_log	table	chen
public	system_log_id_seq	sequence	chen
public	trust	table	chen
public	trust_id_seq	sequence	chen
public	user	table	chen
public	user_id_seq	sequence	chen

```

public | user_log          | table    | chen
public | user_log_id_seq     | sequence | chen
public | userinfo              | table    | chen
public | userinfo_id_seq       | sequence | chen
public | vgroup                 | view     | chen
public | vgroupmember           | view     | chen
public | vsystem_log            | view     | chen
public | vuser                   | view     | chen
public |                          | table    | chen
public | _ _seq                 | sequence | chen
(24 rows)

```

```
member=> \d
```

```

                                Table "public. "
Column |          Type          |          Modifiers          |
-----+-----+-----+-----+-----+-----
      | integer                | not null default nextval('public." _ _seq"::text)
      | character varying(20) | not null
      | character varying(255) |
Indexes:  _pkey primary key btree (" "),
          _ _key unique btree (" ")

```

```
member=> insert into ( , ) values(' ', '9812.net ');
```

```
INSERT 110971 1
```

```
member=> insert into " "(" ", " ") values(' ', '9812.net ');
```

```
ERROR:  Cannot insert a duplicate key into unique index _ _key
```

```
member=> insert into " "(" ", " ") values(' ';
```

```
INSERT 110973 1
```

```
member=> select * from ;
```

```

|          |
-----+-----+-----+-----+-----+-----
 1 |          | 9812.net
 3 |          |

```

```
(2 rows)
```

```
member=> select * from " ";
```

```

|          |
-----+-----+-----+-----+-----+-----
 1 |          | 9812.net
 3 |          |

```

(2 rows)

" "" " ""

API

### 3.3 “::” 数据转换

PostgreSQL

" "

#### 3.3.1 text to varchar

```

    vperson    gender                boolean    true        "        "    false
"            " CASE WHEN
CASE WHEN p.gender = true THEN '    ' ELSE '    ' END as gender,
    '    ', '    ' PostgreSQL    ''            text

```

```

postgres=# CREATE OR REPLACE VIEW vperson AS
postgres-#      SELECT p.uid,p.name,
postgres-#      CASE WHEN p.gender = true THEN '    ' ELSE '    ' END as gender,
postgres-#      p.nickname,p.mobile,p.tel,p.fax,p.email,p.province,p.city,p.adre
ss,p.postalcode
postgres-#      FROM "person" p
postgres-#      Order By p.uid;
CREATE VIEW
postgres=# \dv vperson

```

List of relations

Schema	Name	Type	Owner
public	vperson	view	postgres

-----+-----+-----+-----  
public | vperson | view | postgres  
(1 row)

postgres=# \d person

Table "public.person"

Column	Type	Modifiers
uid	integer	not null default 0
name	character varying(20)	not null
gender	boolean	not null default 'F'
nickname	character varying(20)	
mobile	character varying(13)	
tel	character varying(20)	not null
fax	character varying(20)	
email	character varying(60)	

```

province | character varying(10) | not null
city     | character varying(10) | not null
address  | character varying(255) | not null
postalcode | character varying(6) | not null
rate     | character varying(20) | default '0'
bank     | character varying(20) | not null default ''
bankaccount | character varying(20) | not null default ''

```

Indexes: person\_pkey primary key btree (uid)

Check constraints: "person\_rate" ((((((rate = '0'::character varying) OR (rate = '1'::character varying)) OR (rate = '2'::character varying)) OR (rate = '3'::character varying)) OR (rate = '4'::character varying)) OR (rate = '5'::character varying)))

postgres=#

postgres=# \d vperson

View "public.vperson"

Column	Type	Modifiers
uid	integer	
name	character varying(20)	
gender	text	
nickname	character varying(20)	
mobile	character varying(13)	
tel	character varying(20)	
fax	character varying(20)	
email	character varying(60)	
province	character varying(10)	
city	character varying(10)	
address	character varying(255)	
postalcode	character varying(6)	

View definition: SELECT p.uid, p.name, CASE WHEN (p.gender = true) THEN ' ' ::text ELSE ' ' ::text END AS gender, p.nickname, p.mobile, p.tel, p.fax, p.email, p.province, p.city, p.address, p.postalcode FROM person p ORDER BY p.uid;

```

" " test varchar
CASE WHEN p.gender = true THEN ' ' ::varchar(2) ELSE ' ' ::varchar(2) END as gender,

```

CREATE OR REPLACE VIEW vperson AS

```

SELECT p.uid,p.name,
CASE WHEN p.gender = true THEN ' ' ::varchar(2) ELSE ' ' ::varchar(2) END as gender,
p.nickname,p.mobile,p.tel,p.fax,p.email,p.province,p.city,p.address,p.postalcode
FROM "person" p
Order By p.uid;

```

postgres=# drop view vperson ;



## DROP VIEW

```
postgres=# CREATE OR REPLACE VIEW vperson AS
postgres-#     SELECT p.uid,p.name,
postgres-#     CASE WHEN p.gender = true THEN '      '::varchar(2) ELSE '      '::varchar(2) END as
gender,
postgres-#     p.nickname,p.mobile,p.tel,p.fax,p.email,p.province,p.city,p.address,p.postalcode
postgres-#     FROM "person" p
postgres-#     Order By p.uid;
```

## CREATE VIEW

```
postgres=# \d vperson
```

```
          View "public.vperson"
  Column |          Type          | Modifiers
-----+-----+-----
 uid     | integer                |
 name    | character varying(20) |
 gender  | character varying(2)  |
 nickname| character varying(20) |
 mobile  | character varying(13) |
 tel     | character varying(20) |
 fax     | character varying(20) |
 email   | character varying(60) |
 province| character varying(10) |
 city    | character varying(10) |
 address | character varying(255)|
 postalcode| character varying(6)  |
```

```
View definition: SELECT p.uid, p.name, CASE WHEN (p.gender = true) THEN ('      '::character
varying)::character varying(2) ELSE ('      '::character varying)::character varying(2) END AS gender,
p.nickname, p.mobile, p.tel, p.fax, p.email, p.province, p.city, p.address, p.postalcode FROM person p
ORDER BY p.uid;
```

```
postgres=#
```

## 3.4 序列

### 3.4.1

```
-----
-- 'Region'
```

```
-----
DROP TABLE region;
DROP SEQUENCE   region_id_seq;
DROP INDEX      region_id_index;
```

```
DROP VIEW vregion;
```

```
CREATE TABLE region (  
  id          integer DEFAULT nextval('region_id_seq') NOT NULL,  
  region      varchar(20) DEFAULT " " NOT NULL,  
  description  text ,  
  note        text ,  
  remark      text ,  
  create_date timestamp DEFAULT now() ,  
  modify_date timestamp DEFAULT now() ,  
  PRIMARY KEY (id),  
  UNIQUE (id,region)
```

```
);
```

```
CREATE SEQUENCE region_id_seq;
```

```
CREATE INDEX      region_id_index ON region (id);
```

```
CREATE VIEW vregion AS
```

```
  SELECT  pv.id,pv.region,pv.description,pv.note,pv.remark,to_char(pv.create_date,'YYYY-MM-DD  
HH:MI:SS') as date
```

```
    FROM region pv
```

```
    ORDER BY pv.id;
```

### 3.4.2 “1 2 3 4 5 6 7 8 9...”

```
DROP SEQUENCE      region_id_seq;
```

```
CREATE SEQUENCE    region_id_seq;
```

```
member=> insert into region(region) values(' ');
```

```
INSERT 111264 1
```

```
member=>
```

```
member=> insert into region(region) values(' ');
```

```
INSERT 111265 1
```

```
member=>
```

```
member=> insert into region(region) values(' ');
```

```
INSERT 111266 1
```

```
member=>
```

```
member=> insert into region(region) values(' ');
```

```
INSERT 111267 1
```

```
member=>
```

```
member=> insert into region(region) values(' ');
```

```
INSERT 111268 1
```

```
member=>
```

```
member=> insert into region(region) values(' ');
```

```
INSERT 111269 1
```

```
member=> select * from vregion ;
```

id	region	description	note	remark	date
1					2003-11-01 10:44:26
2					2003-11-01 10:44:26

3						2003-11-01 10:44:26
4						2003-11-01 10:44:26
5						2003-11-01 10:44:26
6						2003-11-01 10:44:26
7						2003-11-01 10:44:26
8						2003-11-01 10:44:26
9						2003-11-01 10:44:26
10						2003-11-01 10:44:26
11						2003-11-01 10:44:26
12						2003-11-01 10:44:26
(12 rows)						

### 3.4.3 “1 3 5 7 9...”

```
DROP SEQUENCE region_id_seq;
Delete from region;
CREATE SEQUENCE region_id_seq INCREMENT 2 START 1;
```

```
member=> DROP SEQUENCE region_id_seq;
DROP SEQUENCE
member=> Delete from region;
DELETE 15
member=>
member=> CREATE SEQUENCE region_id_seq INCREMENT 2 START 1;
CREATE SEQUENCE
member=> insert into region(region) values(' ');
INSERT 111282 1
member=>
member=> insert into region(region) values(' ');
INSERT 111283 1
member=>
member=> insert into region(region) values(' ');
INSERT 111284 1
member=>
member=> insert into region(region) values(' ');
INSERT 111285 1
member=>
member=> insert into region(region) values(' ');
INSERT 111286 1
member=>
member=> insert into region(region) values(' ');
INSERT 111287 1
member=>
```

```

member=> insert into region(region) values(' ');
INSERT 111288 1
member=> select * from region ;
  id | region | description | note | remark |          create_date          |          modify_date
-----+-----+-----+-----+-----+-----+-----
   1 |        |              |      |        | 2003-11-01 11:49:58.004475 | 2003-11-01
11:49:58.004475
   3 |        |              |      |        | 2003-11-01 11:49:58.093188 | 2003-11-01
11:49:58.093188
   5 |        |              |      |        | 2003-11-01 11:49:58.138582 | 2003-11-01
11:49:58.138582
   7 |        |              |      |        | 2003-11-01 11:49:58.166903 | 2003-11-01
11:49:58.166903
   9 |        |              |      |        | 2003-11-01 11:49:58.195132 | 2003-11-01
11:49:58.195132
  11 |        |              |      |        | 2003-11-01 11:49:58.239133 | 2003-11-01
11:49:58.239133
  13 |        |              |      |        | 2003-11-01 11:49:58.267372 | 2003-11-01
11:49:58.267372
  15 |        |              |      |        | 2003-11-01 11:49:58.295643 | 2003-11-01
11:49:58.295643
  17 |        |              |      |        | 2003-11-01 11:49:58.324202 | 2003-11-01
11:49:58.324202
  19 |        |              |      |        | 2003-11-01 11:49:58.352543 | 2003-11-01
11:49:58.352543
  21 |        |              |      |        | 2003-11-01 11:49:58.381273 | 2003-11-01
11:49:58.381273
  23 |        |              |      |        | 2003-11-01 11:49:58.415112 | 2003-11-01
11:49:58.415112
(12 rows)

```

### 3.4.4 “2 4 6 8 10...”

```

DROP SEQUENCE    region_id_seq;
Delete from region;
CREATE SEQUENCE region_id_seq INCREMENT 2 START 2;

```

```

member=> DROP SEQUENCE region_id_seq;
ERROR:  sequence "region_id_seq" does not exist
member=> Delete from region;
DELETE 0
member=> CREATE SEQUENCE region_id_seq INCREMENT 2 START 2;
CREATE SEQUENCE
member=> insert into region(region) values(' ');

```

```

INSERT 111303 1
member=> insert into region(region) values(' ');
INSERT 111304 1
.....
.....
member=> insert into region(region) values(' ');
INSERT 111311 1
member=> insert into region(region) values(' ');
INSERT 111312 1
member=> select * from vregion;
id | region | description | note | remark | date
-----+-----+-----+-----+-----+-----
 2 |      |      |      |      | 2003-11-01 12:00:28
 4 |      |      |      |      | 2003-11-01 12:00:28
 6 |      |      |      |      | 2003-11-01 12:00:28
 8 |      |      |      |      | 2003-11-01 12:00:28
10 |      |      |      |      | 2003-11-01 12:00:28
12 |      |      |      |      | 2003-11-01 12:00:28
14 |      |      |      |      | 2003-11-01 12:00:28
16 |      |      |      |      | 2003-11-01 12:00:28
18 |      |      |      |      | 2003-11-01 12:00:28
20 |      |      |      |      | 2003-11-01 12:00:28
(10 rows)

```

### 3.4.5 $n_1+n_2$

```
CREATE SEQUENCE region_id_seq INCREMENT  $n_2$  START  $n_1$ ;
```

## 3.5 约束

### 3.6 检查约束

```

1
0 1 2 3 4 5 " 0"
=====
-- 'trust'
=====
Create table "trust"
(
    "id" Serial NOT NULL UNIQUE,

```

```

    "uid" integer NOT NULL Default 0,
    "rate" Varchar(20) Default '0' Check (rate in ('0','1','2','3','4','5')),
    primary key ("id")
);
Alter table "trust" add foreign key ("uid") references "user" ("id") on update restrict on delete restrict;

```

```

member=> Insert into trust (uid) values((select id from "user" where userid='netkiller'));
INSERT 111237 1
member=> Insert into trust (uid,rate) values((select id from "user" where userid='netkiller'),5);
INSERT 111220 1
member=> Insert into trust (uid,rate) values((select id from "user" where userid='netkiller'),2);
INSERT 111236 1
member=> Insert into trust (uid,rate) values((select id from "user" where userid='netkiller'),6);
ERROR:  ExecInsert: rejected due to CHECK constraint "trust_rate" on "trust"
member=> Insert into trust (uid,rate) values((select id from "user" where userid='netkiller'),10);
ERROR:  ExecInsert: rejected due to CHECK constraint "trust_rate" on "trust"
member=> select * from trust;
 id | uid | rate
-----+-----+-----
  1 | 257 |  2
  4 | 257 |  0
  5 | 257 |  5
(3 rows)

```

ERROR: ExecInsert: rejected due to CHECK constraint "trust\_rate" on "trust"

2

not in

```

DROP TABLE ctoc.bid CASCADE;
CREATE TABLE ctoc.bid(
    id Serial NOT NULL UNIQUE,
    salesroom_id integer DEFAULT '1' NOT NULL, -- foreign key
    bidder integer DEFAULT '1' NOT NULL, -- foreign key
    price numeric(8,2) DEFAULT '0.00' NOT NULL,
    quantity Integer DEFAULT '1' NOT NULL Check (quantity not in ('0')),
    created timestamp DEFAULT current_timestamp::timestamp (0) without time zone,
    status boolean DEFAULT true,
    PRIMARY KEY (id),
    FOREIGN KEY (salesroom_id) REFERENCES ctoc.salesroom (id) ON UPDATE CASCADE ON DELETE CASCADE,
    FOREIGN KEY (bidder) REFERENCES person (uid) ON UPDATE CASCADE ON DELETE CASCADE
);

```

```

netkiller=> insert into ctoc.bid(salesroom_id,bidder,price,quantity,status) values(1,8,100,0,true);

```

```
ERROR: ExecInsert: rejected due to CHECK constraint "bid_quantity" on "bid"
```

## 3.7 非空约束

```
note
member=> select * from vregion where note is null;
```

## 3.8 唯一约束

### 3.8.1

```
groupname
-- =====
-- 'group'
-- =====
Create table "group"
(
    "id" Serial NOT NULL UNIQUE,
    "groupname" Varchar(20) NOT NULL,
    "description" Varchar(255),
    UNIQUE (groupname),
    PRIMARY KEY ("id")
);
```

```
member=> insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxx');
INSERT 110497 1
member=> insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxx');
ERROR: Cannot insert a duplicate key into unique index group_groupname_key
member=> insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxx');
ERROR: Cannot insert a duplicate key into unique index group_groupname_key
```

```
Psq      ERROR: Cannot insert a duplicate key into unique index group_groupname_key
```

### 3.8.2

```
UNIQUE (rid,uid)          rid,uid
```

```
1 1
1 2
```

1 1  
2 1

2 1  
1 1  
2 2  
1 2  
1 1

" 1 1" " 1 1"

rid,uid

1,1,1  
1,1,2  
1,2,1  
2,1,2  
2,1,1  
2,2,2

1 2 1  
2 1 2  
2 2 1  
1 1 2  
2 2 1

" 2 2 1" " 2 2 1"

" 2 2 1"

-- =====  
-- 'rolemember'  
-- =====

-- drop table rolemember CASCADE ;

Create table "rolemember"

(  
    "id" Serial NOT NULL UNIQUE,  
    "rid" integer NOT NULL Default 0,  
    "uid" integer NOT NULL Default 0,  
    UNIQUE (rid,uid),  
    primary key ("id")  
);

```
member=> insert into rolemember(rid,uid) values((select id from role where rolename ='System'),(select id  
from vuser where userid='sysop'));
```

```
INSERT 110954 1
```

```
member=> insert into rolemember(rid,uid) values((select id from role where rolename ='System'),(select id  
from vuser where userid='sysop'));
```



```

ERROR:  Cannot insert a duplicate key into unique index rolemember_rid_key
member=> insert into rolemember(rid,uid) values((select id from role where rolename ='System'),(select id
from vuser where userid='admin'));
ERROR:  More than one tuple returned by a subselect used as an expression.
member=> insert into rolemember(rid,uid) values((select id from role where rolename ='System'),(select id
from vuser where userid='test'));
INSERT 110956 1
member=> insert into rolemember(rid,uid) values((select id from role where rolename ='System'),(select id
from vuser where userid='test'));
ERROR:  Cannot insert a duplicate key into unique index rolemember_rid_key

```

### 3.8.3

```

          groupname
-----
-- 'group'
-----

Create table "group"
(
    "id" Serial NOT NULL UNIQUE,
    "groupname" Varchar(20) NOT NULL,
    "description" Varchar(255),
    UNIQUE (id,groupname),
    PRIMARY KEY ("id")
);

```

```

postgres=# Create table "group"
postgres-# (
postgres-# "id" Serial NOT NULL UNIQUE,
postgres-# "groupname" Varchar(20) NOT NULL,
postgres-# "description" Varchar(255),
postgres-# UNIQUE (id,groupname),
postgres-# PRIMARY KEY ("id")
postgres-# );
NOTICE: CREATE TABLE will create implicit sequence 'group_id_seq' for SERIAL column
'group.id'
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index 'group_pkey' for table
'group'
NOTICE: CREATE TABLE / UNIQUE will create implicit index 'group_id_key' for table
'group'
CREATE TABLE

```

```
insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxx');
insert into "group"(groupname,description) values('Guest','xxxxxxxxxxxxxxxxxxxx');
insert into "group"(groupname,description) values('Domain','xxxxxxxxxxxxxxxxxxxx');
```

```
postgres=# insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxx');
INSERT 110466 1
postgres=# insert into "group"(groupname,description) values('Guest','xxxxxxxxxxxxxxxxxxxx');
INSERT 110467 1
postgres=# insert into "group"(groupname,description) values('Domain','xxxxxxxxxxxxxxxxxxxx');
INSERT 110468 1
postgres=#
postgres=# insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxx');
INSERT 110469 1
postgres=# insert into "group"(groupname,description) values('Guest','xxxxxxxxxxxxxxxxxxxx');
INSERT 110470 1
postgres=# insert into "group"(groupname,description) values('Domain','xxxxxxxxxxxxxxxxxxxx');
INSERT 110471 1
postgres=# insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxx');
INSERT 110472 1
postgres=# insert into "group"(groupname,description) values('Guest','xxxxxxxxxxxxxxxxxxxx');
INSERT 110473 1
postgres=# insert into "group"(groupname,description) values('Domain','xxxxxxxxxxxxxxxxxxxx');
INSERT 110474 1
postgres=# insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxx');
INSERT 110475 1
postgres=# insert into "group"(groupname,description) values('Guest','xxxxxxxxxxxxxxxxxxxx');
INSERT 110476 1
postgres=# insert into "group"(groupname,description) values('Domain','xxxxxxxxxxxxxxxxxxxx');
INSERT 110477 1
postgres=# insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxx');
INSERT 110478 1
postgres=# insert into "group"(groupname,description) values('Guest','xxxxxxxxxxxxxxxxxxxx');
INSERT 110479 1
postgres=# insert into "group"(groupname,description) values('Domain','xxxxxxxxxxxxxxxxxxxx');
INSERT 110480 1
```

```
postgres=# select * from "group";
```

id	groupname	description
1	Admin	xxxxxxxxxxxxxxxxxxxx
2	Guest	xxxxxxxxxxxxxxxxxxxx
3	Domain	xxxxxxxxxxxxxxxxxxxx
4	Admin	xxxxxxxxxxxxxxxxxxxx
5	Guest	xxxxxxxxxxxxxxxxxxxx
6	Domain	xxxxxxxxxxxxxxxxxxxx
7	Admin	xxxxxxxxxxxxxxxxxxxx

8   Guest	xxxxxxxxxxxxxxxxxxxxxxxx
9   Domain	xxxxxxxxxxxxxxxxxxxxxxxx
10   Admin	xxxxxxxxxxxxxxxxxxxxxxxx
11   Guest	xxxxxxxxxxxxxxxxxxxxxxxx
12   Domain	xxxxxxxxxxxxxxxxxxxxxxxx
13   Admin	xxxxxxxxxxxxxxxxxxxxxxxx
14   Guest	xxxxxxxxxxxxxxxxxxxxxxxx
15   Domain	xxxxxxxxxxxxxxxxxxxxxxxx
(15 rows)	

```

groupname
"id" Serial NOT NULL UNIQUE, (          )
UNIQUE (id,groupname), (id              )

```

Create table "group"

```

(
  "id" Serial NOT NULL UNIQUE,
  "groupname" Varchar(20) NOT NULL,
  "description" Varchar(255),
  UNIQUE (groupname),
  PRIMARY KEY ("id")
);

```

```

member=> insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxxx');
INSERT 110497 1
member=> insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxxx');
ERROR:  Cannot insert a duplicate key into unique index group_groupname_key
member=> insert into "group"(groupname,description) values('Admin','xxxxxxxxxxxxxxxxxxxxx');
ERROR:  Cannot insert a duplicate key into unique index group_groupname_key

```

Psql ERROR: Cannot insert a duplicate key into unique index group\_groupname\_key

### 3.9 主键/外键

#### 3.9.1

```

CREATE TABLE products (
  product_no integer PRIMARY KEY,
  name text,
  price numeric
);

```

```

CREATE TABLE example (

```

```
a integer,  
b integer,  
c integer,  
PRIMARY KEY (a, c)  
);
```

## 3.9.2

1.

```
CREATE TABLE orders (  
    order_id integer PRIMARY KEY,  
    product_no integer REFERENCES products,  
    quantity integer  
);
```

2

```
CREATE TABLE orders (  
    order_id integer PRIMARY KEY,  
    product_no integer REFERENCES products (product_no),  
    quantity integer  
);
```

3.

```
CREATE TABLE table1 (  
    a integer PRIMARY KEY,  
    b integer,  
    c integer,  
    FOREIGN KEY (b, c) REFERENCES other_table (c1, c2)  
);
```

4.

SQL

```
Alter table "groupmember" add foreign key ("uid") references "user" ("id") on update restrict on delete restrict;
```

```
Alter table "groupmember" add foreign key ("gid") references "group" ("id") on update restrict on delete restrict;
```

```
Alter table "rolemember" add foreign key ("uid") references "user" ("id") on update restrict on delete restrict;
```

```
Alter table "rolemember" add foreign key ("rid") references "role" ("id") on update restrict on delete restrict;
```

## 3.9.3 PostgreSQL 7.3.x

```
CREATE TABLE order_items (  
    product_no integer REFERENCES products ON DELETE RESTRICT,  
    order_id integer REFERENCES orders ON DELETE CASCADE,
```

```

quantity integer,
PRIMARY KEY (product_no, order_id)
);
ON DELETE ON UPDATE
CASCADE

```

### 3.9.4 -

```

(3431)
/ / (109)
(197)
(33)
(73)
(75)
(11)
(36)

```

```

XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX
XXXXXXXXXX

```

Drop table "directory" CASCADE;

Create table "directory"

```

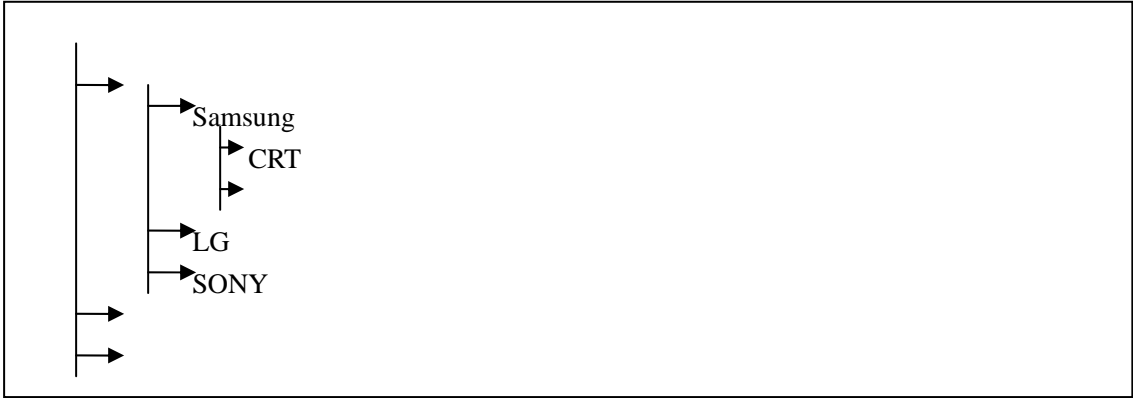
(
  "id" Serial NOT NULL,
  "root_id" Integer NOT NULL Default 0,
  "name" Varchar(20)NOT NULL ,
  "status" boolean Default 'true',
  "created" Timestamp Default current_timestamp,
  "modified" Timestamp Default current_timestamp,
  UNIQUE (id,root_id),
  PRIMARY KEY ("id")
-- FOREIGN KEY (root_id) REFERENCES directory (id) ON DELETE CASCADE
);
INSERT INTO directory (id,root_id,name) VALUES (0,0,'');

```

Alter table "directory" add FOREIGN KEY (root\_id) REFERENCES directory (id) ON DELETE CASCADE;

Create index "directory\_index" on "directory" using btree ("id","root\_id","name");

Id	Root_id	Name
0	0	/
1	0	
2	1	
3	1	
4	1	
5	2	Samsung
6	2	LG
7	2	SONY



```

id
root_id REFERENCES id ON DELETE CASCADE pk fk
name
status true false
created
modified
  
```

```

create table
FOREIGN KEY (root_id) REFERENCES directory (id) ON DELETE CASCADE
" id"
FOREIGN KEY
INSERT INTO directory (id,root_id,name) VALUES (0,0,'/');
  
```

Alter table "directory" add FOREIGN KEY (root\_id) REFERENCES directory (id) ON DELETE

CASCADE;

```
postgres=# Create table "directory"
postgres-# (
postgres(#      "id" Serial NOT NULL,
postgres(#      "root_id" Integer NOT NULL Default 0,
postgres(#      "name" Varchar(20)NOT NULL ,
postgres(#      "status"boolean Default 'true',
postgres(#      "created" Timestamp Default current_timestamp,
postgres(#      "modified" Timestamp Default current_timestamp,
postgres(#      UNIQUE (id,root_id),
postgres(#      PRIMARY KEY ("id")
postgres(# --      FOREIGN KEY (root_id) REFERENCES directory (id) ON DELETE CASCADE
postgres(# );
NOTICE: CREATE TABLE will create implicit sequence 'directory_id_seq' for SERIAL column
'directory.id'
NOTICE: CREATE TABLE / PRIMARY KEY will create implicit index 'directory_pkey' for table
'directory'
NOTICE: CREATE TABLE / UNIQUE will create implicit index 'directory_id_key' for table 'directory'
CREATE TABLE
postgres=# INSERT INTO directory (id,root_id,name) VALUES (0,0,'');
INSERT 17110 1
postgres=# Alter table "directory" add FOREIGN KEY (root_id) REFERENCES directory (id) ON
DELETE CASCADE;
NOTICE: ALTER TABLE will create implicit trigger(s) for FOREIGN KEY check(s)
ALTER TABLE
postgres=# Create index "directory_index" on "directory" using btree ("id","root_id","name");
CREATE INDEX
postgres=# INSERT INTO directory (root_id,name) VALUES (0,'');
INSERT 17116 1
postgres=# SELECT * from directory ;
 id | root_id | name | status | created | modified
-----+-----+-----+-----+-----+-----
  0 |      0 | / | t | 2003-11-12 16:55:39.727365 | 2003-11-12 16:55:39.727365
  1 |      0 | | t | 2003-11-12 16:56:39.663584 | 2003-11-12 16:56:39.663584
(2 rows)

postgres=# INSERT INTO directory (root_id,name) VALUES (0,'');
INSERT 17117 1
postgres=# SELECT * from directory ;
 id | root_id | name | status | created | modified
-----+-----+-----+-----+-----+-----
  0 |      0 | / | t | 2003-11-12 16:55:39.727365 | 2003-11-12 16:55:39.727365
  1 |      0 | | t | 2003-11-12 16:56:39.663584 | 2003-11-12 16:56:39.663584
```

```
2 | 0 | | t | 2003-11-12 16:57:50.509436 | 2003-11-12 16:57:50.509436
```

(3 rows)

```
postgres=# INSERT INTO directory (root_id,name) VALUES (1, ' ');
```

```
INSERT 17118 1
```

```
postgres=# INSERT INTO directory (root_id,name) VALUES (1, ' ');
```

```
INSERT 17119 1
```

```
postgres=# INSERT INTO directory (root_id,name) VALUES (1, ' ');
```

```
INSERT 17120 1
```

```
postgres=# SELECT * from directory ;
```

id	root_id	name	status	created	modified
0	0	/	t	2003-11-12 16:55:39.727365	2003-11-12 16:55:39.727365
1	0		t	2003-11-12 16:56:39.663584	2003-11-12 16:56:39.663584
2	0		t	2003-11-12 16:57:50.509436	2003-11-12 16:57:50.509436
3	1		t	2003-11-12 16:59:15.911196	2003-11-12 16:59:15.911196
4	1		t	2003-11-12 16:59:30.646916	2003-11-12 16:59:30.646916
5	1		t	2003-11-12 16:59:44.400317	2003-11-12 16:59:44.400317

(6 rows)

```
postgres=# INSERT INTO directory (root_id,name) VALUES (3,'Samsung ');
```

```
INSERT 17121 1
```

```
postgres=# INSERT INTO directory (root_id,name) VALUES (3,'LG ');
```

```
INSERT 17122 1
```

```
postgres=# INSERT INTO directory (root_id,name) VALUES (3,'SONY ');
```

```
INSERT 17123 1
```

```
postgres=# SELECT * from directory ;
```

id	root_id	name	status	created	modified
0	0	/	t	2003-11-12 16:55:39.727365	2003-11-12 16:55:39.727365
1	0		t	2003-11-12 16:56:39.663584	2003-11-12 16:56:39.663584
2	0		t	2003-11-12 16:57:50.509436	2003-11-12 16:57:50.509436
3	1		t	2003-11-12 16:59:15.911196	2003-11-12 16:59:15.911196
4	1		t	2003-11-12 16:59:30.646916	2003-11-12 16:59:30.646916
5	1		t	2003-11-12 16:59:44.400317	2003-11-12 16:59:44.400317
6	3	Samsung	t	2003-11-12 17:00:45.964053	2003-11-12 17:00:45.964053
7	3	LG	t	2003-11-12 17:01:03.736121	2003-11-12 17:01:03.736121
8	3	SONY	t	2003-11-12 17:01:18.257337	2003-11-12 17:01:18.257337

(9 rows)

```
postgres=# INSERT INTO directory (root_id,name) VALUES (7,'CRT ');
```

```
INSERT 17124 1
```

```
postgres=# INSERT INTO directory (root_id,name) VALUES (7, ' ');
```



```

INSERT 17125 1
postgres=# INSERT INTO directory (root_id,name) VALUES (8,'
');
INSERT 17126 1
postgres=# INSERT INTO directory (root_id,name) VALUES (8,'
');
INSERT 17127 1
postgres=# INSERT INTO directory (root_id,name) VALUES (7,'
');
INSERT 17128 1
postgres=# SELECT * from directory ;
 id | root_id |      name      | status |      created      |      modified
-----+-----+-----+-----+-----+-----
  0 |      0 | /              | t      | 2003-11-12 16:55:39.727365 | 2003-11-12
16:55:39.727365
  1 |      0 |              | t      | 2003-11-12 16:56:39.663584 | 2003-11-12 16:56:39.663584
  2 |      0 |              | t      | 2003-11-12 16:57:50.509436 | 2003-11-12
16:57:50.509436
  3 |      1 |              | t      | 2003-11-12 16:59:15.911196 | 2003-11-12 16:59:15.911196
  4 |      1 |              | t      | 2003-11-12 16:59:30.646916 | 2003-11-12
16:59:30.646916
  5 |      1 |              | t      | 2003-11-12 16:59:44.400317 | 2003-11-12
16:59:44.400317
  6 |      3 | Samsung        | t      | 2003-11-12 17:00:45.964053 | 2003-11-12 17:00:45.964053
  7 |      3 | LG             | t      | 2003-11-12 17:01:03.736121 | 2003-11-12
17:01:03.736121
  8 |      3 | SONY           | t      | 2003-11-12 17:01:18.257337 | 2003-11-12
17:01:18.257337
  9 |      7 | CRT            | t      | 2003-11-12 17:03:05.594891 | 2003-11-12
17:03:05.594891
 10 |      7 |              | t      | 2003-11-12 17:03:21.793674 | 2003-11-12 17:03:21.793674
 11 |      8 |              | t      | 2003-11-12 17:03:30.688531 | 2003-11-12 17:03:30.688531
 12 |      8 |              | t      | 2003-11-12 17:03:57.697321 | 2003-11-12 17:03:57.697321
 13 |      7 |              | t      | 2003-11-12 17:04:28.61153  | 2003-11-12 17:04:28.61153
(14 rows)

```

```

1.          /      /LG      /CRT
CRT          id  9
SQL  DELETE FROM directory WHERE id=9;

```

```

postgres=# DELETE FROM directory WHERE id=9;
DELETE 1
postgres=# SELECT * from directory ;
 id | root_id |      name      | status |      created      |      modified
-----+-----+-----+-----+-----+-----
  0 |      0 | /              | t      | 2003-11-12 16:55:39.727365 | 2003-11-12
16:55:39.727365

```

```

 1 |      0 |          | t | 2003-11-12 16:56:39.663584 | 2003-11-12 16:56:39.663584
 2 |      0 |          | t | 2003-11-12 16:57:50.509436 | 2003-11-12
16:57:50.509436
 3 |      1 |          | t | 2003-11-12 16:59:15.911196 | 2003-11-12 16:59:15.911196
 4 |      1 |          | t | 2003-11-12 16:59:30.646916 | 2003-11-12
16:59:30.646916
 5 |      1 |          | t | 2003-11-12 16:59:44.400317 | 2003-11-12
16:59:44.400317
 6 |      3 | Samsung | t | 2003-11-12 17:00:45.964053 | 2003-11-12 17:00:45.964053
 7 |      3 | LG      | t | 2003-11-12 17:01:03.736121 | 2003-11-12
17:01:03.736121
 8 |      3 | SONY   | t | 2003-11-12 17:01:18.257337 | 2003-11-12
17:01:18.257337
10 |      7 |          | t | 2003-11-12 17:03:21.793674 | 2003-11-12 17:03:21.793674
11 |      8 |          | t | 2003-11-12 17:03:30.688531 | 2003-11-12 17:03:30.688531
12 |      8 |          | t | 2003-11-12 17:03:57.697321 | 2003-11-12 17:03:57.697321
13 |      7 |          | t | 2003-11-12 17:04:28.61153  | 2003-11-12 17:04:28.61153
(13 rows)

postgres=#

```

2. / /LG

LG

```

postgres=# DELETE FROM directory WHERE id=7;
DELETE 1
postgres=# SELECT * from directory ;
 id | root_id |      name      | status |      created      |      modified
-----+-----+-----+-----+-----+-----
  0 |      0 | /              | t      | 2003-11-12 16:55:39.727365 | 2003-11-12
16:55:39.727365
  1 |      0 |              | t      | 2003-11-12 16:56:39.663584 | 2003-11-12 16:56:39.663584
  2 |      0 |              | t      | 2003-11-12 16:57:50.509436 | 2003-11-12
16:57:50.509436
  3 |      1 |              | t      | 2003-11-12 16:59:15.911196 | 2003-11-12 16:59:15.911196
  4 |      1 |              | t      | 2003-11-12 16:59:30.646916 | 2003-11-12
16:59:30.646916
  5 |      1 |              | t      | 2003-11-12 16:59:44.400317 | 2003-11-12
16:59:44.400317
  6 |      3 | Samsung      | t      | 2003-11-12 17:00:45.964053 | 2003-11-12 17:00:45.964053
  8 |      3 | SONY        | t      | 2003-11-12 17:01:18.257337 | 2003-11-12
17:01:18.257337
11 |      8 |              | t      | 2003-11-12 17:03:30.688531 | 2003-11-12 17:03:30.688531
12 |      8 |              | t      | 2003-11-12 17:03:57.697321 | 2003-11-12 17:03:57.697321
(10 rows)

```

3. / / SONY

```
postgres=# DELETE FROM directory WHERE id=8;
DELETE 1
postgres=# SELECT * from directory ;
 id | root_id |      name      | status |      created      |      modified
-----+-----+-----+-----+-----+-----
  0 |      0 | /              | t      | 2003-11-12 16:55:39.727365 | 2003-11-12
16:55:39.727365
  1 |      0 |                | t      | 2003-11-12 16:56:39.663584 | 2003-11-12 16:56:39.663584
  2 |      0 |                | t      | 2003-11-12 16:57:50.509436 | 2003-11-12 16:57:50.509436
  3 |      1 |                | t      | 2003-11-12 16:59:15.911196 | 2003-11-12 16:59:15.911196
  4 |      1 |                | t      | 2003-11-12 16:59:30.646916 | 2003-11-12 16:59:30.646916
  5 |      1 |                | t      | 2003-11-12 16:59:44.400317 | 2003-11-12 16:59:44.400317
  6 |      3 | Samsung       | t      | 2003-11-12 17:00:45.964053 | 2003-11-12 17:00:45.964053
(7 rows)
```

4. /

LG /CRT SONY /..... LG /.....

```
postgres=# INSERT INTO directory (root_id,name) VALUES (3,'LG ');
INSERT 17129 1
postgres=# INSERT INTO directory (root_id,name) VALUES (3,'SONY ');
INSERT 17130 1
postgres=# INSERT INTO directory (root_id,name) VALUES (6,'CRT ');
INSERT 17131 1
postgres=# INSERT INTO directory (root_id,name) VALUES (14,'CRT ');
INSERT 17132 1
postgres=# INSERT INTO directory (root_id,name) VALUES (15,'CRT ');
INSERT 17133 1
postgres=# INSERT INTO directory (root_id,name) VALUES (15,' ');
INSERT 17134 1
postgres=# INSERT INTO directory (root_id,name) VALUES (15,' ');
INSERT 17135 1
postgres=# INSERT INTO directory (root_id,name) VALUES (6,' ');
INSERT 17136 1
postgres=# INSERT INTO directory (root_id,name) VALUES (14,' ');
INSERT 17137 1
postgres=# INSERT INTO directory (root_id,name) VALUES (15,' ');
INSERT 17138 1
postgres=# SELECT * from directory ;
 id | root_id |      name      | status |      created      |      modified
-----+-----+-----+-----+-----+-----
```

```

0 | 0 | / | t | 2003-11-12 16:55:39.727365 | 2003-11-12
16:55:39.727365
1 | 0 | | t | 2003-11-12 16:56:39.663584 | 2003-11-12 16:56:39.663584
2 | 0 | | t | 2003-11-12 16:57:50.509436 | 2003-11-12
16:57:50.509436
3 | 1 | | t | 2003-11-12 16:59:15.911196 | 2003-11-12 16:59:15.911196
4 | 1 | | t | 2003-11-12 16:59:30.646916 | 2003-11-12
16:59:30.646916
5 | 1 | | t | 2003-11-12 16:59:44.400317 | 2003-11-12
16:59:44.400317
6 | 3 | Samsung | t | 2003-11-12 17:00:45.964053 | 2003-11-12 17:00:45.964053
14 | 3 | LG | t | 2003-11-12 17:28:03.927651 | 2003-11-12
17:28:03.927651
15 | 3 | SONY | t | 2003-11-12 17:28:15.235316 | 2003-11-12
17:28:15.235316
16 | 6 | CRT | t | 2003-11-12 17:28:49.586084 | 2003-11-12
17:28:49.586084
17 | 14 | CRT | t | 2003-11-12 17:28:55.290861 | 2003-11-12
17:28:55.290861
18 | 15 | CRT | t | 2003-11-12 17:28:59.731191 | 2003-11-12
17:28:59.731191
19 | 15 | | t | 2003-11-12 17:29:10.747115 | 2003-11-12 17:29:10.747115
20 | 15 | | t | 2003-11-12 17:29:30.770079 | 2003-11-12 17:29:30.770079
21 | 6 | | t | 2003-11-12 17:29:47.006177 | 2003-11-12 17:29:47.006177
22 | 14 | | t | 2003-11-12 17:29:51.904914 | 2003-11-12 17:29:51.904914
23 | 15 | | t | 2003-11-12 17:29:57.355213 | 2003-11-12 17:29:57.355213
(17 rows)

```

```

postgres=# DELETE FROM directory WHERE id=3;
DELETE 1

```

```

postgres=# SELECT * from directory ;

```

```

id | root_id | name | status | created | modified
-----+-----+-----+-----+-----+-----
0 | 0 | / | t | 2003-11-12 16:55:39.727365 | 2003-11-12 16:55:39.727365
1 | 0 | | t | 2003-11-12 16:56:39.663584 | 2003-11-12 16:56:39.663584
2 | 0 | | t | 2003-11-12 16:57:50.509436 | 2003-11-12 16:57:50.509436
4 | 1 | | t | 2003-11-12 16:59:30.646916 | 2003-11-12 16:59:30.646916
5 | 1 | | t | 2003-11-12 16:59:44.400317 | 2003-11-12 16:59:44.400317

```

```

(5 rows)

```

```

id=0

```

```

id=0

```



```

-----
$user,public
(1 row)
netkiller=> \dt
          List of relations
 Schema |      Name      | Type | Owner
-----+-----+-----+-----
 public | company        | table | netkiller
 public | group          | table | netkiller
 public | groupmember    | table | netkiller
 public | guestbook      | table | netkiller
 public | prodorder      | table | netkiller
 public | role           | table | netkiller
 public | rolemember     | table | netkiller
 public | system_log     | table | netkiller
 public | templates      | table | netkiller
 public | trust          | table | netkiller
 public | user           | table | netkiller
 public | user_log       | table | netkiller
 public | userinfo       | table | netkiller
(13 rows)

```

" \dt" public

SET search\_path TO public,btob,auction;

```

netkiller=> SET search_path TO public,btob,auction;
SET
netkiller=> \dt
          List of relations
 Schema |      Name      | Type | Owner
-----+-----+-----+-----
 auction | messages       | table | netkiller
 auction | product        | table | netkiller
 auction | product_order  | table | netkiller
 btob    | directory      | table | netkiller
 btob    | trade          | table | netkiller
 btob    | trade_message  | table | netkiller
 public  | company        | table | netkiller
 public  | group          | table | netkiller
 public  | groupmember    | table | netkiller
 public  | guestbook      | table | netkiller
 public  | prodorder      | table | netkiller
 public  | role           | table | netkiller
 public  | rolemember     | table | netkiller
 public  | system_log     | table | netkiller
 public  | templates      | table | netkiller

```

public | trust

| table | netkiller

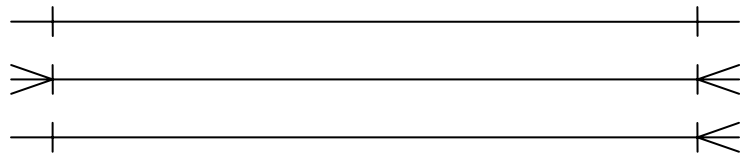
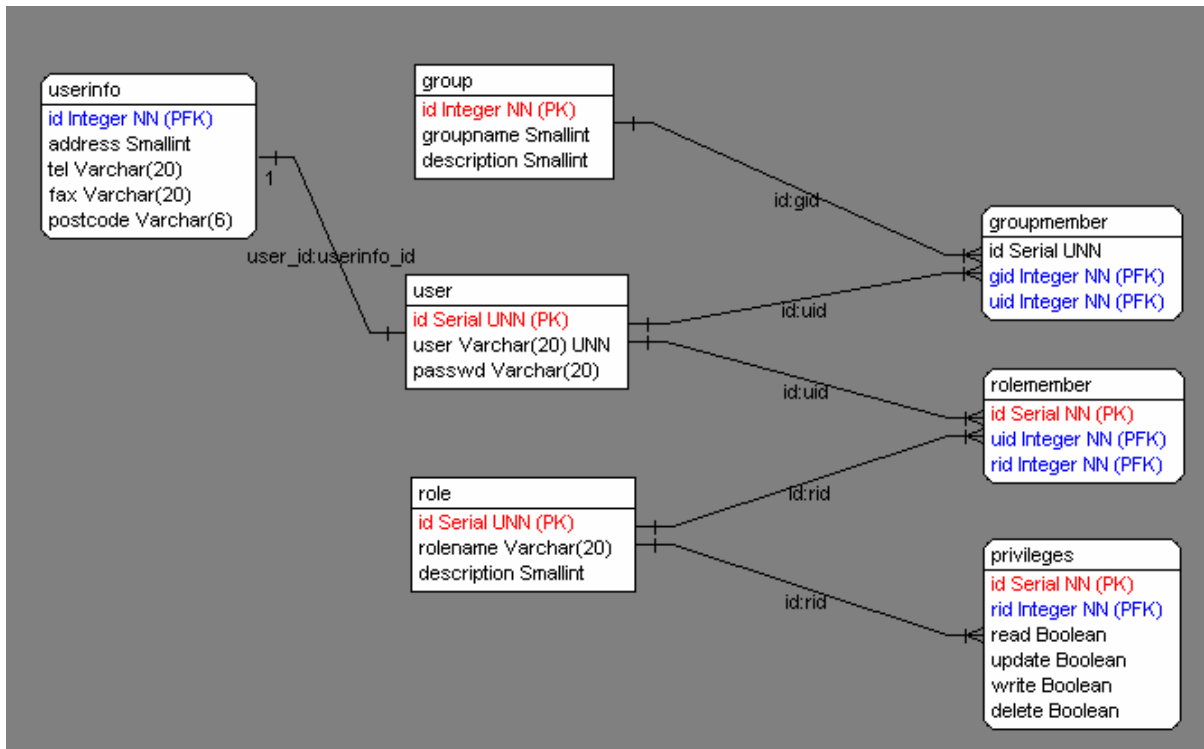
titles publishers titles publishers

titles title\_id sales title\_id

## 4.1 E-R 图 (Entity-Relation)

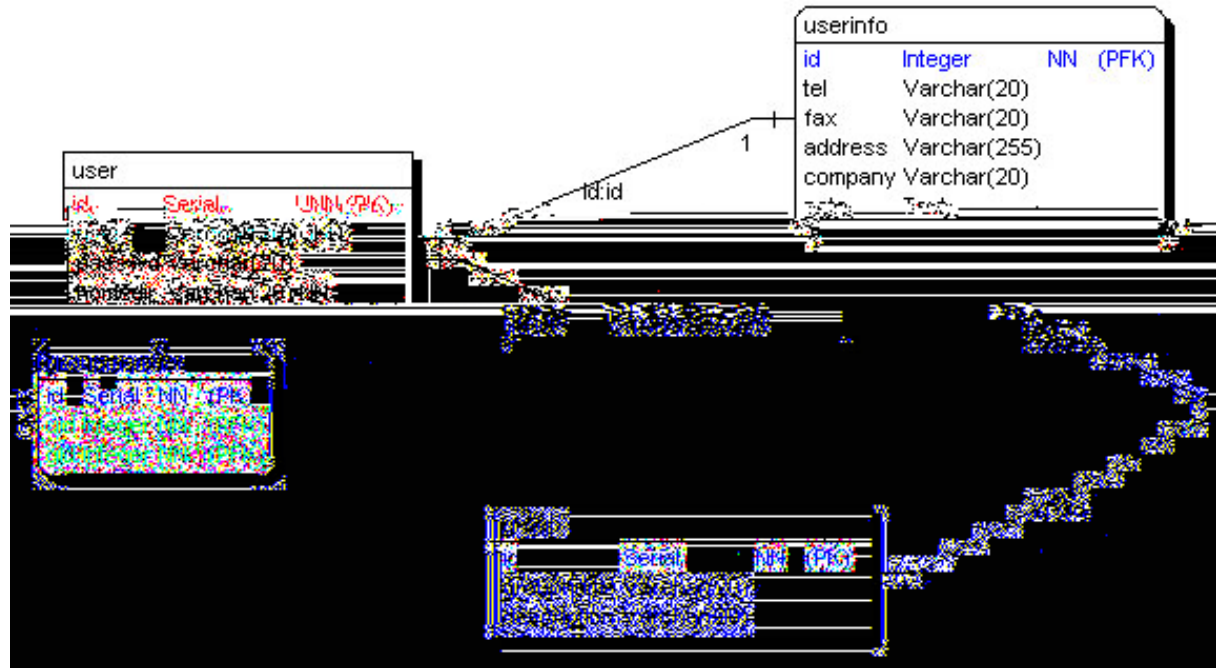
- |             |               |                                  |     |
|-------------|---------------|----------------------------------|-----|
| PostgreSQL  | GUI           | GUI                              | E-R |
| pgAdmin III |               | <a href="#">" CASE Studio 2"</a> |     |
| 1.          |               |                                  |     |
| 2.          | E-R           | SQL DDL                          |     |
| 3.          |               |                                  |     |
| 4.          |               | HTML RTF                         |     |
| 5.          | Add-ins       |                                  |     |
| 6.          | Galery, To-Do |                                  |     |
| 7.          |               |                                  |     |
| 8.          |               |                                  |     |





## 4.2 一对多关系





```
Drop table "groupmember" Restrict;
Drop table "userinfo" Restrict;
Drop table "group" Restrict;
Drop table "user" Restrict;
```

```
Create table "user"
(
    "id" Serial NOT NULL Default 0 UNIQUE ,
    "user" Varchar(20) NOT NULL UNIQUE ,
    "passwd" Varchar(20),
    "homedir" Varchar(20) NOT NULL Default /home/,
    "shell" Varchar(20) NOT NULL Default /bin/bash,
    primary key ("id")
);
```

```
Create table "group"
(
    "id" Serial NOT NULL Default 0,
    "groupname" Varchar(20),
    "description" Varchar(20),
    primary key ("id")
);
```

```
Create table "userinfo"
(
    "id" integer NOT NULL Default 0,
    "tel" Varchar(20),
```

```

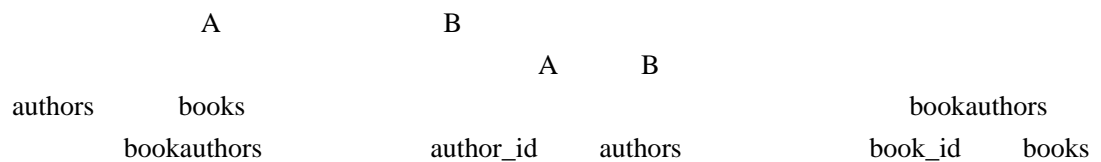
    "fax" Varchar(20),
    "address" Varchar(255),
    "company" Varchar(20),
    "note" Text,
    primary key ("id")
);

Create table "groupmember"
(
    "id" Serial NOT NULL,
    "uid" integer NOT NULL,
    "gid" integer NOT NULL,
    primary key ("id","uid","gid")
);

Alter table "userinfo" add foreign key ("id") references "user" ("id") on update restrict on delete restrict;
Alter table "groupmember" add foreign key ("uid") references "user" ("id") on update restrict on delete restrict;
Alter table "groupmember" add foreign key ("gid") references "group" ("id") on update restrict on delete restrict;

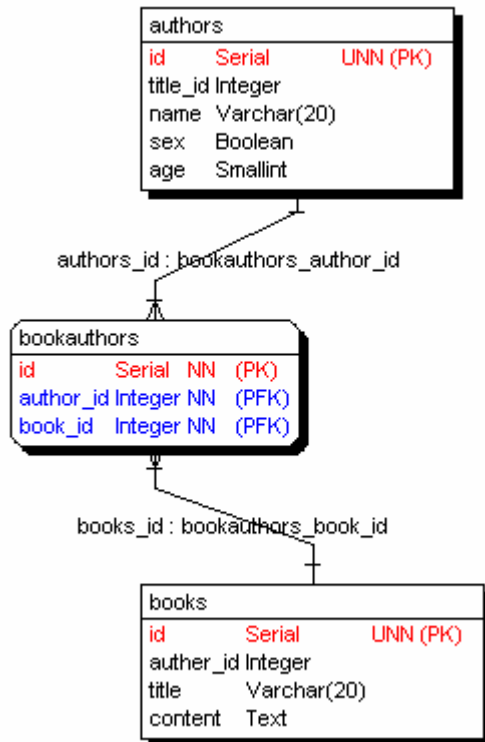
```

## 4.3 多对多关系



Case Studio 2

SQL DDL



Drop table "bookauthors" Restrict;

Drop table "books" Restrict;

Drop table "authors" Restrict;

Create table "authors"

```
(
    "id" Serial NOT NULL UNIQUE ,
    "title_id" integer,
    "name" Varchar(20),
    "sex" Boolean,
    "age" Smallint,
    primary key ("id")
);
```

Create table "books"

```
(
    "id" Serial NOT NULL UNIQUE ,
    "author_id" integer,
    "title" Varchar(20),
    "content" Text,
    primary key ("id")
);
```

Create table "bookauthors"

```
(
```

```

    "id" Serial NOT NULL,
    "author_id" integer NOT NULL,
    "book_id" integer NOT NULL,
    primary key ("id","author_id","book_id")

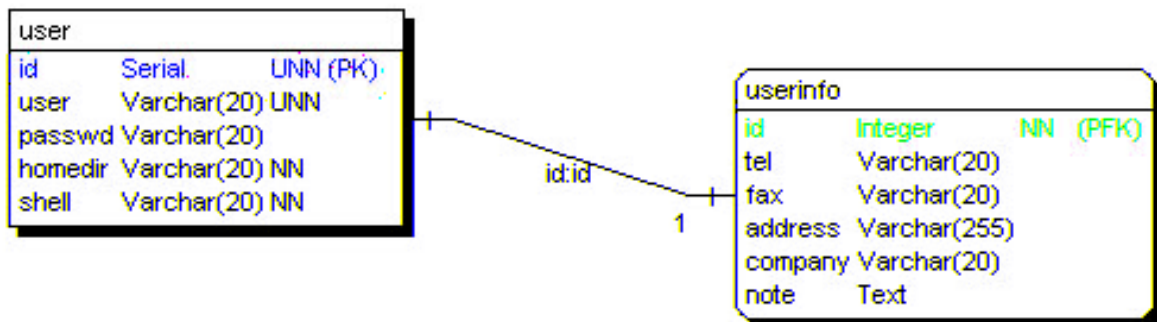
primary key ("id")
);

Alter table "bookauthors" add foreign key ("author_id") references "authors" ("id") on update restrict on delete restrict;
Alter table "bookauthors" add foreign key ("book_id") references "books" ("id") on update restrict on delete restrict;

```

## 4.4 一对一关系

A B



```

Drop table "userinfo" Restrict;
Drop table "user" Restrict;

Create table "user"
(
    "id" Serial NOT NULL Default 0 UNIQUE ,
    "user" Varchar(20) NOT NULL UNIQUE ,
    "passwd" Varchar(20),

```

```

    "homedir" Varchar(20) NOT NULL Default /home/,
    "shell" Varchar(20) NOT NULL Default /bin/bash,
primary key ("id")
);

Create table "userinfo"
(
    "id" integer NOT NULL Default 0,
    "tel" Varchar(20),
    "fax" Varchar(20),
    "address" Varchar(255),
    "company" Varchar(20),
    "note" Text,
primary key ("id")
);

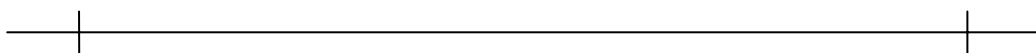
Alter table "userinfo" add foreign key ("id") references "user" ("id") on update restrict on delete restrict;

```

## 4.5 引用完整性



- 1.
- 2.
- 3.
- 4.



---

# 5 SQL DML

## 5.1 子查询

groupmember rolemember

Create table "user"

```
(  
    "id" Serial NOT NULL,  
    "userid" Varchar(50) NOT NULL,  
    "passwd" Varchar(50),  
    "name" Varchar(20)NOT NULL ,  
    "nickname" Varchar(20)NOT NULL ,  
    "active" Boolean Default 'F',  
    "email" Varchar(50) NOT NULL,  
    "question" Varchar(255) NOT NULL,  
    "answer" Varchar(255) NOT NULL,  
    "begin_date" Timestamp Default now(),  
    "end_date" Timestamp Default now(),  
    UNIQUE (userid,email),  
    primary key ("id")  
);  
Create index "user_index" on "user" using btree ("id","userid");
```

Create table "group"

```
(  
    "id" Serial NOT NULL UNIQUE,  
    "groupname" Varchar(20) NOT NULL,  
    "description" Varchar(255),  
    UNIQUE (groupname),  
    PRIMARY KEY ("id")  
);  
Create index "group_index" on "group" using btree ("id","groupname");
```

Create table "groupmember"

```
(  
    "id" Serial NOT NULL UNIQUE,  
    "gid" integer NOT NULL Default 0,  
    "uid" integer NOT NULL Default 0,
```

```
primary key ("id")
);
```

```
-- =====
-- 'Foreign Key'
```

```
-- =====
Alter table "groupmember" add foreign key ("uid") references "user" ("id") on update restrict on delete restrict;
Alter table "groupmember" add foreign key ("gid") references "group" ("id") on update restrict on delete restrict;
```

```
-- -----
-- 'vgroupmember'
```

```
-- -----
-- DROP VIEW vgroupmember;
CREATE VIEW vgroupmember AS
    SELECT gm.id, gm.gid, g.groupname, gm.uid, u.userid, u.name
        FROM "group" g, "user" u, groupmember gm
        Where u.id = gm.uid and g.id = gm.gid
    ORDER BY gm.id;
```

```
member=> select * from groupmember ;
```

id	gid	uid
1	1	245
10	7	200
11	7	201
12	7	202
13	3	200
14	3	201
15	3	202
16	3	203
17	3	204
18	3	205
19	6	247
20	6	201
21	6	203
22	6	204
23	6	205
24	3	249
25	3	250
26	7	251
27	3	252

(19 rows)



```

ID 3
member=> select * from groupmember where gid = 3;
id | gid | uid
-----+-----+-----
13 | 3 | 200
14 | 3 | 201
15 | 3 | 202
16 | 3 | 203
17 | 3 | 204
18 | 3 | 205
24 | 3 | 249
25 | 3 | 250
27 | 3 | 252
(9 rows)

```

1. id select id from vuser where userid='sysop'
2. id select id from "group" where groupname ='System'
3. groupmember insert into groupmember(gid,uid) values ID ID

SQL

```

insert into groupmember(gid,uid) values((select id from "group" where groupname ='System'),(select id
from vuser where userid='sysop'));

```

```

member=> insert into groupmember(gid,uid) values((select id from "group" where groupname
='System'),(select id from vuser where userid='sysop'));
INSERT 110940 1

member=> select * from vgroupmember where gid = 1;
id | gid | groupname | uid | userid | name
-----+-----+-----+-----+-----+-----
1 | 1 | System | 245 | sysop | chen
(2 rows)

```

id

## 5.2 substring()函数截取部分汉字

UNICODE UNICODE =

UNICODE EUC\_CN " "

```

pureftpd=> select substring(' ' from 1 for 4);
pureftpd'>

```

```

pureftpd>'
pureftpd(> )
pureftpd-> ;
ERROR:  parser: unterminated quoted identifier at or near ""
)" at character 69

```

### SQL\_ASCII

```

help=> select substring('
                        ' from 1 for 4);
 substring
-----
          |
(1 row)

```

```

member=> \l
          List of databases
 Name      | Owner   | Encoding
-----+-----+-----
 help      | postgres | SQL_ASCII
 member    | chen    | UNICODE
 mydatabase | postgres | UNICODE
 postgres  | postgres | SQL_ASCII
 pureftpd  | pureftpd | EUC_CN
 site      | postgres | EUC_CN
 template0 | postgres | SQL_ASCII
 template1 | postgres | SQL_ASCII
(8 rows)

```

```

member=> select substring('
                        ' from 1 for 4);
 substring
-----
          |
(1 row)

```

```

member=> select substring('
                        ' from 1 for 2);
 substring
-----
          |
(1 row)

```

```

member=> select * from
          |
-----+-----+-----

```

```

1 |          | 9812.net
3 |          |
(2 rows)

member=> select          ,substring(          from 1 for 5) as          from          ;
          |
-----+-----
          | 9812.
          |
(2 rows)

```

### 5.3 sum()使用技巧

sum()

Select num(field) from table;

Select num(field) from table where field .....

sum()

id 1 4 10 8 2

price

Select field from table where id = 1;

Select field from table where id = 4;

.....

Select field from table where id = 2;

in

select sum(price) from diy.part where id in (1,4,10,8,2);

```

netkiller=> select price from diy.part where id =1;
price
-----
848.00
(1 row)

netkiller=> select price from diy.part where id =2;
price
-----
880.00
(1 row)

netkiller=> select sum(price) as price from diy.part where id in (1,2);
price

```

```
-----  
1728.00  
(1 row)  
  
netkiller=>
```

## 5.4 集合查询（合并 n 个表）

UNION

UNION

2

1

UNION

```
SQL Statement 1 UNION [ALL] SQL Statement 2 UNION [ALL] SQL Statement n
```

```
netkiller=> select rolename from roles ;
```

```
    rolename
```

```
-----  
Administrator  
gold  
silver  
advance  
free  
System  
OA  
(7 rows)
```

```
netkiller=> select groupname from groups;
```

```
    groupname
```

```
-----  
System  
Administrator  
gold  
silver  
advance  
free  
Company  
Person
```

OA

(9 rows)

```
netkiller=> select rolename from roles union select groupname from groups;
```

rolename

-----

Administrator

advance

Company

free

gold

OA

Person

silver

System

(9 rows)

```
netkiller=>
```

```
netkiller=> select rolename from roles union select groupname from groups union select name from person;
```

rolename

-----

Administrator

advance

baba

chen

chenhongxi

Company

CRM

dddd

free

Genenral

gold

HEHE

none

OA

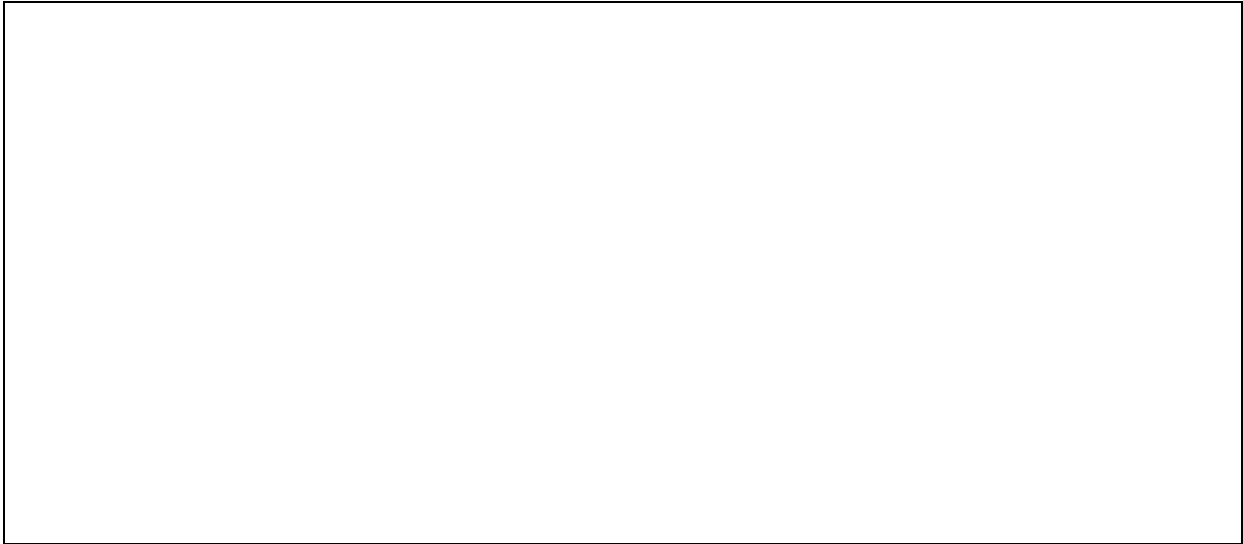
Person

Romain

silver

System

USB



## UNION

### UNION ALL

```
netkiller=> select rolename from roles union all select groupname from groups as chen;
```

```
  rolename
```

```
-----
```

```
Administrator
```

```
gold
```

```
silver
```

```
advance
```

```
free
```

```
System
```

```
OA
```

```
System
```

```
Administrator
```

```
gold
```

```
silver
```

```
advance
```

```
free
```

```
Company
```

```
Person
```

```
OA
```

```
(16 rows)
```

```
netkiller=>
```

```
Insert into table1(field1,field2.....fieldn) select field1,field2.....fieldn from table2
```

<http://www.pgsql.org/pgsqldoc-cvs/queries-union.html>

# 6

- 1.
- 2.
- 3.
- 4.

SQL

100

```
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
```

```
!@% %# @% @^# $& $% * & ^ * % ^ (&)(*) faint " " join
```

VIEW

Select \* from table where xxx = xxxx order by xxxx ASC limit x offset x

## 6.1 VIEW 基本使用实例

Timestamp 2003-09-24 11:23:29

```
CASE WHEN u.active=true THEN 'Y' ELSE 'N' END as "active" SQL CASE
if/else
```

```
-- =====
-- 'user'
-- =====
```

```

Create table "user"
(
    "id" Serial NOT NULL,
    "userid" Varchar(50) NOT NULL,
    "passwd" Varchar(50),
    "name" Varchar(20)NOT NULL ,
    "nickname" Varchar(20)NOT NULL ,
    "active" Boolean Default 'F',
    "email" Varchar(50) NOT NULL,
    "question" Varchar(255) NOT NULL,
    "answer" Varchar(255) NOT NULL,
    "begin_date" Timestamp Default now(),
    "end_date" Timestamp Default now(),
    UNIQUE (userid,email),
    primary key ("id")
);
Create index "user_index" on "user" using btree ("id","userid");

```

```

-----
-- 'vuser'
-----

drop view vuser;
CREATE VIEW vuser AS
    SELECT u.id,u.userid,u."name",u.nickname,
           CASE WHEN u.active=true THEN 'Y' ELSE 'N' END as "active",
           u.email,u.question,u.answer,
           to_char(u.begin_date,'YYYY-MM-DD HH:MI:SS') as begin_date,
           to_char(u.end_date,'YYYY-MM-DD HH:MI:SS') as end_date
    FROM "user" u
    Order By u.id;

```

## 6.2 使用 HTML 格式化 VIEW 的实例

```

-----
-- 'user'
-----

Create table "user"
(
    "id" Serial NOT NULL,
    "userid" Varchar(50) NOT NULL,
    "passwd" Varchar(50),
    "name" Varchar(20)NOT NULL ,
    "nickname" Varchar(20)NOT NULL ,
    "active" Boolean Default 'F',

```



```

    "email" Varchar(50) NOT NULL,
    "question" Varchar(255) NOT NULL,
    "answer" Varchar(255) NOT NULL,
    "begin_date" Timestamp Default now(),
    "end_date" Timestamp Default now(),
    UNIQUE (userid,email),
    primary key ("id")
);
Create index "user_index" on "user" using btree ("id","userid");

```

```

-----
-- 'vuser'
-----
drop view vuser1;

```

```

CREATE OR REPLACE VIEW vuser1 AS
    SELECT '<center>||u.id||</center>' as ID,<font color=red>'||u.userid||</font>' as "
as "      ",u.nickname as "      ",
    CASE WHEN u.active=true THEN '      ' ELSE '      ' END as "active",
    u.email,u.question,u.answer,
    to_char(u.begin_date,'YYYY MM DD HH:MI:SS') as begin_date,
    to_char(u.end_date,'YYYY MM DD HH MI SS ') as end_date
FROM "user" u
Order By u.id;

```

```

member=> select * from vuser1;

```

id	question	answer	begin_date	end_date	active	email
<center>1</center> xxxxxxx	<font color=red>sysop</font>	chen	chen	chen@chen.com		
<center>2</center> xxxxxxx	<font color=red>admin</font>	chen	chen	chen@chen.com		

```

=====
-- 'group'
=====

```

```

Create table "group"
(
    "id" Serial NOT NULL UNIQUE,
    "groupname" Varchar(20) NOT NULL,
    "description" Varchar(255),

```

```

        UNIQUE (groupname),
        PRIMARY KEY ("id")
    );
Create index "group_index" on "group" using btree ("id","groupname");
DROP VIEW vgroup;
CREATE VIEW vgroup AS
    SELECT '<tr>'||g.id||'</td>' as id,'<tr>'||g.groupname||'</td>' as groupname,'<tr>'||g.description||'</td>'
as desc
    FROM "group" g
    ORDER BY g.id;

```

```

postgres=# DROP VIEW vgroup;
ERROR:  view "vgroup" does not exist
postgres=# CREATE VIEW vgroup AS
postgres-#         SELECT '<tr>'||g.id||'</td>' as id,'<tr>'||g.groupname||'</td>' as
groupname,'<tr>'||g.description||'</td>' as desc
postgres-#         FROM "group" g
postgres-#         ORDER BY g.id;
CREATE VIEW
postgres=# select * from vgroup ;

```

id	groupname	desc
1	Admin	xxxxxxxxxxxxxxxxxxxx
2	Guest	xxxxxxxxxxxxxxxxxxxx
3	Domain	xxxxxxxxxxxxxxxxxxxx
4	Admin	xxxxxxxxxxxxxxxxxxxx
5	Guest	xxxxxxxxxxxxxxxxxxxx
6	Domain	xxxxxxxxxxxxxxxxxxxx
7	Admin	xxxxxxxxxxxxxxxxxxxx
8	Guest	xxxxxxxxxxxxxxxxxxxx
9	Domain	xxxxxxxxxxxxxxxxxxxx
10	Admin	xxxxxxxxxxxxxxxxxxxx
11	Guest	xxxxxxxxxxxxxxxxxxxx
12	Domain	xxxxxxxxxxxxxxxxxxxx
13	Admin	xxxxxxxxxxxxxxxxxxxx
14	Guest	xxxxxxxxxxxxxxxxxxxx
15	Domain	xxxxxxxxxxxxxxxxxxxx

(15 rows)

### VIEW

```

DROP VIEW vgroup;
CREATE VIEW vgroup AS
    SELECT '<tr align=center>'||g.id||'</td>' as id,
    '<tr align=left><font color=red><b>'||g.groupname||'</b></font></td>' as groupname,

```

```
'<tr><i>||g.description||</i></td>' as desc
FROM "group" g
ORDER BY g.id;
```

```
postgres=# select * from vgroup ;
```

id	desc	groupname
1		Admin
2		Guest
3		Domain
4		Admin
5		Guest
6		Domain
7		Admin
8		Guest
9		Domain
10		Admin
11		Guest
12		Domain
13		Admin
14		Guest
15		Domain

(15 rows)

HTML B/S  
CREATE OR REPLACE VIEW your\_view AS

## 6.3 view 中使用汉字做字段名

ID		
3	Backup Admin	
2	Power Admin	
1	System Admin	

```
ID |          |
---+-----+-----
 3 | Backup Admin |
 2 | Power Admin  |
 1 | System Admin |
```

DROP VIEW vgroup;

```
CREATE VIEW vgroup AS
  SELECT g.id as "ID",g.groupname as "      ",g.description as "      "
  FROM "group" g
  ORDER BY g.groupname;
```

```
postgres=# CREATE VIEW vgroup AS
postgres-#      SELECT g.id as "ID",g.groupname as "      ",g.description as "      "
postgres-#      FROM "group" g
postgres-#      ORDER BY g.groupname;
CREATE VIEW

postgres=# \d vgroup
          View "public.vgroup"

Column |          Type          | Modifiers
-----+-----+-----
 ID    | integer                |
      | character varying(20) |
      | character varying(255)|
View definition: SELECT g.id AS "ID", g.groupname AS "      ", g.description AS "      " FROM "group"
g ORDER BY g.groupname;

postgres=# insert into "group"(groupname,description) values('System Admin','      ');
INSERT 35031 1
postgres=# insert into "group"(groupname,description) values('Power Admin','      ');
INSERT 35032 1
postgres=# insert into "group"(groupname,description) values('Backup Admin','      ');
```

```

INSERT 35033 1
postgres=# select * from vgroup ;
 ID |          |
-----+-----
 3 | Backup Admin |
 2 | Power Admin  |
 1 | System Admin |

(3 rows)

postgres=#

```

## 6.4 取出字符如果超过 20 个在后尾加 “...”

```

1.          character_length() char_length()
2.          20
3.    20    substring()        20    " ... "
4.          20
5.          ::varchar(    )

```

```

DROP SCHEMA oa CASCADE;
CREATE SCHEMA oa;
SET search_path TO public, btob, btoc, ctoc, oa;

-- =====
-- 'oa.meeting'
-- =====

Drop table oa.meeting CASCADE;

Create table oa.meeting
(
    "id" Serial NOT NULL UNIQUE,
    "subject" varchar(100) NOT NULL,
    "caller" Varchar(10) NOT NULL ,
    "begin_time" Timestamp Default current_timestamp::timestamp (0) without time zone,
    "end_time" Timestamp Default current_timestamp::timestamp (0) without time zone,
    "place" Varchar(100) NOT NULL ,
    "prolocutor" Varchar(10) NOT NULL ,
    "conferee" Varchar(255) NOT NULL ,
    "recorder" Varchar(10) NOT NULL ,
    "leitmotiv" Varchar(255) NOT NULL ,
    "details" text,
    UNIQUE (subject),
    PRIMARY KEY ("id")
)

```

```
);

DROP VIEW oa.vmeeting;
CREATE OR REPLACE VIEW oa.vmeeting AS
  SELECT      id,CASE          WHEN          character_length(subject)>20          THEN
substring(subject,1,20)||'...'::varchar(20) ELSE subject::varchar(20) END,
  caller,to_char(begin_time,'YYYY-MM-DD HH:MI') as begin_time,
  CASE WHEN character_length(place)>10 THEN substring(place,1,10)||'...'::varchar(15) ELSE
place::varchar(20) END as place,
  prolocutor
  FROM oa.meeting
  Order By id DESC;
select * from oa.vmeeting;
```

## 6.5 视图中使用子查询

```
          status          ctoc.vbid          status
(select status from ctoc.vbid where uid = f.uid limit 1) as status          limit
```

ERROR: More than one tuple returned by a subselect used as an expression.

```
-----
-- 'ctoc.vfavorites'
```

```
DROP VIEW ctoc.vfavorites;
CREATE OR REPLACE VIEW ctoc.vfavorites AS
  SELECT      f.id,
             f.uid,
             f.salesroom_id,
             s.subject as subject,
             (select status from ctoc.vbid where uid = f.uid limit 1) as status,
             to_char(f.created,'YYYY/MM/DD HH24:MI:SS') as date
  FROM ctoc.favorites f, person p,ctoc.salesroom s
  WHERE f.uid = p.uid and f.salesroom_id = s.id
  ORDER BY f.id;
```

```
netkiller=> select * from ctoc.vfavorites;
 id | uid | salesroom_id | subject | status | date
-----+-----+-----+-----+-----+-----
 12 |  8 |           1 | MP3 | <font color="#3399FF"> </font> | 2004/03/01 16:30:29
 13 | 95 |           1 | MP3 | <font color=red> </font> | 2004/03/01 16:36:39
(2 rows)

netkiller=>
```

```
select * from ctoc.vfavorites;
```

# 7

PostgreSQL RETURN CREATE FUNCTION SQL

## 7.1 基本使用实例

system\_log add\_system\_log() SQL

```
DROP TABLE system_log CASCADE;
Create table system_log
(
  id Serial NOT NULL UNIQUE,
  uid integer NOT NULL Default 0,
  ip inet ,
  status varchar(255),
  description varchar(255),
  login_date Timestamp Default now(),
  PRIMARY KEY("id"),
  FOREIGN KEY (uid) REFERENCES "user" (id)
);
-----
-- 'Function'
-----
-- DROP FUNCTION add_system_log(integer,inet,varchar);
CREATE OR REPLACE FUNCTION add_system_log(integer,inet,varchar,varchar) RETURNS boolean
AS '
  DECLARE
    vUID          ALIAS FOR $1;
    vIP           ALIAS FOR $2;
    vSTATUS       ALIAS FOR $3;
    vDESC         ALIAS FOR $4;
  BEGIN
    insert into system_log(uid,ip,status,description) values(vUID,vIP,vSTATUS,vDESC);
    RETURN true;
  
```

```

END;
' LANGUAGE 'plpgsql';
select add_system_log(1,'127.0.0.1','Create Database','Initialization Database');
member=> select add_system_log(1,'127.0.0.1','Create Database','Initialization Database');
add_system_log
-----
t
(1 row)
member=> select * from system_log ;
id | uid | ip      | status | description | login_date
---+---+-----+-----+-----+-----
1 | 2 | 192.168.0.5 |      |      | 2003-09-26 15:33:18.6732
2 | 2 | 192.168.0.2 |      | admin | 2003-09-26 16:19:21.824283
3 | 2 | 192.168.1.31 |      | admin | 2003-09-26 17:10:47.269064
(3 rows)

member=> select add_system_log(1,'127.0.0.1','Create Database','Initialization Database') as log;
log
----
t
(1 row)

```

## 7.2 过程中使用 Select Into

```

adduser(vvarchar,vvarchar) deluser(integer)
adduser(      ,      )
ture      false
false
true
deluser(      ID)
ture      false
true      false
-----
-- 'siteuser'
-----
--DROP TABLE IF EXISTS siteuser;
DROP TABLE siteuser CASCADE;
DROP SEQUENCE siteuser_id_seq;
DROP INDEX siteuser_id_index;

CREATE TABLE siteuser (
id integer DEFAULT nextval('siteuser_id_seq') NOT NULL,

```



```

username varchar(20) DEFAULT '0' NOT NULL,
Password varchar(50) DEFAULT '0' NOT NULL,
realname varchar(10) DEFAULT '0' NOT NULL,
email      varchar(50) DEFAULT '0' NOT NULL,
create_date timestamp DEFAULT now(),
modify_date timestamp DEFAULT now(),
UNIQUE      (id,username),
PRIMARY    KEY (id)
);
CREATE SEQUENCE siteuser_id_seq;
CREATE INDEX siteuser_id_index ON siteuser (id);

DROP FUNCTION adduser(varchar,varchar);
CREATE OR REPLACE FUNCTION adduser(varchar,varchar) RETURNS boolean AS '
DECLARE
    bool boolean := false;
    name text;
    uid integer;
    user ALIAS FOR $1;
    pass ALIAS FOR $2;
    su siteuser%ROWTYPE;
BEGIN
    SELECT INTO name username FROM siteuser WHERE username = user;
    IF NOT FOUND then
        insert into siteuser(username,password) values(user,pass);
        SELECT INTO uid id FROM siteuser WHERE username = user;
        bool := true;
    ELSE
        bool := false;
        RAISE NOTICE "Calling adduser() return %",bool;
    END IF;
    RETURN bool;
END;
' LANGUAGE 'plpgsql';

```

```

select adduser('ccsc','eeee');
select * from siteuser ;

```

```

DROP FUNCTION deluser(integer);
CREATE OR REPLACE FUNCTION deluser(integer) RETURNS boolean AS '
DECLARE
    bool boolean := false;
    userid text;
BEGIN

```

```

SELECT INTO userid id FROM siteuser WHERE id = $1;
IF FOUND then
    delete from siteuser where id = $1;
    bool := true;
ELSE
END IF;
RETURN bool;
END;
' LANGUAGE 'plpgsql';

```

```
select deluser(1);
```

## 7.3 返回 integer

```

CREATE OR REPLACE FUNCTION get_user_id(vvarchar) RETURNS integer AS '
DECLARE
    vUser    ALIAS FOR $1;
    uid integer;
BEGIN
    SELECT INTO uid id FROM "users" WHERE userid = vUser;
    RETURN uid;
END;
' LANGUAGE 'plpgsql';

```

## 7.4 返回 void

```

CREATE OR REPLACE FUNCTION enable_user(vvarchar) RETURNS void AS '
DECLARE
    vUser    ALIAS FOR $1;
BEGIN
    Update "users" set status = "true" where userid = vUser;
    RETURN;
END;
' LANGUAGE 'plpgsql';

```

```

CREATE OR REPLACE FUNCTION disable_user(vvarchar) RETURNS void AS '
DECLARE
    vUser    ALIAS FOR $1;
BEGIN
    Update "users" set status = "false" where userid = vUser;
    RETURN;

```

```
END;  
'LANGUAGE 'plpgsql';
```

## 7.5 返回结果集 record

```
create table test(a int, b int);  
insert into test values(1,1);  
insert into test values(2,2);  
insert into test values(3,3);  
insert into test values(4,4);  
insert into test values(5,5);  
insert into test values(6,6);  
insert into test values(7,7);  
insert into test values(8,8);  
insert into test values(9,9);
```

```
create or replace function fun_test_record() returns setof record as '  
    declare  
        rec record;  
    begin  
  
        FOR rec IN SELECT * FROM test LOOP  
            RETURN NEXT rec;  
        END LOOP;  
        return NULL;  
  
    end;  
'language 'plpgsql';
```

```
select a,b from fun_test_record() test(a int, b int);
```

```
9812_net=> select a,b from fun_test_record() test(a int, b int);  
 a | b  
---+---  
 1 | 1  
 2 | 2  
 3 | 3  
 4 | 4  
 5 | 5  
 6 | 6  
 7 | 7  
 8 | 8
```

```

9 | 9
(9 rows)

9812_net=> select test.a,test.b from fun_test_record() test(a int, b int);
 a | b
---+---
 1 | 1
 2 | 2
 3 | 3
 4 | 4
 5 | 5
 6 | 6
 7 | 7
 8 | 8
 9 | 9
(9 rows)

9812_net=>
9812_net=> EXPLAIN select a,b from fun_test_record() test(a int, b int);
                                QUERY PLAN
-----
Function Scan on fun_test_record test  (cost=0.00..12.50 rows=1000 width=8)
(1 row)

9812_net=> EXPLAIN select * from test;
                                QUERY PLAN
-----
Seq Scan on test  (cost=0.00..20.00 rows=1000 width=8)
(1 row)

9812_net=>

```

JDBC    API

SQL

record

## 7.6 例子

```

1
-----
-- 'btoc.order_form'
-----
/*
number        /        30   .   14        , -        15
*/

```

```

DROP TABLE btoc.order_form CASCADE;
CREATE TABLE btoc.order_form (
    id          Serial NOT NULL UNIQUE,
    category_id integer DEFAULT '0' NOT NULL, -- foreign key
    number      varchar (35) NOT NULL,
    person_id   integer DEFAULT '0' NOT NULL, -- foreign key
    status      Varchar(10) Default '0' Check (status in ('0','1','2','3','4','5')),
    created     Timestamp Default current_timestamp::timestamp (0) without time zone,
    modified    Timestamp Default current_timestamp::timestamp (0) without time zone,
    UNIQUE (number),
    PRIMARY KEY (id),
    FOREIGN KEY (category_id) REFERENCES btoc.product_category (id) ON UPDATE CASCADE
ON DELETE CASCADE,
    FOREIGN KEY (person_id) REFERENCES person (uid) ON UPDATE CASCADE ON DELETE
CASCADE
);
CREATE INDEX btoc_order_form_index ON btoc.order_form (id,number,status);

CREATE OR REPLACE FUNCTION btoc.sp_order_form(Integer,Integer,varchar) RETURNS Integer AS '
DECLARE
    ofid integer;
    cid ALIAS FOR $1;
    ofuid ALIAS FOR $2;
    ofstatus ALIAS FOR $3;
    num varchar(35);
BEGIN
    select
        into
        num
to_char(CURRENT_TIMESTAMP,"YYYYMMDDHH24MISS")||"-"||to_char(random(),"FM9V999999999
9999990")::varchar(15);
    Insert
    into
    btoc.order_form(category_id,number,person_id,status)
values
(cid,num,ofuid,ofstatus);
    Select into ofid id FROM btoc.order_form WHERE number = num;
    IF FOUND then
        RETURN ofid;
    ELSE
        RAISE EXCEPTION "Inexistent ID --> %",ofid;
    END IF;
END;
' LANGUAGE 'plpgsql';

```

## 7.7 shell 过程语言

<http://developer.postgresql.org/~petere/pgplsh/pgplsh-1.0-7.3.tar.gz>

# 8



role\_rule group\_rule

Select Update Insert Delect

Update,Insert, Delect

```
IF TG_OP = "UPDATE" THEN
    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
END IF;
IF TG_OP = "INSERT" THEN
    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
END IF;
IF TG_OP = "DELETE" THEN
    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
END IF;
```

## 8.1 规则实例

```
-- =====
-- 'group'
-- =====
Create table "group"
(
    "id" Serial NOT NULL UNIQUE,
    "groupname" Varchar(20) NOT NULL,
    "description" Varchar(255),
    UNIQUE (groupname),
    PRIMARY KEY ("id")
);
Create index "group_index" on "group" using btree ("id","groupname");
-- =====
-- 'groupmember'
-- =====
Create table "groupmember"
(
```

```

        "id" Serial NOT NULL UNIQUE,
        "gid" integer NOT NULL Default 0,
        "uid" integer NOT NULL Default 0,
        primary key ("id")
    );

-----
-- 'vgroupmember'
-----
-- DROP VIEW vgroupmember;
CREATE VIEW vgroupmember AS
    SELECT gm.id, gm.gid, g.groupname, gm.uid, u.userid, u.name
        FROM "group" g, "user" u, groupmember gm
        Where u.id = gm.uid and g.id = gm.gid
    ORDER BY gm.id;

-----
-- 'RULE'
-----
CREATE RULE group_rule AS ON Delete TO "group"
    DO Delete From groupmember where gid = OLD.id;

-----
-- 'Insert Data'
-----
Insert into "group"(groupname,description) values('System',' ');
Insert into "group"(groupname,description) values('Administrator',' ');
Insert into "group"(groupname,description) values('gold',' ');
Insert into "group"(groupname,description) values('silver',' ');
Insert into "group"(groupname,description) values('copper',' ');
Insert into "group"(groupname,description) values('advance',' ');
Insert into "group"(groupname,description) values('free',' ');

-----
-- 'role'
-----
-- drop table role CASCADE;
Create table "role"
(
    "id" Serial NOT NULL UNIQUE,
    "rolename" Varchar(20) NOT NULL,
    "description" Varchar(255),
    UNIQUE (rolename),
    PRIMARY KEY ("id")
);
Create index "role_index" on "role" using btree ("id", "rolename");

```

```

-- =====
-- 'rolemember'
-- =====

-- drop table rolemember CASCADE ;
Create table "rolemember"
(
    "id" Serial NOT NULL UNIQUE,
    "rid" integer NOT NULL Default 0,
    "uid" integer NOT NULL Default 0,
    primary key ("id")
);

-----
-- 'vrolemember'
-----

CREATE VIEW vrolemember AS
    SELECT rm.id,rm.rid,r.rolename,rm.uid,u.userid,u.name
        FROM "role" r,"user" u,rolemember rm
        Where u.id = rm.uid and r.id = rm.rid
        ORDER BY rm.id;

-----
-- 'RULE'
-----

CREATE RULE role_rule AS ON Delete TO role
    DO Delete From rolemember where rid = OLD.id;

-----
-- 'Insert Data'
-----

Insert into role(rolename,description) values('System',' ');
Insert into role(rolename,description) values('Administrator',' ');
Insert into role(rolename,description) values('gold',' ');
Insert into role(rolename,description) values('silver',' ');
Insert into role(rolename,description) values('copper',' ');
Insert into role(rolename,description) values('advance',' ');
Insert into role(rolename,description) values('free',' ');

-----
-- 'Foreign Key'
-----

Alter table "groupmember" add foreign key ("uid") references "user" ("id") on update restrict on delete restrict;
Alter table "groupmember" add foreign key ("gid") references "group" ("id") on update restrict on delete restrict;

```



```
Alter table "rolemember" add foreign key ("uid") references "user" ("id") on update restrict on delete restrict;
```

```
Alter table "rolemember" add foreign key ("rid") references "role" ("id") on update restrict on delete restrict;
```

## 9

### 9.1 一般用法

```
clientinfo_tri      clientinfo_rule
```

```
CREATE OR REPLACE FUNCTION clientinfo_tri_func () RETURNS opaque AS '  
  BEGIN  
    Delete from prodorder where clientinfo_id = OLD.id;  
  RETURN OLD;  
  END;  
' LANGUAGE 'plpgsql';
```

```
DROP TRIGGER clientinfo_tri on clientinfo;  
CREATE TRIGGER clientinfo_tri  
  BEFORE Delete ON clientinfo FOR EACH ROW  
  EXECUTE PROCEDURE clientinfo_tri_func ();
```

```
CREATE RULE clientinfo_rule AS ON Delete TO prodorder  
  DO Delete from prodorder where clientinfo_id = OLD.id;
```

### 9.2 多个触发器使用同一个过程

```
siteuser_delete_tri  
  siteuser
```

```
  Begin;  
  Delete from company where uid = id;  
  Delete from link where uid = id;  
  Delete from product_sort where uid = id;  
  Delete from news where uid = id;  
  Delete from count where uid = id;  
  Delete from guestbook where uid = id;  
  Delete from clientinfo where uid = id;
```

```
Delete from column_bar where uid = id;
Delete from drumbeating where uid = id;
Commit;
```

```
String sql1="Delete from company where uid = ?";
String sql2="Delete from link where uid = ?";
String sql3="Delete from product_sort where uid = ?";
String sql4="Delete from news where uid = ?";
String sql5="Delete from count where uid = ?";
String sql6="Delete from guestbook where uid = ?";
String sql7="Delete from clientinfo where uid = ?";
String sql8="Delete from column_bar where uid = ?";
String sql9="Delete from drumbeating where uid = ?";
String id = <your-id>;
DBConnect odb = null;
try{
    odb = new DBConnect(oDatabase);
    odb.Begin();
    odb.prepareStatement(sql1);
    odb.setString(1,id);
    rs = odb.executeUpdate ();
    odb.prepareStatement(sql2);
    odb.setString(1,id);
    rs = odb.executeUpdate ();
    odb.prepareStatement(sql3);
    odb.setString(1,id);
    rs = odb.executeUpdate ();
    odb.prepareStatement(sql4);
    odb.setString(1,id);
    rs = odb.executeUpdate ();
    odb.prepareStatement(sql5);
    odb.setString(1,id);
    rs = odb.executeUpdate ();
    odb.prepareStatement(sql6);
    odb.setString(1,id);
    rs = odb.executeUpdate ();
    odb.prepareStatement(sql7);
    odb.setString(1,id);
    rs = odb.executeUpdate ();
    odb.prepareStatement(sql8);
    odb.setString(1,id);
    rs = odb.executeUpdate ();
    odb.prepareStatement(sql9);
    odb.setString(1,id);
```

```

        rs = odb.executeUpdate ();

        odb.Commit();
    }
    catch(Exception e){
        e.printStackTrace();
    }
    finally{
        try{
            odb.close();
        }catch(Exception e){
            e.printStackTrace();
        }
    }
    siteuser_insert_tri
siteuser                company count drumbeating.....
-----
-- 'siteuser'
-----
--DROP TABLE IF EXISTS siteuser;
DROP TABLE siteuser CASCADE;
DROP SEQUENCE siteuser_id_seq;
DROP INDEX siteuser_id_index;

CREATE TABLE siteuser (
    id integer DEFAULT nextval('siteuser_id_seq') NOT NULL,
    username varchar(20) DEFAULT '0' NOT NULL,
    Password varchar(50) DEFAULT '0' NOT NULL,
    realname varchar(10) DEFAULT '0' NOT NULL,
    email      varchar(50) DEFAULT '0' NOT NULL,
    create_date timestamp DEFAULT now(),
    modify_date timestamp DEFAULT now(),
    UNIQUE      (id,username),
    PRIMARY    KEY (id)
);
CREATE SEQUENCE siteuser_id_seq;
CREATE INDEX siteuser_id_index ON siteuser (id);

DROP FUNCTION siteuser_tri_func() CASCADE ;
CREATE OR REPLACE FUNCTION siteuser_tri_func () RETURNS opaque AS '
--    DECLARE
--        user_id CONSTANT INTEGER := OLD.id;
BEGIN
    IF TG_OP = "DELETE" THEN

```

```

Delete from company where uid = OLD.id;
Delete from link where uid = OLD.id;
Delete from product_sort where uid = OLD.id;
Delete from news where uid = OLD.id;
Delete from count where uid = OLD.id;
Delete from guestbook where uid = OLD.id;
Delete from clientinfo where uid = OLD.id;
Delete from column_bar where uid = OLD.id;
Delete from drumbeating where uid = OLD.id;
END IF;
IF TG_OP = "INSERT" THEN
    INSERT INTO company(uid) values(NEW.id);
    INSERT INTO count(uid,number,fontcolor,backgroundcolor)
values(NEW.id,0,"fffff","000000");
    INSERT INTO drumbeating(uid,logourl,bannerurl)
values(NEW.id,"default_logo.jpg","default_banner.jpg");
END IF;
RETURN OLD;
END;
' LANGUAGE 'plpgsql';

```

```

DROP TRIGGER siteuser_delete_tri on siteuser;
CREATE TRIGGER siteuser_delete_tri
    BEFORE Delete ON siteuser FOR EACH ROW
    EXECUTE PROCEDURE siteuser_tri_func ();

```

```

DROP TRIGGER siteuser_insert_tri on siteuser;
CREATE TRIGGER siteuser_insert_tri
    AFTER INSERT ON siteuser FOR EACH ROW
    EXECUTE PROCEDURE siteuser_tri_func ();

```

## 9.3 时间调度触发器

PostgreSQL

```

[root@linux pgsq]# cat function.sh
#!/bin/bash

psql -Uchen member -c "select add_system_log(1,'127.0.0.1','Create Database','Initialization Database');"
# or
# psql -Uchen member -f your_sql_script.sql

```

## 9.3.1

function.sh

```
[root@linux8 pgsql]# at -f function.sh tomorrow
warning: commands will be executed using (in order) a) $SHELL b) login shell c) /bin/sh
job 15 at 2003-10-14 18:05
```

8 00 optimize.sh Shell

```
[root@linux8 pgsql]# at -f optimize.sh 20:00 + 1 days
warning: commands will be executed using (in order) a) $SHELL b) login shell c) /bin/sh
job 18 at 2003-10-14 20:00
```

4 1 00 function.sh

```
[root@linux8 pgsql]# at -f function.sh + 5 days
warning: commands will be executed using (in order) a) $SHELL b) login shell c) /bin/sh
job 20 at 2003-10-18 01:00
```

at queue

```
[root@linux8 pgsql]# atq
11      2003-10-14 01:00 a root
12      2003-10-14 18:00 a root
13      2003-10-14 18:00 b root
14      2003-10-14 18:03 a root
15      2003-10-14 18:05 a root
16      2003-10-14 01:00 a root
17      2003-10-13 20:00 a root
18      2003-10-14 20:00 a root
19      2003-10-15 01:00 a root
20      2003-10-18 01:00 a root
```

```
[root@linux8 pgsql]# at -l
11      2003-10-14 01:00 a root
12      2003-10-14 18:00 a root
13      2003-10-14 18:00 b root
14      2003-10-14 18:03 a root
15      2003-10-14 18:05 a root
16      2003-10-14 01:00 a root
17      2003-10-13 20:00 a root
18      2003-10-14 20:00 a root
19      2003-10-15 01:00 a root
20      2003-10-18 01:00 a root
```

## 9.3.2

3 00 function.sh

```
[root@linux etc]# cat crontab
SHELL=/bin/bash
PATH=/sbin:/bin:/usr/sbin:/usr/bin
MAILTO=root
HOME=/
```

```
# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * 0 root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.monthly
0 3 * * * root /usr/local/pgsql/function.sh
```

function.sh

```
0 */2 * * * root /usr/local/pgsql/function.sh
```

1            2            function.sh

```
0 2 1 * * root /usr/local/pgsql/function.sh
```

17 00            function.sh

```
0 17 * * mon root /usr/local/pgsql/function.sh
```

## 9.4 其它例子

1

Id	other	Status
1		False
2		False
3		False
4		True

status            false,

status            true

Status    false            true

```

-- =====
-- 'ctoc.bid'
-- =====
/*
    bidder      (          )
    price
    quantity

*/

DROP TABLE ctoc.bid CASCADE;
CREATE TABLE ctoc.bid(
    id Serial NOT NULL UNIQUE,
    salesroom_id integer DEFAULT '1' NOT NULL, -- foreign key
    bidder integer  DEFAULT '1' NOT NULL, -- foreign key
    price numeric(8,2) DEFAULT '0.00' NOT NULL,
    quantity Integer DEFAULT '1' NOT NULL Check (quantity not in ('0')),
    created timestamp DEFAULT current_timestamp::timestamp (0) without time zone,
    status boolean DEFAULT true,
    PRIMARY KEY (id),
    FOREIGN KEY (salesroom_id) REFERENCES ctoc.salesroom (id) ON UPDATE CASCADE ON
DELETE CASCADE,
    FOREIGN KEY (bidder) REFERENCES person (uid) ON UPDATE CASCADE ON DELETE
CASCADE
);
CREATE INDEX ctoc_bid_index ON ctoc.bid (salesroom_id);
-----
-- 'FUNCTION TRIGGER '
-----
CREATE OR REPLACE FUNCTION ctoc_bid_tri_func () RETURNS TRIGGER AS '
    DECLARE
    BEGIN
        Update ctoc.bid set status = false where salesroom_id = NEW.salesroom_id;
    RETURN NEW;
    END;
' LANGUAGE 'plpgsql';

DROP TRIGGER ctoc_bid_tri on ctoc.bid;
CREATE TRIGGER ctoc_bid_tri
    BEFORE INSERT ON ctoc.bid FOR EACH ROW
    EXECUTE PROCEDURE ctoc_bid_tri_func ();
-----
-- 'Test '

```

```
-----  
insert into ctoc.bid(salesroom_id,bidder,price,quantity,status) values(1,8,100,1,true);  
-----
```

```
-- 'ctoc.vbid'  
-----
```

```
DROP VIEW ctoc.vbid;
```

```
CREATE OR REPLACE VIEW ctoc.vbid AS
```

```
SELECT
```

```
    b.id,
```

```
    b.salesroom_id,
```

```
    b.bidder as uid,
```

```
    p.name as bidder,
```

```
    b.price,
```

```
    b.quantity,
```

```
    CASE WHEN b.status = true THEN b.quantity::varchar ELSE '0' END as quantity1,
```

```
    to_char(b.created,'YYYY-MM-DD HH24:MI:SS') as created,
```

```
    CASE WHEN b.status = true THEN '<font color=red>          </font>':::varchar ELSE '<font  
color="#3399FF">          </font>':::varchar END as status
```

```
FROM ctoc.bid b,ctoc.salesroom room, person p
```

```
WHERE b.salesroom_id = room.id and b.bidder = p.uid
```

```
order by b.salesroom_id,id DESC;
```

```
1
```

```
ERROR:  Calling ctoc_bid_tri_func ()
```

```
CREATE OR REPLACE FUNCTION ctoc_bid_tri_func () RETURNS TRIGGER AS '
```

```
DECLARE
```

```
BEGIN
```

```
    IF NEW.price > (select max(price) from ctoc.bid where salesroom_id = NEW.salesroom_id)
```

```
THEN
```

```
    Update ctoc.bid set status = false where salesroom_id = NEW.salesroom_id;
```

```
ELSE
```

```
--    RAISE NOTICE "Calling ctoc_bid_tri_func ()";
```

```
    RAISE EXCEPTION "Calling ctoc_bid_tri_func ()";
```

```
END IF;
```

```
RETURN NEW;
```

```
END;
```

```
' LANGUAGE 'plpgsql';
```

```
DROP TRIGGER ctoc_bid_tri on ctoc.bid;
```

```
CREATE TRIGGER ctoc_bid_tri
```

```
BEFORE INSERT ON ctoc.bid FOR EACH ROW
```

```
EXECUTE PROCEDURE ctoc_bid_tri_func ();
```



```

netkiller=> insert into ctoc.bid(salesroom_id,bidder,price,quantity,status) values(1,8,100,1,true);
ERROR:  Calling ctoc_bid_tri_func ()
netkiller=> insert into ctoc.bid(salesroom_id,bidder,price,quantity,status) values(1,8,120,1,true);
ERROR:  Calling ctoc_bid_tri_func ()
netkiller=> insert into ctoc.bid(salesroom_id,bidder,price,quantity,status) values(1,8,140,1,true);
INSERT 49772 1
netkiller=>

```

## 10

```

CREATE FUNCTION reffunc(refcursor) RETURNS refcursor AS '
BEGIN
    OPEN $1 FOR SELECT * FROM system_log;
    RETURN $1;
END;
' LANGUAGE 'plpgsql';

```

```

BEGIN;
SELECT reffunc('funcursor');
FETCH ALL IN funcursor;
COMMIT;

```

```

member=>
member=> BEGIN;
BEGIN
member=> SELECT reffunc('funcursor');
    reffunc
-----
    funcursor
(1 row)

member=> FETCH ALL IN funcursor;
 id | uid | ip | status | description | login_date
-----+-----+-----+-----+-----+-----
  8 |  1 | 192.168.0.1 | eeee | ddddddddddddddddddddddddddddddd | 2003-09-25
14:55:19.62117
53 |  1 | 127.0.0.1 | Create Database | Initialization Database | 2003-09-25
15:23:34.941765
 54 |  2 | 192.168.0.5 | | | 2003-09-26
15:33:18.6732
 55 |  2 | 192.168.0.2 | | admin | 2003-09-26
16:19:21.824283

```

## 10.1 游标结果集

```
Java
public void Cursors(){
    String sql          = "SELECT reffunc('funcursor')";
    String sql2        = "FETCH ALL IN funcursor";

    DBConnect odb = null;
    try{
        odb = new DBConnect(oDatabase);
        odb.Begin();
        odb.prepareStatement(sql);
        rs = odb.executeQuery();
        odb.prepareStatement(sql2);

        //odb.setString(1,user);
        rs = odb.executeQuery();
        if(rs!=null) {
            while(rs.next()){
                System.out.print(rs.getString(1)+"|");
                System.out.print(rs.getString(2)+"|");
                System.out.print(rs.getString(3)+"|");
                System.out.println(rs.getString(4));
            }
        }
        odb.Commit();
    }
    catch(Exception e){
        e.printStackTrace();
    }
    finally{
        try{
            odb.close();
        }catch(Exception e){
            e.printStackTrace();
        }
    }
}
```

```
55|2|192.168.0.2|
56|2|192.168.1.31|
57|2|127.0.0.1|
58|1|127.0.0.1|
```

## 10.2 例子

1

```
CREATE FUNCTION reffunc(refcursor,integer) RETURNS refcursor AS '
BEGIN
    OPEN $1 FOR SELECT id,pname,num as amount,price,newprice,(newprice*num) as
subtotal,((price-newprice)*num) as cost_saving FROM product WHERE id = $2;
    RETURN $1;
END;
' LANGUAGE 'plpgsql';
```

2

```
CREATE OR REPLACE FUNCTION btoc.porder(refcursor,Integer,Integer) RETURNS refcursor AS '
DECLARE
    search_id ALIAS FOR $2;
    num ALIAS FOR $3;
BEGIN
    OPEN $1 FOR
        SELECT
            id,name,
            CASE WHEN money is null THEN "RMB" ELSE money
END||to_char(price,"99G999G999D99") as price,
            CASE WHEN money is null THEN "RMB" ELSE money
END||to_char(newprice,"99G999G999D99") as newprice,
            num as quantity,
            newprice*num as subtotal,
            price*num - newprice*num as cost_saving
        FROM btoc.product
        WHERE id = search_id;
    IF FOUND then
        RETURN $1;
    ELSE
        RETURN null;
    END IF;
END;
' LANGUAGE 'plpgsql';

BEGIN;
```

```

select btoc.porder('funcursor',75,500);
FETCH ALL IN funcursor;
-- CLOSE funcursor;
-- select porder('funcursor',16,5);
-- FETCH ALL IN funcursor;
COMMIT;

```

# 11

## 11.1 批量插入、更新、删除

- 1.
- 2.

### 11.1.1 - 1

```

insert into "group"(groupname,description) values('System Admin',' ');
insert into "group"(groupname,description) values('Power Admin',' ');
insert into "group"(groupname,description) values('Backup Admin',' ');
.....
.....
.....
insert into "group"(groupname,description) values('xxxxxxxxxxxxx','xxxxxxxxxxxxx');

for( ) {
    SQL insert into "group"(groupname,description) values(' ' ');
}

begin
for( ) {
    SQL insert into "group"(groupname,description) values(' ' ');
}
commit

```

commit

1G-2G

4G

CASE

```

Java
public void delGroupMember(ArrayList id) {
    String sql = "delete from groupmember where id=?";
    try{
        odb = new DBConnect(oDatabase);
        odb.Begin();
        for(int i=0;i<id.size();i++){
            odb.prepareStatement(sql);
            odb.setString(1,(String)id.get(i));
            odb.executeUpdate();
        }
        odb.Commit();
        odb.AutoCommit();
    }
    catch(Exception e){
        odb.Rollback();
        e.printStackTrace();
    }
    finally{
        try{
            odb.close();
        }catch(Exception e){
            e.printStackTrace();
        }
    }
}
}

```

## 11.2 保持数据完整-例 2

A
B C D E F.....  
 RULE TRIGGER

```

Begin;
Delete from company where uid = id;
Delete from link where uid = id;
Delete from product_sort where uid = id;
Delete from news where uid = id;
Delete from count where uid = id;
Delete from guestbook where uid = id;
Delete from clientinfo where uid = id;
Delete from column_bar where uid = id;
Delete from drumbeating where uid = id;

```

Commit;

# 12

1. PostgreSQL
- 2.
3. PostgreSQL
- 4.
- 5.

1. pg\_hba.conf 1
2. 2 3
3. VALID UNTIL 4

[www.9812.net](http://www.9812.net) :9812\_net :9812\_net  
" \_" "9812\_net", "xuser\_net"  
:

9812net xusernet kdeopencom

9812 xuser kdeopen

## 12.1.1

### 12.1.1.1 创建组

```
CREATE GROUP groupname;
```

```
ALTER GROUP groupname ADD USER user1, user2... ;
```

```
ALTER GROUP groupname DROP USER user1, user2... ;
```

```
postgres=# CREATE GROUP company;  
CREATE GROUP  
postgres=# CREATE GROUP person;
```

```
CREATE GROUP
postgres=#
```

### 12.1.1.2 删除组

```
DROP GROUP groupname;
postgres=# DROP GROUP vhost;
DROP GROUP
```

## 12.1.2

### 12.1.2.1 创建用户

1. netkiller chen  
CREATE USER netkiller WITH PASSWORD 'chen';

2. dbuser1 chen 2003-12-1  
CREATE USER dbuser1 WITH PASSWORD 'chen' VALID UNTIL '2003-12-1';

```
postgres=# CREATE USER dbuser1 WITH PASSWORD 'chen' VALID UNTIL '2003-12-1';
CREATE USER
postgres=# CREATE USER "kdeopen_comt" WITH PASSWORD 'chen' VALID UNTIL '2003-12-30';
CREATE USER
postgres=# CREATE USER "9812_net" WITH PASSWORD 'chen' VALID UNTIL '2004-2-1';
CREATE USER
```

3. vhost  
CREATE USER vhostdbuser WITH PASSWORD 'chen' IN GROUP vhost;

```
postgres=# CREATE GROUP vhost;
CREATE GROUP
postgres=# CREATE USER vhostdbuser WITH PASSWORD 'chen' IN GROUP vhost;
CREATE USER
postgres=# CREATE USER vhostdbuser1 WITH PASSWORD 'chen' VALID UNTIL '2003-12-1' IN
GROUP vhost;
CREATE USER
postgres=#
postgres=# \du
```

```
                List of database users
  User name  | User ID |          Attributes
-----+-----+-----
 9812_net   |    102 |
 dbuser1    |    101 |
```

```
kdeopen_com |    103 |
netkiller    |    100 |
postgres     |      1 | superuser, create database
vhostdbuser  |    104 |
vhostdbuser1 |    105 |
(7 rows)

postgres=#
```

### 12.1.2.2 删除用户

```
DROP USER username;
```

```
postgres=# DROP USER dbuser1;
DROP USER
postgres=#
```

### 12.1.2.3 修改密码

```
ALTER USER " " PASSWORD ' ' ;
```

```
postgres=# ALTER USER "9812_net" PASSWORD '123456';
ALTER USER
postgres=#
```

```
bash-2.05b$ psql -h 127.0.0.1 -U9812
Password: ██████████
psql: FATAL: Password authentication failed for user "9812"

bash-2.05b$ psql -h 127.0.0.1 -U9812_net
Password:
Welcome to psql 7.3.4, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms
      \h for help with SQL commands
      \? for help on internal slash commands
      \g or terminate with semicolon to execute query
      \q to quit

9812_net=>
```



## 12.1.3

```
CREATE DATABASE kdeopen WITH OWNER = 9812_net TEMPLATE = template0 ENCODING = 'UNICODE';
```

```
postgres=# CREATE DATABASE "9812_net" WITH OWNER = "9812_net" TEMPLATE = template0 ENCODING = 'UNICODE';
```

```
CREATE DATABASE
```

```
postgres=# \l
```

```
      List of databases
```

```
   Name   |   Owner   | Encoding
```

```
-----+-----+-----
```

```
9812_net | 9812_net | UNICODE
```

```
netkiller | netkiller | UNICODE
```

```
postgres | postgres | SQL_ASCII
```

```
template0 | postgres | SQL_ASCII
```

```
template1 | postgres | SQL_ASCII
```

```
(5 rows)
```

```
postgres=#
```

## 12.1.4

### 12.1.4.1 本地连接

```
# local   DATABASE USER METHOD [OPTION]
```

```
# host    DATABASE USER IP-ADDRESS IP-MASK METHOD [OPTION]
```

```
# hostssl DATABASE USER IP-ADDRESS IP-MASK METHOD [OPTION]
```

```
pg_hba.conf
```

```
[root@linux data]# cat pg_hba.conf
```

```
host    all             all             127.0.0.1      255.255.255.255 md5
```

```
local   all             all                                     trust
```

```
127.0.0.1
```

```
[root@linux data]# psql -h127.0.0.1 -u member
```

```
psql: Warning: The -u option is deprecated. Use -U.
```

```
User name: chen
```

```
Password:
```

```
Welcome to psql 7.3.3, the PostgreSQL interactive terminal.
```

```
Type: \copyright for distribution terms
```

```
      \h for help with SQL commands
```

```
\? for help on internal slash commands
\g or terminate with semicolon to execute query
\q to quit
```

```
member=>
```

LAN

```
[root@linux data]# psql -hwww.9812.net -u member
psql: Warning: The -u option is deprecated. Use -U.
User name: chen
Password:
psql: FATAL: No pg_hba.conf entry for host 61.145.139.113, user chen, database member

[root@linux2 data]#
```

```
-h 127.0.0.1                                UNIX Domain Socket
/tmp/ .s.PGSQL.5432
```

## 12.1.4.2 允许任何 IP 连接主机

```
[root@linux data]# cat pg_hba.conf
host    all             all             0.0.0.0         0.0.0.0         md5
```

## 12.1.5

```
9812
kdeopen          202.103.190.130
xuser           IP                    xuser
```

pg\_hba.conf

```
[root@linux data]# cat pg_hba.conf
host    9812             9812           127.0.0.1       255.255.255.255  md5
host    kdeopen       kdeopen       202.103.190.130 255.255.255.0   md5
host    xuser          xuser         0.0.0.0         0.0.0.0         md5
```

SQL

```
CREATE GROUP company;
CREATE GROUP person;
CREATE USER 9812 WITH PASSWORD 'chen' VALID UNTIL '2003-12-1' IN GROUP person;
CREATE DATABASE 9812 WITH OWNER = 9812 TEMPLATE = template0 ENCODING =
'UNICODE';
CREATE USER kdeopen WITH PASSWORD 'chen' VALID UNTIL '2003-12-1' IN GROUP company;
CREATE DATABASE kdeopen WITH OWNER = kdeopen TEMPLATE = template0 ENCODING =
'UNICODE';
```



BBS  
 ISO8859-1( ) JDBC Database (UNICODE)  
 ISO8859-1( ) JDBC Database (EUC\_CN)  
 GB2312( ) JDBC Database (SQL\_ASCII)

UTF-8( ) JDBC (UTF-8) Database (Unicode)  
 psql UTF-8

### 13.2.1 Jsp/Java

### 13.2.2 toChinese()

BBS Jsp/Java " ??????????"

```
public static String toChinese(String strvalue){
  try{
    if(strvalue==null)
      return null;
    else{
      strvalue = new String(strvalue.getBytes("ISO8859_1"), "GBK");
      return strvalue;
    }
  }catch(Exception e){
    return null;
  }
}
```

### 13.2.3 Unicode (UTF-8)

Unicode(UTF-8) GBK GB2312 EUC\_CN Unicode  
 Unicode Jsp UTF-8

#### 13.2.3.1 setCharacterEncoding() 方案

```
<%@ page contentType="text/html; charset=gb2312" language="java" import="java.sql.*"
errorPage="error.jsp" %>
<html>
```

```

<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset= gb2312">
<link href="../includes/style.css" rel="stylesheet" type="text/css">
</head>

```

```

<% @ page contentType="text/html; charset=utf-8" language="java" import="java.sql.*" errorPage="" %>
<jsp:include page="../includes/unicode.jsp" flush="true" />
<% @ page import="java.util.*"%>
<jsp:useBean id="sd" scope="page" class="ebusiness.btob.supplydemand"/>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
<link href="../includes/style.css" rel="stylesheet" type="text/css">
</head>
<%
    String CharacterEncoding = "UTF-8";
    response.setContentType("text/html; charset=UTF-8");
    request.setCharacterEncoding(CharacterEncoding);
%>

```

```

String CharacterEncoding = "UTF-8";
request.setCharacterEncoding(CharacterEncoding);
    utf-8

```

```

bash-2.05b$ createdb -E Unicode mydatabase
CREATE DATABASE
bash-2.05b$ psql -l
    List of databases
  Name      | Owner   | Encoding
-----+-----+-----
 member    | postgres | SQL_ASCII
 mydatabase | postgres | UNICODE
 postgres  | postgres | SQL_ASCII
 pureftpd  | pureftpd | EUC_CN
 site      | postgres | EUC_CN
 template0 | postgres | SQL_ASCII
 template1 | postgres | SQL_ASCII

```

(7 rows)

bash-2.05b\$

" " " "

```
CREATE GROUP person;
```

```
CREATE USER xuser WITH PASSWORD 'chen' VALID UNTIL '2003-12-1' IN GROUP person;
```

```
CREATE DATABASE xuser WITH OWNER = xuser TEMPLATE = template0 ENCODING =  
'UNICODE';
```

Inc.jsp

<%



```
* permission of the Apache Group.
*
* THIS SOFTWARE IS PROVIDED ``AS IS" AND ANY EXPRESSED OR IMPLIED
* WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES
* OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE
* DISCLAIMED. IN NO EVENT SHALL THE APACHE SOFTWARE FOUNDATION OR
* ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
* SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT
* LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF
* USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND
* ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY,
* OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT
* OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF
* SUCH DAMAGE.
```

```
* =====
```

```
*
* This software consists of voluntary contributions made by many
* individuals on behalf of the Apache Software Foundation. For more
* information on the Apache Software Foundation, please see
* <http://www.apache.org/>.
```

```
*
* [Additional notices, if required by prior licensing conditions]
```

```
*
*/
```

```
package filters;
```

```
import java.io.IOException;
import javax.servlet.Filter;
import javax.servlet.FilterChain;
import javax.servlet.FilterConfig;
import javax.servlet.ServletException;
import javax.servlet.ServletRequest;
import javax.servlet.ServletResponse;
import javax.servlet.UnavailableException;
```

```
/**
 * <p>Example filter that sets the character encoding to be used in parsing the
 * incoming request, either unconditionally or only if the client did not
 * specify a character encoding. Configuration of this filter is based on
 * the following initialization parameters:</p>
```



- \* <ul>
- \* <li><strong>encoding</strong> - The character encoding to be configured
- \* for this request, either conditionally or unconditionally based on
- \* the <code>ignore</code> initialization parameter. This parameter
- \* is required, so there is no default.</li>
- \* <li><strong>ignore</strong> - If set to "true", any character encoding
- \* specified by the client is ignored, and the value returned by the
- \* <code>selectEncoding()</code> method is set. If set to "false,
- \* <code>selectEncoding()</code> is called <strong>only</strong> if the
- \* client has not already specified an encoding. By default, this
- \* parameter is set to "true".</li>

\* </ul>

\*

\* <p>Although this filter can be used unchanged, it is also easy to

\* subclass it and make the <code>selectEncoding()</code> method more

\* intelligent about what encoding to choose, based on characteristics of

\* the incoming request (such as the values of the <code>Accept-Language</code>

\* and <code>User-Agent</code> headers, or a value stashed in the current

\* user's session.</p>

\*

\* @author Craig McClanahan

\* @version \$Revision: 1.1.2.1 \$ \$Date: 2001/10/17 22:52:17 \$

\*/

```
public class SetCharacterEncodingFilter implements Filter {
```

```
    // ----- Instance Variables
```

```
    /**
```

```
     * The default character encoding to set for requests that pass through
     * this filter.
```

```
    */
```

```
    protected String encoding = null;
```

```
    /**
```

```
     * The filter configuration object we are associated with. If this value
     * is null, this filter instance is not currently configured.
```

```
    */
```

```
    protected FilterConfig filterConfig = null;
```

```

/**
 * Should a character encoding specified by the client be ignored?
 */
protected boolean ignore = true;

// ----- Public Methods

/**
 * Take this filter out of service.
 */
public void destroy() {

    this.encoding = null;
    this.filterConfig = null;

}

/**
 * Select and set (if specified) the character encoding to be used to
 * interpret request parameters for this request.
 *
 * @param request The servlet request we are processing
 * @param result The servlet response we are creating
 * @param chain The filter chain we are processing
 *
 * @exception IOException if an input/output error occurs
 * @exception ServletException if a servlet error occurs
 */
public void doFilter(ServletRequest request, ServletResponse response,
                    FilterChain chain)
    throws IOException, ServletException {

    // Conditionally select and set the character encoding to be used
    if (ignore || (request.getCharacterEncoding() == null)) {
        String encoding = selectEncoding(request);
        if (encoding != null)
            request.setCharacterEncoding(encoding);
    }

    // Pass control on to the next filter
    chain.doFilter(request, response);

```

```

}

/**
 * Place this filter into service.
 *
 * @param filterConfig The filter configuration object
 */
public void init(FilterConfig filterConfig) throws ServletException {

this.filterConfig = filterConfig;
    this.encoding = filterConfig.getInitParameter("encoding");
    String value = filterConfig.getInitParameter("ignore");
    if (value == null)
        this.ignore = true;
    else if (value.equalsIgnoreCase("true"))
        this.ignore = true;
    else if (value.equalsIgnoreCase("yes"))
        this.ignore = true;
    else
        this.ignore = false;

}

// ----- Protected Methods

/**
 * Select an appropriate character encoding to be used, based on the
 * characteristics of the current request and/or filter initialization
 * parameters.  If no character encoding should be set, return
 * <code>null</code>.
 * <p>
 * The default implementation unconditionally returns the value configured
 * by the <strong>encoding</strong> initialization parameter for this
 * filter.
 *
 * @param request The servlet request we are processing
 */
protected String selectEncoding(ServletRequest request) {

    return (this.encoding);
}

```

```
}
```

```
}
```

```
filters ( ) WEB-INF/classes  
WEB-INF web.xml
```

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"  
"http://java.sun.com/dtd/web-app_2_3.dtd">  
  
<web-app>  
  
<welcome-file-list>  
<welcome-file>index.jsp</welcome-file>  
</welcome-file-list>  
  
<filter>  
<filter-name>Set Character Encoding</filter-name>  
<filter-class>filters.SetCharacterEncodingFilter</filter-class>  
<init-param>  
<param-name>encoding</param-name>  
<param-value>utf-8</param-value>  
</init-param>  
<init-param>  
<param-name>ignore</param-name>  
<param-value>>true</param-value>  
</init-param>  
</filter>  
  
<filter-mapping>  
<filter-name>Set Character Encoding</filter-name>  
<url-pattern>/*</url-pattern>  
</filter-mapping>  
  
</web-app>
```

### 13.2.3.3 Jdbc url charSet 方案

jdbc:postgresql://localhost:5432/database?charSet=UTF-8

## 13.3 Tomcat JNDI Datasource 配置

PostgreSQL JDBC <http://jdbc.postgresql.org/>  
jar                    /usr/local/jakarta-tomcat/common/lib/  
xml                    webapps.xml                    /usr/local/jakarta-tomcat/conf/Catalina/localhost/  
Tomcat 4.x    jakarta-tomcat /webapps                    5.x    jakarta-tomcat/conf/Catalina/localhost/

webapps.xml

```
[root@linux root]# cat /usr/local/jakarta-tomcat/webapps/webapps.xml
<Context path="/test" docBase="/var/www/html/webapps" debug="0" reloadable="true">

  <Logger className="org.apache.catalina.logger.FileLogger"
    prefix="localhost_webapps_log." suffix=".txt"
    timestamp="true"/>

  <Resource name="jdbc/postgrespool" auth="Container" type="javax.sql.DataSource"/>
  <ResourceParams name="jdbc/postgrespool">
    <parameter>
      <name>factory</name>
      <value>org.apache.commons.dbcp.BasicDataSourceFactory</value>
    </parameter>
    <parameter>
      <name>driverClassName</name>
      <value>org.postgresql.Driver</value>
    </parameter>
    <parameter>
      <name>url</name>
      <value>jdbc:postgresql://127.0.0.1:5432/netkiller</value>
    </parameter>
    <parameter>
      <name>username</name>
      <value>netkiller</value>
    </parameter>
    <parameter>
      <name>password</name>
      <value>chen</value>
    </parameter>
    <parameter>
```

```

    <name>maxActive</name>
    <value>10</value>
  </parameter>
  <parameter>
    <name>maxIdle</name>
    <value>10</value>
  </parameter>
  <parameter>
    <name>maxWait</name>
    <value>-1</value>
  </parameter>
</ResourceParams>
</Context>

```

```

DriverClassName      ClassName
url                  URL   jdbc:postgresql://      :      /
username
password
maxActive
maxIdle
maxWait

```

web.xml

```

[root@linux root]# cat /var/www/html/webapps/WEB-INF/web.xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
"http://java.sun.com/dtd/web-app_2_3.dtd">

<web-app>

  <welcome-file-list>
    <welcome-file>index.jsp</welcome-file>
  </welcome-file-list>

  <filter>
    <filter-name>Set Character Encoding</filter-name>
    <filter-class>filters.SetCharacterEncodingFilter</filter-class>
    <init-param>
      <param-name>encoding</param-name>
      <param-value>utf-8</param-value>
    </init-param>
    <init-param>
      <param-name>ignore</param-name>
      <param-value>true</param-value>

```

```

        </init-param>
    </filter>

    <filter-mapping>
        <filter-name>Set Character Encoding</filter-name>
        <url-pattern>/*</url-pattern>
    </filter-mapping>

<description>Postgresql Test App</description>
<resource-ref>
    <description>DB Connection</description>
    <res-ref-name>jdbc/postgrespool</res-ref-name>
    <res-type>javax.sql.DataSource</res-type>
    <res-auth>Container</res-auth>
</resource-ref>

<servlet>
    <servlet-name>DisplayChart</servlet-name>
    <servlet-class>org.jfree.chart.servlet.DisplayChart</servlet-class>
</servlet>

</web-app>

```

db-connect-test.jsp

```

<% @ page contentType="text/html; charset=GBK"%>
<% @ page import="java.sql.*,
    javax.naming.*"%>
<%
try {
    Context initCtx = new InitialContext();
    Context ctx = (Context) initCtx.lookup("java:comp/env");
    Object obj = (Object) ctx.lookup("jdbc/postgrespool");
    javax.sql.DataSource ds = (javax.sql.DataSource)obj;
    Connection conn = ds.getConnection();
    Statement stmt = conn.createStatement(java.sql.ResultSet.TYPE_SCROLL_INSENSITIVE,
        java.sql.ResultSet.CONCUR_READ_ONLY);
    String strSQL = "SELECT * FROM users";
    ResultSet rs = stmt.executeQuery(strSQL);
}
%>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
"http://www.w3.org/TR/html4/loose.dtd">

```

```
<html>
<head>
<title>PostgreSQL Test</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
</head>

<body>
<%
    while(rs.next()){
        out.print(rs.getString(2)+"<br>");
    }
%>
</body>
</html>
<%
    rs.close();
    stmt.close();
    conn.close();
} catch (Exception e) {
    e.printStackTrace();
    throw e;
}
%>
```

URL

<http://192.168.0.1:8080/test/db-connect-test.jsp>

## 13.4 JDBC 通过 SSL 安全连接数据库

```
java.sql.DriverManager.getConnection("jdbc:postgresql://db.example.com/testdb?ssl","netkiller","password")
```

## 13.5 开发相关

### 13.5.1 Create Java Entity Bean (not EJB CMP)

<http://home.9812.net/linux/article/postgres/python/>



## 13.5.2

```
package netkiller.database;

/**
 * Title PostgreSQL database JDBC interface.
 * @author: netkiller ( )
 * Company: none
 * Homepage: http://www.xuser.net
 * Copyright: Copyright (c) 2003
 * @version 1.0
 * @since
 * @PostgreSQL database JDBC interface.
 * @see 2003-3-19
 */

import java.sql.*;
import java.util.*;
import java.io.*;
import javax.sql.*;
import org.postgresql.jdbc3.*;

//import net.xuser.database.DBConnectionManager;
//implements Database
public class DBConnect{

    private Connection conn = null;
    private Statement stmt = null;
    private PreparedStatement prepstmt = null;
    private ResultSet rs = null;
    private DBConnectionManager dcm=null;

    private String Driver = null;
    private String url = null;
    private String user = null;
    private String passwd = null;
    private boolean isConnect = false;

    private String odbc = null;

    private void init(String odb) {
        try{
            dcm = DBConnectionManager.getInstance(odb);
```

```

        //System.out.println(dcm);
        if(dcm == null){
            //this.conn == null ||
            System.out.println("Database Connect false!");
            //throw new Exception("Database Connect false!");
            //return;
        }else{
            this.conn = dcm.getConnection();
        }

    }

    catch (Exception e) {
        e.printStackTrace();
    }
}

public DBConnect() {
    init("jdbc");
    this.odbc = "jdbc";
}

public DBConnect(String odb) {

    this.Driver      = "org.postgresql.Driver";
    //this.url       = "jdbc:postgresql://localhost:5432/site";
    this.user        = "netkiller";
    this.passwd      = "chen";
    //this.setUrl("localhost","site");
    this.setUrl("192.168.0.1","netkiller");

    //System.out.print("Database Connection:"+getConnection());

    if(odb == null)odb="jdbc";
    this.odbc = odb;
    init(odb);
}

/*
InitialContext ctx = new InitialContext();
DataSource ds = (DataSource) ctx.lookup("java:/MSSQLDS");
Connection conn = ds.getConnection();
Statement stmt = conn.createStatement();
ResultSet rs = stmt.executeQuery("SELECT * FROM Employees");
while ( rs.next() ) {
    out.println( rs.getInt("EmployeeID") + "<br>");
}
conn.close();
*/

```

```
}

public void setDriver(String dri){
    this.Driver      = dri;
}

public void setUrl(String db){
    this.url         = "jdbc:postgresql://localhost:5432/"+db;
}

public void setUrl(String host,String db){
    this.url         = "jdbc:postgresql://" + host + ":5432/"+db;
}

public void setUrl(String host,int port,String db){
    this.url         = "jdbc:postgresql://" + host + ":" + port + "/" +db;
}

public void setUser(String user){
    this.user        = user;
}

public void setPasswd(String pass){
    this.passwd      = pass;
}

public void setHost(){ }
public void setPort(){ }

public String getUrl(){
    return this.url;
}

public String getDriver(){
    return this.Driver;
}

public String getUser(){
    return this.user;
}

public String getPasswd(){
    return this.passwd;
}
```

```

public Connection Connection(){
    this.isConnect = false;
    try{
        Class.forName(this.getDriver()).newInstance();
        this.conn = DriverManager.getConnection(this.getUrl(), this.getUser(), this.getPasswd());
        isConnect = true;
    }
    catch (Exception e) {
        e.printStackTrace();
    }
    return this.conn;
}

public boolean getConnection(){

    this.Connection();
    if(this.isConnect){
        return this.isConnect;
    }else{
        System.out.println("
");
        return this.isConnect;
    }
}

public String getCatalog(){
    String tmp = "";
    try{
        if(this.conn != null)
            tmp = this.conn.getCatalog();
    }catch(SQLException e) {
        e.printStackTrace();
    }
    return tmp;
}

public void AutoCommit(){
    try{
        this.conn.setAutoCommit(true);
    }catch(SQLException e) {
        e.printStackTrace();
    }
}

```

```

public void Begin(){
    try{
        this.conn.setAutoCommit(false);
    }catch(SQLException e) {
        e.printStackTrace();
    }
}

public void Commit(){
    try{
        if(this.conn != null)
            this.conn.commit();
    }catch(SQLException e) {
        e.printStackTrace();
    }
}

public void Rollback(){
    try{
        if(this.conn != null)
            this.conn.rollback();
    }catch(SQLException e) {
        e.printStackTrace();
    }
}

public ResultSet executeQuery(String strSQL) {

    try {
        this.stmt = this.conn.createStatement();
        if(this.stmt != null){
            this.rs = this.stmt.executeQuery(strSQL);
        }
    } catch(SQLException e) {
        e.printStackTrace();
    }
    /*
    finally {
        try{
            // Always close properly
            this.close();
        }catch(Exception e){
            e.printStackTrace();
        }
    }
}

```

```

    }*/
    return this.rs;
}

public int getCount(String strTable){

    String strSQL      = "select count(*) from "+strTable;
    int  nCount        = 0;

    try {
        this.stmt = this.conn.createStatement();
        if(this.stmt != null){
            this.rs = stmt.executeQuery(strSQL);
            if(rs!=null) {
                while(rs.next()){
                    nCount = rs.getInt(1);
                }
            }
        }
    }
    catch(SQLException e) {
        e.printStackTrace();
    }

    finally {
        try{
            // Always close properly
            this.close();
        }catch(Exception e){
            e.printStackTrace();
        }
    }

    return nCount;
}

public boolean executeUpdate(String strSQL){
    boolean isSuccess = false;
    try{
        this.stmt = this.conn.createStatement();
        //stmt = this.conn.createStatement();
        stmt.executeUpdate(strSQL);
        isSuccess = true;
    }catch(SQLException e) {

```

```

        e.printStackTrace();
    }
    /*
    finally {
        try{
            // Always close properly
            this.close();
        }catch(Exception e){
            e.printStackTrace();
        }
    }*/
    return isSuccess;
}

public void setString(int index,String value) throws SQLException {
    pstmt.setString(index, value);
}

public void setInt(int index,int value) throws SQLException {
    pstmt.setInt(index,value);
}

public void setBoolean(int index,boolean value) throws SQLException {
    pstmt.setBoolean(index,value);
}

public void setDate(int index,java.sql.Date value) throws SQLException {
    pstmt.setDate(index,value);
}

public void setLong(int index,long value) throws SQLException {
    pstmt.setLong(index,value);
}

public void setFloat(int index,float value) throws SQLException {
    pstmt.setFloat(index,value);
}

public void setBytes(int index,byte[] value) throws SQLException{
    pstmt.setBytes(index,value);
}

public void addBatch() throws SQLException{
    pstmt.addBatch();
}

```

```

}

public void prepareStatement(String sql) throws SQLException {
    if(this.conn != null)
        this.prepstmt = this.conn.prepareStatement(sql);
}

public void prepareStatement(String sql, int resultSetType, int resultSetConcurrency) throws
SQLException {
    this.prepstmt = this.conn.prepareStatement(sql, resultSetType, resultSetConcurrency);
}

public ResultSet executeQuery() throws SQLException {
    if (prepstmt != null) {
        return prepstmt.executeQuery();
    }
    else return null;
}

public int executeUpdate() throws SQLException {
    int count = 0;
    if (this.prepstmt != null)
        count = this.prepstmt.executeUpdate();
    // System.out.println(count);
    return count;
}

public int[] executeBatch() throws SQLException {
    int count[] = null;
    if (this.prepstmt != null)
        count = this.prepstmt.executeBatch();
    return count;
}

public void clearParameters()throws SQLException
{
    prepstmt.clearParameters();
    prepstmt=null;
}

public PreparedStatement getPreparedStatement() {
    return prepstmt;
}

```



```

public Statement getStatement() {
    return stmt;
}

public boolean prepareCall(String process){ //throws Exception
    boolean isSuccess = false;
    //String procedure = "{call p_test2 (?) }";
    //String procedure = "{call "+process+" }";
    String procedure = "{select "+process+"}";

    System.out.print(procedure);
    try{

        CallableStatement cstmt = this.conn.prepareCall(procedure);
        cstmt.setString(1,"dddddd");
        cstmt.setString(2,"fffff");
        cstmt.executeUpdate();

        //rs = cstmt.executeQuery();
        isSuccess = true;

    }catch(SQLException e) {
        e.printStackTrace();
    }
    return isSuccess;
}

public boolean Transaction(String sql){ //throws Exception
    boolean isSuccess = false;
    try{
        this.prepstmt = this.conn.prepareStatement(sql);
    }catch(SQLException e){
        e.printStackTrace();
    }
}

/*
connection.beginTransaction();
try{
    connection.commit();
}catch(SQLException e) {
    connection.rollback();
    e.printStackTrace();
}
}
*/

```

```

        return isSuccess;
    }

    public void close() throws Exception{
        //System.out.println(this.conn.getCatalog());
        //System.out.println(preparedStatement.EXECUTE_FAILED);
        if (this.rs != null) {
            this.rs.close();
            this.rs = null;
        }
        if (this.stmt != null) {
            this.stmt.close();
            this.stmt = null;
        }
        if (this.preparedStatement != null) {
            this.preparedStatement.close();
            this.preparedStatement = null;
        }
        if (conn!=null)
        {

            //this.conn.close();
            dcm.freeConnection(conn);
            //dcm.freeConnection("PostgreSQL",conn);
            dcm.release();
            //this.conn = null;
        }
        //System.out.println(new java.util.Date() +": Database Connection close :"+this.conn+
        "|"+this.conn.isClosed());
    }

    public static void main(String args[]) {
        System.out.println("Starting DBConnect...");
        //DBConnect pg = new DBConnect();
        Connection conn = null;
        Statement stmt = null;
        PreparedStatement preparedStatement = null;
        try{
            Jdbc3PoolingDataSource source = new Jdbc3PoolingDataSource();
            source.setDataSourceName("A Data Source");
            source.setServerName("192.168.0.1");
            source.setDatabaseName("netkiller");
            source.setUser("netkiller");
            source.setPassword("chen");

```

```

source.setMaxConnections(10);
source.setInitialConnections(5);

//new InitialContext().rebind("DataSource", source);
//DataSource source = (DataSource)new InitialContext().lookup("DataSource");

int n = 1;

for(int i=0;i<100000;i++){
//source.setLoginTimeout();
conn = source.getConnection();
//prepstmt.clearParameters();
prepstmt = conn.prepareStatement("select * from \"users\" where id=?");
//System.out.println(conn.nativeSQL("select sss from users"));
prepstmt.setInt(1,1);
ResultSet rs = prepstmt.executeQuery();
System.out.println(prepstmt.getQueryTimeout());
if(rs!=null) {
    while(rs.next()){
        System.out.print(n + "|");
        System.out.print(rs.getInt(1) + "|");
        System.out.print(rs.getString(2) + "|");
        System.out.print(rs.getString(3) + "|");
        System.out.print(rs.getString(4) + "|");
        System.out.print(rs.getString(5) + "|");
        System.out.print(rs.getString(6) + "|");
        System.out.println();
        n++;
    }

}
conn.close();
}
}catch(Exception e)    {
    e.printStackTrace();
}

//pg.getConnection();

//System.out.println(pg.executeQuery("select * from siteuser"));
//pg.executeUpdate();

```

/\*

```

try{
pg.prepareStatement("select * from \"users\" where id=?");
pg.setInt(1,1);
ResultSet rs = pg.executeQuery();

    if(rs!=null) {

        while(rs.next()){
            System.out.print(rs.getInt(1) + "|");
            System.out.print(rs.getString(2) + "|");
            System.out.print(rs.getString(3) + "|");
            System.out.print(rs.getString(4) + "|");
            System.out.print(rs.getString(5) + "|");
            System.out.print(rs.getString(6) + "|");
            System.out.println();
        }

    }
    pg.close();
} catch(Exception e) {
    e.printStackTrace();
}
System.out.println(org.postgresql.util.UnixCrypt.crypt("hi"));
//System.out.println(org.postgresql.util.MD5Digest.encode("hi"));

*/
}
}

```

### 13.5.3 SQL

```

package netkiller.database;

import java.util.*;

public class Database extends DBConnect{

    private String sqlFrom = null;
    private String sqlWhere = null;
    private String sqlOrder = null;
    private String sqlGroup = null;
    private String sqlLimit = null;
    private String sqlOffset = null;

```

```
//private HashMap sqlHashField = new HashMap();  
private Hashtable sqlHashField = new Hashtable();  
private Vector      sqlVectorField = new Vector();  
private StringBuffer sql = new StringBuffer();  
private boolean isError = false;
```

```
}
public void setLimit(int value){
    this.sqlLimit = value+"";
}
public void setOffset(int value){
    this.sqlOffset = value+"";
}
public void put(Object key,Object value){
    sqlHashField.put(key,value);
}
public void add(Object object){
    sqlVectorField.add(object);
}
public String Select(){
    sql = new StringBuffer(1024);
    sql.append("Select ");
    Enumeration enumer = sqlVectorField.elements();
    while(enumer.hasMoreElements()){
        sql.append(enumer.nextElement());
        if(enumer.hasMoreElements()) sql.append(",");
    }
    sql.append(" From "+ getFrom());
    if(sqlOrder != null) sql.append(" Where "+ getWhere());
    if(sqlOrder != null) sql.append(" Order By "+ sqlOrder);
}
```

```

        Map.Entry map = (Map.Entry)iterator.next();
        key = (String)map.getKey();
        value = (String)map.getValue();
        sql.append(key+"="+value);
        if(iterator.hasNext())sql.append(",");
    }
    if(sqlWhere != null) sql.append(" Where "+ sqlWhere);
    return sql.toString();
}
*/

```

```

public String Update(){
    sql = new StringBuffer(1024);
    Iterator iterator = sqlHashField.entrySet().iterator();
    sql.append("Update ");
    String key,value,swap = "";
    sql.append(getFrom()+ " Set ");
    while(iterator.hasNext()){
        Map.Entry map = (Map.Entry)iterator.next();
        key = (String)map.getKey();
        value = (String)map.getValue();

        if(iterator.hasNext()){
            swap = ","+key+"="+value + swap;
        }else{
            swap = key+"="+value + swap;
        }
    }
    sql.append(swap);
    if(sqlOrder != null) sql.append(" Where "+ getWhere());
    return isError?ERROR+sql.toString():sql.toString();
}

```

```

public String Insert(){
    sql = new StringBuffer(1024);
    Enumeration enumer = sqlHashField.keys();
    sql.append("Insert Into ");
    String key = "", value = "";
    sql.append(getFrom()+ "(");
    sqlHashField.size();
    while(enumer.hasMoreElements()){
        String tmp = (String)enumer.nextElement();
        if(enumer.hasMoreElements()){
            key = "," + tmp + key;

```

```

        value = "," + sqlHashField.get(tmp) + value;
    }else{
        key = tmp + key;
        value = sqlHashField.get(tmp) + value;
    }

    }
    sql.append(key);
    sql.append(" value(");
    sql.append(value);
    sql.append(")");

    return isError?ERROR+sql.toString():sql.toString();
}

```

```

public static void main(String[] args) {
    // TODO: Add your code here
    Database db = new Database();
    db.add("id");
    db.add("name");
    db.add("company");
    db.add("pages");

    db.put("id", "1");
    db.put("name", "netkiller");
    db.put("company", "none");
    db.put("number", "10");
    db.put("tel", "0755-1234666");
    db.put("fax", "0755-2556888");

    db.setFrom("ctoc.test");
    db.setWhere("chen = chen");
    db.setOrder("id");
    db.setGroup("name");
    db.setLimit(10);
    db.setOffset("0");

    String sql = db.Select();
    System.out.println(sql);
    System.out.println(db.Delete());
    System.out.println(db.Update());
    System.out.println(db.Insert());
}

```



```
}  
  
}
```

## 13.5.4 HTML

```
package netkiller.table;  
  
import java.sql.*;  
import java.util.*;  
import java.util.regex.*;  
import javax.servlet.jsp.*;  
import javax.servlet.ServletException;  
import javax.servlet.http.HttpServletRequest;  
import netkiller.util.*;  
import netkiller.database.*;  
  
public class BaseTable { //extends AnySQL  
  
    //AnySQL oTable = new AnySQL();  
    private static String oDatabase = null;  
    private static final String Auth = "cleartext"; //crypt  
    public static final String DESC = "DESC";  
    public static final String ASC = "ASC";  
    public static final String RADIO = "radio";  
    public static final String CHECKBOX = "checkbox";  
  
    private DBConnect dbc = null;  
    private ResultSet rs = null;  
    private PageContext pageContext = null;  
    private Vector vError = new Vector();  
  
    private static boolean debug = false;  
  
    private int nTableColumn= 0;  
    private int nTableRow = 0;  
    private int nColumnCount = 0;  
    private int nRowCount = 0;  
  
    private boolean isChoose = false;  
    private String strChooseStyle = "";
```

```

private boolean isDelete      = false;
// private String strDeleteLink = null;
private boolean isEdit       = false;
private String strEditLink   = null;
private boolean isFilter     = false;

private String[][] aTableData      = null;
private String[] strTableHeader    = null;
private String[] strTableField    = null;
private int      nSqlLimit         = 1;
private int      nSqlOffset        = 0;
private static String strSQL       = null;
private String   strWhere          = null;
private int      nPage             = 0;

private String sqlWhere = null;
private String sqlOrder = "";
private String sqlLimit = "";
private String sqlOffset= "";

private Vector v = new Vector();
private Vector cursor = new Vector();
private Vector filter = new Vector();

// HTML
// private String strTableName = null;
// private String strCssClass   = null;
// private int    nHeight       = 0;
// private int    nWidth        = 0;

public BaseTable() {
}

public BaseTable(String odb) {
    this.setDatabaseHandle(odb);
}

public void setDatabaseHandle(String odb){
    this.oDatabase = odb;
}

public String getDatabaseHandle(){
    return this.oDatabase;
}

```

```

//
//javax.servlet.jsp.PageContext pageContext
public void setPageContext(PageContext pc){
    pageContext = pc;
    try{
        ServletRequest request = pageContext.getRequest();
        jspParameter jsppar = new jspParameter(request);
        int page = jsppar.getInt("page",0);

        String filter      = jsppar.getString("filter");
        String condition    = jsppar.getString("condition");
        String key          = jsppar.getString("key");

        this.setPage(page);
        this.setWhere(filter,condition,key);
        //this.setWhere("id","like","5");
        //this.setWhere("id = 67");
        this.MakeHtmlTable();

        if(debug){
            pageContext.getOut().print("SQL:"+this.getSQL()+"<br>");
            pageContext.getOut().print("Table:"+this.sqlAnalyzer(this.getSQL())+"<br>");
            pageContext.getOut().print("Where:"+this.getWhere()+"<br>");
            pageContext.getOut().print("Order:"+this.getOrderBy()+"<br>");
            pageContext.getOut().print("Limit:"+this.getLimit()+"<br>");
            pageContext.getOut().print("Offset:"+this.getOffset()+"<br>");
            pageContext.getOut().print("Count:"+this.getRowCount()+" sql = select count(*) from "+this.sqlAnalyzer
            this.getException();
        }
    }catch(Exception e){
        e.printStackTrace();
    }
}

// Debug
public void setDebug(boolean debug){
    System.out.println("SQL:"+this.getSQL()+"<br>");
    this.debug = debug;
}

//          radio,checkbox
public void setChoose(boolean bool,String style){
    this.isChoose = bool;
}

```

```

        this.strChooseStyle = style;
        if(this.strChooseStyle.equals("radio")){
        }else if(this.strChooseStyle.equals("checkbox")){
        }else{
            this.isChoose = false;
        }

    }

    public String getChooseStyle(String id){
        String tmp = null;
        if(this.strChooseStyle.equals("radio")){
            tmp = "<input type=\"radio\" name=\"id\" value=\""+id+"\">";
        }else if(this.strChooseStyle.equals("checkbox")){
            tmp = "<input type='checkbox' name='id' value='"+id+"'>";
        }else{
        }
        return tmp;
    }

    /*
        if(isEdit){
            //v.add("<td><a href=\"javascript:OpenDialog('http://localhost:8080/autosite/member/'+strEditLink+'?id="+id+"')\">"+this.getHtmlTd()+"><a href="+strEditLink+rs.getString(1)+">"+rs.getString(1)+"</a></td>");
            v.add("<td "+this.getHtmlTd()+"><a href="+strEditLink+rs.getString(1)+">"+rs.getString(1)+"</a></td>");
        }
        if(isDelete){
            //v.add("<td><input type=\"button\" name=\"Delete\" value=\"Delete\" onclick=\"javascript:mysubmit('del_selected')\">"+this.getHtmlTd()+"><input type=\"button\" name=\"Delete\" value=\"Delete\" onclick=\"javascript:mysubmit('del_selected')\">"+this.getHtmlTd()+">";
            //v.add("<td><a href="+strDeleteLink+aTableData[0][r]+">"+rs.getString(1)+"</a></td>"); //onclick="javascript:mysubmit('del_selected')"
            v.add("<td "+this.getHtmlTd()+">"+rs.getString(1)+" align='center'><a href="+this.getDeleteLink(strDeleteLink+aTableData[0][r]+">"+rs.getString(1)+"</a></td>"); //onclick="javascript:mysubmit('del_selected')"
        }
    */
    //
    public void setDelete(boolean bool){
        this.isDelete = bool;
    }
    //
    public void setEdit(boolean bool,String url){
        this.isEdit = bool;
        strEditLink = url;
    }

    public void setFilter(boolean filter){
        this.isFilter = filter;
    }
}

```

```

public boolean getFilter(){
    return this.isFilter;
}

//TableModel
public void setTableModel(){ }
public void setSize(){ }
public void getSize(){ }

public void setBorder(int border){
    this.htmlTableBorder = " border="+border;
}
public String getBorder(){
    return this.htmlTableBorder;
}
public void setFont(){ }
public void getFont(){ }
public void setGridground(){ }
public void getGridground(){ }
public void setForeground(){ }
public void getForeground(){ }
public void setBackground(){ }
public void getBackground(){ }
/**
 *
 **/
public void setLayout(String css){ }
public void getLayout(String css){ }
public void setMaximumSize(){ }
public void setMinimumSize(){ }

public void setTableHeader(String[] sth){
    strTableHeader = sth;
}
public String[] getTableHeader(){
    return strTableHeader;
}

public void setPage(int page){
    this.nPage = page;
}
//
public int getPage(){
    return this.nPage;
}

```

```

}

/**
 * <table></table>,<tr></tr>,<td></td>
 **/
private String htmlTableCaption = null;
private String htmlTable = "";
private String htmlTableBorder = "";
private String htmlTableHeight = "";
private String htmlTableWidth = "";
private String htmlTr = "";
private String htmlTd = "";

private String htmlRowHeight= "";
private String htmlRowWidth= "";

private String htmlHeaderTr = "";
private String htmlHeaderTd = "";

public void setHtmlTable(String html){
    this.htmlTable = html;
}
public void setHtmlTableCaption(String html){
    htmlTableCaption = "<caption>";
    htmlTableCaption += html;
    htmlTableCaption += "</caption>";
}
public String getHtmlTableCaption(){
    if(htmlTableCaption == null)htmlTableCaption = "";
    return htmlTableCaption;
}
public void setHtmlTableHeaderTr(String html){
    this.htmlHeaderTr= html;
}
public String getHtmlTableHeaderTr(){
    return this.htmlHeaderTr;
}

public void setHtmlTableHeaderTd(String html){
    this.htmlHeaderTd = html;
}
public String getHtmlTableHeaderTd(){
    return this.htmlHeaderTd;
}
}

```

```

public void setHtmlTr(String html){
    this.htmlTr    = html;
}
public void setHtmlTd(String html){
    this.htmlTd    = html;
}
public String getHtmlTable(){
    return this.htmlTable + getBorder();
}
public String getHtmlTr(){
    return this.htmlTr;
}
public String getHtmlTd(){
    return this.htmlTd + getRowHeight() + getRowWidth();
}
public void setRowHeight(int high){
    this.htmlRowHeight = " height="+high;
}
public String getRowHeight(){
    return this.htmlRowHeight;
}
public void setRowWidth(int width){
    this.htmlRowWidth = " width="+width;
}
public String getRowWidth(){
    return this.htmlRowWidth;
}

private HashMap hashAddHtmlColumn = new HashMap();
private HashMap htmlTableRow=    new HashMap();
private HashMap htmlColumn=    new HashMap();

//HTML Row Operator
//public void addRow(String html){ }
//public String getRow(){ }

public void setHtmlTableRow(int col,String beginmarkup,String field,String endmarkup,String end){
    String key = col + "";
    String value = beginmarkup+"<table:row key="+field+" />"+endmarkup+"<table:row value />"+end;
    this.htmlTableRow.put(key,value);
}
public void setHtmlTableRow(int col,String begin,String end){
    String key = col + "";

```

```

        String value = begin+"<table:row value />" + end;
        this.htmlTableRow.put(key,value);
    }

    //HTML Comlumn Operator

    /**
     * Hide Column
     **/
    private int arrayHideColumn[] = null;

    public void setHideColumn(int col){
        arrayHideColumn = new int[1];
        arrayHideColumn[0] = col;
    }
    public void setHideColumn(int col[]){
        arrayHideColumn = col;
    }
    public boolean getHideColumn(int col){
        boolean isHide = false;
        if(arrayHideColumn == null){
            isHide = true;
        }else{
            for(int hide=0;hide<arrayHideColumn.length;hide++){
                if(arrayHideColumn[hide] == col){
                    isHide = false;
                    break;
                }else{
                    isHide = true;
                }
            }
        }
        return isHide;
    }

    public void addHtmlColumn(int key,String value){
        this.hashAddHtmlColumn.put(key + "",value);
    }
    public void addRs2HtmlColumn(int key,String begin0,String field,String begin1,String value,String end){
        value = begin0+"<!-- key='"+field+"' -->" + begin1 + "<!-- value='"+value+"' -->" + end;
        this.hashAddHtmlColumn.put(key + "",value);
    }
}

```



```

public void addHtmlColumn(int key,String begin0,String field,String begin1,String value,String end){
    value = begin0+"<!-- key="+field+" -->" +begin1+value+end;
    this.hashAddHtmlColumn.put(key + "",value);
}

public void addHtmlColumn(int key,String begin,String value,String end){
    value = begin+value+end;
    this.hashAddHtmlColumn.put(key + "",value);
}

public void setColumnWidth(){ }

public void setColumnCount(int number){
    nTableColumn= number;
}
public int getColumnCount(){
    return nTableColumn;
}
public void getColumnName(){ }
public void getColumnValue(){ }

public void setHtmlColumn(int col,String beginmarkup,String field,String endmarkup,String end){
    String key = col + "";
    String value = beginmarkup+"<!-- key="+field+" -->" +endmarkup+"<!-- value -->" +end;
    this.htmlColumn.put(key,value);
}
public void setHtmlColumn(int col,String begin,String end){
    String key = col + "";
    String value = begin+"<!-- value -->" +end;
    this.htmlColumn.put(key,value);
}
/**
 *
 */
// private int nSequence = 0;
private boolean isShowSequence = true;
public void setSequence(boolean bool){
    isShowSequence = bool;
}
/* public int getSequence(){
    return nSequence;
}
*/
/**

```

```

*
**/
public void setRowCount(int number){
    nTableRow = number;
}
public int getRowCount(){
    return nTableRow;
}
public int getCount(String strTable){
    int nCount = 0;
    try{
        dbc = new DBConnect(oDatabase);
        dbc.prepareStatement("select count(*) from "+strTable+getWhere());
        rs = dbc.executeQuery();
        if(rs != null){
            while(rs.next()){
                nCount = rs.getInt(1);
            }
            setRowCount(nCount);
        }
    }
    catch(Exception e){
        e.printStackTrace();
    }
    finally{
        try{
            if(rs != null)rs.close();
            dbc.close();
        }catch(Exception e){
            e.printStackTrace();
        }
    }
    return nCount;
}

public void setValueAT(){ }
public void getCursor(){ }

// public void setListData(Vector){ }
// public void setListData(String[]){ }

public boolean setVisible(boolean bool){
    return true;
}

```

```

}
public void visible(){
    setVisible(true);
}
public void hide(){
    setVisible(false);
}

public void enable(){};
public void disable(){};

public void setSQL(String sql){
    this.strSQL = sql;
}

public String getSQL(){
    return this.strSQL + getWhere() + " " + sqlOrder + " " + sqlLimit + " " + sqlOffset;
}

public void setWhere(String field,String cond,String val){
    if(val != null && field != null && cond != null){
        if(cond.equals("like")){
            this.strWhere = field+" "+cond+" "+"%"+val+"%";
        }else{
            this.strWhere = field+" "+cond+" "+"'+val+'";
        }
        if(sqlWhere != null)
            sqlWhere += " And "+strWhere;
        else
            sqlWhere = this.strWhere;
        this.isFilter = true;
    }
}

public void setWhere(String where){
    if(this.sqlWhere == null){
        this.sqlWhere = where;
    }else{
        this.sqlWhere = " and " + where;
    }
}

public String getWhere(){

```

```

String tmp = null;
if(sqlWhere == null){
    tmp = "";
}else{
    tmp = " Where "+sqlWhere;
}
return tmp;
}

public void setOrderBy(String order,String sort){
    if(order != null){
        if(sort.equals(DESC)){
            this.sqlOrder += " ORDER BY "+ order + " DESC";
        }else{
            this.sqlOrder += " ORDER BY "+ order + " ASC";
        }
    }else
        sqlOrder = "";
}

public String getOrderBy(){
    return sqlOrder;
}

public void setSort(String order,String sort){
    if(order != null){
        if(sort.equals(DESC)){
            this.sqlOrder += " ORDER BY "+ order + " DESC";
        }else{
            this.sqlOrder += " ORDER BY "+ order + " ASC";
        }
    }else
        sqlOrder = "";
}

public void setLimit(int limit){
    this.nSqlLimit = limit;
    this.sqlLimit = " limit "+limit;
}

public int getLimit(){
    return this.nSqlLimit;
}

public void setOffset(int offset){

```

```

        this.nSqlOffset    = offset;
        if(this.isFilter == false){
            sqlOffset = " offset "+offset;
        }
    }
    public int getOffset(){
        return this.nSqlOffset;
    }

    public String sqlAnalyzer(String sql){
        String tmp= sql.substring(sql.toLowerCase().indexOf(" from ")+6);
        String value = "";
        if(tmp.indexOf(" ") == -1){
            value = tmp.substring(0,sql.length());
        }else{
            value = tmp.substring(0,tmp.indexOf(" "));
        }
        return value;
    }

    public java.sql.ResultSet TableResultSet(){
        //this.setOffset(this.nSqlLimit*this.nPage);
        this.getCount(sqlAnalyzer(this.getSQL()));
        try{
            this.setOffset(this.nSqlLimit*this.nPage);
            dbc = new DBConnect();
            dbc.prepareStatement(this.getSQL());
            rs = dbc.executeQuery();

            //while(rs.next()){
            //System.out.println("test:" +rs.getString(1));
            //}
            vError.add("TableResultSet()");
        }
        catch(Exception e){
            vError.add(e.toString());
        }
        return rs;
    }
    public String getJavaScript(){
        String script = "";
        script += "<script language=\"JavaScript\" type=\"text/JavaScript\"> \n";
        script += "<!-- \n";
        script += "function Delete(url){ \n";

```

```

script += "        if(confirm(\"                ?\")){ \n";
script += "            location.href=url; \n";
script += "        } \n";
script += "    } \n";
script += "!--> \n";
script += "</script> \n";
return script;
}
public Vector MakeHtmlTable(){
    v.add("<!-- Html Table        netkiller                netkillers@sohu.com --> \n");
    v.add("<!-- Begin --> \n");

    String field    = "";
    Pattern pattern = null;
    Matcher matcher  = null;
    int begin,end = 0;

    if(isDelete) v.add(this.getJavaScript());
    this.setOffset(this.nSqlLimit*this.nPage);
    this.getCount(sqlAnalyzer(this.getSQL()));
    try{
        dbc = new DBConnect();
        dbc.prepareStatement(this.getSQL());
        rs = dbc.executeQuery();

        ResultSetMetaData rsmd = rs.getMetaData() ;
        this.setColumnCount(rsmd.getColumnCount());
        //----- Header Begin -----
        if(strTableHeader == null){
            strTableHeader = new String[this.getColumnCount()];
            strTableField  = new String[this.getColumnCount()];
            for(int col=1;col<=this.getColumnCount();col++){
                strTableHeader[col-1] = rsmd.getColumnName(col);
                strTableField[col-1] = rsmd.getColumnName(col);
            }
        }else{
            strTableField  = new String[this.getColumnCount()];
            for(int col=1;col<=this.getColumnCount();col++){
                strTableField[col-1] = rsmd.getColumnName(col);
            }
        }
        //----- Header End -----
        if(rs!=null) {
            v.add("<table "+this.getHtmlTable()+"> \n");

```

```

v.add(getHtmlTableCaption() + "\n");
if(strTableHeader != null){
    v.add("<tr "+this.getHtmlTableHeaderTr()+"> \n");
    if(isShowSequence){
        v.add("\t<td "+this.getHtmlTableHeaderTd()+">&nbsp;  </td> \n");
    }
    for(int i=0;i<this.getTableHeader().length;i++){
        if(getHideColumn(i+1)){
            if(htmlTableRow.containsKey("0")){

                field = ((String)this.htmlTableRow.get("0"));
                if(field.indexOf("key=") != -1){
                    begin = field.indexOf("<table:row key=");
                    end = field.indexOf("/>",begin);
                    String sql_field = field.substring(begin+16,end);
                    pattern = Pattern.compile("<table:row key='(.*)' />");
                    matcher = pattern.matcher(field);
                    field = matcher.replaceAll(sql_field);
                    field = field.replaceAll("<table:row value />",this.getTableHeader()[i]);
                }else{
                    field = field.replaceAll("<table:row value />",this.getTableHeader()[i]);
                }

                v.add("  <td "+this.getHtmlTableHeaderTd()+">"+field+"</td> \n");
            }else{
                v.add("  <td "+this.getHtmlTableHeaderTd()+">"+this.getTableHeader()[i+"</td>
            }

        }
    }
    v.add("</tr> \n");
}

if(isEdit){
    //v.add("<td><a href='\"javascript:OpenDialog('http://localhost:8080/autosite/member/\""+strEditLink+"\"')\">"+this.getHtmlTd()+"><a href='\""+strEditLink+rs.getString(1)+\"\">"+rs.getString(1)+"</a></td>");
    this.addHtmlColumn(-1,"<a href='\""+strEditLink,"id","\">"+rs.getString(1)+"</a>");
}

if(isDelete){
    //v.add("<td><input type='\"button\"' name='\"Delete\"' value='\""+rs.getString(1)+"\" onclick='\"javascript:mysubmit(\""+strDeleteLink+aTableData[0][r]+\"\">"+rs.getString(1)+"</a></td>"); //onclick='\"javascript:mysubmit(\""+strDeleteLink+aTableData[0][r]+\"\">"+rs.getString(1)+"</a></td>");
    //v.add("<td "+this.getHtmlTd()+\" align='center'><a href='\""+this.getDeleteLink(strDeleteLink+aTableData[0][r]+\"\">"+rs.getString(1)+"</a>");
    //onclick='\"javascript:mysubmit('del_selected')\"");
    this.addHtmlColumn(-2,"<a href='\"javascript:Delete('?op=delete&id="+rs.getString(1)+"\">"+rs.getString(1)+"</a>");
}

```

```

    }

    boolean bool = false;
    int row = 1;
    while(rs.next()){
        v.add(" <tr "+this.getHtmlTr()+> \n");
                if(isShowSequence){
                    v.add("\t<td "+this.getHtmlTd()+><center>"+row+"</center></td> \n");
                }
        for(int col=1;col<=this.getColumnCount();col++){
            if(getHideColumn(col)){ //
                String key = col + "";
                bool = htmlColumn.containsKey(key);
                if(bool){ //
                    field = ((String)this.htmlColumn.get(key));
                    begin = field.indexOf("key=");
                    if(begin != -1){
                        begin = field.indexOf("<!-- key=");
                        end = field.indexOf(" -->",begin);
                        String sql_field = field.substring(begin+9,end);
                        field = field.replaceAll("<!-- key="+sql_field+" -->",rs.getString(sql_field));
                    }
                    field = field.replaceAll("<!-- value -->",rs.getString(col));
                    v.add(" <td "+this.getHtmlTd()+>"+ field +</td> \n");
                }else if(htmlTableRow.containsKey(row+"")){
                    field = ((String)this.htmlTableRow.get(row+""));
                    if(field.indexOf("key=") != -1){
                        begin = field.indexOf("<table:row key=");
                        end = field.indexOf("' />",begin);
                        String sql_field = field.substring(begin+16,end);
                        pattern = Pattern.compile("<table:row key='(.*)' />");
                        matcher = pattern.matcher(field);
                        field = matcher.replaceAll(rs.getString(sql_field));
                    }
                    field = field.replaceAll("<table:row value />",rs.getString(col));
                    v.add(" <td "+this.getHtmlTd()+>"+ field +</td> \n");
                }else{
                    v.add(" <td "+this.getHtmlTd()+>"+ rs.getString(col) + "&nbsp;</td> \n");
                }
            }
            // add table column
            bool = hashAddHtmlColumn.containsKey(key);
            if(bool){
                field = (String)this.hashAddHtmlColumn.get(key);
                begin = field.indexOf("key=");

```



```

        if(begin == -1){
            v.add("  <td "+this.getHtmlTd()+">"+(String)this.hashAddHtmlColumn.get(k)
        }else{
            begin = field.indexOf("<!-- key=");
            end = field.indexOf(" -->",begin);
            String sql_field = field.substring(begin+10,end);
            field = field.replaceAll("<!-- key="+sql_field+" -->",rs.getString(sql_field));

            begin = field.indexOf("<!-- value=");
            if(begin != -1){
                end = field.indexOf(" -->",begin);
                sql_field = field.substring(begin+12,end);
                field = field.replaceAll("<!-- value="+sql_field+" -->",rs.getString(sql_fie
            )
            v.add("  <td "+this.getHtmlTd()+">" + field + "</td> \n");
        }
    }
}
if(col == this.getColumnCount()){
    if(hashAddHtmlColumn.containsKey("-1")){
        field = (String)this.hashAddHtmlColumn.get("-1");
        begin = field.indexOf("key=");
        if(begin == -1){
            v.add("  <td "+this.getHtmlTd()+">"+(String)this.hashAddHtmlColumn.g
        }else{
            begin = field.indexOf("<!-- key=");
            end = field.indexOf(" -->",begin);
            String sql_field = field.substring(begin+10,end);
            field = field.replaceAll("<!-- key="+sql_field+" -->",rs.getString(sql_fie

            begin = field.indexOf("<!-- value=");
            if(begin != -1){
                end = field.indexOf(" -->",begin);
                sql_field = field.substring(begin+12,end);
                field = field.replaceAll("<!-- value="+sql_field+" -->",rs.getString(so
            )
            v.add("  <td "+this.getHtmlTd()+">" + field + "</td> \n");
        }
    }
}
if(hashAddHtmlColumn.containsKey("-2")){
    field = (String)this.hashAddHtmlColumn.get("-2");
    begin = field.indexOf("key=");
    if(begin == -1){
        v.add("  <td "+this.getHtmlTd()+">"+(String)this.hashAddHtmlColumn.g
    }else{

```

```

begin = field.indexOf("<!-- key=");
end = field.indexOf("-->",begin);
String sql_field = field.substring(begin+10,end);
field = field.replaceAll("<!-- key="+sql_field+" -->",rs.getString(sql_field));

begin = field.indexOf("<!-- value=");
if(begin != -1){
    end = field.indexOf("-->",begin);
    sql_field = field.substring(begin+12,end);
    field = field.replaceAll("<!-- value="+sql_field+" -->",rs.getString(sql_field));
}
v.add("    <td "+this.getHtmlTd()+">" + field + "</td> \n");
}
}
}
}

row ++;
v.add(" </tr> \n");
}
v.add("</table> \n");
}else{
    v.add(" ");
}

//=====
//
//=====

int npage = this.getPage();
int pagecount = 0;
if(this.getRowCount()%this.getLimit() == 0){
    pagecount = this.getRowCount()/this.getLimit();
}else{
    pagecount = this.getRowCount()/this.getLimit()+1;
}

cursor.add("<!-- Begin          --> \n");
cursor.add("<table border=0 style='border-collapse: collapse' cellpadding='2'> \n"); //align=right
    cursor.add("<tr> \n");
        cursor.add("<td>          "+ this.getRowCount() +"          </td>");

```



```

        filter.add("</td>");
        filter.add("<td>");
        filter.add("<select name='condition' id='condition'> \n");
            filter.add("<option value='like' selected>    </option> \n");
            filter.add("<option value='='>=</option> \n");
            filter.add("<option value='!='> </option> \n");
            filter.add("<option value='>'>></option> \n");
            filter.add("<option value='>='>=</option> \n");
            filter.add("<option value='<'><</option> \n");
            filter.add("<option value='<='><=</option> \n");
        filter.add("</select> \n");
        filter.add("</td>");
        filter.add("<td>");
        filter.add("<input name='key' type='text' id='key'>");
        filter.add("</td>");
        filter.add("<td>");
        filter.add("<input type='submit' name='Submit' value='    '>");
        filter.add("</td>");
        filter.add("</form>");
        filter.add("</table> \n");
    filter.add("<!-- ----- End Filter ----- --> \n");
}
catch(Exception e){
    v.add("                ");
    e.printStackTrace();
}
finally{
    try{
        if(rs != null)rs.close();
        dbc.close();
    }catch(Exception e){
        e.printStackTrace();
    }
}
v.add("<!-- End --> \n");
return v;
}
public boolean getHtmlStyleTable(){
    boolean isSuccess = false;
    Vector html = this.v;
    try{
        Enumeration enumer = html.elements();
        while(enumer.hasMoreElements()){
            String strHtml = (String) enumer.nextElement();

```

```

        pageContext.getOut().print(strHtml);
    }
    isSuccess = true;
} catch(Exception e){
    System.out.print(e.toString());
}
return isSuccess;
}

public boolean getCursorControlKeys(){
    boolean isSuccess = false;
    try{
        Enumeration enumer = this.cursor.elements();
        while(enumer.hasMoreElements()){
            String strHtml = (String) enumer.nextElement();
            pageContext.getOut().print(strHtml);
        }
        isSuccess = true;
    } catch(Exception e){
        e.printStackTrace();
    }
    return isSuccess;
}

public void getFilterControl(){
    try{
        Enumeration enumer = this.filter.elements();
        while(enumer.hasMoreElements()){
            String strHtml = (String) enumer.nextElement();
            pageContext.getOut().print(strHtml);
        }
    } catch(Exception e){
        e.printStackTrace();
    }
}

public String[][] getTableData(){
    int row = 0;
    String[][] data = null;
    this.setOffset(this.nSqlLimit*this.nPage);
    //this.setRowCount();
    this.getCount(sqlAnalyzer(this.getSQL()));
    try{
        dbc = new DBConnect(oDatabase);
    }
}

```

```

dbc.prepareStatement(this.getSQL());
//System.out.println(this.getSQL());
rs = dbc.executeQuery();

ResultSetMetaData rsmd = rs.getMetaData() ;
this.setColumnCount(rsmd.getColumnCount());
//data = new String[this.getColumnCount()][this.getRowCount()];
data = new String[this.getColumnCount()][this.nSqlLimit];
strTableHeader = new String[this.getColumnCount()];
for(int col=1;col<=this.getColumnCount();col++){
    strTableHeader[col-1] = rsmd.getColumnName(col);
}
if(rs!=null) {
    while(rs.next()){
        for(int col=1;col<=this.getColumnCount();col++){
            data[col-1][row]=rs.getString(col);
        }
        row++;
    }
}

if(row != this.getLimit()){
    String[][] tmp = new String[this.getColumnCount()][row];
    //tmp = data;

    for(int r=0;r<row;r++){
        for(int col=0;col<this.getColumnCount();col++){
            tmp[col][r] = data[col][r];
        }
    }
    data = tmp;
    /*
    for(int r=row;r<=this.getLimit();r++){
        for(int col=1;col<=this.getColumnCount();col++){
            data[col-1][r]="";
        }
    }
    */
}
rs.close();
}
catch(Exception e){
    e.printStackTrace();
}
}

```

```

        finally{
            try{
                if(rs != null)rs.close();
                dbc.close();
            }catch(Exception e){
                e.printStackTrace();
            }
        }
        return data;
    }
}

public Vector MakeHtmlTable(int page,int limit,String order,String seq){
    Vector v      = new Vector();
    String script = null;
    //String script_edit = "window.showModelessDialog(\"http://www.9812.net\",\"dialogwin\",\"scroll:0;status:0;help:1;resizable:1;dialogW
    String old_seq = seq;
    //this.getCount(this.sqlAnalyzer(this.getSQL()));

    if(seq.equals("ASC")){
        this.setSort(order,ASC);
        seq = "DESC";
    }else{
        this.setSort(order,DESC);           //      DESC  ASC
        seq = "ASC";
    }

    this.setLimit(limit);           //
    this.setOffset(this.nSqlLimit*page); //
    this.setPage(page);
    //v.add(this.strSQL);           //      SQL
    try{
        aTableData = getTableData();

        script = "<script language=\"JavaScript\"> \n";
        script += "<!-- \n";
        script += "function CheckAll(form){ \n";
        script += "    for(var i=0;i<form.elements.length;i++) { \n";
        script += "        var e = form.elements[i]; \n";
        script += "        if(e.name != 'chkall') \n";
        script += "            e.checked = form.chkall.checked; \n";
        script += "    } \n";
        script += "} \n";

        script += "function OpenDialog(url){ \n";
        script += "    window.showModelessDialog(url,'dialogwin','scroll:0;status:0;help:1;resizable:1;dialogW

```

```

script += "} \n";

script += "function Delete(url){ \n";
script += "    if(confirm(\"          ?\")){ \n";
script += "        location.href=url; \n";
script += "    } \n";
script += "} \n";

script += "function allChecked(radio){ \n";
script += "    for(i=0; i<radio.length; i++) \n";
script += "        if(radio[i].checked) \n";
script += "            return (true); \n";
script += "            return(false); \n";
script += "} \n";

script += "//--> \n";
script += "</script> \n";

v.add(script);

v.add("<table border='1' cellpadding='1' cellspacing='0' style='border-collapse: collapse' bordercolor='#111111");
v.add("<tr align=middle>");
if(isChoose && this.strChooseStyle.equals("checkbox")){
    v.add("<td><input type=\"checkbox\" name=\"chkall\" value=\"on\" onclick=\"CheckAll(this.form)\"></td>");
}else if(this.strChooseStyle.equals("radio")){
    v.add("<td></td>");
}
v.add("<td>    </td>");
for(int i=0;i<this.getTableHeader().length;i++){
    v.add("<td><a
href='?order="+this.getTableHeader()[i]+"&seq="+seq+"&page="+this.getPage()+">"+(this.getTableHeader()[i].equals(ord
":"   "):this.getTableHeader()[i])+"</a></td>");
}
// -----
v.add("</tr> \n");
//System.out.println(aTableData[0].length);

for(int r=0;r<aTableData[0].length;r++){
    v.add("<tr onmouseover=\"this.style.backgroundColor='#f2f2f2\" style=\"CURSOR: hand\" onmouseover");
    if(isChoose){
        v.add("<td>"+getChooseStyle(aTableData[0][r])+"</td>");
    }
}

```



```

        v.add("<td>" + (r+1) + "</td>");    //
        for(int c=0;c<aTableData.length;c++){
            v.add("<td>" + aTableData[c][r] + "</td>");
        }
/*
        if(isEdit){
            //v.add("<td><a href=\"javascript:OpenDialog('http://localhost:8080/autosite/member/'+strEditLink+
            v.add("<td><a href="+strEditLink+aTableData[0][r]+">    </a></td>");
        }
        if(isDelete){
            //v.add("<td><input type=\"button\" name=\"Delete\" value=\"    \" onclick=\"javascript:mymethod
            //v.add("<td><a href="+strDeleteLink+aTableData[0][r]+">    </a></td>"); //onclick="javascript:
            v.add("<td><a href="+this.getDeleteLink(strDeleteLink+aTableData[0][r]+">    </a></td>"); //o
        }
*/
        v.add("</tr> \n");
    }
    v.add("</table> \n");
    v.add("<BR>");
    //=====
    v.add("<table border=1 style='border-collapse: collapse' bordercolor='#111111' cellpadding='2' align=right> \n");
    v.add("<tr>");
    v.add("<td>    "+ this.getRowCount() + "    </td>");

    if(page == 0){
        v.add("<td><font face='Webdings' size='4'>7</font></td>");
        v.add("<td><font face='Webdings' size='4'>3</font></td>");
    }else{
        v.add("<td><a href='?page=0'><font face='Webdings' size='4'>7</font></a></td>");
        v.add("<td><a href='?order="+order+"&seq="+old_seq+"&page="+ (page-1) +"'><font face='Webdings'
    }
    int pagecount = 0;
    if(this.getRowCount()%this.getLimit() == 0){
        pagecount = this.getRowCount()/this.getLimit();
    }else{
        pagecount = this.getRowCount()/this.getLimit()+1;
    }

    v.add("<td>");
    v.add("<select name='page' id='page' size=0 onChange=window.location.href='?page='+this.options[this
    v.add("<option value='"+page+"' selected>["+(page+1)+"]</option>");
    for(int i=0;i<pagecount;i++){
        v.add("<option value='"+i+"'>"+(i+1)+"</option>");
    }

```



```

        v.add("</table> \n");
    }
    catch(Exception e){
        e.printStackTrace();
    }
    finally{
        try{
            }catch(Exception e){
                e.printStackTrace();
            }
        }
    return v;
}

public Vector getHtmlTables(int page,int limit,String order,String seq,String sql){
    this.setSQL(sql);
    this.setChoose(true,RADIO);
//    this.setEdit(true,"?op=edit&id=");
//    this.setDelete(true,"?op=delete&id=");
    return MakeHtmlTable(page,limit,order,seq);
}

//public void close() throws Exception {}
public void clear(){ }
public void close(){
    try{
        if(rs != null)rs.close();
        dbc.close();
        vError.add("close()");
    }catch(Exception e){
        vError.add(e.toString());
    }
}

public void getException(){
    AnalyseException ae = new AnalyseException(this.vError);
    try{
        pageContext.getOut().print(ae.getException());
    }catch(Exception e){ }
    System.out.println(ae.getException());
}

public static void main(String[] args) {
    BaseTable t = new BaseTable();
    //System.out.println(t.getCount("oa.away"));
}

```

```

String sql = "select * from btob.supply_demand";
//System.out.println(t.sqlAnalyzer("select id,subject,docfrom,created,type From oa.vdocument Where pid=38 limit

//t.getHtmlTable()
t.setSQL(sql);
try{
    ResultSet rs = t.TableResultSet();
    while(rs.next()){
        System.out.println(rs.getString(1));
    }
}catch(SQLException e){
    System.out.println(e.toString());
}
/*
t.setWhere("id like %4%");
t.setLimit(2);
t.setPage(0);
t.setHtmlTable("border=0");
t.setHtmlTr("bgcolor=#FF0000");
t.setHtmlTd("bordercolor=#0000FF");
t.setRowHeight(50);
t.setRowWidth(100);
t.setBorder(10);
String head[]={ "      ", "aaa", "bbb", "ccc", "ddd" };
t.setTableHeader(head);

t.setEdit(true, "?op=edit&id=");
t.setDelete(true, "?op=delete&id=");
t.setHideColumn(1);
t.setHtmlColumn(2, "<font color=blue>", "</font>");
//t.setHtmlColumn(3, "<font color=blue>", "</font>");
t.setHtmlColumn(5, "<font color=blue>", "</font>");
t.addHtmlColumn(3, "<a target=meeting href='details.jsp?id=", "id", ">", "subject", "</a>");
t.setHtmlColumn(4, "<a href='?id=", "id", ">", "</a>");
Vector v = t.MakeHtmlTable();
for(int i=0;i<v.size();i++){
    //System.out.print(v.get(i));
}
*/
//
// v = t.getFilterControl();
//
// for(int i=0;i<v.size();i++){
//
//     System.out.print(v.get(i));
//
// }
//
//t.getFilterControl();

```

```
t.getException();  
  
}  
}
```

## 13.5.5

.....

SSL

```
DROP TABLE images;  
CREATE TABLE images (  
    id Serial NOT NULL UNIQUE,  
    filename varchar(255) NOT NULL Default 'unknown',  
    img bytea,  
    size Integer NOT NULL Default '0',  
    created Timestamp Default current_timestamp::timestamp (0) without time zone,  
    modified Timestamp Default current_timestamp::timestamp (0) without time zone,  
    PRIMARY KEY ("id")  
);
```

JSP

```
<% @ page contentType="text/html; charset=utf-8" language="java" import="java.sql.*" errorPage="" %>  
<% @ page import="java.util.*"%>  
<% @ page import="java.io.*"%>  
<jsp:useBean id="test" scope="page" class="netkiller.database.DBConnect"/>  
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">  
<html>
```

```

<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
</head>

<body>
<%
    try{

        File file = new File("/tmp/1078215958235.jpg");
        FileInputStream fis = new FileInputStream(file);

        test.prepareStatement("insert into images(filename,img) values(?,?)");
        test.setString(1,file.getName());
        test.setBinaryStream(2,fis,fis.available());
        System.out.println(test.executeUpdate());
        fis.close();

        test.prepareStatement("select * from images");
        ResultSet rs = test.executeQuery();
        if(rs!=null) {
            while(rs.next()){
                String filename = rs.getString("filename");
                FileOutputStream file1 = new
FileOutputStream("/var/www/html/webapps/tmp/"+filename);
                byte[] imgBytes = rs.getBytes("img");
                file1.write(imgBytes);
                file1.close();
                out.print("");
            }
        }
        test.close();
    }catch(Exception e) {
        e.printStackTrace();
    }

%>
</body>
</html>

```

# 14 PHP

## 14.1 PHP 连接 PostgreSQL

PHP --with-pgsql

```
tar zxf php-?.?.?.tar.gz
cd php-?.?.?
./configure --prefix=/usr/local/php --with-apxs=/usr/local/apache/bin/apxs \
  --with-config-file-path=/usr/local/etc --enable-track-vars --with-xml \
  --with-pgsql --with-mysql \
  --with-ldap --enable-ftp --with-openssl --with-iconv --with-kerberos \
  --with-java=/usr/local/java
make
make install
cp php.ini-dist /usr/local/etc/php.ini
```

RPM php-pgsql

```
[root@linux root]# rpm -qa |grep php
php-imap-4.3.4-1.1
php-4.3.4-1.1
php-snmp-4.3.4-1.1
php-devel-4.3.4-1.1
php-mysql-4.3.4-1.1
php-xmlrpc-4.3.4-1.1
php-domxml-4.3.4-1.1
php-odbc-4.3.4-1.1
php-pgsql-4.3.4-1.1
php-ldap-4.3.4-1.1
[root@linux root]#
```

## 14.2 set CLIENT\_ENCODING TO 'GB18030';方案

PHP PostgreSQL PHPBB Class  
UNICODE

```
<?php
/*****
*
*          postgres7.php
*          -----
*   begin          : Saturday, Feb 13, 2001
*   copyright      : (C) 2001 The phpBB Group
*   email          : supportphpbb.com
*
*****/
```

```

*   $Id: postgres7.php,v 1.19 2002/03/05 02:19:38 psotfx Exp $
*
*****/

/*****

*
*   This program is free software; you can redistribute it and/or modify
*   it under the terms of the GNU General Public License as published by
*   the Free Software Foundation; either version 2 of the License, or
*   (at your option) any later version.
*
*****/

if(!defined("SQL_LAYER"))
{

define("SQL_LAYER","postgresql");

class sql_db
{

    var $db_connect_id;
    var $query_result;
    var $in_transaction = 0;
    var $row = array();
    var $rowset = array();
    var $rownum = array();
    var $num_queries = 0;

    //
    // Constructor
    //
    function sql_db($sqlserver, $sqluser, $sqlpassword, $database, $persistence = true)
    {
        $this->connect_string = "";

        if( $sqluser )
        {
            $this->connect_string .= "user=$sqluser ";
        }

        if( $sqlpassword )
        {
            $this->connect_string .= "password=$sqlpassword ";

```



```

    }

    if( $sqlserver )
    {
        if( ereg(":", $sqlserver) )
        {
            list($sqlserver, $sqlport) = split(":", $sqlserver);
            $this->connect_string .= "host=$sqlserver port=$sqlport ";
        }
        else
        {
            if( $sqlserver != "localhost" )
            {
                $this->connect_string .= "host=$sqlserver ";
            }
        }
    }

    if( $database )
    {
        $this->dbname = $database;
        $this->connect_string .= "dbname=$database";
    }

    $this->persistence = $persistence;

    $this->db_connect_id = ( $this->persistence ) ? pg_pconnect($this->connect_string) :
pg_connect($this->connect_string);

    return ( $this->db_connect_id ) ? $this->db_connect_id : false;
}

//
// Other base methods
//
function sql_close()
{
    if( $this->db_connect_id )
    {
        //
        // Commit any remaining transactions
        //
        if( $this->in_transaction )
        {

```

```

        @pg_exec($this->db_connect_id, "COMMIT");
    }

    if( $this->query_result )
    {
        @pg_freeresult($this->query_result);
    }

    return @pg_close($this->db_connect_id);
}
else
{
    return false;
}
}

//
// Query method
//
function sql_query($query = "", $transaction = false)
{
    //
    // Remove any pre-existing queries
    //
    unset($this->query_result);
    if( $query != "" )
    {
        $this->num_queries++;

        $query = preg_replace("/LIMIT ([0-9]+),([ 0-9]+)/", "LIMIT \\2 OFFSET \\1", $query);
        $encoding = " set CLIENT_ENCODING TO 'GB18030'";
        //$encoding = "SET CLIENT_ENCODING TO 'GBK'";
        $this->query_result = @pg_exec($this->db_connect_id, $encoding);

        if( $transaction == BEGIN_TRANSACTION && !$this->in_transaction )
        {
            $this->in_transaction = TRUE;

            if( !@pg_exec($this->db_connect_id, "BEGIN" ) )
            {
                return false;
            }
        }
    }
}

```

```

$this->query_result = @pg_exec($this->db_connect_id, $query);
if( $this->query_result )
{
    if( $transaction == END_TRANSACTION )
    {
        $this->in_transaction = FALSE;

        if( !@pg_exec($this->db_connect_id, "COMMIT") )
        {
            @pg_exec($this->db_connect_id, "ROLLBACK");
            return false;
        }
    }

    $this->last_query_text[$this->query_result] = $query;
    $this->rownum[$this->query_result] = 0;

    unset($this->row[$this->query_result]);
    unset($this->rowset[$this->query_result]);

    return $this->query_result;
}
else
{
    if( $this->in_transaction )
    {
        @pg_exec($this->db_connect_id, "ROLLBACK");
    }
    $this->in_transaction = FALSE;

    return false;
}
}
else
{
    if( $transaction == END_TRANSACTION && $this->in_transaction )
    {
        $this->in_transaction = FALSE;

        if( !@pg_exec($this->db_connect_id, "COMMIT") )
        {
            @pg_exec($this->db_connect_id, "ROLLBACK");
            return false;
        }
    }
}
}

```

```

    }

    return true;
}

//
// Other query methods
//
function sql_numrows($query_id = 0)
{
    if( !$query_id )
    {
        $query_id = $this->query_result;
    }

    return ( $query_id ) ? @pg_numrows($query_id) : false;
}

function sql_numfields($query_id = 0)
{
    if( !$query_id )
    {
        $query_id = $this->query_result;
    }

    return ( $query_id ) ? @pg_numfields($query_id) : false;
}

function sql_fieldname($offset, $query_id = 0)
{
    if( !$query_id )
    {
        $query_id = $this->query_result;
    }

    return ( $query_id ) ? @pg_fieldname($query_id, $offset) : false;
}

function sql_fieldtype($offset, $query_id = 0)
{
    if( !$query_id )
    {
        $query_id = $this->query_result;
    }

```

```

    }

    return ( $query_id ) ? @pg_fieldtype($query_id, $offset) : false;
}

function sql_fetchrow($query_id = 0)
{
    if( !$query_id )
    {
        $query_id = $this->query_result;
    }

    if($query_id)
    {
        $this->row = @pg_fetch_array($query_id, $this->rownum[$query_id]);

        if( $this->row )
        {
            $this->rownum[$query_id]++;
            return $this->row;
        }
    }

    return false;
}

function sql_fetchrowset($query_id = 0)
{
    if( !$query_id )
    {
        $query_id = $this->query_result;
    }

    if( $query_id )
    {
        unset($this->rowset[$query_id]);
        unset($this->row[$query_id]);
        $this->rownum[$query_id] = 0;

        while( $this->rowset = @pg_fetch_array($query_id, $this->rownum[$query_id],
PGSQL_ASSOC) )
        {
            $result[] = $this->rowset;
            $this->rownum[$query_id]++;
        }
    }
}

```

```

    }

    return $result;
}

return false;
}

function sql_fetchfield($field, $row_offset=-1, $query_id = 0)
{
    if( !$query_id )
    {
        $query_id = $this->query_result;
    }

    if( $query_id )
    {
        if( $row_offset != -1 )
        {
            $this->row = @pg_fetch_array($query_id, $row_offset, PGSQL_ASSOC);
        }
        else
        {
            if( $this->rownum[$query_id] )
            {
                $this->row = @pg_fetch_array($query_id, $this->rownum[$query_id]-1,
PGSQL_ASSOC);
            }
            else
            {
                $this->row = @pg_fetch_array($query_id, $this->rownum[$query_id],
PGSQL_ASSOC);

                if( $this->row )
                {
                    $this->rownum[$query_id]++;
                }
            }
        }

        return $this->row[$field];
    }

    return false;
}

```

```

}

function sql_rowseek($offset, $query_id = 0)
{
    if(!$query_id)
    {
        $query_id = $this->query_result;
    }

    if( $query_id )
    {
        if( $offset > -1 )
        {
            $this->rownum[$query_id] = $offset;
            return true;
        }
        else
        {
            return false;
        }
    }

    return false;
}

function sql_nextid()
{
    $query_id = $this->query_result;

    if($query_id && $this->last_query_text[$query_id] != "")
    {
        if(
            preg_match("/^INSERT[\t\n
                ]+INTO[\t\n
                ]+([a-z0-9\_\-]+)/is",
            $this->last_query_text[$query_id], $tablename) )
        {
            $query = "SELECT currval(' . $tablename[1] . "_id_seq') AS last_value";
            $temp_q_id = @pg_exec($this->db_connect_id, $query);
            if( !$temp_q_id )
            {
                return false;
            }

            $temp_result = @pg_fetch_array($temp_q_id, 0, PGSQL_ASSOC);

```

```

        return ( $temp_result ) ? $temp_result['last_value'] : false;
    }
}

return false;
}

function sql_affectedrows($query_id = 0)
{
    if( !$query_id )
    {
        $query_id = $this->query_result;
    }

    return ( $query_id ) ? @pg_cmdtuples($query_id) : false;
}

function sql_freeresult($query_id = 0)
{
    if( !$query_id )
    {
        $query_id = $this->query_result;
    }

    return ( $query_id ) ? @pg_freeresult($query_id) : false;
}

function sql_error($query_id = 0)
{
    if( !$query_id )
    {
        $query_id = $this->query_result;
    }

    $result['message'] = @pg_errormessage($this->db_connect_id);
    $result['code'] = -1;

    return $result;
}

} // class ... db_sql

} // if ... defined

```



```
?>
```

```
SET CLIENT_ENCODING TO 'value'; value
EUC_JP, SJIS, EUC_KR, UHC, JCHAB, EUC_CN, GBK, EUC_TW, BIG5, LATIN1 to LATIN10,
ISO_8859_5, ISO_8859_6, ISO_8859_7, ISO_8859_8, WN, ALT, KOI8, WN1256, TCVN, WN874,
GB18030, WN1250
```

```
GB2312 SET CLIENT_ENCODING TO 'GB2312'
```

```
member=> SET CLIENT_ENCODING TO 'GB2312';
ERROR: invalid value for option 'client_encoding': 'GB2312'
member=>
```

```
$encoding = " set CLIENT_ENCODING TO 'GB18030';" ; set CLIENT_ENCODING TO 'GB18030'
" ;"
```

## 14.3 convert()方案

```
convert()
select convert( , 'UNICODE', 'GBK') as desc from ;
select convert( using utf_8_to_gb18030) from ;
```

## 14.4 PHP iconv() 函数方案

```
PHP iconv()
PHP --with-iconv
function addUser($user,$passwd,$name,$nickname,$active="false",$email){
    global $db;
    $name = iconv( 'GB2312','UTF-8', $name );
    $sql = "insert into \"user\"(userid,passwd,name,nickname,active,email,question,answer)
values('$user','$passwd','$name','$nickname','$active','$email','question','answer')";
    if ( !($result = $db->sql_query($sql)) ){
        message_die(GENERAL_ERROR, 'Error in obtaining userdata', "", __LINE__, __FILE__, $sql);
    }
};

function getUserInfo($uid){
    global $db;
    $sql = "select * from \"user\" where userid='$uid'";

    // $codesql = "set CLIENT_ENCODING TO 'EUC_CN';";
    // $codesql = "set CLIENT_ENCODING TO 'BIG5';";
    // $codesql = "set CLIENT_ENCODING TO 'GB18030';";
    $db->sql_query($codesql);

    if ( !($result = $db->sql_query($sql)) ){
        message_die(GENERAL_ERROR, 'Could not \'select\' userdata', "", __LINE__, __FILE__, $sql);
    }
};
```

```

}
$count=$db->sql_numrows();

for ($i=0;$i<$count;$i++){
    if( $row = $db->sql_fetchrow($result) ){
        $tmp[0] = $row[0];
        $tmp[1] = $row[1];
        $tmp[2] = $row[2];
        $tmp[3] = iconv( 'UTF-8', 'GBK', $row[3] );
//        $tmp[3] = $row[3];
        $tmp[4] = $row[4];
        $tmp[5] = $row[5];
        $tmp[6] = $row[6];
        $tmp[7] = $row[7];
        $tmp[8] = $row[8];
        $tmp[9] = $row[9];
        $tmp[10] = $row[10];
    }
}
return $tmp;
};

```

## 14.5 在标准 I/O 上使用 Linux iconv 命令方案

Iconv          Linux

```

[root@linux script]# iconv --help
Usage: iconv [OPTION...] [FILE...]
Convert encoding of given files from one encoding to another.

Input/Output format specification:
  -f, --from-code=NAME      encoding of original text
  -t, --to-code=NAME        encoding for output

Information:
  -l, --list                 list all known coded character sets

Output control:
  -c                          omit invalid characters from output
  -o, --output=FILE          output file
  -s, --silent                suppress warnings
  --verbose                   print progress information

  -?, --help                 Give this help list
  --usage                     Give a short usage message

```

`-V, --version`                      Print program version

Mandatory or optional arguments to long options are also mandatory or optional for any corresponding short options.

Report bugs using the `glibbug' script to <bugs@gnu.org>.

```
[root@linux script]# iconv -f gb2312 -t big5 /tmp/gb.txt -o /tmp/big5.txt
```

                  PHP                                      --with-iconv  
LINUX iconv                      I/O                      /

```
function iconvs($from,$to,$gb){  
    $fp = popen( "echo \"\$gb\" | iconv -f $from -t $to", "r" );  
    while (!feof($fp)) {  
        $buffer .= fgets($fp, 4096);  
    }  
    pclose($fp);  
    return $buffer;  
}
```

```
function gb18030_utf8($gb){  
    $encode = $this->iconvs("GB18030","UTF-8",$gb);  
    return $encode;  
}
```

```
function utf8_gb18030($gb){  
    $encode = $this->iconvs("UTF-8","GB18030",$gb);  
    return $encode;  
}
```

iconv

```
[root@linux script]# iconv -l
```

The following list contain all the coded character sets known. This does not necessarily mean that all combinations of these names can be used for the FROM and TO command line parameters. One coded character set can be listed with several different names (aliases).

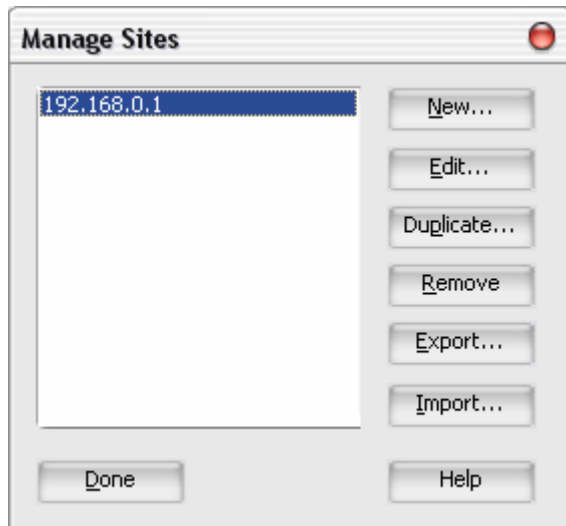
```
437, 500, 500V1, 850, 851, 852, 855, 856, 857, 860, 861, 862, 863, 864, 865,  
866, 869, 874, 904, 1026, 1046, 1047, 8859_1, 8859_2, 8859_3, 8859_4, 8859_5,  
8859_6, 8859_7, 8859_8, 8859_9, 10646-1:1993, 10646-1:1993/UCS4,  
ANSI_X3.4-1968, ANSI_X3.4-1986, ANSI_X3.4, ANSI_X3.110-1983, ANSI_X3.110,  
ARABIC, ARABIC7, ARMSII-8, ASCII, ASMO-708, ASMO_449, BALTIC, BIG-5,  
BIG-FIVE, BIG5-HKSCS, BIG5, BIG5HKSCS, BIGFIVE, BS_4730, CA, CN-BIG5, CN-GB,
```

CN, CP-AR, CP-GR, CP-HU, CP037, CP038, CP273, CP274, CP275, CP278, CP280, CP281, CP282, CP284, CP285, CP290, CP297, CP367, CP420, CP423, CP424, CP437, CP500, CP737, CP775, CP813, CP819, CP850, CP851, CP852, CP855, CP856, CP857, CP860, CP861, CP862, CP863, CP864, CP865, CP866, CP868, CP869, CP870, CP871, CP874, CP875, CP880, CP891, CP903, CP904, CP905, CP912, CP915, CP916, CP918, CP920, CP922, CP930, CP932, CP933, CP935, CP936, CP937, CP939, CP949, CP950, CP1004, CP1026, CP1046, CP1047, CP1070, CP1079, CP1081, CP1084, CP1089, CP1124, CP1129, CP1132, CP1133, CP1160, CP1161, CP1162, CP1163, CP1164, CP1250, CP1251, CP1252, CP1253, CP1254, CP1255, CP1256, CP1257, CP1258, CP1361, CP10007, CPIBM861, CSA7-1, CSA7-2, CSASCII, CSA\_T500-1983, CSA\_T500, CSA\_Z243.4-1985-1, CSA\_Z243.4-1985-2, CSA\_Z243.419851, CSA\_Z243.419852, CSDECMCS, CSEBCDICATDE, CSEBCDICATDEA, CSEBCDICCAFR, CSEBCDICDKNO, CSEBCDICDKNOA, CSEBCDICES, CSEBCDICESA, CSEBCDICESSE, CSEBCDICFISE, CSEBCDICFISEA, CSEBCDICFR, CSEBCDICIT, CSEBCDICPT, CSEBCDICUK, CSEBCDICUS, CSEUCKR, CSEUCPKDFMTJAPANESE, CSGB2312, CSHPROMAN8, CSIBM037, CSIBM038, CSIBM273, CSIBM274, CSIBM275, CSIBM277, CSIBM278, CSIBM280, CSIBM281, CSIBM284, CSIBM285, CSIBM290, CSIBM297, CSIBM420, CSIBM423, CSIBM424, CSIBM500, CSIBM851, CSIBM855, CSIBM856, CSIBM857, CSIBM860, CSIBM863, CSIBM864, CSIBM865, CSIBM866, CSIBM868, CSIBM869, CSIBM870, CSIBM871, CSIBM880, CSIBM891, CSIBM903, CSIBM904, CSIBM905, CSIBM918, CSIBM922, CSIBM930, CSIBM932, CSIBM933, CSIBM935, CSIBM937, CSIBM939, CSIBM943, CSIBM1026, CSIBM1124, CSIBM1129, CSIBM1132, CSIBM1133, CSIBM1160, CSIBM1161, CSIBM1163, CSIBM1164, CSIBM11621162, CSISO4UNITEDKINGDOM, CSISO10SWEDISH, CSISO11SWEDISHFORNAMES, CSISO14JISC6220RO, CSISO15ITALIAN, CSISO16PORTUGUESE, CSISO17SPANISH, CSISO18GREEK7OLD, CSISO19LATINGREEK, CSISO21GERMAN, CSISO25FRENCH, CSISO27LATINGREEK1, CSISO49INIS, CSISO50INIS8, CSISO51INISCYRILLIC, CSISO58GB1988, CSISO60DANISHNORWEGIAN, CSISO60NORWEGIAN1, CSISO61NORWEGIAN2, CSISO69FRENCH, CSISO84PORTUGUESE2, CSISO85SPANISH2, CSISO86HUNGARIAN, CSISO88GREEK7, CSISO89ASMO449, CSISO90, CSISO92JISC62991984B, CSISO99NAPLPS, CSISO103T618BIT, CSISO111ECMACYRILLIC, CSISO121CANADIAN1, CSISO122CANADIAN2, CSISO139CSN369103, CSISO141JUSIB1002, CSISO143IECP271, CSISO150, CSISO150GREEKCCITT, CSISO151CUBA, CSISO153GOST1976874, CSISO646DANISH, CSISO2022CN, CSISO2022JP, CSISO2022JP2, CSISO2022KR, CSISO2033, CSISO5427CYRILLIC, CSISO5427CYRILLIC1981, CSISO5428GREEK, CSISO10367BOX, CSISOLATIN1, CSISOLATIN2, CSISOLATIN3, CSISOLATIN4, CSISOLATIN5, CSISOLATIN6, CSISOLATINARABIC, CSISOLATINCYRILLIC, CSISOLATINGREEK, CSISOLATINHEBREW, CSKOI8R, CSKSC5636, CSMACINTOSH, CSNATSDANO, CSNATSSEFI, CSN\_369103, CSPC8CODEPAGE437, CSPC775BALTIC, CSPC850MULTILINGUAL, CSPC862LATINHEBREW, CSPCP852, CSSHIFTJIS, CSUCS4, CSUNICODE, CUBA, CWI-2, CWI, CYRILLIC, DE, DEC-MCS, DEC, DECMCS, DIN\_66003, DK, DS2089, DS\_2089, E13B, EBCDIC-AT-DE-A, EBCDIC-AT-DE, EBCDIC-BE, EBCDIC-BR, EBCDIC-CA-FR, EBCDIC-CP-AR1, EBCDIC-CP-AR2, EBCDIC-CP-BE, EBCDIC-CP-CA, EBCDIC-CP-CH, EBCDIC-CP-DK, EBCDIC-CP-ES, EBCDIC-CP-FI, EBCDIC-CP-FR, EBCDIC-CP-GB, EBCDIC-CP-GR, EBCDIC-CP-HE, EBCDIC-CP-IS,

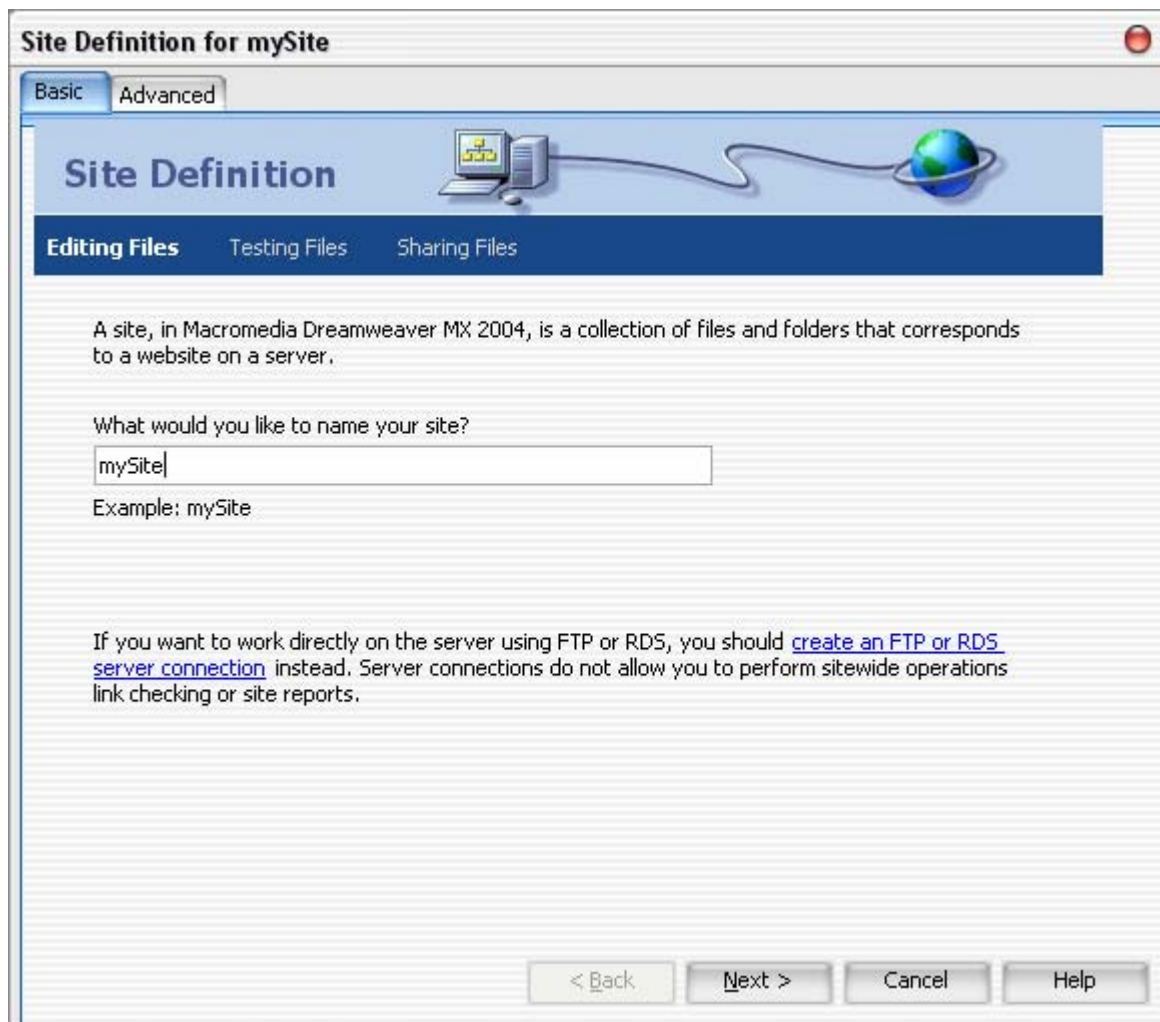
EBCDIC-CP-IT, EBCDIC-CP-NL, EBCDIC-CP-NO, EBCDIC-CP-ROECE, EBCDIC-CP-SE, EBCDIC-CP-TR, EBCDIC-CP-US, EBCDIC-CP-WT, EBCDIC-CP-YU, EBCDIC-CYRILLIC, EBCDIC-DK-NO-A, EBCDIC-DK-NO, EBCDIC-ES-A, EBCDIC-ES-S, EBCDIC-ES, EBCDIC-FI-SE-A, EBCDIC-FI-SE, EBCDIC-FR, EBCDIC-GREEK, EBCDIC-INT, EBCDIC-INT1, EBCDIC-IS-FRISS, EBCDIC-IT, EBCDIC-JP-E, EBCDIC-JP-KANA, EBCDIC-PT, EBCDIC-UK, EBCDIC-US, EBCDICATDE, EBCDICATDEA, EBCDICCAFR, EBCDICDKNO, EBCDICDKNOA, EBCDICES, EBCDICESA, EBCDICESSE, EBCDICFISE, EBCDICFISEA, EBCDICFR, EBCDICISFRISS, EBCDICIT, EBCDICPT, EBCDICUK, EBCDICUS, ECMA-114, ECMA-118, ECMA-128, ECMA-CYRILLIC, ECMACYRILLIC, ELOT\_928, ES, ES2, EUC-CN, EUC-JISX0213, EUC-JP, EUC-KR, EUC-TW, EUCCN, EUCJP, EUCKR, EUCTW, FI, FR, GB, GB2312, GB13000, GB18030, GBK, GB\_1988-80, GB\_198880, GEORGIAN-ACADEMY, GEORGIAN-PS, GOST\_19768-74, GOST\_19768, GOST\_1976874, GREEK-CCITT, GREEK, GREEK7-OLD, GREEK7, GREEK7OLD, GREEK8, GREEKCCITT, HEBREW, HP-ROMAN8, HPROMAN8, HU, IBM-856, IBM-922, IBM-930, IBM-932, IBM-933, IBM-935, IBM-937, IBM-939, IBM-943, IBM-1046, IBM-1124, IBM-1129, IBM-1132, IBM-1133, IBM-1160, IBM-1161, IBM-1162, IBM-1163, IBM-1164, IBM037, IBM038, IBM256, IBM273, IBM274, IBM275, IBM277, IBM278, IBM280, IBM281, IBM284, IBM285, IBM290, IBM297, IBM367, IBM420, IBM423, IBM424, IBM437, IBM500, IBM775, IBM813, IBM819, IBM850, IBM851, IBM852, IBM855, IBM856, IBM857, IBM860, IBM861, IBM862, IBM863, IBM864, IBM865, IBM866, IBM868, IBM869, IBM870, IBM871, IBM874, IBM875, IBM880, IBM891, IBM903, IBM904, IBM905, IBM912, IBM915, IBM916, IBM918, IBM920, IBM922, IBM930, IBM932, IBM933, IBM935, IBM937, IBM939, IBM943, IBM1004, IBM1026, IBM1046, IBM1047, IBM1089, IBM1124, IBM1129, IBM1132, IBM1133, IBM1160, IBM1161, IBM1162, IBM1163, IBM1164, IEC\_P27-1, IEC\_P271, INIS-8, INIS-CYRILLIC, INIS, INIS8, INISCYRILLIC, ISIRI-3342, ISIRI3342, ISO-2022-CN-EXT, ISO-2022-CN, ISO-2022-JP-2, ISO-2022-JP-3, ISO-2022-JP, ISO-2022-KR, ISO-8859-1, ISO-8859-2, ISO-8859-3, ISO-8859-4, ISO-8859-5, ISO-8859-6, ISO-8859-7, ISO-8859-8, ISO-8859-9, ISO-8859-10, ISO-8859-11, ISO-8859-13, ISO-8859-14, ISO-8859-15, ISO-8859-16, ISO-10646, ISO-10646/UCS2, ISO-10646/UCS4, ISO-10646/UTF-8, ISO-10646/UTF8, ISO-CELTIC, ISO-IR-4, ISO-IR-6, ISO-IR-8-1, ISO-IR-9-1, ISO-IR-10, ISO-IR-11, ISO-IR-14, ISO-IR-15, ISO-IR-16, ISO-IR-17, ISO-IR-18, ISO-IR-19, ISO-IR-21, ISO-IR-25, ISO-IR-27, ISO-IR-37, ISO-IR-49, ISO-IR-50, ISO-IR-51, ISO-IR-54, ISO-IR-55, ISO-IR-57, ISO-IR-60, ISO-IR-61, ISO-IR-69, ISO-IR-84, ISO-IR-85, ISO-IR-86, ISO-IR-88, ISO-IR-89, ISO-IR-90, ISO-IR-92, ISO-IR-98, ISO-IR-99, ISO-IR-100, ISO-IR-101, ISO-IR-103, ISO-IR-109, ISO-IR-110, ISO-IR-111, ISO-IR-121, ISO-IR-122, ISO-IR-126, ISO-IR-127, ISO-IR-138, ISO-IR-139, ISO-IR-141, ISO-IR-143, ISO-IR-144, ISO-IR-148, ISO-IR-150, ISO-IR-151, ISO-IR-153, ISO-IR-155, ISO-IR-156, ISO-IR-157, ISO-IR-166, ISO-IR-179, ISO-IR-193, ISO-IR-197, ISO-IR-199, ISO-IR-203, ISO-IR-209, ISO-IR-226, ISO646-CA, ISO646-CA2, ISO646-CN, ISO646-CU, ISO646-DE, ISO646-DK, ISO646-ES, ISO646-ES2, ISO646-FI, ISO646-FR, ISO646-FR1, ISO646-GB, ISO646-HU, ISO646-IT, ISO646-JP-OCR-B, ISO646-JP, ISO646-KR, ISO646-NO, ISO646-NO2, ISO646-PT, ISO646-PT2, ISO646-SE,

ISO646-SE2, ISO646-US, ISO646-YU, ISO2022CN, ISO2022CNEXT, ISO2022JP,  
ISO2022JP2, ISO2022KR, ISO6937, ISO8859-1, ISO8859-2, ISO8859-3, ISO8859-4,  
ISO8859-5, ISO8859-6, ISO8859-7, ISO8859-8, ISO8859-9, ISO8859-10,  
ISO8859-11, ISO8859-13, ISO8859-14, ISO8859-15, ISO8859-16, ISO88591,  
ISO88592, ISO88593, ISO88594, ISO88595, ISO88596, ISO88597, ISO88598,  
ISO88599, ISO885910, ISO885911, ISO885913, ISO885914, ISO885915, ISO885916,  
ISO\_646.IRV:1991, ISO\_2033-1983, ISO\_2033, ISO\_5427-EXT, ISO\_5427,  
ISO\_5427:1981, ISO\_5427EXT, ISO\_5428, ISO\_5428:1980, ISO\_6937-2,  
ISO\_6937-2:1983, ISO\_6937, ISO\_6937:1992, ISO\_8859-1, ISO\_8859-1:1987,  
ISO\_8859-2, ISO\_8859-2:1987, ISO\_8859-3, ISO\_8859-3:1988, ISO\_8859-4,  
ISO\_8859-4:1988, ISO\_8859-5, ISO\_8859-5:1988, ISO\_8859-6, ISO\_8859-6:1987,  
ISO\_8859-7, ISO\_8859-7:1987, ISO\_8859-8, ISO\_8859-8:1988, ISO\_8859-9,  
ISO\_8859-9:1989, ISO\_8859-10, ISO\_8859-10:1992, ISO\_8859-14,  
ISO\_8859-14:1998, ISO\_8859-15:1998, ISO\_9036, ISO\_10367-BOX, ISO\_10367BOX,  
ISO\_69372, IT, JIS\_C6220-1969-RO, JIS\_C6229-1984-B, JIS\_C62201969RO,  
JIS\_C62291984B, JOHAB, JP-OCR-B, JP, JS, JUS\_I.B1.002, KOI-7, KOI-8, KOI8-R,  
KOI8-T, KOI8-U, KOI8, KOI8R, KOI8U, KSC5636, L1, L2, L3, L4, L5, L6, L7, L8,  
L10, LATIN-GREEK-1, LATIN-GREEK, LATIN1, LATIN2, LATIN3, LATIN4, LATIN5,  
LATIN6, LATIN7, LATIN8, LATIN10, LATINGREEK, LATINGREEK1, MAC-CYRILLIC,  
MAC-IS, MAC-SAMI, MAC-UK, MAC, MACCYRILLIC, MACINTOSH, MACIS, MACUK,  
MACUKRAINIAN, MS-ANSI, MS-ARAB, MS-CYRL, MS-EE, MS-GREEK, MS-HEBR,  
MS-MAC-CYRILLIC, MS-TURK, MSCP949, MSCP1361, MSMACCYRILLIC, MSZ\_7795.3,  
MS\_KANJI, NAPLPS, NATS-DANO, NATS-SEFI, NATSDANO, NATSSEFI, NC\_NC0010,  
NC\_NC00-10, NC\_NC00-10:81, NF\_Z\_62-010, NF\_Z\_62-010\_(1973), NF\_Z\_62-010\_1973,  
NF\_Z\_62010, NF\_Z\_62010\_1973, NO, NO2, NS\_4551-1, NS\_4551-2, NS\_45511,  
NS\_45512, OS2LATIN1, OSF00010001, OSF00010002, OSF00010003, OSF00010004,  
OSF00010005, OSF00010006, OSF00010007, OSF00010008, OSF00010009, OSF0001000A,  
OSF00010020, OSF00010100, OSF00010101, OSF00010102, OSF00010104, OSF00010105,  
OSF00010106, OSF00030010, OSF0004000A, OSF0005000A, OSF05010001, OSF100201A4,  
OSF100201A8, OSF100201B5, OSF100201F4, OSF100203B5, OSF1002011C, OSF1002011D,  
OSF1002035D, OSF1002035E, OSF1002035F, OSF1002036B, OSF1002037B, OSF10010001,  
OSF10020025, OSF10020111, OSF10020115, OSF10020116, OSF10020118, OSF10020122,  
OSF10020129, OSF10020352, OSF10020354, OSF10020357, OSF10020359, OSF10020360,  
OSF10020364, OSF10020365, OSF10020366, OSF10020367, OSF10020370, OSF10020387,  
OSF10020388, OSF10020396, OSF10020402, OSF10020417, PT, PT2, R8, ROMAN8, SE,  
SE2, SEN\_850200\_B, SEN\_850200\_C, SHIFT-JIS, SHIFT\_JIS, SHIFT\_JISX0213, SJIS,  
SS636127, ST\_SEV\_358-88, T.61-8BIT, T.61, T.618BIT, TCVN-5712, TCVN,  
TCVN5712-1, TCVN5712-1:1993, TIS-620, TIS620-0, TIS620.2529-1, TIS620.2533-0,  
TIS620, TS-5881, UCS-2, UCS-2BE, UCS-2LE, UCS-4, UCS-4BE, UCS-4LE, UCS2,  
UCS4, UHC, UJIS, UK, UNICODE, UNICODEBIG, UNICODELITTLE, US-ASCII, US, UTF-7,  
UTF-8, UTF-16, UTF-16BE, UTF-16LE, UTF-32, UTF-32BE, UTF-32LE, UTF7, UTF8,  
UTF16, UTF16BE, UTF16LE, UTF32, UTF32BE, UTF32LE, VISCI, WCHAR\_T,  
WIN-SAMI-2, WINBALTRIM, WINDOWS-1250, WINDOWS-1251, WINDOWS-1252,  
WINDOWS-1253, WINDOWS-1254, WINDOWS-1255, WINDOWS-1256, WINDOWS-1257,



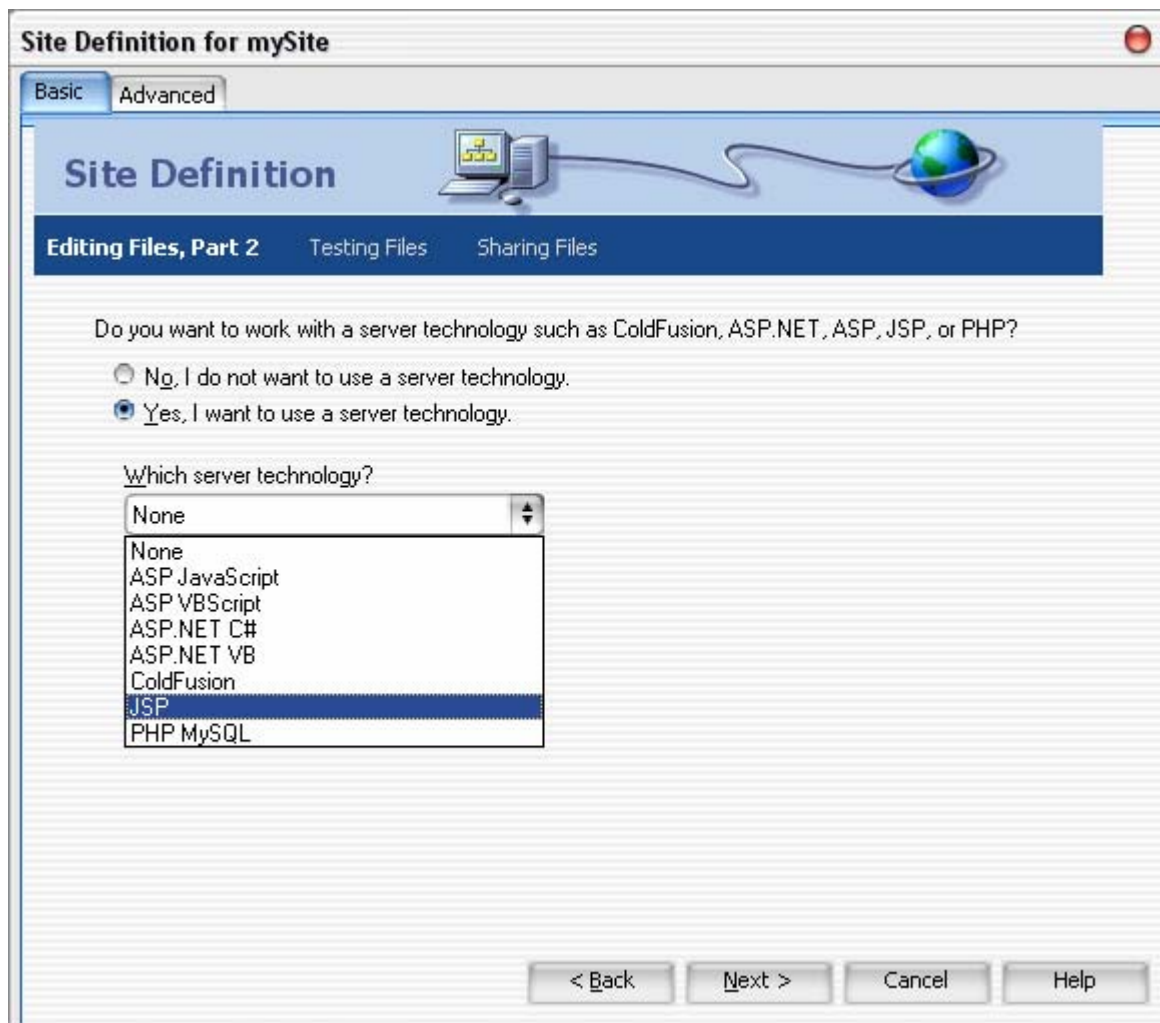


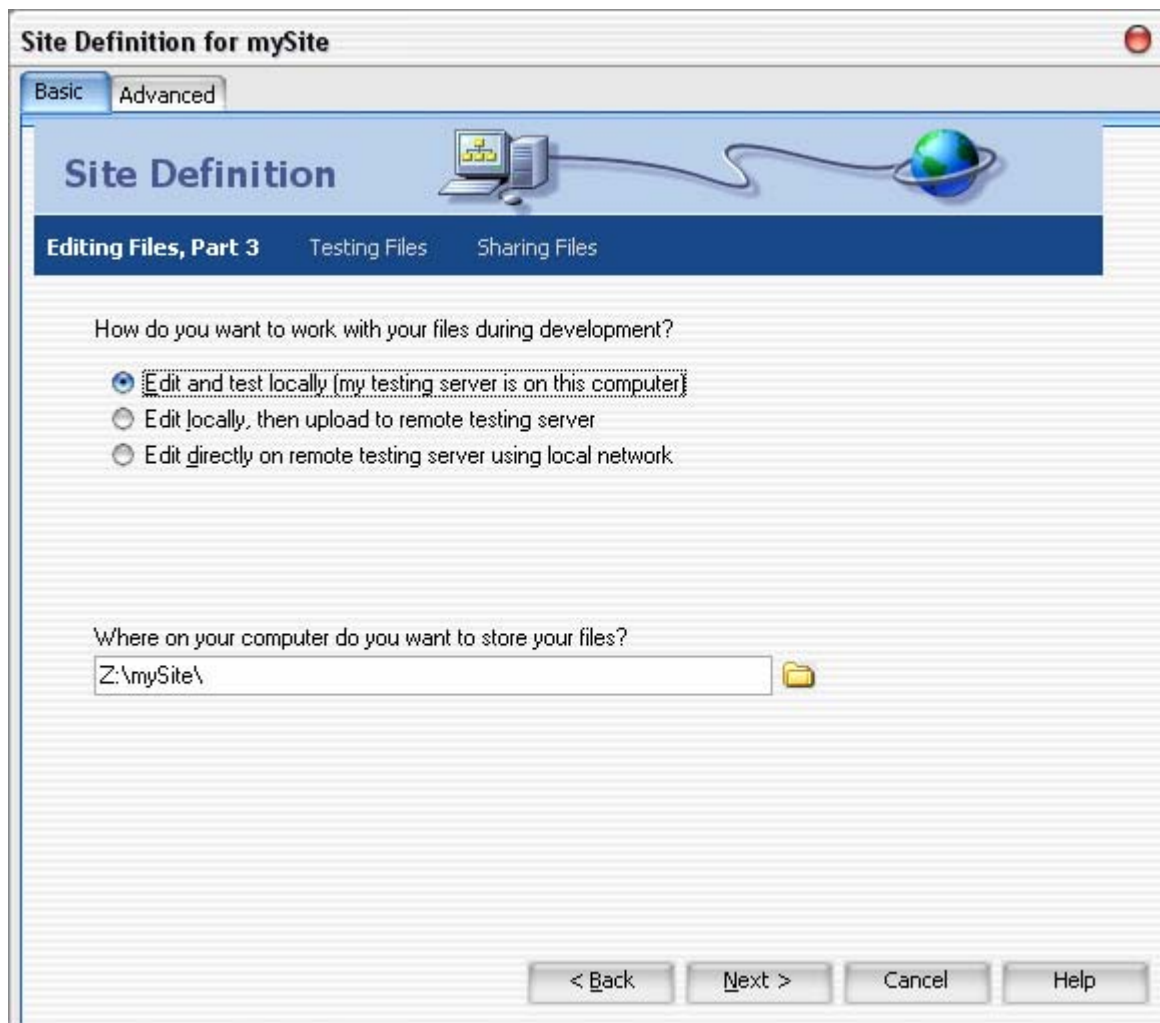
2          New...          Site



3                          mySite







5

```
z: net use z: \\192.168.0.1\web \\192.168.0.1\web linux web Samba
/var/www/html/webapps /var/www/html/webapps netkiller.xml <Context path="/netkiller"
docBase="/var/www/html/webapps" reloadable="true" debug="0" privileged="true">
```

```
[root@linux root]# cat /etc/samba/smb.conf
[web]
comment = PC Directories
path = /var/www/html/webapps
public = yes
writable = yes

[root@linux root]# cat /usr/local/jakarta-tomcat/webapps/netkiller.xml
<!--

Context configuration file for the Tomcat Administration Web App

$Id: admin.xml,v 1.3 2002/07/23 12:12:15 remm Exp $
```

-->

```
<Context path="/netkiller" docBase="/var/www/html/webapps" reloadable="true"
    debug="0" privileged="true">
```

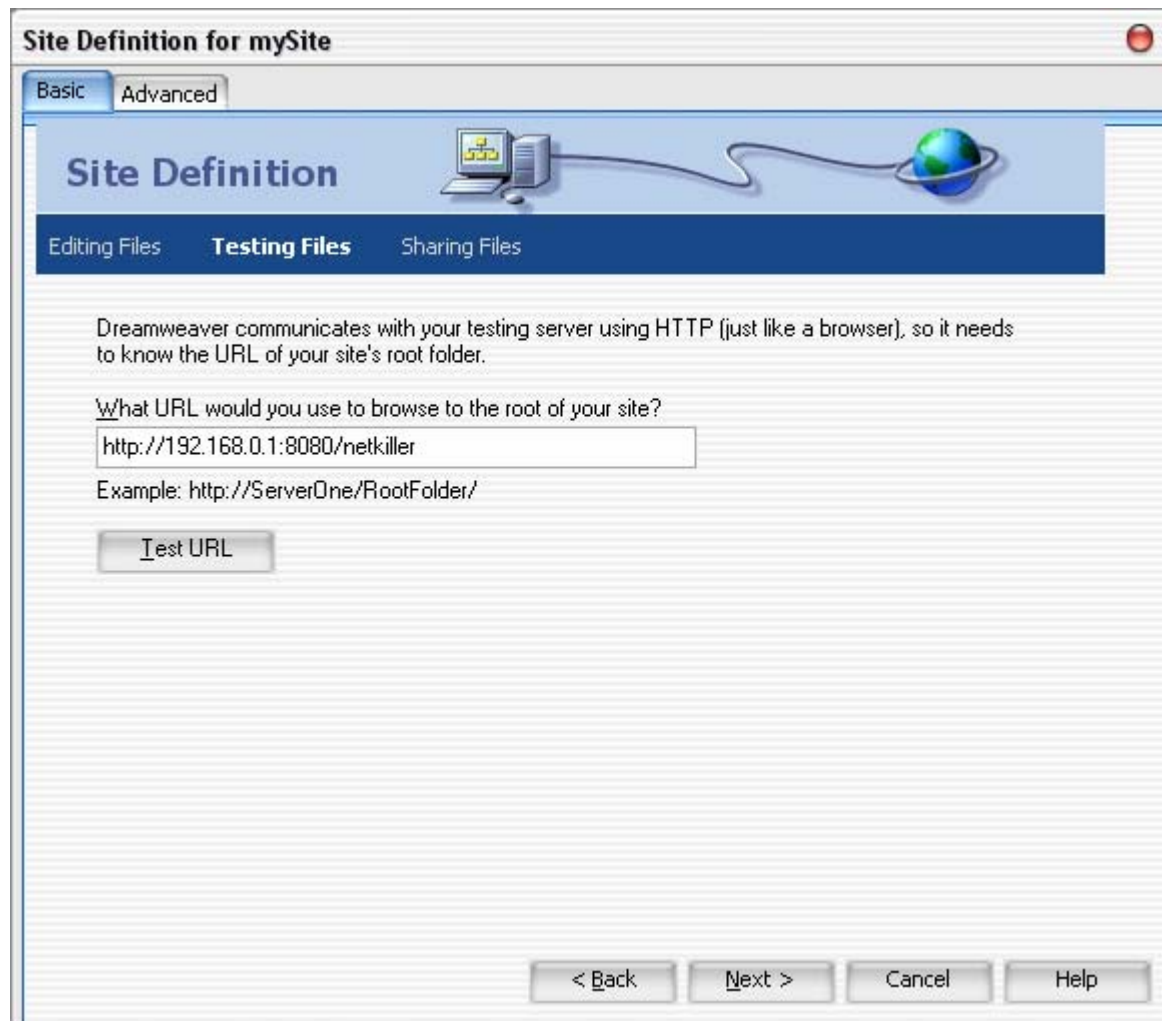
```
<!-- Uncomment this Valve to limit access to the Admin app to localhost
    for obvious security reasons. Allow may be a comma-separated list of
    hosts (or even regular expressions).
```

```
<Valve className="org.apache.catalina.valves.RemoteAddrValve"
    allow="127.0.0.1"/>
```

-->

```
<Logger className="org.apache.catalina.logger.FileLogger"
    prefix="localhost_netkiller_log." suffix=".txt"
    timestamp="true"/>
```

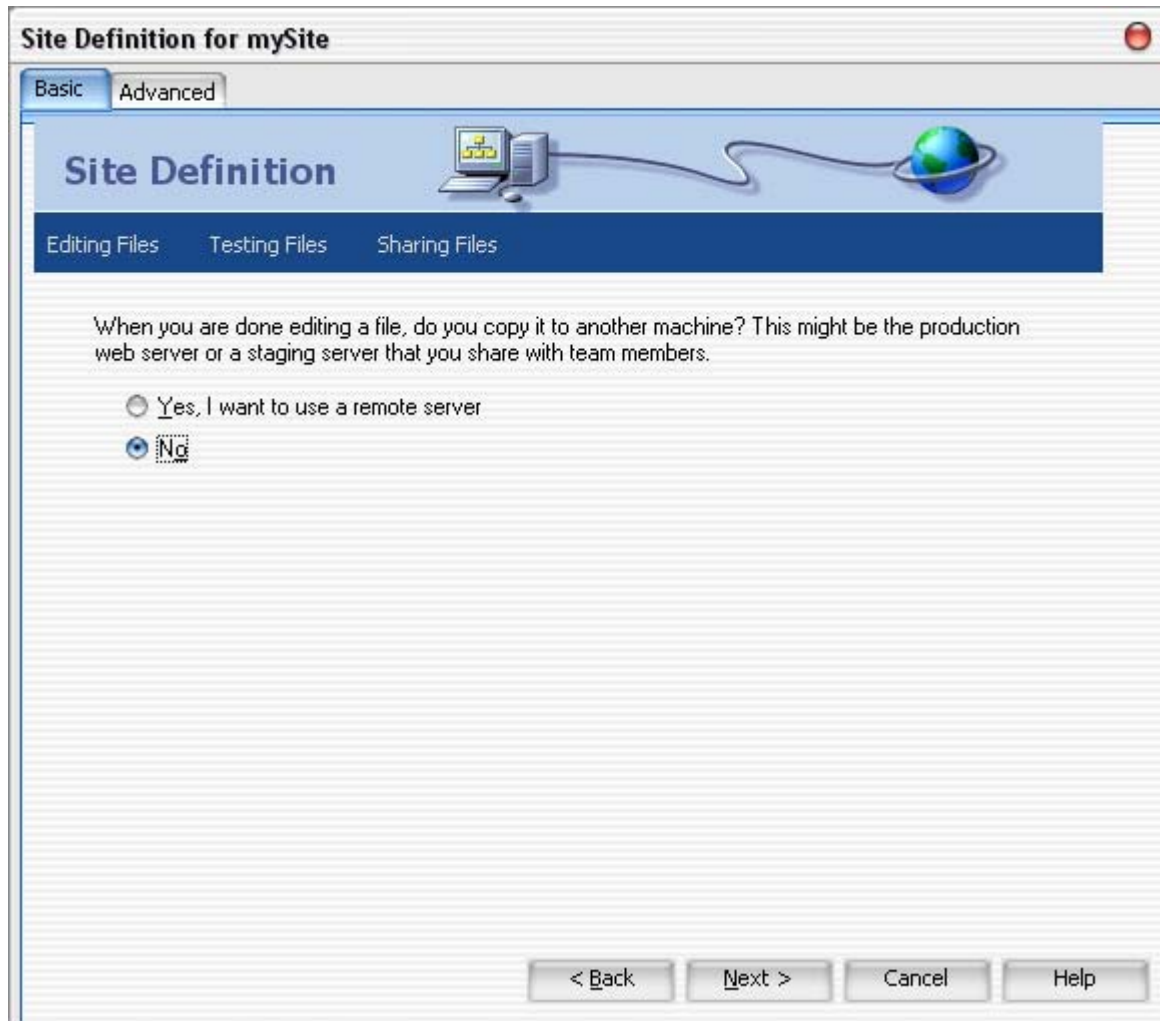
```
</Context>
```



6 URL : <http://192.168.0.1:8080/netkiller> " Test User"

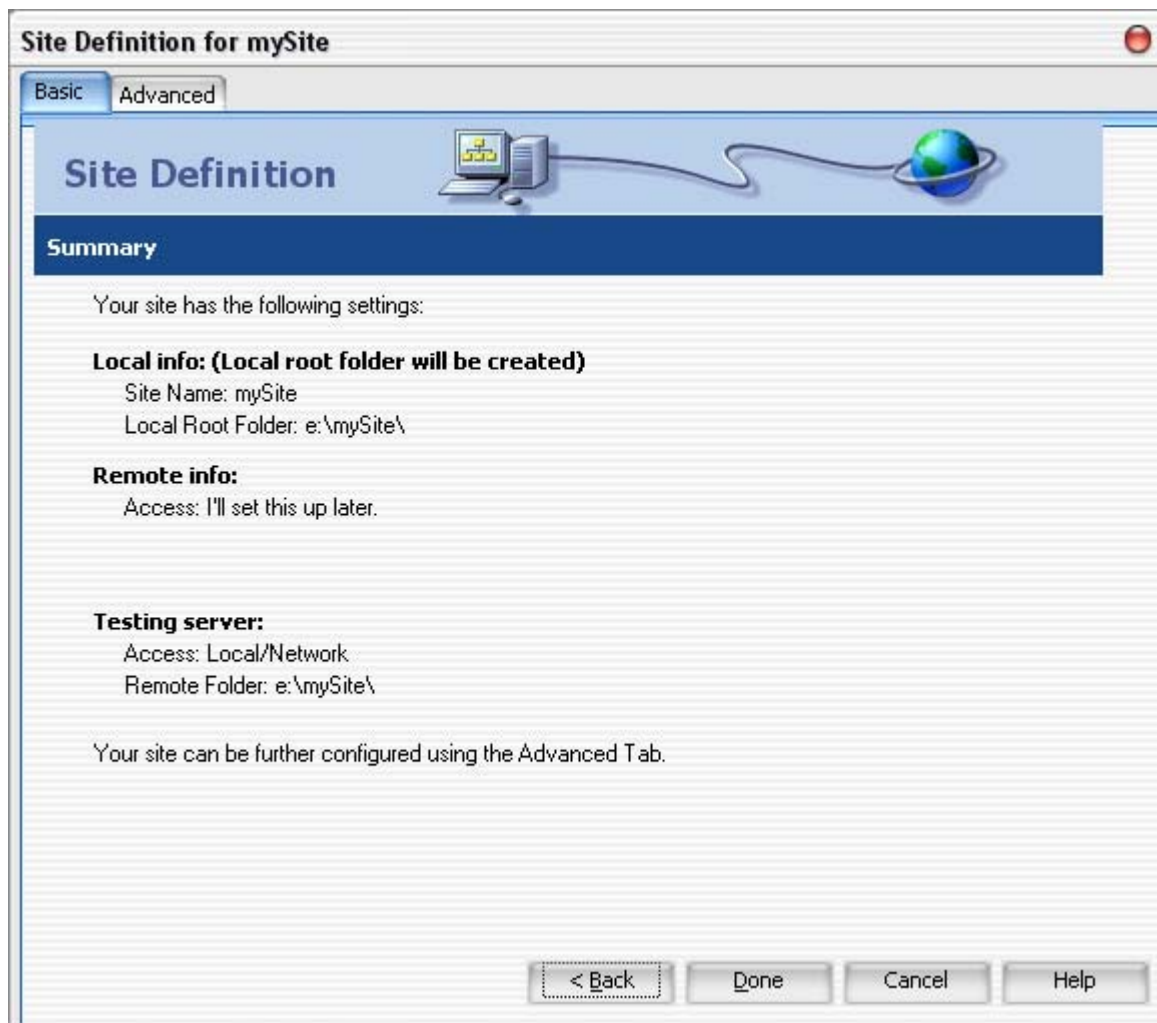


7

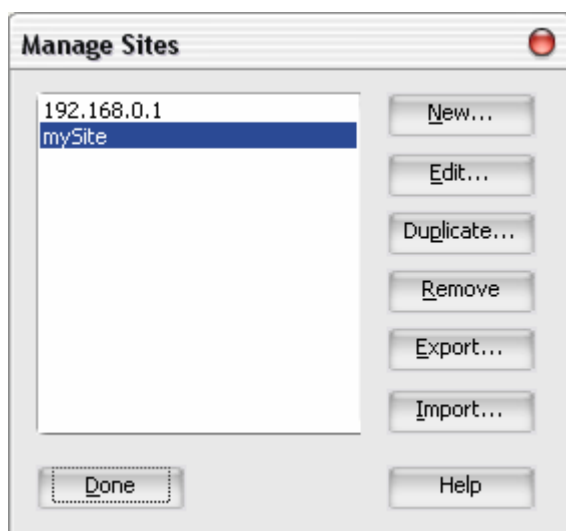


8

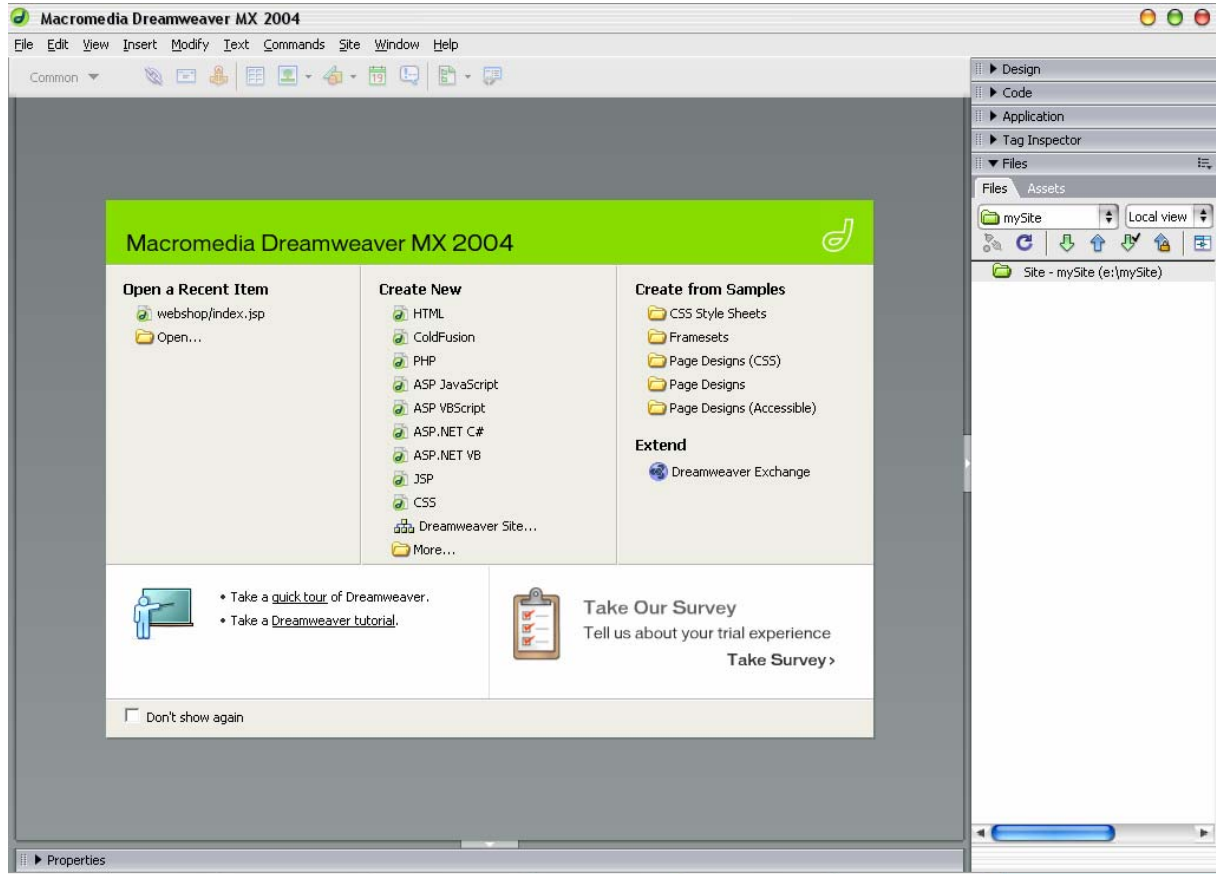
No



9 " Done"



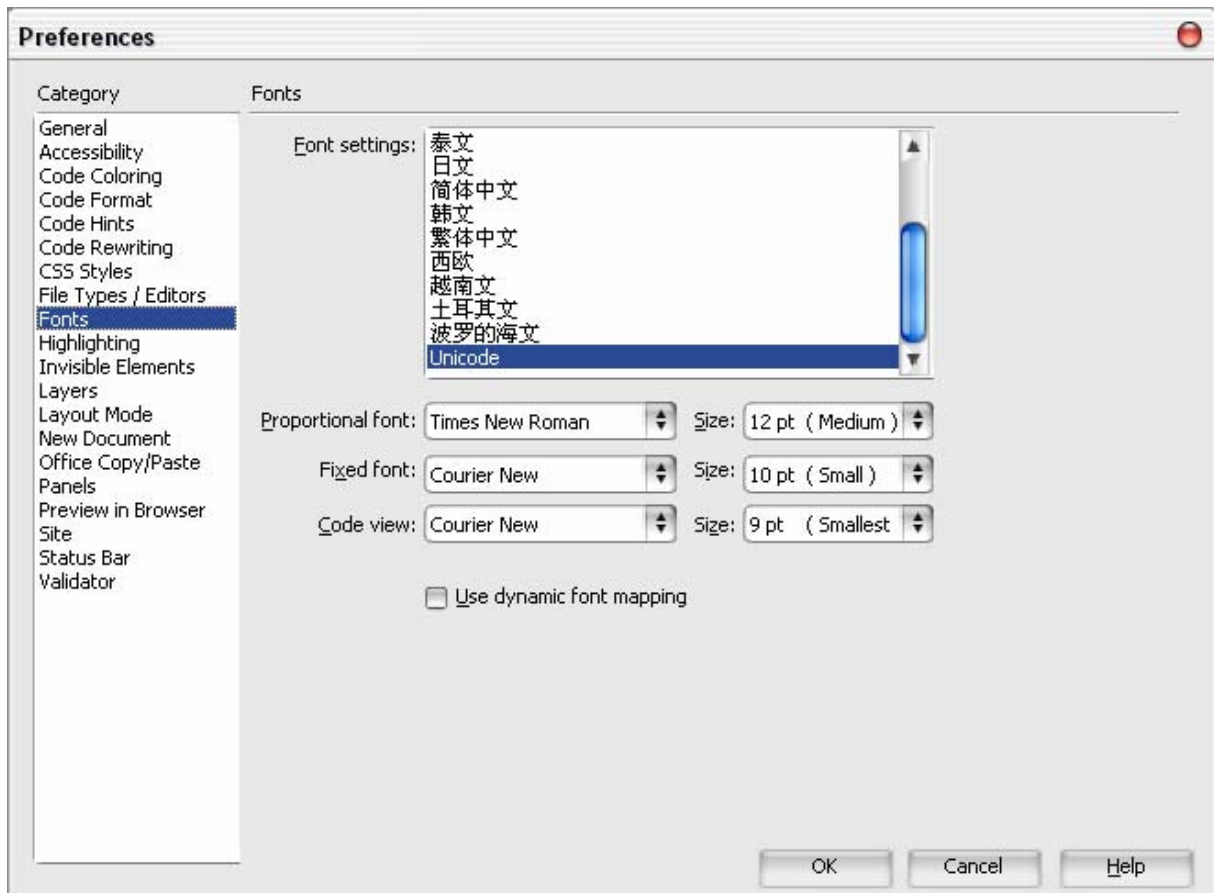
10 " Done"



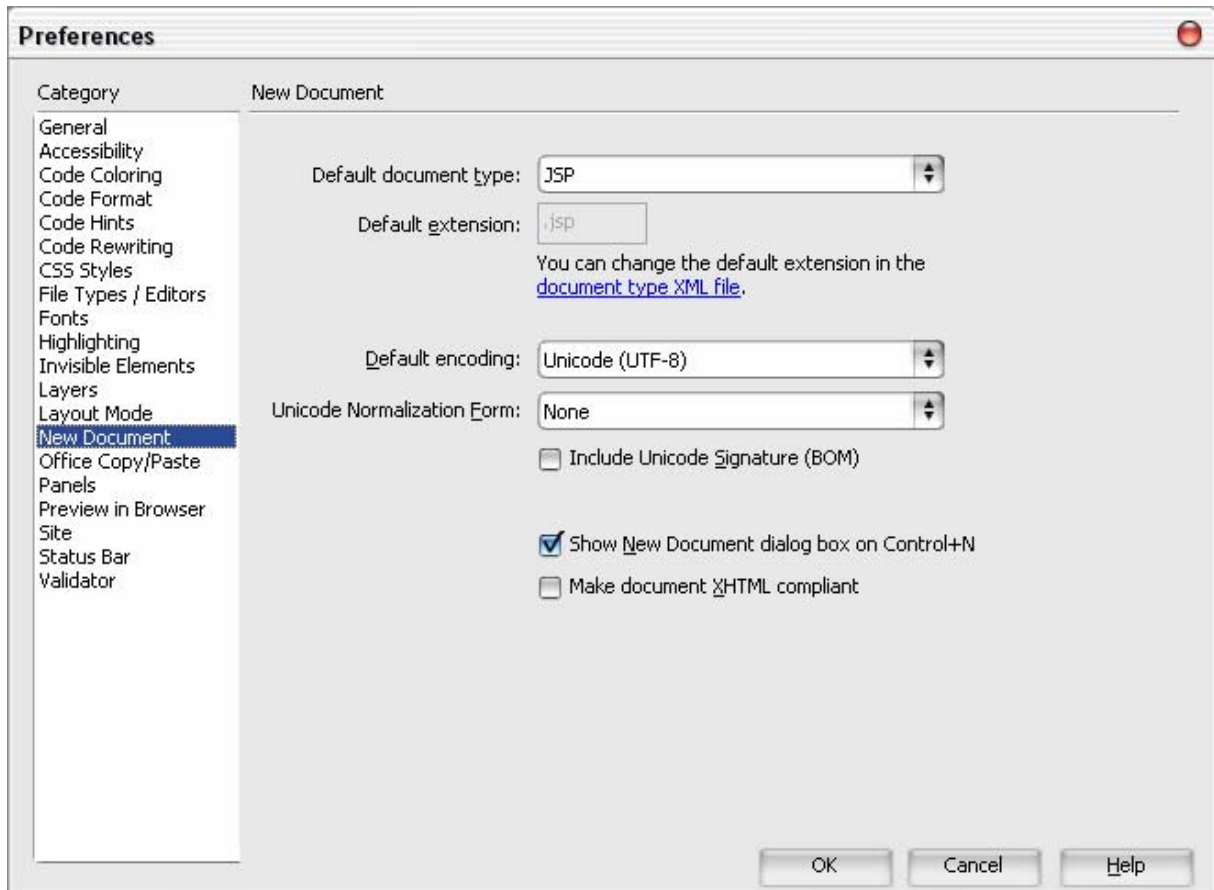
Dreamweaver

11 Ctrl+u Dreamweaver

11.1 Fonts



## 11.2 New Document



## 15.2 Jcreator

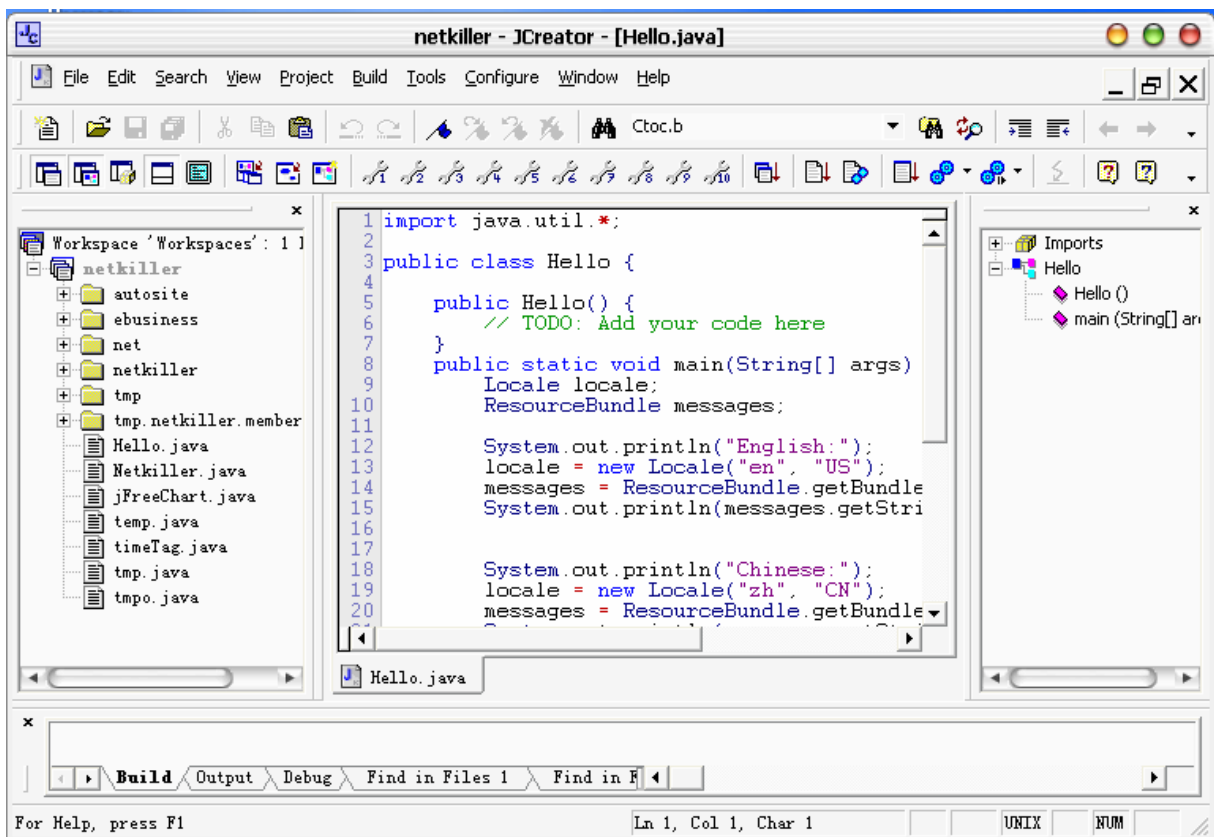
JCreator is written entirely in C++, which makes it fast and efficient compared to the Java based editors/IDEs.

JCreator

Bean

Dreamweaver

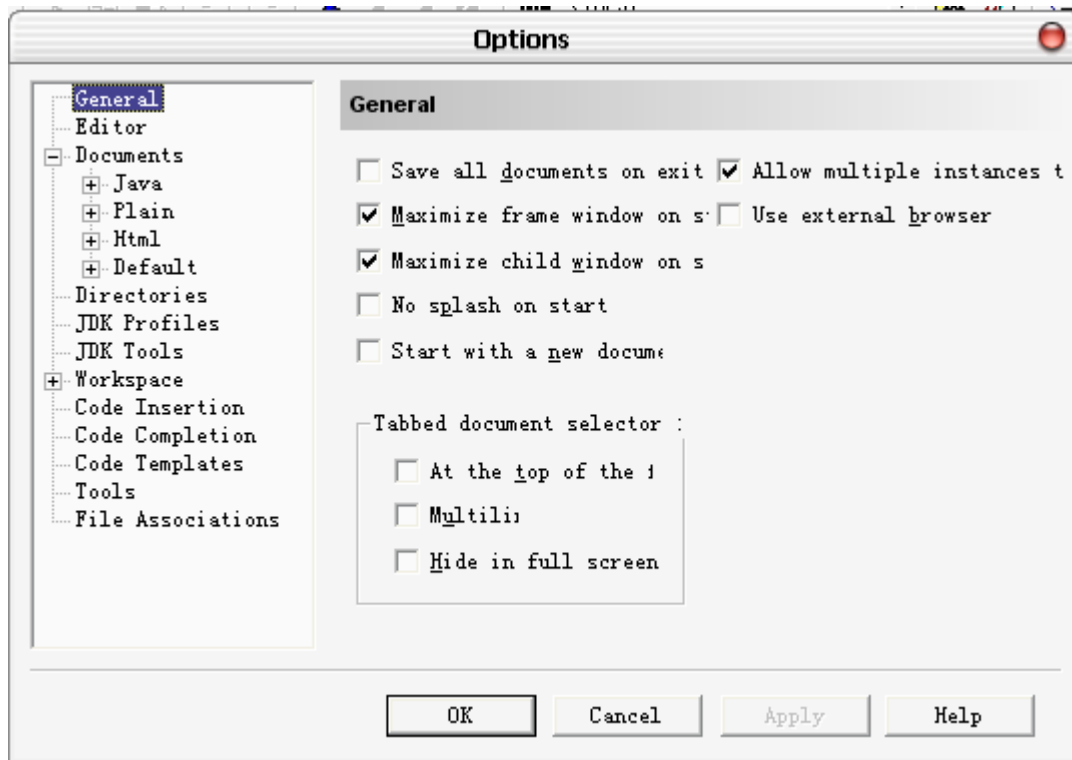
JSP



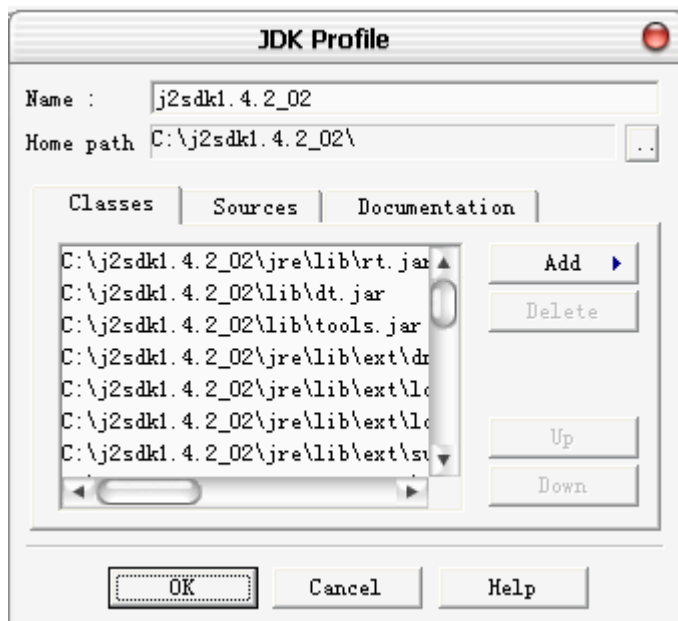
" Tools"

" Options..."



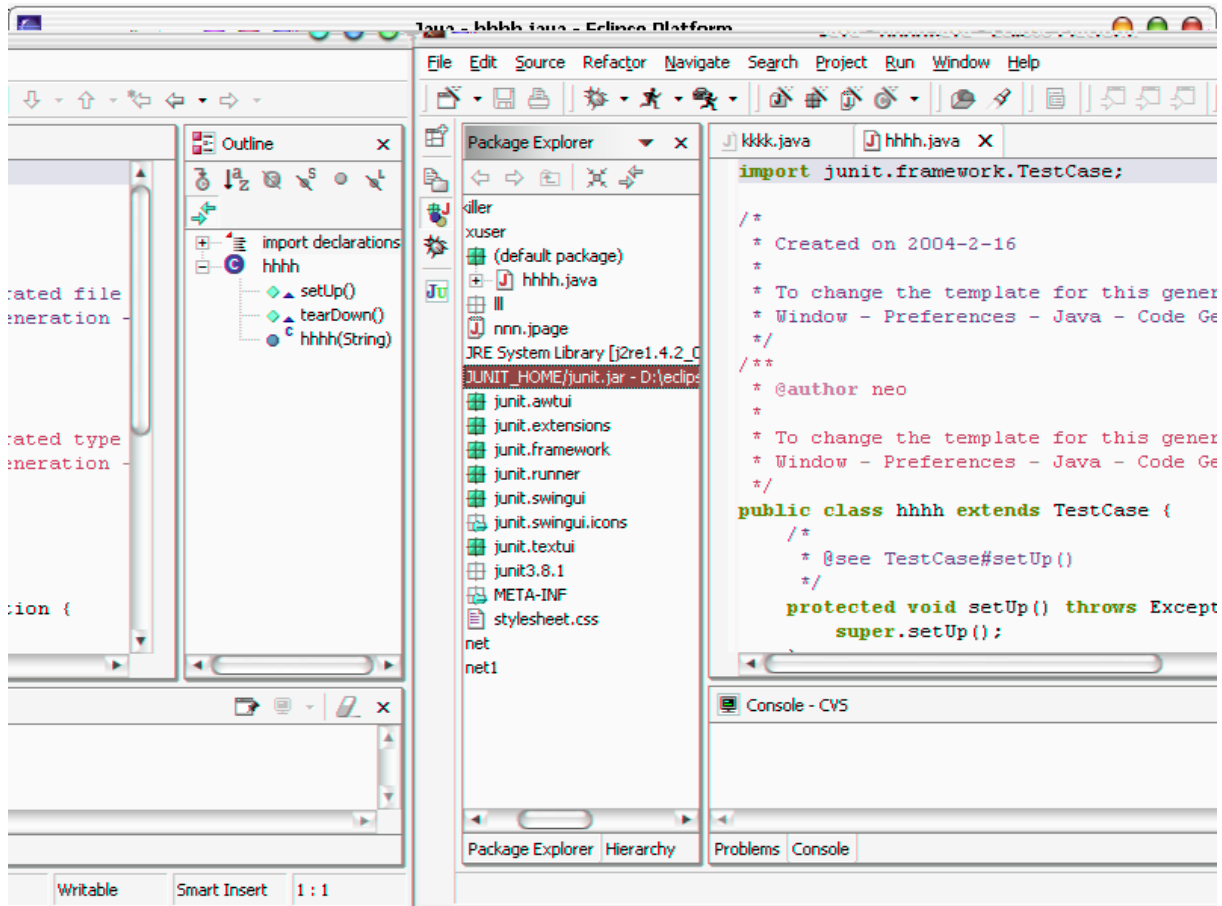


" JDK Profiles"



## 15.3 Eclipse

Eclipse



## 15.4 JBuilder + Weblogic + PostgreSQL 开发环境

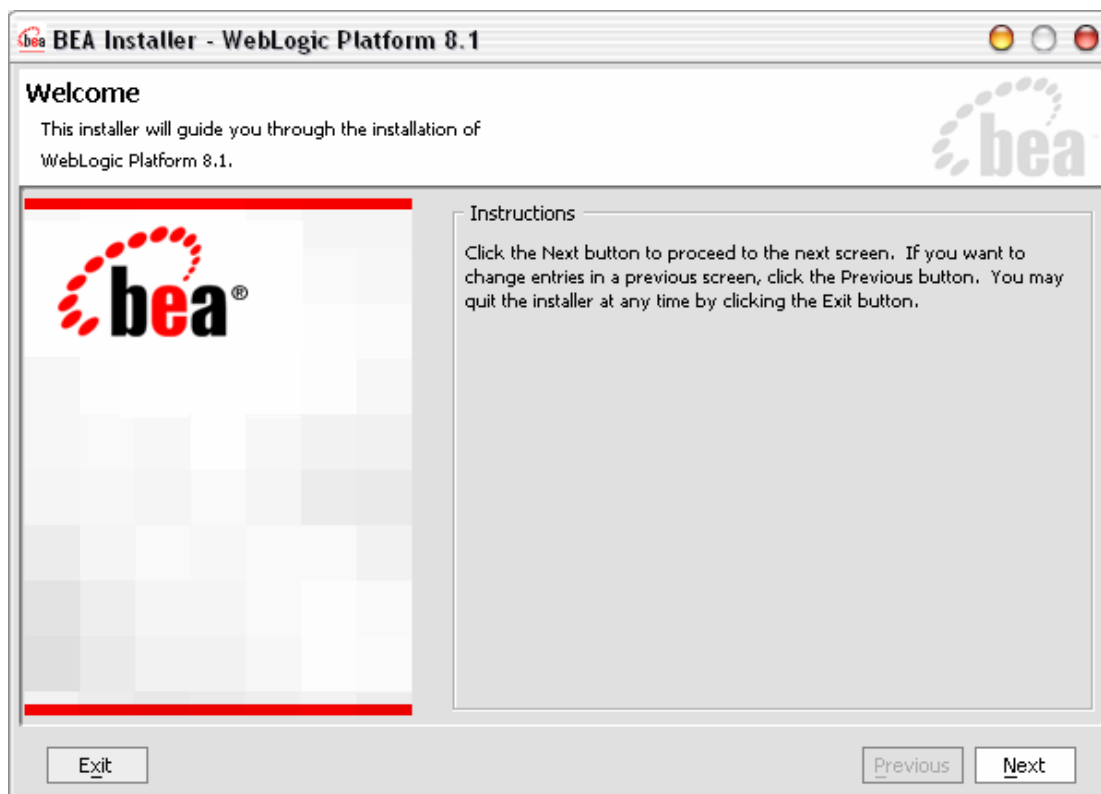
JBuilder9+Weblogic8+PostgreSQL7 How to

BEA Weblogic platform811

1. platform811\_win32.exe



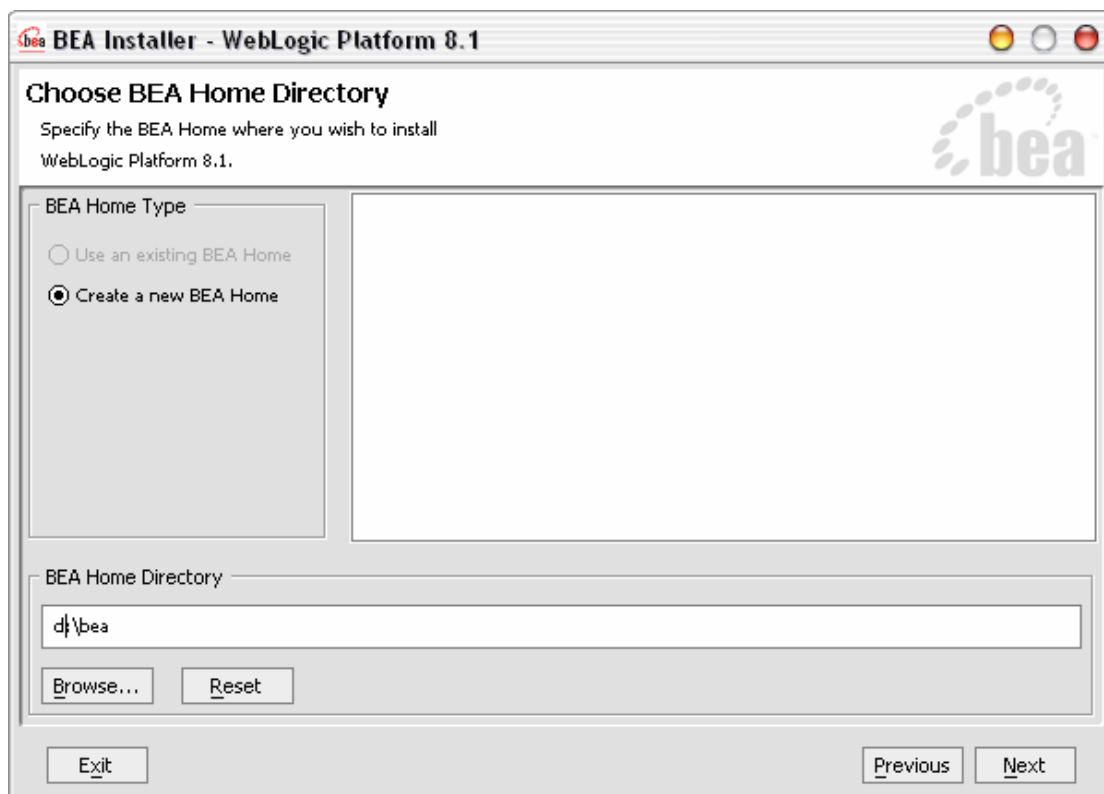
2.



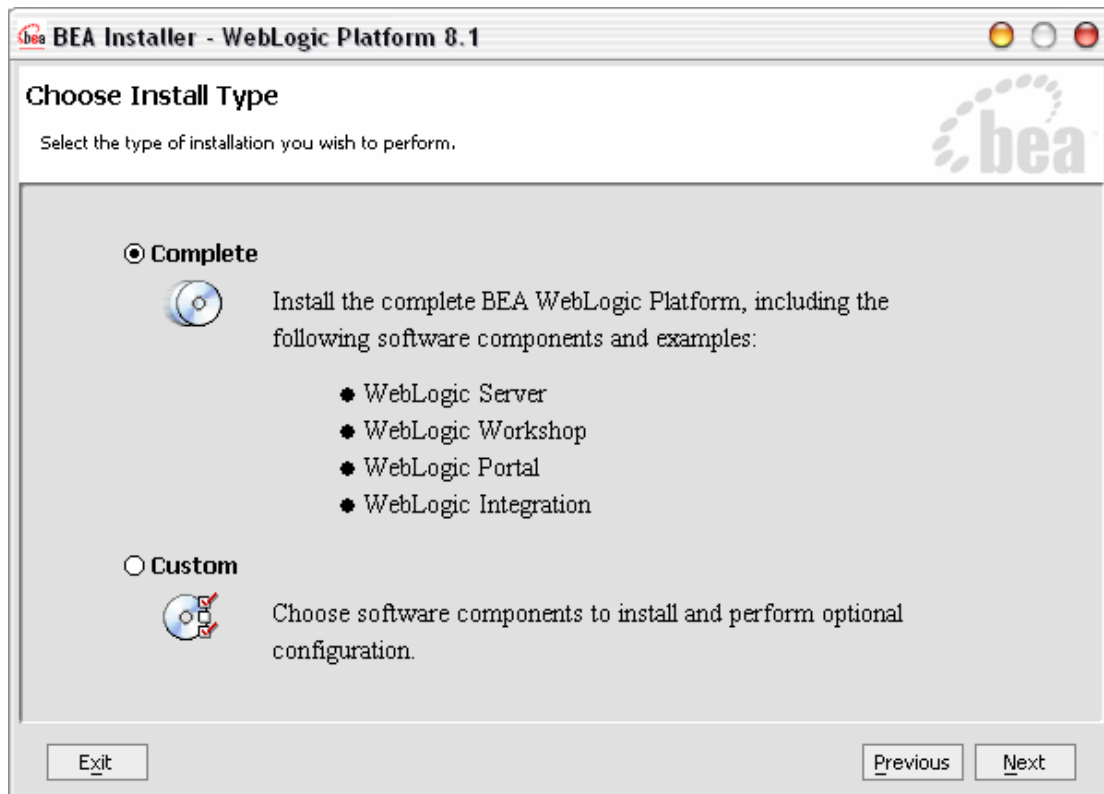
3. Next



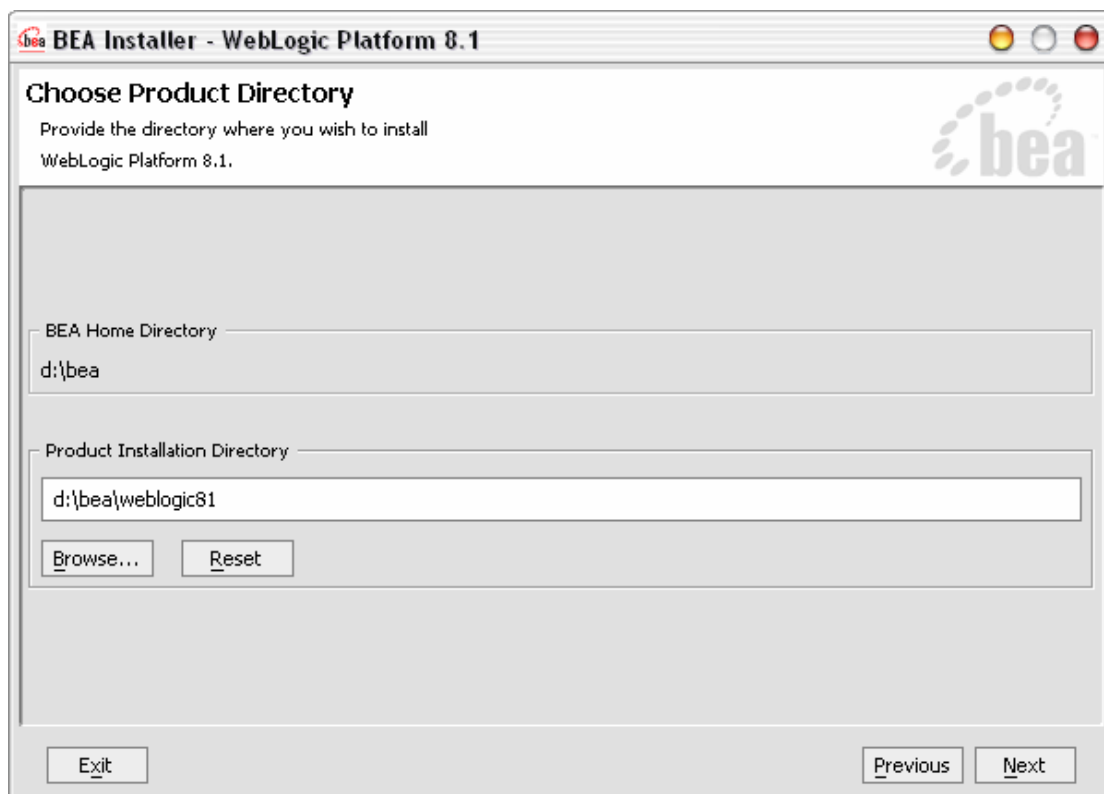
4. Next



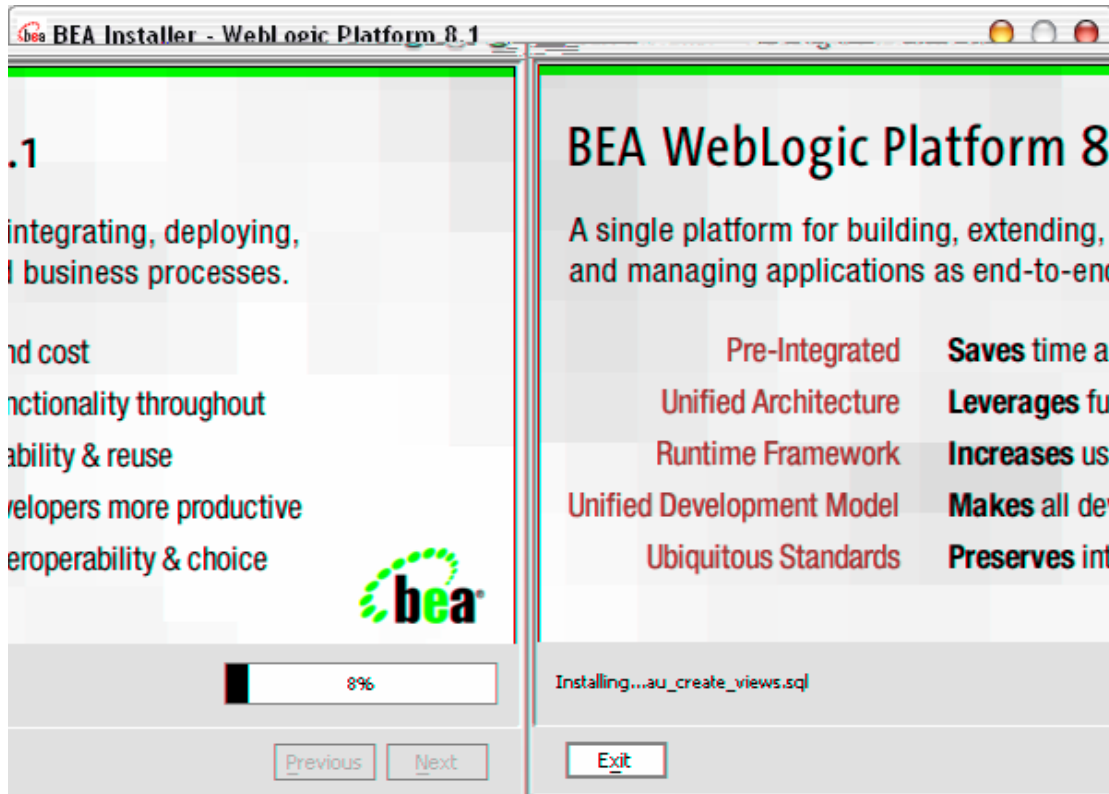
5. BEA



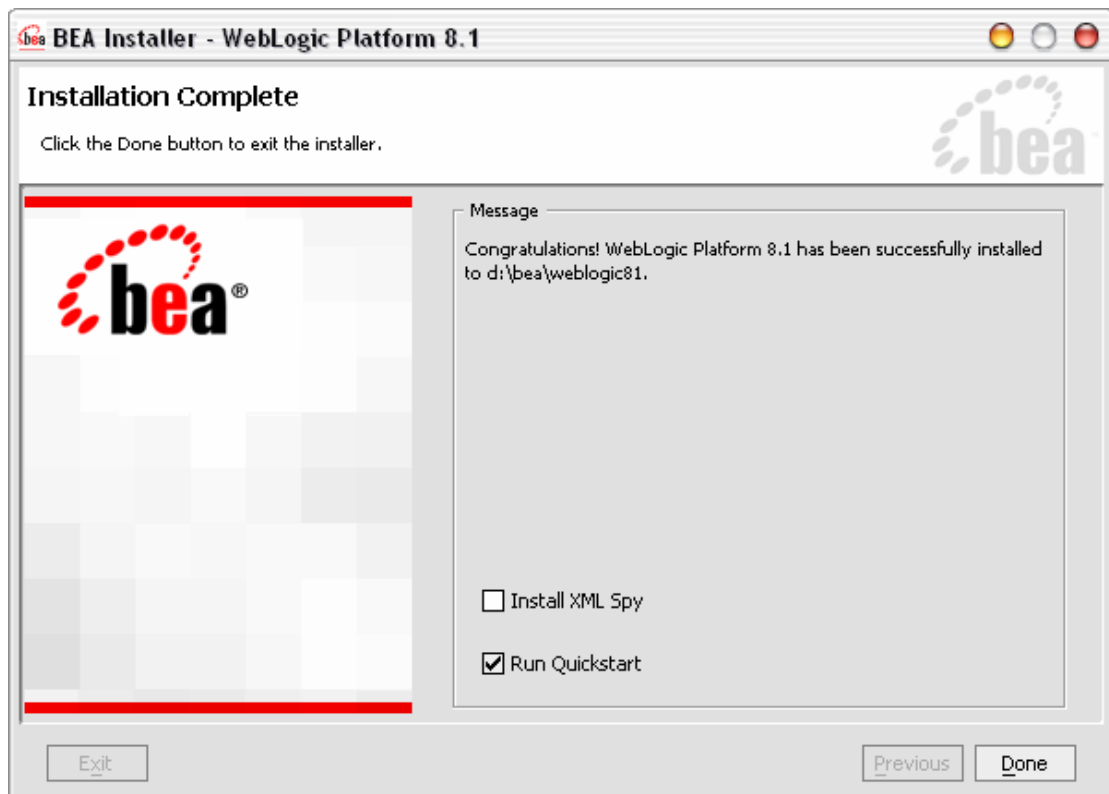
6. Next



7. bea weblogic8.1 Next

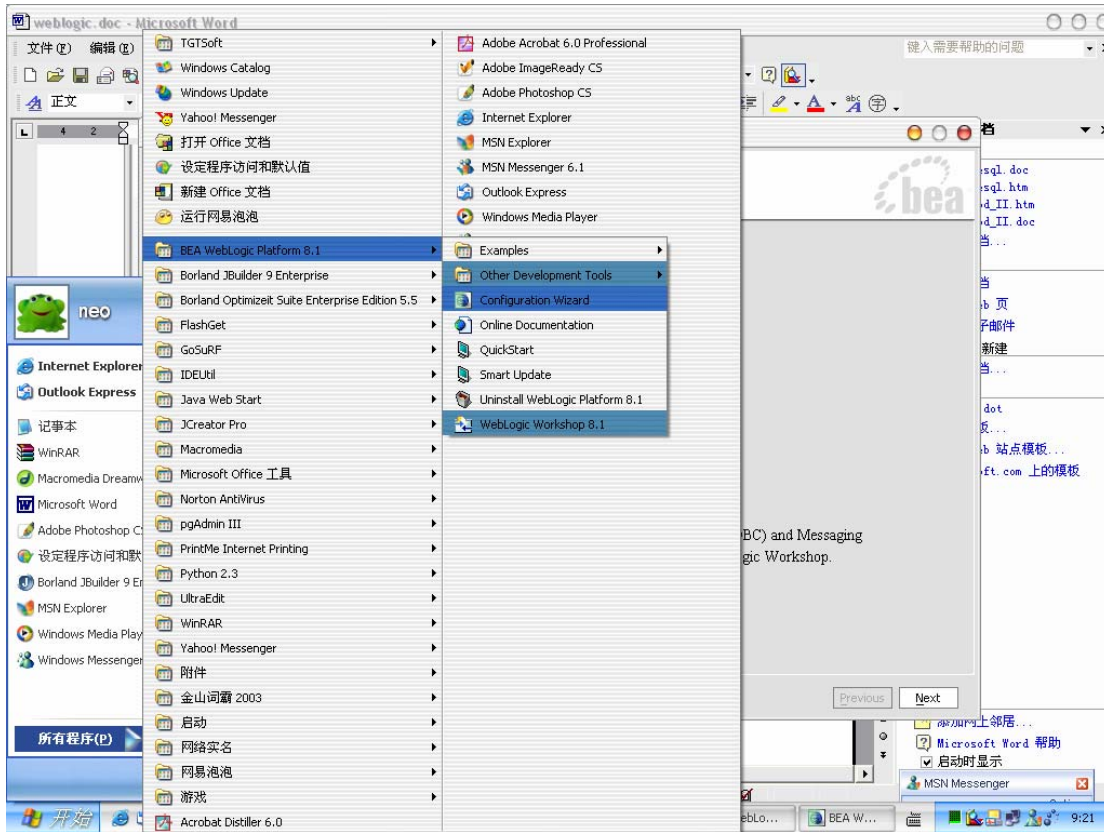


8.

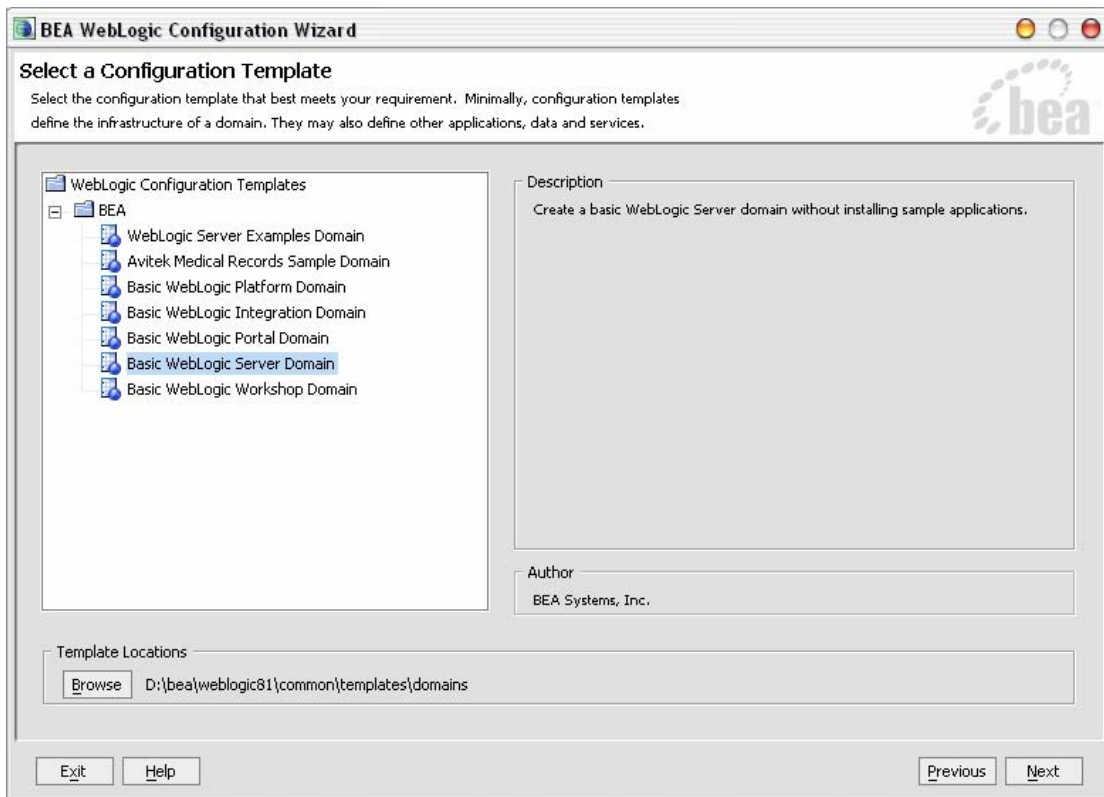


9. Done

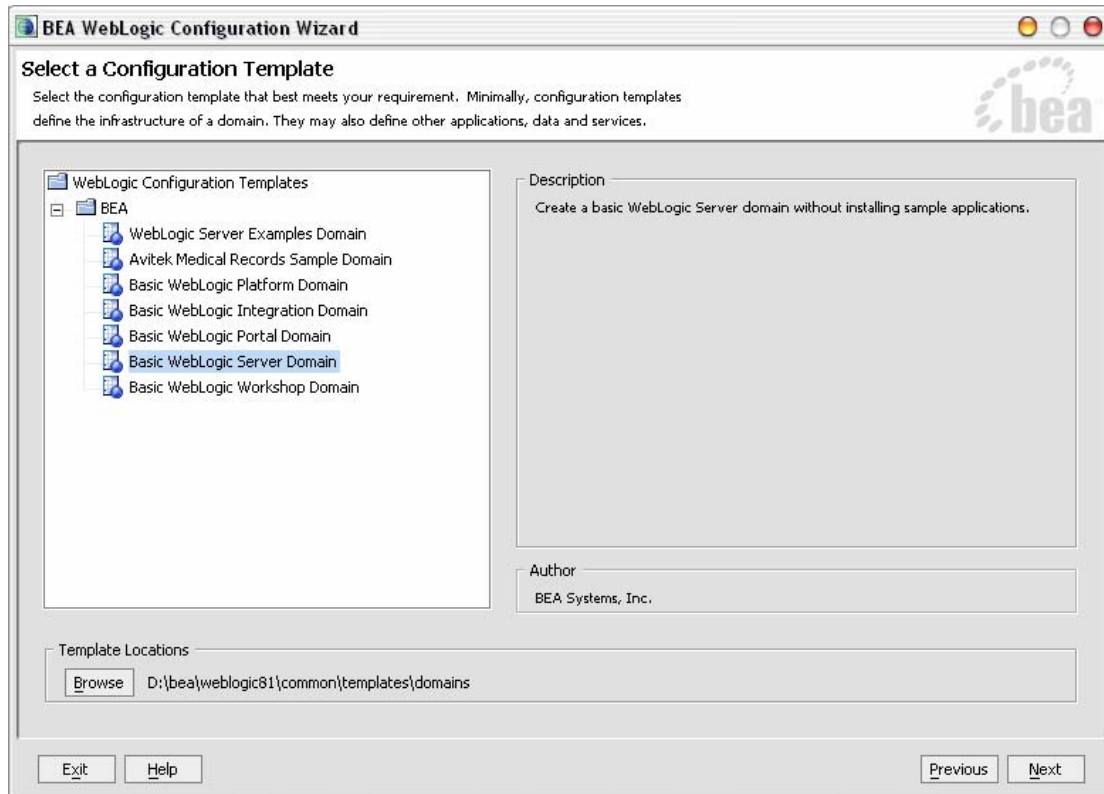
# 1. bea weblogic Configuration Wizard



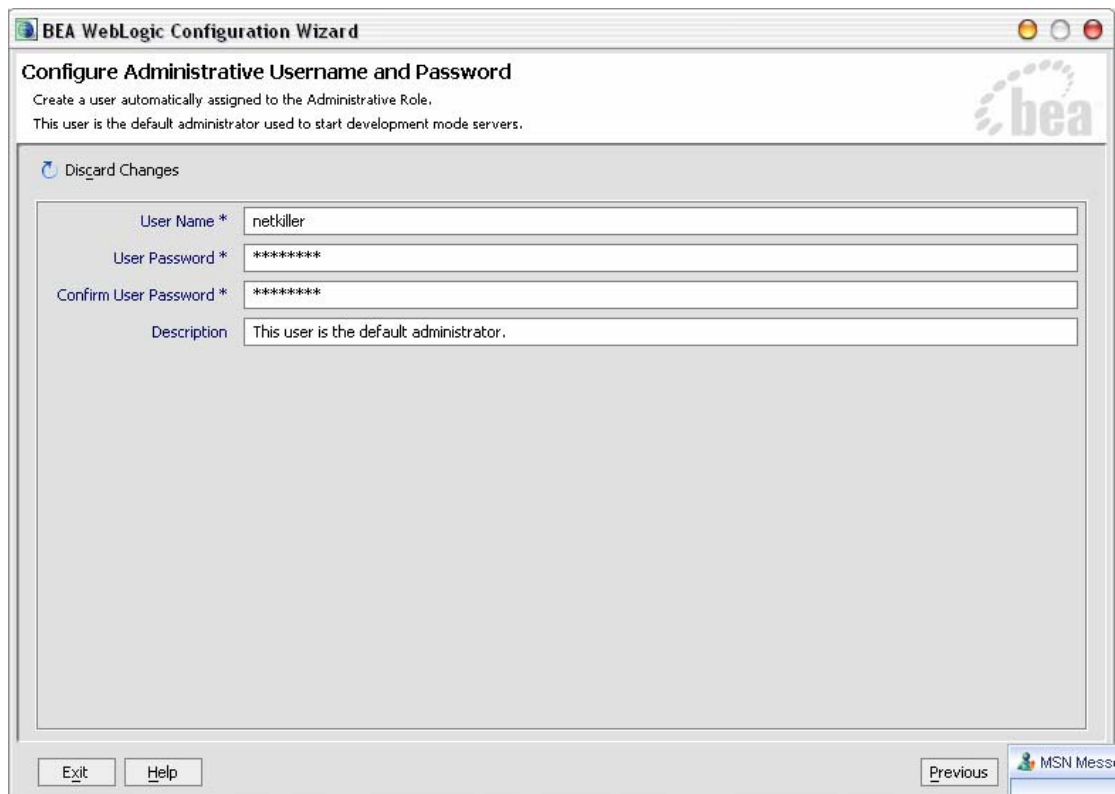
# 1. Configuration Wizard



# 2. Next

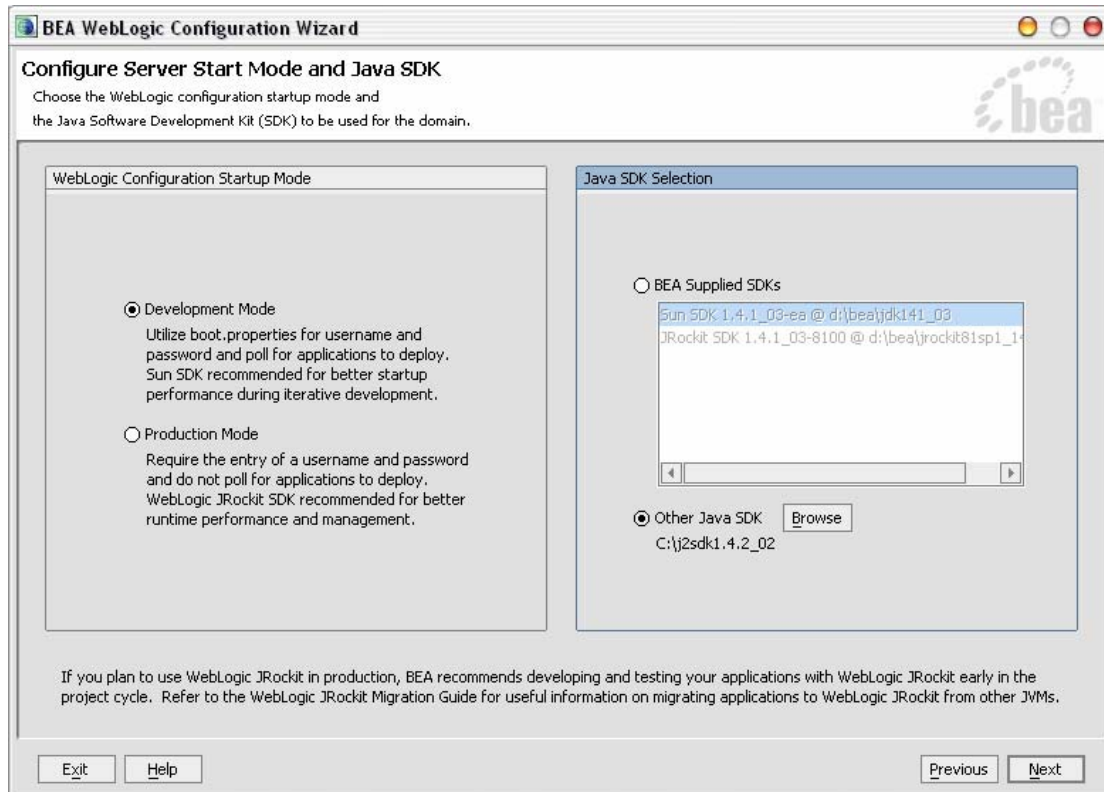


3. Next

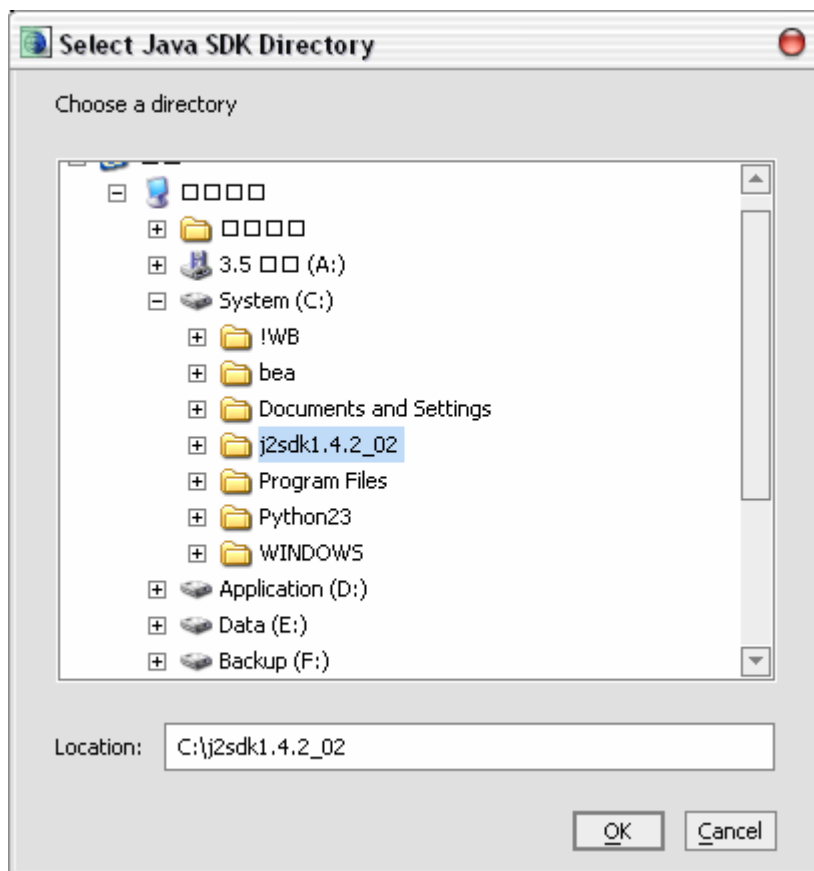


4.

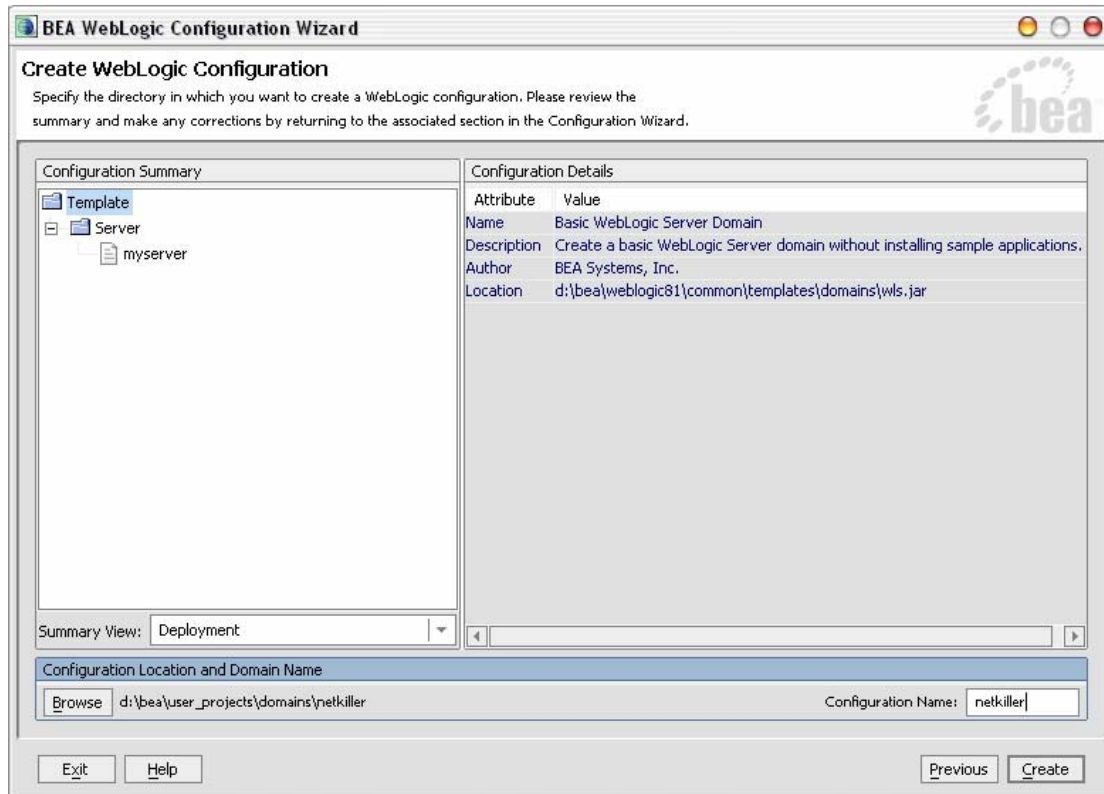




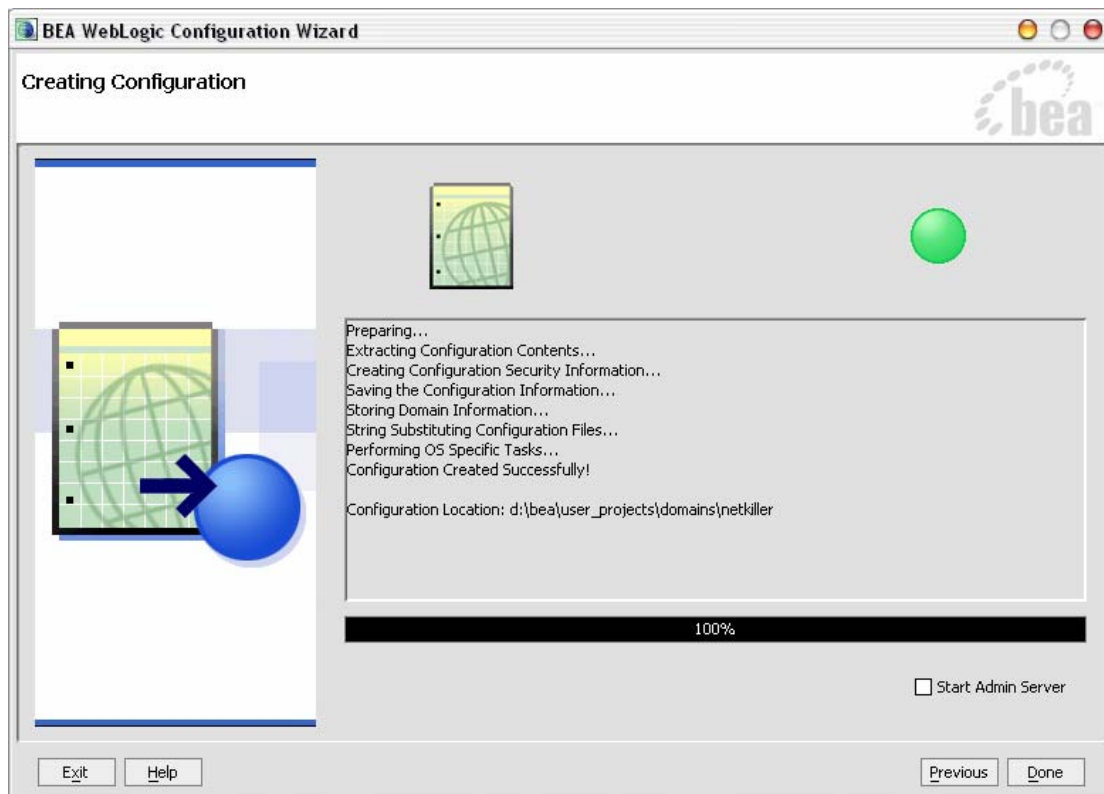
5.   bae weblogic                      JDK1.4.1                      JDK 1.4.2  
       Other Java SDK



6.       Next

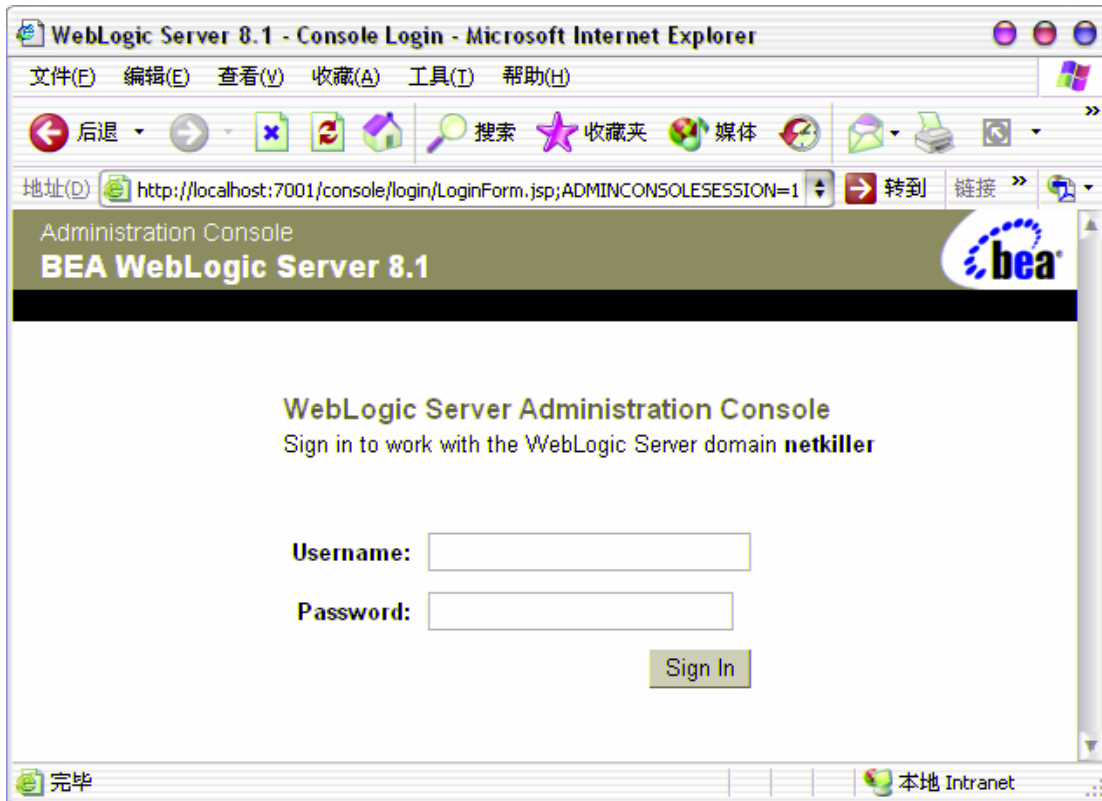


7. Configuration Name : netkiller

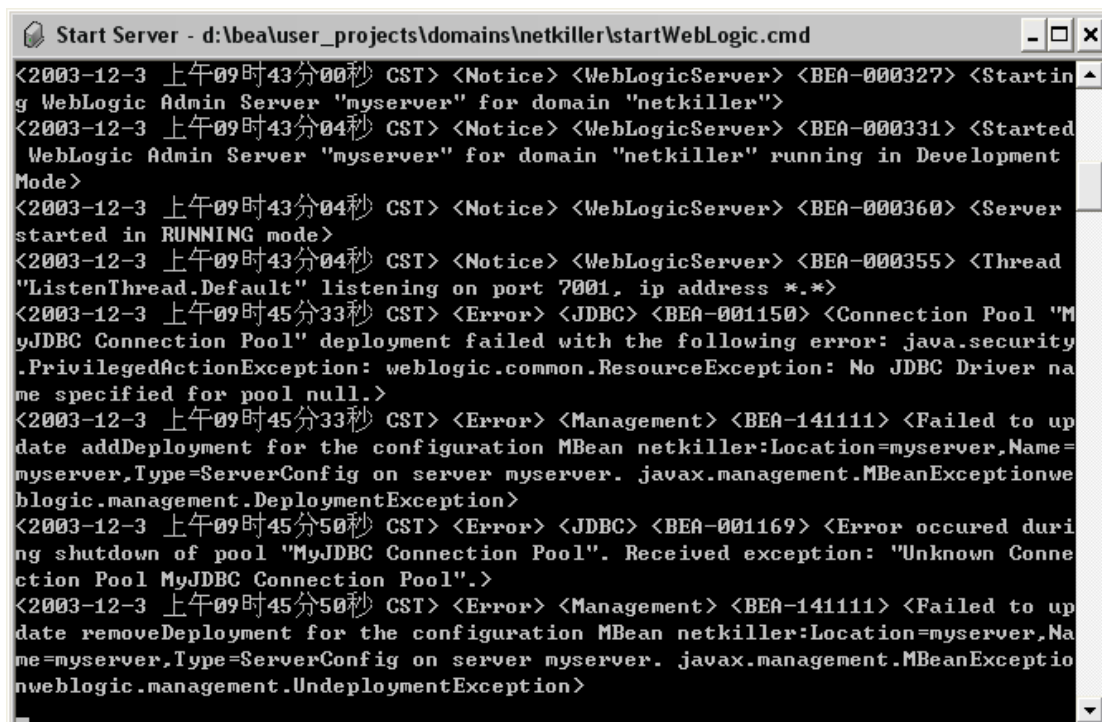


8. Start Admin Server Done

9. IE http://localhost:7001/console



10. weblogic

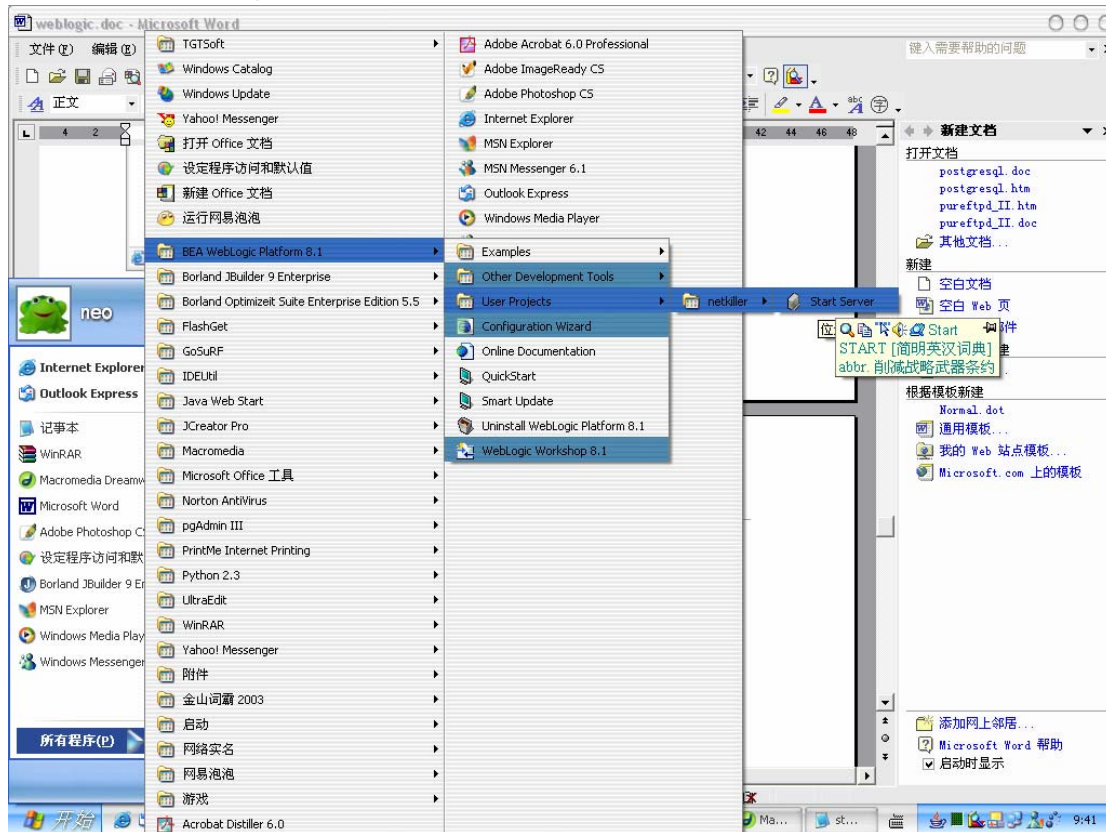


11. PostgreSQL JDBC

PostgreSQL JDBC pg73jdbc3.jar D:\bea\weblogic81\server\lib  
D:\bea\user\_projects\domains\netkiller\startWebLogic.cmd  
set CLASSPATH

set CLASSPATH=%CLASSPATH%;%WL\_HOME%\server\lib\pg73jdbc3.jar

## 12. weblogic netkiller



d:\bea\WEBLOG~1\server\lib\pg73jdbc3.jar

```
d:\bea\user_projects\domains\netkiller>ECHO OFF
```

```
.  
CLASSPATH=C:\J2SDK1~1.2_0\lib\tools.jar;d:\bea\WEBLOG~1\server\lib\weblogic_sp.jar;d:\bea\WEBLOG~1\server\lib\weblogic.jar;d:\bea\WEBLOG~1\server\lib\ojdbc14.jar;d:\bea\WEBLOG~1\c\common\eval\pointbase\lib\pbserver44.jar;d:\bea\WEBLOG~1\c\common\eval\pointbase\lib\pbclient44.jar;C:\J2SDK1~1.2_0\jre\lib\rt.jar;d:\bea\WEBLOG~1\server\lib\webservices.jar;d:\bea\WEBLOG~1\server\lib\pg73jdbc3.jar
```

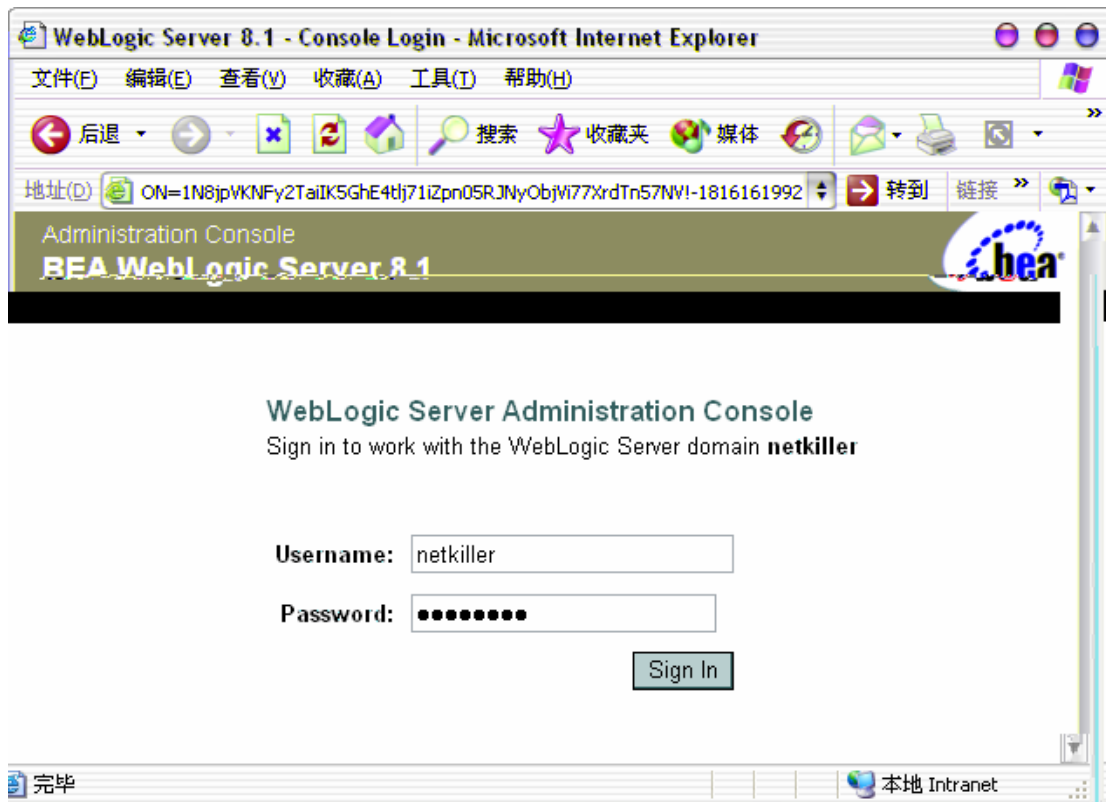
```
.  
PATH=d:\bea\WEBLOG~1\server\bin;C:\J2SDK1~1.2_0\jre\bin;C:\J2SDK1~1.2_0\bin;C:\WINDOWS\system32;C:\WINDOWS;C:\WINDOWS\System32\Wbem;d:\bea\WEBLOG~1\server\bin\oci920_8
```

```
*****
```

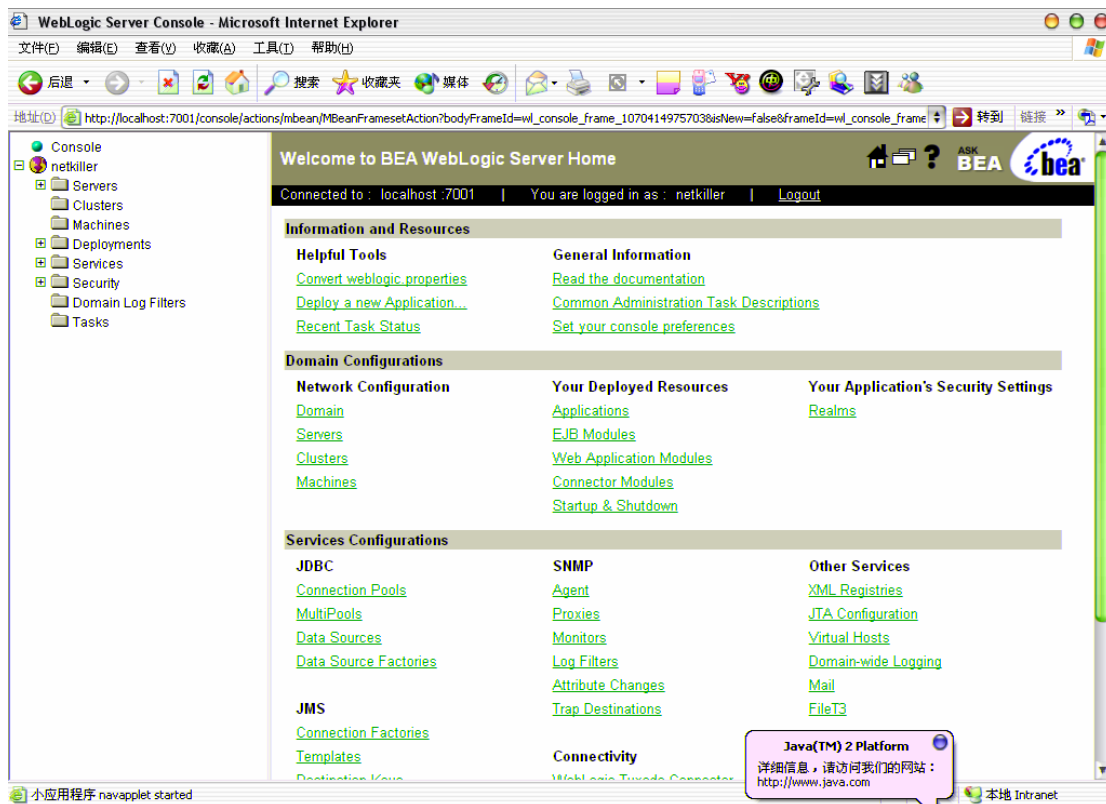
```
* To start WebLogic Server, use a username and *  
* password assigned to an admin-level user. For *  
* server administration, use the WebLogic Server *  
* console at http://[hostname]:[port]\console *
```

```
*****
```

## 13.



14.

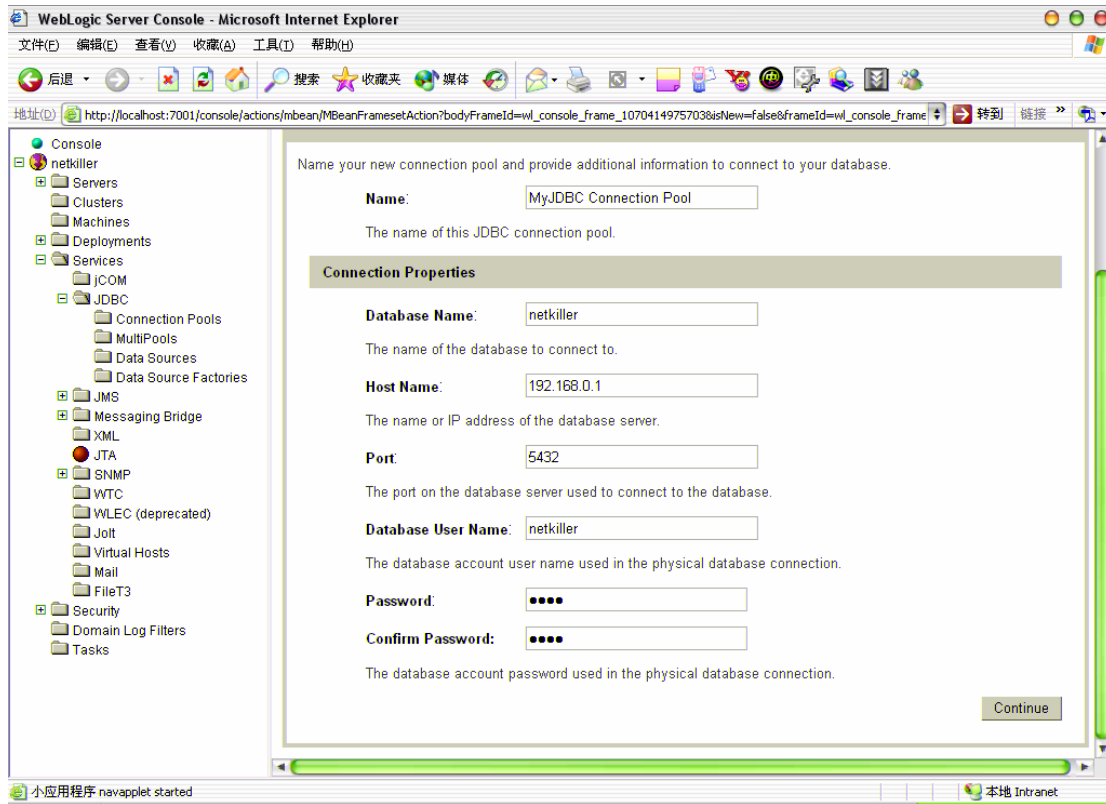


15. netkiller-->Service-->JDBC-->Connection Pools

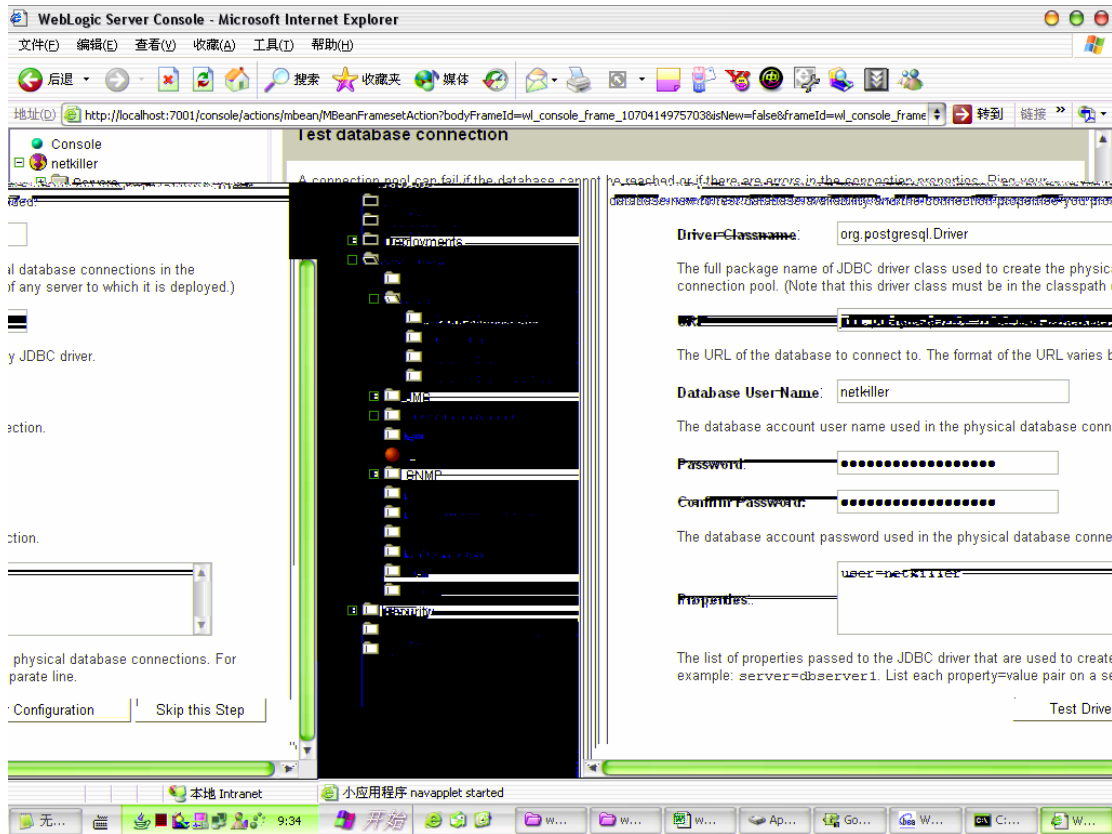
16. Configure a new JDBC Connection Pool...



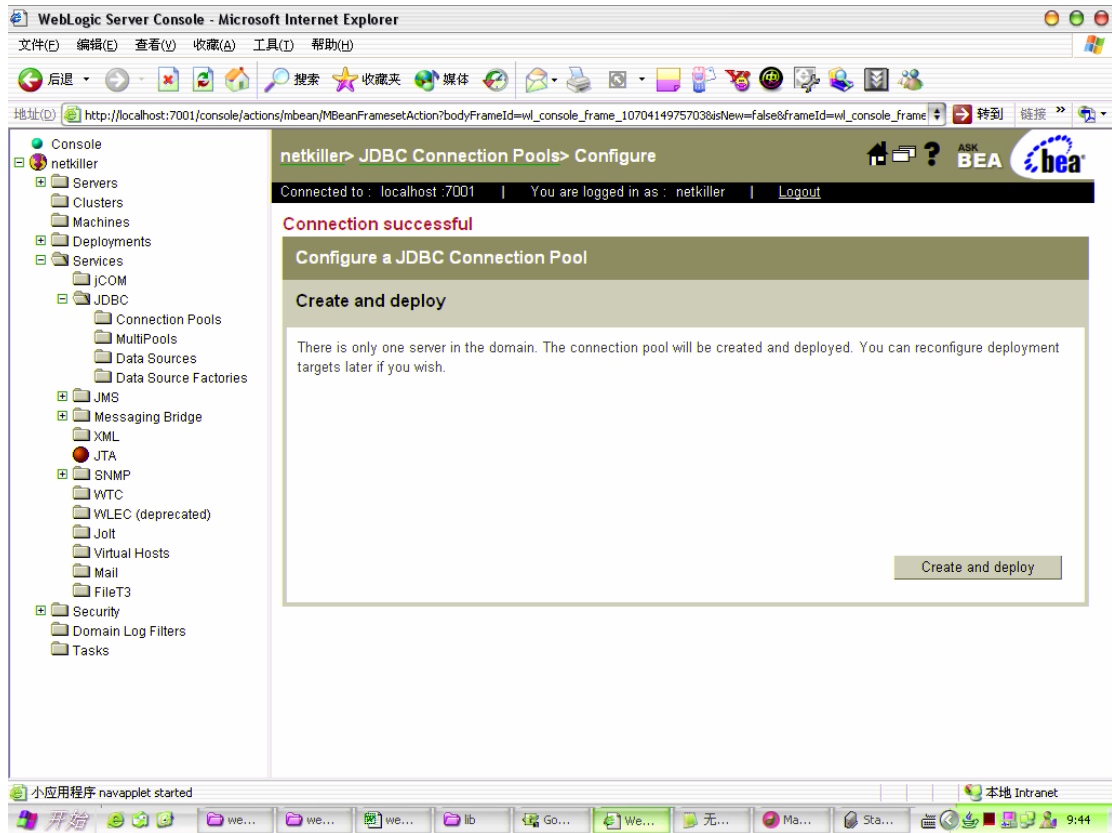
Host Name  
Port  
Database User Name  
Password  
Confirm Password  
Continue



19.

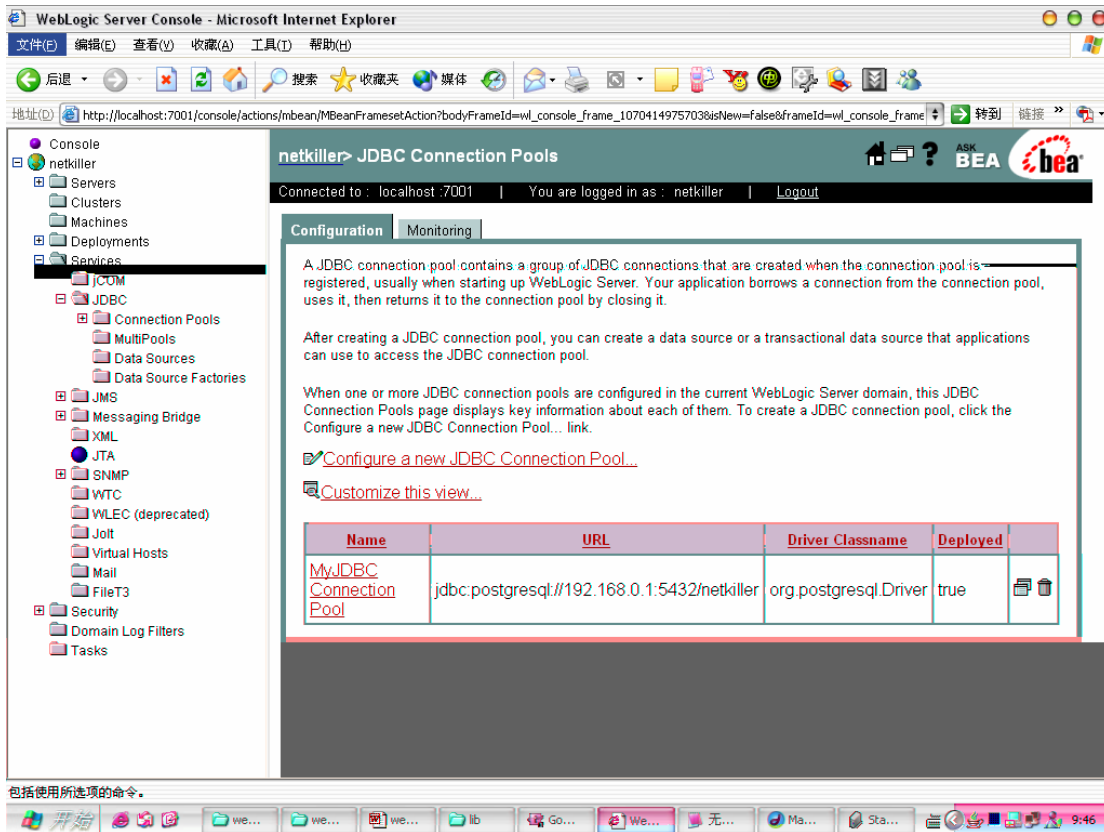


## 1. Test Driver Configuration Connection successful



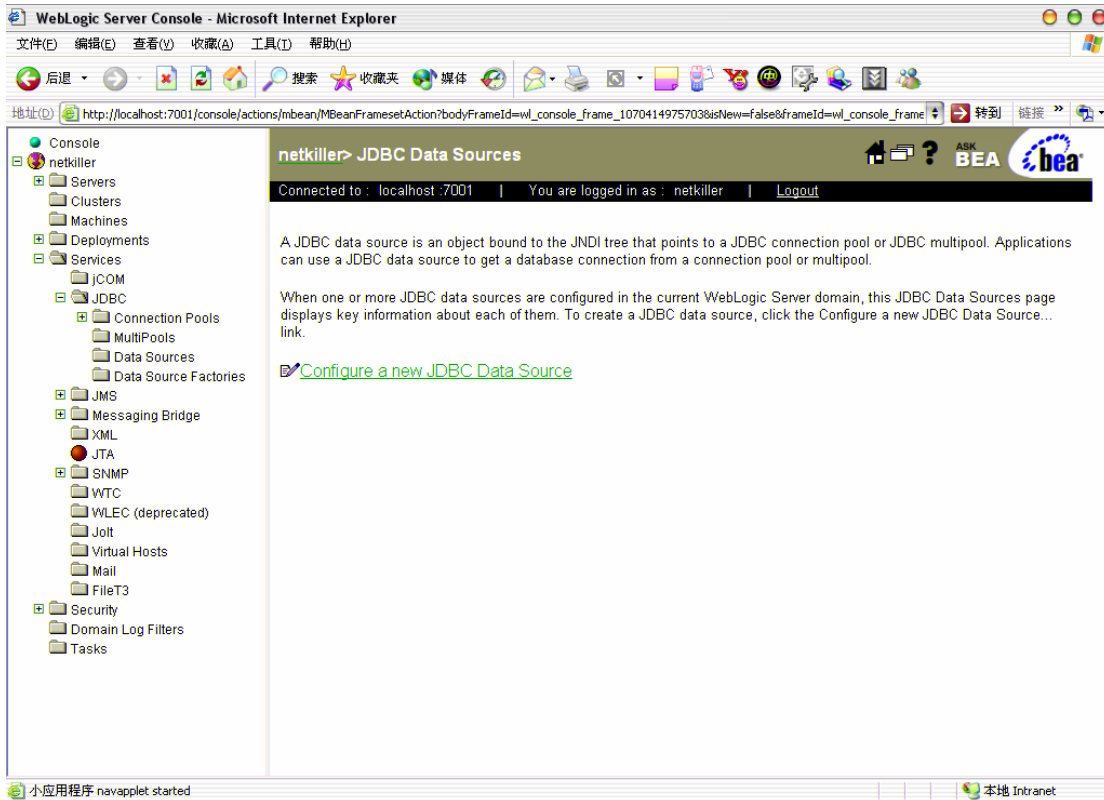
## 2. Create and deploy



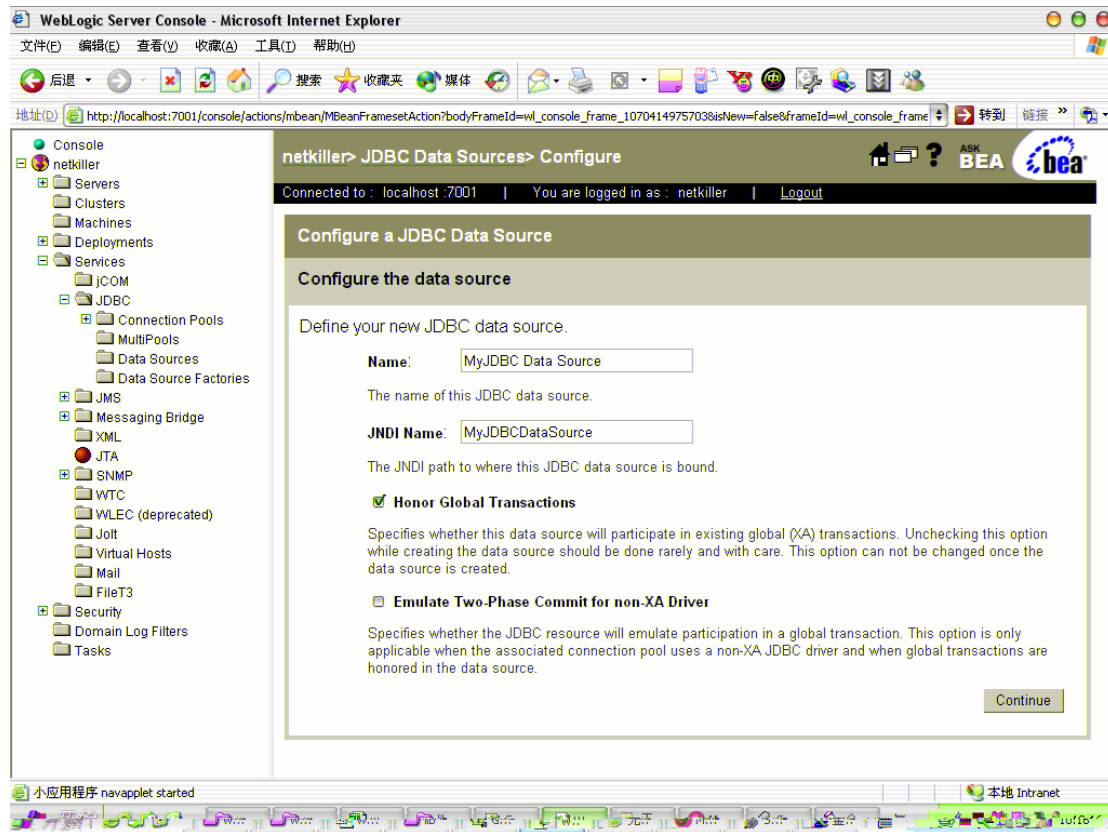


3.

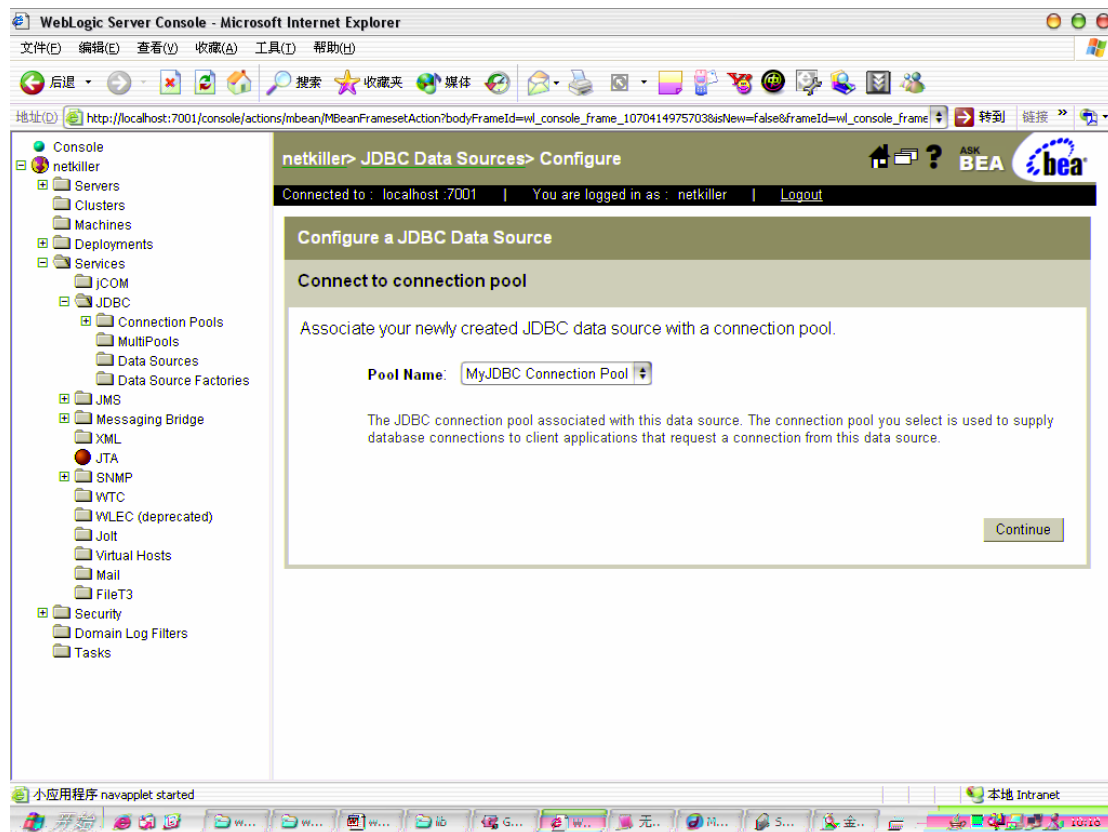
### 1. netkiller-->Service-->JDBC--> Data Source



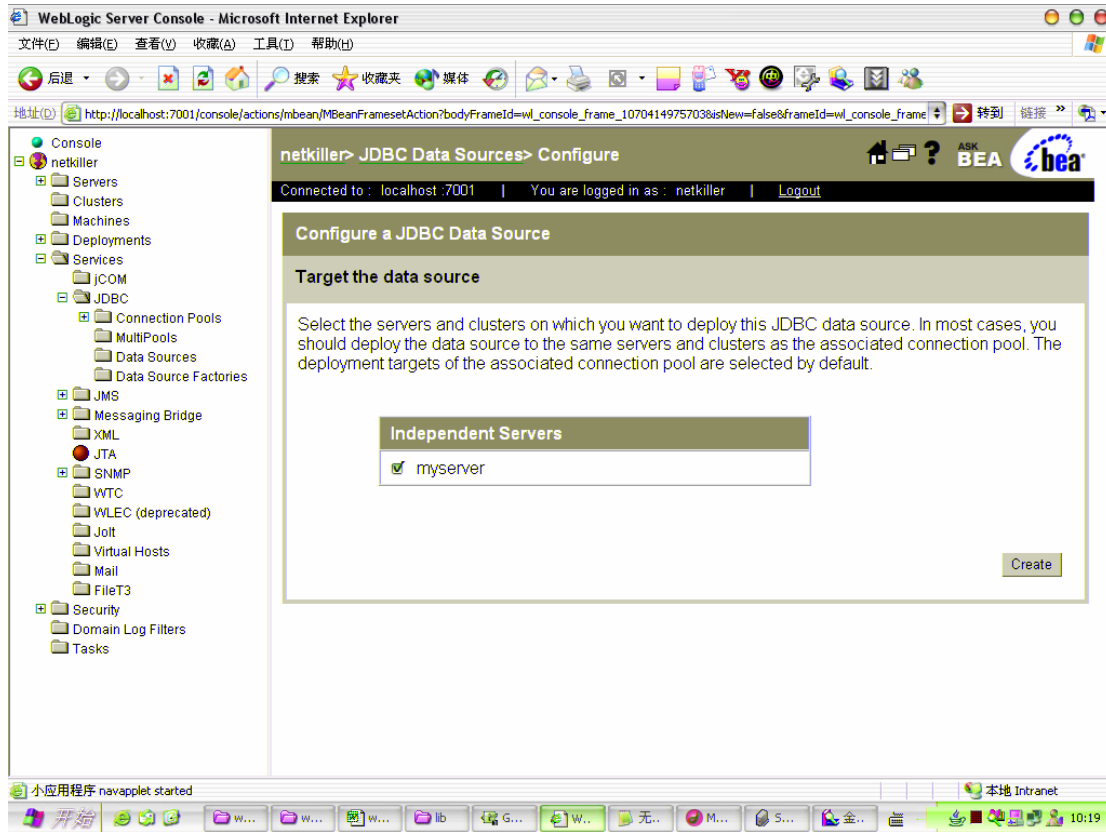
2. Configure a new JDBC Data Source
3. JNDI Name            MyJDBCDataSource



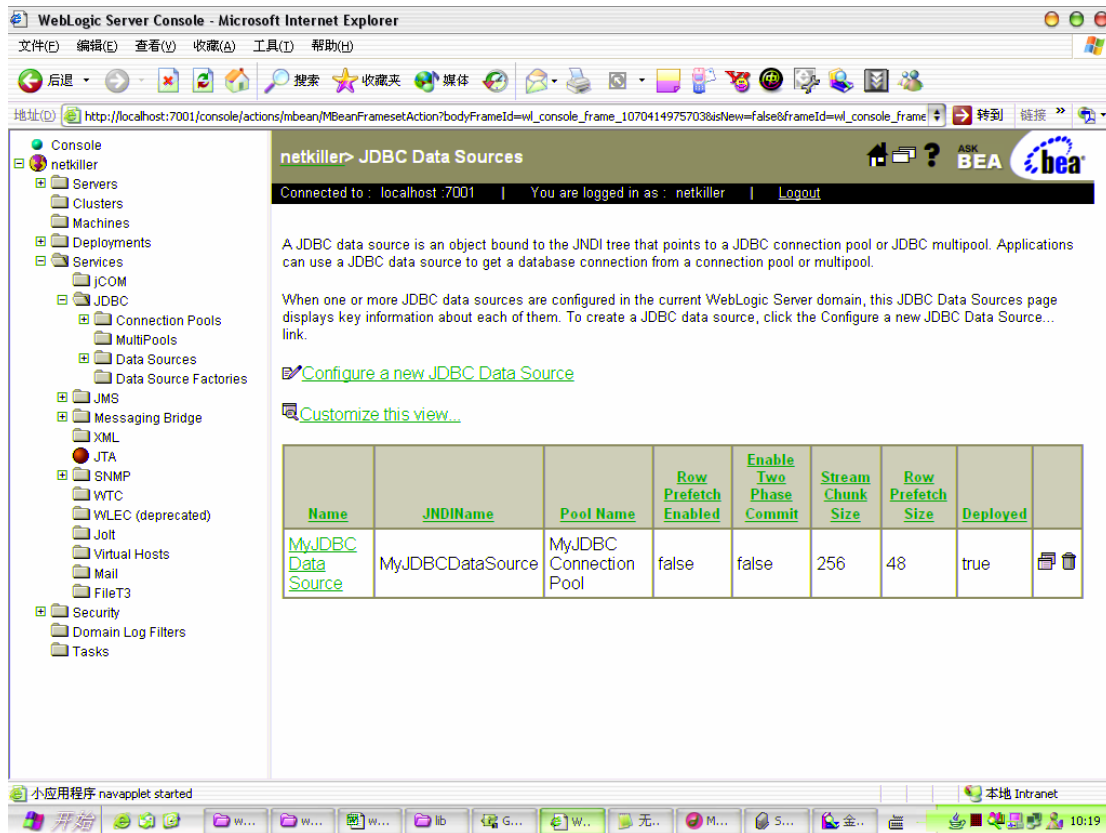
4. Continue



5. Continue

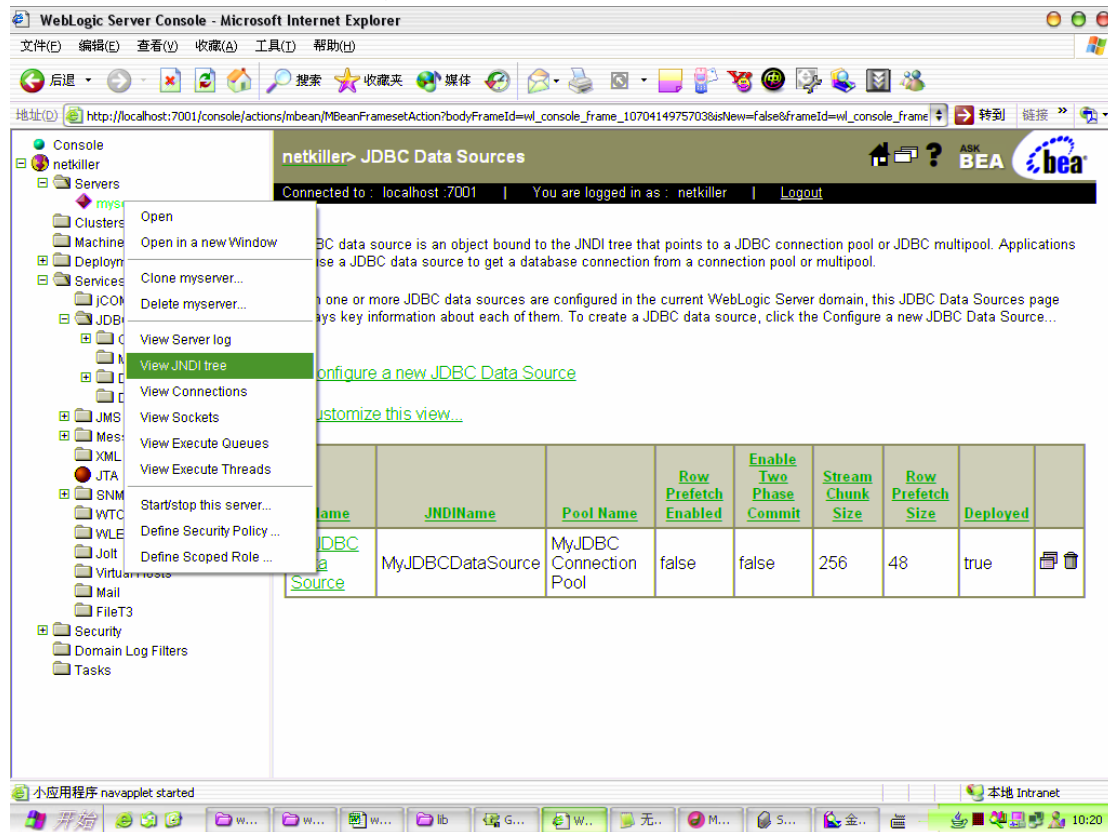


6. Create

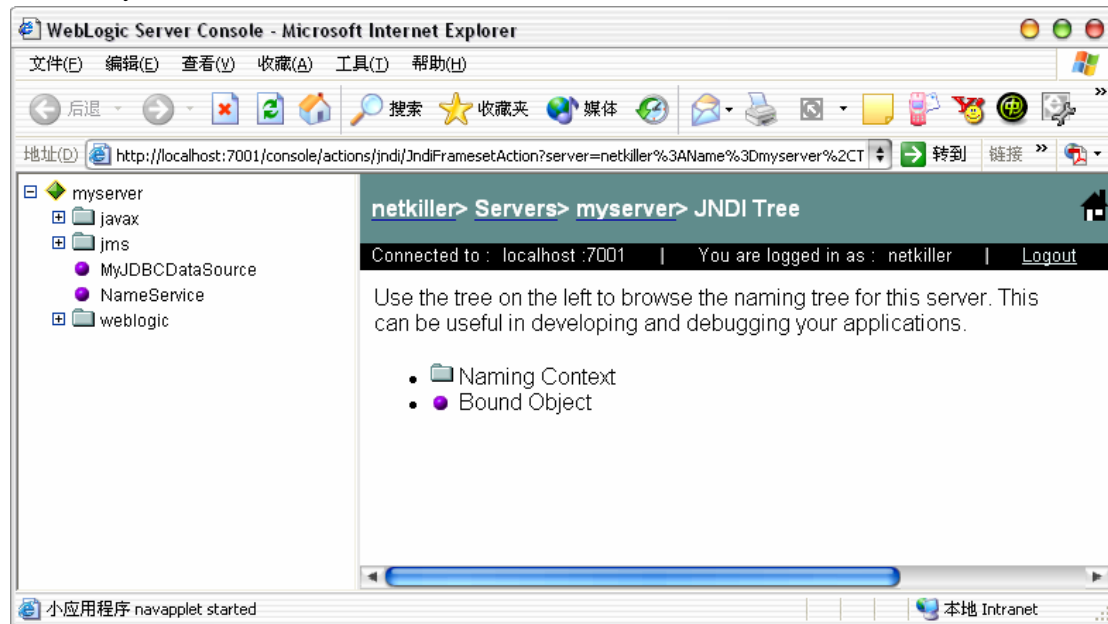


7.

### 1. netkiller Servers myserver

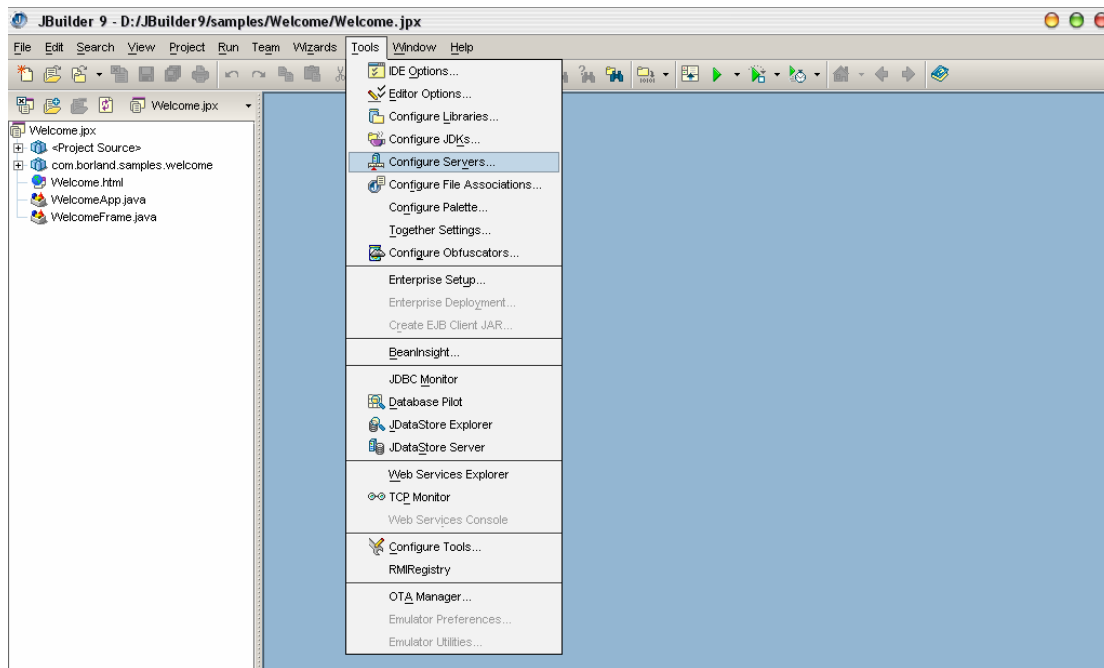


### 2. myserver View JNDI tree

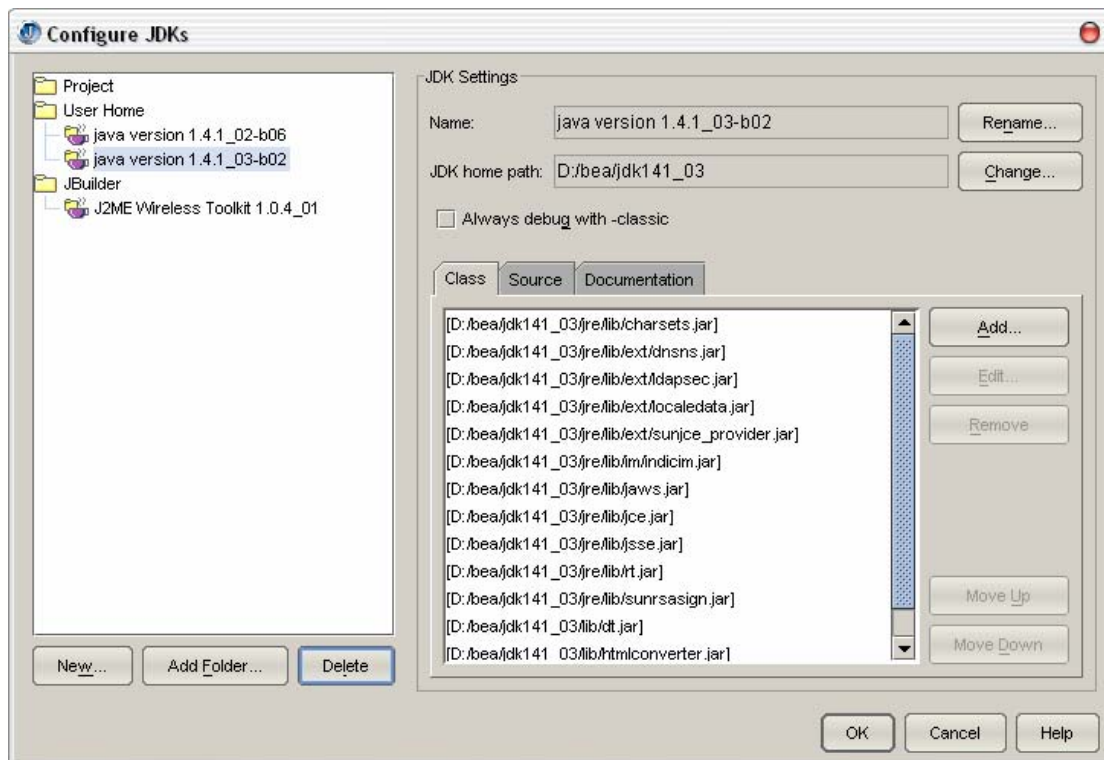


myserver                      myJDBCDataSource

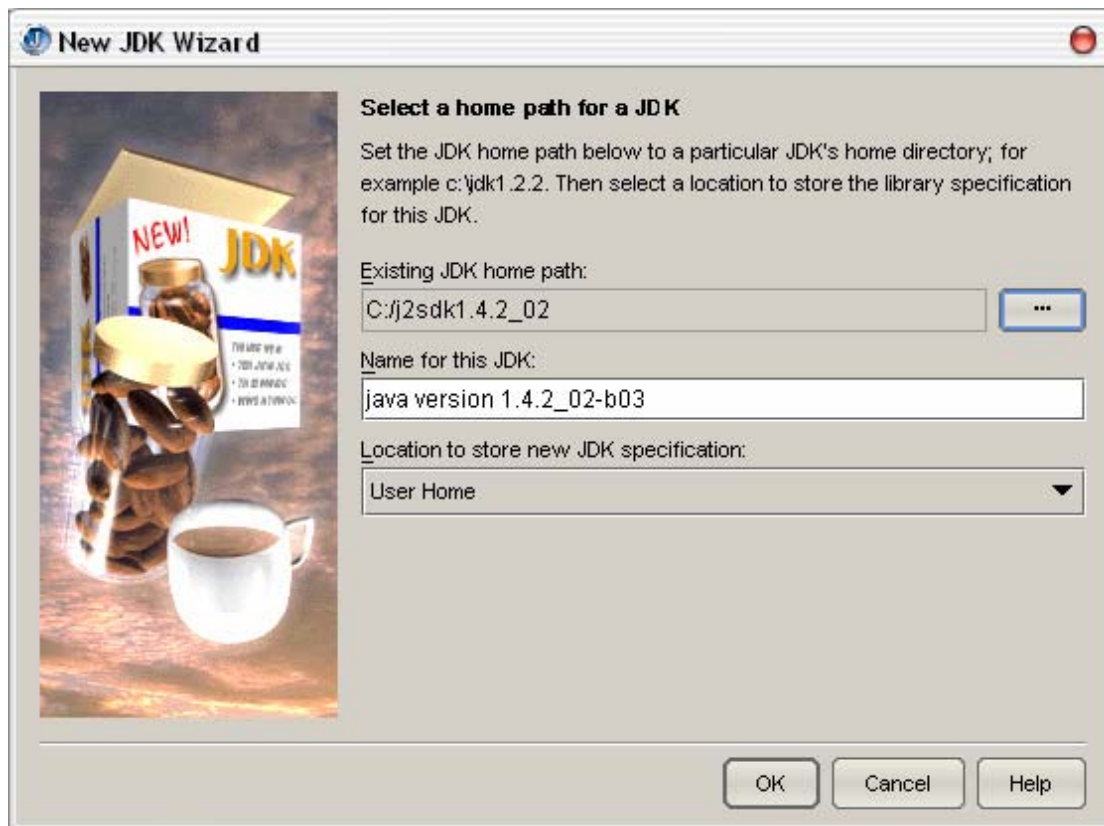
### 3.



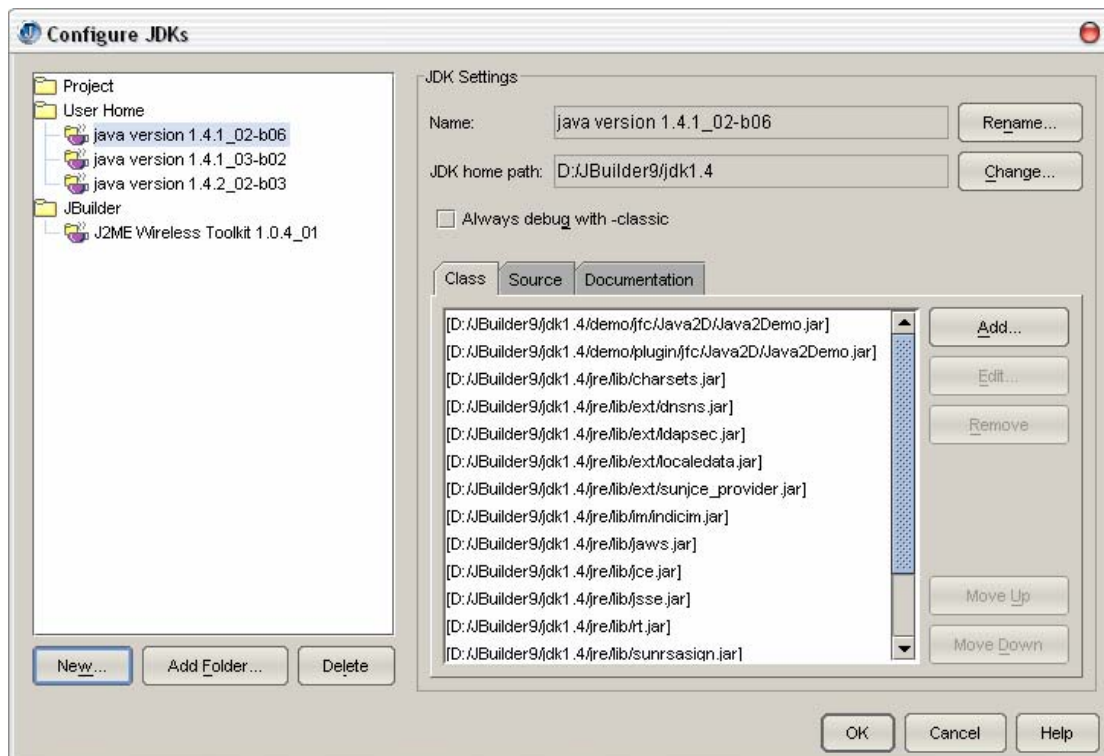
## 1. Tools Configure JDKs



## 2. New

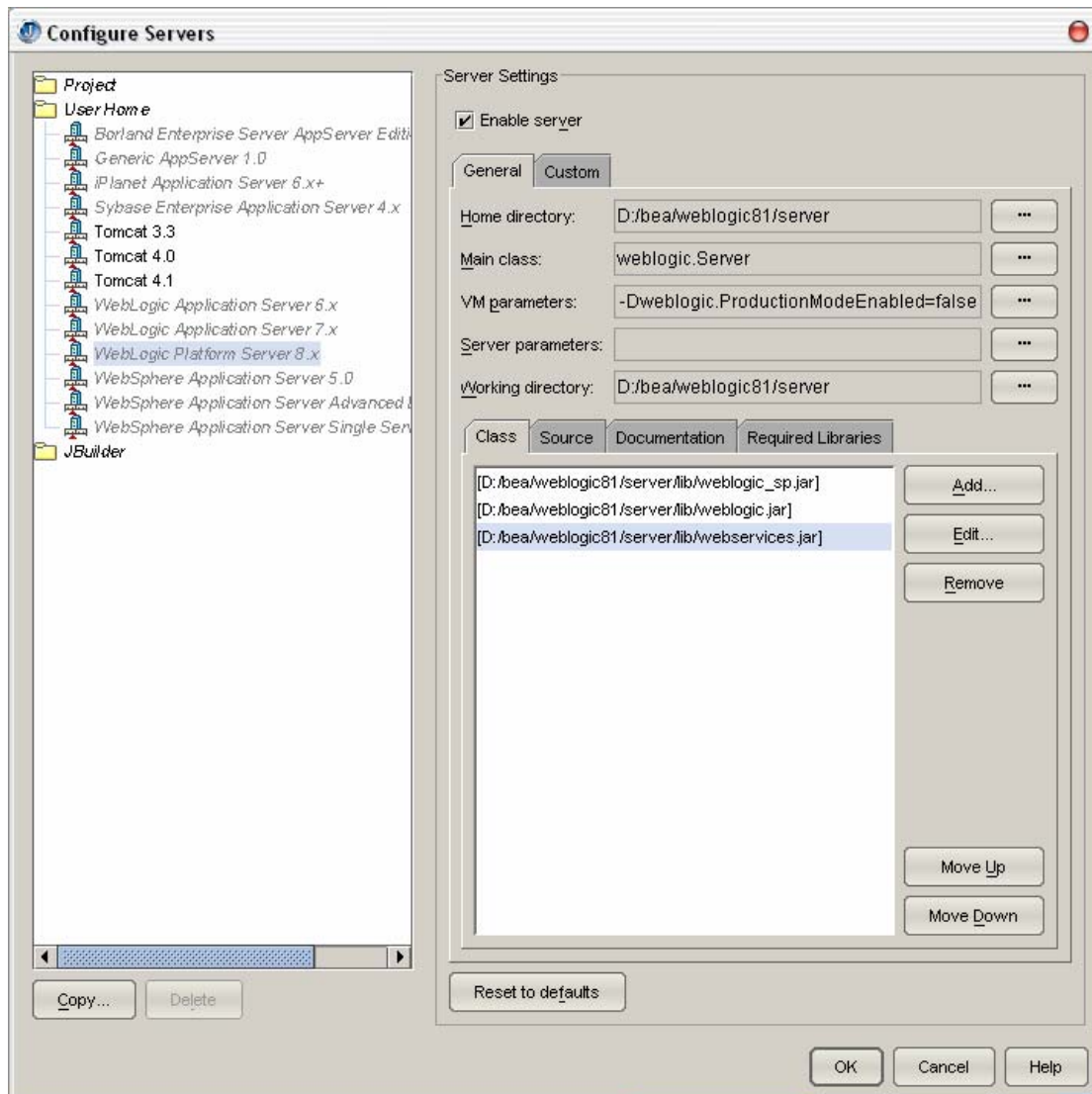


3. JDK

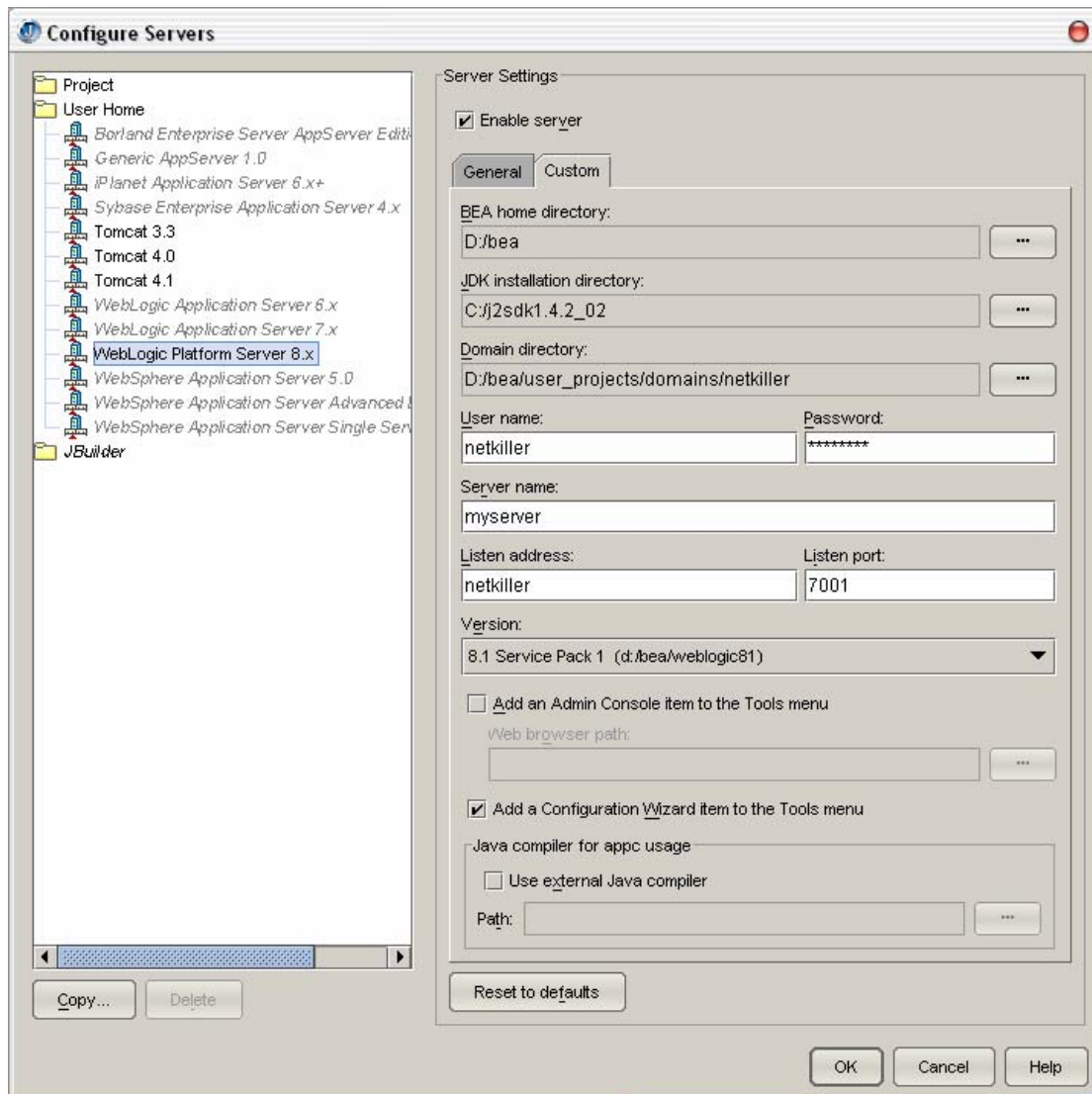


4. OK

5. Tools Configure Servers



6. Weblogic Platform Server 8.x
7.  Enable Server
8. Home directory        D:/bea/weblogic81/server
9.                        Custom



10.

BEA home directory d:/bea

JDK Installation directory JDK

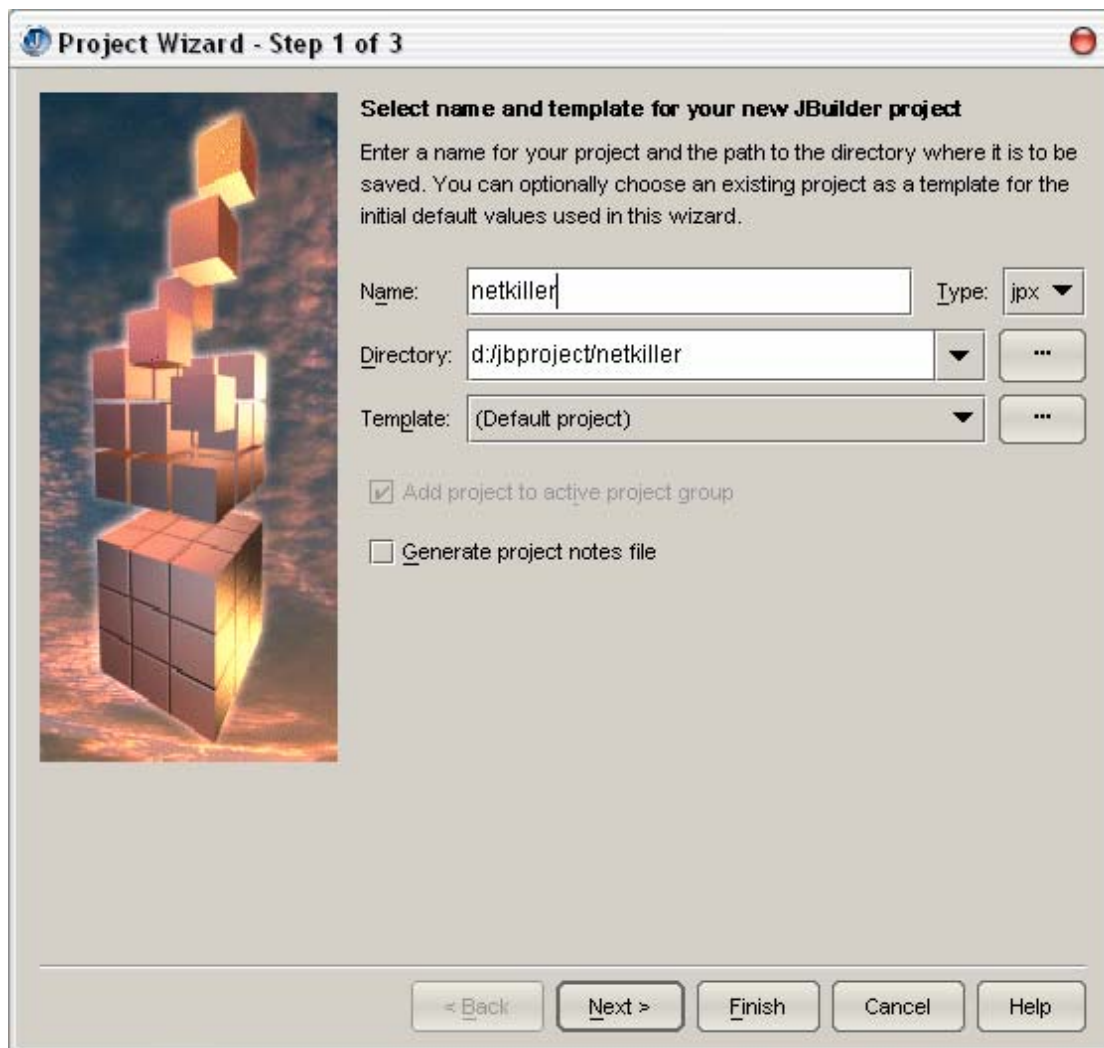
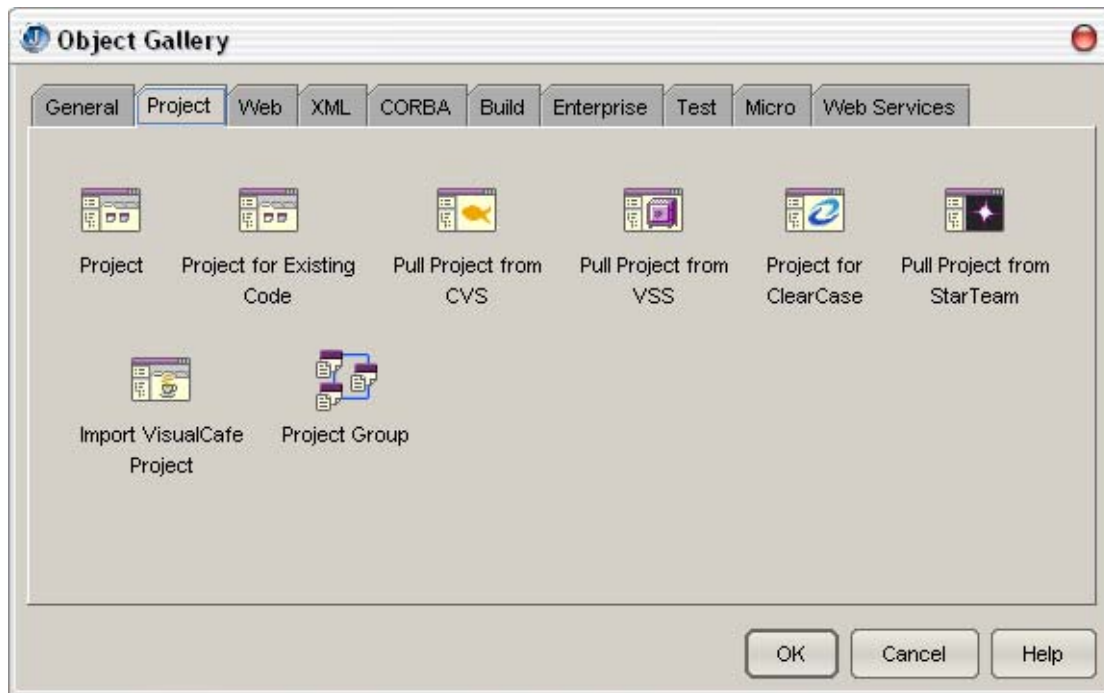
Domain directory D:/bea/user\_projects/domains/netkiller

\*\*\*\*\*

Add an Console item to the Tools menu

OK





**Project Wizard - Step 2 of 3**

**Specify project paths**

Edit the paths and settings here to help define your new project. These and other properties can be changed after the project is created.

JDK:  ...

Output path:  ...

Backup path:  ...

Working directory:  ...

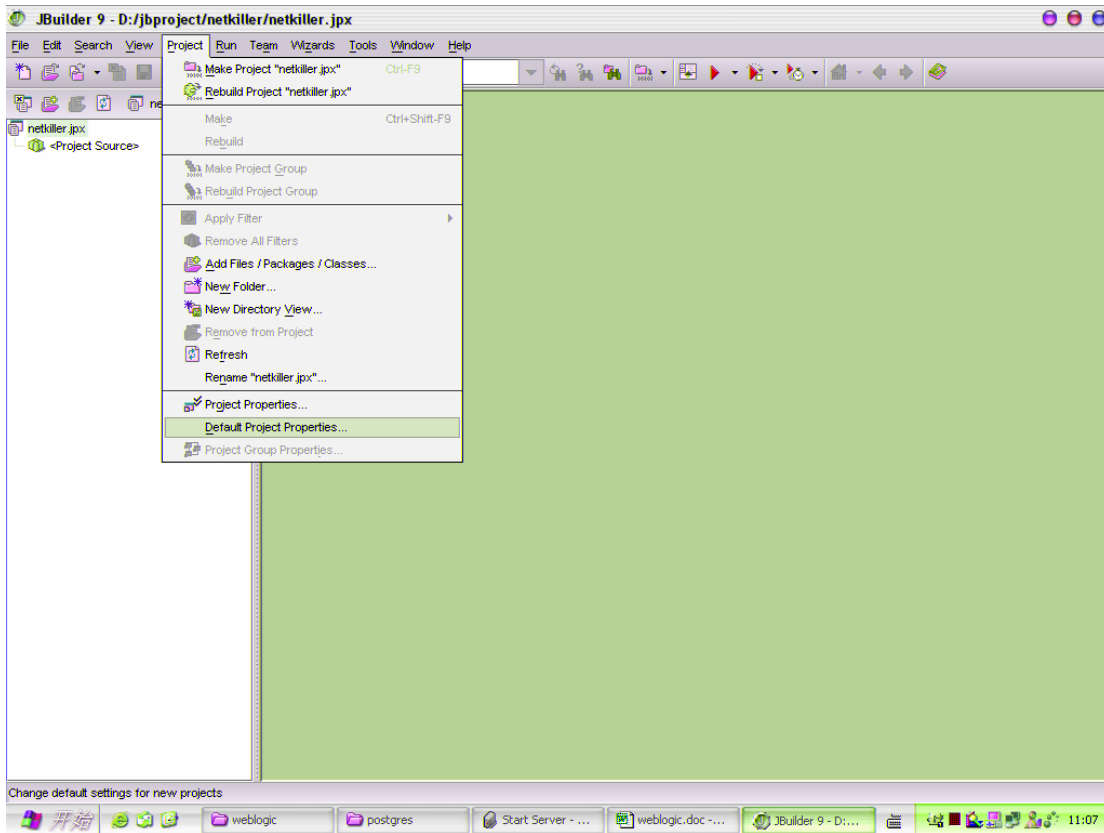
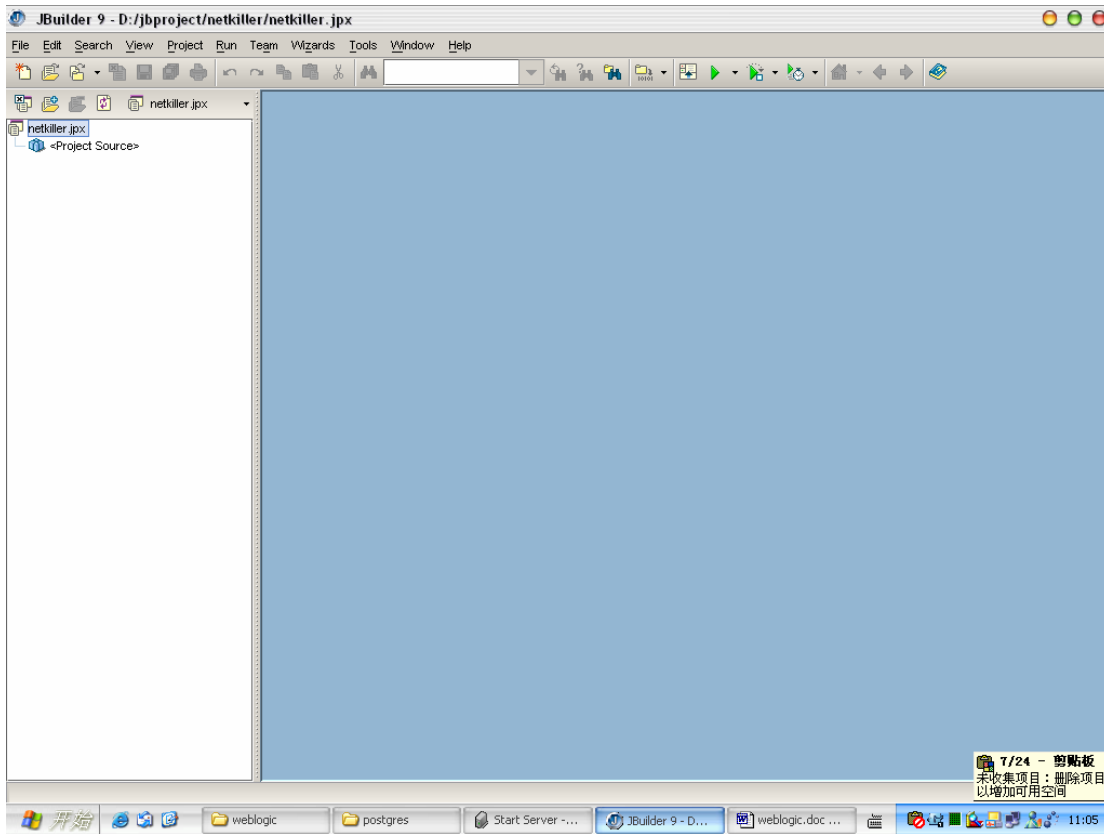
Source   Documentation   Required Libraries

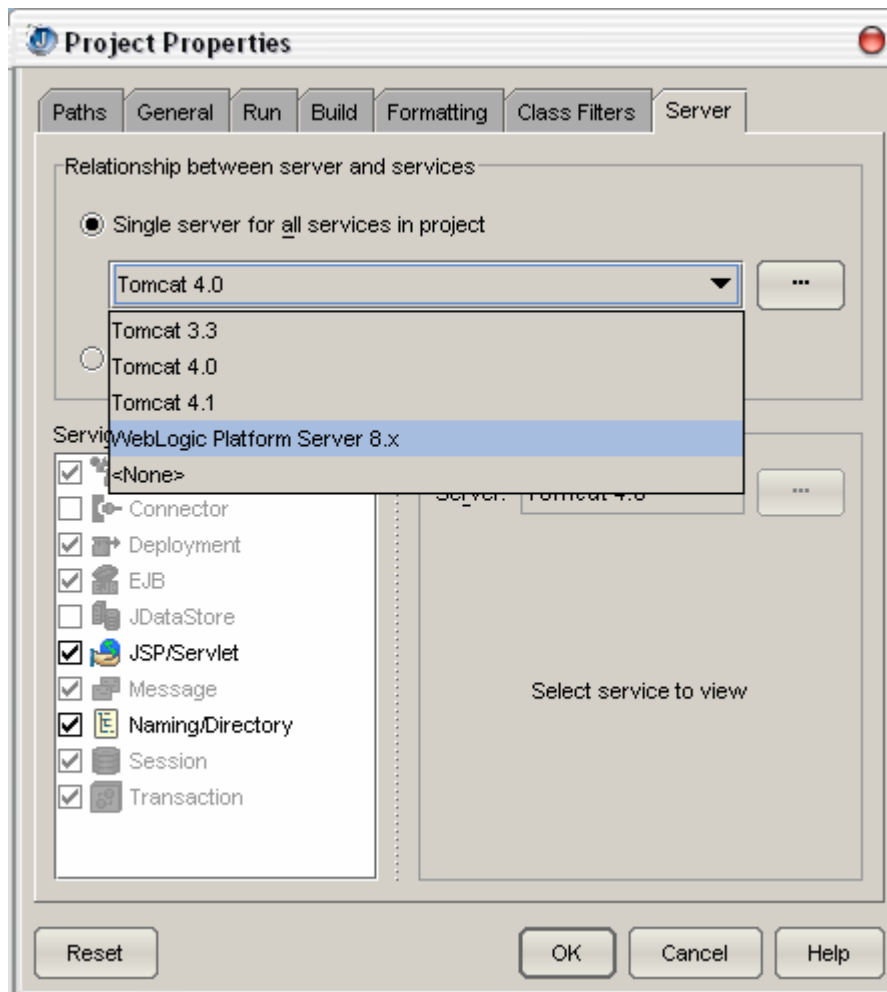
Default	Test	Path
<input checked="" type="radio"/>	<input type="radio"/>	d:/jbproject/netkiller/src
<input type="radio"/>	<input checked="" type="radio"/>	d:/jbproject/netkiller/test

Add...  
Edit...  
Remove  
Move Up  
Move Down

< Back   Next >   Finish   Cancel   Help







OK





**Select EJB module**

Select the EJB module to which to write EJB deployment information. If your project does not have an existing EJB module you may create one using the New button.

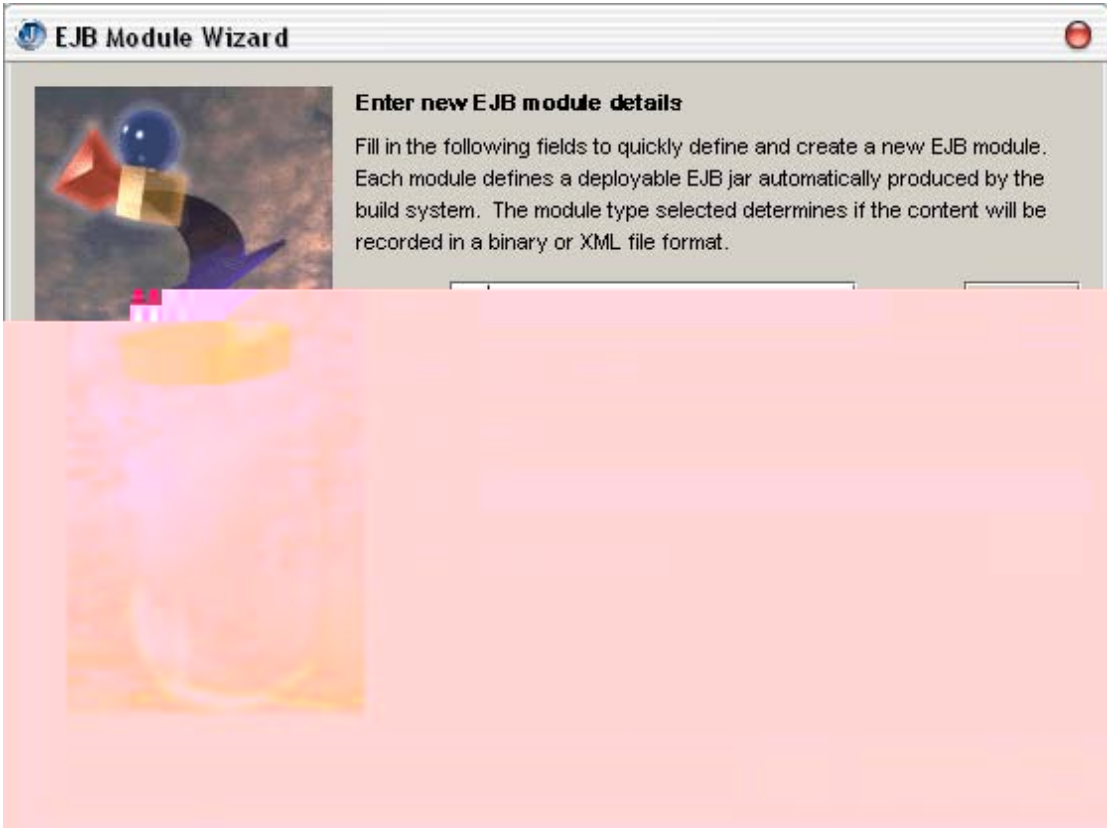
EJB 2.0 compliant modules in project:

New...

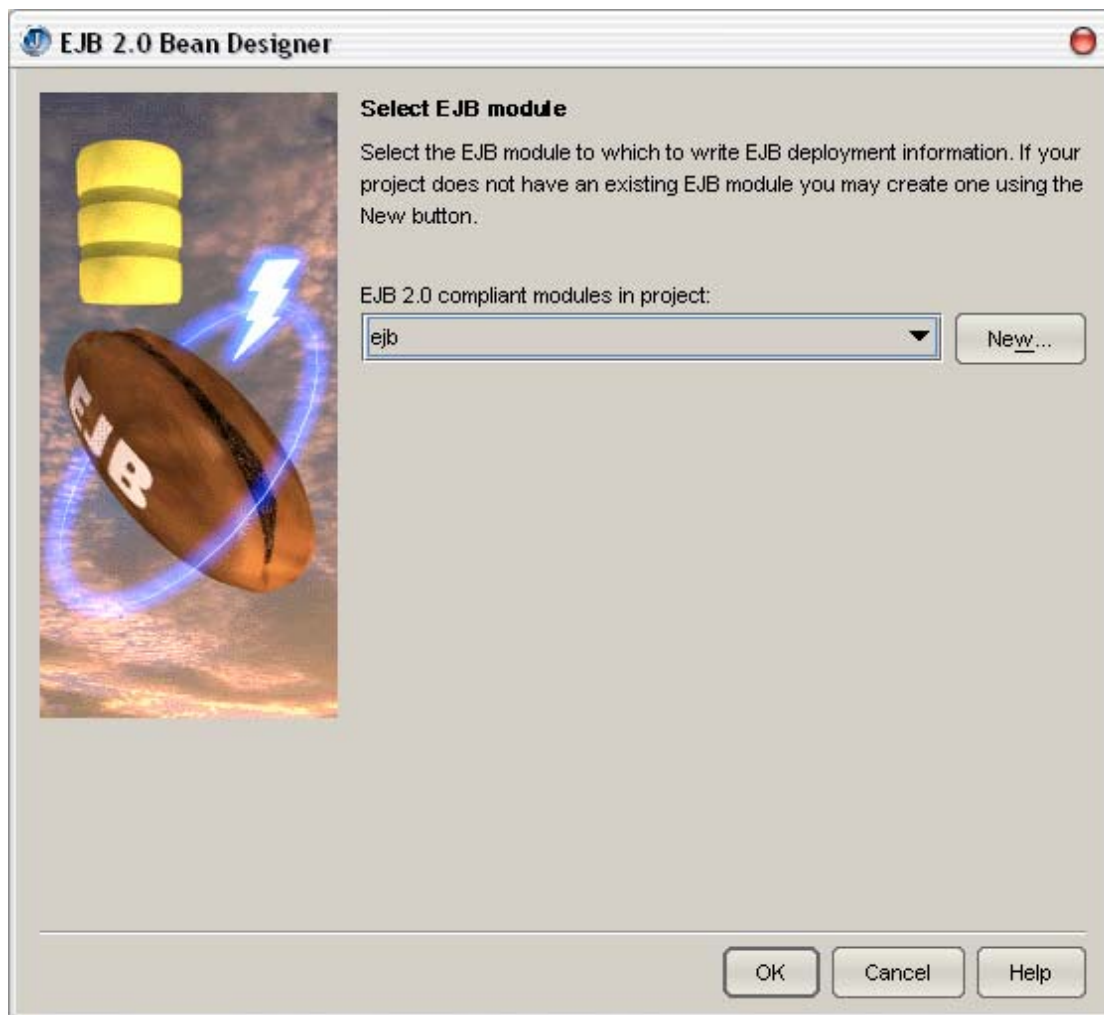
OK

Cancel

Help

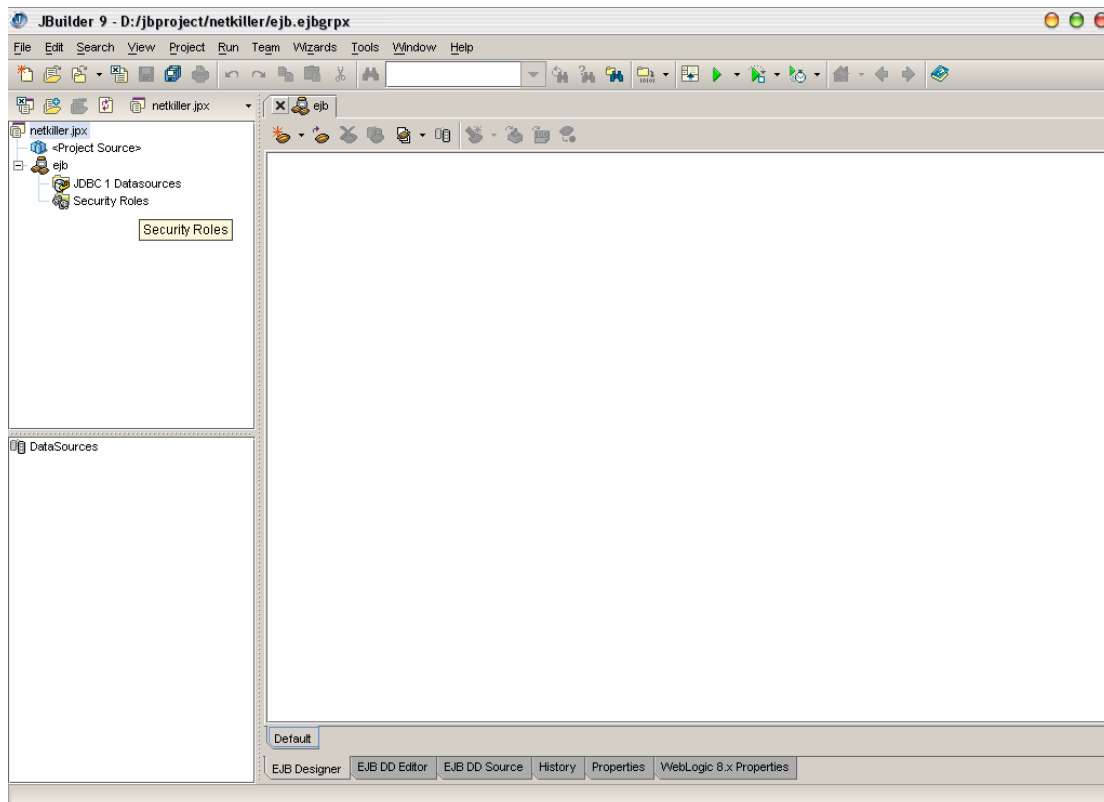


OK



OK





## 15.5 GUI 数据库管理与设计（建模）工具

### 15.5.1 phpPgAdmin

PHP WEB

<http://phppgadmin.sourceforge.net/>

apache

samba phppgAdmin-3.3.tar.gz /var/www/html

phpPgAdmin

```
[root@linux root]# cd /var/www/html
[root@linux html]# tar zxvf phppgAdmin-3.3.tar.gz
[root@linux html]# cd phppgAdmin
[root@linux conf]# cd conf/
[root@linux conf]# vi config.inc.php
```

```
$conf['servers'][0]['host'] = '127.0.0.1';
$conf['servers'][0]['defaultdb'] = 'netkiller';
```

```
// Display name for the server on the login screen
```

```
$conf['servers'][0]['desc'] = 'PostgreSQL';

// Hostname or IP address for server. Use " for UNIX domain socket.
$conf['servers'][0]['host'] = '127.0.0.1';

// Database port on server (5432 is the PostgreSQL default)
$conf['servers'][0]['port'] = 5432;

// Change the default database only if you cannot connect to template1
$conf['servers'][0]['defaultdb'] = 'netkiller';
```

IE <http://192.168.0.1/phpPgAdmin/>

## phpPgAdmin 3.3 Login

**Username:**

**Password:**

**Server:** PostgreSQL

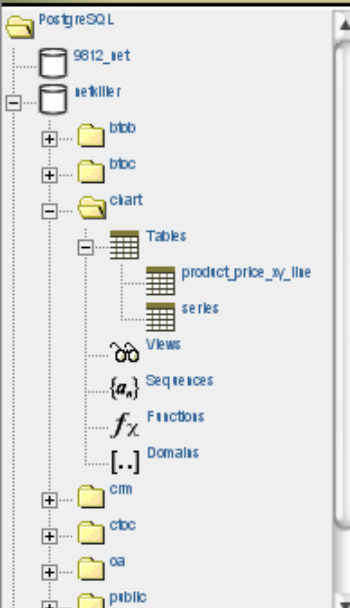
**Language:** English

phpPgAdmin PostgreSQL 7.3.4 running on 127.0.0.1:5432 — You are logged in as user "netkiller", 1st Mar, 2004 7:07 PM

[Account](#) | [Reports](#) | [SQL](#) | [Find](#) | [Logout](#)

### netkiller: Schemas: chart

- Tables
- Views
- Sequences
- Functions
- Domains



## 15.5.2 PgAdmin III

<http://www.pgadmin.org/>

<http://www.pgadmin.org/pgadmin3/download.php>

[http://www.pgadmin.org/pgadmin3/download.php?locale=zh\\_CN](http://www.pgadmin.org/pgadmin3/download.php?locale=zh_CN)



The screenshot shows the pgAdmin III web interface. The left pane displays a tree view of a PostgreSQL server (192.168.0.1) with the following structure:

- 服务器 (2)
  - 192.168.0.1 (192.168.0.1:5432)
    - 数据库 (4)
      - 9812\_net
      - netkiller
        - 类型转换 (0)
        - 语言 (1)
        - 模式 (7)
          - public
            - 聚集 (0)
            - 转换 (0)
            - 域 (0)
            - 函数 (7)
            - 触发器函数 (3)
            - 操作符 (0)
            - 操作符类 (0)
            - 序列 (9)
            - 表 (13)
            - 类型 (0)
            - 查看 (6)
          - btob
          - btoc
          - chart
          - crm
          - ctoc

The right pane shows the properties window for the 'netkiller' database:

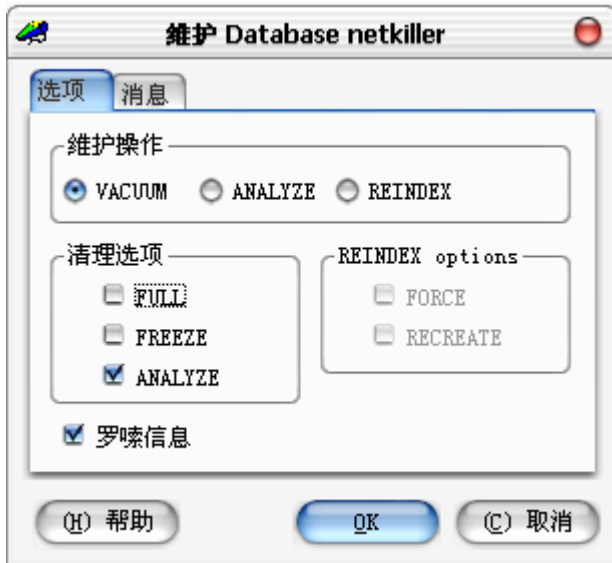
属性	值
名字	netkiller
OID	16978
所有者	netkiller
ACL	
编码	UNICODE
允许连接?	是
连接?	是
系统数据库?	不
注释	

Below the table, the SQL command window shows the following code:

```
-- Database: netkiller
-- DROP DATABASE netkiller;

CREATE DATABASE netkiller
  WITH ENCODING = 'UNICODE';
```

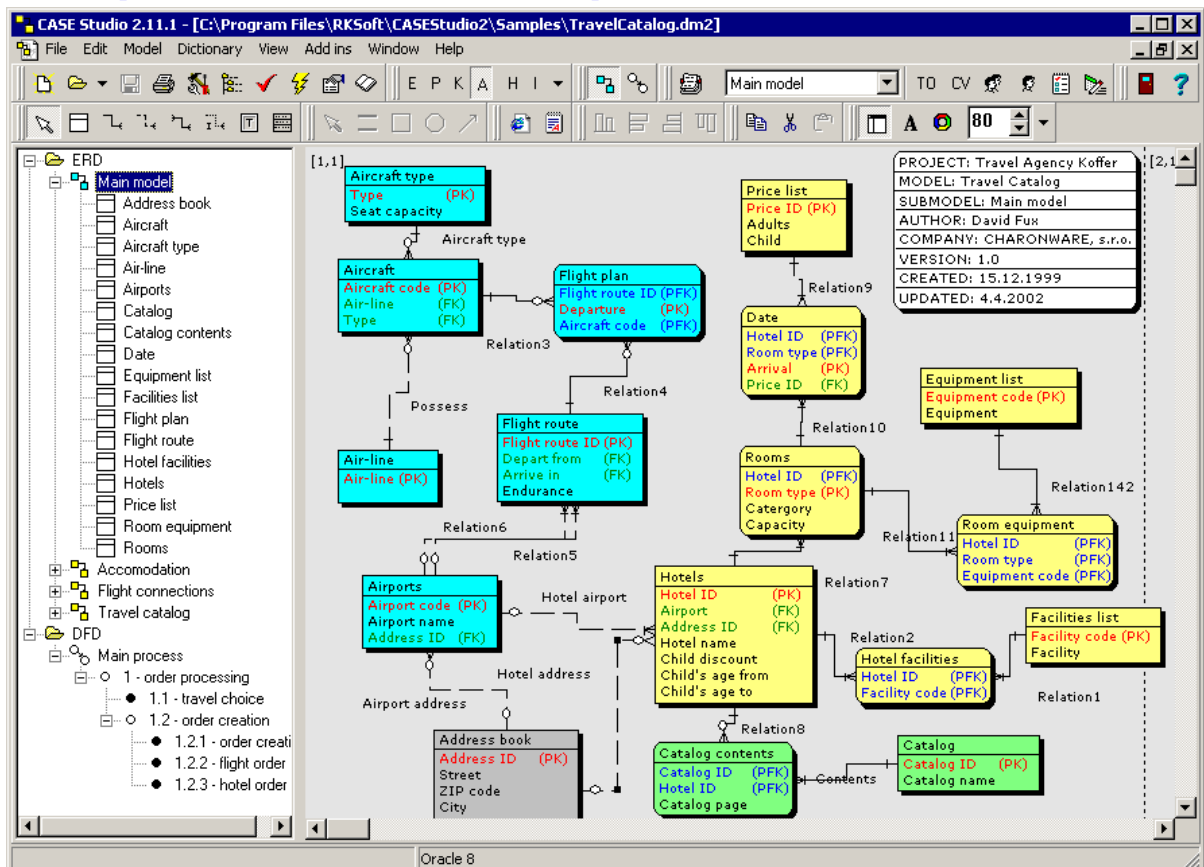
The status bar at the bottom indicates: 检索数据库细节...完成. 0.00 秒

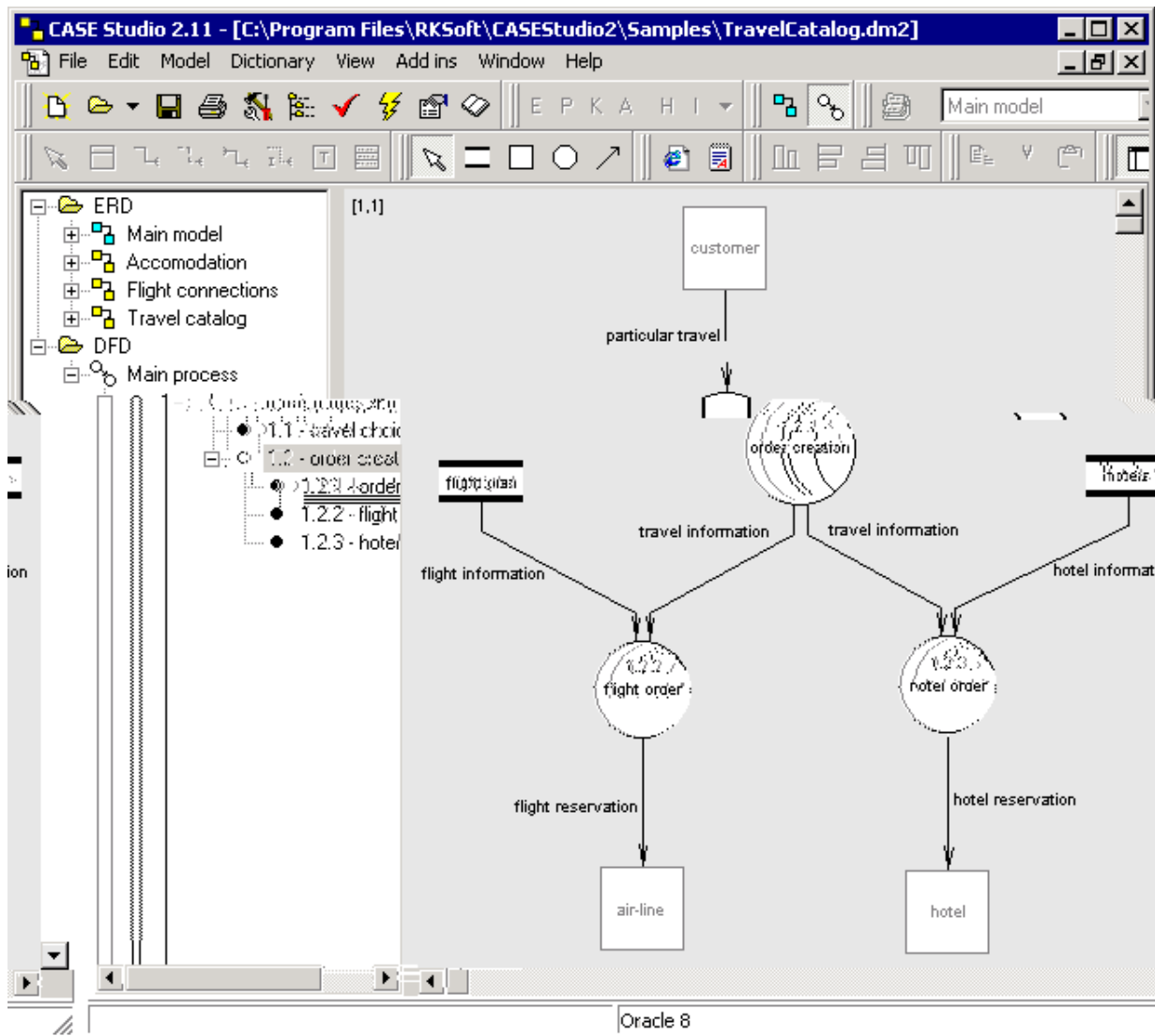


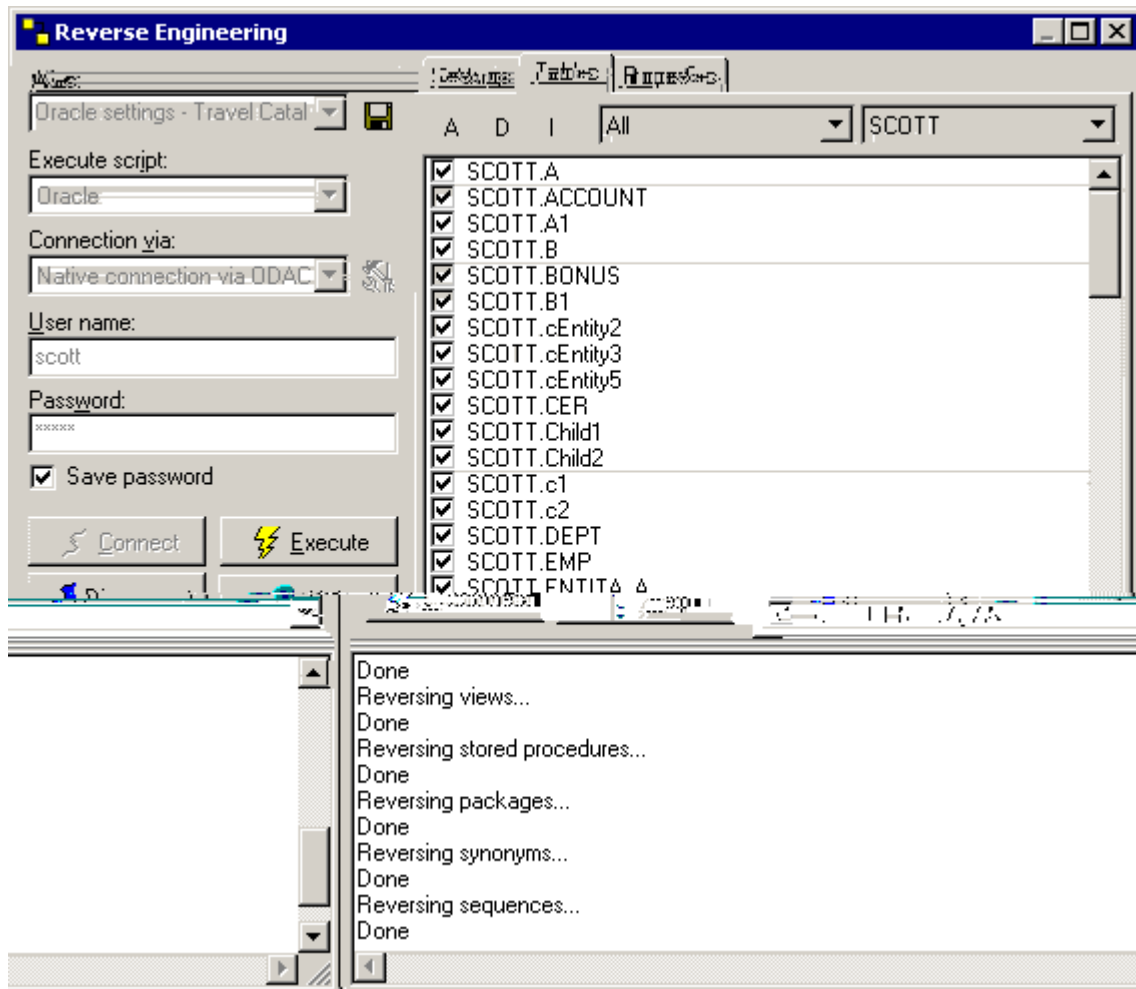
## 15.5.3 Case Studio 2

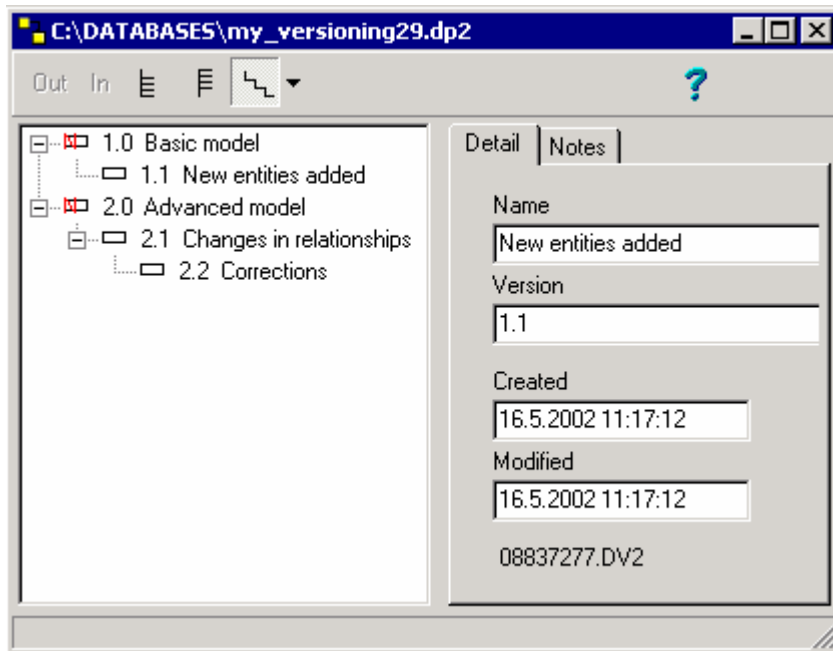
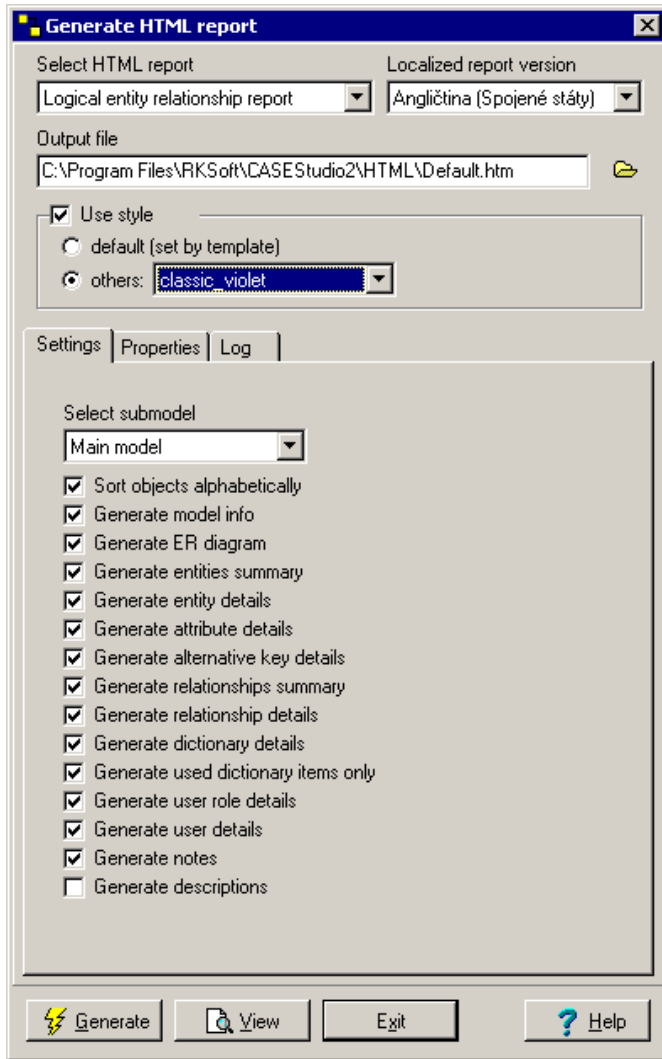
<http://www.casestudio.com/>

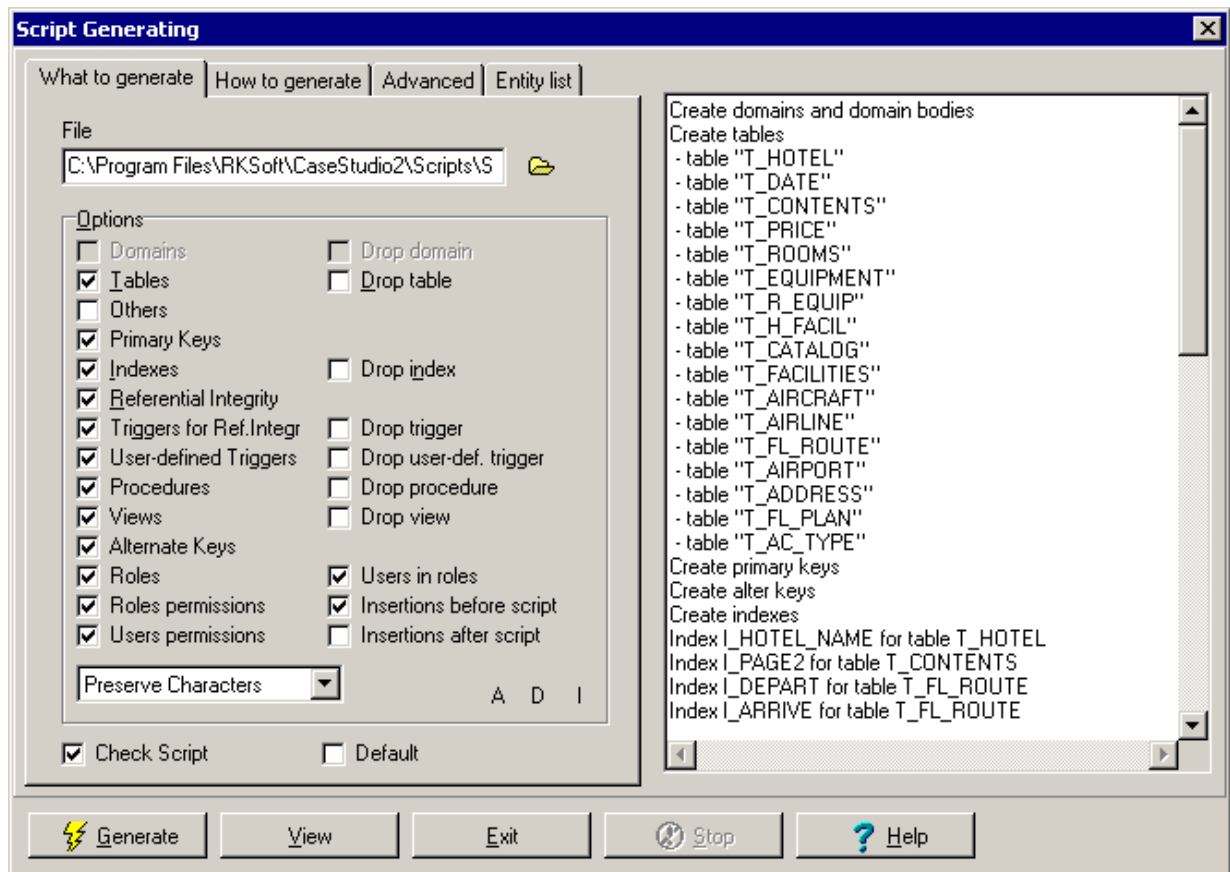
<http://www.casestudio.com/enu/download.aspx>











## CASE Studio 2 - Feature

LITE

FULL

### Entity relationship diagrams

Clipper (older version)



DBI SAM 3



IBM DB2 UDB ver. 8.1



IBM DB2 UDB ver. 7.1



Informix (older version)



Informix 9



Ingres (older version)



InterBase 7



InterBase 6 SQL 3



InterBase 6 SQL 1



InterBase 5



InterBase 4



MS Access 2000



MS Access 97



MS SQL 2000



MS SQL 7



MS SQL 6.5



MySQL 4 (incl. innodb)



MySQL 3.23



Oracle 9





Oracle 8	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Oracle 7	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Paradox (older version)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Pervasive	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PostgreSQL 7.3 BETA	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
PostgreSQL 7	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Sybase 12.5	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

### Data flowdiagram

Data flows	-	<input checked="" type="checkbox"/>
Data stores	-	<input checked="" type="checkbox"/>
Processes	-	<input checked="" type="checkbox"/>
Terminators	-	<input checked="" type="checkbox"/>

### Reverse engineering

Clipper (older version)	-	-
DBI SAM3	-	<input checked="" type="checkbox"/>
IBM DB2 UDB ver. 7	-	<input checked="" type="checkbox"/>
IBM DB2 UDB ver. 8	-	<input checked="" type="checkbox"/>
Informix (older version)	-	-
Informix 9	<input checked="" type="checkbox"/>	-
Ingres (older version)	-	-
InterBase 7	-	<input checked="" type="checkbox"/>
InterBase 6 SQL 3, InterBase 6 SQL 1	-	<input checked="" type="checkbox"/>
InterBase 5	-	<input checked="" type="checkbox"/>
InterBase 4	-	<input checked="" type="checkbox"/>
MS Access 2000	-	<input checked="" type="checkbox"/>
MS Access 97	-	<input checked="" type="checkbox"/>
MS SQL 2000	-	<input checked="" type="checkbox"/>
MS SQL 7	-	<input checked="" type="checkbox"/>
MS SQL 6.5	-	<input checked="" type="checkbox"/>
MySQL 4 (incl. innodb)	-	<input checked="" type="checkbox"/>
MySQL 3.23	-	<input checked="" type="checkbox"/>
Oracle 9	-	<input checked="" type="checkbox"/>
Oracle 8	-	<input checked="" type="checkbox"/>
Oracle 7	-	<input checked="" type="checkbox"/>
Paradox (older version)	-	<input checked="" type="checkbox"/>
Pervasive	<input checked="" type="checkbox"/>	-
PostgreSQL 7.3 BETA	-	<input checked="" type="checkbox"/>
PostgreSQL 7	-	<input checked="" type="checkbox"/>
Sybase 12.5	-	<input checked="" type="checkbox"/>

### Reports & export possibilities

RTF reports	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
-------------	-------------------------------------	-------------------------------------

HTML reports	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Export into JPG, BMP or PNG	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Export into XML	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

## Features

After script items	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Alternate keys	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Alter scripts	-	-
Autolayout	-	-
Before script items	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Cardinality	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Data dictionary (domains)	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Descriptions	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Drops generation	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Foreign keys definition	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Functions	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Indexes	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Informative relationships	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Internal clipboard	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
MIN relationships	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Network features	-	-
New user defined data types	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Non-identifying relationships	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Notes	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Object type bodies*	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Object types*	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Package bodies*	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Packages*	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Patterns for views, procedures and triggers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Primary keys definition	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Referential integrity	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Rough uppercase/lowercase selection	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Self relationships	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Sequences*	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
SQL script generation	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Storages	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Stored procedures	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Support of compound foreign keys	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Synonyms*	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Triggers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
User permissions (in models)	-	<input checked="" type="checkbox"/>
Users (in models)	-	<input checked="" type="checkbox"/>
Users group (in models)	-	<input checked="" type="checkbox"/>
Views	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

## Model maintenance

Database/Model conversion**	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Gallery	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Model check	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Submodels	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
To-Do list	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Version comparison	-	<input checked="" type="checkbox"/>
Version manager	-	<input checked="" type="checkbox"/>

## Scripting and user defined templates

CLE Automation objects	-	<input checked="" type="checkbox"/>
Large COM interface	-	<input checked="" type="checkbox"/>
MS Scripting	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Jscript support	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
VB script support	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Templates editor	-	<input checked="" type="checkbox"/>
Templates export	-	<input checked="" type="checkbox"/>
Templates import	-	<input checked="" type="checkbox"/>
User defined variables editor	-	<input checked="" type="checkbox"/>

## Displaying

Visual creation of DFD	-	<input checked="" type="checkbox"/>
Visual creation of ERD	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Alignment in columns	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Attributes syntax highlighting	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Entity background color selection	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Font selection	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Logical model	-	-
Physical model	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Logical view/ physical view	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Model background color selection	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Shadows	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Stamp	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Straight relationship lines	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Text descriptions	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Variable display level	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

## Support

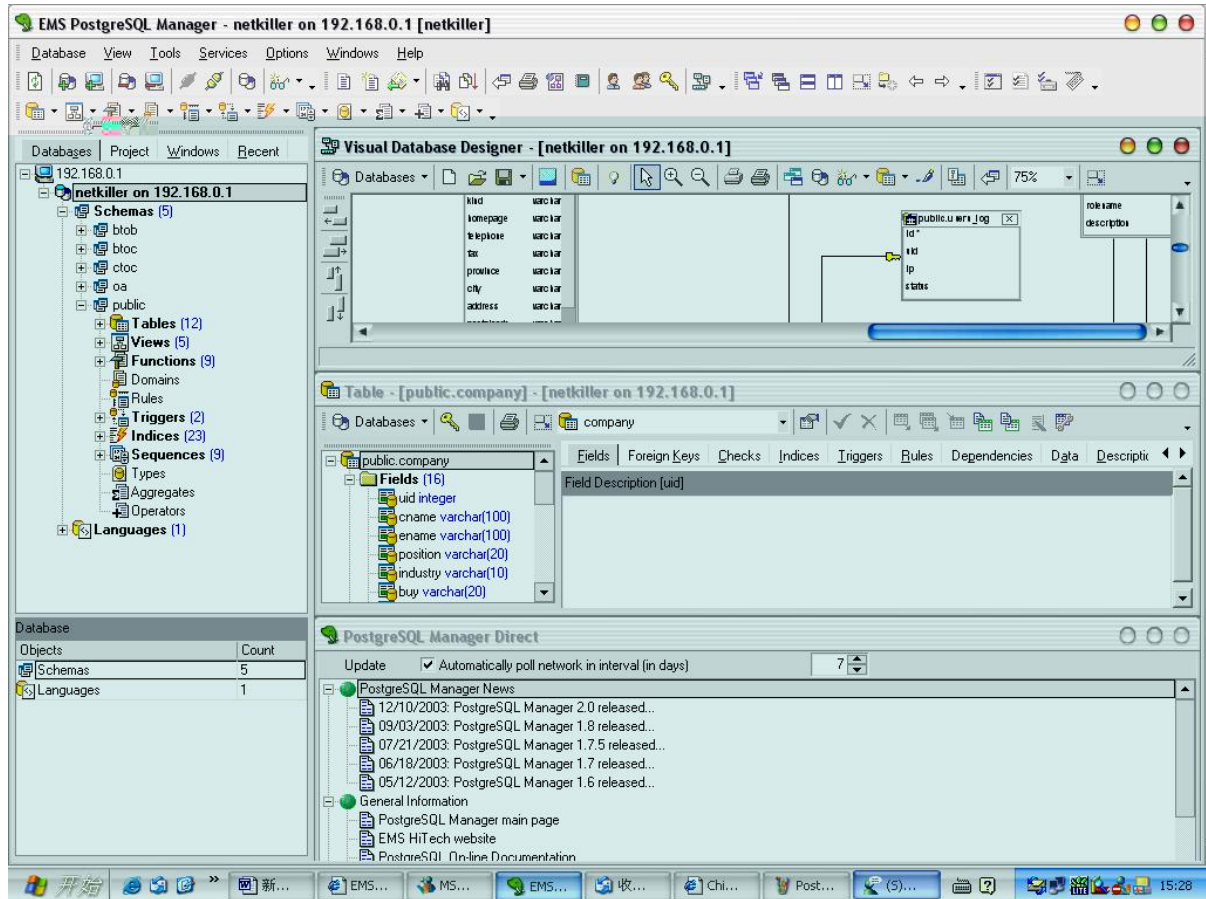
Free email support	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Free updates of CASE Studio 2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

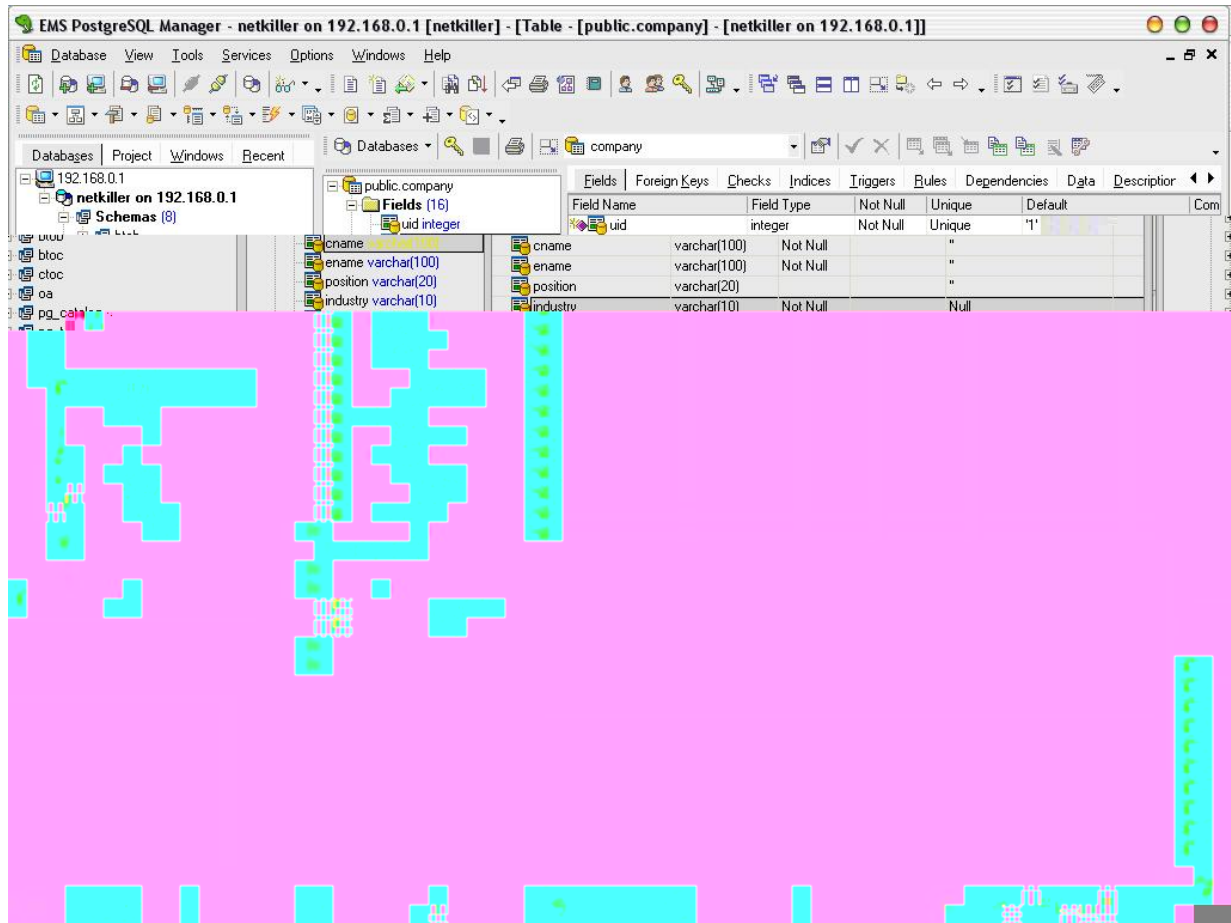
## 15.5.4 PostgreSQL Manager

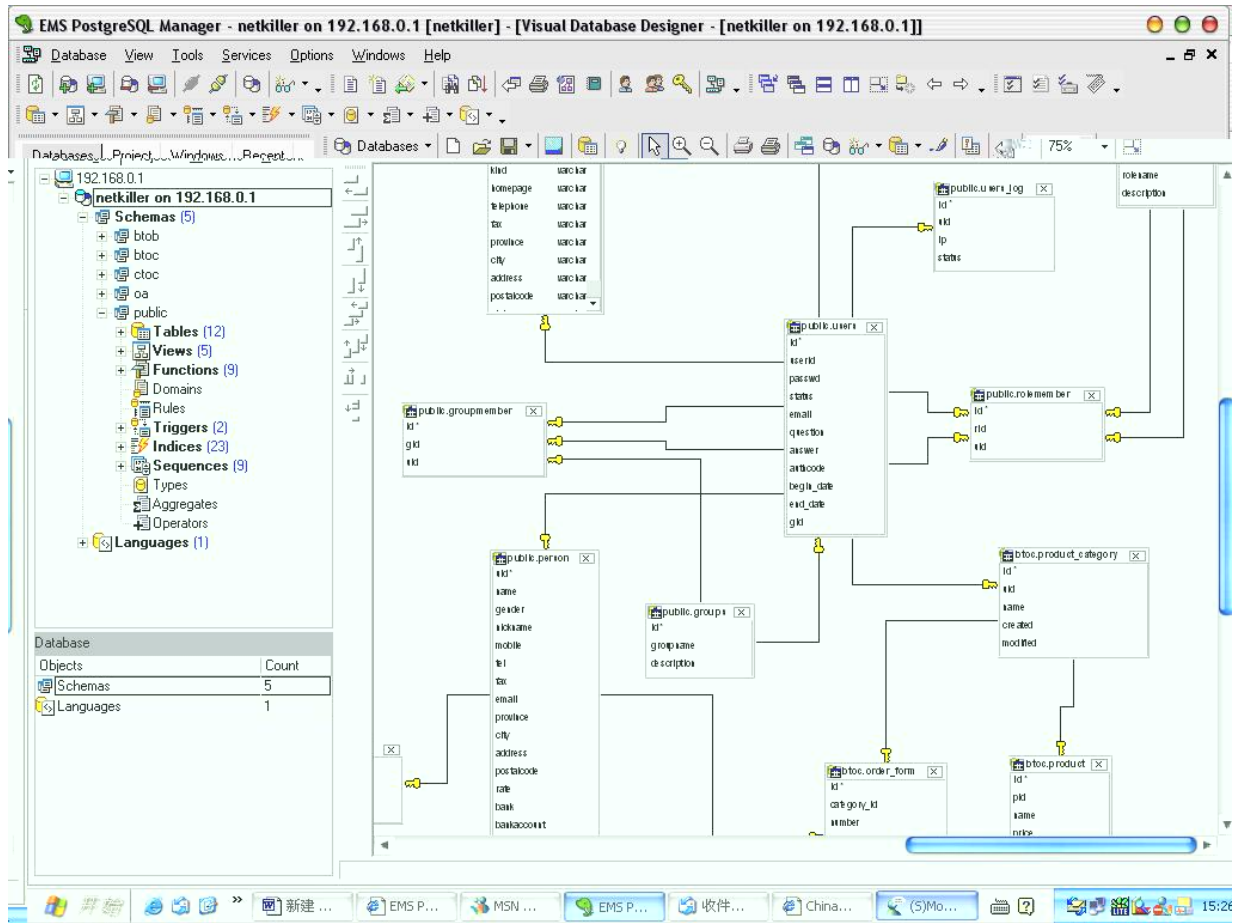


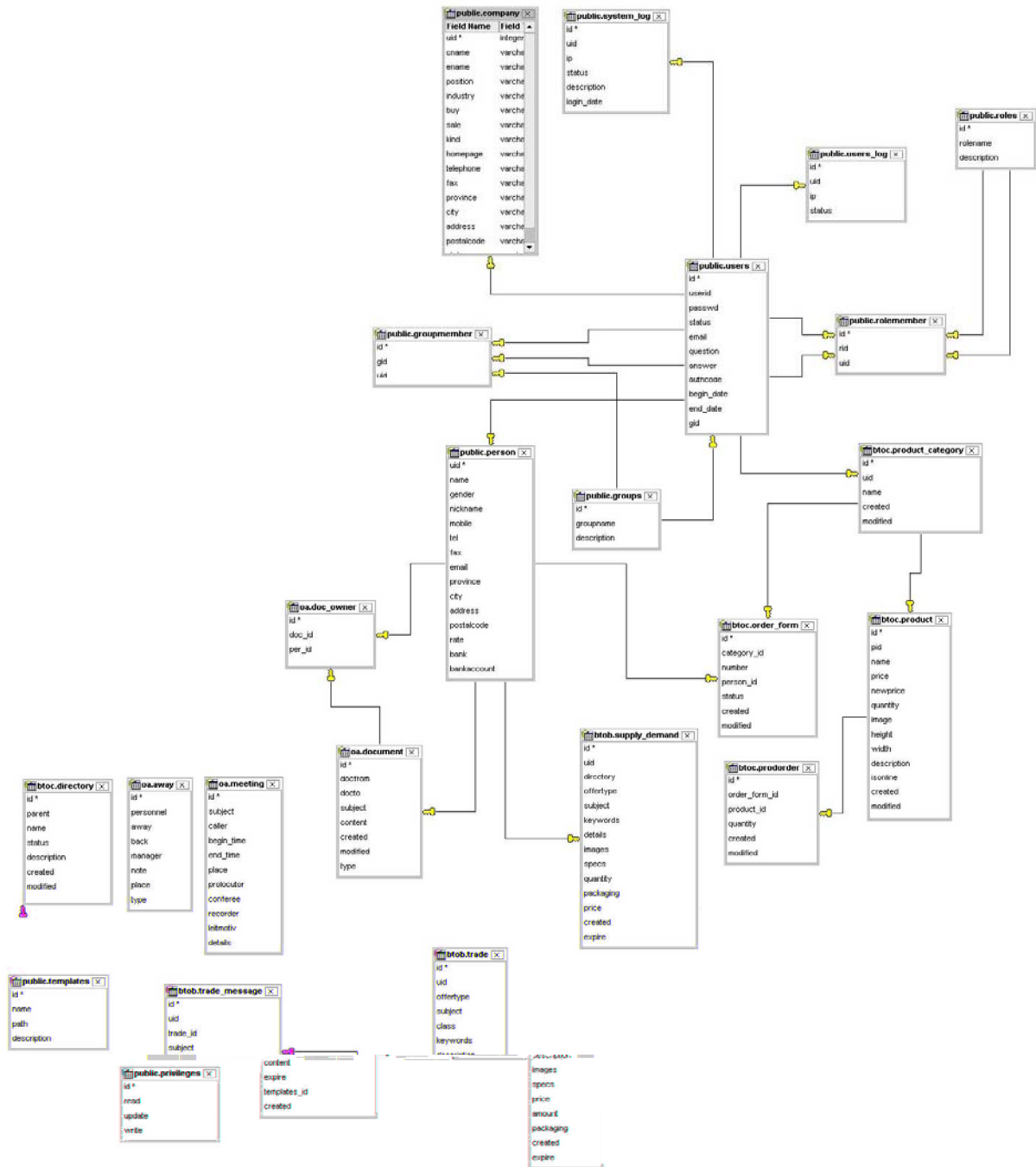
<http://ems-hitech.com/pgmanager/>

<http://ems-hitech.com/pgmanager/download.phtml>









**EMS PostgreSQL Manager** is a powerful graphical tool for PostgreSQL administration and development. It makes creating and editing PostgreSQL database objects easy and fast, and allows you to run SQL scripts, manage users and their privileges, build SQL queries visually, extract, print and search metadata, export data to 14 available formats and import them from most popular formats, view and edit BLOB fields, and many more...

**Version 2.0 Features**

- Full PostgreSQL 7.4 support!
- [Visual Database Designer - to design your database intuitively!](#)
- [Plpgsql Function Debugger - to fix the errors in the function code and to optimize the function work!](#)
- [HTML Report Generator - to generate detailed reports about your database in HTML format!](#)
- Internal SSH Client - to connect to your database using Secure SHell!
- Highly improved [Data Export](#) and [Data Import](#) wizards!
- [And much more...](#)

Find the list of all program features as well as the Professional/Lite and Windows/Linux differences in the [Feature Matrix](#).

#### **Easy management of all PostgreSQL objects:**

- [Creating/Dropping databases;](#)
- [Creating/Dropping/Altering tables, views, functions, sequences](#)
- [Creating/Dropping types, aggregates, operators, languages](#)
- [Creating/Dropping/Altering tables subobjects: foreign keys, checks, indices, triggers](#)
- [Creating/Altering table fields](#)
- [Creating/Dropping/Altering table/view rules](#)
- [Duplicating tables, table fields, views, functions, rules, triggers, sequences, types, aggregates, operators, languages](#)
- [Support of all field types](#)
- [Viewing/Editing data in tables](#)
- [Adding/Editing/Deleting users, groups and their privileges](#)

#### **Powerful tools to work with PostgreSQL server**

- **Database Explorer featuring**
  - [Multiple database connections](#)
  - [Registering/Unregistering databases allows working only with the databases required](#)
  - [Grouping/Ungrouping registered databases by host name](#)
  - [Project View allows combining database object with user items](#)
  - [Powerful User Manager, Group Manager and Grant Manager](#)
- **SQL processing and watching tools, including:**
  - [Powerful SQL Editor](#)
  - [Script Executing \(SQL Script\)](#)
  - [Visual Query Builder](#)
  - [SQL Monitor](#)
  - [Powerful metadata printing module with the opportunity of customizing reports\(fonts, colors, etc.\)](#)
  - [Extracting metadata into text files or SQL Script editor](#)
- **Powerful data manipulation tools, including:**
  - [Powerful BLOB Viewer/Editor with several types of viewing BLOB data:](#)
    - [Export data to Excel, RTF \(Word\), HTML, XML, DBF, TXT, CSV, SYLK, DIF, LaTeX and Windows Clipboard.](#)



- [Import Data Wizard](#)
- [Copy Data to/from file wizards](#)

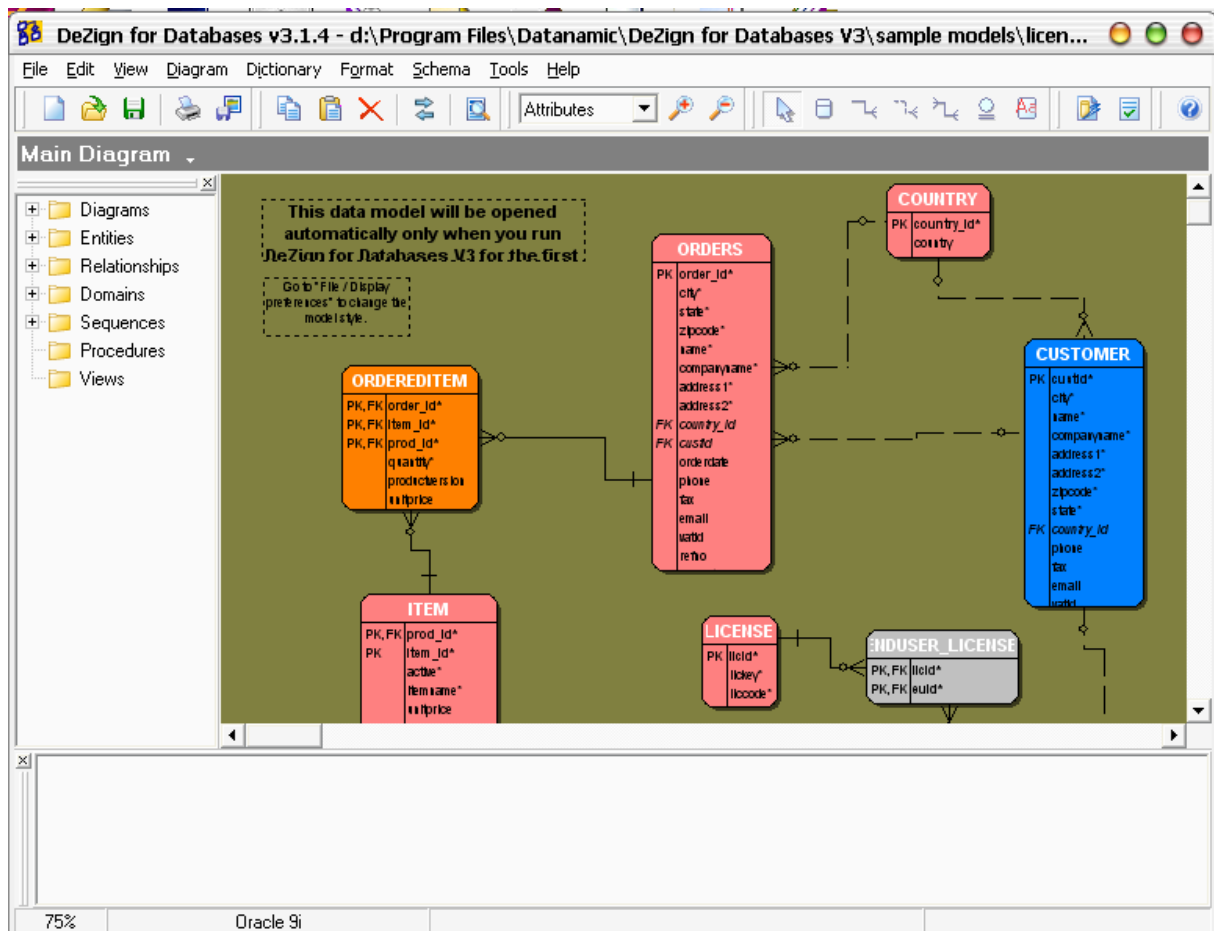
And some other useful features, including:

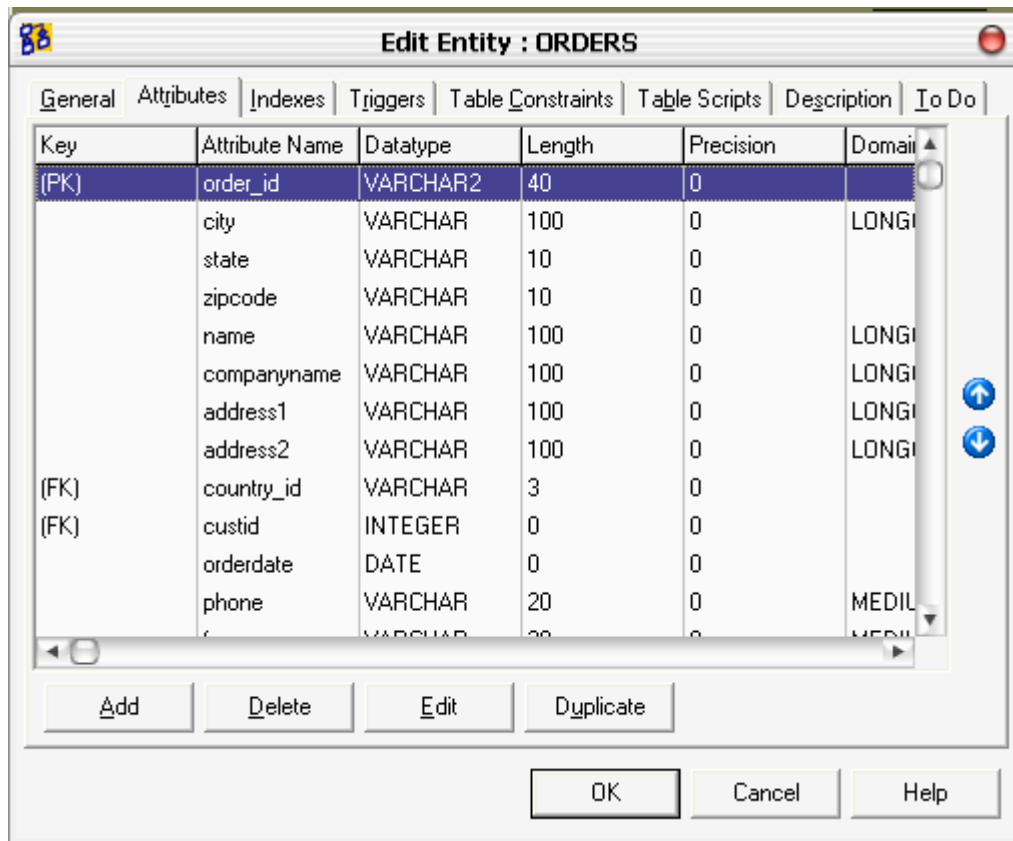
- [Customizable program options: windows, fonts, grid colors, etc](#)
- [Customizable editors: custom colors, fonts etc.](#)
- [Customizable toolbars for all program windows](#)
- [Keyboard templates](#)
- [Powerful Visual Options Module](#)
- [Possibility of saving all the program settings](#)
- [Detailed Help System](#)
- [PostgreSQL Manager Direct](#)
- [And other useful tools, including TO-DO list for each database, External Tool Manager, etc.](#)

## 15.5.5 DeZign for Databases

<http://www.datanamic.com/>

<http://www.datanamic.com/download/index.html>





Feature highlights:

- Entity Relationship Diagram (ERD) editor.
- Template driven schema generation (DDL).
- One project can contain multiple subdiagrams.
- Powerful report generator (HTML, Text, HTML with image map).
- Creation of domains is supported. Usage of domains is supported even for a database that does not support domains.
- Supports the ER modeling techniques including some advanced techniques like sub categories.
- Fast, facile graphics manipulation.
- Automatic resolution of many-to-many relationships.
- Object browser for entities, relationships, attributes and domains.
- Creation of triggers, procedures and sequences is supported.
- Multiple display levels.
- Export diagram to an image file (Bitmap, Gif, JPeg or Windows Metafile).
- Object naming tool.
- Automatic foreign key migration at design time.
- Version control.
- Full cardinality and connectivity support.

## 15.5.6 GUI

phpPgAdmin	▲	✦			✦
PgAdmin III	▲	✦			✦
Case Studio 2			☆	☆	✦ ✦
DeZign for Databases			☆	☆	✦ ✦ ✦
PostgreSQL Manager	▲	✦	☆	☆	✦ ✦ ✦ ✦

PostgreSQL Manager

Oracle DBA Studio MS SQL Server 2000

## 16 FAQ

### 16.1 关于“”与 null

" = \0

.  
 , ???  
 NO, . ""

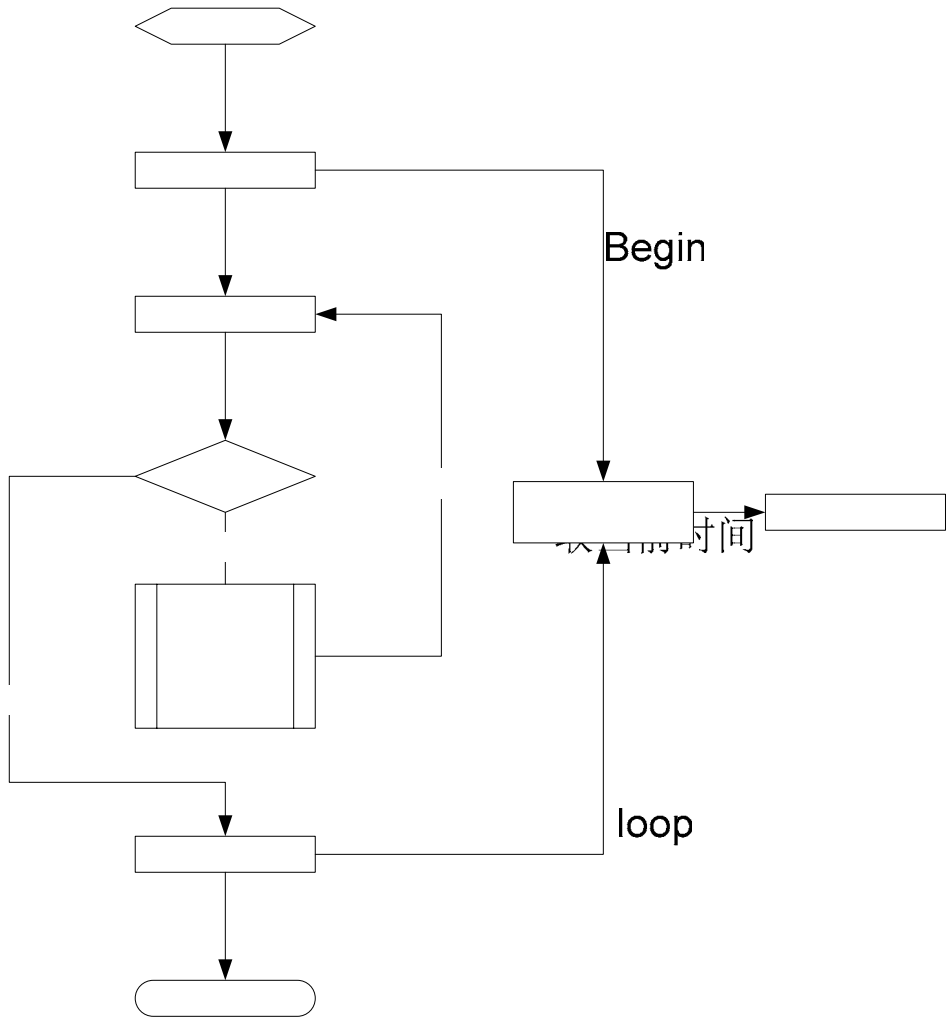
null

### 16.2 Postgresql 与其它数据库

TPC-C TPC-W TPC-H TPC-R  
 PostgreSQL MySQL Oracle , Sybase , MS Sql Server.....  
 MySQL PostgreSQL .....

DBA

API jdbc,odbc,c++ api.....



- 1.
- 2.
- 3.
- 4.
- 5.
- 6.
- 7.

test

$N < 1000$

1000条

开始时

Y

N

Test 表  
 插入数据  
 删除数据  
 修改数据  
 C++ API

win

- 1.
- 2.

100 1000 5000

取当前时间

3.

11	id	PK	10	
11	id	FK	id	10

join

4.

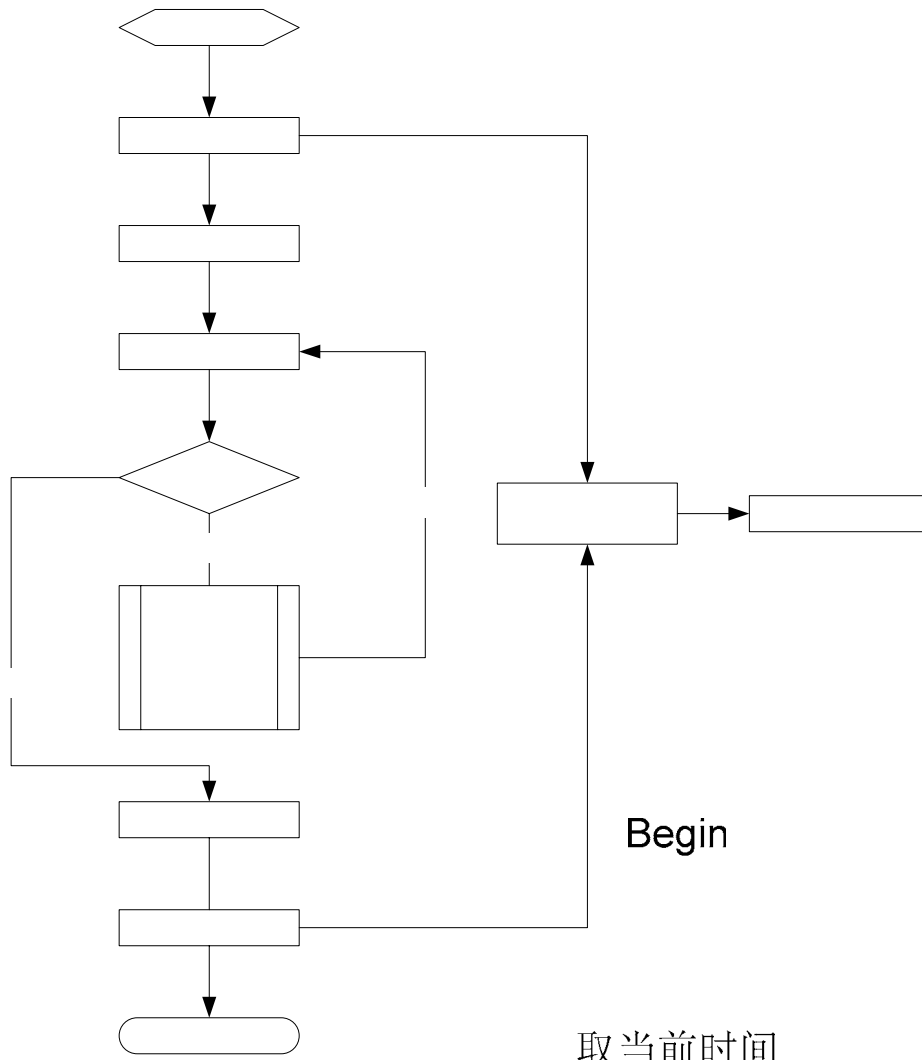
200G.....

5G 10G 50G 100G

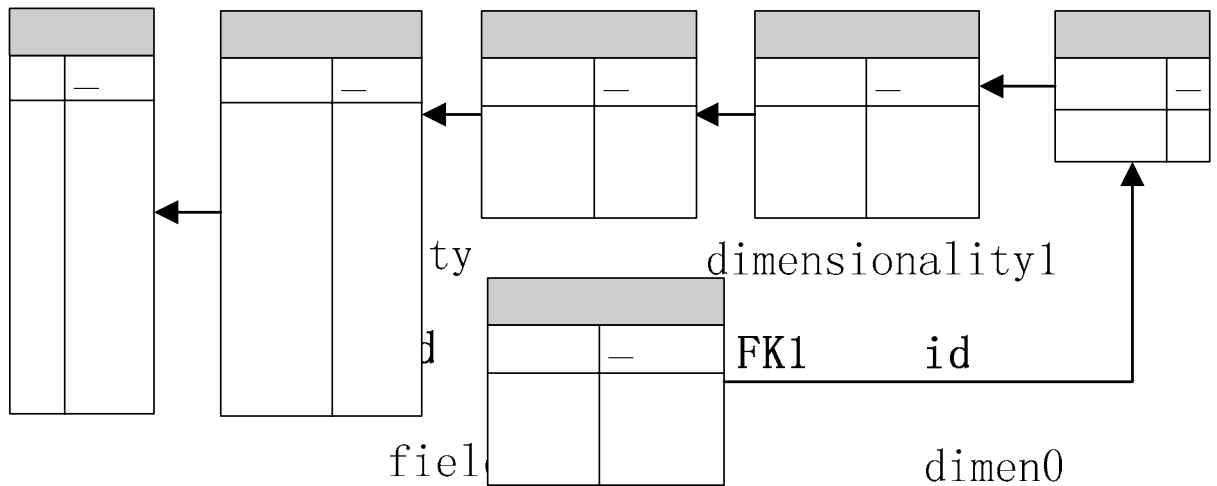
5.

6.

5 10



Begin;



dimen  
FK1

## 16.3 Putty

<http://www.chiark.greenend.org.uk/~sgtatham/putty/>

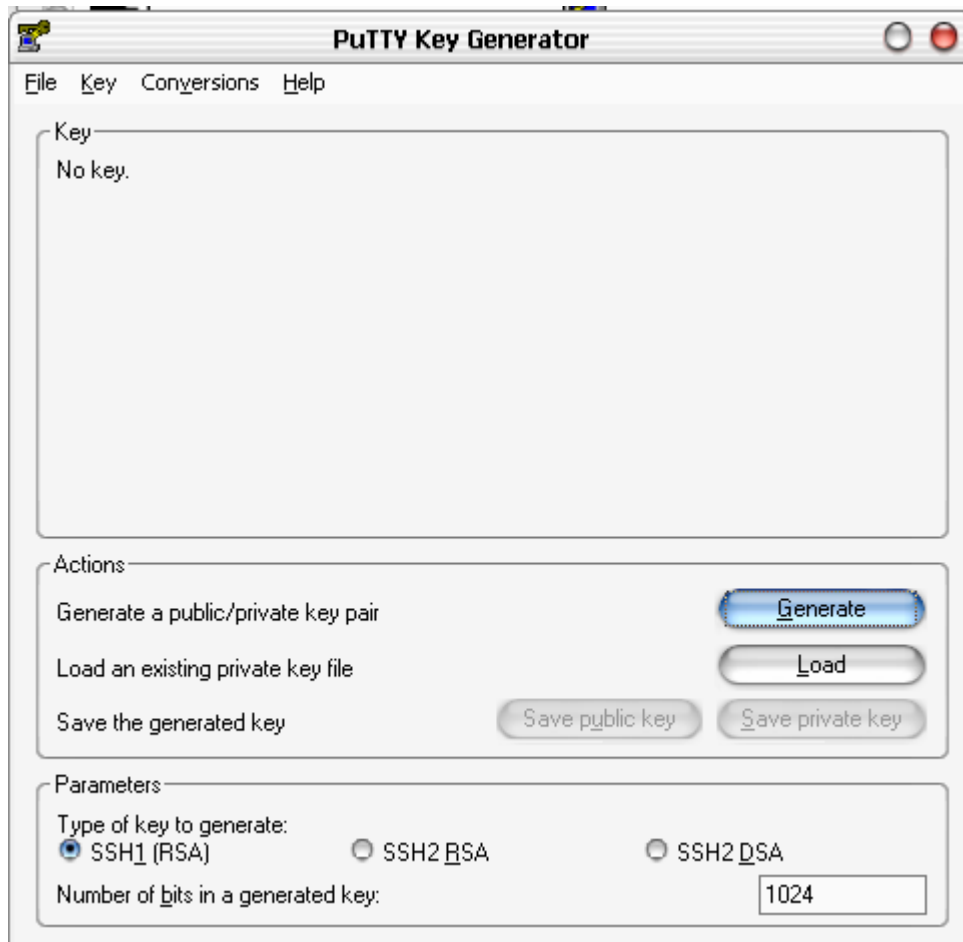


field  
field1  
field2  
field3  
field4  
field5  
dimen0  
dimen1  
dimen2  
dimen3  
dimen4  
dimen5  
dimen6  
dimen7  
dimen8  
dimen9

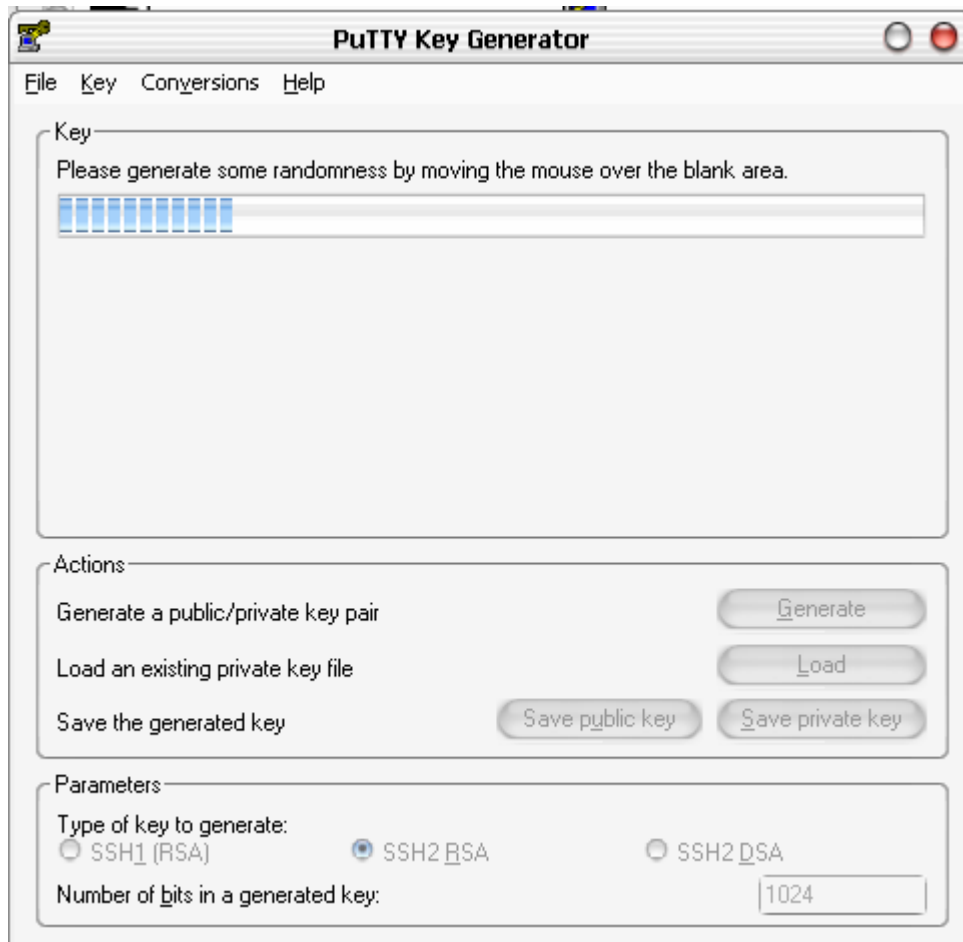
dimen  
FK1

### 16.3.1 Putty

1. puttygen.exe



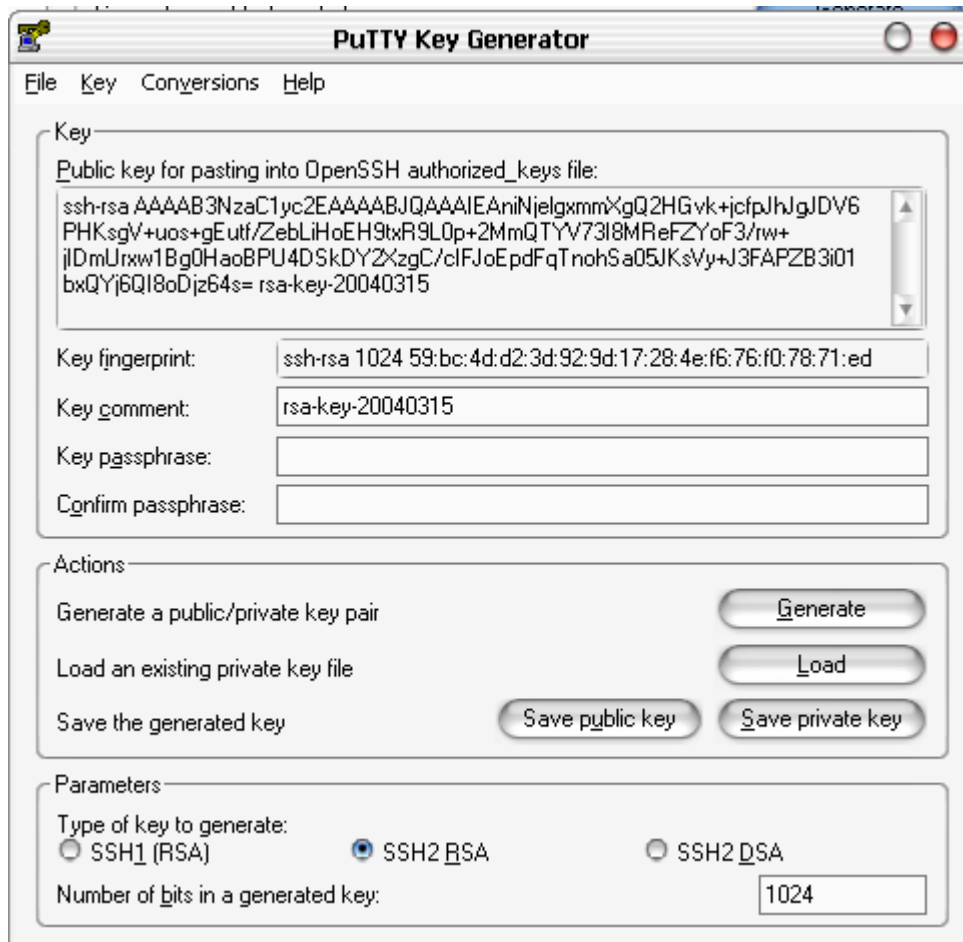
2.       SSH2 RSA       SSH2 DSA
3.       " Generate"



SSH1 RSA

SSH2



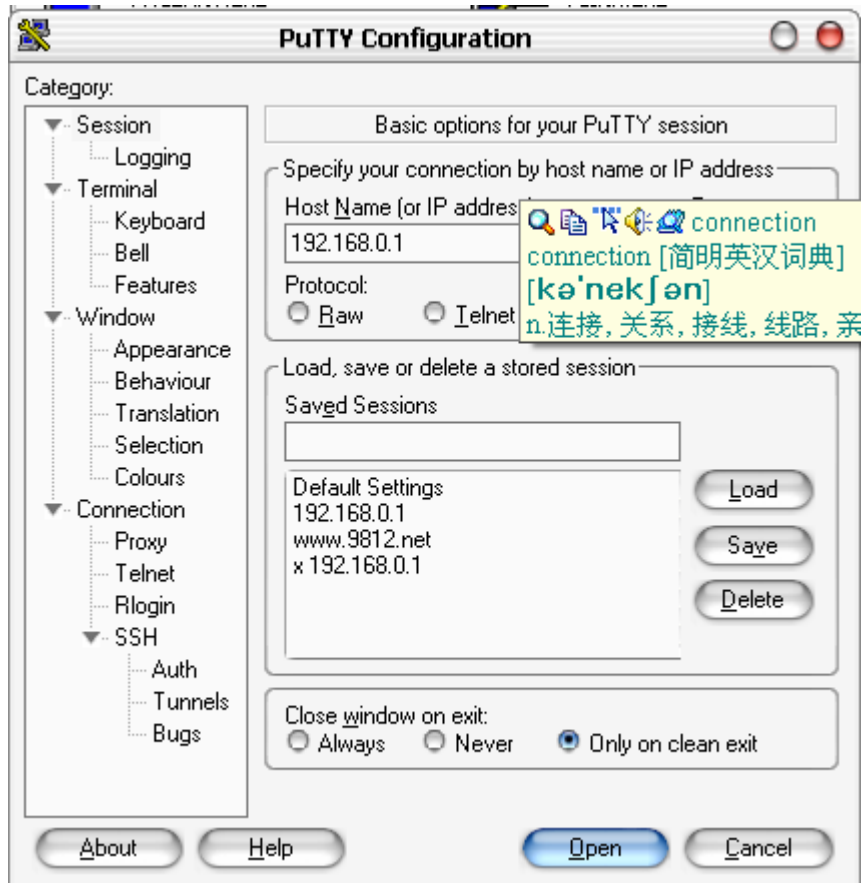


- 4.
5.                   public           private
6.                   linux                            chen
7.   chen        linux

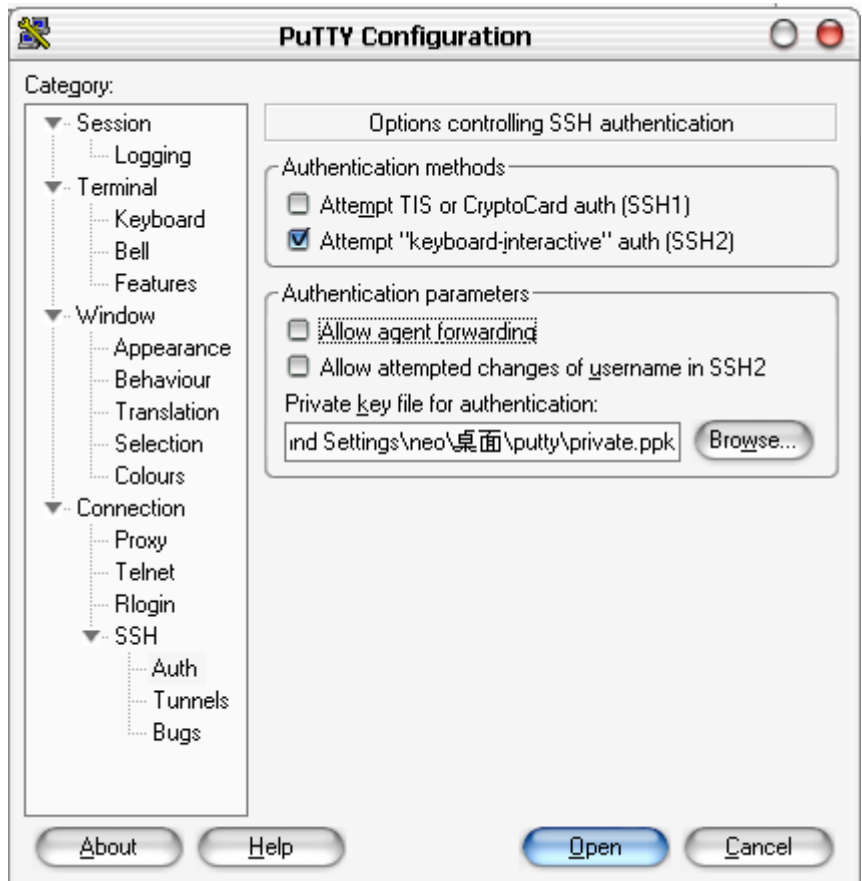
```
[chen@linux chen]$ mkdir .ssh
[chen@linux chen]$ chmod 700 .ssh
[chen@linux chen]$ cat public > .ssh/authorized_keys2
[chen@linux chen]$ chmod 600 .ssh/authorized_keys2
```

Linux

```
putty
Host Name
```



SSH Auth



Private key file for authentication:

" Open"

```
login as: chen
Sent username "chen"
Trying public key authentication.
Passphrase for key "rsa-key-20040315":
[chen@linux chen]$
```

1 Server refused our key

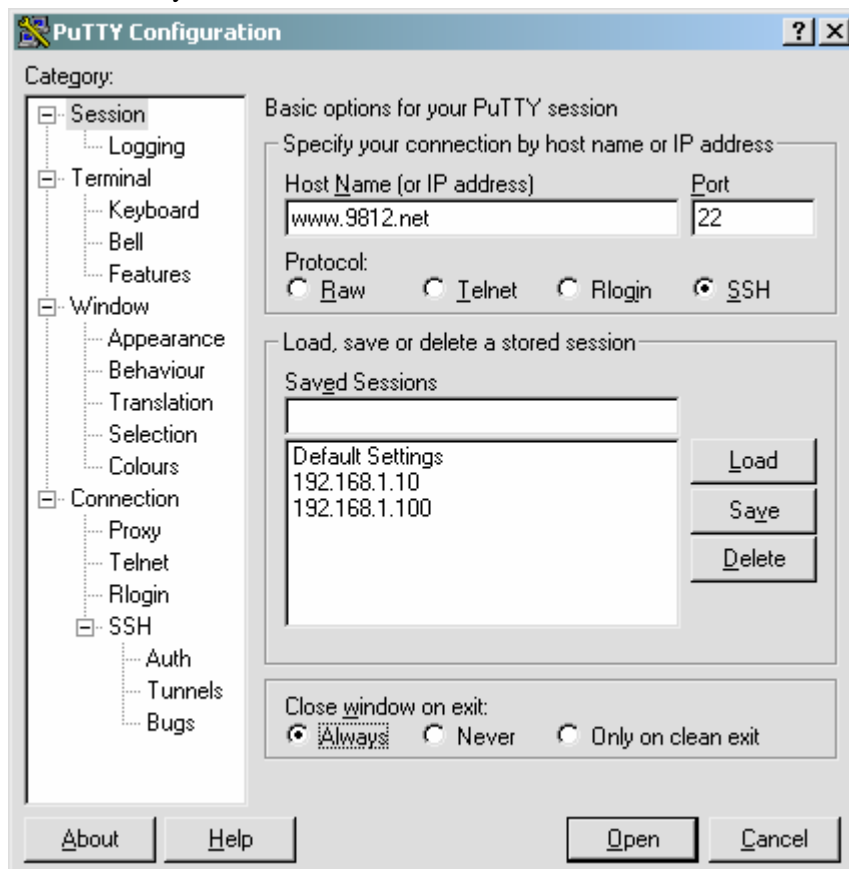
authorized\_keys

2 Unable to use key file (SSH1 private key)

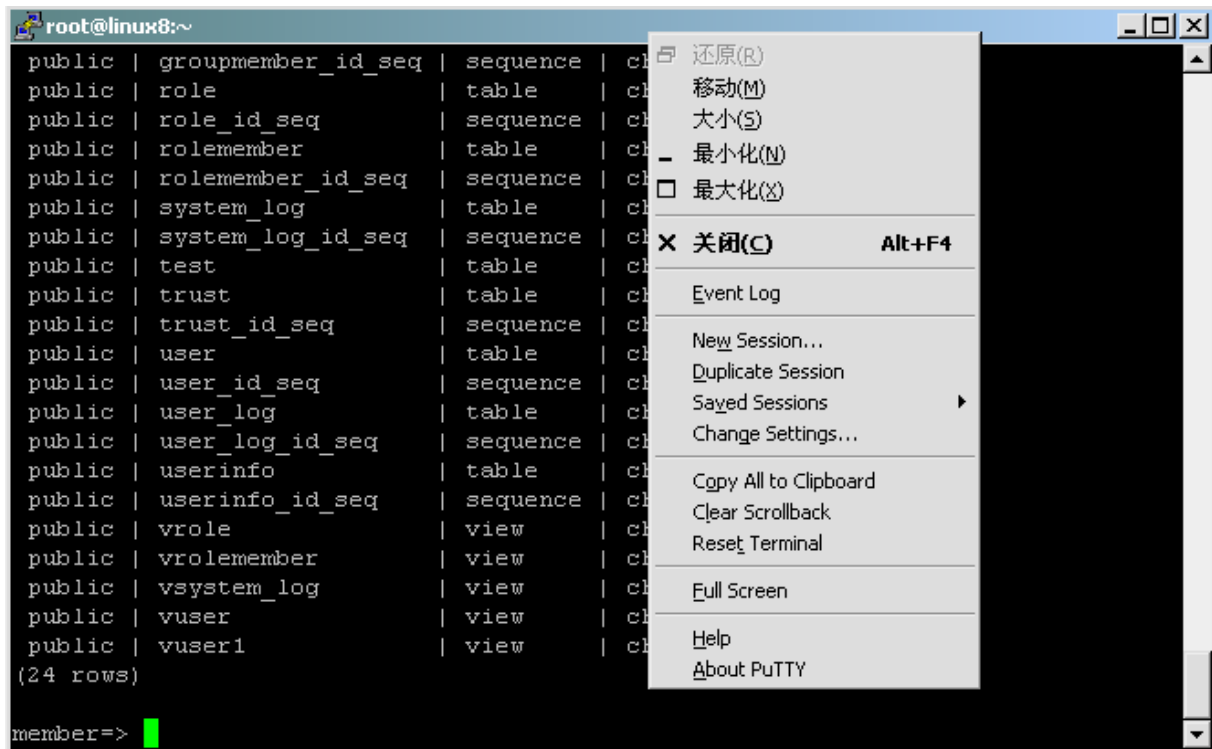
ssh ssh2

## 16.3.2

1. Putty

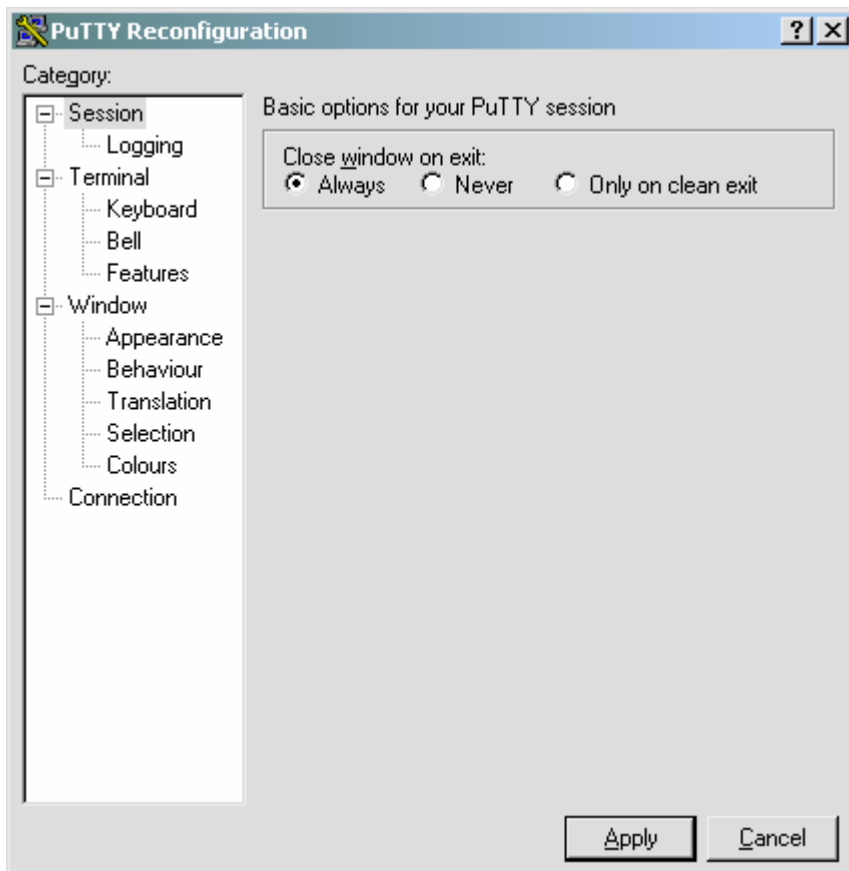


2.



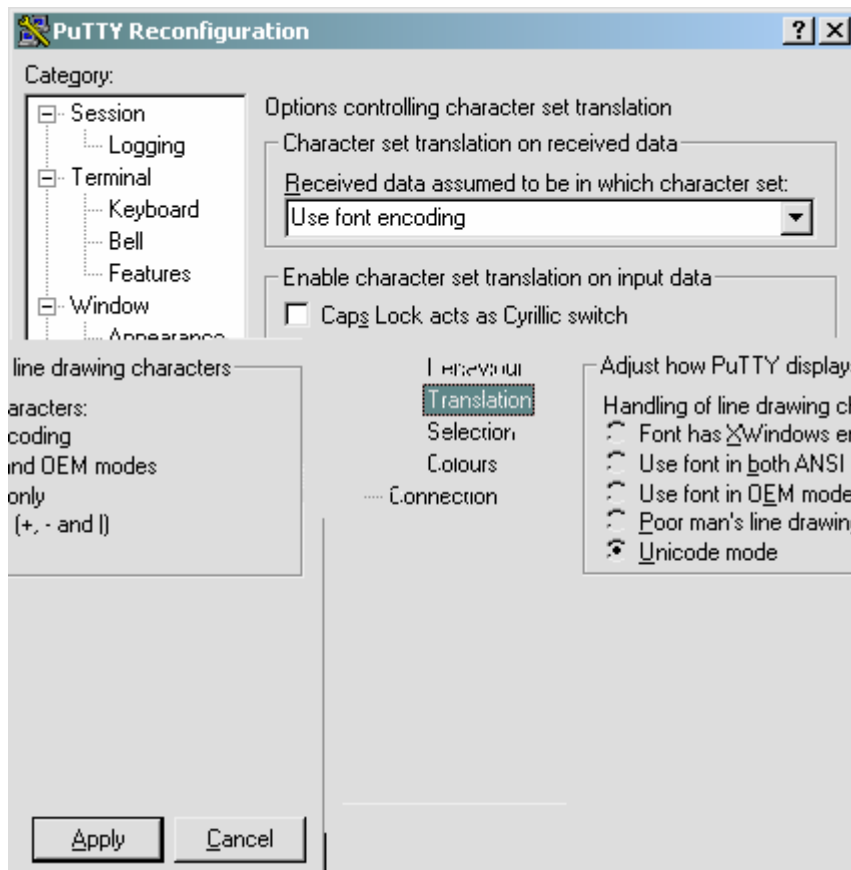
3. " Change Settings..."

4.

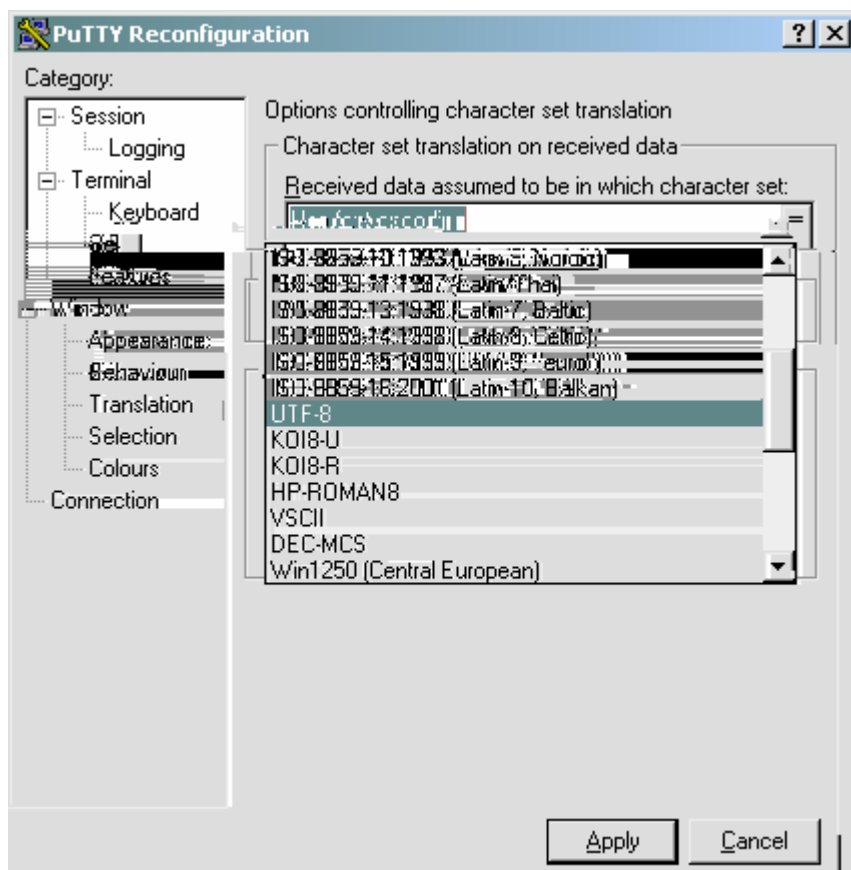


5. Category: Window Translation

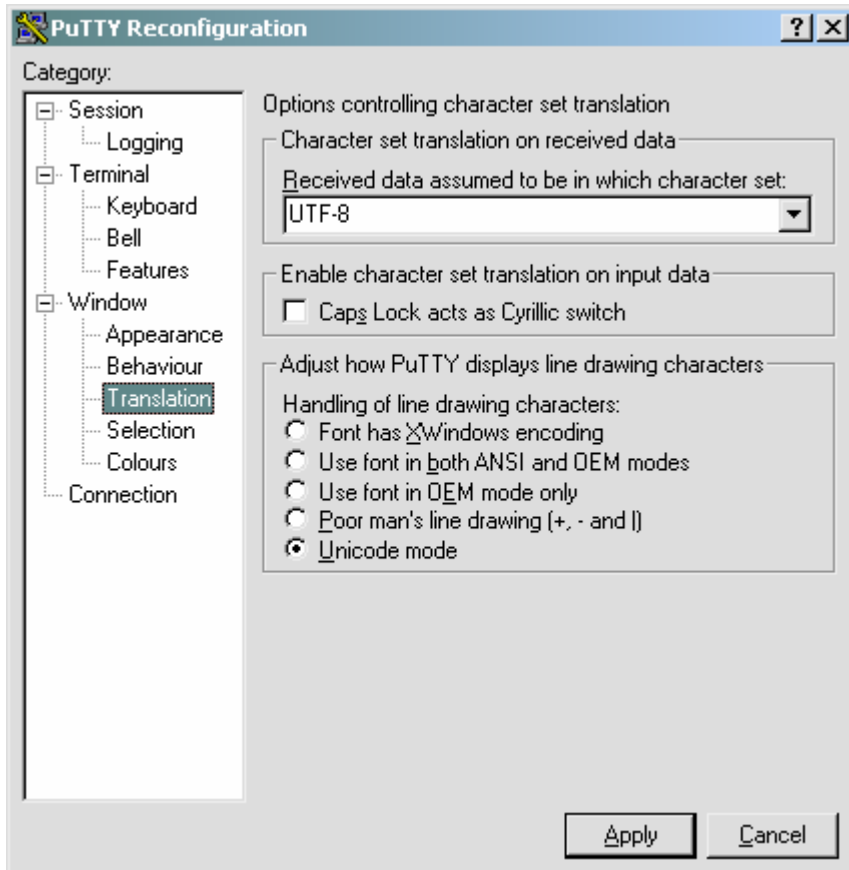
6.



7. Character set translation on received data      Received data assumed to be in which character set:  
       " UTF-8"

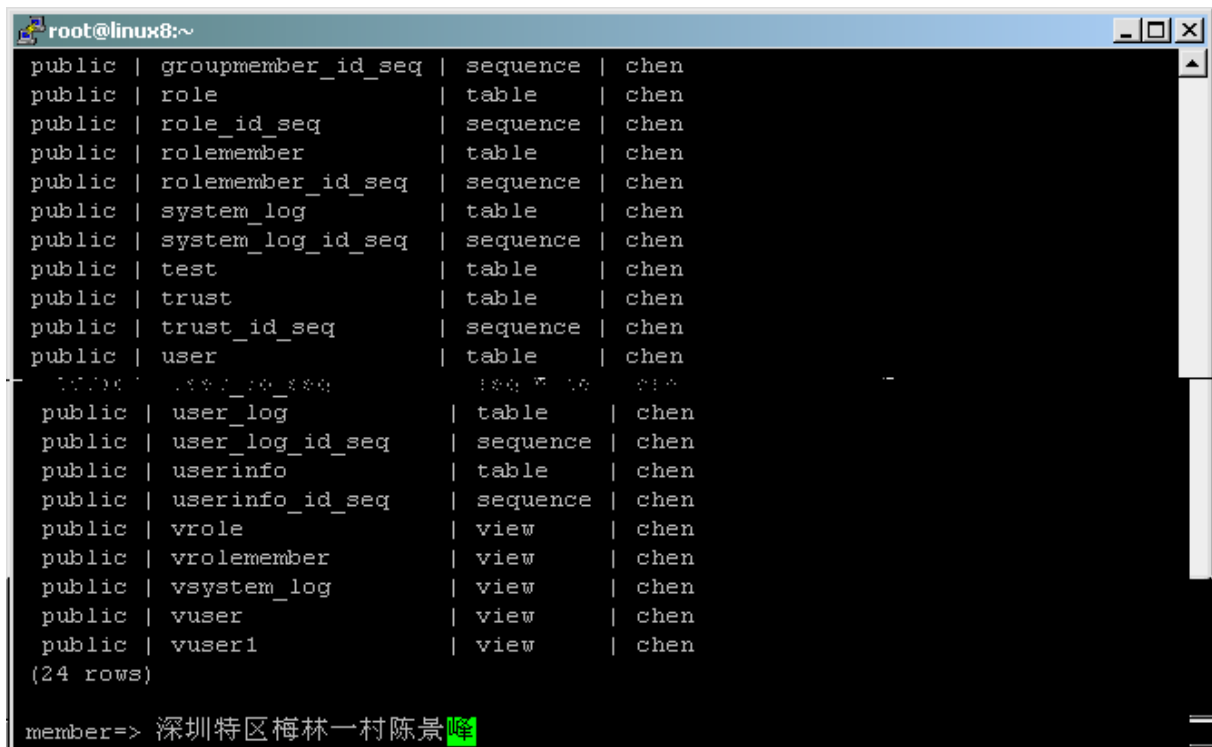


8.



9. " Apply"

10. putty



## 16.4 控制台输入汉字

zhcon                    Linux                    / / (CJK)                    DOS  
UCDOS                    (console)

<http://zhcon.sourceforge.net/>

<http://sourceforge.net/projects/zhcon/>

zhcon

wget <http://keihanna.dl.sourceforge.net/sourceforge/zhcon/zhcon-0.2.3.tar.gz>

[http://zhcon.sourceforge.net/images/scr\\_vim61.png](http://zhcon.sourceforge.net/images/scr_vim61.png)

[http://zhcon.sourceforge.net/images/scr\\_mc.gif](http://zhcon.sourceforge.net/images/scr_mc.gif)

[http://zhcon.sourceforge.net/images/scr\\_lynx.gif](http://zhcon.sourceforge.net/images/scr_lynx.gif)

[http://zhcon.sourceforge.net/images/scr\\_emac.gif](http://zhcon.sourceforge.net/images/scr_emac.gif)

[http://zhcon.sourceforge.net/images/scr\\_bbs.gif](http://zhcon.sourceforge.net/images/scr_bbs.gif)

[http://zhcon.sourceforge.net/images/scr\\_vi.gif](http://zhcon.sourceforge.net/images/scr_vi.gif)

[http://zhcon.sourceforge.net/images/scr\\_overspot.gif](http://zhcon.sourceforge.net/images/scr_overspot.gif)

```
[root@linux zhcon-0.2.3]# ./configure
[root@linux zhcon-0.2.3]# make
[root@linux zhcon-0.2.3]# make install
```

```
[root@linux root]#zhcon
```

## 16.5 PostgreSQL RPM 包安装后，为何没有 5432 端口

```
rpm -Uvh --nodeps postgresql-libs-?.?.?-1PGDG.i386.rpm
rpm -Uvh --nodeps postgresql-?.?.?-1PGDG.i386.rpm
rpm -Uvh --nodeps postgresql-devel-?.?.?-1PGDG.i386.rpm
rpm -Uvh --nodeps postgresql-server-?.?.?-1PGDG.i386.rpm
rpm -Uvh --nodeps postgresql-contrib-?.?.?-1PGDG.i386.rpm
rpm -Uvh --nodeps postgresql-docs-?.?.?-1PGDG.i386.rpm
rpm -Uvh --nodeps postgresql-jdbc-?.?.?-1PGDG.i386.rpm
rpm -Uvh --nodeps postgresql-pl-?.?.?-1PGDG.i386.rpm
rpm -Uvh --nodeps postgresql-python-?.?.?-1PGDG.i386.rpm
rpm -Uvh --nodeps postgresql-tcl-?.?.?-1PGDG.i386.rpm
rpm -Uvh --nodeps postgresql-test-?.?.?-1PGDG.i386.rpm
rpm -qa|grep post
```

```
[root@linux script]# rpm -qa|grep post
postgresql-contrib-7.3.3-1PGDG
postgresql-7.3.3-1PGDG
```

```
postgresql-tcl-7.3.3-1PGDG
postgresql-devel-7.3.3-1PGDG
postgresql-jdbc-7.3.3-1PGDG
postgresql-test-7.3.3-1PGDG
postgresql-server-7.3.3-1PGDG
postgresql-pl-7.3.3-1PGDG
postgresql-libs-7.3.3-1PGDG
postgresql-python-7.3.3-1PGDG
postfix-1.1.11-5
postgresql-docs-7.3.3-1PGDG
```

```
[root@linux root]# service postgresql start
[root@linux root]# su postgres
bash-2.05b$ createdb
bash-2.05b$ psql
Welcome to psql 7.3.3, the PostgreSQL interactive terminal.

Type:  \copyright for distribution terms
        \h for help with SQL commands
        \? for help on internal slash commands
        \g or terminate with semicolon to execute query
        \q to quit

postgres=#
" \q"
```

PostgreSQL	TCP/IP	tcpip_socket	postgresql.conf
------------	--------	--------------	-----------------

```
[root@linux root]# cd /var/lib/pgsql/data/
[root@linux data]# vi postgresql.conf
#
#           Connection Parameters
#

tcpip_socket = true

#ssl = false

#max_connections = 32
#superuser_reserved_connections = 2

#port = 5432
#hostname_lookup = false
#show_source_port = false
:wq ( )
```



```
tcpip_socket = false      tcpip_socket = true      5432
port = 5432
```

```
[root@linux root]# service postgresql restart
[ OK ]
Starting postgresql service:
[ OK ]
```

```
          NAMP      PostgreSQL  tcpip_socket
5432/tcp  open          postgres
```

```
[root@linux root]# nmap localhost

Starting nmap V. 3.00 ( www.insecure.org/nmap/ )
Interesting ports on localhost.localdomain (127.0.0.1):
(The 1592 ports scanned but not shown below are in state: closed)
Port      State      Service
22/tcp    open       ssh
25/tcp    open       smtp
80/tcp    open       http
111/tcp   open       sunrpc
139/tcp   open       netbios-ssn
389/tcp   open       ldap
2401/tcp  open       cvspserver
3306/tcp  open       mysql
5432/tcp  open       postgres

Nmap run completed -- 1 IP address (1 host up) scanned in 2 seconds
[root@linux root]#
```

```
[root@linux root]# psql -h127.0.0.1 -Upostgres
Welcome to psql 7.3.3, the PostgreSQL interactive terminal.

Type: \copyright for distribution terms
       \h for help with SQL commands
       \? for help on internal slash commands
       \g or terminate with semicolon to execute query
       \q to quit

postgres=#
```



[http://www.freebsdchina.org/forum/digest\\_4589.html](http://www.freebsdchina.org/forum/digest_4589.html)

## 16.9 OpenLDAP-PostgreSQL HOWTO

[http://www.samse.fr/GPL/ldap\\_pg/HOWTO/](http://www.samse.fr/GPL/ldap_pg/HOWTO/)

## 16.10 PostgreSQL 成功案例与解决方案

<http://www.isc.org/products/OpenReg/>

OpenReg is an implementation of a domain registry, such as might be used by top-level domain operators to manage the delegation of domains in a "shared registry" environment. OpenReg:

- supports the [Extensible Provisioning Protocol \(EPP\)](#), the IETF standards-track protocol for interaction between registries and registrars;
- is designed and debugged as a distributed multi-process system;
- supports [PostgreSQL](#) and is designed to accommodate to very large registries;
- publishes zone files to be served using [BIND](#);
- gathers comprehensive profiling and load statistics;
- is published as free software, under a BSD-style licence.

# 17

## 17.1 实例

```
/*
#####
Created      2003-7-21
Modified    2003-7-22
Project      XXXXXXXXXXXXXXXX
Model       XXXXXXXXXXXXXXXX
Company     XXXXXXXXXX
Author
```

Version 1.0

Database PostgreSQL 7.3.3

#####

\*/

-- =====

```

"passwd" Varchar(50),
"name" Varchar(20)NOT NULL ,
"nickname" Varchar(20)NOT NULL ,
"active" Boolean Default 'F',
"email" Varchar(50) NOT NULL,
"question" Varchar(255) NOT NULL,
"answer" Varchar(255) NOT NULL,
"begin_date" Timestamp Default now(),
"end_date" Timestamp Default now(),
UNIQUE (userid,email),
primary key ("id")
);
Create index "user_index" on "user" using btree ("id","userid");

-----
-- 'vuser'
-----

drop view vuser;
CREATE VIEW vuser AS
SELECT u.id,u.userid,u."name",u.nickname,
CASE WHEN u.active=true THEN 'Y' ELSE 'N' END as "active",
u.email,u.question,u.answer,
to_char(u.begin_date,'YYYY-MM-DD HH:MI:SS') as begin_date,
to_char(u.end_date,'YYYY-MM-DD HH:MI:SS') as end_date
FROM "user" u
Order By u.id;

-- DROP FUNCTION user_tri_func() CASCADE ;
CREATE OR REPLACE FUNCTION user_tri_func () RETURNS TRIGGER AS '
DECLARE
-- old_id ALIAS FOR OLD.id;
-- new_id CONSTANT INTEGER := New.id;
BEGIN
IF TG_OP = "DELETE" THEN
Delete from groupmember where uid = OLD.id;
Delete from rolemember where uid = OLD.id;
Delete from userinfo where uid = OLD.id;
Delete from trust where uid = OLD.id;
Delete from system_log where uid = OLD.id;
Delete from user_log where uid = OLD.id;
END IF;
IF TG_OP = "INSERT" THEN
-- INSERT INTO company(uid) values(NEW.id);
END IF;

```

```

        RETURN OLD;
    END;
' LANGUAGE 'plpgsql';
-- DROP TRIGGER user_delete_tri on user;
-- DROP TRIGGER user_insert_tri on user;
CREATE TRIGGER users_delete_tri
    BEFORE Delete ON "user" FOR EACH ROW
    EXECUTE PROCEDURE user_tri_func ();

CREATE TRIGGER users_insert_tri
    AFTER INSERT ON "user" FOR EACH ROW
    EXECUTE PROCEDURE user_tri_func ();
/*
=====
-- 'privileges'
=====
--Drop table "privileges" Restrict;
Create table "privileges"
(
    "id" integer NOT NULL UNIQUE ,
    "read" integer,
    "update" integer,
    "write" integer,
    primary key ("id")
);
*/
=====
-- 'group'
=====
Create table "group"
(
    "id" Serial NOT NULL UNIQUE,
    "groupname" Varchar(20) NOT NULL,
    "description" Varchar(255),
    UNIQUE (groupname),
    PRIMARY KEY ("id")
);
Create index "group_index" on "group" using btree ("id", "groupname");
=====
-- 'groupmember'
=====
Create table "groupmember"
(
    "id" Serial NOT NULL UNIQUE,

```

```

        "gid" integer NOT NULL Default 0,
        "uid" integer NOT NULL Default 0,
        UNIQUE (gid,uid),
        primary key ("id")
    );
-----
-- 'vgroupmember'
-----
-- DROP VIEW vgroupmember;
CREATE VIEW vgroupmember AS
    SELECT gm.id,gm.gid,g.groupname,gm.uid,u.userid,u.name
        FROM "group" g,"user" u,groupmember gm
        Where u.id = gm.uid and g.id = gm.gid
    ORDER BY gm.id;
/*
-----
-- 'vgroup'
-----
-- PHP Interface View
CREATE OR REPLACE VIEW vgroup AS
    SELECT gm.id,gm.uid,gm.gid,g.groupname
    FROM "group" g,groupmember gm
    Where g.id = gm.gid
    ORDER BY gm.id;
*/
-----
-- 'RULE'
-----
CREATE RULE group_rule AS ON Delete TO "group"
    DO Delete From groupmember where gid = OLD.id;

-----
-- 'role'
-----
-- drop table role CASCADE;
Create table "role"
(
    "id" Serial NOT NULL UNIQUE,
    "rolename" Varchar(20) NOT NULL,
    "description" Varchar(255),
    UNIQUE (rolename),
    PRIMARY KEY ("id")
);
Create index "role_index" on "role" using btree ("id","rolename");

```

```

-- =====
-- 'rolemember'
-- =====

-- drop table rolemember CASCADE ;
Create table "rolemember"
(
    "id" Serial NOT NULL UNIQUE,
    "rid" integer NOT NULL Default 0,
    "uid" integer NOT NULL Default 0,
    UNIQUE (rid,uid),
    primary key ("id")
);
-----

-- 'vrolemember'
-----

CREATE VIEW vrolemember AS
    SELECT rm.id,rm.rid,r.rolename,rm.uid,u.userid,u.name
        FROM "role" r,"user" u,rolemember rm
        Where u.id = rm.uid and r.id = rm.rid
        ORDER BY rm.id;

/*
-----

-- 'vrole'
-----

CREATE OR REPLACE VIEW vrole AS
    SELECT rm.id,rm.uid,rm.rid,r.rolename
        FROM "user" u,role r,rolemember rm
        Where u.id = rm.uid and r.id = rm.rid
        ORDER BY rm.id;
*/

-----

-- 'RULE'
-----

CREATE RULE role_rule AS ON Delete TO role
    DO Delete From rolemember where rid = OLD.id;

-- =====
-- 'Foreign Key'
-- =====

Alter table "groupmember" add foreign key ("uid") references "user" ("id") on update restrict on delete
restrict;
Alter table "groupmember" add foreign key ("gid") references "group" ("id") on update restrict on delete

```



```
restrict;
Alter table "rolemember" add foreign key ("uid") references "user" ("id") on update restrict on delete
restrict;
Alter table "rolemember" add foreign key ("rid") references "role" ("id") on update restrict on delete
restrict;
```

```
-- =====
-- 'trust'
-- =====

Create table "trust"
(
    "id" Serial NOT NULL UNIQUE,
    "uid" integer NOT NULL Default 0,
    "rate" Varchar(20) Default '0' Check (rate in ('0','1','2','3','4','5')),
    primary key ("uid")
);
Alter table "trust" add foreign key ("uid") references "user" ("id") on update restrict on delete restrict;
```

```
-- =====
-- 'userinfo'
-- =====

-- Drop table userinfo CASCADE;
Create table "userinfo"
(
    id Serial NOT NULL UNIQUE,
    uid integer NOT NULL Default 0,
    tel Varchar (20) NOT NULL,
    fax varchar(20),
    email varchar(60),
    province varchar(8),
    city varchar(10),
    address varchar(255) DEFAULT " NOT NULL,
    postalcode varchar(6) DEFAULT " NOT NULL, -- post office code
    bank varchar(20) DEFAULT " NOT NULL, -- bank
    bankaccount varchar(20) DEFAULT " NOT NULL, -- Back Account
    "rate" Varchar(20) Default '0' Check (rate in ('0','1','2','3','4','5')),
    PRIMARY KEY(id),
    FOREIGN KEY (uid) REFERENCES "user" (id)
);
```

```
-- =====
-- 'log system'
-- =====

Create table user_log
```

```

(
    id Serial NOT NULL UNIQUE,
    uid integer NOT NULL Default 0,
    ip inet,
    status varchar(255),
    PRIMARY KEY("id"),
    FOREIGN KEY (uid) REFERENCES "user" (id)
);
-- drop table system_log CASCADE;
Create table system_log
(
    id Serial NOT NULL UNIQUE,
    uid integer NOT NULL Default 0,
    ip inet ,
    status varchar(255),
    description varchar(255),
    login_date Timestamp Default now(),
    PRIMARY KEY("id"),
    FOREIGN KEY (uid) REFERENCES "user" (id)
);
-----
-- 'system_log'
-----

drop view vsystem_log ;
CREATE OR REPLACE VIEW vsystem_log AS
    SELECT u.userid,u.name,log.ip,log.status,log.description,
           to_char(log.login_date,'YYYY-MM-DD HH:MI:SS') as login_date
    FROM "user" u,system_log log
    Where log.uid = u.id
    ORDER BY log.id;
-----
-- 'Function'
-----
-- DROP FUNCTION add_system_log(integer,inet,varchar);
CREATE OR REPLACE FUNCTION add_system_log(integer,inet,varchar,varchar) RETURNS boolean
AS '
    DECLARE
        vUID ALIAS FOR $1;
        vIP ALIAS FOR $2;
        vSTATUS ALIAS FOR $3;
        vDESC ALIAS FOR $4;
    BEGIN
        insert into system_log(uid,ip,status,description) values(vUID,vIP,vSTATUS,vDESC);
        RETURN true;

```

```

END;
' LANGUAGE 'plpgsql';
select add_system_log(1,'127.0.0.1','Create Database','Initialization Database');

-----
-- 'Insert Data'
-----
insert          into          "user"(userid,passwd,name,nickname,email,question,answer)
values('sysop','chen','chen','chen','chen@chen.com','xxxxxxx','xxxx');
insert          into          "user"(userid,passwd,name,nickname,email,question,answer)
values('admin','chen','chen','chen','chen@chen.com','xxxxxxx','xxxx');
insert into "user"(userid,passwd,name,nickname,email,question,answer) values('netkiller','chen','
','Netkiller','chen@chen.com','xxxxxxx','xxxx');
update "user" set active='true' where userid='sysop';
-----
-- 'Insert Data'
-----
Insert into "group"(groupname,description) values('System',' ');
Insert into "group"(groupname,description) values('Administrator',' ');
Insert into "group"(groupname,description) values('gold','gold diamond');
Insert into "group"(groupname,description) values('silver','silver diamond');
Insert into "group"(groupname,description) values('advance',' ');
Insert into "group"(groupname,description) values('free',' ');
-----
-- 'Insert Data'
-----
Insert into "role"(rolename,description) values('System',' ');
Insert into "role"(rolename,description) values('Administrator',' ');
Insert into "role"(rolename,description) values('gold','gold diamond');
Insert into "role"(rolename,description) values('silver','silver diamond');
Insert into "role"(rolename,description) values('advance',' ');
Insert into "role"(rolename,description) values('free',' ');

insert into groupmember(gid,uid) values((select id from "group" where groupname ='System'),(select id
from vuser where userid='sysop'));
insert into rolemember(rid,uid) values((select id from role where rolename ='System'),(select id from vuser
where userid='sysop'));

```

## 17.2 实例

```

--      : localhost      : XXXX
-- Create Database XXXX;

```

```

-- DROP DATABASE XXXX;
-- createdb -E EUC_CN XXXX
-- createlang plpgsql XXXX
-- CREATE USER XXXX WITH PASSWORD "XXXXXX";

-----
-- 'siteuser'
-----

--DROP TABLE IF EXISTS siteuser;
DROP TABLE siteuser CASCADE;
DROP SEQUENCE siteuser_id_seq;
DROP INDEX siteuser_id_index;

CREATE TABLE siteuser (
    id integer DEFAULT nextval('siteuser_id_seq') NOT NULL,
    username varchar(20) DEFAULT '0' NOT NULL,
    Password varchar(50) DEFAULT '0' NOT NULL,
    realname varchar(10) DEFAULT '0' NOT NULL,
    email varchar(50) DEFAULT '0' NOT NULL,
    create_date timestamp DEFAULT now(),
    modify_date timestamp DEFAULT now(),
    UNIQUE (id,username),
    PRIMARY KEY (id)
);
CREATE SEQUENCE siteuser_id_seq;
CREATE INDEX siteuser_id_index ON siteuser (id);

DROP FUNCTION siteuser_tri_func() CASCADE ;
CREATE OR REPLACE FUNCTION siteuser_tri_func () RETURNS opaque AS '
-- DECLARE
-- user_id CONSTANT INTEGER := OLD.id;
BEGIN
    IF TG_OP = "DELETE" THEN
        Delete from company where uid = OLD.id;
        Delete from link where uid = OLD.id;
        Delete from product_sort where uid = OLD.id;
        Delete from news where uid = OLD.id;
        Delete from count where uid = OLD.id;
        Delete from guestbook where uid = OLD.id;
        Delete from clientinfo where uid = OLD.id;
        Delete from column_bar where uid = OLD.id;
        Delete from drumbeating where uid = OLD.id;
    END IF;
    IF TG_OP = "INSERT" THEN

```

```

        INSERT INTO company(uid) values(NEW.id);
        INSERT          INTO          count(uid,number,fontcolor,backgroundcolor)
values(NEW.id,0,"fffff","000000");
        INSERT          INTO          drumbeating(uid,logourl,bannerurl)
values(NEW.id,"default_logo.jpg","default_banner.jpg");
        END IF;
        RETURN OLD;
        END;
' LANGUAGE 'plpgsql';

```

```

DROP TRIGGER siteuser_delete_tri on siteuser;
CREATE TRIGGER siteuser_delete_tri
        BEFORE Delete ON siteuser FOR EACH ROW
        EXECUTE PROCEDURE siteuser_tri_func ();

```

```

DROP TRIGGER siteuser_insert_tri on siteuser;
CREATE TRIGGER siteuser_insert_tri
        AFTER INSERT ON siteuser FOR EACH ROW
        EXECUTE PROCEDURE siteuser_tri_func ();

```

```

DROP FUNCTION adduser(varchar,varchar);
CREATE OR REPLACE FUNCTION adduser(varchar,varchar) RETURNS boolean AS '
        DECLARE
                bool boolean := false;
                name text;
                uid integer;
                user ALIAS FOR $1;
                pass ALIAS FOR $2;
                su siteuser%ROWTYPE;
        BEGIN
                SELECT INTO name username FROM siteuser WHERE username = user;
                IF NOT FOUND then
                        insert into siteuser(username,password) values(user,pass);
                        SELECT INTO uid id FROM siteuser WHERE username = user;
                        bool := true;
                ELSE
                        bool := false;
                        RAISE NOTICE "Calling adduser() return %",bool;
                END IF;
                RETURN bool;
        END;
' LANGUAGE 'plpgsql';
--select adduser('ccsc','eee');

```

```

--select * from siteuser ;
-- record
DROP FUNCTION deluser(integer);
CREATE OR REPLACE FUNCTION deluser(integer) RETURNS boolean AS '
    DECLARE
        bool boolean := false;
        userid text;
    BEGIN
        SELECT INTO userid id FROM siteuser WHERE id = $1;
        IF FOUND then
            delete from siteuser where id = $1;
            bool := true;
        ELSE
            END IF;
        RETURN bool;
    END;
' LANGUAGE 'plpgsql';
--select deluser(28);

```

```

DROP VIEW vsiteuser;
CREATE VIEW vsiteuser AS
    SELECT su.id,su.username,
           to_char(su.create_date,'YYYY-MM-DD HH:MI:SS') as date
    FROM siteuser su
    ORDER BY su.id;

```

```

-----
-- 'style'
-----

```

```

DROP TABLE style;
DROP SEQUENCE style_id_seq;
DROP INDEX style_id_index;
DROP VIEW vstyle;

```

```

CREATE TABLE style (
    id          integer DEFAULT nextval('style_id_seq') NOT NULL,
--   uid       integer  DEFAULT '1' NOT NULL, -- with foreign key
    number     varchar(20) DEFAULT " NOT NULL, -- sytle number
    stylename  varchar(20) DEFAULT " NOT NULL,
    miniature  varchar(50) DEFAULT " NOT NULL, -- mini image
    url        varchar(50) DEFAULT " NOT NULL, -- link url eg.
/usr/local/apache/htdocs/autosite/style/n0001
    description text    DEFAULT " NOT NULL,

```

```

        create_date timestamp DEFAULT now() ,
        modify_date timestamp DEFAULT now() ,
        PRIMARY KEY (id),
        UNIQUE (id)
-- FOREIGN KEY (uid) REFERENCES siteuser (id)
);
CREATE SEQUENCE style_id_seq;
CREATE INDEX style_id_index ON style (id);

CREATE VIEW vstyle AS
    SELECT st.id,st.number,st.stylename,st.miniature,st.url,st.description,
           to_char(st.create_date,'YYYY-MM-DD HH:MI:SS') as date
    FROM style st
    ORDER BY st.id;

insert into style(number,stylename,miniature,url) values('0001','pink','../style/images/style0001.png');
insert into style(number,stylename,miniature,url) values('0002','green','../style/images/style0002.png');
insert into style(number,stylename,miniature,url) values('0003','cyan','../style/images/style0003.png');
insert into style(number,stylename,miniature,url) values('0004','orange','../style/images/style0004.png');
insert into style(number,stylename,miniature,url) values('0005','offwhite','../style/images/style0005.png');

-----
-- 'company'
-----

DROP TABLE company;
DROP SEQUENCE company_id_seq;
DROP INDEX company_id_index;
DROP VIEW vcompany;

CREATE TABLE company (
    id            integer DEFAULT nextval('company_id_seq') NOT NULL,
    uid           integer   DEFAULT '1' NOT NULL, -- with foreign key
    linkman       varchar(20)  DEFAULT " NOT NULL,
    cncname       varchar(50)  DEFAULT " NOT NULL,    -- Chinese Companyname
    encname       varchar(50)  DEFAULT " NOT NULL,    -- English Companyname
    email         varchar(50)  DEFAULT " NOT NULL,
    telephone     varchar(20)  DEFAULT " NOT NULL,    -- telephone number
    fax           varchar(20)  DEFAULT " NOT NULL,    -- company fax
    Province_id   integer     DEFAULT '1' NOT NULL, -- with foreign key
    state         varchar(20)  DEFAULT " NOT NULL,    -- company state
    city          varchar(20)  DEFAULT " NOT NULL,    -- company city
    address       varchar(50)  DEFAULT " NOT NULL,    -- company address
    postcode      varchar(6)   DEFAULT " NOT NULL,    -- post office code
    range         text         DEFAULT " NOT NULL,

```

```

details      text DEFAULT " NOT NULL,    -- company details
other       text ,                      -- company other
bank        varchar(20)  DEFAULT " NOT NULL,  -- bank
bankaccount varchar(20)  DEFAULT " NOT NULL,  -- Back Account
Category_id integer    DEFAULT '1' NOT NULL,
style_id    integer    DEFAULT '1' NOT NULL,
create_date timestamp DEFAULT now() ,
modify_date timestamp DEFAULT now() ,
UNIQUE (id,uid),
PRIMARY KEY (id),
FOREIGN KEY (uid) REFERENCES siteuser (id),
FOREIGN KEY (style_id) REFERENCES style (id),
FOREIGN KEY (Province_id) REFERENCES Region (id),
FOREIGN KEY (Category_id) REFERENCES Category (id)
);

```

```

CREATE SEQUENCE company_id_seq;
CREATE INDEX company_id_index ON company (id);

```

```

CREATE VIEW vcompany AS
SELECT cp.id,cp.uid,su.username,cp.linkman,cp.cncname,cp.encname,cp.email,cp.telephone,cp.fax,
       vpv.province,cp.city,cp.address,cp.postalcode,cp.bank,cp.bankaccount,cg.category,
       to_char(cp.create_date,'YYYY-MM-DD HH:MI:SS') as date
FROM siteuser su,company cp,vProvince vpv,category cg
where su.id = cp.uid and cp.province_id = vpv.id and cp.category_id = cg.id
ORDER BY cp.id;

```

```

-----
-- 'link'
-----

```

```

DROP TABLE link;
DROP SEQUENCE    link_id_seq;
DROP INDEX       link_id_index;
DROP VIEW vlink;

```

```

CREATE TABLE link (
  id          integer DEFAULT nextval('link_id_seq') NOT NULL,
  uid         integer  DEFAULT '1' NOT NULL,    -- with foreign key
  linkname    varchar(30) DEFAULT " NOT NULL,  -- link name
  url         varchar(100) DEFAULT " NOT NULL,  -- link url
  image       varchar(100) DEFAULT " NOT NULL,  -- image(jpg,png,gif....)
  create_date timestamp DEFAULT now() ,
  modify_date timestamp DEFAULT now() ,
  UNIQUE (id,uid),

```



```

        PRIMARY KEY (id),
        FOREIGN KEY (uid) REFERENCES siteuser (id)
    );
CREATE SEQUENCE link_id_seq;
CREATE INDEX link_id_index ON company (id);

CREATE VIEW vlink AS
    SELECT l.id,l.uid,su.username,l.linkname,l.url,l.image,to_char(l.create_date,'YYYY/MM/DD
HH:MI:SS') as date
        FROM siteuser su, link l
        WHERE su.id = l.uid
        ORDER BY l.id;
-- ORDER BY commit_log.commit_date LIMIT 100;
-----
-- 'product_sort'
-----

DROP TABLE product_sort;
DROP SEQUENCE product_sort_id_seq;
DROP INDEX product_sort_id_index;
DROP VIEW vproduct_sort;

CREATE TABLE product_sort (
    id integer DEFAULT nextval('product_sort_id_seq') NOT NULL,
    uid integer DEFAULT '1' NOT NULL, -- with foreign key
    plist varchar(20) DEFAULT '' NOT NULL, --
    create_date timestamp DEFAULT now() ,
    modify_date timestamp DEFAULT now() ,
    PRIMARY KEY (id),
    UNIQUE (id,uid),
    FOREIGN KEY (uid) REFERENCES siteuser (id)
);
CREATE SEQUENCE product_sort_id_seq;
CREATE INDEX product_sort_id_index ON style (id);

CREATE VIEW vproduct_sort AS
    SELECT ps.id,ps.uid,su.username,ps.plist,to_char(ps.create_date,'YYYY-MM-DD HH:MI:SS') as
date
        FROM siteuser su, product_sort ps
        WHERE su.id = ps.uid
        ORDER BY ps.id;

CREATE OR REPLACE FUNCTION product_sort_id_del_func () RETURNS opaque AS '
-- DECLARE
BEGIN

```

```

        Delete from product where pid = OLD.id;
RETURN OLD;
END;
' LANGUAGE 'plpgsql';

DROP TRIGGER product_sort_tri on product_sort;
CREATE TRIGGER product_sort_tri
    BEFORE Delete
    --AFTER Delete
    ON product_sort FOR EACH ROW
    EXECUTE PROCEDURE product_sort_id_del_func ();

-----
-- 'product'
-----

DROP TABLE product;
DROP SEQUENCE    product_id_seq;
DROP INDEX       product_id_index;
DROP VIEW vproduct;

CREATE TABLE product (
    id            integer DEFAULT nextval('product_id_seq') NOT NULL,
    pid           integer  DEFAULT '1' NOT NULL, -- with foreign key
    pname         varchar(50) DEFAULT '' NOT NULL, -- product name
    price         numeric(8,2) DEFAULT '0.00' NOT NULL,
    newprice      numeric(8,2) DEFAULT '0.00' NOT NULL,
    amount        integer  DEFAULT '0' NOT NULL,
    image         varchar(100) DEFAULT 'default_product.jpg' NOT NULL, -- image(jpg,png,gif....)
    height        integer DEFAULT '320' NOT NULL,
    width         integer DEFAULT '320' NOT NULL,
    description   text ,
    isonline      boolean DEFAULT 'false' NOT NULL,
    create_date   timestamp DEFAULT now() ,
    modify_date   timestamp DEFAULT now() ,
    PRIMARY KEY (id),
    UNIQUE (id,pid),
    FOREIGN KEY (pid) REFERENCES product_sort (id)
);
CREATE SEQUENCE product_id_seq;
CREATE INDEX     product_id_index ON style (id);

CREATE VIEW vproduct AS
    SELECT    p.id,ps.id    as    pid,ps.plist,p.pname,to_char(p.price,'999G999D99')    as
price,to_char(p.newprice,'999G999D99') as newprice,p.amount,p.image,p.height,p.width,p.description,

```

```

CASE WHEN p.isonline=true THEN 'Y' ELSE 'N' END as isonline,
to_char(p.create_date,'YYYY-MM-DD') as date
FROM product_sort ps, product p
WHERE ps.id = p.pid
order by p.id;

```

```

-----
-- 'clientinfo'
-----

```

```

DROP TABLE clientinfo;
DROP SEQUENCE clientinfo_id_seq;
DROP INDEX clientinfo_id_index;
DROP VIEW vclientinfo;

```

```

CREATE TABLE clientinfo (
    id integer DEFAULT nextval('clientinfo_id_seq') NOT NULL,
    uid integer DEFAULT '1' NOT NULL, -- with foreign key
    cname varchar(20) DEFAULT " NOT NULL, -- link of friends
    telephone varchar(20) DEFAULT " NOT NULL,
    fax varchar(20) DEFAULT " NOT NULL,
    email varchar(50) DEFAULT " NOT NULL,
    address varchar(50) DEFAULT '320' NOT NULL,
    postalcode varchar(50) DEFAULT '320' NOT NULL,
    note text DEFAULT " NOT NULL,
    create_date timestamp DEFAULT now() ,
    modify_date timestamp DEFAULT now() ,
    PRIMARY KEY (id),
    UNIQUE (id),
    FOREIGN KEY (uid) REFERENCES siteuser (id)
);
CREATE SEQUENCE clientinfo_id_seq;
CREATE INDEX clientinfo_id_index ON clientinfo (id);

```

```

CREATE VIEW vclientinfo AS
SELECT ci.id,ci.uid,su.username,ci.cname,ci.telephone as
tel,ci.fax,ci.email,ci.address,ci.postalcode,ci.note,
to_char(ci.create_date,'YYYY-MM-DD HH:MI:SS') as date
FROM siteuser su, clientinfo ci
WHERE su.id = ci.uid;

```

```

CREATE OR REPLACE FUNCTION clientinfo_tri_func () RETURNS opaque AS '
-- DECLARE
-- user_id CONSTANT INTEGER := OLD.id;
BEGIN

```

```

        IF TG_OP = "DELETE" THEN
            Delete from prodorder where clientinfo_id = OLD.id;
        END IF;
    RETURN OLD;
END;
' LANGUAGE 'plpgsql';

DROP TRIGGER clientinfo_tri on clientinfo;
CREATE TRIGGER clientinfo_tri
    BEFORE Delete ON clientinfo FOR EACH ROW
    EXECUTE PROCEDURE clientinfo_tri_func ();
-----
-- 'production order'
-----

DROP TABLE prodorder;
DROP SEQUENCE    prodorder_id_seq;
DROP INDEX       prodorder_id_index;

CREATE TABLE prodorder (
    id            integer DEFAULT nextval('prodorder_id_seq') NOT NULL,
    pid          integer  DEFAULT '1' NOT NULL,  -- foreign key
    amount       integer  DEFAULT '1' NOT NULL,
    clientinfo_id integer  DEFAULT '1' NOT NULL,  -- foreign key
    create_date  timestamp DEFAULT now() ,
    modify_date  timestamp DEFAULT now() ,
    PRIMARY KEY (id),
    UNIQUE (id),
    FOREIGN KEY (pid) REFERENCES product (id),
    FOREIGN KEY (clientinfo_id) REFERENCES clientinfo (id)
);
CREATE SEQUENCE prodorder_id_seq;
CREATE INDEX    prodorder_id_index ON style (id);

-----
-- 'drumbeating'
-----

DROP TABLE drumbeating;
DROP SEQUENCE    drumbeating_id_seq;
DROP INDEX       drumbeating_id_index;
DROP VIEW vdrumbeating;

CREATE TABLE drumbeating (
    id            integer DEFAULT nextval('drumbeating_id_seq') NOT NULL,

```

```

uid            integer  DEFAULT '1' NOT NULL, -- with foreign key
logourl        varchar(100) DEFAULT " NOT NULL, -- image(jpg,png,gif....)
bannerurl     varchar(100) DEFAULT " NOT NULL,
lheight       integer DEFAULT '60' NOT NULL,
lwidth        integer DEFAULT '120' NOT NULL,
bheight       integer DEFAULT '60' NOT NULL,
bwidth        integer DEFAULT '468' NOT NULL,
description    text ,
create_date   timestamp DEFAULT now() ,
modify_date   timestamp DEFAULT now() ,
PRIMARY KEY (id),
UNIQUE (id,uid),
FOREIGN KEY (uid) REFERENCES siteuser (id)
);
CREATE SEQUENCE  drumbeating_id_seq;
CREATE INDEX     drumbeating_id_index ON style (id);

CREATE VIEW vdrumbeating AS
    SELECT dr.id,su.username,dr.logourl,dr.bannerurl
           FROM siteuser su, drumbeating dr
           WHERE su.id = dr.uid;

-----
-- 'news'
-----

DROP TABLE news;
DROP SEQUENCE  news_id_seq;
DROP INDEX     news_id_index;
DROP VIEW vnews;

CREATE TABLE news (
    id            integer DEFAULT nextval('news_id_seq') NOT NULL,
    uid           integer  DEFAULT '1' NOT NULL, -- with foreign key
    title        varchar(100) DEFAULT " NOT NULL, -- link of friends
    image        varchar(100) DEFAULT " NOT NULL,
    content      text      DEFAULT " NOT NULL,
    create_date  timestamp DEFAULT now() ,
    modify_date  timestamp DEFAULT now() ,
    PRIMARY KEY (id),
    UNIQUE (id),
    FOREIGN KEY (uid) REFERENCES siteuser (id)
);
CREATE SEQUENCE  news_id_seq;
CREATE INDEX     news_id_index ON news (id);

```

```

CREATE VIEW vnews AS
    SELECT ns.id,ns.uid,su.username,ns.title,ns.image,ns.content,to_char(ns.create_date,'YYYY/MM/DD
HH:MI:SS') as date
        FROM siteuser su, news ns
        WHERE su.id = ns.uid
        order by ns.id;

```

```

-----
-- 'count'
-----

```

```

DROP TABLE count;
DROP SEQUENCE    count_id_seq;
DROP INDEX       count_id_index;
DROP VIEW vcount;

```

```

CREATE TABLE count (
    id            integer DEFAULT nextval('count_id_seq') NOT NULL,
    uid           integer  DEFAULT '1' NOT NULL, -- with foreign key
    number        varchar(20) DEFAULT " NOT NULL,
    fontcolor     varchar(10) DEFAULT 'FFFFFF' NOT NULL,
    backgroundcolor varchar(10) DEFAULT '000000' NOT NULL,
    create_date   timestamp DEFAULT now() ,
    modify_date   timestamp DEFAULT now() ,
    PRIMARY KEY (id),
    UNIQUE (id,uid),
    FOREIGN KEY (uid) REFERENCES siteuser (id)
);

```

```

CREATE SEQUENCE count_id_seq;
CREATE INDEX     count_id_index ON count (id);

```

```

CREATE VIEW vcount AS
    SELECT c.id,su.username,c.number,c.fontcolor,c.backgroundcolor
        FROM siteuser su, count c
        WHERE su.id = c.uid;

```

```

-----
-- 'column_bar'
-----

```

```

DROP TABLE column_bar;
DROP SEQUENCE    column_bar_id_seq;
DROP INDEX       column_bar_id_index;
DROP VIEW vcolumn_bar;

```

```

CREATE TABLE column_bar (
    id            integer DEFAULT nextval('column_bar_id_seq') NOT NULL,
    uid          integer  DEFAULT '1' NOT NULL, -- with foreign key
    colname      varchar(10) DEFAULT " NOT NULL, -- link of friends
    url         varchar(50) DEFAULT " NOT NULL,
    title        varchar(50) DEFAULT " NOT NULL,
    image        varchar(50) DEFAULT " NOT NULL,
    height       varchar(50) DEFAULT " NOT NULL,
    width        varchar(50) DEFAULT " NOT NULL,
    content      text DEFAULT " NOT NULL,
    create_date  timestamp DEFAULT now() ,
    modify_date  timestamp DEFAULT now() ,
    PRIMARY KEY (id),
    UNIQUE (id),
    FOREIGN KEY (uid) REFERENCES siteuser (id)
);
CREATE SEQUENCE column_bar_id_seq;
CREATE INDEX      column_bar_id_index ON column_bar (id);

```

```

CREATE VIEW vcolumn_bar AS
    SELECT
cb.id,cb.uid,su.username,cb.colname,cb.url,cb.title,cb.image,cb.content,to_char(cb.create_date,'YYYY-M
M-DD HH:MI:SS') as date
    FROM siteuser su, column_bar cb
    WHERE su.id = cb.uid
    ORDER BY cb.id;

```

```

-----
-- 'guestbook'
-----

```

```

DROP TABLE guestbook;
DROP SEQUENCE guestbook_id_seq;
DROP INDEX      guestbook_id_index;
DROP VIEW vguestbook;

```

```

CREATE TABLE guestbook (
    id            integer DEFAULT nextval('guestbook_id_seq') NOT NULL,
    uid          integer  DEFAULT '1' NOT NULL, -- with foreign key
    name         varchar(20) DEFAULT " NOT NULL, -- link of friends
    email        varchar(50) DEFAULT " NOT NULL,
    telephone    varchar(20) DEFAULT " NOT NULL,
    fax          varchar(20) ,
    homepage     varchar(50) ,

```

```

        title      varchar(50) DEFAULT " NOT NULL,
        content    text ,
        style_id   integer  DEFAULT '1' NOT NULL,
        create_date timestamp DEFAULT now() ,
        modify_date timestamp DEFAULT now() ,
        PRIMARY KEY (id),
        UNIQUE (id),
        FOREIGN KEY (uid) REFERENCES siteuser (id),
        FOREIGN KEY (style_id) REFERENCES style (id)
    );
CREATE SEQUENCE  guestbook_id_seq;
CREATE INDEX     guestbook_id_index ON guestbook (id);

CREATE VIEW vguestbook AS
    SELECT          gb.id,gb.uid,su.username,gb.name,gb.email,gb.telephone          as
tel,gb.fax,gb.homepage,gb.title,gb.content,to_char(gb.create_date,'YYYY/MM/DD HH:MI:SS') as date
    FROM siteuser su, guestbook gb
    WHERE su.id = gb.uid
    ORDER BY gb.id;

-----
-- 'Category'
-----

DROP TABLE Category;
DROP SEQUENCE  Category_id_seq;
DROP INDEX     Category_id_index;
DROP VIEW vCategory;

CREATE TABLE Category (
    id          integer DEFAULT nextval('Category_id_seq') NOT NULL,
    Category    varchar(20) DEFAULT " NOT NULL,
    description text ,
    note        text ,
    remark      text ,
    create_date timestamp DEFAULT now() ,
    modify_date timestamp DEFAULT now() ,
    PRIMARY KEY (id),
    UNIQUE (id,Category)
);
CREATE SEQUENCE  Category_id_seq;
CREATE INDEX     Category_id_index ON Category (id);

CREATE VIEW vCategory AS
    SELECT          c.id,c.Category,c.description,c.note,c.remark,to_char(c.create_date,'YYYY-MM-DD

```



```

HH:MI:SS') as date
        FROM Category c
        ORDER BY c.id;

-----
-- 'Region'
-----

DROP TABLE region;
DROP SEQUENCE    region_id_seq;
DROP INDEX       region_id_index;
DROP VIEW vregion;

CREATE TABLE region (
    id            integer DEFAULT nextval('region_id_seq') NOT NULL,
    region        varchar(20) DEFAULT " " NOT NULL,
    description   text ,
    note          text ,
    remark        text ,
    create_date   timestamp DEFAULT now() ,
    modify_date   timestamp DEFAULT now() ,
    PRIMARY KEY (id),
    UNIQUE (id,region)
);
CREATE SEQUENCE region_id_seq;
CREATE INDEX     region_id_index ON region (id);

CREATE VIEW vregion AS
    SELECT    pv.id,pv.region,pv.description,pv.note,pv.remark,to_char(pv.create_date,'YYYY-MM-DD
HH:MI:SS') as date
        FROM region pv
        ORDER BY pv.id;

-----
-- 'province' VIEW
-----

DROP VIEW vprovince;
CREATE VIEW vprovince AS
    SELECT                                pv.id,pv.region                                as
province,pv.description,pv.note,pv.remark,to_char(pv.create_date,'YYYY-MM-DD HH:MI:SS') as date
        FROM region pv
        ORDER BY pv.id;

-----
-- 'Country'

```





```
insert into region(region) values(' ');
insert into region(region) values(' ');
insert into region(region) values(' ');
insert into region(region) values(' ');
insert into region(region) values(' ');
insert into region(region) values(' ');
insert into region(region) values(' ');
insert into region(region) values(' ');
insert into region(region) values(' ');
```

```
--insert into country(domain,country) values(",");
insert into country(domain,country) values('AL',' ');
insert into country(domain,country) values('DZ',' ');
insert into country(domain,country) values('AF',' ');
insert into country(domain,country) values('AR',' ');
insert into country(domain,country) values('AE',' ');
insert into country(domain,country) values('AW',' ');
insert into country(domain,country) values('OM',' ');
insert into country(domain,country) values('AZ',' ');
insert into country(domain,country) values('EG',' ');
insert into country(domain,country) values('ET',' ');
insert into country(domain,country) values('IE',' ');
insert into country(domain,country) values('EE',' ');
insert into country(domain,country) values('AD',' ');
insert into country(domain,country) values('AO',' ');
insert into country(domain,country) values('AI',' ');
insert into country(domain,country) values('AG',' ');
insert into country(domain,country) values('AT',' ');
insert into country(domain,country) values('AU',' ');
insert into country(domain,country) values('MO',' ');
insert into country(domain,country) values('BB',' ');
insert into country(domain,country) values('PG',' ');
insert into country(domain,country) values('BS',' ');
insert into country(domain,country) values('PK',' ');
insert into country(domain,country) values('PY',' ');
insert into country(domain,country) values('BH',' ');
insert into country(domain,country) values('PA',' ');
insert into country(domain,country) values('BR',' ');
insert into country(domain,country) values('BY',' ');
insert into country(domain,country) values('BM',' ');
insert into country(domain,country) values('BG',' ');
insert into country(domain,country) values('MP',' ');
insert into country(domain,country) values('BJ',' ');
```











```

insert into country(domain,country) values('UZ',' ');
insert into country(domain,country) values('ES',' ');
insert into country(domain,country) values('GR',' ');
insert into country(domain,country) values('HK',' ');
insert into country(domain,country) values('SG',' ');
insert into country(domain,country) values('NC',' ');
insert into country(domain,country) values('NZ',' ');
insert into country(domain,country) values('HU',' ');
insert into country(domain,country) values('SY',' ');
insert into country(domain,country) values('JM',' ');
insert into country(domain,country) values('AM',' ');
insert into country(domain,country) values('YE',' ');
insert into country(domain,country) values('IQ',' ');
insert into country(domain,country) values('IR',' ');
insert into country(domain,country) values('IL',' ');
insert into country(domain,country) values('IT',' ');
insert into country(domain,country) values('IN',' ');
insert into country(domain,country) values('ID',' ');
insert into country(domain,country) values('UK',' ');
insert into country(domain,country) values('VG',' ');
insert into country(domain,country) values('JO',' ');
insert into country(domain,country) values('VN',' ');
insert into country(domain,country) values('ZM',' ');
insert into country(domain,country) values('TD',' ');
insert into country(domain,country) values('GI',' ');
insert into country(domain,country) values('CL',' ');
insert into country(domain,country) values('CF',' ');
insert into country(domain,country) values('CN',' ');
insert into country(domain,country) values('TW',' ');

commit;
ROLLBACK;

```

## 17.3 安装脚本

### 17.3.1 setenv.sh

```

#=====
# Author:Netkiller
# Script: setenv.sh
#=====

```

```

SRCHOME=../src
BINARYHOME=../binary
RPMHOME=../rpms
CONFHOME=../conf
# Install Portable Threads
PThreads=pth-2.0.0.tar.gz

# Shared Memory Allocation
SMemory=mm-1.3.0.tar.gz

APACHE_HOME=/usr/local/apache
APACHE=apache_1.3.27.tar.gz
MOD_SSL=mod_ssl-2.8.14-1.3.27.tar.gz
MOD_PERL=mod_perl-1.0-current.tar.gz
MOD_FASTCGI=mod_fastcgi-2.4.0.tar.gz
PHP=php-4.3.2.tar.gz

```

## 17.3.2 install.sh

```

#=====
# Author:Netkiller
# Script:install.sh
#=====
if [ -f setenv.sh ]; then
    . setenv.sh
fi

#rpms_location="$base_dir/"

POSTGRESQL=${RPMHOME}
Install_PostgreSQL(){
    cd $POSTGRESQL
    rpm -Uvh --nodeps postgresql-libs-?.?.?-1PGDG.i386.rpm
    rpm -Uvh --nodeps postgresql-?.?.?-1PGDG.i386.rpm
    rpm -Uvh --nodeps postgresql-devel-?.?.?-1PGDG.i386.rpm
    rpm -Uvh --nodeps postgresql-server-?.?.?-1PGDG.i386.rpm
    rpm -Uvh --nodeps postgresql-contrib-?.?.?-1PGDG.i386.rpm
    rpm -Uvh --nodeps postgresql-docs-?.?.?-1PGDG.i386.rpm
    rpm -Uvh --nodeps postgresql-jdbc-?.?.?-1PGDG.i386.rpm
    rpm -Uvh --nodeps postgresql-pl-?.?.?-1PGDG.i386.rpm
    rpm -Uvh --nodeps postgresql-python-?.?.?-1PGDG.i386.rpm
    rpm -Uvh --nodeps postgresql-tcl-?.?.?-1PGDG.i386.rpm
    rpm -Uvh --nodeps postgresql-test-?.?.?-1PGDG.i386.rpm

```

```

    cd ..
    rpm -qa|grep post
}

Uninstall_PostgreSQL(){
    rpm -e --nodeps `rpm -qa |grep postgresql`
}

Status_PostgreSQL(){
    rpm -qa|grep postgresql
}

Install_Apache(){
    cd ${SRCHOME}
    tar zxf apache_?.?.?.tar.gz >/dev/null
    cd apache_?.?.?.
    ./configure --prefix=/usr/local/apache --enable-module=so
    make
    make install
    cd ..
    rm -rf apache_?.?.?.
    echo "/usr/local/apache/bin/apachectl start" >> /etc/rc.d/rc.local
}

Install_PHP(){
    cd ${SRCHOME}
    tar zxf php-?.?.?.tar.gz
    cd php-?.?.?.
    ./configure --prefix=/usr/local/php --with-apxs=/usr/local/apache/bin/apxs \
        --with-config-file-path=/usr/local/etc --enable-track-vars --with-xml \
        --with-pgsql --with-mysql \
        --with-ldap --enable-ftp --with-openssl --with-iconv --with-kerberos \
        --with-java=/usr/local/java
    make
    make install
    cp php.ini-dist /usr/local/etc/php.ini
    #LoadModule php4_module modules/libphp4.so
    #AddType application/x-httpd-php .php .php4 .php3 .phtml
}

Install_MySQL(){
    if ! rpm -qa|grep -q "mysql" ; then
    tar zxvf mysql-standard-4.0.12-pc-linux-i686.tar.gz >/dev/null
    mv mysql-standard-4.0.12-pc-linux-i686 /usr/local/mysql

```

```
cp /usr/local/mysql/support-files/mysql.server /etc/init.d/mysqld
chkconfig mysqld reset
chown mysql.mysql -R /usr/local/mysql
else
    echo "mysql package is already installed"
fi
# echo "su mysql -c"cd /usr/local/mysql/ ;./configure">/dev/null">> /etc/rc.d/rc.local
}
```

```
Install_MySQL(){
cd ${RPMSHOME}
rpm -Uvh MySQL-server-4.0.13-0.i386.rpm
rpm -Uvh MySQL-client-4.0.13-0.i386.rpm
rpm -Uvh MySQL-devel-4.0.13-0.i386.rpm
rpm -Uvh MySQL-shared-4.0.13-0.i386.rpm
rpm -Uvh MySQL-shared-compat-4.0.13-0.i386.rpm

}
```

```
Install_Sun_J2sdk(){
cd ${BINHOME}
chmod 700 j2sdk-1_4_1_02-linux-i586.bin
./j2sdk-1_4_1_02-linux-i586.bin
mv j2sdk1.4.1_02 /usr/local/java
cp ../conf/profile.sh /etc/profile.d/java.sh
chmod 755 /etc/profile.d/java.sh
}
```

```
Install_Tomcat(){
cd ${BINARYHOME}
tar zxvf tomcat-4.1.24.tar.gz >/dev/null
mv jakarta-tomcat-4.1.24 /usr/local/jakarta-tomcat
# echo "/usr/local/jakarta-tomcat/bin/startup.sh" >> /etc/rc.d/rc.local
}
```

```
Install_Ant(){
cd ${BINARYHOME}
tar zxvf apache-ant-1.5.3-1-bin.tar.gz >/dev/null
mv apache-ant-1.5.3-1 /usr/local/apache-ant
}
```

```
Install_Zhcon(){
```

```

tar zxvf zhcon-?.?.?.tar.gz
./configure
make
make install
}

usage()
{
    echo "Usage: $0 {install|uninstall|status|reinstall|config} [base_dir]"
    echo "Usage: $0 {apache|php|mysql} [base_dir]"
    echo "Usage: $0 {j2sdk|tomcat|ant|jdbc|postgresql} [base_dir]"
#   echo "base_dir          the location of install disk, optional"
}

# --- main process starts ---

case "$1" in
    install)
        banner
        install
        ;;
    uninstall)
        banner
#       uninstall
#Uninstall_PostgreSQL
        ;;
    status)
        banner
        status
        ;;
    reinstall)
        banner
        uninstall
        install
        ;;
    config)
        banner
        config
        ;;
    apache)
        Install_Apache
        ;;
    php)
        Install_PHP

```

```
;;
mysql)
    Install_MySQL
    ;;
j2sdk)
    Install_Sun_J2sdk
    ;;
tomcat)
    Install_Tomcat
    ;;
ant)
    Install_Ant
    ;;
postgresql)
    Install_PostgreSQL
    ;;
*)
    usage
    exit 1
esac
```

# --- main process ends ---

## 17.4 附件

<http://linux.9812.net/>

## 17.5 其它

Mailing Lists

<http://groups.yahoo.com/group/pgsql/>

Post message: [pgsql@yahoogroups.com](mailto:pgsql@yahoogroups.com)

Subscribe: [pgsql-subscribe@yahoogroups.com](mailto:pgsql-subscribe@yahoogroups.com)

Unsubscribe: [pgsql-unsubscribe@yahoogroups.com](mailto:pgsql-unsubscribe@yahoogroups.com)

List owner: [pgsql-owner@yahoogroups.com](mailto:pgsql-owner@yahoogroups.com)

<http://www.dm2.com.cn/>

latex

[http://learn.tsinghua.edu.cn/homepage/2001315450/tex\\_frame.html](http://learn.tsinghua.edu.cn/homepage/2001315450/tex_frame.html)

# 18

- <http://www.postgresql.org>
- <http://www.pgsqldb.org/pgsqldoc-cvs/index.html>
- <http://www.oasis-open.org/>
- <http://www.linuxfocus.org/>
- <http://www.opendocspublishing.com/>
- <http://www.casestudio.com/>
- <http://www.datanamic.com/>
- <http://www.commandprompt.com/>
- <http://www.commandprompt.com/ppbook/>
- <http://www.tldp.org/>

# 19

		PostgreSQL
		2.0
		Netkiller( )
		2003 10 7 2003-12-5

## 版本历史

/				
1.0.0/			2003-10-15	
1.1.0/			2003-10-17	PostgreSQL RPM Jsp/Java
1.1.1			2003-10-22	,
1.1.2			2003-10-23	substring() PHP

1.1.3			2003-10-24	
1.2.3			2003-10-25	SSL,SSH
1.2.3			2003-10-29	
1.3.3			2003-11-1	
1.4.3			2003-11-11	RPM
1.4.4			2003-11-12	-
1.5.0			2003-11-25	
1.6.0			2003-11-28	" "
1.6.1			2003-11-29	-
1.6.2			2003-12-1	DW JSP Java web.xml
/			2003-12-5	Jbuilder+weblogic+postgresql
2.0.0				
2.1.0			2003/12/12	20 " ..." -PGP/GPG
2.2.1			2003-12-27	
2.2.0			2004-2-24	check
2.3.0			2004-2-26	
2.3.1		Vvlinux	2004-3-4	
2.3.2			2004-3-5	
			2004-3-12	
			2004-4-12	
		<a href="#">qzhou9887</a>	2004-4-13	<a href="#">qzhou9887</a>
			2004-5-20	Tomcat
			2004-6-30	
			2004-7-8	APT