



Documentation

dbRadar

- [Features](#)
- [Connect](#)
- [Worksheets](#)
- [Databases](#)
- [Oracle Application](#)
- [Debug](#)
- [System load](#)
- [Database Objects](#)
- [F.A.Q](#)



dbRadar is a tool for the administration, monitoring and development of a variety of Databases. It is available for Linux and Windows (Mac OS X coming soon). dbRadar competes very favorably with products such as TOAD from Quest Software and NORAD from Bradmark, but at a fraction of the price. dbRadar works with the following databases: · Oracle · PostgreSQL · MySQL · DB2.

See more on [features](#) ...

[Home](#)

Copyright © 2004 [theKompany](#) dbRadar version 1.1



Features

- [Features](#)
- [Planned features](#)

dbRadar is a tool for the administration, monitoring and development of a variety of Databases. It is available for Linux and Windows (Mac OS X coming soon). dbRadar competes very favorably with products such as TOAD from Quest Software and NORAD from Bradmark, but at a fraction of the price.

dbRadar works with the following databases:

- Oracle
- PostgreSQL
- MySQL
- DB2
- ODBC (only for Windows)

If the connection is successful the worksheet will open and the user can then use any DDL and DML that the target database supports. dbRadar uses the MDI approach for representing all of its child windows. This means that the user can open as many windows as he/she wants, close and navigate through them. From the File menu the user can Print, Save and Load any text files into the worksheet and execute them as sql scripts. The user can drop the actual connection with the database and start a new one from the File menu.

In the worksheet the user can cut, copy and paste any text using the appropriate key (CTRL-X, CTRL-C, CTRL-V) combination or by choosing the edit menu after the text has selected. The Database menu offers the user ability to open a new worksheet or a new system load window. To switch from one window to another, use the Window menu. For Oracle and PostgreSQL connections the user has the ability to use code completion for selecting columns from the tables. The user can save the result set by right clicking the mouse over the grid and choosing a name for the file.

For Oracle connections only, the user can browse the objects from the database using the tree representation from the left and also can choose to see the execution path for a specific select statement by clicking the explain button from the worksheet toolbar.

Features

- **Worksheet** - Issue any DDL or DML that is supported by the target database. The result set is

presented in a grid from which it can also be saved. A status display will indicate the progress of the query.

- Threaded Queries - Queries run in their own thread so that the application remains responsive even during long executions. A special feature for Oracle databases is the ability to break any long running query.
- Multiple, simultaneous database connections. The user can switch from one connection to another.
- Powerfull editor with Syntax highlighting for : SQL, PL/SQL, PGSQL, MYSQL, DB2
- Code completion - Makes it quick and easy to select columns you want from your particular database. This feature is only available for Oracle and PostgreSQL currently.
- Oracle Applications - Only for Oracle Business Suite 11.0.3 and 11.5. You can monitor OA users and kill sessions. An upcoming version will allow you to view concurrent requests.
- Remote Info - Gather remote/local HP-UX and Linux load (system, user, idle) and processes.
- Database Objects - Users can visually browse and edit database objects (Oracle only).
- Save/Load any text files and execute them as SQL scripts.
- Print any text in a worksheet.
- Tuning - The worksheet provides a tabbed page where the user can see the execution path for the current statement executed in a tree fashion (only for Oracle currently)
- Supports statements with placeholders in the worksheet. This way a user can insert something like a large object into the database.

Planned Features

- FTP integration in a project.
- Windows remote load shown in graphically.
- Users can visually browse and edit database objects for Postgres, Mysql and DB2.
- Support for Sql Server, Sybase, Sap, Interbase, Firebird, SqlLite
- Show online database performance indicators (buffer cache hits, library cache hits - for Oracle databases)
- By making use of our multiple database connections we will also be able to migrate data from one database to another.

[Home](#)



Connect to Databases

When dbRadar starts the connect screen will appear:

Driver	Database	User	Mode	Host
--------	----------	------	------	------

Connection settings

Driver: ORACLE

Database Name: ORACLE

Username: PostgreSQL

Password: MySQL

Connection Mode: Normal

Hostname:

Port: Default

Oracle Applications

Unix Login

OK Cancel

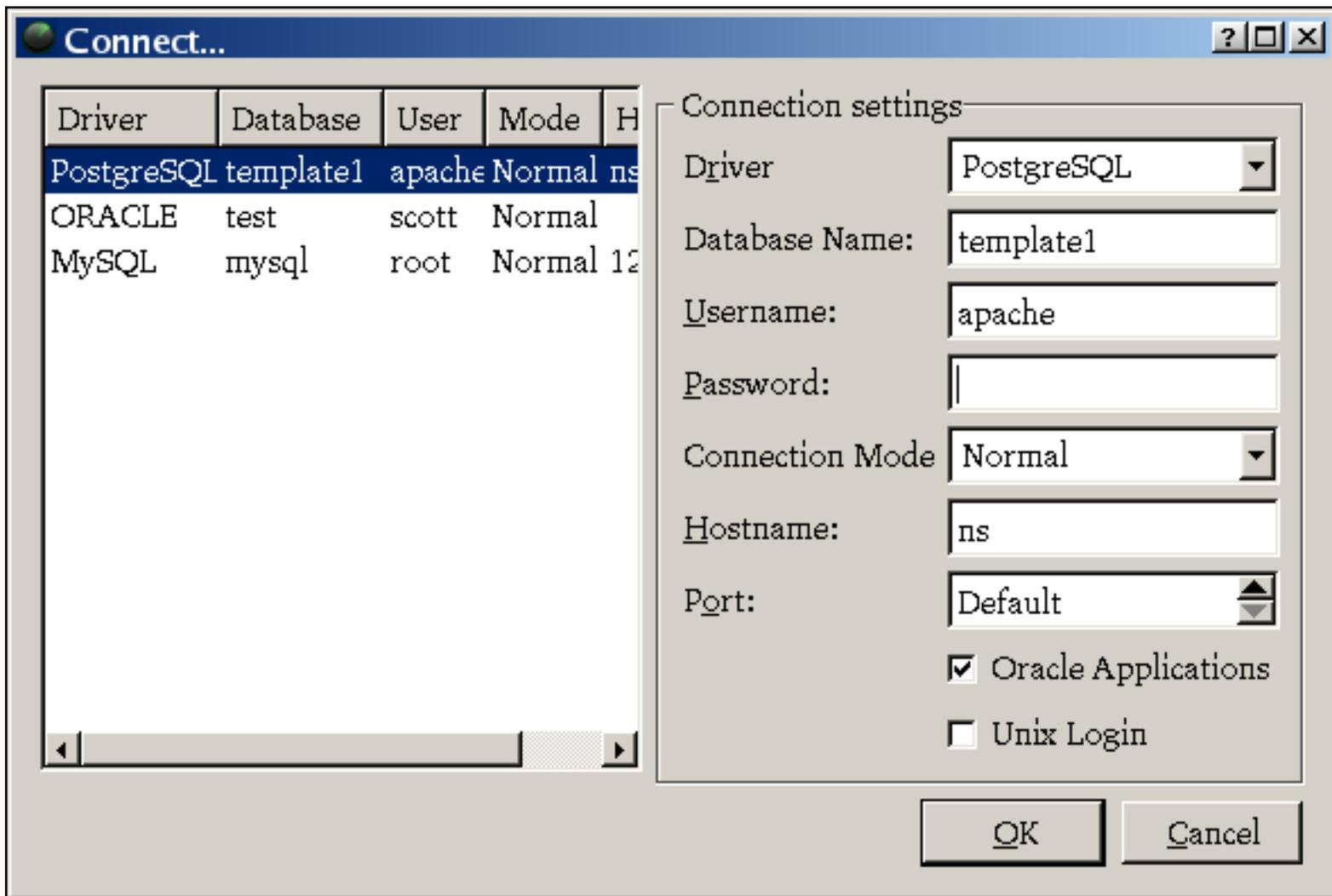
Oracle client libraries 8i/9i and DB2 client libraries need to be installed to have access to those databases. Choose the connection you want from the driver combo box and fill the text boxes with the appropriate information. Ex for Oracle connections:

The screenshot shows the 'Connect...' dialog box. It features a table with columns for Driver, Database, User, Mode, and Host. Below the table is a 'Remote Info' section with fields for Username (oracle) and Password (***). The 'Connection settings' section includes fields for Driver (ORACLE), Database Name (test), Username (scott), Password (*****), Connection Mode (Normal), Hostname (192.10.10.10), and Port (Default). There are also checkboxes for 'Oracle Applications' and 'Unix Login'. The dialog concludes with 'OK' and 'Cancel' buttons.

- Driver - the driver that creates the connection

- Database Name - the name of the database we want to connect(for Oracle - connect string)
- Username - the user who will make the connection
- Password - the password for the user
- Connection Mode - (only for Oracle) :Normal, Sys_Dba, Sys_Oper
- Hostname - the name or IP address of database server we want to connect to. (Not needed for Oracle). Based on this address we can monitor the remote server
- Port - the port for the server database. Use this port if the database process is running on other port than the default one.
- Oracle Applications - if the Oracle database is running Oracle Applications 11.0 or 11.5 the sysdba can monitor users and kill sessions. An upcoming version will allow you to view concurrent requests.
- Unix login - asks you for the username and password of a valid user that has shell access on the server. This way you can monitor the load of the server and all running processes. The remote server needs to be Linux or HP-UX and have rexecd (512 port) daemon process running so you can have access to those informations. The connections are logged so when you connect next time you don't have to fill in all the information for the connection. Just choose the appropriate

one from the list of connections:



If the connection is successful the worksheet is open and the user can use any DDL and DML that the target database supports. dbRadar uses the MDI approach for representing all of its child windows.

This means that the user can open as many windows as he wants, close and navigate through them. From the File menu the user can Print, Save and Load any text files into the worksheet and execute them as SQL scripts. The user can drop the actual connection with the database and start a new one from the File menu. In the worksheet the user can cut, copy and paste any text using the appropriate key (CTRL-X, CTRL-C, CTRL-V) combination or by choosing the edit menu after the text has selected. The Database menu offers the user the ability to open a new worksheet or a new system load window. To switch from one window to another use the Window menu.

[Home](#)



Worksheet

- [Sintax Highlight](#)
- [Code Completion](#)
- [Explain Plan](#)
- [Placeholders](#)

Sintax Highlight

In the worksheet the user can Issue any DDL or DML that the target database supports. For Oracle PL/SQL code users can navigate through the lines which have errors. Having support for Syntax highlighting makes programming more intuitive.

The screenshot shows the DB Radar 1.1 interface. On the left, a tree view displays database objects, with 'SYS.PSTUB' highlighted under the 'Procedures' folder. The main window is the 'Sql Editor', which contains the following PL/SQL code:

```

CREATE OR REPLACE procedure      SYS.PSTUB(pname varchar2, uname va
      stubSpec in out varchar2, stubText in out varchar2,
      flags varchar2 := '6') is
rc varchar2(40);
ty varchar2(5);
cursor tub (una varchar2, dbna varchar2, luna varchar2, luty varchar2) is
  select line from sys.pstubtbl
  where (una is null or username = una) and
        (dbna is null or dbname = dbna) and
        lun = luna and luty = luty
  order by lineno;
begin -- main
  sys.pstubtbl(pname, uname, '', flags, rc);
  if rc like '$$$%' then stubText := rc; return; end if;
  if not (rc = 'PKG' or rc = 'SUB')
    then stubText := '$$$ other'; return;
  end if;
  stubSpec := '';          asasa
  stubText := '';
  if rc = 'PKG' then
    for s in tub(uname, '', pname, 'PKS') loop
      stubSpec := stubSpec || s.line;
    end loop;
  end if;
  if rc = 'PKG' then ty := 'PKB'; else ty := 'SUB'; end if;
  for s in tub(uname, '', pname, ty) loop
    stubText := stubText || s.line;
  
```

At the bottom of the editor, an error message is displayed: "19 3 PLS-00103: Encountered the symbol "STUBTEXT" when expecting one of the follow". The status bar at the bottom indicates "Query executed!" and "Ln:18, Col:37".

Code Completion

For Oracle and PostgreSQL connections the user has the ability to use code completion for selecting columns from the tables. The user can save the result set by right clicking the mouse over the grid and choosing a name for the file.

The screenshot shows the DB Radar 1.1 interface. The left pane displays a tree view of database objects for the SCOTT user, including tables (SMP_SE, SMP_UF, SMP_VA) and sequences (BLOBID, CHESS_, CUSTID, EVT_NO, EVT_OP, EVT_PR, ORDID, PRODID, SMACTL). The main workspace shows a SQL query in the editor:

```
--code completion
select * from user_tables t where t.
```

The execution plan is displayed in the Result pane, showing 11 rows retrieved in 0.431 seconds. A dropdown menu is open over the table names, listing columns: TABLE_NAME, TABLESPACE_NAME, CLUSTER_NAME, IOT_NAME, PCT_FREE, PCT_USED, INI_TRANS, MAX_TRANS, INITIAL_EXTENT, and NEXT_EXTENT.

TABLE_NAME	TAB	IOT_NAM
1	ACCTS	SYS
2	ACCT_ADDRS	SYS
3	BLOB	SYS
4	BONUS	SYS
5	CALENDAR_RANGES	SYSTEM
6	CHESS_SAVE	SYSTEM
7	CHESS_SAVE_BOARDSTATE	SYSTEM
8	CHESS_SAVE_CAPTURED	SYSTEM
9	CHESS_SAVE_PLAYER	SYSTEM
10	CITIES	SYSTEM
11	COMPANY_SUMMARY	SYSTEM

11 rows retrieved in 0.431 s Ln:2, Col:37

Explain Plan

For Oracle connections only, the user can browse the objects from the database using the tree representation from the left and can also choose to see the execution path for a specific select statement by clicking the explain button from the worksheet toolbar.

The screenshot shows the DB Radar 1.1 interface. The top menu includes File, Edit, View, Database, Worksheet, Oracle Applications, Window, and Help. The current connection is SYS@mar1. The left sidebar shows a tree view of database objects, including SYS.TIMESTAMP WITH, SYS.UNSIGNED BINARY, SYS.VARCHAR, SYS.VARCHAR2, SYS.VARYING ARRAY, Type bodies, Triggers, and Synonyms. The main area displays a SQL query in the Sql Editor:

```
select * from all_tables t where t.TABLE_NAME like '%ALL%'
```

Below the query, the Explain Plan is shown. The plan consists of 19 steps (0-18) involving nested loops, table access, and index scans. The status bar indicates 3 rows retrieved in 0.122 s.

Id	Operation	Options	Object name	Object
0	SELECT STATEMENT			
1	FILTER			
2	NESTED LOOPS			
3	NESTED LOOPS			
4	NESTED LOOPS	OUTER		
5	NESTED LOOPS	OUTER		
6	NESTED LOOPS	OUTER		
7	NESTED LOOPS	OUTER		
8	NESTED LOOPS	OUTER		
9	TABLE ACCESS	FULL	OBJ\$	
10	TABLE ACCESS	CLUSTER	TAB\$	
11	INDEX	UNIQUE SCAN	I_OBJ#	NON-U
12	TABLE ACCESS	BY INDEX ROWID	OBJ\$	
13	INDEX	UNIQUE SCAN	I_OBJ1	UNIQUE
14	TABLE ACCESS	BY INDEX ROWID	OBJ\$	
15	INDEX	UNIQUE SCAN	I_OBJ1	UNIQUE
16	TABLE ACCESS	CLUSTER	USER\$	
17	INDEX	UNIQUE SCAN	I_USER#	NON-U
18	TABLE ACCESS	CLUSTER	SEG\$	

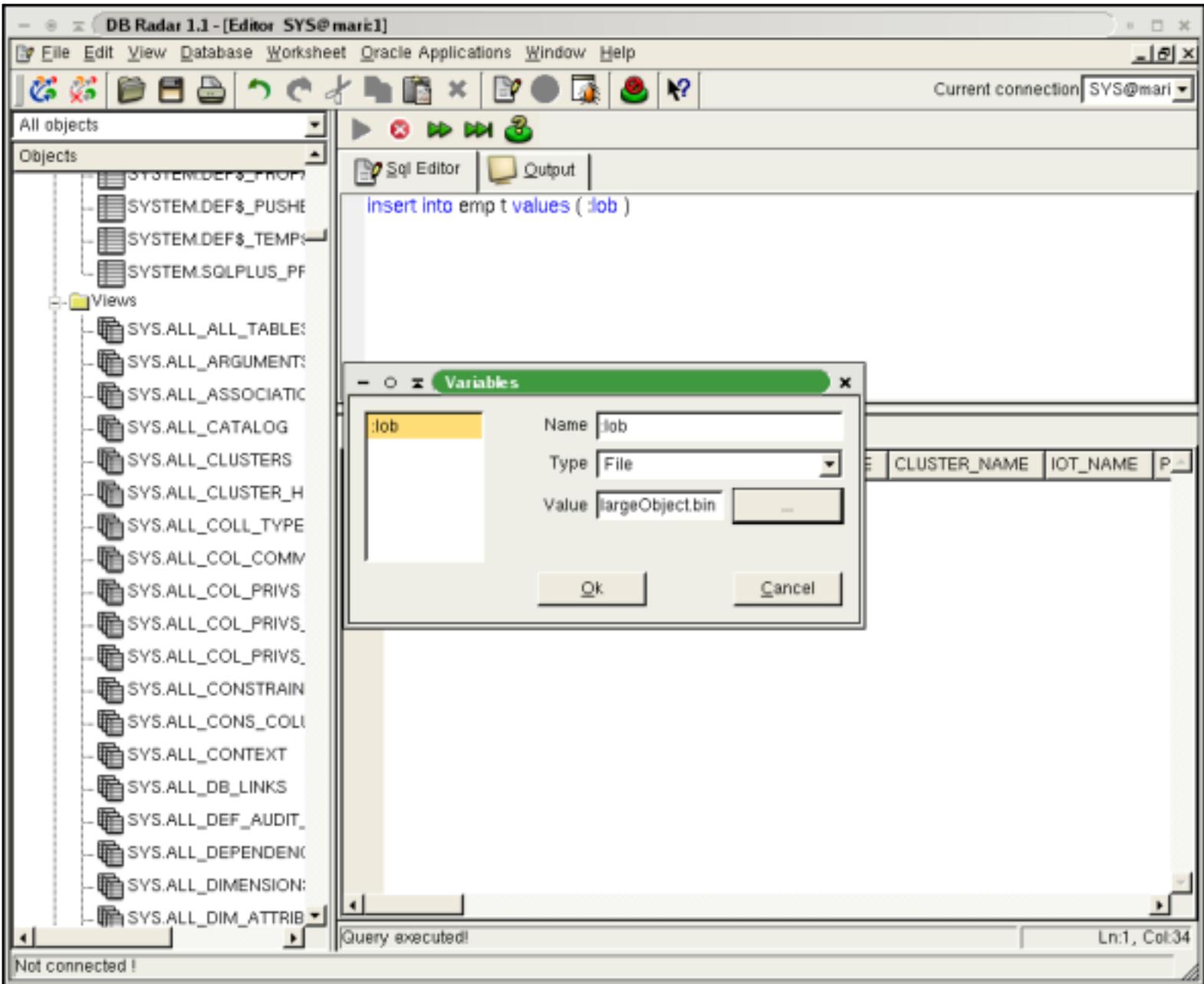
3 rows retrieved in 0.122 s

Ln:1, Col:59

Not connected!

Placeholders

Supports statements with placeholders in the worksheet. This way a user can insert something like a large object into the database.



[Home](#)



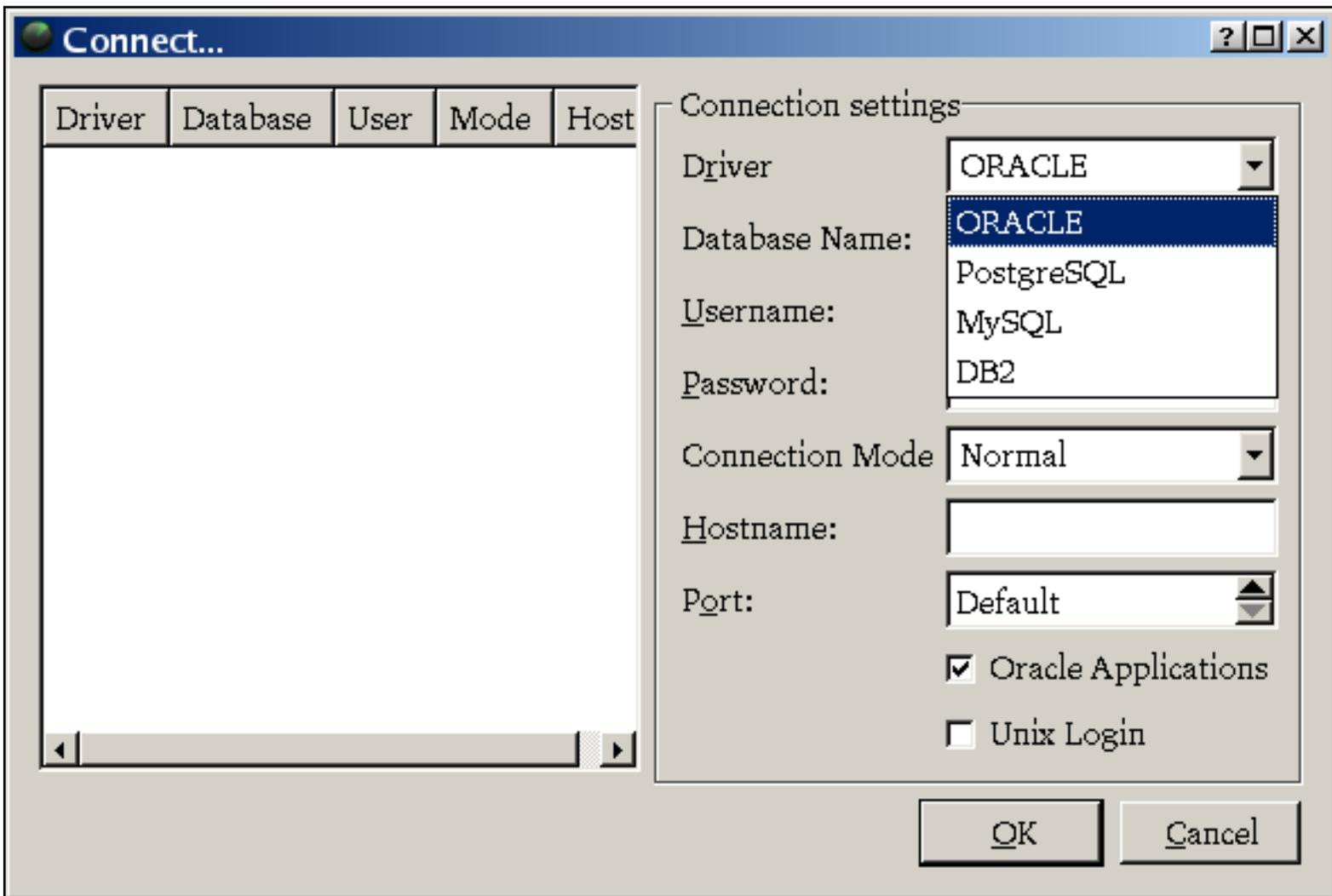
Databases

dbRadar works with the following databases:

- [Oracle](#)
- [PostgreSql](#)
- [MySql](#)
- [DB2](#)

Choose the connection you want from the driver combo box and fill in the text boxes with appropriate information. Ex for Oracle connections:

Connections



There is support for multiple simultaneous database connections. Users can run multiple queries against any of the above queries without causing the user interface to hang.

The screenshot displays the DB Radar 1.1 application window. The top menu bar includes File, Edit, View, Database, Worksheet, Oracle Applications, Window, and Help. The current connection is set to DB2INST1. The interface is divided into several panes:

- My objects:** A tree view on the left showing the hierarchy of database objects for DB2INST1@sample, ROOT@mysql, APACHE@ulogd, and SYS@mari.
- Editor SYS@mari:1:** Shows an Oracle connection with the SQL query `select * from all_tables`. The result table lists system tables:

	OWNER	TABLE_NAME	TABLESPACE_1
1	SYS	IND\$	SYSTEM
2	SYS	FILE\$	SYSTEM
3	SYS	UNDO\$	SYSTEM
4	SYS	CLU\$	SYSTEM
5	SYS	BOOTSTRAP\$	SYSTEM

- Editor APACHE@ulogd:1:** Shows a PostgreSQL connection with the SQL query `select * from pg_tables`. The result table lists PostgreSQL tables:

	schemaname	tablename	tableowner
1	pg_catalog	pg_description	postgres
2	pg_catalog	pg_group	postgres
3	pg_catalog	pg_proc	postgres
4	pg_catalog	pg_rewrite	postgres
5	pg_catalog	pg_xactlock	postgres

- Editor ROOT@mysql:1:** Shows a MySQL connection with the SQL query `show tables`. The result table lists MySQL tables:

	Tables_in_mysql
1	columns_priv
2	db
3	func
4	host
5	person

- Editor DB2INST1@sample:1:** Shows a DB2 connection with the SQL query `select * from sales`. The result table lists sales data:

	SALES_DATE	SALES_PERSON	REGION
1	1995-12-31	LUCCHESI	Ontario
2	1995-12-31	LEE	Ontario
3	1995-12-31	LEE	Quebec
4	1995-12-31	LEE	Manitob
5	1995-12-31	GOUNOT	Quebec

[Home](#)



Oracle Applications

- [Oracle Applications](#)

System administrators and DBA for Oracle Applications 11.0. and 11.5 can monitor user activity and kill sessions more elegantly than ever before by choosing from the Oracle Applications menu the kill session window.

Oracle Applications

The screenshot shows the DB Radar 1.1 application window. The title bar reads "DB Radar 1.1 - [Kill session for Oracle Applications SYSTEM@prod:3]". The menu bar includes File, Edit, View, Database, Worksheet, Oracle Applications, Window, and Help. The current connection is set to "DB2INST1".

A dialog box titled "Kill session for Oracle Application" is open, displaying the following information:

- System administrators and DBA for Oracle Applications Business Suite 11.0. and 11.5 can monitor user activity and kill sessions
- Data: 02/02/2004
- Time: 0
- IP: 192.168.32.114
- USER_NAME: []
- PID: 176

The dialog box also contains a "Kill Session" button and an "Update" button. In the background, a table titled "Current open sessions on Oracle Applications" is visible, showing the following data:

	Form	Login name	Time	Pid
1	Invoice Workbench	appaprod	2:43	126
2	Run Reports	appaprod	0:02	129
3	F4 Payroll and Assignment Processes	appaprod	0:00	120
4	Payment Workbench	appaprod	0:38	110
5	NULL	appatest	11089:02	169
6	Receipts	appaprod	0:19	176
7	Run Reports	appaprod	4:33	138
8	Invoice Workbench	appaprod	0:34	36
9	Combined Person & Assignment Form	appaprod	4:20	131
10	NULL	appaprod	4197:02	12
11	Combined Person & Assignment Form	appaprod	1:20	148
12	Run Reports	appaprod	0:23	115
13	Enter Requisitions GUI	appaprod	6:05	152
14	NULL	appaprod	7659:58	142
15	Enter Requisitions GUI	appaprod	0:06	172

The table also includes a "Refresh" button on the right side.



[Home](#)



Debug

- [Debug](#)

Users can debug any Oracle statements and step in, out, and over the code.

Debug

DB Radar 1.1 - [Debugger SCOTT@prod:1]

File Edit View Database Debugger Oracle Applications Window Help

Current connection SCOTT@p

My objects

Objects

- SCOTT.SMP_VAB_SA
- SCOTT.SMP_VAR_LIS
- Sequences
 - SCOTT.BLOBID
 - SCOTT.EVT_NOTIFY_
 - SCOTT.EVT_OPERAT
 - SCOTT.EVT_PROFILE
 - SCOTT.SMACTUALPF
 - SCOTT.SMAGENTJOE
 - SCOTT.SMARCHIVES
 - SCOTT.SMCONSOLE:
 - SCOTT.SMDATABASE
 - SCOTT.SMDBAUTHSE
 - SCOTT.SMDEFAULTS
 - SCOTT.SMDISTRIBUTI
 - SCOTT.SMFOLDERSE
 - SCOTT.SMFORMALP.
 - SCOTT.SMGLOBALC
 - SCOTT.SMHOSTAUTH
 - SCOTT.SMHOSTSEGI
 - SCOTT.SMINSTALLA1
 - SCOTT.SMLOGMESS.
 - SCOTT.SMMONTHLY

Sql Editor

```
--debugging PL/SQL code
declare
  i number := 100;
  n number;
begin
  select count(*) into n from all_tables;
  i := add_two(i);
  if i < n then
    i := n + i;
  end if;
end;
```

Watches

	Variable	Value	Exception
1	i	102	SUCCESS
2	n	161	SUCCESS

Ln:9, Col:1

Ready.

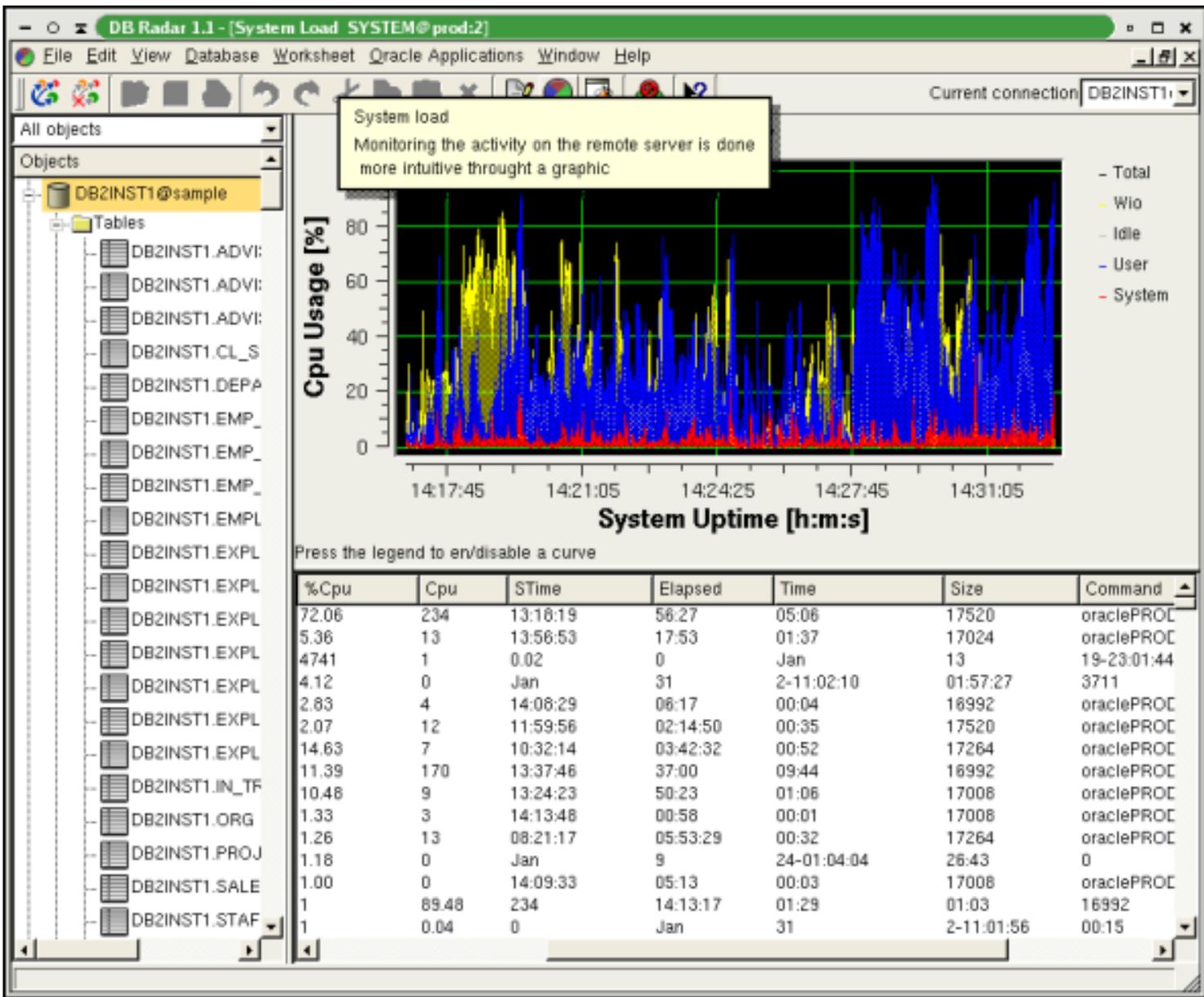


System load

From the Database menu the user can open a window with the system load from the server. This way monitoring the activity on the remote server is done more easily. The remote processes are also shown and by clicking on the lists header the user can sort those processes on every aspect of them. For example by clicking on %CPU we can see which process is consuming the most CPU.

The following information is provided:

- wait for input output
- idle
- user load
- system load



[Home](#)



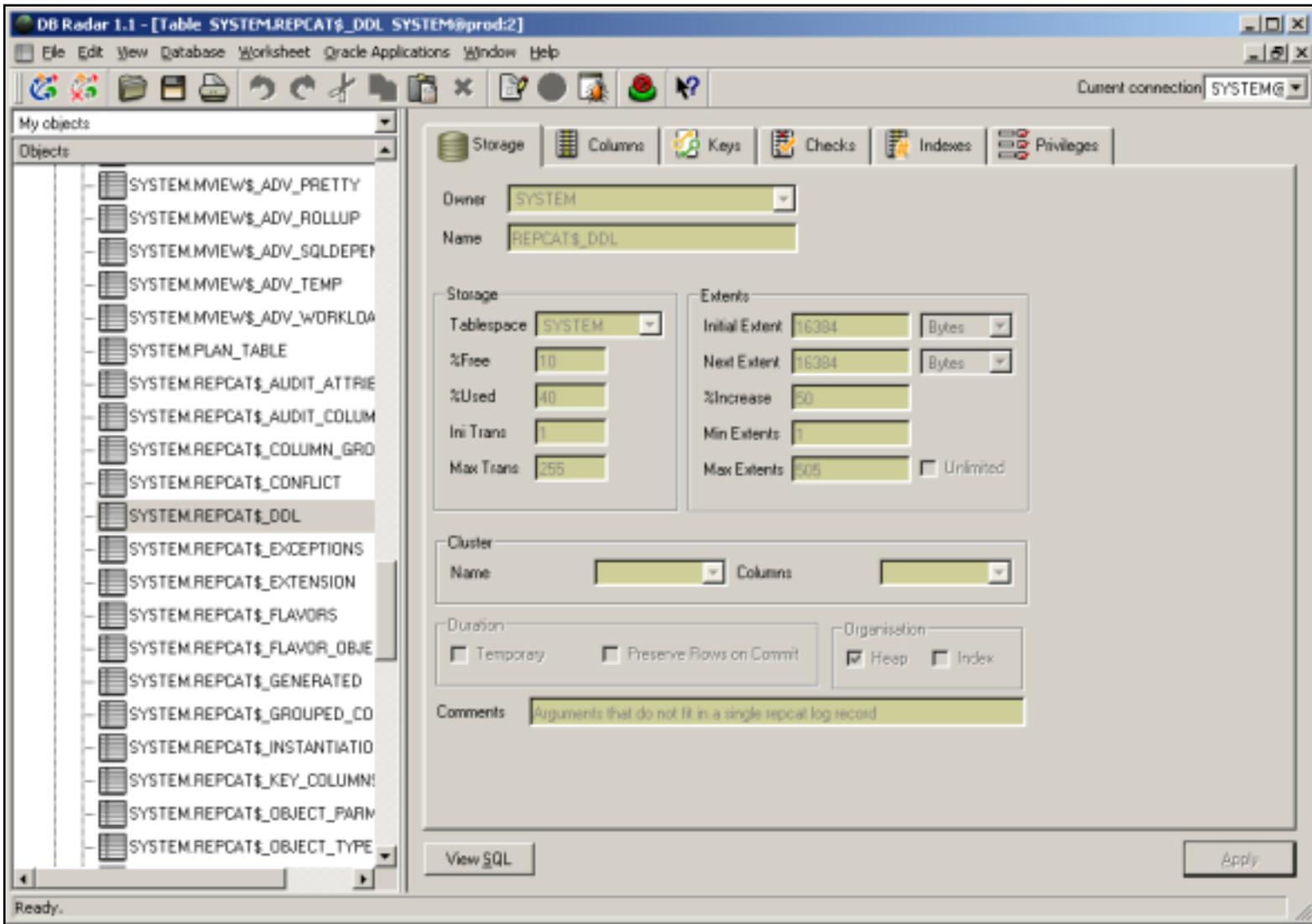
Database Objects

dbRadar supports the following operations on database objects: create, edit, browse, drop, compile : (Currently only for Oracle)

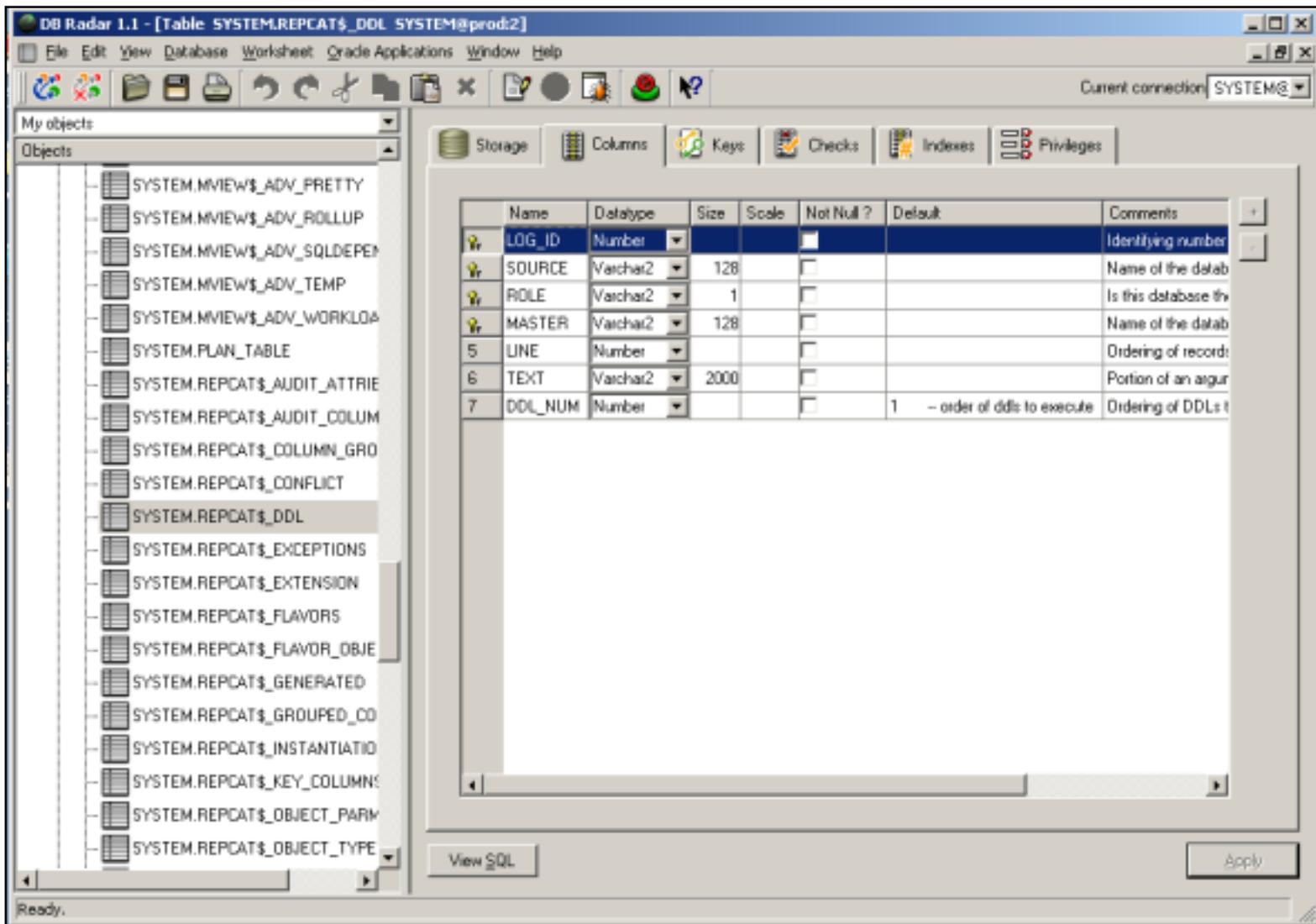
- [New](#)
- [Edit](#)
- [Browse](#)
- [Drop](#)
- [Compile](#)

Users can create, edit, drop, browse and compile tables, views, sequences, functions, procedures, packages, triggers and types.

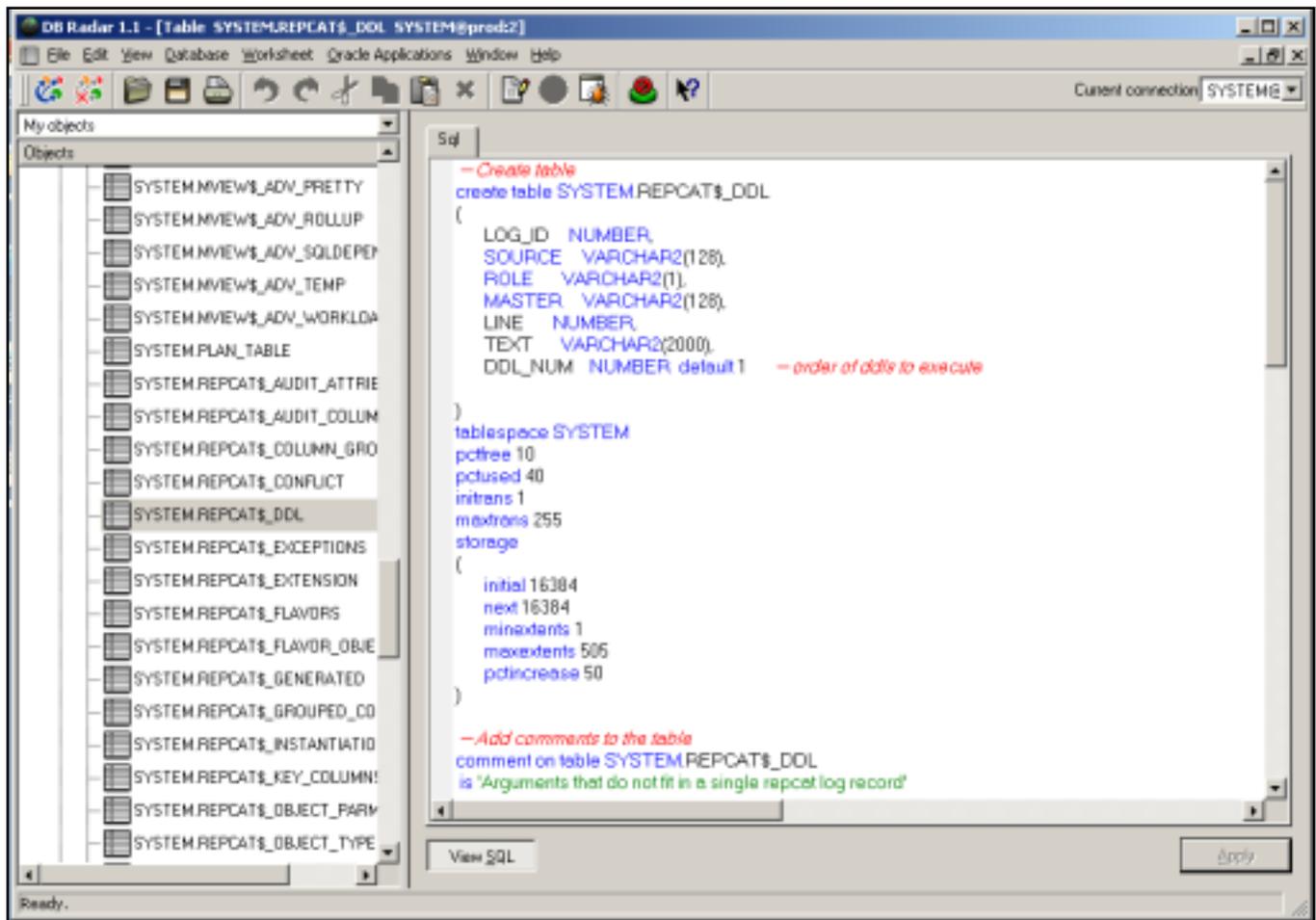
Objects



All the information available for a table is retrieved from the database: storage, columns, keys, indexes, checks, privileges.



You can easily see all the DDL for every database object.



[Home](#)



F.A.Q

- [dbRadar for Linux doesn't load Oracle plugin:](#)

dbRadar for Linux doesn't load Oracle plugin:

Check if the oracle plugin can load all of its dynamic libraries. For oracle 8i & 9i the client libraries are:

- > cd dbRadar-1.1.0-version/database/plugins/sqldrivers
- > ldd libqsqloci.so Notice if you have a "not found" library.

Please check this path \$ORACLE_HOME/lib to be included in /etc/ld.so.conf

In /etc/ld.so.conf you should have something like: /home/oracle/produc/8i/lib where the above is your \$ORACLE_HOME/lib path. After setting this path please run ldconfig as root.

Please make sure that the user who is using dbRadar has access to read into \$ORACLE_HOME/lib.

Please check with ldd \$ORACLE_HOME/bin/sqlplus to see if there are any missing libraries.

[Home](#)