

Module of the subsystem “DB” <PostgreSQL>

<i>Module:</i>	PostgreSQL
<i>Name:</i>	DB PostgreSQL
<i>Type:</i>	DB
<i>Source:</i>	bd_PostgreSQL.so
<i>Version:</i>	0.9.2
<i>Author:</i>	Maxim Lysenko
<i>Translated:</i>	Maxim Lysenko
<i>Description:</i>	DB module. It provides the support for DB PostgreSQL.
<i>License:</i>	GPL

Contents table

Module of the subsystem “DB” <PostgreSQL>	1
Introduction	1
1. Operations over the database	1
2. Operations over the table	2
3. Operations over the contents of the table	2
4. Access rights	3
5. Productivity of DB	3

Introduction

Module <PostgreSQL> gives to the system OpenSCADA support of DB PostgreSQL. PostgreSQL database is a powerful multi-platform database available for free license. Manufacturer of PostgreSQL database is the PostgreSQL Global Development Group <http://www.postgresql.org>. The module is based on the library with API of the manufacturer of DB PostgreSQL. The module allows you to perform operations over databases, tables and contents of tables.

1. Operations over the database

The operations of opening and closing of the database are supported, with the possibility of creating a new database when you try to open one and delete the existing at the close. In terms of the subsystem "DB" of system OpenSCADA opening of DB is its registration for further using of it in the system. It also supported the operation of requesting the list of tables in the database.

DB PostgreSQL address by string of following type:

[<host>;<hostaddr>;<user>;<pass>;<bd>;<port>;<connect_timeout>]. Where:

- *host* - the name of host to connect to. If this begins with a slash, it specifies Unix-domain communication rather than TCP/IP communication; the value is the name of the directory in which the socket file is stored.
- *hostaddr* - Numeric IP address of host to connect to. This should be in the standard IPv4 address format, e.g., 172.28.40.9. If your machine supports IPv6, you can also use those addresses. TCP/IP communication is always used when a nonempty string is specified for this parameter.
- *user* - the name of the user of database;
- *pass* - user password to access the database;

- *bd* - the name of the database;
- *port* - port to listen to by the database server (default is 5432);
- *connect_timeout* - maximum wait for connection, in seconds. Zero or not specified means wait indefinitely. It is not recommended to use a timeout of less than 2 seconds.

In the case of local access to the database in the same host the address string should be as follows: `[;;roman;123456;OpenSCADA;;10]`

In the case of remote access to the database you must use the address and port of the server of the database. For example: `[server.nm.org;;roman;123456;OpenSCADA;;10]`

2. Operations over the table

The operations of opening and closing of the table with the possibility of creating a new table when you open and deleting the existing one at the closing, and also the operation of the requesting of the table's structure are supported.

3. Operations over the contents of the table

- scanning of the records of the table;
- request the values of these records;
- setting the values of these records;
- removing the records.

API of subsystem “DB” suppose the access to the contents of the table on the value of key(s) fields. Thus, the operation of request of the record implies the preset of key columns of the object TConfig, which will fulfill the request. Creating a new record(string) is the installation of the values of record, which does not exist.

The module allows you to dynamically change the structure of the PostgreSQL database tables. Thus, in the event of a discrepancy of the table and the structure determined by record, the structure of the table will be set to the required structure of record. In the case of the request of the value of the record, and mismatching of the structures of record and the table there will be available only to the values of common elements of the record and table. The module does not track the order of the elements in the record and in the structure of the table!

The module provides the support of multilanguage text variables. For fields with multilanguage text variables the columns of the appropriate language are created in format `<lang>#<FldID> (en#NAME)`. In this time the base column contain value for base language. The columns of other languages are created by needs, at the time of saving to DB and execution OpenSCADA with appropriate language. In the case of the value's absence for the language it will be used the values for basic language.

The types of the elements of DB PostgreSQL correspond to types of elements of system OpenSCADA in the following way:

The types of fields of the system OpenSCADA	Types of fields of DB PostgreSQL
TFld::String	character(n), character varying(n), text
TFld::Integer	integer, bigint, timestamp with time zone [for the fields with the flag TFld::DateTimeDec]
TFld::Real	double precision
TFld::Boolean	smallint

4. Access rights

PostgreSQL database contains some mechanism of separation of access, which is to specify the user privileges for database. The table below lists the necessary privileges for the work in the OpenSCADA.

Operation	SQL-commands
Creation of the DB	CREATEDB
Creation of the connection	LOGIN

Briefly we will look at the initial configuration of the PostgreSQL server to connect to it using this module:

- Installing the PostgreSQL database server as a package or building.
- Primary server initialization:
 - # DB initialization
 - \$ service postgresql initdb
 - # DB start
 - \$ service postgresql start
- Lets allow trusted access from the local subnet or desired one by editing the file /var/lib/pgsql/data/pg_hba.conf setting the 'trust':

local	all	all		trust
host	all	all	127.0.0.1/32	trust
- Restart the server after you edit the access rights:
 - \$ service postgresql restart
- Set the password for the system user 'postgres':
 - \$ psql -U postgres -d template1 -c "ALTER USER postgres PASSWORD '123456'"
- Connect to the database server by using this module by entering the database address:
"localhost;postgres;123456;test"

5. Productivity of DB

Measurement of productivity of DB were carried out by the test "DB" of the module of system tests "SystemTests", by performing operations over the records of the structure: <name char (20), descr char (50), val double (10.2), id int (7), stat bool>. OpenSCADA was launched with the demo configuration.

Operation	K8-3000+, 384M, 120G, PostgreSQL 8.3 (local)	PostgreSQL 8.3 (remote)	Nokia N800, PostgreSQL 8.3 (remote)
<i>Creation of the 1000 records (sec.):</i>	0.89	1.04	5
<i>Updating of the 1000 records (sec.):</i>	1.02	1.1	4.8
<i>Getting of the 1000 records (sec.):</i>	0.61	0.63	2.96
<i>Deleting of the 1000 record (sec.):</i>	0.36	0.4	1.73