15 ODBC monitoring

Overview

ODBC monitoring corresponds to the *Database monitor* item type in the Zabbix frontend.

ODBC is a C programming language middle-ware API for accessing database management systems (DBMS). The ODBC concept was developed by Microsoft and later ported to other platforms.

Zabbix may query any database, which is supported by ODBC. To do that, Zabbix does not directly connect to the databases, but uses the ODBC interface and drivers set up in ODBC. This function allows for more efficient monitoring of different databases for multiple purposes - for example, checking specific database queues, usage statistics and so on. Zabbix supports unixODBC, which is one of the most commonly used open source ODBC API implementations.

Installing unixODBC

The suggested way of installing unixODBC is to use the Linux operating system default package repositories. In the most popular Linux distributions unixODBC is included in the package repository by default. If it's not available, it can be obtained at the unixODBC homepage: http://www.unixodbc.org/download.html.

Installing unixODBC on RedHat/Fedora based systems using the *yum* package manager:

```
shell> yum -y install unixODBC unixODBC-devel
```

Installing unixODBC on SUSE based systems using the *zypper* package manager:

```
# zypper in unixODBC-devel
```

The unixODBC-devel package is needed to compile Zabbix with unixODBC support.

Installing unixODBC drivers

A unixODBC database driver should be installed for the database, which will be monitored. unixODBC has a list of supported databases and drivers: http://www.unixodbc.org/drivers.html. In some Linux distributions database drivers are included in package repositories. Installing MySQL database driver on RedHat/Fedora based systems using the *yum* package manager:

```
shell> yum install mysql-connector-odbc
```

Installing MySQL database driver on SUSE based systems using the *zypper* package manager:

```
zypper in MyODBC-unixODBC
```

Configuring unixODBC

ODBC configuration is done by editing the **odbcinst.ini** and **odbc.ini** files. To verify the configuration file location, type:

```
shell> odbcinst -j
```

odbcinst.ini is used to list the installed ODBC database drivers:

```
[mysql]
Description = ODBC for MySQL
Driver      = /usr/lib/libmyodbc5.so
```

Parameter details:

Attribute	Description
mysql	Database driver name.
Description	Database driver description.
Driver	Database driver library location.

odbc.ini is used to define data sources:

```
[test]
Description = MySQL test database
Driver      = mysql
Server      = 127.0.0.1
User        = root
Password    =
Port        = 3306
Database    = zabbix
```

Parameter details:

Attribute	Description
test	Data source name (DSN).
Description	Data source description.
Driver	Database driver name - as specified in odbcinst.ini
Server	Database server IP/DNS.
User	Database user for connection.
Password	Database user password.
Port	Database connection port.
Database	Database name.

To verify if ODBC connection is working successfully, a connection to database should be tested. That can be done with the **isql** utility (included in the unixODBC package):

```
shell> isql test
+---------------------------------------+
| Connected!                            |
|                                       |
| sql-statement                         |
| help [tablename]                      |
| quit                                  |
|                                       |
+---------------------------------------+
SQL>
```

Compiling Zabbix with ODBC support

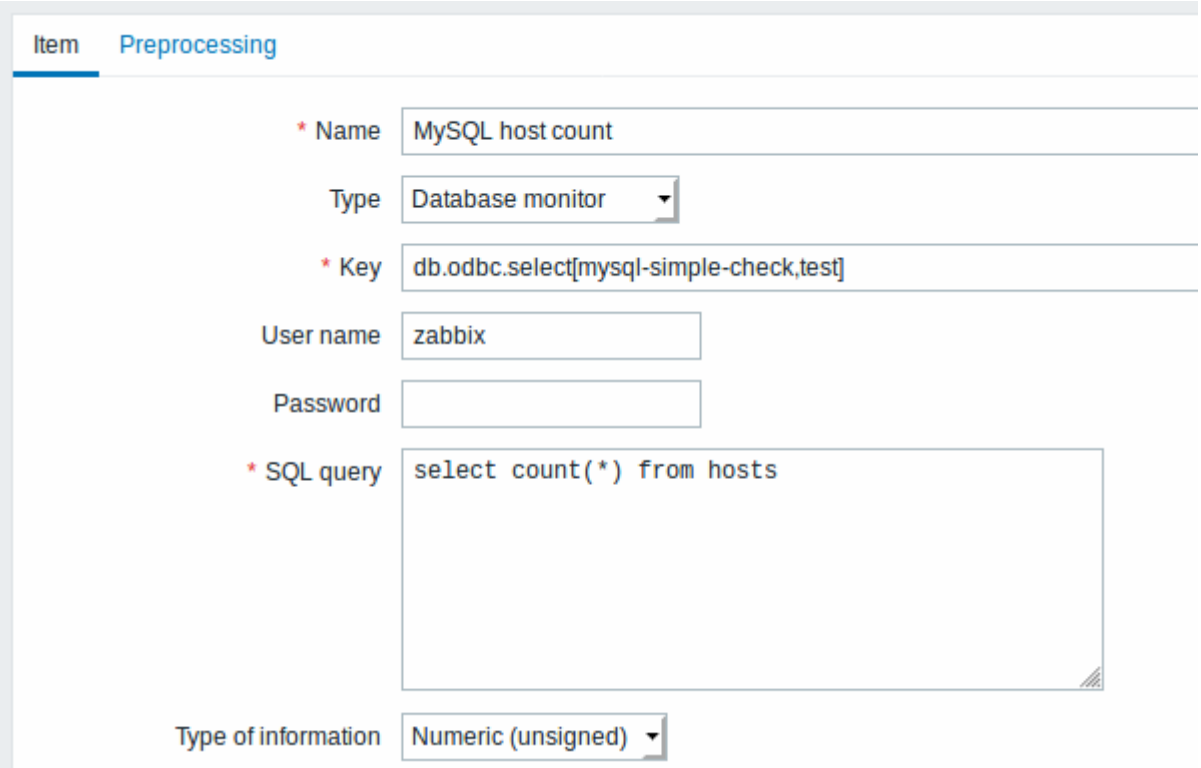
To enable ODBC support, Zabbix should be compiled with the following flag:

```
  --with-unixodbc[=ARG]   use odbc driver against unixODBC package
```

See more about Zabbix installation from the source code.

Item configuration in Zabbix frontend

Configure a database monitoring item.



All mandatory input fields are marked with a red asterisk.

Specifically for database monitoring items you must enter:

Type	Select *Database monitor* here.

Key	Enter one of the two supported item keys: **db.odbc.select**[unique_description,data_source_name] - this item is designed to return one value, i.e. the first column of the first row of the SQL query result. If a query returns more than one column, only the first column is read. If a query returns more than one line, only the first line is read. **db.odbc.get**[unique_description,data_source_name] - this item is capable of returning multiple rows/columns in JSON format. Thus it may be used as a master item that collects all data in one system call, while JSONPath preprocessing may be used in dependent items to extract individual values. For more information, see an example of the returned format, used in low-level discovery. This item is supported since Zabbix 4.4. The unique description will serve to identify the item in triggers etc. The data source name (DSN) must be set as specified in odbc.ini.
User name	Enter the database user name (optional if user is specified in odbc.ini)
Password	Enter the database user password (optional if password is specified in odbc.ini)
SQL query	Enter the SQL query. Note that with the `db.odbc.select[]` item the query must return one value only.
Type of information	It is important to know what type of information will be returned by the query, so that it is selected correctly here. With an incorrect *type of information* the item will turn unsupported.

Important notes

- Zabbix does not limit the query execution time. It is up to the user to choose queries that can be executed in a reasonable amount of time.
- The Timeout parameter value from Zabbix server is used as the ODBC login timeout (note that depending on ODBC drivers the login timeout setting might be ignored).
- The SQL command must return a result set like any query with `select …`. The query syntax will depend on the RDBMS which will process them. The syntax of request to a storage procedure must be started with `call` keyword.
- See also known issues for ODBC checks

Error messages

ODBC error messages are structured into fields to provide detailed information. For example:

```
Cannot execute ODBC query: [SQL_ERROR]:[42601][7][ERROR: syntax error at or
near ";"; Error while executing the query]
└───────────┬───────────┘  └────┬────┘
└──┬──┘ └┬┘ └──────────────────────────────┬──────────────────────────────
──┘
            │                   │         │    └─ Native error code
└─ Native error message
            │                   │         └─ SQLState
            └─ Zabbix message   └─ ODBC return code
```

Note that the error message length is limited to 2048 bytes, so the message can be truncated. If there is more than one ODBC diagnostic record Zabbix tries to concatenate them (separated with `|`) as far as the length limit allows.