

## 15 Monitoração de Banco de dados (ODBC)

### 15.1 Visão geral

O ODBC é uma API 'middle-ware' escrita em C para o acesso a sistemas gerenciadores de bancos de dados (DBMS). O conceito do ODBC foi desenvolvido pela Microsoft e portado para outras plataformas.

O Zabbix pode consultar qualquer banco de dados, bastando para isso que exista o suporte através do driver ODBC. É importante ressaltar que o Zabbix não acessa diretamente os SGDBs, todo o acesso é feito através das interfaces e drivers ODBC. Esta funcionalidade permite uma monitoração mais eficiente de bancos de dados para diversos propósitos - por exemplo, verificar filas no banco de dados, estatísticas, quantidade de registros, etc. O Zabbix suporta o unixODBC, que uma das implementações mais comuns da API ODBC.

### 15.2 Instalando o unixODBC

A forma que sugerimos a instalação do unixODBC é através do gerenciador de pacotes do Linux. A maioria das distribuições Linux tem este pacote em seu repositório padrão. Se ele não estiver disponível em sua distribuição, poderá ser obtido na página do unixODBC:

<http://www.unixodbc.org/download.html>.

Para instalar o unixODBC com o *yum* (ambientes baseados em RedHat/Centos/Fedora):

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

Para instalar o unixODBC com o *zypper* (SUSE):

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

O pacote 'unixODBC-devel' é necessário para compilar o Zabbix com o suporte ao unixODBC.

### 15.3 Instalando os drivers unixODBC

Para cada banco de dados, além do unixODBC, deverá ser instalado o driver apropriado. O unixODBC possui uma lista de bancos de dados e drivers suportados: <http://www.unixodbc.org/drivers.html>. Em algumas distribuições os drivers ODBC são instalados em conjunto com o unixODBC. O comando a seguir instala o driver MySQL através do *yum*:

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

A mesma instalação pode ser feita em um ambiente SUSE com o comando a seguir:

```
shell> zypper install MyODBC-unixODBC
```

## 15.4 Configurando o unixODBC

A configuração do ODBC é feita através dos arquivos **odbcinst.ini** e **odbc.ini**. Para verificar a localização dos mesmos pode ser utilizado o comando a seguir:

```
shell> odbcinst -j
```

O arquivo **odbcinst.ini** é utilizado para listar os drivers ODBC instalados:

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

Detalhes dos parâmetros:

Atributo	Descrição
<i>mysql</i>	Nome do driver de BD.
<i>Description</i>	Descrição do driver de BD.
<i>Driver</i>	Localização do driver de BD.

O arquivo **odbc.ini** é utilizado para definir as fontes de dados:

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

Detalhes dos parâmetros:

Atributo	Descrição
<i>test</i>	Nome da fonte de dados (DSN).
<i>Description</i>	Descrição da fonte de dados.
<i>Driver</i>	Nome do driver ODBC, conforme definido no <code>odbcinst.ini</code>
<i>Server</i>	Endereço/DNS do servidor de BD.
<i>User</i>	Usuário para conexão.
<i>Password</i>	Senha de conexão.
<i>Port</i>	Porta do BD.
<i>Database</i>	Nome da base de dados.

Para verificar se a conexão ODBC está funcional, é possível um teste via linha de comando verificando se é possível se conectar ao banco de dados. Isso pode ser feito através do utilitário **isql**

(incluso no pacote unixODBC):

```

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

```

### 15.5 Compilando o Zabbix com o suporte ao ODBC

Para habilitar o suporte ao ODBC, o Zabbix precisa ser compilado com o parâmetro abaixo:

```

--with-unixodbc[=ARG]    utilizar o drive ODBC ao invés do pacote unixODBC

```

Leia mais sobre a instalação do Zabbix em [instalação a partir dos fontes](#).

### 15.6 Configuração do item no frontend do Zabbix

Configure um monitoramento de banco de dados [item](#):

\* Name

Type

\* Key

User name

Password

\* SQL query

Type of information

Para itens de monitoração de banco de dados, os dados abaixo deve ser fornecidos:

<i>Tipo</i>	Selecione <i>Monitoração de banco de dados</i> .
<i>Chave</i>	Informe <b>db.odbc.select</b> [identificação única, ao nível de nome da fonte de dados] A identificação única serve para identificar o item em triggers e outros locais. A fonte de dados (DSN) precisa ser definida no arquivo odbc.ini.

<i>Nome de usuário</i>	Informe o nome de usuário para acesso ao BD (opcional se o usuário já estiver definido em odbc.ini)
<i>Senha</i>	Informe a senha para o banco de dados (opcional se a senha já for definida em odbc.ini)
<i>Consulta SQL</i>	Informe a consulta
<i>Tipo da informação</i>	É muito importante saber qual tipo de informação será retornada pela consulta, pois um tipo incorreto de dados nesta configuração poderá fazer com que o item passe para o estado 'não suportado'.

## 15.7 Observações Importantes

- A consulta não pode demorar mais do que o tempo definido no parâmetro [Timeout](#) do arquivo de configuração do Zabbix Server. A partir do Zabbix 2.0.8 o parâmetro [Timeout](#) também pode ser utilizado para limitar o tempo para a autenticação ODBC (observe que dependendo do driver ODBC o 'timeout' de autenticação poderá ser ignorado).
- A consulta precisa retornar somente um valor.
  - Se a consulta retornar mais de uma coluna, apenas a primeira será lida.
  - Se a consulta retornar mais de uma linha, apenas a primeira linha será lida.
- O comando SQL deverá iniciar com `select`.
- O comando SQL não pode conter quebra de linha.

Consulte [problemas conhecidos](#) para verificações ODBC.

## 15.8 Mensagens de erro

A partir do Zabbix 2.0.8 as mensagens de erro ODBC foram estruturadas em campo para prover maior detalhamento de informações. Exemplo:

```
Cannot execute ODBC query:[SQL_ERROR]:[42601][7][ERROR: syntax error at or
near ";"; Error while executing the query]
-----|-----
|
|- Mensagem de erro. | - Código de erro nativo
| | - Separador de registros
| | -SQLState
|- Mensagem Zabbix | - Código de retorno ODBC
```

Observe que o tamanho da mensagem de erro é limitado a 2048 bytes, se a mensagem for maior ela será truncada. Se existir mais de um diagnóstico ODBC o Zabbix tentará concatenar os valores até o tamanho máximo permitido.

From: <https://www.zabbix.com/documentation/4.4/> - **Zabbix Documentation 4.4**

Permanent link: [https://www.zabbix.com/documentation/4.4/pt/manual/config/items/itemtypes/odbc\\_checks](https://www.zabbix.com/documentation/4.4/pt/manual/config/items/itemtypes/odbc_checks)

Last update: **2019/04/02 06:04**

