DATE	REVISION	COMMENTS
2009 → 2010	V1 → V4	Draft versions
Sept 2011	V5	Oracle, Mysql & MSSQL monitoring for Zabbix versions 1.8
Sept 2012	V6	Oracle, Mysql, MSSQL, Postgresql & DB2 monitoring for Zabbix versions 2.0
July 2014	V7	Oracle, Mysql, MSSQL, Postgresql & DB2 monitoring for Zabbix versions 2.2. Low Level Discovery for Instances.

You are welcome for any comment, request or modification. Please, send back your feedbacks to me alain@coreit.fr

Table of Contents

1. OVERVIEW

1.1. Design

The Zabbix Database Monitoring is based on:

- php scripts that query the databases via the database client installed on the Zabbix server or proxy, items are mainly defined as « external check »
- Zabbix Templates that get output values from php scripts
- Zabbix Macros that are used to configure Alarm Levels, some are configured by default at Template level, some have to be set at host level.
- Zabbix Macros that localize the Databases
- And only for MSSQL, zabbix templates that monitor Database Server parameters directly on the agent for MSSQL Server (mainly based on performance counters)
- And since V7, a json.php script that discovers Database Instances

1.2. Versions

The Zabbix Database Monitoring Version 7 is synchronized with Zabbix V2.2.x and is able to monitor Oracle (9, 10, 11), mysql (4, 5), mssql (MS SQL Server 2000, 2005, 2008 versions FR or US on Windows Servers FR or US), postgresql (8.x, 9.x) & DB2 (8.x, 9.x).

The Zabbix Database Monitoring Version 7 is also synchronized with :

- zabora.php version 3.0
- zabmy.php version 3.0
- zabpgsql.php version 3.0
- zabsql.php version 3.0
- zabdb2.php version 3.0

The Database Monitoring Version 7 is **NOT SUPPORTED** on Zabbix 1.8.x. Any feedback on other Database versions is appreciated. For Zabbix 1.8.x please prefer the Zabbix Database Monitoring Version 5.

	Php scripts	**Zabbix Templates**	Zabbix	Oracle	MSSQL	MYSQL	Postgresql	DB2
V5 ->	**V1.3**	**V1**	V1.8.x	V9 V10 V11	2000 2005 2008	V4 V5	Not supported	Not supported
V6 ->	**V2.0**	**V6**	V2.0.x	V9 V10 V11	2000 2005 2008	V4 V5	V8 V9	V8 V9
V7 ->	**V3.0**	**V7**	V2.0.x V2.2.x	V9 V10 V11	2000 2005 2008	V4 V5	V8 V9	V8 V9

Notes :

- Grey Backgrounds are Zabbix Database Monitoring components
- It is not a big effort to migrate Zabbix Database Monitoring V6 on Zabbix 1.8 or Postgresql/DB2 on Zabbix Database Monitoring V5, but it was not done.

1.3. Templates

Templates will be loaded in the group **Templates_V7**, they are generic and need to be customized for your database instances, that will be done automatically by script json.php that uses Zabbix Low Level Discovery to feed the templates with new items, triggers & graphs.

Templates items are mainly of type « External Check ».

For example, the one that measures MYSQL database index:
zabmy.php[database_index_size,{#MYSQL_INSTANCE_NAME},{$MYSQL_DATABASE_NAME}]

Note :

- In Zabbix langage {#MYSQL_INSTANCE_NAME} is a Low Level Discovery Macro (begins with #), it means that it will be replaced with the name of different Database instances.
- {$MYSQL_DATABASE_NAME} is a User Macro (begins with $), you have to configure this one with a name of a database preferably at host level.

1.4. Php scripts

1.4.1. Database monitoring scripts

Php scripts normally need no customization except the path of the configuration file for Instance parameters. Please configure directly in the script the parameter $INSTANCENAMES according to the location of this configuration file.

For example, the default one from zabmy.php
$INSTANCENAMES = "/usr/local/zabbix/etc/zabmy.conf" ;

Reminder : *ExternalScripts* parameter must be configured in your zabbix_server.conf or zabbix_proxy.conf.

For example, below is what I usually configure:
ExternalScripts=/usr/local/zabbix/etc/externalscripts

All php scripts require an user to be created in the database with appropriate rights (described in next paragraphs), off course this user is used to login onto the database and to query appropriate database tables.

1.4.2.Json.php script

This script is new with V7, it feeds the database templates with new discovered items, triggers and graphs ; this is done by Low Level Discovery mechanism introduced in Zabbix version 2.2.

1.5. Macros

Three kinds of Zabbix macros are configured :

- Macros at Template level that generally configure a default threshold for all hosts that inherite the template. For example, {$ORACLE_MIN_BUFFER_CACHE_HIT_RATIO} is the macro for the Oracle default Buffer cache hit ratio threshold.
- Macros at Host level that could overwrite the default template threshold, they have the same name as the one used in the template. For example, {$ORACLE_MIN_BUFFER_CACHE_HIT_RATIO} could be configured at host level with a different threshold that is customized for a specific host.
- Macros at Host level that localize the Database instance & Databases on the DB server. For example, {#ORACLE_INSTANCE_NAME} is the macro that configures the INSTANCE_NAME, unique entry in zabora.conf ; starting from version 7, this is done automatically by Low Level Discovery process.

Reminder: Zabbix takes into account by priority : Host level macro, then Template level macro and at last Global level macro.

1.6. Demo

To have a demo of this solution, please go to CORE IT Demo Site

1.7. Installation steps

1. Install prerequisites on your Zabbix server or proxy, ie php libraries for database connections & php client. We recommend also to install database client for debugging purpose. Oracle needs also ZabbixAPI.class.php from Andrew Farley, fetch it on the Net, Zabbix 1.8 API – PHP Class v1.0 is OK with Zabbix 2.0 and 2.2 (needed only if you want to auto-configure alertlog path, but you can configure it manually also with the macro {$<YOUR_INSTANCE_NAME>_ORA_ALERTLOGPATH} at host level.
2. Install php scripts in the directory configured with *ExternalScripts* parameter defined in your zabbix_server.conf or zabbix_proxy.conf. Scripts are :
    - zabxxxx.php Database monitoring scripts

- json.php Discovery script for all Databases

3. Import templates, they will be loaded in a new group called Templates_V7
4. Assign templates to your hosts
5. Customize macros according next paragraphs
    1. Template level macros define default thresholds
    2. Host level macros define Databases and customized thresholds

2. Database Instances Low Level Discovery

You have to install the json.php script in the externalscripts location of your Zabbix Installation, it is common for all Database scripts.

For example, below is what I usually configure:
ExternalScripts=/usr/local/zabbix/etc/externalscripts

This script will be launched by the Discovery rules located in Database Templates. It will feed templates with multiple dynamic items, triggers and graphs. Except for MSSQL, you will find no static items.

In installation mode, it could be a good idea to configure a low delay for discovery rule, the default configured one is 86400 (1 day), it means you could wait one day before seing any dynamic items. And when all is running fine, rollback to 86400 for performance considerations (Discoveries are Zabbix ressources big consummers).

In the json.php script, you have to configure the location of your Databases Instances list file.

For example, for MSSQL :
$file_conf = "/usr/local/zabbix/etc/zabsql.conf";

json.php script will read all the Instances definition in this file and automatically create the items accordingly.

3. ORACLE

Oracle monitoring is based on:

- the script zabora.php
- the template T_ORACLE-LLD

3.1. Prerequisites

- Oracle client on the Zabbix server or proxy (recommended for debugging purpose)
- php-cli, php-oci8 and ZabbixAPI.class.php for alertlogpath check
- File zabora.conf contains Oracle Instance Infos and must be readable by zabbix user

3.2. /usr/local/zabbix/etc/zabora.conf

```
[global]
        apiurl = https://localhost:1443/        # 3 parameters needed for alertlogpath
        apiuser = apiuser                       # autoconfiguration in Zabbix. Apiuser must
        apipass = xxxxxxxxxx                    # have Zabbix admin rights on oracle server.

        default_user = xxxxxx                   # default user who queries dictionary
        default_password = xxxxxx

[ORACLE10G_XE]
        host = xxx.xxx.xxx.xxx
        port = 1521
        service = XE
        password = manager
        username = system                       # specific user per instance, if not configured
                                                # default_user is used
```

CAUTION : Alertlogpath item will be supported only if the entry in zabora.conf is upper case, *[ORACLE10G_XE] is supported, [Oracle10g_XE] is not supported. The reason comes from Zabbix macros that are supported only with upper case and no special characters.*

For Oracle10g
connectstring=[//]host_name[:port][/service_name]

For Oracle11g
connectstring=[//]host_name[:port][/service_name][:server_type] [/instance_name]

You can get service info on the oracle server by the *lsnrctl status* command.

With Version 7, /etc/tnsnames.ora is no more mandatory except if you want to connect to Oracle Database through the Oracle Client. Generally, this is something like that:

/etc/tnsnames.ora

```
TNS-NAME =
 (DESCRIPTION =
  (ADDRESS_LIST =
   (ADDRESS = (PROTOCOL = TCP)(HOST = server_IP_or_Name)(PORT = 1521))
  )
 (CONNECT_DATA =
  (SERVICE_NAME = service_name)
 )
)
```

3.3. Templates

3.3.1. T_ORACLE-LLD

Template level Macros (inherited at host level)

- {$ORACLE_MIN_BUFFER_CACHE_HIT_RATIO} → Buffer cache hit ratio below alarm level
- {$ORACLE_MIN_FREE_EXTENTS} → Segments with less than alarm level free extents
- {$ORACLE_MIN_FREE_INCREMENTS} → Autoextensible datafile with less alarm level free increments
- {$ORACLE_MIN_LIBRARY_CACHE_HIT_RATIO} → Library cache hit ratio below alarm level
- {$ORACLE_MAX_TBS_USED_SPACE} → Non autoextensible tablespaces above alarm level used% space

3.4. Host Configuration

Host level Macros

- {<YOUR_INSTANCE_NAME>_$ORA_ALERTLOGPATH} → path of Instance Alertlog that can be auto-configured if Zabbix API parameters are configured in zabora.conf
- And other template T_ORACLE-LLD macros you want to over-write with host specific values

3.5. Oracle alert_SID.log Monitoring

Monitoring alert_SID.log file is based on several technologies:

- zabora.php with alertlogpath parameter whose purpose is to store into the Zabbix server the alert_SID.log file path. This storage is done as follows:
  - zabora.php requests dictionary and retrieves the file path
  - then stores the value in a host macro ($ <YOUR_INSTANCE_NAME>_ORA_ALERTLOGPATH) via Zabbix APIs

Two items are included in the Oracle template:

- *zabora.php[alertlogpath,{#ORACLE_INSTANCE_NAME}]* that performs the configuration of the macro
- *log[{${#ORACLE_INSTANCE_NAME}_ORA_ALERTLOGPATH},"ORA-",,]* looking for ORA- strings

Notes:

- Caution : Unix zabbix user must have read access to the Oracle alertlog file on the Oracle server.

3.6. ORACLE User rights

```
CREATE USER "ZABBIX" IDENTIFIED BY "password";
GRANT "CONNECT" TO "ZABBIX";
grant select on v_$instance to zabbix;
grant select on v_$sysstat to zabbix;
grant select on v_$session to zabbix;
grant select on dba_free_space to zabbix;
grant select on dba_data_files to zabbix;
grant select on dba_tablespaces to zabbix;
grant select on v_$log to zabbix;
grant select on v_$archived_log to zabbix;
grant select on v_$loghist to zabbix;
grant select on v_$system_event to zabbix;
grant select on v_$event_name to zabbix;
grant select on v_$parameter to zabbix;
grant select on v_$librarycache to zabbix;
```

4. MYSQL

Mysql monitoring is based on:

- the script zabmy.php
- the template T_MYSQL-LLD

4.1. Prerequisites

- Mysql client on the Zabbix server or proxy
- php-cli, php-mysql
- File zabmy.conf contains MYSQL Instance Infos and must be readable by zabbix user

4.2. /usr/local/zabbix/etc/zabmy.conf

```
[global]
        default_user = xxxxxx                   # default user who queries mysql
        default_password = xxxxxx

[INSTANCE_NAME]
        host = xxx.xxx.xxx.xxx
        port = 3306
        username = xxxxxx                       # specific user per instance; if not configured,
        password = xxxxxx                       # default_user is used
```

4.3. Templates

4.3.1. T_MYSQL-LLD

Template that contains Mysql Instance monitoring.

Template level Macros (inherited at host level)

- {$MYSQL_MAX_ALLDB_SIZE} → All databases size above alarm level
- {$MYSQL_MAX_THREADS} → Number of threads above alarm level
- {$MYSQL_MAX_DB_SIZE} → Database size above alarm level

4.4. Host Configuration

Host level Macros

- {$MYSQL_DATABASE_NAME} → Name of monitored database
- And other template T_MYSQL-LLDmacros you want to over-write with host specific values

4.5. MYSQL User rights

*mysql> grant select on *.* to <user>@<host> identified by <passwd>*
*mysql> grant process on *.* to <user>@<host> identified by <passwd>*
mysql> flush privileges;

Note: Do configure properly bind-address parameter in mysql configuration file, usually my.cnf

For example, this configuration enable all mysql requests from outside
 # bind-address = 127.0.0.1

5. MSSQL

MSSQL Monitoring is based on:

- the script zabsql.php
- T_MSSQL-LLD contains items for default MSSQL Instance
- T_MSSQL_NAMED_INSTANCE contains perf counters for the named instance

5.1. Prerequisites

- client mssql (freetds) on the Zabbix server or proxy
- php-cli, php-sybase or php-mssql (depending on Linux distribution)

5.2. /usr/local/zabbix/etc/zabsql.conf

```
[global]
  text_size = 64512              # parameters used by freetds
  tds_version = 8.0

  default_user = xxxxxx          # default user who queries mssql
  default_password = xxxxxx

[INSTANCE_NAME]
  host = xxx.xxx.xxx.xxx
  port = 1435
  username = sa                  # specific user per instance; if not configured,
  password = xxxxxxxx            # default_user is used
```

5.3. Templates

5.3.1. T_MSSQL-LLD

Template that contains MSSQL Default Instance monitoring.

Template level Macros (inherited at host level)

- {$MSSQL_MAX_DBLOG_SIZE} → Database Log size above alarm level
- {$MSSQL_MAX_DB_SIZE} → Database size above alarm level

5.3.2. T_MSSQL_NAMED_INSTANCE

Template that contains MSSQL Named Instance monitoring.

Template level Macros (inherited at host level)

- No defined macros

5.4. Host Configuration

Host level Macros

- {$MSSQL_DATABASE_NAME} → Name of monitored database
- {$MSSQL_NAMED_INSTANCE_NAME}
- And other template T-MSSQL-DATABASE macros you want to over-write with host specific values

5.4.1. Misc

- **Note:** Some counters are related to french Windows or SQL servers, they all are redundant with english counters. On non french servers they will be simply flagged « unsupported » by Zabbix, off course you can delete them.

5.5. MSSQL User rights

```
DECLARE @SQL NVARCHAR(1000);
Declare @login  as Varchar(35);
Declare @user  as Varchar(35);
SET @login = 'zabbix';
SET @user = 'zabbix';
SET NOCOUNT ON

SET @SQL = '
IF "?" NOT IN ("master", "model", "msdb",
"tempdb","pubs","northwind")
BEGIN
EXEC ?.dbo.sp_grantdbaccess "' + @login + '","' + @user + '"
EXEC ?.dbo.sp_addrolemember "db_datareader","' + @user + '"
END '

Exec sp_addlogin @login, 'xxxxxxxx'
EXEC sp_MSForEachDb @sql
```

6. POSTGRESQL

Postgresql monitoring is based on:

- the script zabpgsql.php
- the template T_PGSQL-LLD

6.1. Prerequisites

- Postgresql client on the Zabbix server or proxy
- php-cli, php-pgsql
- File zabpgsl.conf contains POSTGRESQL Instance Infos and must be readable by zabbix user

6.2. /usr/local/zabbix/etc/zabpgsql.conf

```
[global]
        default_user = xxxxxx                   # default user who queries mysql
        default_password = xxxxxx

[INSTANCE_NAME]
        host = xxx.xxx.xxx.xxx
        port = 5432
        username = xxxxxx                       # specific user per instance; if not configured,
        password = xxxxxx                       # default_user is used
```

6.3. Templates

6.3.1. T_PGSQL-LLD

Template that contains Postgresql Instance monitoring.

Template level Macros (inherited at host level)

- {$PGSQL_MAX_DB_SIZE} → Database size above alarm level

6.4. POSTGRESQL User rights

```
CREATE USER zabbix WITH PASSWORD 'password';
GRANT SELECT ON pg_stat_activity to zabbix;
GRANT SELECT ON pg_stat_activity to zabbix;
GRANT SELECT ON pg_database to zabbix;
GRANT SELECT ON pg_authid to zabbix;
GRANT SELECT ON pg_stat_bgwrite r to zabbix;
GRANT SELECT ON pg_locks to zabbix;
GRANT SELECT ON pg_stat_database to zabbix;
```

7. DB2

DB2 monitoring is based on:

- the script zabdb2.php
- the templates T_DB2-LLD

7.1. Prerequisites

- DB2 client on the Zabbix server or proxy
- php-cli, php libs for db2
- File zabdb2.conf contains DB2 Instance Infos and must be readable by zabbix user

7.2. /usr/local/zabbix/etc/zabdb2.conf

```
[global]
        default_user = xxxxxx           # default user who queries mysql
        default_password = xxxxxx

[INSTANCE_NAME]
        host = xxx.xxx.xxx.xxx
        port = 55998
        username = xxxxxx               # specific user per instance; if not configured,
        password = xxxxxx               # default_user is used
```

7.3. Templates

7.3.1.T_DB2-LLD

Template that contains DB2 Instance monitoring.

Template level Macros (inherited at host level)

- {$DB2_MAX_AGENT_WAITING} → Number of agent waiting to work above alarm level
- {$DB2_MAX_SORT_OVERFLOW} → Percentage of sorts in overflow above alarm level
- {$DB2_MAX_LOG_USED} → % of used log above alarm level
- {$DB2_MAX_LOCK_ESCALS} → Number of locks escalations above alarm level
- {$DB2_MAX_XLOCK_ESCALS}→ Number of locks escalations in exclusive mode above alarm level
- {$DB2_MIN_GLOBAL_DATA_HIT_RATIO} → Overall bufferpool data hit ratio below alarm level level
- {$DB2_MIN_GLOBAL_INDEX_HIT_RATIO} → Overall bufferpool index hit ratio below alarm level level
- {$DB2_MIN_BACKUPFREQUENCY} → If no backup during this period (in seconds) then alarm
- {$DB2_MAX_DEADLOCK} → Number of dealocks above alarm level

- {$DB2_MAX_TRANS_IN_DOUBT} → Number of transactions in doubt (which hold locks) above alarm level

7.4. DB2 User rights

-> db2icrt -s client zabbix
Log as user zabbix
-> db2 catalog tcpip node DB2SERVER remote <IP of DB2 server> server <port of DB2 server> # will catalog db2 server
-> db2 catalog database <Database used for connection on db2 server> at node DB2SERVER # will catalog database
Test it
-> db2 'connect to <Database used for connection on db2 server> user <DB2 user> using "<password of DB2 user"'

8. Requests/Comments/Support

CORE IT PROJECT
http://www.coreit.fr

Alain Ganuchaud
alain@coreit.fr