Documentation relative to Release:

1.2.2

Authors:

Andrea Dalle Vacche
Jason Chatfield

Download & Installation Instructions

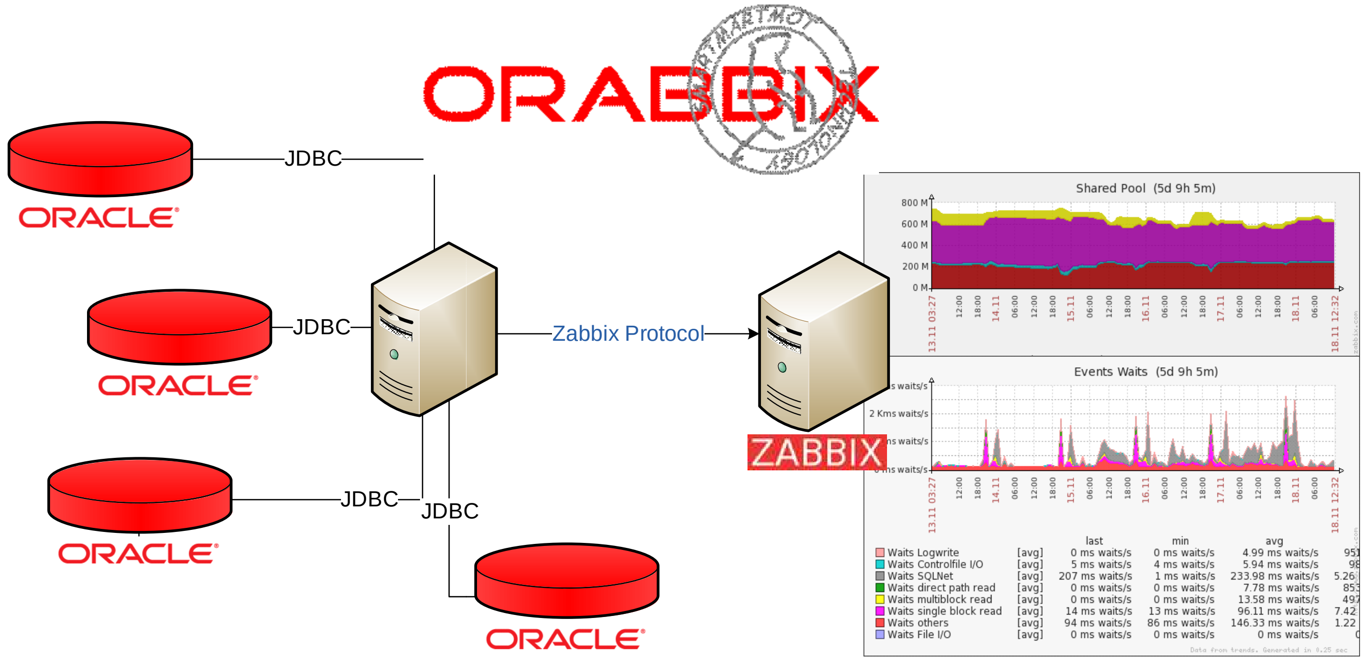
About Orabbix

Product Description

Orabbix is a plugin designed to work with Zabbix Enterprise Monitor to provide multi-tiered monitoring, performance and availability reporting and measurement for Oracle Databases, along with server performance metrics.

It provides an effective mechanism to acquire data from numerous Oracle instances, and in turn provides this information for monitoring and performance metrics to your Zabbix server. You can then utilize the reporting capabilities of Zabbix for all data collected, and provide analysis such as graphs and service level agreement metrics for stakeholders.

The current distribution contains a set of pre-defined templates which incorporate alerting and graphing capabilities from initial deployment. However these can be fine tuned to suit your needs and data/monitoring requirements.



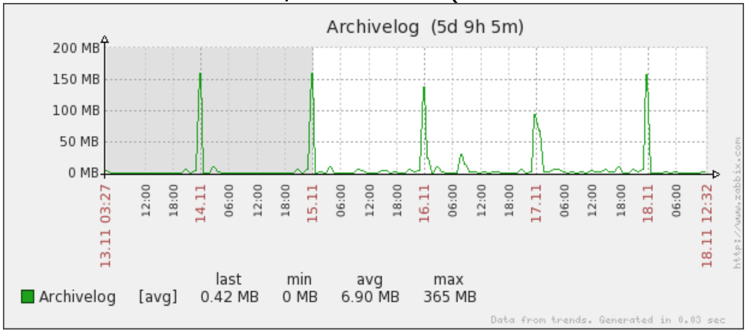
		last	min	avg	
Waits Logwrite	[avg]	0 ms waits/s	0 ms waits/s	4.99 ms waits/s	951
Waits Controlfile I/O	[avg]	5 ms waits/s	4 ms waits/s	5.94 ms waits/s	98
Waits SQLNet	[avg]	207 ms waits/s	1 ms waits/s	233.98 ms waits/s	5.26
Waits direct path read	[avg]	0 ms waits/s	0 ms waits/s	7.78 ms waits/s	853
Waits multiblock read	[avg]	0 ms waits/s	0 ms waits/s	13.58 ms waits/s	497
Waits single block read	[avg]	14 ms waits/s	13 ms waits/s	96.11 ms waits/s	7.42
Waits others	[avg]	94 ms waits/s	86 ms waits/s	146.33 ms waits/s	1.22
Waits File I/O	[avg]	0 ms waits/s	0 ms waits/s	0 ms waits/s	0

Fig. 1 - Orabbix Architecture

What can it monitor?

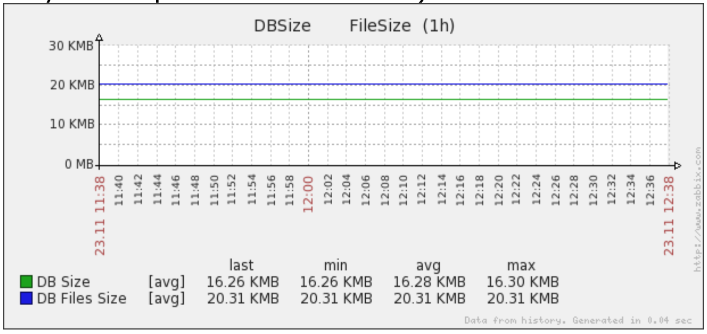
Below are some examples of what Orabbix is capable of monitoring:

- DB Version (i.e. Validity of package)
- Archiving (Archive log production with trend analysis)
- Event Waits (Files I/O, single block read, multi-block read, direct path read, SQLNet Messages, Control file I/O, Log Write)
- Hit Ratio (Hit Ratio on Triggers, Tables/Procedures, SQL Area, Body)
- Logical I/O (Server performance on Logical I/O of: Current Read, Consistent Read, Block Change)
- Physical I/O (Redo Writes, Datafile Writes, Datafile Reads)
- PGA
- SGA (In particular; Fixed Buffer, Java Pool, Large Pool, Log Buffer, Shared Poolm Buffer Cache)
- Shared Pool (Pool Dictionary Cache, Pool Free Memory, Library Chache, SQL Area, MISC.)
- Pin Hit Ratio (Oracle library cache pin are caused by contention with the library cache, the area used to store SQL executables for re-use)
- Sessions / Processes
- Sessions (Active Sessions, Inactive Sessions, System Sessions)
- DBSize/DBFileSize (DBSize size of database really used space and of Filesize)



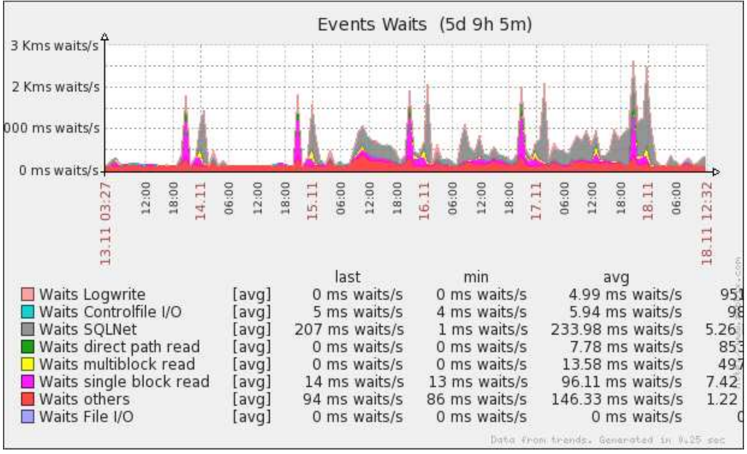
		last	min	avg	max
🟩 Archivelog	[avg]	0.42 MB	0 MB	6.90 MB	365 MB

Fig. 1 – Archivelog



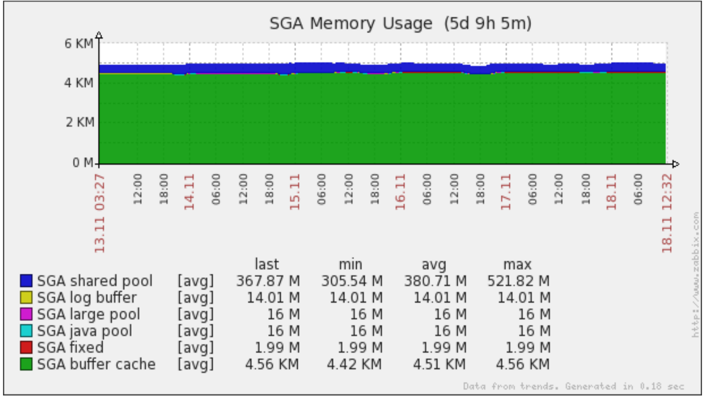
		last	min	avg	max
🟩 DB Size	[avg]	16.26 KMB	16.26 KMB	16.28 KMB	16.30 KMB
🟦 DB Files Size	[avg]	20.31 KMB	20.31 KMB	20.31 KMB	20.31 KMB

Fig. 2 - DBSize and DBFileSize



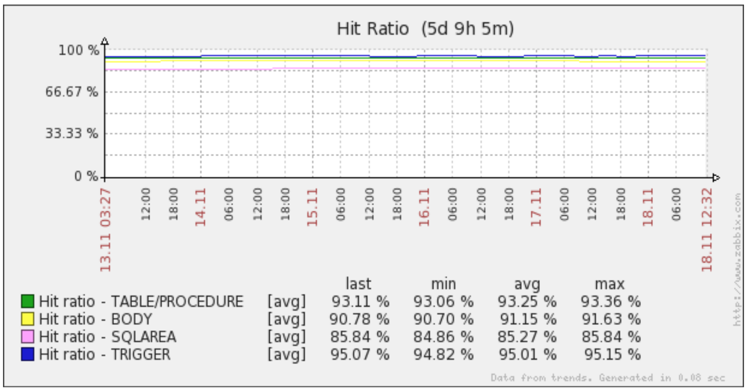
		last	min	avg	
🟥 Waits Logwrite	[avg]	0 ms waits/s	0 ms waits/s	4.99 ms waits/s	951
🟦 Waits Controlfile I/O	[avg]	5 ms waits/s	4 ms waits/s	5.94 ms waits/s	98
⬛ Waits SQLNet	[avg]	207 ms waits/s	1 ms waits/s	233.98 ms waits/s	5.26
🟩 Waits direct path read	[avg]	0 ms waits/s	0 ms waits/s	7.78 ms waits/s	853
🟨 Waits multiblock read	[avg]	0 ms waits/s	0 ms waits/s	13.58 ms waits/s	497
🟪 Waits single block read	[avg]	14 ms waits/s	13 ms waits/s	96.11 ms waits/s	7.42
🟥 Waits others	[avg]	94 ms waits/s	86 ms waits/s	146.33 ms waits/s	1.22
🟦 Waits File I/O	[avg]	0 ms waits/s	0 ms waits/s	0 ms waits/s	0

Fig. 3 - Events Waits

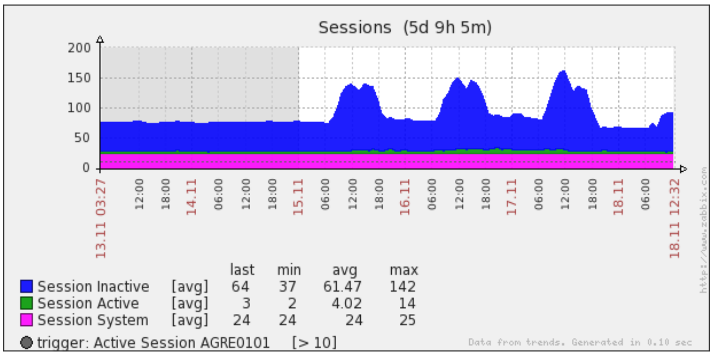


		last	min	avg	max
🟦 SGA shared pool	[avg]	367.87 M	305.54 M	380.71 M	521.82 M
🟨 SGA log buffer	[avg]	14.01 M	14.01 M	14.01 M	14.01 M
🟪 SGA large pool	[avg]	16 M	16 M	16 M	16 M
🟦 SGA java pool	[avg]	16 M	16 M	16 M	16 M
🟥 SGA fixed	[avg]	1.99 M	1.99 M	1.99 M	1.99 M
🟩 SGA buffer cache	[avg]	4.56 KM	4.42 KM	4.51 KM	4.56 KM

Fig. 4 – SGA



		last	min	avg	max
🟩 Hit ratio - TABLE/PROCEDURE	[avg]	93.11 %	93.06 %	93.25 %	93.36 %
🟨 Hit ratio - BODY	[avg]	90.78 %	90.70 %	91.15 %	91.63 %
🟪 Hit ratio - SQLAREA	[avg]	85.84 %	84.86 %	85.27 %	85.84 %
🟦 Hit ratio - TRIGGER	[avg]	95.07 %	94.82 %	95.01 %	95.15 %

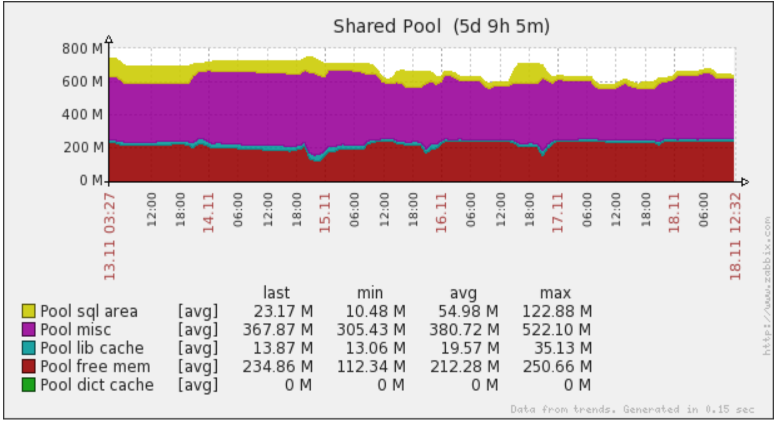


		last	min	avg	max
🟦 Session Inactive	[avg]	64	37	61.47	142
🟩 Session Active	[avg]	3	2	4.02	14
🟪 Session System	[avg]	24	24	24	25

● trigger: Active Session AGRE0101 [> 10]

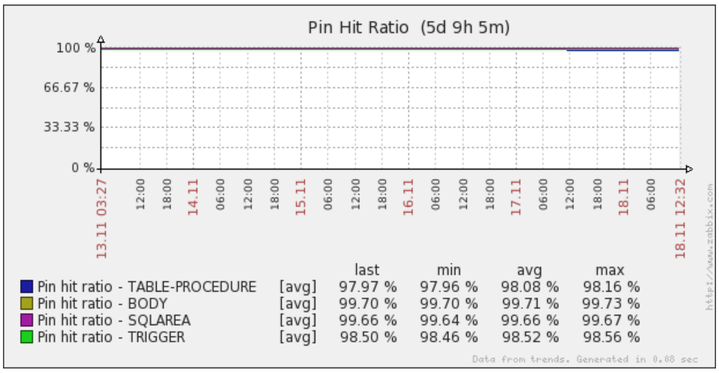
Fig. 6 – Sessions

Fig. 5 - Hit Ratio



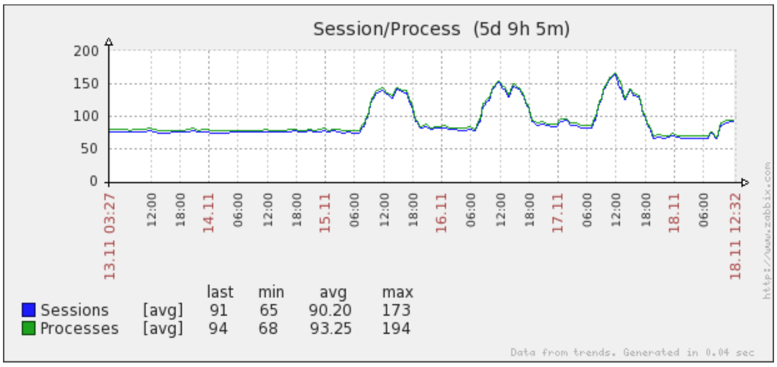
		last	min	avg	max
🟨 Pool sql area	[avg]	23.17 M	10.48 M	54.98 M	122.88 M
🟪 Pool misc	[avg]	367.87 M	305.43 M	380.72 M	522.10 M
🟦 Pool lib cache	[avg]	13.87 M	13.06 M	19.57 M	35.13 M
🟥 Pool free mem	[avg]	234.86 M	112.34 M	212.28 M	250.66 M
🟩 Pool dict cache	[avg]	0 M	0 M	0 M	0 M

Fig. 72 - Shared Pool



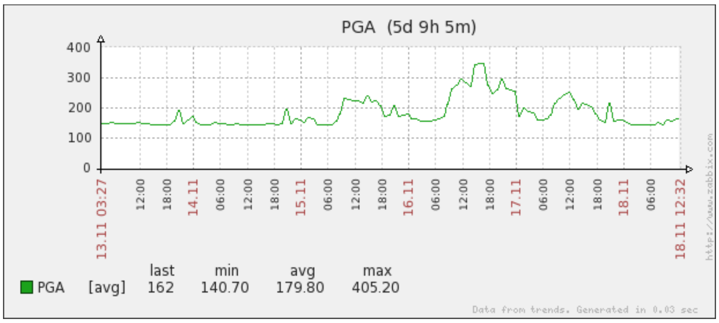
		last	min	avg	max
🟦 Pin hit ratio - TABLE-PROCEDURE	[avg]	97.97 %	97.96 %	98.08 %	98.16 %
🟨 Pin hit ratio - BODY	[avg]	99.70 %	99.70 %	99.71 %	99.73 %
🟪 Pin hit ratio - SQLAREA	[avg]	99.66 %	99.64 %	99.66 %	99.67 %
🟩 Pin hit ratio - TRIGGER	[avg]	98.50 %	98.46 %	98.52 %	98.56 %

Fig. 3 - Pin Hit Ratio



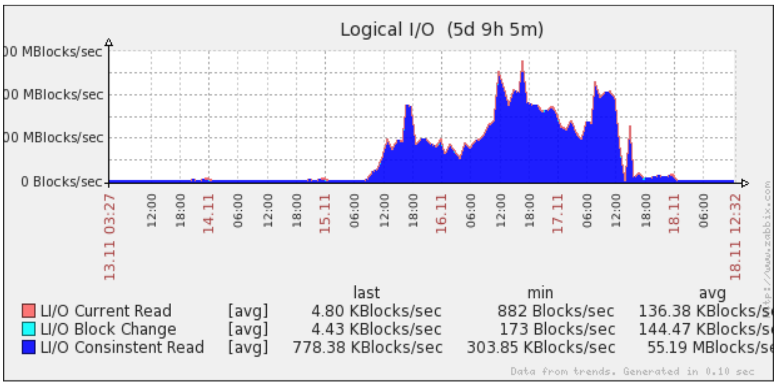
		last	min	avg	max
🟦 Sessions	[avg]	91	65	90.20	173
🟩 Processes	[avg]	94	68	93.25	194

Fig. 94 - Sessions Processes



		last	min	avg	max
🟩 PGA	[avg]	162	140.70	179.80	405.20

Fig. 10 – PGA



		last	min	avg
🟥 LI/O Current Read	[avg]	4.80 KBlocks/sec	882 Blocks/sec	136.38 KBlocks/s
🟦 LI/O Block Change	[avg]	4.43 KBlocks/sec	173 Blocks/sec	144.47 KBlocks/s
🟦 LI/O Consinstent Read	[avg]	778.38 KBlocks/sec	303.85 KBlocks/sec	55.19 MBlocks/s

Fig. 11 - Logical I/O

.

Orabbix Author

Andrea Dalle Vacche: http://www.smartmarmot.com

History

You can read about the history and evolution of the Orabbix plug-in in these threads below:

Original here: http://www.zabbix.com/forum/showthread.php?t=13666
Latest here: http://www.zabbix.com/forum/showthread.php?t=16391

Distribution

You can find the latest versions of Orabbix at the locations listed below:

http://www.smartmarmot.com/product/orabbix/download/
or
https://sourceforge.net/projects/orabbix/

Requirements

To use Orabbix, you will require the following on your Zabbix Server:

- Zabbix 1.8.x Server
- Java Runtime Environment 6

Current known successfully tested Oracle host platforms:

- RHEL5.X (CentOS 5.4 & 5.5)
- Windows 2003 (with Java SE 1.6)
- HP-UX 11.31
- AIX 5.3

Currently known tested Oracle Versions:

- 10g
- 10.2

***** IMPORTANT*****

You do NOT need to install an Oracle client of any kind for Orabbix to work.

Orabbix uses Java objects and connection strings to connect to the Oracle Database, and as such doesn't require an Oracle client to be installed on your Zabbix Server.

Installation

Assumptions

The installation instructions have been created based on instructions for most *nix deployments (i.e. RHEL/CentOS), and is assumed that any alterations required for your specific environment are to be taken into account accordingly.

This guide is based upon there being two (2) Hosts, 1x Zabbix Server and 1x Oracle Server. If you are planning on monitoring an Oracle instance that is running on your Zabbix Server, the steps are the same, with minor adjustments required for your connection information.

The steps also assume you are configuring Orabbix to monitor a new installation or setup of Oracle. The installation steps will have you GRANT access for the Zabbix user to all tables, and this will include any USER tables present at the time of execution. If you do not want Zabbix to have access to specific tables or resources within your database, you will need to set a DENY to the Zabbix users access as required. Consult your DBA for details, as this is beyond the scope of these instructions.

Steps for Installation

1. Download Orabbix to your Zabbix Server
2. On your Zabbix server, unzip Orabbix to: /opt/orabbix
3. Copy file /opt/orabbix/init.d/orabbix to /etc/init.d/orabbix
4. Grant execute permissions to the following files:
   - /etc/init.d/orabbix
   - /opt/orabbix/run.sh
5. Create a User (ZABBIX) for Orabbix to access your Oracle Database. You can use the following script:

```
CREATE USER ZABBIX
IDENTIFIED BY <REPLACE WITH PASSWORD>
DEFAULT TABLESPACE SYSTEM
TEMPORARY TABLESPACE TEMP
PROFILE DEFAULT
ACCOUNT UNLOCK;
– 2 Roles for ZABBIX
GRANT CONNECT TO ZABBIX;
GRANT RESOURCE TO ZABBIX;
ALTER USER ZABBIX DEFAULT ROLE ALL;
– 5 System Privileges for ZABBIX
GRANT SELECT ANY TABLE TO ZABBIX;
GRANT CREATE SESSION TO ZABBIX;
GRANT SELECT ANY DICTIONARY TO ZABBIX;
GRANT UNLIMITED TABLESPACE TO ZABBIX;
GRANT SELECT ANY DICTIONARY TO ZABBIX;
```

**** NOTE :*** If you are using Oracle 11g, you will need to add the following:

```
exec dbms_network_acl_admin.create_acl(acl => ‘resolve.xml’,description => ‘resolve acl‘, principal =>’ZABBIX’,
is_grant => true, privilege => ‘resolve’);
exec dbms_network_acl_admin.assign_acl(acl => ‘resolve.xml’, host =>’*');
```

You can verify the above is correct by running:

```
select utl_inaddr.get_host_name(’127.0.0.1′) from dual;
```

6. For this example on RedHat, run:

```
chkconfig -add orabbix
```

7. Verify with:

```
chkconfig -list
```

8. Create a Host entry in Zabbix for the DB Instance/s you are planning to monitor, and import the templates found at: /opt/orabbix/template

** Please note, instructions for step (8) can be found in the Zabbix Manual at http://www.zabbix.com/documentation.php
With this step (8), ensure the name of your host in Zabbix is the same name of the SID or Oracle instance name.

Configuration

Now we need to configure your Orabbix setup. The tags below are listed as they will appear within the respective configuration files. The first you will need to modify is your config.props file to define your connection properties for Zabbix and Oracle.

The config.props file can be found at ***/opt/orabbix/conf/config.props***

ZabbixServerList

#comma separed list of Zabbix servers
The settings under this tag allow you to configure your Zabbix Server information.
Orabbix can also be configured to send your Oracle data to multiple Zabbix servers.

TIP: *This can be beneficial for distributed monitoring scenarios, server migrations or replicating Orabbix data to a Disaster Recovery site or server.*

Replace with your Zabbix Server info where appropriate;

```
ZabbixServerList=ZabbixServer1,ZabbixServer2

ZabbixServer1.Address=192.168.0.1
ZabbixServer1.Port=10051

ZabbixServer2.Address=192.168.0.2
ZabbixServer2.Port=10051
```

OrabbixDaemon

Entries under this tag allow you to set your Orabbix Daemon parameters.

#MaxThreadNumber should be >= than the number of your databases
Set the number of threads the Orabbix Daemon should have inside his internal pool of DB Jobs. This number *should* be at least equal to (or more than) then number of databases monitored by Orabbix, now this parameter *if not set is automatically calculated.*

e.g. For 50 Databases, we are using 100 threads

```
OrabbixDaemon.MaxThreadNumber=100
```

#pidFile
Next you can set the location of the Daemons PID file. Default location is: /opt/orabbix/logs

```
OrabbixDaemon.PidFile=./logs/orabbix.pid
```

** NOTE: Relative path is permitted here

TIP: *With the features of the Orabbix Daemon, it is possible to clone Orabbix and use different configurations. As such, it is therefore possible to have one Daemon that checks your TEST databases, another iteration that checks your DEVELOPMENT databases and a third that checks your PRODUCTION databases, each with different timings and check loops!*

DatabaseList

#put here your databases in a comma separated list

This is where you define your database instances. You can specify more than one instance here, separated using a comma.

DatabaseList=EXAMPLE1,EXAMPLE2

** Note: The names of the instances must match those you have specified as your HOST name in Zabbix

#Configuration of Connection pool

From here, you will configure settings that are specific to the connection pool.

As the comments in config.props suggest, if you do not specify these values, Orabbix will use default values which have been hard-coded.

#Maximum number of active connection inside pool

Set the maximum number of connections that can be allocated to this pool at any time,or alternatively set a negative value for no limit.

DatabaseList.MaxActive=10

#The maximum number of milliseconds

Here you define how long that the pool will wait (when there are no available connections) for a connection to be returned before throwing an exception, or set the value <= 0 to wait indefinitely.

DatabaseList.MaxWait=100

Also under the same section, you can define the maximum number of connections that can remain idle within the connection pool, without being released. Alternatively, you can set a negative value for no limit.

DatabaseList.MaxIdle=1

TIP: *You can specify the Database connection parameters for each database you wish to monitor individually, underneath your Database Connection Parameters. This allows you to customise your connection settings based on the Database constraints I.e one connection pool for your PRODUCTION DB and another for you TEST DB, for example;*

DB1.MaxActive=10
DB1.MaxWait=100
DB1.MaxIdle=1

Note that any settings defines this way will override the general settings for your generic connection pool.

Database Connection Parameters

This section sets your connection string to the Oracle Database. This string invokes a Java Database Connector (JDBC) to your Oracle Databases, and as such does not need the Oracle Client to be installed.

#define here your connection string for each database

Here you will define the connection string. These are formatted as: DBName.Url(*as specified in your DatabaseList*) followed by the jdbc string and your Database Server information.

***EXAMPLE*1.Url**=*jdbc:oracle:thin:@SERVER1.EXAMPLE.COM:1521:DB1*

Set your Database username and password below. e.g.

EXAMPLE1.User=zabbix
EXAMPLE1.Password=zabbix_password

TIP: Setting connections to multiple databases is made easy by adding more connection strings, with their corresponding credentials. Example given below.

EXAMPLE2.Url=jdbc:oracle:thin:@server2.domain.example.com:<LISTENER_PORT>:EXAMPLE2
EXAMPLE2.User=zabbix
EXAMPLE2.Password=zabbix_password

EXAMPLE3.Url=jdbc:oracle:thin:@server3.domain.example.com:<LISTENER_PORT>:EXAMPLE3
EXAMPLE3.User=zabbix
EXAMPLE3.Password=zabbix_password

** Note: After these entries, you can set your MaxActive, MaxWait and MaxIdle for the individual database connections if you so wish.

There is a pair of default parameter:

DefaultUser = <USERNAME>
DefaultPassword = <PASSWORD>
this default parameter can be overridden using the following pair

<DB_NAME>.User and <DB_NAME>.User
It is only possible to override Username or Password.

Query List File

This defines where the file containing the Oracle (SQL) queries can be found. This is a customisable file. You can find the query parameter file at:

QueryListFile=/opt/orabbix/conf/query.props
you can also specify :
<DBNAME>.QueryListFile=./confQueryTest.props this file specify the query file of a database relative path are allowed

TIP: This is really useful to use different query file for different databases or to have a pool of query file one for each Oracle Release (if you have different release) and/or to have different query files for Production environment, Test environment, Developement environment etc..
the following parameter

<DB_NAME>.ExtraQueryListFile ==./confQueryTest_2.props
adds another custom query to the default query file.
If there are duplicates the ExtraQueryListFile will override them.

Configuration of query.props file

The query.props file can be modified or added to, so you can supply your own customised queries through Orabbix against your Oracle instances/databases. Each query created has an associated 'Item' or item name that Zabbix will use to identify the query.

** Note: Item names must be unique.

The configurable items are formatted as follows.

#QueryList
You have to set the query name under the QueryList in the query.props file. Each query name is comma separated. For example;

QueryList=queryName1,queryName2,queryName3

#Query
You must identify the query by the unique item name you specified in the QueryList, followed by '.Query' for Orabbix to recognize that this is the query string. Its important to remember NOT to add the semi-colon ";" to the end of your custom query.

customQueryItemName.Query=yourQueryHere

Now you can define what you want Orabbix to return to your Zabbix Server if no data is found for your query.

customQueryItemName.NoDataFound=none

TIP: In the example above, Orabbix would send the string "none" to the Zabbix Server.

You can specify if you want a different execution period for your query
customQueryItemName.Period=<Express a period in minute>
TIP:You can change the execution time of dbversion that don't change often

Next you can specify a query that will be executed and if return *RaceCondiftionValue* the query *customQueryItemName is executed* otherwise is *skipped*

customQueryItemName.ACTIVE=[true|false] if true query is executed otherwise skipped

```
archive.Query=select round( A.LOGS*B.AVG/1024/1024/10 )
from ( SELECT  COUNT (*)  LOGS FROM V$LOG_HISTORY WHERE
FIRST_TIME >= (sysdate -10/60/24)) A,
( SELECT Avg(BYTES) AVG,  Count(1), Max(BYTES) Max_Bytes,Min(BYTES) Min_Bytes  FROM
v$log) B

archive.RaceConditionQuery=select value
                           from
                           v$parameter where name='log_archive_start'

archive.RaceConditionValue=FALSE
```

TIP: In the example above, Orabbix would execute "archive.Query" only if the query "archive.RaceConditionQuery" return the value "archive.RaceConditionValue". Pratically the "Archive" query is execute only if database is in arhivelogmode (Parameter "log_archive_start" set to TRUE)

<QueryName>.Trim=[true|false]
if true the resultset is trimmed (default is true)

<QueryName>.AddSpaces=[true|false]
if true the add a space between columns of resultset(default is true)

<QueryName>.ExcludeColumnsList=[1,2,3,..n]
exclude from result set the 1st, the 2nd etc.. columns from resultset
<QueryName>.Period

<QUERY_NAME>.WhenNotAlive = <VALUE>
is used to specify a value or a string to send if a database is not working,
doing such will "clean" the graphs generated in Zabbix with a "set" value.

```
Added the following parameter on main configuration file:
    DefaultUser = <USERNAME>
    DefaultPassword = <PASSWORD>
```

< QueryName >.ExtraQueryListFile = <VALUE>

```
    on query's properties file I've added the following parameter:
    <QUERY_NAME>.WhenNotAlive = <VALUE>

    with these parameter you can define a default username and password for
    all the database.
    DefaultUser = <USERNAME>
    DefaultPassword = <PASSWORD>
    this default parameter can be overridden using the couple
    <DB_NAME>.User and <DB_NAME>.User
    Is possible to override only Username or Password.

    the following parameter
    <DB_NAME>.ExtraQueryListFile =
    add another custom query to a default query file.
    if there are duplicate the ExtraQueryListFile will override them.

    the following parameter
    <QUERY_NAME>.WhenNotAlive = <VALUE>
    is used to specify a value or a string to send if a database is not working,
    doing so you are going to "clean" the graphs generated in Zabbix with a "set" value.
```

How to Use Orabbix/FAQ

How do I start/stop the Daemon?

To start the Orabbix Daemon, simply run:
/etc/init.d/orabbix start
To stop the Orabbix Daemon, simply run:
/etc/init.d/orabbix start

How does Logging work?

The Orabbix daemon outputs its log file to the default location of ***/opt/orabbix/logs/***

Logging properties can be modified by making your required changes to;
/opt/orabbix/conf/log4j.properties

The property setting responsible for defining the output location is;
log4j.appender.Orabbix.File=logs/orabbix.log

TIP: From this properties file, you can modify the location, file name and log format as desired. For additional information, please refer to the official log4j documentation, found at; http://logging.apache.org/log4j/1.2/index.html

Is there a way for orabbix to connect to RAC or DataGuard?

Yes is possible.
e.g.
If you have two host RAC1 and RAC2 and one instance RACINST
you should write the connection string as follow:
RACINST.Url=jdbcracle:thin:@(DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=RAC1.EXAMPLE.COM)(PORT=1521))(ADDRESS=(PROTOCOL=TCP)(HOST=RAC2.EXAMPLE.COM)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=RACINST)))

What can I modify without restarting the daemon?

The parameters that are dynamically read at each iteration of “OrabbixDaemon.Sleep” are as follows;

1. Any Query added to query.props
2. Modifications to config.props

Available Upcoming Release; To be advised

3. Database List

Currently, the only known items that don't dynamically update are the *ZabbixDaemon.MaxThreadNumber,* and changes to Connection Pool info.

LICENCE Information

Orabbix is released under and according to the
GNU GENERAL PUBLIC LICENSE
Version 3, 29 June 2007