IDERA | MySQL and MariaDB

OVERVIEW OF SQL DIAGNOSTIC MANAGER FOR MYSQL

TABLE OF CONTENTS

1. SUMMARY

In April 2018, IDERA's parent company Idera, Inc. acquired Webyog. Since then, IDERA has incorporated Webyog's MySQL monitoring tool 'Monyog' into its product portfolio and rebranded it as 'SQL Diagnostic Manager for MySQL'.

The MySQL and MariaDB database systems are rapidly becoming the de-facto choice for traditional enterprises as well as pioneering Web 2.0 companies. Webyog has been consistently delivering powerful tools exclusively on the MySQL and MariaDB platform for the last 15 years.

While the adoption rate of MySQL and MariaDB continues to grow in momentum, it poses some unique challenges for traditional database administrators and their organizations. Some of these challenges are:

- Find problematic SQL queries
- Analyze MySQL and MariaDB performance data collected over a period
- Ensure availability of critical MySQL and MariaDB systems
- Keep the systems well-tuned
- Get proactive alerts before problems start surfacing
- Identify problems quickly
- Monitor MySQL and MariaDB server health continuously
- Profile queries
- Protect critical systems

SQL Diagnostic Manager for MySQL enables database administrators to meet these challenges. It could help database administrators working for large organizations who have deployed hundreds of MySQL and MariaDB servers. It could also help database administrators working for small organizations with only a handful of MySQL and MariaDB servers. It is designed to scale resources related database administration by providing a unified view of the health, security, performance, and availability of the entire MySQL and MariaDB server environment. The solution brief explores SQL Diagnostic Manager for MySQL in detail and explains how to leverage it as a 'database administrator in a box'.

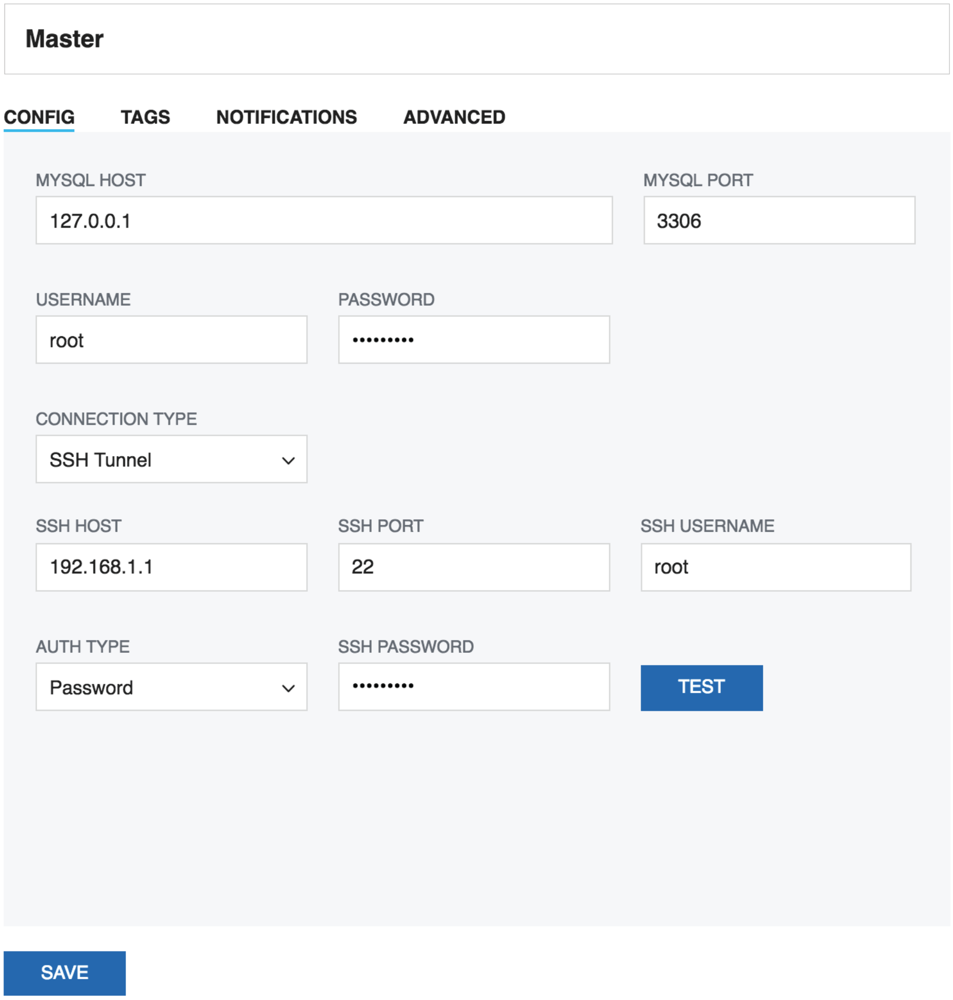
2. INTRODUCTION

SQL Diagnostic Manager for MySQL is a 'database administrator in a box' for MySQL and MariaDB that enables database administrators to manage more database servers, tune their database servers, and find and fix problems with their database applications before they become serious problems and costly outages. It proactively monitors enterprise database environments. It provides expert advice on how even those who are new to MySQL and MariaDB can tighten security, optimize performance and reduce downtime of their MySQL and MariaDB powered systems.

3. AGENTLESS MONITORING

Unlike other monitoring and advisory tools for MySQL and MariaDB, SQL Diagnostic Manager for MySQL does not require installing monitoring agents on each MySQL and MariaDB server. Installing and maintaining monitoring agents can be a complex administration task by itself.

SQL Diagnostic Manager for MySQL uses a standard MySQL connection for monitoring. To collect data for the operating system from remote database servers, it uses secure shell (SSH) on Linux systems. SQL Diagnostic Manager for MySQL also supports secure shell (SSH) tunneling to connect to database servers. The connection feature means that it collects all monitoring data by using remote connections. The collection feature is a considerable advantage that sets it apart from all other monitoring and advisory tools for MySQL and MariaDB.



MYSQL HOST	127.0.0.1			MYSQL PORT	3306
USERNAME	root	PASSWORD	•••••••••		
CONNECTION TYPE	SSH Tunnel				
SSH HOST	192.168.1.1	SSH PORT	22	SSH USERNAME	root
AUTH TYPE	Password	SSH PASSWORD	•••••••••		

Figure 1: The Config web page to configure the connection to each database server without the need to install monitoring agents.

4. CUSTOMIZATION

Each of the included Advisor Rules enables database administrators to customize the thresholds that are acceptable for a specific MySQL and MariaDB server. For example, a database administrator using the included Advisor Rule 'MySQL Key Cache Has Sub-Optimal Hit Ratio' may use lower threshold values for their database servers running online transaction processing (OLTP) applications, while using higher thresholds for online analytical processing (OLAP) applications.

The entire application logic of SQL Diagnostic Manager for MySQL consists of JavaScript Objects that are parsed and executed by the embedded JavaScript Runtime of SQL Diagnostic Manager for MySQL. These JavaScript Objects are available with SQL Diagnostic Manager for MySQL as source code. The access to JavaScript Objects means that it is possible to add new monitors, modify existing monitors, and disable some of the predefined monitors that are included with SQL Diagnostic Manager for MySQL.

5. COMPARISON WITH OTHER MONITORING TOOLS

	SQL Diagnostic Manager for MySQL	Other monitoring tools
Optimize server configuration file 'my.cnf' and the server initialization file 'my.ini'	Using slow query logs, general query logs, MySQL Proxy application, and by sniffing process list	Using MySQL Proxy application and connectors not publicly available
Identify problematic SQL queries	Using slow query logs, general query logs, MySQL Proxy application, and by sniffing process list	Using MySQL Proxy application and connectors not publicly available
All components included with base price	Yes	No
Agentless monitoring	Yes	No (need to install agents on each server)
Licensing	Perpetual	Subscription
Customization	Fully (using JavaScript and the Object Model)	Limited
Low footprint	Yes (with embedded web server and database)	No (bloatware; need to install multiple agents, web servers, and runtimes)

Simple installation	Yes	No
Zero maintenance	Yes	No
Modern and intuitive and web interface	Yes (using asynchronous JavaScript and XML (AJAX))	No (using classic technologies)
No annoying web page refreshes	Yes (even with real-time charting)	No (full refreshes during updates)
Identify deadlocks	Yes	No
Monitor MySQL error logs	Yes	No
Maintain server configuration and tracking changes made to server initialization file ‘my.ini’	Yes	No
Snapshot of disk space used by MySQL objects.	Yes	No
User-defined queries that can monitor MySQL	Yes	No

6. CLOUD READINESS

SQL Diagnostic Manager for MySQL is ready for the cloud by design. It requires no agents to be installed on the MySQL and MariaDB servers. Consequently, it can monitor MySQL and MariaDB instances running in the cloud. Add and remove database servers with just a single click of a mouse button. According to our internal benchmark reports, a single instance of SQL Diagnostic Manager for MySQL can monitor more than 500 MySQL and MariaDB servers (even at collection intervals of 1 second) effortlessly. SQL Diagnostic Manager for MySQL is a perfect match for monitoring MySQL and MariaDB databases in the cloud.

7. BENEFITS OF SQL DIAGNOSTIC MANAGER FOR MYSQL

SQL Diagnostic Manager for MySQL is designed to enable organizations to scale their existing resources for database administration. It provides a single, consolidated view of the health, security, performance, and availability of all MySQL or MariaDB servers in the environment. It proactively monitors all MySQL or MariaDB servers using a set of predefined expert monitors. It identifies and alerts database administrators of problems, security vulnerabilities, and tuning opportunities such that they can be acted upon well in advance of a problem and an outage.

7.1 FIND PROBLEMATIC SQL QUERIES

MySQL and MariaDB currently lack advanced tools for profiling SQL queries (such as SQL Profiler of Microsoft' SQL Server). While other monitoring tools for MySQL and MariaDB provide monitoring and advisory information on various system metrics, they do not pinpoint the problematic SQL queries. No amount of hardware upgrades and tuning of the parameters in the database server configuration file 'my.cnf' and the database server initialization file 'my.ini' can match the performance gains that are achievable when identifying and rewriting problematic queries and creating appropriate indexes.

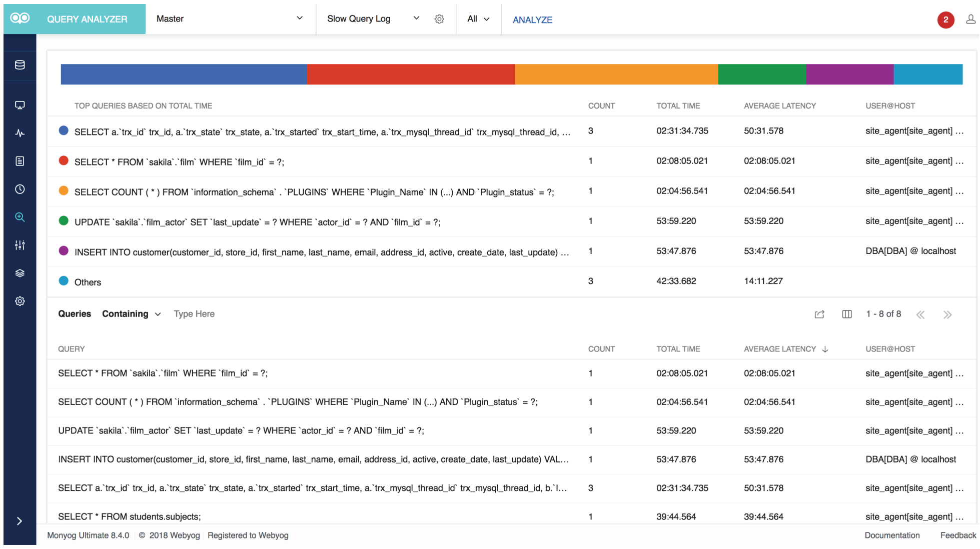
SQL Diagnostic Manager for MySQL finds problematic SQL queries by

- Querying the MySQL Proxy application that clients and applications are configured to connect through
- Analyzing slow query logs
- Analyzing general query logs
- Issuing the SHOW PROCESSLIST statement at regular intervals
- Utilizing Performance Schema tables

Additionally, export the reports created by the above methods as comma-separated values (CSV) files. The export functionality means that it is possible to further customize the report using a spreadsheet and by directly importing the comma-separated values (CSV) output into a MySQL and MariaDB table for further analysis.

7.2 MONITOR AND ANALYZE SQL QUERIES

Identifying problematic SQL queries (that is, by looking at slow query logs, general query logs, the Performance Schema feature, the SHOW PROCESSLIST statement, and the MySQL Proxy application) is important. However, this analysis is usually performed well after the problematic SQL queries executed. Consequently, real-time notifications for long-running queries are needed. SQL Diagnostic Manager for MySQL continuously monitors SQL queries in real-time. It sends notifications (via email, simple network management protocol (SNMP), the collaboration platform Slack, the incident response platform PagerDuty, and the messaging logging standard syslog) for SQL queries that take more than a specified amount of time to execute. It is also possible to specify an option to automatically kill such queries.



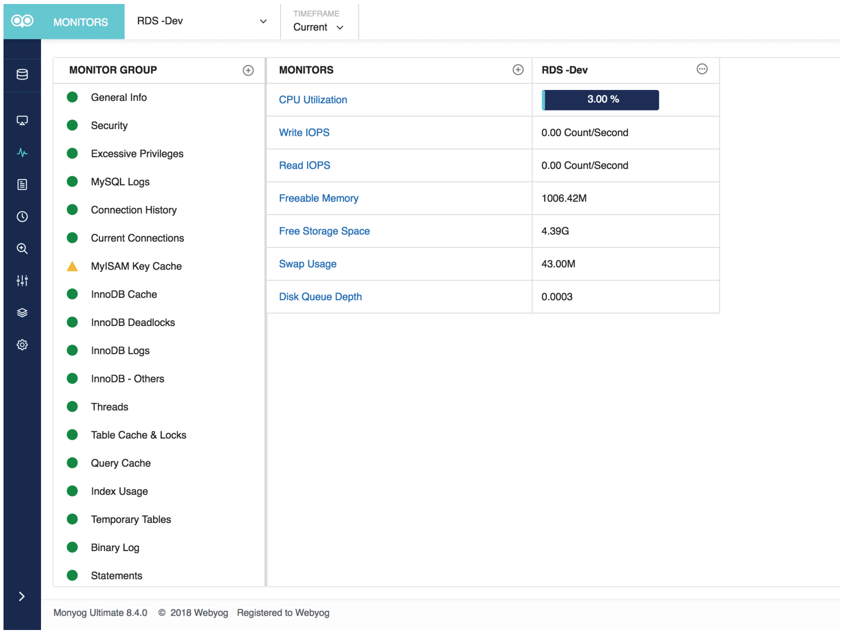
TOP QUERIES BASED ON TOTAL TIME	COUNT	TOTAL TIME	AVERAGE LATENCY	USER@HOST
● SELECT a.`trx_id` trx_id, a.`trx_state` trx_state, a.`trx_started` trx_start_time, a.`trx_mysql_thread_id` trx_mysql_thread_id, …	3	02:31:34.735	50:31.578	site_agent[site_agent] …
● SELECT * FROM `sakila`.`film` WHERE `film_id` = ?;	1	02:08:05.021	02:08:05.021	site_agent[site_agent] …
● SELECT COUNT (*) FROM `information_schema` . `PLUGINS` WHERE `Plugin_Name` IN (...) AND `Plugin_status` = ?;	1	02:04:56.541	02:04:56.541	site_agent[site_agent] …
● UPDATE `sakila`.`film_actor` SET `last_update` = ? WHERE `actor_id` = ? AND `film_id` = ?;	1	53:59.220	53:59.220	site_agent[site_agent] …
● INSERT INTO customer(customer_id, store_id, first_name, last_name, email, address_id, active, create_date, last_update) …	1	53:47.876	53:47.876	DBA[DBA] @ localhost
● Others	3	42:33.682	14:11.227	

QUERY	COUNT	TOTAL TIME	AVERAGE LATENCY ↓	USER@HOST
SELECT * FROM `sakila`.`film` WHERE `film_id` = ?;	1	02:08:05.021	02:08:05.021	site_agent[site_agent] …
SELECT COUNT (*) FROM `information_schema` . `PLUGINS` WHERE `Plugin_Name` IN (...) AND `Plugin_status` = ?;	1	02:04:56.541	02:04:56.541	site_agent[site_agent] …
UPDATE `sakila`.`film_actor` SET `last_update` = ? WHERE `actor_id` = ? AND `film_id` = ?;	1	53:59.220	53:59.220	site_agent[site_agent] …
INSERT INTO customer(customer_id, store_id, first_name, last_name, email, address_id, active, create_date, last_update) VAL…	1	53:47.876	53:47.876	DBA[DBA] @ localhost
SELECT a.`trx_id` trx_id, a.`trx_state` trx_state, a.`trx_started` trx_start_time, a.`trx_mysql_thread_id` trx_mysql_thread_id, b.`l…	3	02:31:34.735	50:31.578	site_agent[site_agent] …
SELECT * FROM students.subjects;	1	39:44.564	39:44.564	site_agent[site_agent] …

Figure 2: The Query Analyzer web page that displays the Slow Query Log option.

7.3 MONITOR AMAZON RDS AND ITS OPERATING SYSTEM

SQL Diagnostic Manager for MySQL provides monitoring capabilities for Amazon RDS for MySQL, MariaDB, and Amazon Aurora and its operating system. It uses the application programming interface (API) of Amazon CloudWatch. It uses the different metrics for the operating system that are available with the application programming interface (API) to fetch and display the data. All of the operating system monitors for Amazon RDS are shown under the monitor RDS/Aurora Instance Metrics group in the Monitors page. The corresponding charts are available on the Dashboard page. Just enable system metric for the Amazon RDS Aurora instance to see the data for the operating system.



MONITOR GROUP

- ● General Info
- ● Security
- ● Excessive Privileges
- ● MySQL Logs
- ● Connection History
- ● Current Connections
- ▲ MyISAM Key Cache
- ● InnoDB Cache
- ● InnoDB Deadlocks
- ● InnoDB Logs
- ● InnoDB - Others
- ● Threads
- ● Table Cache & Locks
- ● Query Cache
- ● Index Usage
- ● Temporary Tables
- ● Binary Log
- ● Statements

MONITORS	RDS -Dev
CPU Utilization	3.00 %
Write IOPS	0.00 Count/Second
Read IOPS	0.00 Count/Second
Freeable Memory	1006.42M
Free Storage Space	4.39G
Swap Usage	43.00M
Disk Queue Depth	0.0003

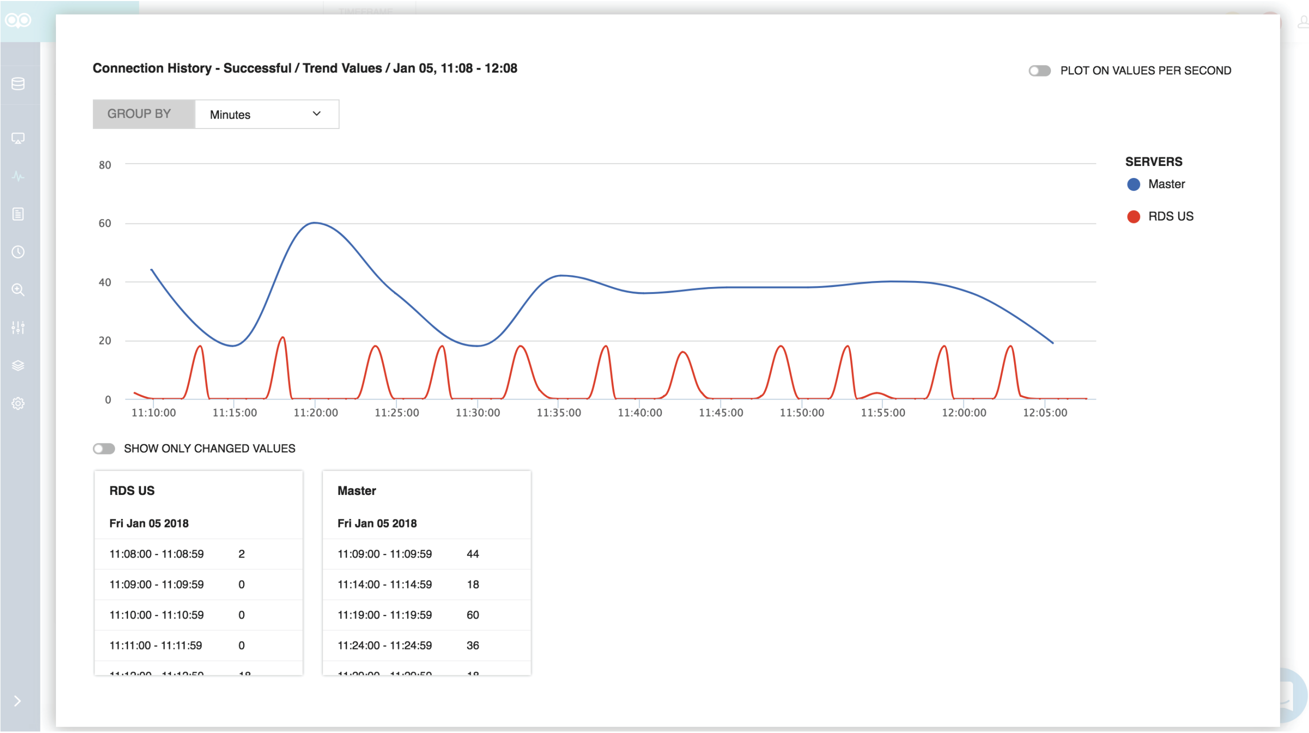
Figure 3: The Monitors web page that displays data for an instance of Amazon RDS.

7.4 VIEW AND UNDERSTAND TRENDS BY ANALYZING HISTORICAL DATA

Configure the time duration for storing the data collected by SQL Diagnostic Manager for MySQL. It stores the data in a high-performance database (that is, the embedded relational database management system SQLite). By analyzing historical data, quickly obtain answers to questions like:

- How many times and when did database servers go down during the last six months? connect through
- Which day of the week has maximum database activity?
- How many time and when were logins attempted with incorrect passwords yesterday?

Sudden changes in performance parameters and problems (such as due to a change of application) will also be visible immediately.



Connection History - Successful / Trend Values / Jan 05, 11:08 - 12:08

PLOT ON VALUES PER SECOND

GROUP BY: Minutes

SHOW ONLY CHANGED VALUES

RDS US

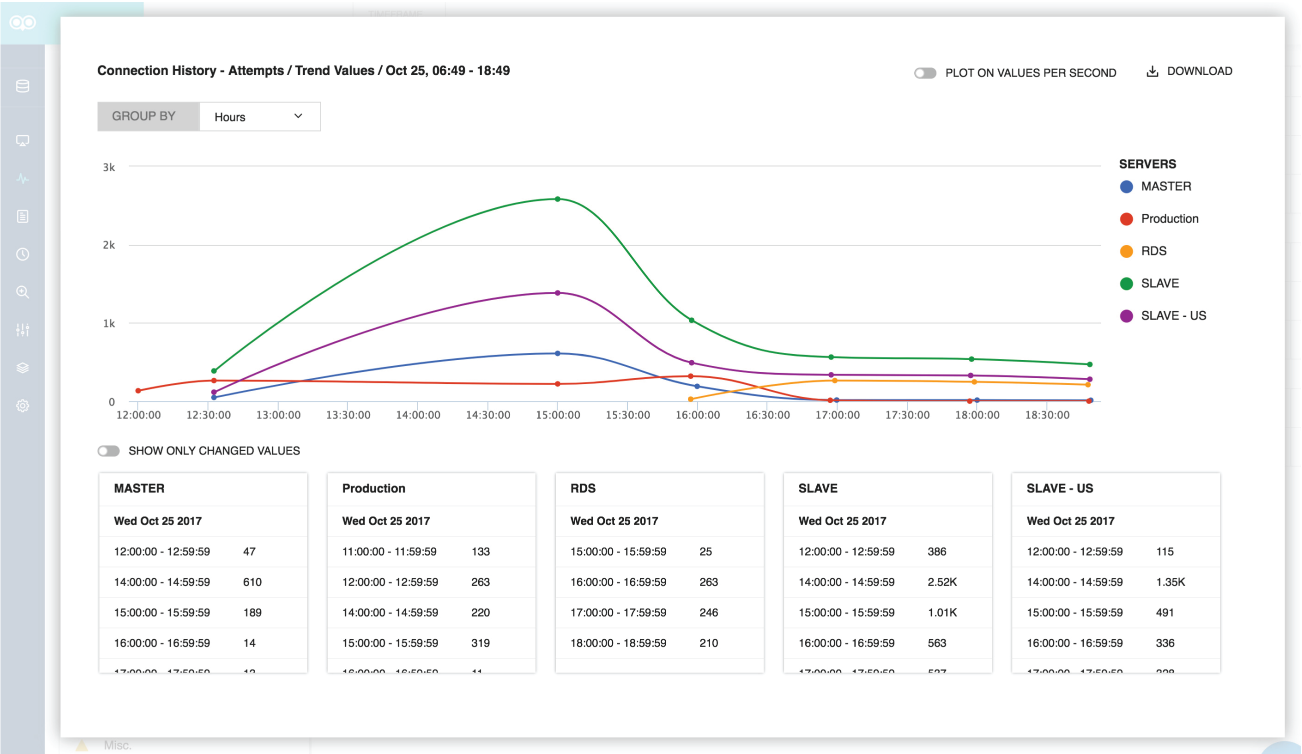
Fri Jan 05 2018	
11:08:00 - 11:08:59	2
11:09:00 - 11:09:59	0
11:10:00 - 11:10:59	0
11:11:00 - 11:11:59	0
11:12:00 - 11:12:59	18

Master

Fri Jan 05 2018	
11:09:00 - 11:09:59	44
11:14:00 - 11:14:59	18
11:19:00 - 11:19:59	60
11:24:00 - 11:24:59	36
11:29:00 - 11:29:59	18

Figure 4: A graph that displays historical data for two different database servers. It shows that the load on the database server 'Master' is high at 60% around 11:20 hours.

SQL Diagnostic Manager for MySQL provides trend graph analysis that makes it easy to compare the state and performance of multiple database servers in a single chart. Group an important single metric from different database servers into a single unified chart. Visually analyze a metric across database servers at various points in time.


Connection History - Attempts / Trend Values / Oct 25, 06:49 - 18:49

GROUP BY: Hours

MASTER	
Wed Oct 25 2017	
12:00:00 - 12:59:59	47
14:00:00 - 14:59:59	610
15:00:00 - 15:59:59	189
16:00:00 - 16:59:59	14

Production	
Wed Oct 25 2017	
11:00:00 - 11:59:59	133
12:00:00 - 12:59:59	263
14:00:00 - 14:59:59	220
15:00:00 - 15:59:59	319

RDS	
Wed Oct 25 2017	
15:00:00 - 15:59:59	25
16:00:00 - 16:59:59	263
17:00:00 - 17:59:59	246
18:00:00 - 18:59:59	210

SLAVE	
Wed Oct 25 2017	
12:00:00 - 12:59:59	386
14:00:00 - 14:59:59	2.52K
15:00:00 - 15:59:59	1.01K
16:00:00 - 16:59:59	563

SLAVE - US	
Wed Oct 25 2017	
12:00:00 - 12:59:59	115
14:00:00 - 14:59:59	1.35K
15:00:00 - 15:59:59	491
16:00:00 - 16:59:59	336

Figure 5: A graph that displays historical data for different database servers.

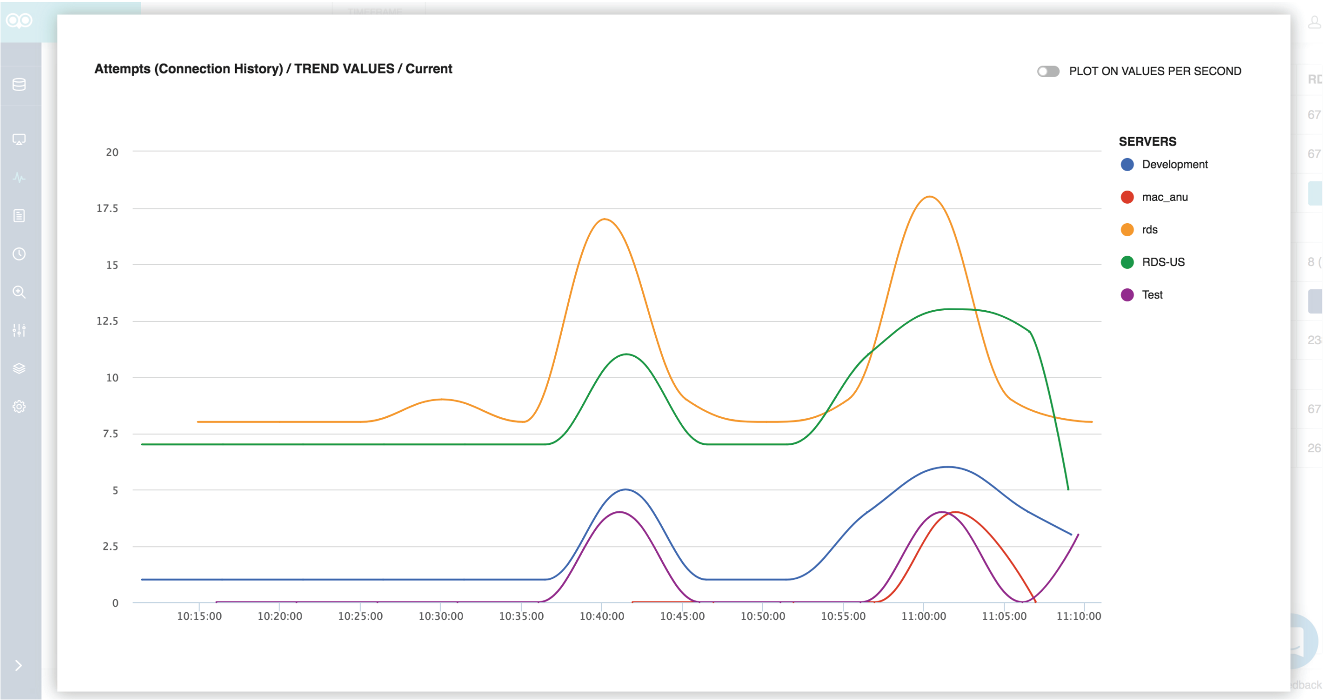

Attempts (Connection History) / TREND VALUES / Current

Figure 6: A graph that displays current metrics for different database servers.

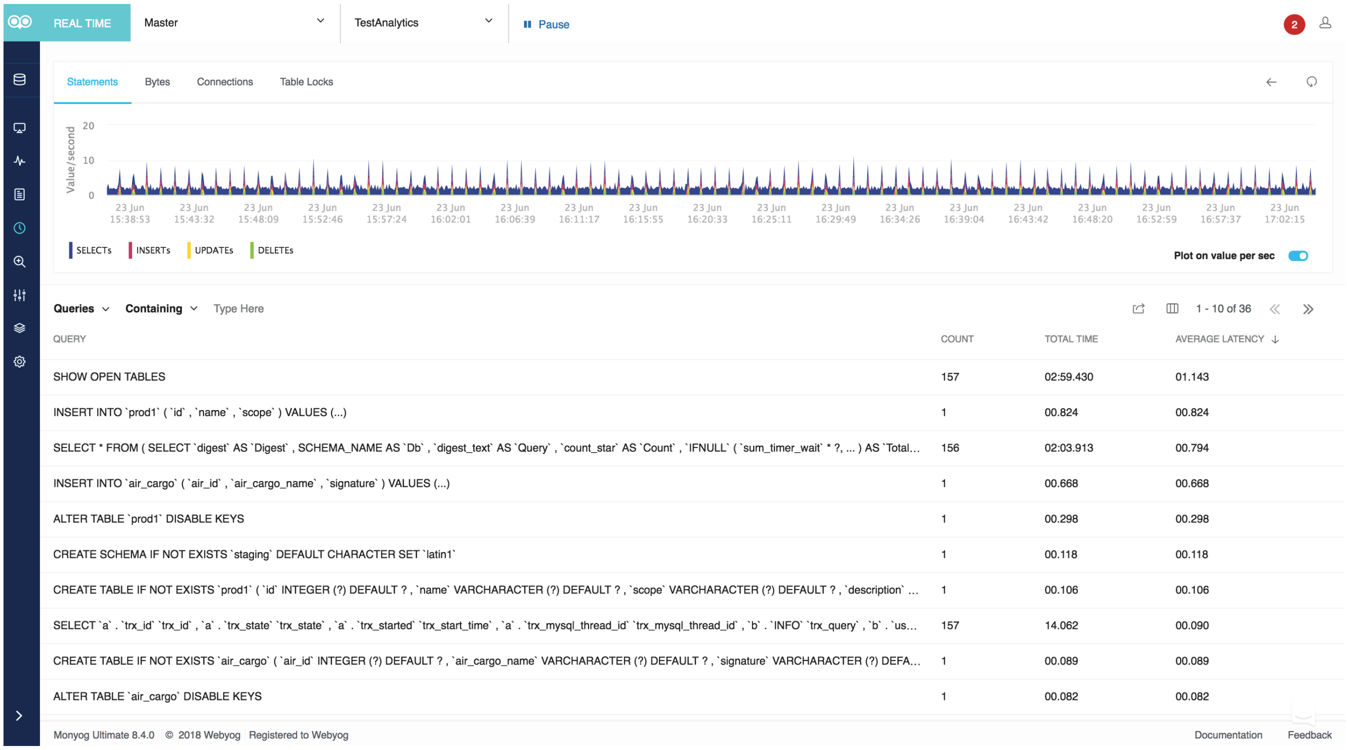
7.5 FAST STARTUP TIME TO START MONITORING

Database administrators can start monitoring MySQL and MariaDB servers in less than a single minute. The unique architecture and low-footprint of SQL Diagnostic Manager for MySQL enable database administrators to install and configure all of the components that are required for monitoring MySQL and MariaDB servers very quickly.

The fast startup time is in sharp contrast with other monitoring and advisory tools for MySQL and MariaDB. Before database administrators can even start monitoring MySQL and MariaDB servers, such tools require installing agents, web servers, multiple language runtimes, and more.

7.6 REAL-TIME MONITORING

The Real-Time feature enables database administrations to know what is happening to MySQL and MariaDB servers without delay. With a single click of a mouse button, obtain critical data (such as the top 200 SQL queries, slow SQL queries, locked and locking SQL queries, along with the most active users, hosts, databases, and tables). There is no need to enable slow query logs and general query logs. SQL Diagnostic Manager for MySQL records the data in sessions, and saves the sessions for later analysis.



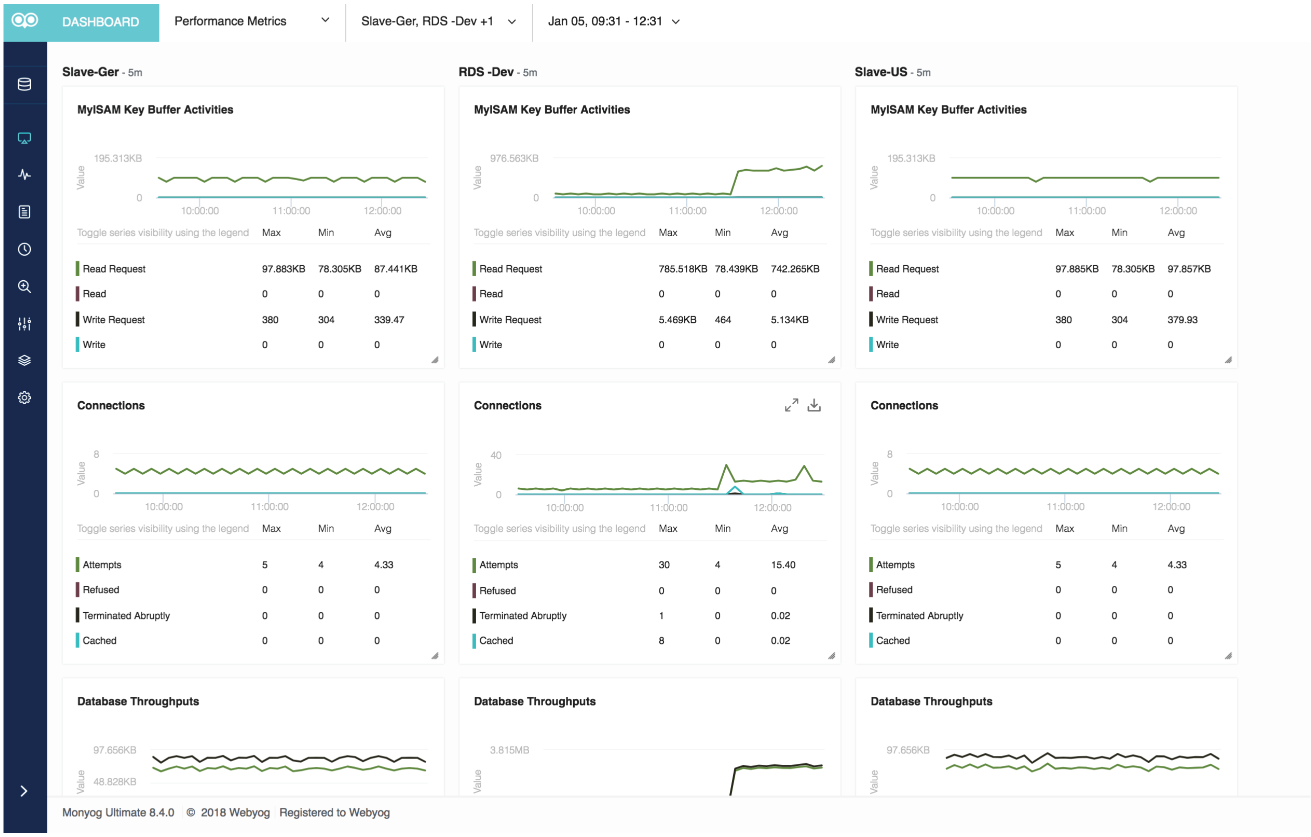
QUERY	COUNT	TOTAL TIME	AVERAGE LATENCY ↓
SHOW OPEN TABLES	157	02:59.430	01.143
INSERT INTO `prod1` (`id` , `name` , `scope`) VALUES (...)	1	00.824	00.824
SELECT * FROM (SELECT `digest` AS `Digest` , SCHEMA_NAME AS `Db` , `digest_text` AS `Query` , `count_star` AS `Count` , `IFNULL` (`sum_timer_wait` * ?, ...) AS `Total...	156	02:03.913	00.794
INSERT INTO `air_cargo` (`air_id` , `air_cargo_name` , `signature`) VALUES (...)	1	00.668	00.668
ALTER TABLE `prod1` DISABLE KEYS	1	00.298	00.298
CREATE SCHEMA IF NOT EXISTS `staging` DEFAULT CHARACTER SET `latin1`	1	00.118	00.118
CREATE TABLE IF NOT EXISTS `prod1` (`id` INTEGER (?) DEFAULT ? , `name` VARCHARACTER (?) DEFAULT ? , `scope` VARCHARACTER (?) DEFAULT ? , `description` ...	1	00.106	00.106
SELECT `a` . `trx_id` `trx_id` , `a` . `trx_state` `trx_state` , `a` . `trx_started` `trx_start_time` , `a` . `trx_mysql_thread_id` `trx_mysql_thread_id` , `b` . `INFO` `trx_query` , `b` . `us...	157	14.062	00.090
CREATE TABLE IF NOT EXISTS `air_cargo` (`air_id` INTEGER (?) DEFAULT ? , `air_cargo_name` VARCHARACTER (?) DEFAULT ? , `signature` VARCHARACTER (?) DEFA...	1	00.089	00.089
ALTER TABLE `air_cargo` DISABLE KEYS	1	00.082	00.082

Figure 7: The Real Time page.

7.7 DISPLAY DASHBOARD OVERVIEW

One of the most significant challenges database administrators face is managing an ever-growing number of database servers and databases. Regardless of the size of the database environment, each database server requires specific attention when it comes to necessary administration, security, performance monitoring, and availability. To provide database administrators for MySQL and MariaDB with a proactive advantage, SQL Diagnostic Manager for MySQL provides the Dashboard web page feature. It lets database administrators determine the cause of a performance spike by zooming in on it and viewing the SQL queries for the time frame of the spike. The design of the dashboard is such that database administrators easily understand the complete security, availability, and performance of all of their MySQL or MariaDB servers in a single place, all from a modern and intuitive web interface based on asynchronous JavaScript and XML (AJAX)

Unlike other monitoring tools for MySQL and MariaDB that use annoying web page refreshes for real-time charting, SQL Diagnostic Manager for MySQL uses charts based on JavaScript to ensure the display of true real-time charts.



Slave-Ger - 5m

MyISAM Key Buffer Activities

Toggle series visibility using the legend	Max	Min	Avg
Read Request	97.883KB	78.305KB	87.441KB
Read	0	0	0
Write Request	380	304	339.47
Write	0	0	0

Connections

Toggle series visibility using the legend	Max	Min	Avg
Attempts	5	4	4.33
Refused	0	0	0
Terminated Abruptly	0	0	0
Cached	0	0	0

Database Throughputs

RDS -Dev - 5m

MyISAM Key Buffer Activities

Toggle series visibility using the legend	Max	Min	Avg
Read Request	785.518KB	78.439KB	742.265KB
Read	0	0	0
Write Request	5.469KB	464	5.134KB
Write	0	0	0

Connections

Toggle series visibility using the legend	Max	Min	Avg
Attempts	30	4	15.40
Refused	0	0	0
Terminated Abruptly	1	0	0.02
Cached	8	0	0.02

Database Throughputs

Slave-US - 5m

MyISAM Key Buffer Activities

Toggle series visibility using the legend	Max	Min	Avg
Read Request	97.885KB	78.305KB	97.857KB
Read	0	0	0
Write Request	380	304	379.93
Write	0	0	0

Connections

Toggle series visibility using the legend	Max	Min	Avg
Attempts	5	4	4.33
Refused	0	0	0
Terminated Abruptly	0	0	0
Cached	0	0	0

Database Throughputs

Monyog Ultimate 8.4.0 © 2018 Webyog Registered to Webyog

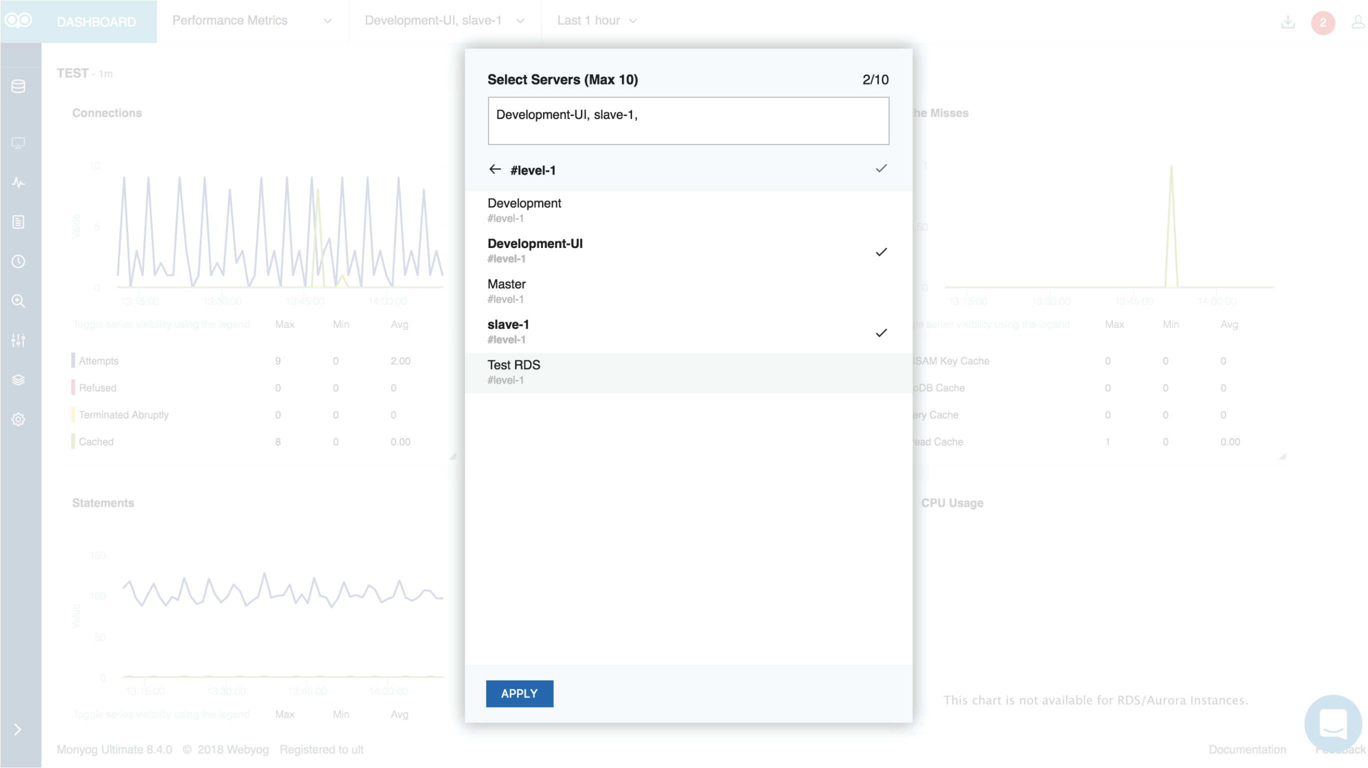
Figure 8: The Dashboard web page that displays the Performance Metrics option.

7.8 COMPARE LARGE NUMBER OF SERVERS SIDE-BY-SIDE

The Enterprise Dashboard feature shows real-time charts of all critical metrics. It provides a consolidated view of the availability and performance of all MySQL or MariaDB servers. From these real-time charts, instantly determine:

- The availability status of all database servers
- Essential operating system metrics that may be affecting database servers
- Which database servers need attention
- Where and how they need to spend their limited time

It is not rare to find database administrators who monitor hundreds of database servers. With SQL Diagnostic Manager for MySQL, managing a large number of database servers is easy. With tagging, categorize database servers into logical groups. Monitor a large number of database servers using a single instance of SQL Diagnostic Manager for MySQL.



Select Servers (Max 10) 2/10

Development-UI, slave-1,

← #level-1 ✓

Development
#level-1

Development-UI ✓
#level-1

Master
#level-1

slave-1 ✓
#level-1

Test RDS
#level-1

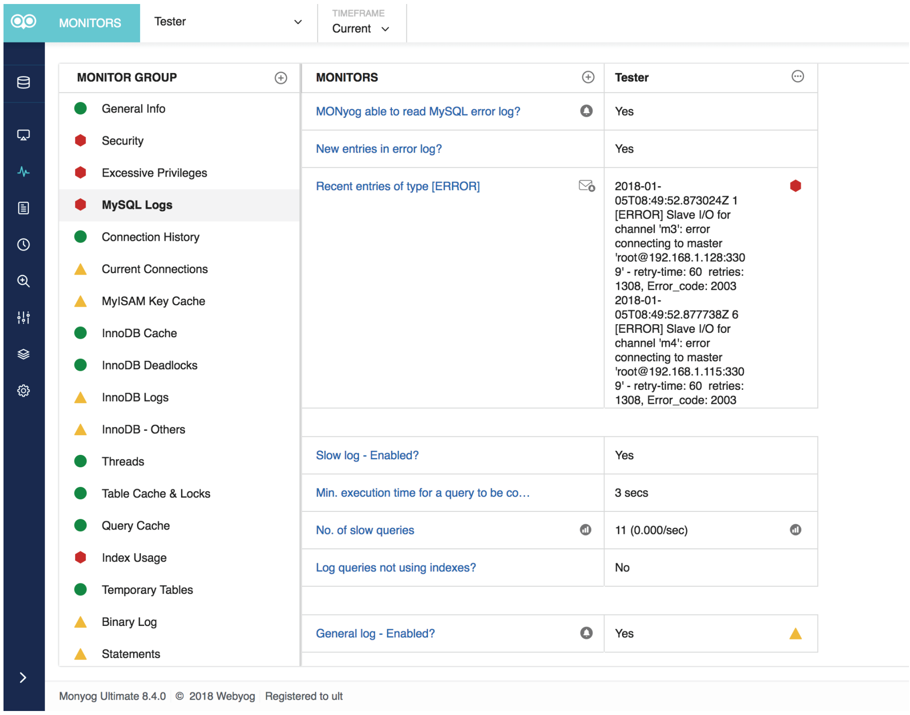
This chart is not available for RDS/Aurora Instances.

Monyog Ultimate 8.4.0 © 2018 Webyog Registered to ult

Figure 9: The Select Servers page.

7.9 MONITOR ERROR LOGS

Monitoring MySQL error logs is critical for any database administrator. SQL Diagnostic Manager for MySQL is the first monitoring tool for MySQL and MariaDB to monitor MySQL error logs. It sends notifications over simple mail transfer protocol (SMTP) and simple network management protocol (SNMP) for error log events that require attention.



MONITOR GROUP

- ● General Info
- ⬢ Security
- ⬢ Excessive Privileges
- ⬢ **MySQL Logs**
- ● Connection History
- ▲ Current Connections
- ▲ MyISAM Key Cache
- ● InnoDB Cache
- ● InnoDB Deadlocks
- ▲ InnoDB Logs
- ▲ InnoDB - Others
- ● Threads
- ● Table Cache & Locks
- ● Query Cache
- ⬢ Index Usage
- ● Temporary Tables
- ▲ Binary Log
- ▲ Statements

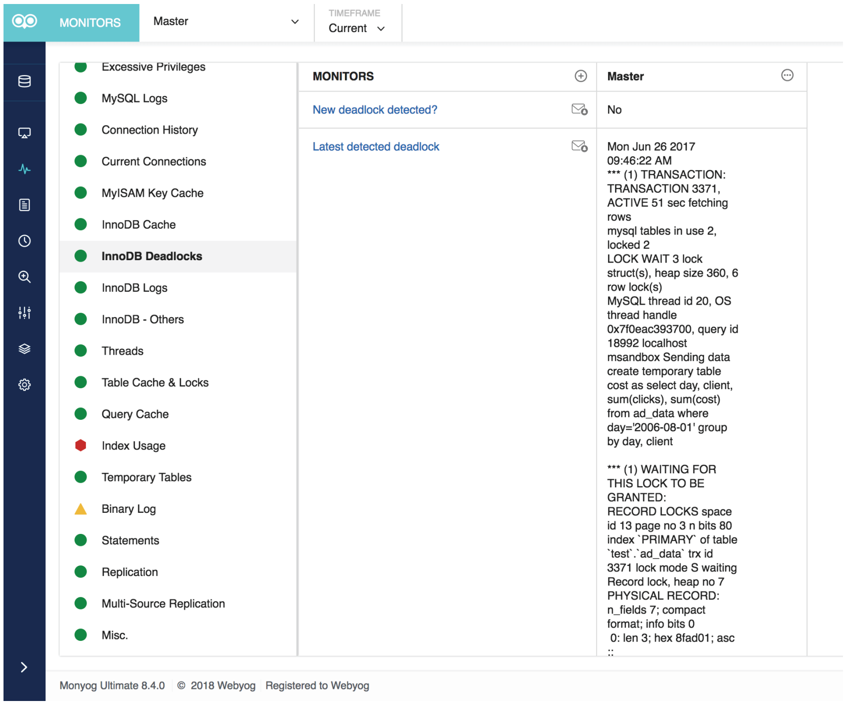
MONITORS	Tester
MONyog able to read MySQL error log?	Yes
New entries in error log?	Yes
Recent entries of type [ERROR]	2018-01-05T08:49:52.873024Z 1 [ERROR] Slave I/O for channel 'm3': error connecting to master 'root@192.168.1.128:3309' - retry-time: 60 retries: 1308, Error_code: 2003 2018-01-05T08:49:52.877738Z 6 [ERROR] Slave I/O for channel 'm4': error connecting to master 'root@192.168.1.115:3309' - retry-time: 60 retries: 1308, Error_code: 2003 ⬢
Slow log - Enabled?	Yes
Min. execution time for a query to be co…	3 secs
No. of slow queries	11 (0.000/sec)
Log queries not using indexes?	No
General log - Enabled?	Yes ▲

Monyog Ultimate 8.4.0 © 2018 Webyog Registered to ult

Figure 10: The Monitors web page that displays the MySQL Logs.

7.10 MONITOR DEADLOCKS

SQL Diagnostic Manager for MySQL monitors MySQL and MariaDB servers for deadlocks and optionally sends alerts immediately in the form of emails and simple network management protocol (SNMP) traps. In addition to detecting deadlocks, it also provides data on the latest deadlock found.



- ● Excessive Privileges
- ● MySQL Logs
- ● Connection History
- ● Current Connections
- ● MyISAM Key Cache
- ● InnoDB Cache
- ● **InnoDB Deadlocks**
- ● InnoDB Logs
- ● InnoDB - Others
- ● Threads
- ● Table Cache & Locks
- ● Query Cache
- ⬢ Index Usage
- ● Temporary Tables
- ▲ Binary Log
- ● Statements
- ● Replication
- ● Multi-Source Replication
- ● Misc.

MONITORS	Master
New deadlock detected?	No
Latest detected deadlock	Mon Jun 26 2017 09:46:22 AM *** (1) TRANSACTION: TRANSACTION 3371, ACTIVE 51 sec fetching rows mysql tables in use 2, locked 2 LOCK WAIT 3 lock struct(s), heap size 360, 6 row lock(s) MySQL thread id 20, OS thread handle 0x7f0eac393700, query id 18992 localhost msandbox Sending data create temporary table cost as select day, client, sum(clicks), sum(cost) from ad_data where day='2006-08-01' group by day, client *** (1) WAITING FOR THIS LOCK TO BE GRANTED: RECORD LOCKS space id 13 page no 3 n bits 80 index `PRIMARY` of table `test`.`ad_data` trx id 3371 lock mode S waiting Record lock, heap no 7 PHYSICAL RECORD: n_fields 7; compact format; info bits 0 0: len 3; hex 8fad01; asc ::

Monyog Ultimate 8.4.0 © 2018 Webyog Registered to Webyog

Figure 11: The Monitors web page that displays data for the InnoDB Deadlocks.

7.11 VIEW DISK METRICS

The Disk Space Usage Analyzer feature displays the size of data and the size of indexes of the databases on the servers. It also displays a graphical chart to identify the largest databases quickly. It also allows to drill down to the table level.

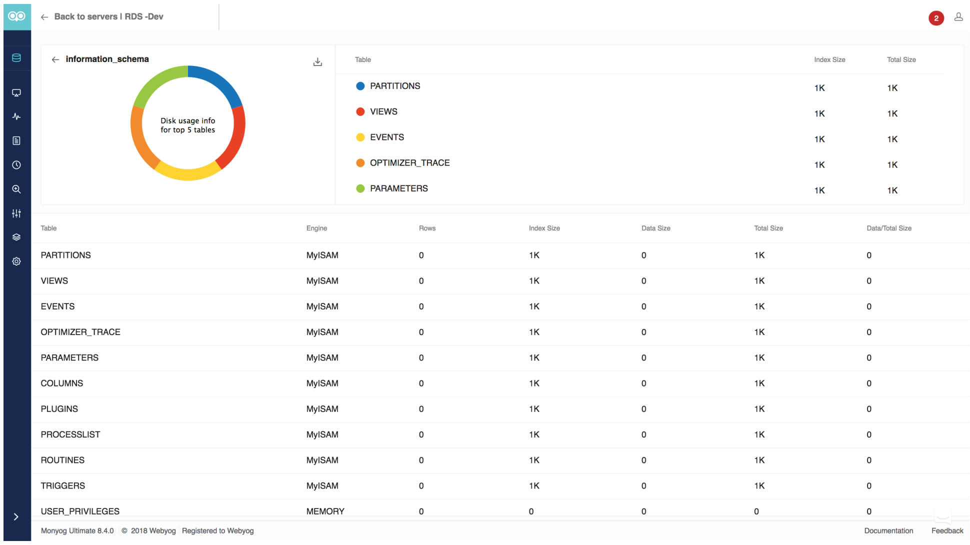


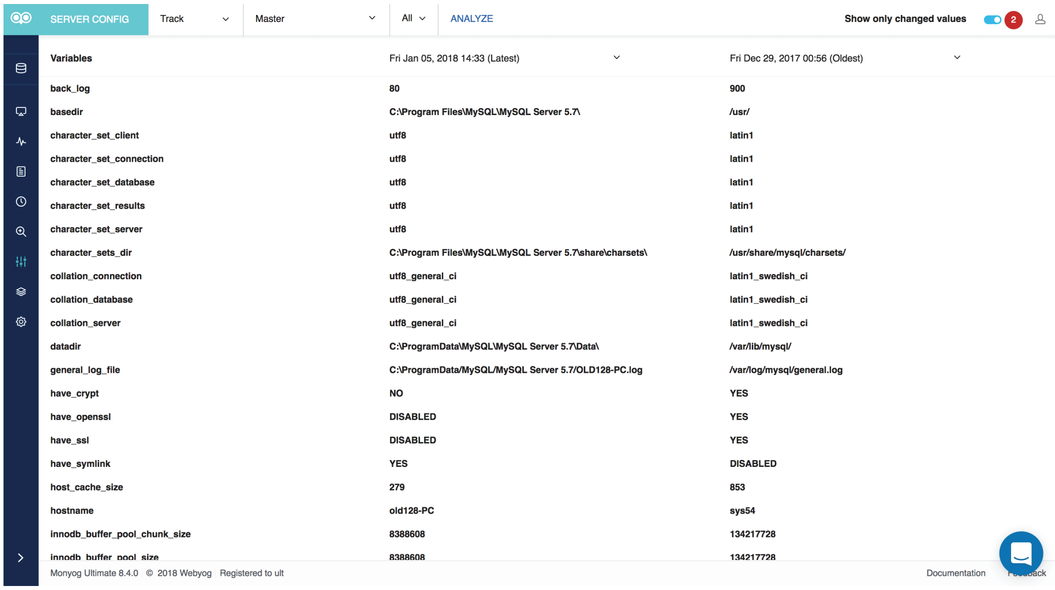
Table	Index Size	Total Size
● PARTITIONS	1K	1K
● VIEWS	1K	1K
● EVENTS	1K	1K
● OPTIMIZER_TRACE	1K	1K
● PARAMETERS	1K	1K

Table	Engine	Rows	Index Size	Data Size	Total Size	Data/Total Size
PARTITIONS	MyISAM	0	1K	0	1K	0
VIEWS	MyISAM	0	1K	0	1K	0
EVENTS	MyISAM	0	1K	0	1K	0
OPTIMIZER_TRACE	MyISAM	0	1K	0	1K	0
PARAMETERS	MyISAM	0	1K	0	1K	0
COLUMNS	MyISAM	0	1K	0	1K	0
PLUGINS	MyISAM	0	1K	0	1K	0
PROCESSLIST	MyISAM	0	1K	0	1K	0
ROUTINES	MyISAM	0	1K	0	1K	0
TRIGGERS	MyISAM	0	1K	0	1K	0
USER_PRIVILEGES	MEMORY	0	0	0	0	0

Figure 12: The Disk Space Usage Analyzer.

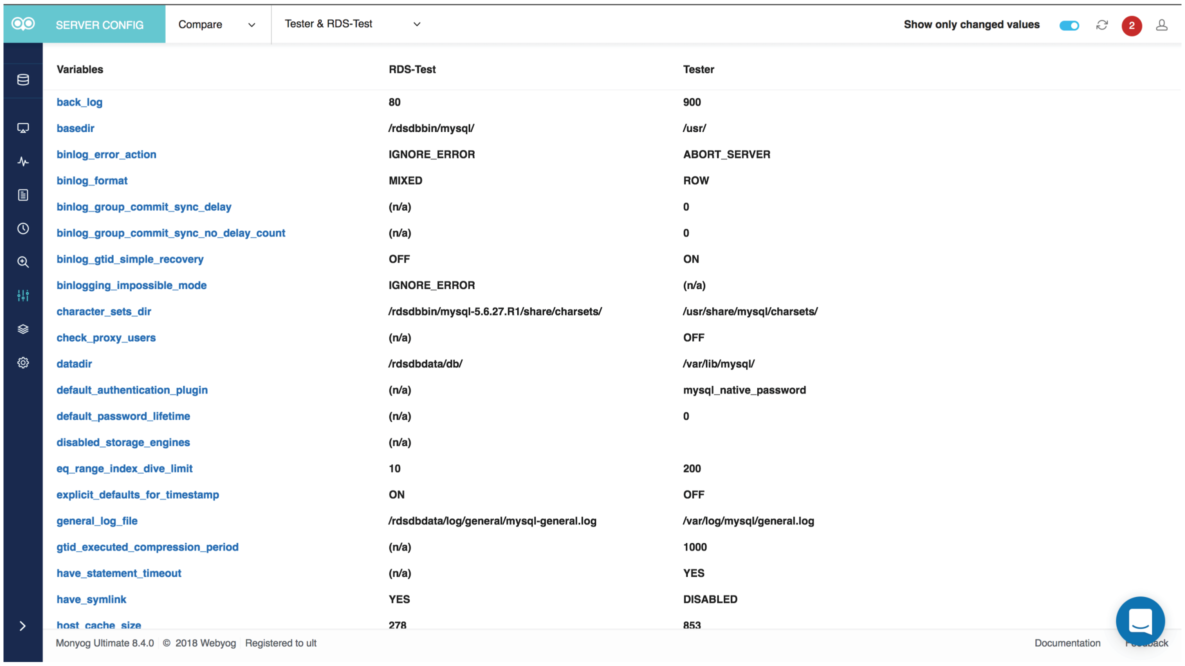
7.12 CONFIGURE SERVERS

Maintaining and tracking changes to the configuration of MySQL and MariaDB servers is vital for the maintenance of the database servers. The Server Config feature compares the configurations of multiple database servers side-by-side, with all changes highlighted. The Server Config feature also tracks changes to the configuration files of database servers over a period to have full control of what goes into those files and the impact they have on the database servers.



Variables	Fri Jan 05, 2018 14:33 (Latest)	Fri Dec 29, 2017 00:56 (Oldest)
back_log	80	900
basedir	C:\Program Files\MySQL\MySQL Server 5.7\	/usr/
character_set_client	utf8	latin1
character_set_connection	utf8	latin1
character_set_database	utf8	latin1
character_set_results	utf8	latin1
character_set_server	utf8	latin1
character_sets_dir	C:\Program Files\MySQL\MySQL Server 5.7\share\charsets\	/usr/share/mysql/charsets/
collation_connection	utf8_general_ci	latin1_swedish_ci
collation_database	utf8_general_ci	latin1_swedish_ci
collation_server	utf8_general_ci	latin1_swedish_ci
datadir	C:\ProgramData\MySQL\MySQL Server 5.7\Data\	/var/lib/mysql/
general_log_file	C:\ProgramData/MySQL/MySQL Server 5.7/OLD128-PC.log	/var/log/mysql/general.log
have_crypt	NO	YES
have_openssl	DISABLED	YES
have_ssl	DISABLED	YES
have_symlink	YES	DISABLED
host_cache_size	279	853
hostname	old128-PC	sys54
innodb_buffer_pool_chunk_size	8388608	134217728
innodb_buffer_pool_size	8388608	134217728

Figure 13: The Server Config web page displaying the Track option.

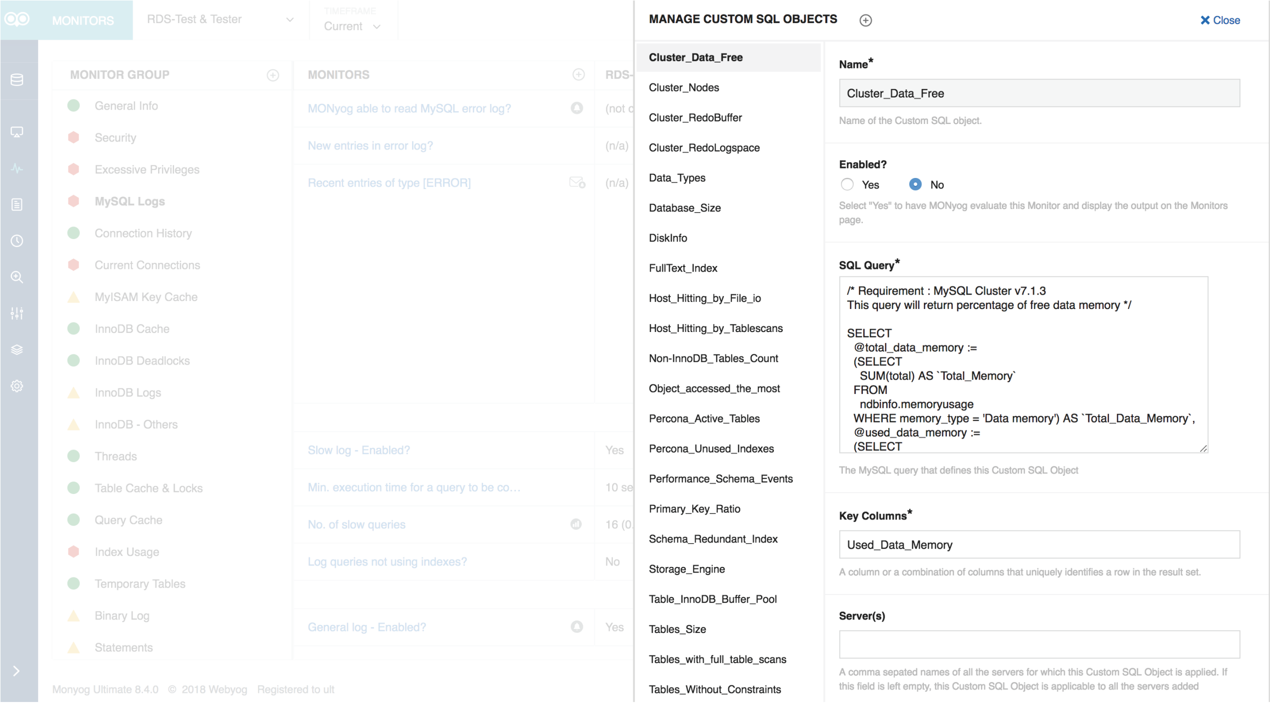


Variables	RDS-Test	Tester
back_log	80	900
basedir	/rdsdbbin/mysql/	/usr/
binlog_error_action	IGNORE_ERROR	ABORT_SERVER
binlog_format	MIXED	ROW
binlog_group_commit_sync_delay	(n/a)	0
binlog_group_commit_sync_no_delay_count	(n/a)	0
binlog_gtid_simple_recovery	OFF	ON
binlogging_impossible_mode	IGNORE_ERROR	(n/a)
character_sets_dir	/rdsdbbin/mysql-5.6.27.R1/share/charsets/	/usr/share/mysql/charsets/
check_proxy_users	(n/a)	OFF
datadir	/rdsdbdata/db/	/var/lib/mysql/
default_authentication_plugin	(n/a)	mysql_native_password
default_password_lifetime	(n/a)	0
disabled_storage_engines	(n/a)	
eq_range_index_dive_limit	10	200
explicit_defaults_for_timestamp	ON	OFF
general_log_file	/rdsdbdata/log/general/mysql-general.log	/var/log/mysql/general.log
gtid_executed_compression_period	(n/a)	1000
have_statement_timeout	(n/a)	YES
have_symlink	YES	DISABLED
host_cache_size	278	853

Figure 14: The Server Config web page displays the Compare option.

7.13 CREATE CUSTOM SQL OBJECTS

Instead of monitoring MySQL and MariaDB servers by writing SQL queries, create Custom SQL Objects. Custom SQL Objects return an array of MySQL rows. SQL Diagnostic Manager for MySQL exposes these rows as a JavaScript array, monitors it, and references it like any SQL Diagnostic Manager for MySQL object.



MONITORS	RDS-
MONyog able to read MySQL error log?	(not c
New entries in error log?	(n/a)
Recent entries of type [ERROR]	(n/a)
Slow log - Enabled?	Yes
Min. execution time for a query to be co…	10 se
No. of slow queries	16 (0.
Log queries not using indexes?	No
General log - Enabled?	Yes

MANAGE CUSTOM SQL OBJECTS ⊕ ✕ Close

- Cluster_Data_Free
- Cluster_Nodes
- Cluster_RedoBuffer
- Cluster_RedoLogspace
- Data_Types
- Database_Size
- DiskInfo
- FullText_Index
- Host_Hitting_by_File_io
- Host_Hitting_by_Tablescans
- Non-InnoDB_Tables_Count
- Object_accessed_the_most
- Percona_Active_Tables
- Percona_Unused_Indexes
- Performance_Schema_Events
- Primary_Key_Ratio
- Schema_Redundant_Index
- Storage_Engine
- Table_InnoDB_Buffer_Pool
- Tables_Size
- Tables_with_full_table_scans
- Tables_Without_Constraints

Name*

Cluster_Data_Free

Name of the Custom SQL object.

Enabled?

○ Yes ● No

Select "Yes" to have MONyog evaluate this Monitor and display the output on the Monitors page.

SQL Query*

```
/* Requirement : MySQL Cluster v7.1.3
This query will return percentage of free data memory */

SELECT
 @total_data_memory :=
 (SELECT
  SUM(total) AS `Total_Memory`
 FROM
  ndbinfo.memoryusage
 WHERE memory_type = 'Data memory') AS `Total_Data_Memory`,
 @used_data_memory :=
 (SELECT
```

The MySQL query that defines this Custom SQL Object

Key Columns*

Used_Data_Memory

A column or a combination of columns that uniquely identifies a row in the result set.

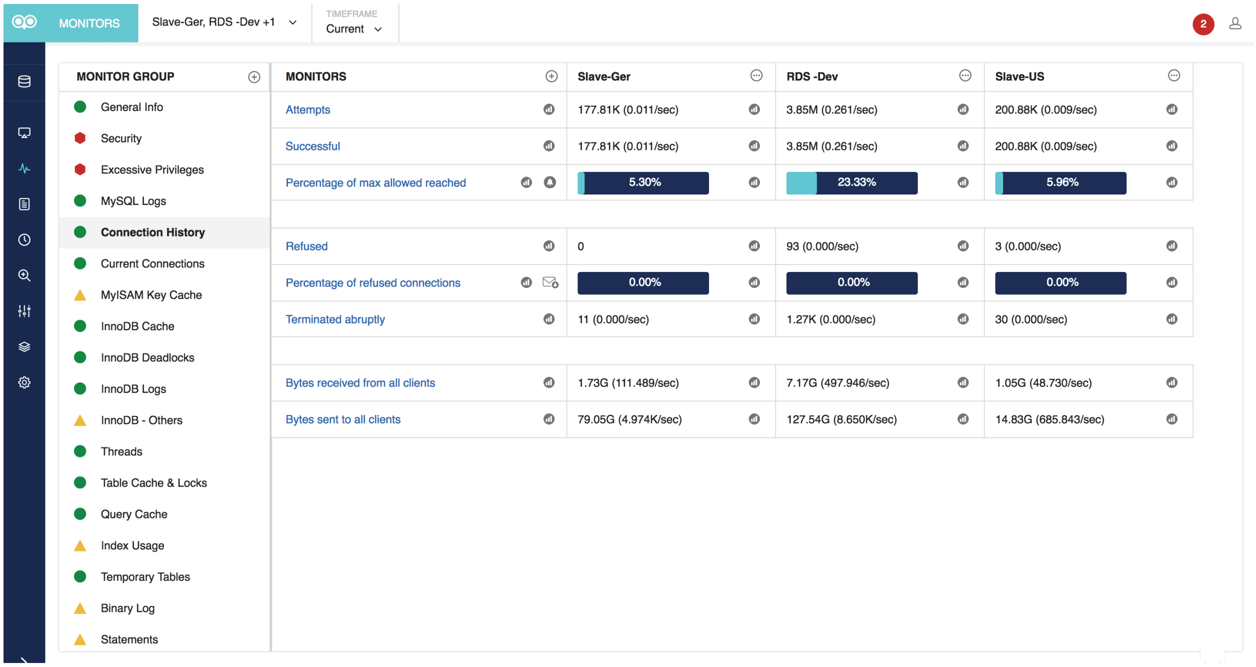
Server(s)

A comma sepated names of all the servers for which this Custom SQL Object is applied. If this field is left empty, this Custom SQL Object is applicable to all the servers added

Figure 15: The Manage Custom SQL Objects web page displaying data for the Cluster_Data_Free object.

7.14 FIX PROBLEMS PROACTIVELY WITH HUNDREDS OF MONITORS

SQL Diagnostic Manager for MySQL includes hundreds of monitors that are designed to examine the configuration and security of MySQL and MariaDB servers automatically, identify problems and tuning opportunities, and provides database administrators with specific corrective actions.



MONITOR GROUP

- ● General Info
- ⬢ Security
- ⬢ Excessive Privileges
- ● MySQL Logs
- ● **Connection History**
- ● Current Connections
- ▲ MyISAM Key Cache
- ● InnoDB Cache
- ● InnoDB Deadlocks
- ● InnoDB Logs
- ▲ InnoDB - Others
- ● Threads
- ● Table Cache & Locks
- ● Query Cache
- ▲ Index Usage
- ● Temporary Tables
- ▲ Binary Log
- ▲ Statements

MONITORS	Slave-Ger	RDS -Dev	Slave-US
Attempts	177.81K (0.011/sec)	3.85M (0.261/sec)	200.88K (0.009/sec)
Successful	177.81K (0.011/sec)	3.85M (0.261/sec)	200.88K (0.009/sec)
Percentage of max allowed reached	5.30%	23.33%	5.96%
Refused	0	93 (0.000/sec)	3 (0.000/sec)
Percentage of refused connections	0.00%	0.00%	0.00%
Terminated abruptly	11 (0.000/sec)	1.27K (0.000/sec)	30 (0.000/sec)
Bytes received from all clients	1.73G (111.489/sec)	7.17G (497.946/sec)	1.05G (48.730/sec)
Bytes sent to all clients	79.05G (4.974K/sec)	127.54G (8.650K/sec)	14.83G (685.843/sec)

Figure 16: The Monitors web page displaying the Connection History group of monitors.

7.15 USE ADVISOR RULES

The Advisor Rules feature is a set of best practices that enables database administrators to monitor MySQL and MariaDB servers with confidence and to proactively manage the dynamic nature of all of their database servers over time. The Advisor Rules feature monitors all database servers. It notifies database administrators with specific instructions on how to proactively address the problems found to align with the best practices.

7.16 AUTOMATE ADVISOR RULES TO INCREASE PRODUCTIVITY

SQL Diagnostic Manager for MySQL makes database administrators more productive by enabling them to automate each of the Advisor Rules for unattended, around the clock operations. The automation minimizes human errors, improves overall productivity, and lowers the total cost associated with managing MySQL and MariaDB servers.

7.17 RECEIVE PROACTIVE ALERTS

For all the monitors, Advisor Rule violations trigger notification events. SQL Diagnostic Manager for MySQL sends these notifications via simple mail transfer protocol (SMTP), simple network management protocol (SNMP), the collaboration platform Slack, the incident response platform PagerDuty, and the messaging logging standard syslog. It also provides expert advice on the reported problem.

The ‘Delayed alert notifications’ feature ensures that SQL Diagnostic Manager for MySQL does not send alerts for insignificant events. Define that a problem must have existed for some sample intervals continuously in a row before sending an alert. A global setting for each MySQL and MariaDB server is available from the graphical user interface (GUI).

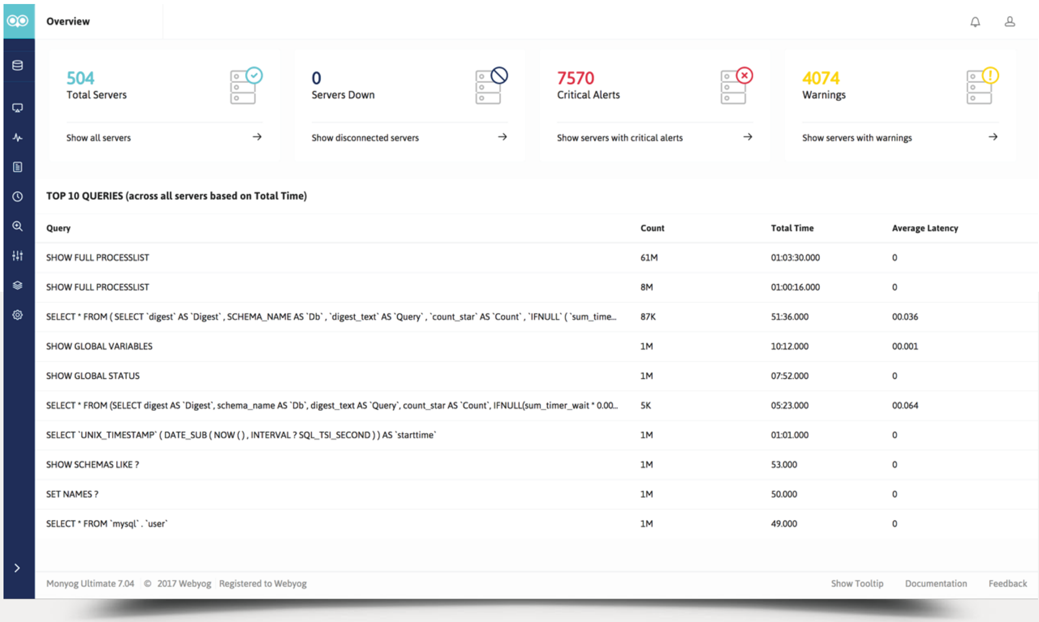
8. CONCLUSION

Serving as a 'MySQL database administrator in a box', SQL Diagnostic Manager for MySQL proactively monitors all of the MySQL and MariaDB servers across the enterprise. It empowers database administrators to address specific problems and tuning opportunities before problems start surfacing. The combination of enterprise visibility, proactive monitoring and expert advice and guidance in problem identification and resolution makes SQL Diagnostic Manager for MySQL the perfect addition for managing and tuning MySQL and MariaDB servers.

SQL DIAGNOSTIC MANAGER FOR MYSQL

Achieve Comprehensive 24/7 MySQL and MariaDB Monitoring

- Monitor in real-time
- Track configuration changes
- Monitor locked and long-running queries
- Monitor Amazon RDS
- Find top 10 SQL queries across servers



Overview

504 Total Servers — Show all servers →

0 Servers Down — Show disconnected servers →

7570 Critical Alerts — Show servers with critical alerts →

4074 Warnings — Show servers with warnings →

TOP 10 QUERIES (across all servers based on Total Time)

Query	Count	Total Time	Average Latency
SHOW FULL PROCESSLIST	61M	01:03:30.000	0
SHOW FULL PROCESSLIST	8M	01:00:16.000	0
SELECT * FROM (SELECT `digest` AS `Digest` , SCHEMA_NAME AS `Db` , `digest_text` AS `Query` , `count_star` AS `Count` , `IFNULL` (`sum_time...	87K	51:36.000	00.036
SHOW GLOBAL VARIABLES	1M	10:12.000	00.001
SHOW GLOBAL STATUS	1M	07:52.000	0
SELECT * FROM (SELECT digest AS `Digest`, schema_name AS `Db`, digest_text AS `Query`, count_star AS `Count`, IFNULL(sum_timer_wait * 0.00...	5K	05:23.000	00.064
SELECT `UNIX_TIMESTAMP` (DATE_SUB (NOW ( ) , INTERVAL ? SQL_TSI_SECOND)) AS `starttime`	1M	01:01.000	0
SHOW SCHEMAS LIKE ?	1M	53.000	0
SET NAMES ?	1M	50.000	0
SELECT * FROM `mysql` . `user`	1M	49.000	0

Monyog Ultimate 7.04 | © 2017 Webyog | Registered to Webyog

START FOR FREE

IDERA

IDERA.com

877 GO IDERA 464.3372
TWITTER twitter.com/Idera_Software
FACEBOOK facebook.com/IderaSoftware
LINKEDIN linkedin.com/company/idera-software

EMEA +44 (0) 1753 218410
APAC +61 1300 307 211
MEXICO +52 (55) 8421 6770
BRAZIL +55 (11) 3230 7938