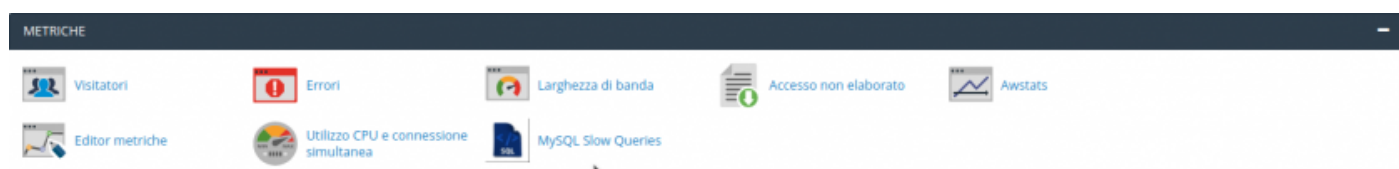


# MySQL Slow Queries

MySQL Slow Queries è un esclusivo strumento sviluppato da Artera per monitorare le operazioni eseguite sul database, in modo da identificare in modo immediato le eventuali richieste responsabili di rallentamenti sul portale. I tempi di esecuzione infatti, possono incidere sui tempi di risposta del sito.

In questa guida illustreremo come utilizzare questo strumento.

Prima di tutto è necessario accedere al proprio pannello di controllo visitando l'indirizzo <https://mail.DOMINIO:2083> con proprio browser internet, sostituendo a DOMINIO il dominio del vostro sito internet senza www, ed entrare nella sezione "MySQL Slow Queries" che trovate nel blocco "Metriche".



Una volta dentro avrete subito l'elenco delle query più lente lanciate sul database. Questa lista è compilata con le operazioni che hanno richiesto più di 2 secondi per essere eseguite.

## MySQL Slow Queries

Questa funzione consente di visualizzare le query al database MySQL che hanno impiegato più tempo nell'esecuzione. Può essere molto utile per individuare le cause di lentezza dei siti.

☐ Raggruppa query simili

Dimensioni pagina 20 << < 1 2 3 4 > >>

Utente	Database	Tempo query	Tempo lock	Righe inviate	Righe esaminate	Data
cancprov_AtnMfV	cancprov_AtnM	238,656 s	0,006 s	33.080	1.095.609.644	19/9/2018, 13:34:01
SELECT `log_customer_main`.`log_id` FROM `log_customer` AS `log_customer_main` LEFT JOIN `log_customer` ON log_customer_main.customer_id = log_customer.customer_id AND log_customer_main.log_id < log_customer.log_id WHERE (log_customer.customer_id IS NULL) AND (log_customer_main.log_id < 33549);						
cancprov_AtnMfV	cancprov_AtnM	229,862 s	0 s	33.078	1.095.708.811	18/9/2018, 13:33:52
SELECT `log_customer_main`.`log_id` FROM `log_customer` AS `log_customer_main` LEFT JOIN `log_customer` ON log_customer_main.customer_id = log_customer.customer_id AND log_customer_main.log_id < log_customer.log_id WHERE (log_customer.customer_id IS NULL) AND (log_customer_main.log_id < 33543);						
cancprov_AtnMfV	cancprov_AtnM	10,13 s	0 s	0	77.112	18/9/2018, 00:01:04
DELETE FROM `catalogrule_product_price` WHERE (website_id = '1');						
cancprov_AtnMfV	cancprov_AtnM	235,294 s	0 s	33.078	1.095.543.411	16/9/2018, 13:33:59
SELECT `log_customer_main`.`log_id` FROM `log_customer` AS `log_customer_main` LEFT JOIN `log_customer` ON log_customer_main.customer_id = log_customer.customer_id AND log_customer_main.log_id < log_customer.log_id WHERE (log_customer.customer_id IS NULL) AND (log_customer_main.log_id < 33538);						

L'interfaccia di MySQL Slow Queries è molto semplice ed intuitiva:

- le colonne "Utente" e "Database" indicano l'utente che ha avviato l'operazione e il database su cui è stata eseguita;
- "Tempo query" indica il tempo di esecuzione effettiva;

- "Tempo lock" indica il tempo di attesa che ha dovuto aspettare la query prima di essere effettivamente eseguita. Questa situazione può capitare se è già in corso una operazione che interessa le stesse tabelle interpellate da un'altra query;
- "Righe inviate" mostra il risultato dell'operazione, quindi il numero delle righe restituite alla sua conclusione;
- "Righe esaminate" mostra il numero delle righe che sono state esaminate dalla query per restituire il risultato;
- "Data" indica la data e l'ora di esecuzione. Come potete vedere nella figura di esempio l'ordinamento è stato valutato in base a questo parametro, ma è possibile modificarlo cliccando sul nome della colonna di riferimento;

Con questa prima vista è già possibile identificare le richieste più lente e quindi le cause di possibili rallentamenti riscontrati sul portale. Questo permette di capire se ci sono problemi o se si tratta semplicemente di operazioni troppo pesanti che possono essere eventualmente ottimizzate per snellire il processo e ridurre i tempi di esecuzione.

Spuntando la voce "Raggruppa query simili" è possibile intervenire sull'elenco per raggruppare tutte le operazioni dello tipo.

MySQL Slow Queries

Questa funzione consente di visualizzare le query ai database MySQL che hanno impiegato più tempo nell'esecuzione. Può essere molto utile per individuare le cause di lentezza dei siti.

☒ Raggruppa query simili

Dimensioni pagina 20 << < 1 2 3 4 > >>

Utente	Database	Nr.	Tempo query	Tempo lock	Righe inviate	Righe esaminate	Data ▲
cancprov_AtnMFV	cancprov_AtnM	50	164,08 s	0,002 s	25.606,88	756.001.515,76	19/9/2018, 13:34:01
SELECT `log_customer_main`.`log_id` FROM `log_customer` AS `log_customer_main` LEFT JOIN `log_customer` ON log_customer_main.customer_id = log_customer.customer_id AND log_customer_main.log_id < log_customer.log_id WHERE (log_customer.customer_id IS NULL) AND (log_customer_main.log_id < 33549);							
cancprov_AtnMFV	cancprov_AtnM	6	8,947 s	0 s	0	77.144	18/9/2018, 00:01:04
DELETE FROM `catalogrule_product_price` WHERE (website_id = '1');							
cancprov_AtnMFV	cancprov_AtnM	2	3,209 s	0 s	3.472	3.472	16/9/2018, 00:01:57
SELECT /*#140001 SQL_NO_CACHE */ * FROM `wp_posts`;							
cancprov_AtnMFV	cancprov_AtnM	4	3,109 s	0 s	26.587	26.587	15/9/2018, 19:30:06
SELECT `ID`, `post_author`, `post_date`, `post_date_gmt`, `post_content`, `post_title`, `post_excerpt`, `post_status`, `comment_status`, `ping_status`, `post_password`, `post_name`, `to_ping`, `pinged`, `post_modified`, `post_modified_gmt`, `post_content_filtered`, `post_parent`, `guid`, `menu_order`, `post_type`, `post_mime_type`, `comment_count` FROM `homeayit_cgjsPU`.`ha_posts`;							

Questa funzione è molto utile per avere evidenza di quante volte viene eseguita una query e capire quindi se si tratta di una operazione ricorrente su cui vale la pena intervenire. A differenza della vista precedente, il sistema aggiunge una colonna in più "Nr.", che mostra il numero di volte che query simili a quella indicata sono state eseguite.

Con il raggruppamento i dati mostrati nelle varie colonne sono una media di tutte le query conteggiate, non sono la somma totale delle cifre relative alle richieste raggruppate.