

Tuning MySQL parameters for HP Output Server

This technical note provides information about how to tune various MySQL server parameters for HP Output Server.

Platforms affected

All platforms

Problem

The external physical destination hangs when attribute modification fails at the end of job processing.

Solution

HP Output Server 3.5 is an InnoDB-oriented application. You should tune the default values of MySQL server tunable parameters for better performance, stability, and reliability.

Tuning with or without defragmentation of databases

HP recommends tuning MySQL with defragmentation of databases because it helps to improve performance. Before tuning MySQL, compare the value of the `innodb_data_file_path ibdata1: 250M: autoextend` parameter in the `my.cnf` file, which indicates the currently set size of the `ibdata1` file, with the actual size of the `ibdata1` file.

If the actual size of the `ibdata1` file is larger than the value set in the `my.cnf` file, then the database is defragmented. Proceed with “MySQL tuning with defragmentation of databases” on page 2.

If your database is not defragmented, follow the instructions in “MySQL tuning without defragmentation of databases” on page 3.

NOTE: With MySQL tuning, if you change the value of the `-database-client-max-connections` attribute to 25 on delivery servers and the CM server, performance should improve significantly.

Solution

MySQL tuning with defragmentation of databases

The advantages and disadvantages of tuning with defragmentation of databases are as follows:

Advantages	<ul style="list-style-type: none"> • Increases the performance of reads per second • Decreases the response time • Configures indexes • Increases the query performance • Cleans temporary data in the datafiles and undo segments
Disadvantages	<ul style="list-style-type: none"> • Re-creates the database • Corrupts the database if the re-creation of the database fails

You can tune MySQL with defragmentation of databases as follows:

1. Stop HP Output Server servers.
2. Execute the following command to back up the databases from the `mysqlInstallDir/bin` directory:

```
./mysqldump --defaults-file=mysqlInstallDir/my.cnf --quote-names
--add-drop-table --quick --extended-insert --all-databases
--result-file=<dump_file>
```

3. Stop the MySQL server.
4. Make a backup of the data directory.
5. Start the MySQL server.
6. Create a script with the following commands, which will extract the database names and execute the drop database command on all the existing databases except MySQL internal databases:

```
./bin/mysql --defaults-file=./my.cnf -e " show databases;" | \
sed 's/\\|//g' | \
sed '/^+/d' | \
grep -v Database | \
grep domain name | \
while read db_name
do
./bin/mysql --defaults-file=./my.cnf -e "drop database ``$db_name``"
done
```

7. Shut down the MySQL server.
8. Remove the `ibdata` and `iblog` files.
9. Append the following lines in the `my.cnf` file to tune the MySQL parameters:

```
table_cache=250
innodb_buffer_pool_size=150M
innodb_additional_mem_pool_size=22M
read_buffer_size=18M
read_rnd_buffer_size=18M
innodb_log_file_size=30M
innodb_log_buffer_size=20M
innodb_sync_spin_loops=120 *
innodb_thread_concurrency=18 *
```

```
query_cache_size=800K
innodb_data_file_path ibdata1: 250M: autoextend
innodb_autoextend_increment=25
```

NOTE: Execute the following command to get the values of the parameters marked *:
 innodb_sync_spin_loops= 10 * Number of CPUs
 innodb_thread_concurrency=2 * (Number of CPUs + Number of disks in use for MySQL)

For more information, see the following topics at the end of this technical note: “Parameters that affect memory,” “Parameters that affect disk,” and “Parameters that affect CPU.”

NOTE: If the system does not have sufficient memory, the MySQL server will not start.

10. Start the MySQL server.
11. Ensure that the size of the file created in the `$MYSQL_HOME/data/ibdata1` is approximately 250 MB.

NOTE: See the *Installation and Configuration Guide* for MySQL for HP Output Management 5.0.24p2 for the formula on data file sizing.

12. Execute the following command to check if the settings have taken effect:

```
./mysql --defaults-file=mysqlInstallDir/my.cnf -e " show variables
\"%inno%\" "
```

13. Execute the following command to load a dumpfile:

```
./mysql --defaults-file=mysqlInstallDir/my.cnf < <dump_file> >
```

MySQL tuning without defragmentation of databases

If the size of the `ibdata1` file is less than the value set in the `my.cnf` file, then the database is not defragmented and only MySQL tuning is recommended.

The advantages and disadvantages of tuning without defragmentation of databases are as follows:

Advantages	<ul style="list-style-type: none"> • Eliminates the risk involved in re-creating a database • Helps tune memory pools and increases query performance
Disadvantages	None

You can tune MySQL without defragmentation of databases as follows:

1. Stop HP Output Server servers.
2. Stop the MySQL server.
3. Append the following lines in the `my.cnf` file to tune the MySQL parameters:

```
table_cache=250
innodb_buffer_pool_size=150M
innodb_additional_mem_pool_size=22M
read_buffer_size=18M
read_rnd_buffer_size=18M
```

```
innodb_log_buffer_size=20M
innodb_sync_spin_loops=120 *
innodb_thread_concurrency=18 *
query_cache_size=800K
```

NOTE: Execute the following command to get the values of the parameters marked *:

```
innodb_sync_spin_loops=10 * Number of CPUs
innodb_thread_concurrency=2 * (Number of CPUs + Number of disks in use
for MySQL)
```

For more information, see the following topics in the current technical note: “Parameters that affect memory,” “Parameters that affect disk,” and “Parameters that affect CPU.”

NOTE: If the system does not have sufficient memory, the MySQL server will not start.

4. Start the MySQL server.
5. Start HP Output Server servers.

MySQL tunable parameters

The following sections describe the MySQL tunable parameters.

Parameters that affect memory

- `innodb_buffer_pool_size`

This is the size of the memory chunk used by InnoDB to store the index and related data in memory. It should be large enough for all operations on InnoDB tables.

This parameter is dependent on the size of the biggest table in MySQL server databases.

- `innodb_log_buffer_size`

This is the size of the buffer that InnoDB uses to write to the log files on disk. Values typically range from 1 MB to half the combined size of log files. A large log buffer allows large transactions to run without a need to write the log to disk until the transaction is committed. Therefore, a value of 25 to 30 percent of buffer pool size is required.

- `innodb_additional_mem_pool_size`

This parameter is used to hold the internal structure and the hash table. It must be approximately 50 percent of the `innodb_buffer_pool_size`.

- `read_buffer_size`

This parameter helps to load a huge amount of data from the disk for operations such as `pdq` and `pdls`. Increase the value by 20 to 50 percent of 128 KB (the current value). This parameter helps in table scans.

- `read_rnd_buffer_size`

This parameter helps in `pdls` queries. During `pdls` queries, data is selected randomly and not through sequential queries. Increase the value by 20 to 50 percent of 256 KB (the current value).

Parameters that affect CPU

- `innodb_sync_spin_loops`

This parameter balances the CPU time wasted while running spin locks with unnecessary context switches. If the spin lock setting is high, the CPU time wasted will increase. Alternatively, if spin lock setting is low, it results in unnecessary context switches.

- `innodb_thread_concurrency`

This parameter increases the InnoDB kernel execution threads, which use the CPU efficiently.

You can further increase the performance by setting the `innodb_flush_logs_at_trx_commit` parameter to 0 instead of 1 (1 is the default value).

CAUTION: This could lead to disk or operating system crash and, therefore, a loss of data.

The recommended memory setting for a MySQL database server is 256 MB. However, large enterprise installations can benefit from having a second `my.cnf` file with the tuning parameter for the MySQL server set to 512 MB.

Parameters that affect disk

- `innodb_data_file_path ibdata1`

This parameter specifies the value in the `ibdata1` file.

- `innodb_autoextend_increment`

This parameter specifies the amount of disk space to be extended to the current tablespace. By default, the amount of disk space is extended by increments of 8 MB.

- `innodb_log_file_size`

This parameter specifies the log file size.

