

# Administrative Tools

## Command-line utilities

*mysql2mysql myisamchk myisamlog myisampack mysql\_config\_editor mysql\_install\_db mysqlaccess  
mysqladmin mysqlbinlog mysqlbug mysqlcheck mysqldump mysqlhotcopy mysqlimport mysqlmanager  
mysqlpump mysqlshow mysqlshutdown mysqlslap mysqlwatch my\_print\_defaults perror replace*

## GUI Utilities

*MySQL Workbench phpMyAdmin Third-party tools*

## Command-line utilities

MySQL ships with many command-line database administration utilities:

**Table 18-1: MySQL 5&6 Command-line Utilities under \*Nix and Windows**

<i>Utility</i>	<i>*Nix</i>	<i>Windows</i>	<i>Reads Option File</i>
<i>ibd2sdi</i>	Since 8.0	Since 8.0	No
<i>innochecksum</i>	Since 5.5	Since 5.5	No
<i>mysql2mysql</i>	Discontinued 5.7	No	No
<i>myisamchk</i>	Yes	Yes	Yes
<i>myisamlog</i>	Yes	Yes	Yes
<i>myisampack</i>	Yes	Yes	Yes
<i>mysql_config_editor</i>	Since 5.6.6	Since 5.6.6	Yes
<i>mysql_install_db</i>	Discontinued 5.6	No	No
<i>mysql_upgrade</i>	Since 5.0.19/5.1.7	Yes	No
<i>mysqlaccess</i>	Discontinued 5.7	No	No
<i>mysqladmin</i>	Yes	Yes	Yes
<i>mysqlbinlog</i>	Yes	Yes	No
<i>mysqlbug</i>	Yes	Yes	No
<i>mysqlcheck</i>	Yes	Yes	Yes
<i>mysqldump</i>	Yes	Yes	Yes
<i>mysqldumpslow</i>	Since 5.5	Since 5.5	Yes
<i>mysqlhotcopy</i>	Yes	Yes	Yes
<i>mysqlimport</i>	Yes	Yes	Yes
<i>mysqlmanager</i>	Discontinued 5.4, 6.0.5	Discontinued 5.4, 6.0.5	Yes
<i>mysql proxy (Ch 3)</i>	Yes	Yes	No
<i>mysqlpump</i>	Since 5.7.8	Since 5.7.8	Yes
<i>mysqlshow</i>	Yes	Yes	Yes
<i>mysqlshutdown</i>	No	To 5.0.5	No
<i>mysqlslap</i>	5.1.4	5.1.4	Yes
<i>mysqlwatch</i>	No	To 5.0.5	No
<i>my_print_defaults</i>	Yes	Yes	Yes
<i>pack_isam</i>	Yes	Yes	Yes
<i>perror</i>	Yes	Yes	No
<i>replace</i>	Discontinued 8.0	Discontinued 8.0	No

In mid-2010 *MySQL Workbench* replaced all previous GUI admin utilities. There is also since 5.5 a downloadable *utility pack* of tools to compare, clone and export databases; report database disk usage; check *.frm* and index files; do DDL REGEX searches; search and stop query processes; clone user accounts; for replication do setup, administer and

troubleshoot, set up automatic failover and round-robin, check consistency; for the server, clone, report, monitor the audit log. And *Fabric* manages server farms.

## Command-line utility syntax and option files

Utilities that read option files generally do so under the conventions described in *Chapter 3* (Using option files). They read from the [client] section of these files, and in addition from an options file section named after themselves, so for example *mysqladmin* reads from a [mysqladmin] section, if present. Options are single words, for example *check*, or hyphenated words, for example *update-state*. If the option sets a variable, the syntax is the same as for the server:

```
varName=value
```

You can also use command-line arguments to tell these utilities which configuration file(s) to read, or to read no configuration file at all:

```
--defaults-file=fileName
```

This tells the utility to read options from *fileName*.

```
--defaults-extra-file=fileName
```

This tells the utility to read *fileName* after reading the global option file(s).

```
--no-defaults
```

This tells the utility to read no option file.

Generally, utilities that read option files also accept command-line arguments which are those same options prefixed with a double hyphen, for example *--check*, *--update-state*, or *--varname=value*. Often the program recognises one-character abbreviations of such commands, for example *-c* for *--check*, *-U* for *--update-state*, and *-O* for the deprecated syntax *--set-variable=varName=value*. These abbreviations can be concatenated, for example *-cU*. You can also instruct the programs to simply print help, arguments or version information, then exit:

```
-, --help
```

Display help and exit.

```
--print-defaults
```

Display the options that would be in force without additional arguments, then exit (not *mysqlhotcopy*).

```
-V, --version
```

Display version information and exit.

Since 5.1.21, *mysql*, *mysqladmin*, *mysqlbinlog*, *mysqlcheck*, *mysqldump*, *mysqlimport*, *mysqlshow*, *mysqslap*, *mysqltest* and *mysql\_upgrade* accept the options *--debug-check* (show debug info on exit) and *--debug-info* (also show CPU and memory statistics). Since 5.5.8, *mysql*, *mysqladmin*, *mysqlbinlog*, *mysqlcheck*, *mysqldump*, *mysqlimport* and *mysqlshow* accept *--bind-address=ipaddr* to select a network interface.

CREATE|ALTER USER ... REQUIRE SSL requires that user to connect securely. Since 5.7.7, client programs attempt a secure connection where supported, and with arguments *--ssl* or *--ssl=1* or *--enable-ssl*. It's skipped with *--ssl=0*, *--skip-ssl*, *--disable-ssl*. For SSL options see Table 3-2 in Chapter 3; for more info see [here](#).

## ibd2sdi

**ibd2sdi [options] file1 [file2 ...**

This InnoDB tablespace serialised dictionary information (SDI) extraction utility does uncommitted reads on *.ibd* and *ibdata\** files. There are the usual general options and ...

<code>--dumpfile=.</code> , <code>-d</code>	Send output to specified file instead of stdout
<code>--skip-data</code> , <code>-s</code>	Skip data, send only ids and keys
<code>--id=#</code> , <code>-i #</code>	Send SDI for the specified table or tablespace object id as specified in <code>mysql.tables</code> and <code>mysql.tablespace</code> data dictionary tables.
<code>--type=#</code> , <code>-t #</code>	Send SDI for table ( <code>type=1</code> ), tablespace ( <code>type=2</code> ) objects
<code>--strict-check=.</code> , <code>-c</code>	Do a strict <code>innodb</code> or <code>crc32</code> checksum to validate page checksums
<code>--no-check</code> , <code>-n</code>	Skip checksum validation
<code>--pretty</code> , <code>-p</code>	Send SDI data in JSON pretty format, disable with <code>-skip-pretty</code>

To read the rest of this and other chapters, *buy a copy of the book*

---

[TOC](#) [Previous](#) [Next](#)

---