- Close prepared statements (always happens with PHP)
- Close handler
- Release locks acquired with `GET_LOCK`

This ensures that persistent connections are in a clean state on return from the connection pool, before the client process uses them.

The `mysqli` extension does this cleanup by automatically calling the C-API function `mysql_change_user()`.

The automatic cleanup feature has advantages and disadvantages though. The advantage is that the programmer no longer needs to worry about adding cleanup code, as it is called automatically. However, the disadvantage is that the code could *potentially* be a little slower, as the code to perform the cleanup needs to run each time a connection is returned from the connection pool.

It is possible to switch off the automatic cleanup code, by compiling PHP with `MYSQLI_NO_CHANGE_USER_ON_PCONNECT` defined.

> **Note**
>
> The `mysqli` extension supports persistent connections when using either MySQL Native Driver or MySQL Client Library.

3.6 Predefined Constants

The constants below are defined by this extension, and will only be available when the extension has either been compiled into PHP or dynamically loaded at runtime.

`MYSQLI_READ_DEFAULT_GROUP`	Read options from the named group from `my.cnf` or the file specified with `MYSQLI_READ_DEFAULT_FILE`
`MYSQLI_READ_DEFAULT_FILE`	Read options from the named option file instead of from `my.cnf`
`MYSQLI_OPT_CONNECT_TIMEOUT`	Connect timeout in seconds
`MYSQLI_OPT_LOCAL_INFILE`	Enables command `LOAD LOCAL INFILE`
`MYSQLI_INIT_COMMAND`	Command to execute when connecting to MySQL server. Will automatically be re-executed when reconnecting.
`MYSQLI_CLIENT_SSL`	Use SSL (encrypted protocol). This option should not be set by application programs; it is set internally in the MySQL client library
`MYSQLI_CLIENT_COMPRESS`	Use compression protocol
`MYSQLI_CLIENT_INTERACTIVE`	Allow `interactive_timeout` seconds (instead of `wait_timeout` seconds) of inactivity before closing the connection. The client's session `wait_timeout` variable will be set to the value of the session `interactive_timeout` variable.
`MYSQLI_CLIENT_IGNORE_SPACE`	Allow spaces after function names. Makes all functions names reserved words.
`MYSQLI_CLIENT_NO_SCHEMA`	Don't allow the `db_name.tbl_name.col_name` syntax.

`MYSQLI_CLIENT_MULTI_QUERIES`	Allows multiple semicolon-delimited queries in a single `mysqli_query` call.
`MYSQLI_STORE_RESULT`	For using buffered resultsets
`MYSQLI_USE_RESULT`	For using unbuffered resultsets
`MYSQLI_ASSOC`	Columns are returned into the array having the fieldname as the array index.
`MYSQLI_NUM`	Columns are returned into the array having an enumerated index.
`MYSQLI_BOTH`	Columns are returned into the array having both a numerical index and the fieldname as the associative index.
`MYSQLI_NOT_NULL_FLAG`	Indicates that a field is defined as `NOT NULL`
`MYSQLI_PRI_KEY_FLAG`	Field is part of a primary index
`MYSQLI_UNIQUE_KEY_FLAG`	Field is part of a unique index.
`MYSQLI_MULTIPLE_KEY_FLAG`	Field is part of an index.
`MYSQLI_BLOB_FLAG`	Field is defined as `BLOB`
`MYSQLI_UNSIGNED_FLAG`	Field is defined as `UNSIGNED`
`MYSQLI_ZEROFILL_FLAG`	Field is defined as `ZEROFILL`
`MYSQLI_AUTO_INCREMENT_FLAG`	Field is defined as `AUTO_INCREMENT`
`MYSQLI_TIMESTAMP_FLAG`	Field is defined as `TIMESTAMP`
`MYSQLI_SET_FLAG`	Field is defined as `SET`
`MYSQLI_NUM_FLAG`	Field is defined as `NUMERIC`
`MYSQLI_PART_KEY_FLAG`	Field is part of an multi-index
`MYSQLI_GROUP_FLAG`	Field is part of `GROUP BY`
`MYSQLI_TYPE_DECIMAL`	Field is defined as `DECIMAL`
`MYSQLI_TYPE_NEWDECIMAL`	Precision math `DECIMAL` or `NUMERIC` field (MySQL 5.0.3 and up)
`MYSQLI_TYPE_BIT`	Field is defined as `BIT` (MySQL 5.0.3 and up)
`MYSQLI_TYPE_TINY`	Field is defined as `TINYINT`
`MYSQLI_TYPE_SHORT`	Field is defined as `SMALLINT`
`MYSQLI_TYPE_LONG`	Field is defined as `INT`
`MYSQLI_TYPE_FLOAT`	Field is defined as `FLOAT`
`MYSQLI_TYPE_DOUBLE`	Field is defined as `DOUBLE`
`MYSQLI_TYPE_NULL`	Field is defined as `DEFAULT NULL`
`MYSQLI_TYPE_TIMESTAMP`	Field is defined as `TIMESTAMP`
`MYSQLI_TYPE_LONGLONG`	Field is defined as `BIGINT`

MYSQLI_TYPE_INT24	Field is defined as MEDIUMINT
MYSQLI_TYPE_DATE	Field is defined as DATE
MYSQLI_TYPE_TIME	Field is defined as TIME
MYSQLI_TYPE_DATETIME	Field is defined as DATETIME
MYSQLI_TYPE_YEAR	Field is defined as YEAR
MYSQLI_TYPE_NEWDATE	Field is defined as DATE
MYSQLI_TYPE_INTERVAL	Field is defined as INTERVAL
MYSQLI_TYPE_ENUM	Field is defined as ENUM
MYSQLI_TYPE_SET	Field is defined as SET
MYSQLI_TYPE_TINY_BLOB	Field is defined as TINYBLOB
MYSQLI_TYPE_MEDIUM_BLOB	Field is defined as MEDIUMBLOB
MYSQLI_TYPE_LONG_BLOB	Field is defined as LONGBLOB
MYSQLI_TYPE_BLOB	Field is defined as BLOB
MYSQLI_TYPE_VAR_STRING	Field is defined as VARCHAR
MYSQLI_TYPE_STRING	Field is defined as CHAR or BINARY
MYSQLI_TYPE_CHAR	Field is defined as TINYINT. For CHAR, see MYSQLI_TYPE_STRING
MYSQLI_TYPE_GEOMETRY	Field is defined as GEOMETRY
MYSQLI_NEED_DATA	More data available for bind variable
MYSQLI_NO_DATA	No more data available for bind variable
MYSQLI_DATA_TRUNCATED	Data truncation occurred. Available since PHP 5.1.0 and MySQL 5.0.5.
MYSQLI_ENUM_FLAG	Field is defined as ENUM. Available since PHP 5.3.0.
MYSQLI_BINARY_FLAG	Field is defined as BINARY. Available since PHP 5.3.0.
MYSQLI_CURSOR_TYPE_FOR_UPDATE	
MYSQLI_CURSOR_TYPE_NO_CURSOR	
MYSQLI_CURSOR_TYPE_READ_ONLY	
MYSQLI_CURSOR_TYPE_SCROLLABLE	
MYSQLI_STMT_ATTR_CURSOR_TYPE	
MYSQLI_STMT_ATTR_PREFETCH_ROWS	
MYSQLI_STMT_ATTR_UPDATE_MAX_LENGTH	
MYSQLI_SET_CHARSET_NAME	
MYSQLI_REPORT_INDEX	Report if no index or bad index was used in a query.

MYSQLI_REPORT_ERROR	Report errors from mysqli function calls.
MYSQLI_REPORT_STRICT	Throw a mysqli_sql_exception for errors instead of warnings.
MYSQLI_REPORT_ALL	Set all options on (report all).
MYSQLI_REPORT_OFF	Turns reporting off.
MYSQLI_DEBUG_TRACE_ENABLED	Is set to 1 if mysqli_debug functionality is enabled.
MYSQLI_SERVER_QUERY_NO_GOOD_INDEX_USED	
MYSQLI_SERVER_QUERY_NO_INDEX_USED	
MYSQLI_REFRESH_GRANT	Refreshes the grant tables.
MYSQLI_REFRESH_LOG	Flushes the logs, like executing the FLUSH LOGS SQL statement.
MYSQLI_REFRESH_TABLES	Flushes the table cache, like executing the FLUSH TABLES SQL statement.
MYSQLI_REFRESH_HOSTS	Flushes the host cache, like executing the FLUSH HOSTS SQL statement.
MYSQLI_REFRESH_STATUS	Reset the status variables, like executing the FLUSH STATUS SQL statement.
MYSQLI_REFRESH_THREADS	Flushes the thread cache.
MYSQLI_REFRESH_SLAVE	On a slave replication server: resets the master server information, and restarts the slave. Like executing the RESET SLAVE SQL statement.
MYSQLI_REFRESH_MASTER	On a master replication server: removes the binary log files listed in the binary log index, and truncates the index file. Like executing the RESET MASTER SQL statement.
MYSQLI_TRANS_COR_AND_CHAIN	Appends "AND CHAIN" to mysqli_commit or mysqli_rollback.
MYSQLI_TRANS_COR_AND_NO_CHAIN	Appends "AND NO CHAIN" to mysqli_commit or mysqli_rollback.
MYSQLI_TRANS_COR_RELEASE	Appends "RELEASE" to mysqli_commit or mysqli_rollback.
MYSQLI_TRANS_COR_NO_RELEASE	Appends "NO RELEASE" to mysqli_commit or mysqli_rollback.

3.7 Notes

Some implementation notes:

1. Support was added for MYSQL_TYPE_GEOMETRY to the MySQLi extension in PHP 5.3.
2. Note there are different internal implementations within libmysqlclient and mysqlnd for handling columns of type MYSQL_TYPE_GEOMETRY. Generally speaking, mysqlnd will allocate significantly less memory. For example, if there is a POINT column in a result set, libmysqlclient may pre-allocate up to 4GB of RAM although less than 50 bytes are needed for holding a POINT column in memory. Memory allocation is much lower, less than 50 bytes, if using mysqlnd.

3.8 The MySQLi Extension Function Summary

Table 3.5 Summary of `mysqli` methods

mysqli Class			
OOP Interface	**Procedural Interface**	**Alias (Do not use)**	**Description**
Properties			
$mysqli::affected_rows	`mysqli_affected_rows`	N/A	Gets the number of affected rows in a previous MySQL operation
$mysqli::client_info	`mysqli_get_client_info`	N/A	Returns the MySQL client version as a string
$mysqli::client_version	`mysqli_get_client_version`	N/A	Returns MySQL client version info as an integer
$mysqli::connect_errno	`mysqli_connect_errno`	N/A	Returns the error code from last connect call
$mysqli::connect_error	`mysqli_connect_error`	N/A	Returns a string description of the last connect error
$mysqli::errno	`mysqli_errno`	N/A	Returns the error code for the most recent function call
$mysqli::error	`mysqli_error`	N/A	Returns a string description of the last error
$mysqli::field_count	`mysqli_field_count`	N/A	Returns the number of columns for the most recent query
$mysqli::host_info	`mysqli_get_host_info`	N/A	Returns a string representing the type of connection used
$mysqli::protocol_version	`mysqli_get_proto_info`	N/A	Returns the version of the MySQL protocol used
$mysqli::server_info	`mysqli_get_server_info`	N/A	Returns the version of the MySQL server
$mysqli::server_version	`mysqli_get_server_version`	N/A	Returns the version of the MySQL server as an integer
$mysqli::info	`mysqli_info`	N/A	Retrieves information about the most recently executed query
$mysqli::insert_id	`mysqli_insert_id`	N/A	Returns the auto generated id used in the last query

mysqli Class			
OOP Interface	**Procedural Interface**	**Alias (Do not use)**	**Description**
$mysqli::sqlstate	`mysqli_sqlstate`	N/A	Returns the SQLSTATE error from previous MySQL operation
$mysqli::warning_count	`mysqli_warning_count`	N/A	Returns the number of warnings from the last query for the given link
Methods			
`mysqli::autocommit`	`mysqli_autocommit`	N/A	Turns on or off auto-committing database modifications
`mysqli::change_user`	`mysqli_change_user`	N/A	Changes the user of the specified database connection
`mysqli::character_set_name`, mysqli::client_encoding	`mysqli_character_set_name`	`mysqli_client_encoding`	Returns the default character set for the database connection
`mysqli::close`	`mysqli_close`	N/A	Closes a previously opened database connection
`mysqli::commit`	`mysqli_commit`	N/A	Commits the current transaction
`mysqli::__construct`	`mysqli_connect`	N/A	Open a new connection to the MySQL server [Note: static (i.e. class) method]
`mysqli::debug`	`mysqli_debug`	N/A	Performs debugging operations
`mysqli::dump_debug_info`	`mysqli_dump_debug_info`	N/A	Dump debugging information into the log
`mysqli::get_charset`	`mysqli_get_charset`	N/A	Returns a character set object
`mysqli::get_connection_stats`	`mysqli_get_connection_stats`	N/A	Returns client connection statistics. Available only with mysqlnd.
`mysqli::get_client_info`	`mysqli_get_client_info`	N/A	Returns the MySQL client version as a string
`mysqli::get_client_stats`	`mysqli_get_client_stats`	N/A	Returns client per-process statistics. Available only with mysqlnd.
`mysqli::get_cache_stats`	`mysqli_get_cache_stats`	N/A	Returns client Zval cache statistics. Available only with mysqlnd.
`mysqli::get_server_info`	`mysqli_get_server_info`	N/A	NOT DOCUMENTED

mysqli Class			
OOP Interface	**Procedural Interface**	**Alias (Do not use)**	**Description**
mysqli::get_warnings	mysqli_get_warnings	N/A	NOT DOCUMENTED
mysqli::init	mysqli_init	N/A	Initializes MySQLi and returns a resource for use with mysqli_real_connect. [Not called on an object, as it returns a $mysqli object.]
mysqli::kill	mysqli_kill	N/A	Asks the server to kill a MySQL thread
mysqli::more_results	mysqli_more_results	N/A	Check if there are any more query results from a multi query
mysqli::multi_query	mysqli_multi_query	N/A	Performs a query on the database
mysqli::next_result	mysqli_next_result	N/A	Prepare next result from multi_query
mysqli::options	mysqli_options	mysqli_set_opt	Set options
mysqli::ping	mysqli_ping	N/A	Pings a server connection, or tries to reconnect if the connection has gone down
mysqli::prepare	mysqli_prepare	N/A	Prepare an SQL statement for execution
mysqli::query	mysqli_query	N/A	Performs a query on the database
mysqli::real_connect	mysqli_real_connect	N/A	Opens a connection to a mysql server
mysqli::real_escape_string mysqli::escape_string	mysqli_real_escape_string	mysqli_escape_string	Escapes special characters in a string for use in an SQL statement, taking into account the current charset of the connection
mysqli::real_query	mysqli_real_query	N/A	Execute an SQL query
mysqli::refresh	mysqli_refresh	N/A	Flushes tables or caches, or resets the replication server information
mysqli::rollback	mysqli_rollback	N/A	Rolls back current transaction
mysqli::select_db	mysqli_select_db	N/A	Selects the default database for database queries

mysqli Class			
OOP Interface	**Procedural Interface**	**Alias (Do not use)**	**Description**
`mysqli::set_charset`	`mysqli_set_charset`	N/A	Sets the default client character set
`mysqli::set_local_infile_default`	`mysqli_set_local_infile_default`	N/A	Unsets user defined handler for load local infile command
`mysqli::set_local_infile_handler`	`mysqli_set_local_infile_handler`	N/A	Set callback function for LOAD DATA LOCAL INFILE command
`mysqli::ssl_set`	`mysqli_ssl_set`	N/A	Used for establishing secure connections using SSL
`mysqli::stat`	`mysqli_stat`	N/A	Gets the current system status
`mysqli::stmt_init`	`mysqli_stmt_init`	N/A	Initializes a statement and returns an object for use with mysqli_stmt_prepare
`mysqli::store_result`	`mysqli_store_result`	N/A	Transfers a result set from the last query
`mysqli::thread_id`	`mysqli_thread_id`	N/A	Returns the thread ID for the current connection
`mysqli::thread_safe`	`mysqli_thread_safe`	N/A	Returns whether thread safety is given or not
`mysqli::use_result`	`mysqli_use_result`	N/A	Initiate a result set retrieval

Table 3.6 Summary of `mysqli_stmt` methods

MySQL_STMT			
OOP Interface	**Procedural Interface**	**Alias (Do not use)**	**Description**
Properties			
$mysqli_stmt::affected_rows	`mysqli_stmt_affected_rows`	N/A	Returns the total number of rows changed, deleted, or inserted by the last executed statement
$mysqli_stmt::errno	`mysqli_stmt_errno`	N/A	Returns the error code for the most recent statement call
$mysqli_stmt::error	`mysqli_stmt_error`	N/A	Returns a string description for last statement error
$mysqli_stmt::field_count	`mysqli_stmt_field_count`	N/A	Returns the number of field in the given statement - not documented

MySQL_STMT			
OOP Interface	**Procedural Interface**	**Alias (Do not use)**	**Description**
$mysqli_stmt::insert_id	`mysqli_stmt_insert_id`	N/A	Get the ID generated from the previous INSERT operation
$mysqli_stmt::num_rows	`mysqli_stmt_num_rows`	N/A	Return the number of rows in statements result set
$mysqli_stmt::param_count	`mysqli_stmt_param_count`	`mysqli_param_count`	Returns the number of parameter for the given statement
$mysqli_stmt::sqlstate	`mysqli_stmt_sqlstate`	N/A	Returns SQLSTATE error from previous statement operation
Methods			
`mysqli_stmt::attr_get`	`mysqli_stmt_attr_get`	N/A	Used to get the current value of a statement attribute
`mysqli_stmt::attr_set`	`mysqli_stmt_attr_set`	N/A	Used to modify the behavior of a prepared statement
`mysqli_stmt::bind_param`	`mysqli_stmt_bind_param`	`mysqli_bind_param`	Binds variables to a prepared statement as parameters
`mysqli_stmt::bind_result`	`mysqli_stmt_bind_result`	`mysqli_bind_result`	Binds variables to a prepared statement for result storage
`mysqli_stmt::close`	`mysqli_stmt_close`	N/A	Closes a prepared statement
`mysqli_stmt::data_seek`	`mysqli_stmt_data_seek`	N/A	Seeks to an arbitrary row in statement result set
`mysqli_stmt::execute`	`mysqli_stmt_execute`	`mysqli_execute`	Executes a prepared Query
`mysqli_stmt::fetch`	`mysqli_stmt_fetch`	`mysqli_fetch`	Fetch results from a prepared statement into the bound variables
`mysqli_stmt::free_result`	`mysqli_stmt_free_result`	N/A	Frees stored result memory for the given statement handle
`mysqli_stmt::get_result`	`mysqli_stmt_get_result`	N/A	Gets a result set from a prepared statement. Available only with mysqlnd.
`mysqli_stmt::get_warnings`	`mysqli_stmt_get_warnings`	N/A	NOT DOCUMENTED

MySQL_STMT			
OOP Interface	**Procedural Interface**	**Alias (Do not use)**	**Description**
$mysqli_stmt::more_results()	mysqli_stmt_more_results()	N/A	NOT DOCUMENTED Available only with mysqlnd.
$mysqli_stmt::next_result()	mysqli_stmt_next_result()	N/A	NOT DOCUMENTED Available only with mysqlnd.
mysqli_stmt::num_rows	mysqli_stmt_num_rows	N/A	See also property $mysqli_stmt::num_rows
mysqli_stmt::prepare	mysqli_stmt_prepare	N/A	Prepare an SQL statement for execution
mysqli_stmt::reset	mysqli_stmt_reset	N/A	Resets a prepared statement
mysqli_stmt::result_metadata	mysqli_stmt_result_metadata	mysqli_get_metadata	Returns result set metadata from a prepared statement
mysqli_stmt::send_long_data	mysqli_stmt_send_long_data	mysqli_send_long_data	Send data in blocks
mysqli_stmt::store_result	mysqli_stmt_store_result	N/A	Transfers a result set from a prepared statement

Table 3.7 Summary of `mysqli_result` methods

mysqli_result			
OOP Interface	**Procedural Interface**	**Alias (Do not use)**	**Description**
Properties			
$mysqli_result::current_field	mysqli_field_tell	N/A	Get current field offset of a result pointer
$mysqli_result::field_count	mysqli_num_fields	N/A	Get the number of fields in a result
$mysqli_result::lengths	mysqli_fetch_lengths	N/A	Returns the lengths of the columns of the current row in the result set
$mysqli_result::num_rows	mysqli_num_rows	N/A	Gets the number of rows in a result
Methods			
mysqli_result::data_seek	mysqli_data_seek	N/A	Adjusts the result pointer to an arbitrary row in the result
mysqli_result::fetch_all	mysqli_fetch_all	N/A	Fetches all result rows and returns the result set as an associative array, a numeric array, or both. Available only with mysqlnd.

mysqli_result			
OOP Interface	**Procedural Interface**	**Alias (Do not use)**	**Description**
mysqli_result::fetch_array	mysqli_fetch_array	N/A	Fetch a result row as an associative, a numeric array, or both
mysqli_result::fetch_assoc	mysqli_fetch_assoc	N/A	Fetch a result row as an associative array
mysqli_result::fetch_field_direct	mysqli_fetch_field_direct	N/A	Fetch meta-data for a single field
mysqli_result::fetch_field	mysqli_fetch_field	N/A	Returns the next field in the result set
mysqli_result::fetch_fields	mysqli_fetch_fields	N/A	Returns an array of objects representing the fields in a result set
mysqli_result::fetch_object	mysqli_fetch_object	N/A	Returns the current row of a result set as an object
mysqli_result::fetch_row	mysqli_fetch_row	N/A	Get a result row as an enumerated array
mysqli_result::field_seek	mysqli_field_seek	N/A	Set result pointer to a specified field offset
mysqli_result::free, mysqli_result::close, mysqli_result::free_result	mysqli_free_result	N/A	Frees the memory associated with a result

Table 3.8 Summary of `mysqli_driver` methods

MySQL_Driver			
OOP Interface	**Procedural Interface**	**Alias (Do not use)**	**Description**
Properties			
N/A			
Methods			
mysqli_driver::embedded_server_end	mysqli_embedded_server_end	N/A	NOT DOCUMENTED
mysqli_driver::embedded_server_start	mysqli_embedded_server_start	N/A	NOT DOCUMENTED

> **Note**
>
> Alias functions are provided for backward compatibility purposes only. Do not use them in new projects.

3.9 The mysqli class

Represents a connection between PHP and a MySQL database.

```
mysqli {
```