- For multiple-table deletes and updates, an error is produced with safe updates enabled only if any target table uses a table scan.

Disabling mysql Auto-Reconnect

If the `mysql` client loses its connection to the server while sending a statement, it immediately and automatically tries to reconnect once to the server and send the statement again. However, even if `mysql` succeeds in reconnecting, your first connection has ended and all your previous session objects and settings are lost: temporary tables, the autocommit mode, and user-defined and session variables. Also, any current transaction rolls back. This behavior may be dangerous for you, as in the following example where the server was shut down and restarted between the first and second statements without you knowing it:

```
mysql> SET @a=1;
Query OK, 0 rows affected (0.05 sec)

mysql> INSERT INTO t VALUES(@a);
ERROR 2006: MySQL server has gone away
No connection. Trying to reconnect...
Connection id:    1
Current database: test

Query OK, 1 row affected (1.30 sec)

mysql> SELECT * FROM t;
+------+
| a    |
+------+
| NULL |
+------+
1 row in set (0.05 sec)
```

The `@a` user variable has been lost with the connection, and after the reconnection it is undefined. If it is important to have `mysql` terminate with an error if the connection has been lost, you can start the `mysql` client with the `--skip-reconnect` option.

For more information about auto-reconnect and its effect on state information when a reconnection occurs, see Automatic Reconnection Control.

mysql Client Parser Versus Server Parser

The `mysql` client uses a parser on the client side that is not a duplicate of the complete parser used by the `mysqld` server on the server side. This can lead to differences in treatment of certain constructs. Examples:

- The server parser treats strings delimited by `"` characters as identifiers rather than as plain strings if the `ANSI_QUOTES` SQL mode is enabled.

  The `mysql` client parser does not take the `ANSI_QUOTES` SQL mode into account. It treats strings delimited by `"`, `'`, and `` ` `` characters the same, regardless of whether `ANSI_QUOTES` is enabled.

- Within `/*! ... */` and `/*+ ... */` comments, the `mysql` client parser interprets short-form `mysql` commands. The server parser does not interpret them because these commands have no meaning on the server side.

  If it is desirable for `mysql` not to interpret short-form commands within comments, a partial workaround is to use the `--binary-mode` option, which causes all `mysql` commands to be disabled except `\C` and `\d` in noninteractive mode (for input piped to `mysql` or loaded using the `source` command).

4.5.2 mysqladmin — A MySQL Server Administration Program