/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

ACID MYSQL SECURITY

A Guide to ACID In MySQL

AUTHOR: LUKAS VILEIKIS LENGTH: 6 MINS TYPE: GUIDE PUBLISHED: 2023-03-17

/* INTRO */

ACID is a set of four crucial database properties that ensure data integrity and consistency even in the event of a catastrophic server failure. Learn all about how the concept works in MySQL here!

/* TOOLS USED IN THE TUTORIAL */

TOOL	DESCRIPTION
	TOP RATED DATABASE MANAGEMENT TOOL AND SQL CLIENT DOWNLOAD →
	THE MYSQL DATABASE VERSION 8 OR LATER

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

What Is ACID?

ACID has been a thing ever since databases have entered the spotlight. ACID stands for four properties crucial to properly manage databases and these are as follows:

- A stands for Atomicity.
- C stands for Consistency.
- I stands for Isolation.
- D stands for Durability.

In each database management system ACID works differently, but nonetheless, its main concepts remain unchanged.

ACID in MySQL

In MySQL, ACID works in the following way:

- Atomicity is ensured by making sure that all statements in a transaction operate as a single unit and their results are seen as a whole or not seen at all.

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

log file that tracks all of the changes to its infrastructure.

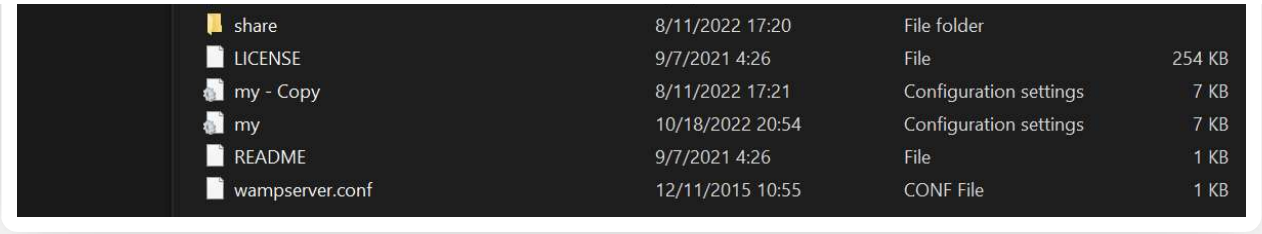
ACID principles are a known cornerstone of database functionality, yet they‘re not set in stone at all – some of them can be exchanged for speed if we modify certain parameters within my.cnf (we‘ll get into that a little later.)

It‘s also worth noting that the only storage engine that support ACID principles in MySQL is InnoDB and its counterpart – XtraDB developed by Percona. Both storage engines support the ACID principles by default.

ACID in MySQL – Configuration

All settings in MySQL can be set and / or modified by modifying the contents of the main file relevant to the functionality of MySQL – my.cnf (the file will be called my.ini if we‘re using a Windows infrastructure.) The file can be found in various locations depending on your operating system – the most likely location of the file in Linux infrastructure will be the `/var/lib/mysql/` folder, however, you can also find the file hiding in the `/etc/mysql/` folder. For Windows users, everything is simpler: `my.ini` will most likely be found in the `/bin/mysql/mysql*.*.** folder` – *.*.** refers to your MySQL version (and yes, you can have a copy of the file in the same directory as well – only the original my.ini file will be scanned through.)

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW



share	8/11/2022 17:20	File folder	
LICENSE	9/7/2021 4:26	File	254 KB
my - Copy	8/11/2022 17:21	Configuration settings	7 KB
my	10/18/2022 20:54	Configuration settings	7 KB
README	9/7/2021 4:26	File	1 KB
wampserver.conf	12/11/2015 10:55	CONF File	1 KB

↑ Files in the MySQL Directory.

Once you open the file up, you will see a lot of settings related to the inner workings of MySQL – Windows users got it lucky because the file also comes with a lot of comments within itself to help users understand how everything works, while those of you using Linux will have to read through the docs (or this blog) to understand how everything works.

Here‘s how some of the parameters within the file will look like for those using Windows:

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW


```
49  lc-messages-dir="d:/wamp64/bin/mysql/mysql5.7.36/share"
50  lc-messages=en_US
51
52  ; Default storage engine that will be used if the engine is not specified when creating tables.
53  ; This does not prevent the creation of InnoDB tables, just specify ENGINE=InnoDB
54  default-storage-engine=InnoDB
55  ;default-storage-engine=InnoDB
56
57  ;To avoid warning messages
58  secure_file_priv="d:/wamp64/tmp"
59  skip-ssl
```

↑ The parameters relevant to MySQL

As you can see, there are rather a lot of settings and in Windows MySQL really does its best to walk us through them and even provides links to the documentation – `my.cnf` isn't so generous, so the only settings you will see when using the Windows counterpart of my.cnf are these:

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW


```
170 innodb-file-per-table=1
171 innodb-flush-log-at-trx-commit=1
172 innodb-flush-method=normal
173 ;innodb-force-recovery=1
174 innodb-ft-enable-stopword=off
175 innodb-ft-max-token-size=10
176 innodb-ft-min-token-size=0
177 innodb-io-capacity=2000
178 innodb-max-dirty-pages-pct=90
179 innodb-lock-wait-timeout=600
180 innodb-log-buffer-size=16M
181 innodb-log-file-size=512M
182 innodb-log-files-in-group=2
183 innodb-max-dirty-pages-pct=80
184 innodb-optimize-fulltext-only=1
185 innodb-page-size=16K
186 innodb-purge-threads=10
187 innodb-read-io-threads=10
188 innodb-stats-on-metadata=0
189 ;innodb-strict-mode=off
190 innodb-thread-concurrency=16
191 innodb-undo-log-truncate=on
192 innodb-write-io-threads=4
```

↑ The configuration of MySQL - InnoDB Parameters.

Well, not exactly all of them – sysadmins would go crazy – however, you will certainly see some of the following settings:

- innodb-flush-method

  innodb-file-per-table

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

`innodb-flush-log-at-trx-commit` parameter – deal with ACID within MySQL and its friends like Percona Server and MariaDB.

innodb-flush-log-at-trx-commit

ACID within MySQL is primarily controlled by this parameter. The purpose of this parameter is to control the balance between ACID and speed within MySQL – since InnoDB (the primary storage engine in MySQL) by default has this parameter set to 1, ACID is always being maintained. However, this parameter can also be set to different values, such as 0 or 2. The bottom line is this – if you aren't necessarily dealing with millions of rows and don't necessarily need more speed to complete insertion operations (`INSERT` or `LOAD DATA INFILE` queries), leave the value at default. If you do, however, keep the following in mind:

A value of 0 or 2 will make your queries faster at the expense of letting MySQL lose up to one second's worth of transactions in the event of query failure (electricity going out, etc.)

- The value of 0 tells MySQL that when data is written to the log file and a flushing (saving) operation is performed, MySQL should do nothing once the transaction commits thus saving us time in the process.

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

cluster.

For most, leaving the setting at its default value – 1 – will be enough, however, for those who are into data analysis or perhaps just experimenting with the capabilities of the RDBMS, changing this setting can be beneficial. Choose wisely!

If you are looking for an easy and powerful SQL client and database manager, then you've got to try DbVisualizer. It connects to nearly any database.

ACID vs. BASE

Some might say that ACID is a competitor to BASE – the ACID equivalent in non-relational database management systems. And that‘s not exactly false – BASE stands for Basically Available and Eventually Consistent data. For most non-relational (NoSQL) databases, the ACID model is overkill and that‘s why most of them go into a different direction. The BASE model is a softer model offering the flexibility for data, but at the same time sacrificing integrity.

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

should enter a consistent state. This doesn't always happen, hence the problem itself – but that offers NoSQL databases a lot more flexibility than their ACID counterparts.

NoSQL databases are also known for their Soft State capabilities – Soft State essentially says „data integrity is the problem of the developer, not the database, do what you want." To sum up, BASE isn't exactly the best friend of ACID.

Problems Related to ACID and BASE

Both ACID and BASE solve distinct problems – ACID ensures data integrity in the event of a server failure, while BASE offers more flexibility for those working with data. However, both of those approaches come with their distinct problems – BASE doesn't ensure data integrity, while ACID is only available in some aspects of relational database management systems (SQL Server, PostgreSQL and TimescaleDB, also in MySQL and all of its flavors like Percona Server and MariaDB.)

The primary requirement for those working with ACID in MySQL is InnoDB or its counterpart Percona XtraDB – no other MySQL storage engines support such a model: MyISAM is obsolete (the only real reason to use it is if we want faster `COUNT(*)` query performance because MyISAM stores the number inside of its metadata and other storage

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

If you're curious how to solve other problems related to your relational or non-relational database infrastructure, grab a free trial and ask DbVisualizer (seriously – the tool is used by NASA, AMD, Netflix, Twitter, Tesla, Uber, and a whole lot of other companies), come back to our blog after a while to learn more news in the database space, and until next time!

/* ABOUT THE AUTHOR */

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

multiple places including his blog over at lukasvileikis.com.

/* SIGN UP TO RECEIVE THE TABLE'S ROUNDUP */

ENTER EMAIL

BY SUBMITTING THIS FORM, I AGREE TO THE DBVIS PRIVACY POLICY

SUBMIT →

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

/* MORE FROM THE TABLE */

TITLE
SQL STRING FUNCTIONS: EVERYTHING YOU NEED TO KNOW

AUTHOR
ANTONELLO ZANINI

TAGS
MYSQL ORACLE POSTGRESQL SQL SQL SERVER

TITLE
MYSQL LOCATE FUNCTION: FIND SUBSTRING POSITION

AUTHOR
ANTONELLO ZANINI

TAGS
MYSQL

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

TITLE
BEST DATABASE TOOLS FOR ADMINISTRATORS: ULTIMATE LIST

AUTHOR
THETABLE

TAGS
DBA PERFORMANCE SECURITY

TITLE
THE HEAP DATA STRUCTURE AND IN-MEMORY DATA EXPLAINED

AUTHOR
LUKAS VILEIKIS

TAGS
MYSQL SQL

TITLE
SQL BOOLEAN TYPE: HOW TO USE IT IN ALL MAJOR RELATIONAL DATABASES

AUTHOR
ANTONELLO ZANINI

TAGS
MYSQL ORACLE POSTGRESQL SQL SQL SERVER

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

TITLE
UNDERSTANDING MYSQL DATA TYPES: A COMPLETE GUIDE FOR BEGINNERS

AUTHOR
LUKAS VILEIKIS

TAGS
DATA TYPES MARIADB MYSQL

TITLE
SQL IS NOT NULL CONDITION: DEFINITIVE GUIDE

AUTHOR
ANTONELLO ZANINI

TAGS
MYSQL ORACLE POSTGRESQL SQL SQL SERVER

TITLE
CAN A PRIMARY KEY BE NULL IN AN SQL TABLE?

AUTHOR
THETABLE

TAGS
MYSQL ORACLE POSTGRESQL SQL SQL SERVER

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW

ENGINEERED IN STOCKHOLM, SWEDEN.
Copyright 2025.

/* Notice */
DbVisualizer 25.3 has just been released
SEE WHAT'S NEW