

GLOSSAR

Was ist MySQL? Eine anfangergefreundliche Erklärung

August 26, 2019



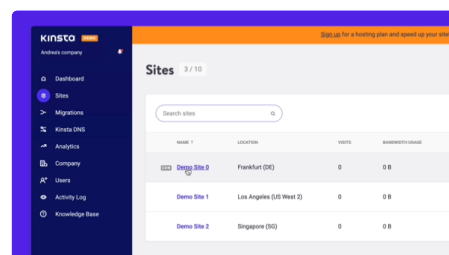
MySQL ist ein relationales Open-Source-SQL-Datenbankverwaltungssystem, das von Oracle entwickelt und unterstützt wird.

Das ist die kurze, ein Satz lange Antwort auf die Frage „Was ist MySQL?“, aber lass uns das in Begriffe aufteilen, die etwas verständlicher sind.

Eine Database ist lediglich eine strukturierte Sammlung von Daten, die zur einfachen Verwendung und zum einfachen Abrufen organisiert sind. Für eine WordPress-Seite sind diese „Daten“ Dinge wie der Text deines Blog-Posts, Informationen für alle registrierten User auf deiner Seite, [automatisch geladene Daten](#), wichtige Einstellungskonfigurationen usw.

MySQL ist nur ein beliebtes System, das diese Daten für dich speichern und verwalten kann, und es ist eine besonders beliebte Datenbaselösung für WordPress-Seiten.

Kommen wir jetzt etwas intensiver in die Frage.



Testen Sie die kostenlose Demo

MySQL wurde ursprünglich bereits 1995 eingeführt. Seitdem hat es einige Besitzer- / Verantwortungsänderungen durchlaufen, bevor es 2010 zur Oracle Corporation kam. Während Oracle jetzt die Verantwortung trägt, ist MySQL immer noch **Open-Source-Software**, was bedeutet, dass du es frei verwenden und ändern kannst.



— MySQL-Logo (Bildquelle: [MySQL/Oracle](#))

Der Name stammt aus der Kombination von „My“ – dem Namen der Tochter des Mitbegründers – mit SQL – der Abkürzung für Structured Query Language, einer Programmiersprache, mit der du auf Daten in einer relationalen Database zugreifen und diese verwalten kannst.

Um zu verstehen, wie MySQL funktioniert, ist es wichtig, zwei miteinander verbundene Konzepte zu kennen:

- [Relationale Database](#)
- [Client-Server-Modell](#)

Relationale Databases

Beim Speichern von Daten in einer Database stehen dir verschiedene Ansätze zur Verfügung.

MySQL wählt einen Ansatz, der als **relationale Database** bezeichnet wird.

Bei einer relationalen Database werden deine Daten in mehrere separate Speicherbereiche, sogenannte **Tabellen**, aufgeteilt, anstatt alles in einer großen Speichereinheit zusammenzufassen.

Angenommen, du möchtest zwei Arten von Informationen speichern:

- **Kunden** – ihre Namen, Adresse, Details usw.
- **Bestellungen** – wie welche Produkte gekauft wurden, der Preis, wer die Bestellung gemacht hat, etc.

Wenn du versuchst, all diese Daten in einem großen Topf zusammenzufassen, treten folgende Probleme auf:

- **Unterschiedliche Daten** – Die Daten, die du für eine Bestellung sammeln musst,

- **Doppelte Daten** – jeder Kunde hat einen Namen und jede Bestellung hat auch den Namen eines Kunden. Der Umgang mit diesen doppelten Daten wird chaotisch.
- **Keine Organisation** – wie verbindest du Bestellinformationen zuverlässig mit Kundeninformationen?

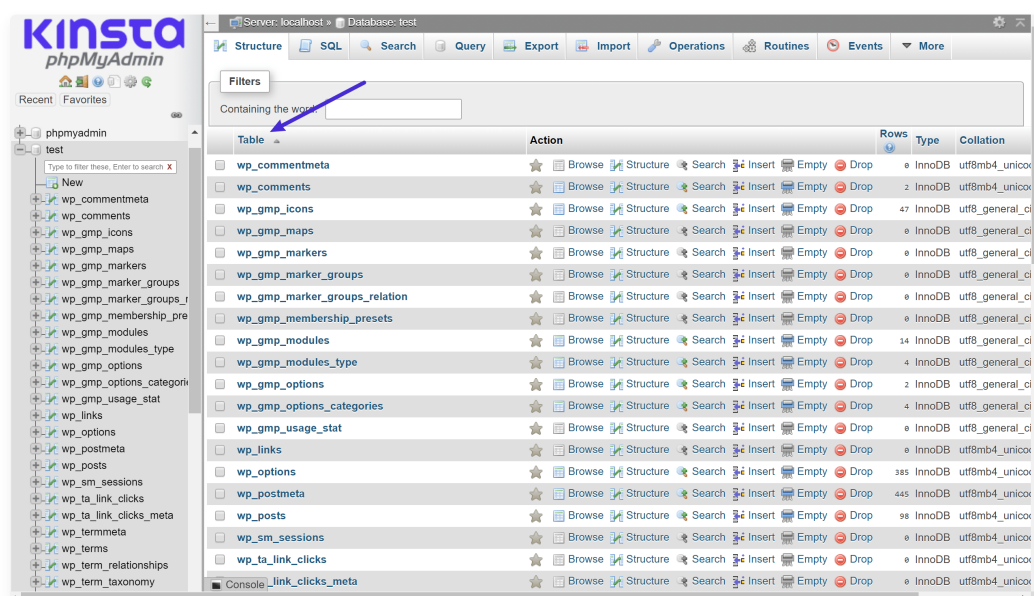
Um diese Probleme zu lösen, würde eine relationale Database eine separate Tabelle für Kunden und eine andere separate Tabelle für Bestellungen verwenden.

Du möchtest aber wahrscheinlich auch sagen können: zeige mir alle Bestellungen für John Doe. Hier kommt der **relationale** Teil ins Spiel.

Mithilfe eines so genannten „**Schlüssels**“ kannst du die Daten aus diesen beiden Tabellen miteinander verknüpfen, sodass du die Daten in verschiedenen Tabellen nach Bedarf bearbeiten und kombinieren kannst. Es ist wichtig zu beachten, dass ein Schlüssel nicht der Name des Kunden ist. Stattdessen verwendest du etwas, das zu 100% eindeutig ist, beispielsweise eine numerische ID-Nummer.

Wenn du dir jemals die Database deiner WordPress-Seite angesehen hast, wirst du feststellen, dass dieses relationale Modell verwendet wird, bei dem alle deine Daten in separate Tabellen unterteilt sind.

Standardmäßig verwendet WordPress 12 separate Tabellen, aber viele WordPress-Plugins fügen auch ihre eigenen Tabellen hinzu. Zum Beispiel hat die Database für die WordPress-Seite unten 44 separate Tabellen!



– Ein Beispiel für verschiedene Tabellen in MySQL

Um dieses relationale Konzept zu vervollständigen, sollten wir es spezifisch für WordPress machen.

WordPress speichert Blog-Posts in der Tabelle **wp_posts** und User in der Tabelle **wp_users**. Da diese beiden Tabellen jedoch durch einen **Schlüssel** verbunden sind, kannst du jedes Userkonto mit allen Blogbeiträgen verknüpfen, die jeder User verfasst hat.

So sieht das in der Database aus.

Jedem Post wird ein **post_author** zugewiesen, bei dem es sich um eine eindeutige

M chten Sie wissen, wie wir unseren Traffic um  ber 1000% gesteigert haben?

Geh re zu den 20 000+ Personen, die w chentlich Newsletter mit Insider-Tipps rund um WordPress erhalten!

Jetzt abonnieren

ID	post_author	post_date	post_date_gmt	post_content	post_title	post_excerpt
2	1	2015-03-19 17:58:30	2015-03-19 17:58:30	<h2>Sports coach at home</h2><h2>Profess...	Home	
4	1	2018-01-31 06:50:16	0000-00-00 00:00:00	Make This Nofollo...	Nofollow Example	
6	1	2018-02-14 03:58:41	2018-02-14 03:58:41		IMG_1656	
7	1	2018-02-14 03:58:46	2018-02-14 03:58:46		IMG_1657	
9	1	2018-02-22 02:24:25	2018-02-22 02:24:25		kinsta-wordpress-hosting	
10	1	2018-02-22 02:58:01	0000-00-00 00:00:00	Yoast SEO Metabox Example	Yoast SEO Metabox Example	
11	1	2018-02-22 02:44:36	2018-02-22 02:44:36		Yoast SEO Metabox Example	
13	1	2018-03-08 04:13:21	0000-00-00 00:00:00	It's so easy to do things with a content managemen...	Look How Easy This Is	
123	1	2016-10-19 17:08:36	2016-10-19 17:08:36	Lorem ipsum dolor sit amet, consectetur adipiscing...	Torquent per conubia nostra	
127	1	2016-10-19 17:23:49	2016-10-19 17:23:49	Lorem ipsum dolor sit amet, consectetur adipiscing...	Neque adipiscing an	

– Die wp_posts Tabelle

Wenn du dann sehen m chtest, welches Userkonto dieser Nummer entspricht, kannst du dir die **ID** in der Tabelle **wp_users** ansehen:

ID	user_login	user_pass	user_nicename	user_email	user_url	user_registered
1	testadmin	\$P\$BsZhalvycfolai49unZX/1.T3lvazb1	testadmin	@gmail.com		2017-11-09 07:53:2

– Die wp_users Tabelle

Der Schl ssel – die ID-Nummer – verbindet alles miteinander. Und so sind sie „miteinander verbunden“, obwohl die Daten in separaten Tabellen gespeichert sind.

Client-Server Modell

MySQL ist nicht nur ein **relationales** Databasesystem, sondern verwendet auch das sogenannte **Client-Server-Modell**.

Im **Server**teil befinden sich deine Daten tatsächlich. Um auf diese Daten zugreifen zu können, musst du sie jedoch anfordern. Hier kommt der **Client** ins Spiel.

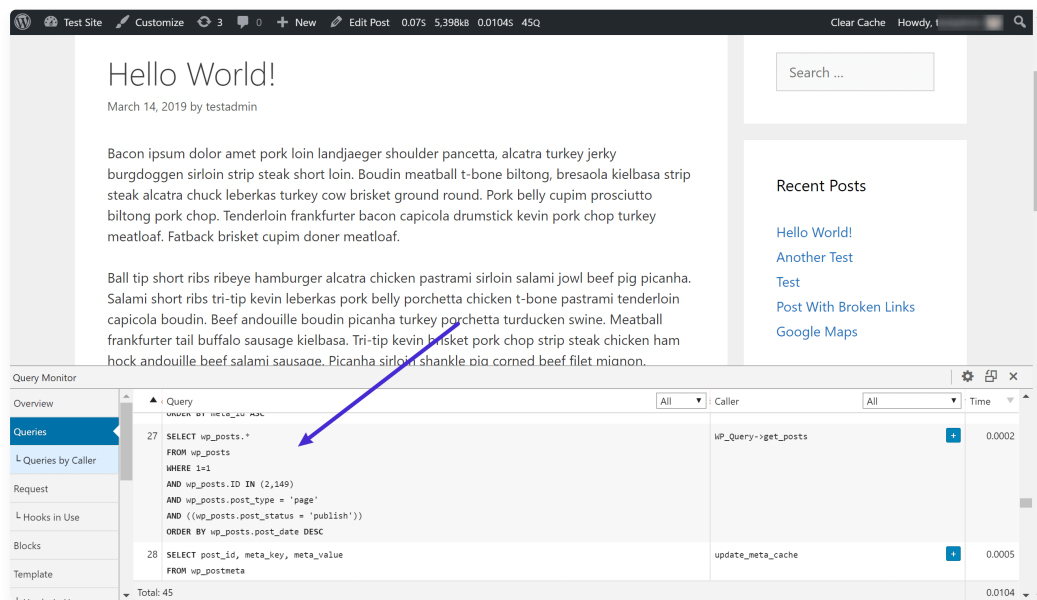
Unter Verwendung von SQL – der zuvor erwähnten Programmiersprache – sendet der **Client** eine Anfrage nach den vom Client benötigten Daten an den Databaseserver.

Wenn beispielsweise jemand einen Blogeintrag auf deiner Website besucht, sendet deine WordPress-Website mehrere SQL-Anforderungen an den Databaseserver, um alle Informationen abzurufen, die für die Übermittlung des Blogeintrags an den Webbrowser des jeweiligen Besuchers erforderlich sind. Es würde:

- die Tabelle **wp_posts** abfragen, um den Inhalt für den Blog-Beitrag abzurufen
- die Tabelle **wp_users** abfragen, um Informationen zur Autorenbox zu erhalten (mit dem oben gezeigten Schlüssel).
- Etc.

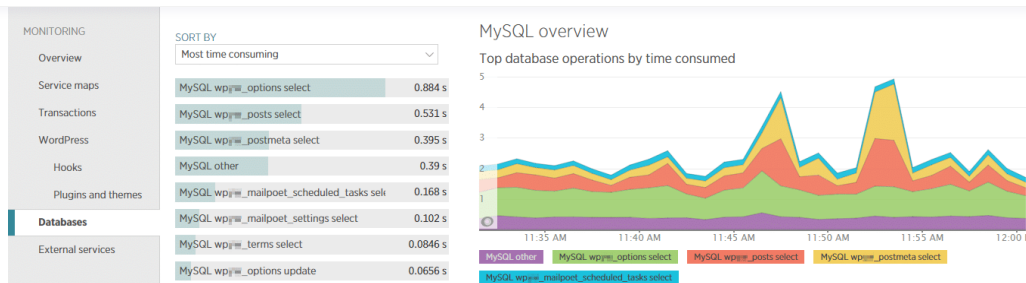
Wenn du genau sehen möchtest, welche Arten von Databaseanfragen von deiner WordPress-Seite gestellt werden, kannst du ein wunderbares kostenloses Plugin namens [Query Monitor](#) verwenden, um das genaue Zusammenspiel zwischen deiner WordPress-Seite (dem Client) und dem Databaseserver zu sehen

Haben Sie mit Ausfallzeiten und WordPress-Problemen zu kämpfen? Kinsta ist die Hosting-Lösung, die Ihnen Zeit spart! [Sieh dir unsere Features an](#)



– Das Query Monitor-Plugin zeigt dir einzelne SQL-Abfragen an, die an den MySQL-Server gesendet wurden

Eine weitere Premium-Lösung, die du verwenden kannst, ist [New Relic](#). Unsere Ingenieure verwenden dieses Tool häufig, um Leistungsprobleme auf einer WordPress-Seite zu beheben, die möglicherweise auf eine langsame MySQL-Abfrage zurückzuführen sind.



Wenn deine WordPress-Seite nicht auf den Databaseserver zugreifen kann, wird die Meldung **error establishing a database connection** ausgelöst.

Kinsta verwendet MariaDB, nicht MySQL: Was ist der Unterschied?

Hier bei Kinsta verwenden wir ein Databasesystem namens **MariaDB**, nicht MySQL. Alle Eigenschaften, die du mit MySQL gesehen hast, gelten jedoch auch für MariaDB.

Tatsächlich ist MariaDB eine Fork von MySQL, und MariaDBs führender Entwickler ist einer der Gründer von MySQL. Ein „Fork“ bedeutet lediglich, dass die MariaDB-Entwickler den ursprünglichen Open-Source-MySQL-Code als Basis genommen und darauf aufbauend MariaDB erstellt haben.

MariaDB hat zwar einen anderen Namen, ist jedoch eng mit MySQL verknüpft und bietet die Möglichkeit, vollständig durch „Drop-In“ zu ersetzen (dh du kannst nahtlos von MySQL zu MariaDB wechseln, ohne besondere Vorkehrungen treffen zu müssen).

MariaDB bietet zwar Interoperabilität mit MySQL, in einigen Bereichen jedoch auch eine **verbesserte Leistung**, was unserer Philosophie entspricht, die leistungstärkste Architektur für die Stromversorgung deiner WordPress-Seite zu verwenden.

“ Kinsta verwendet MariaDB, eine Fork von MySQL, um die beste Leistung für deine WordPress-Seite zu gewährleisten. 🚀

CLICK TO TWEET

Rekapituliere: Was ist MySQL?

MySQL ist ein relationales Open-Source-Databaseverwaltungssystem. Für WordPress-Seiten bedeutet dies, dass du alle deine Blog-Posts, User, [Plugin-Informationen](#) usw. speichern kannst.

Diese Informationen werden in separaten „Tabellen“ gespeichert und mit „Schlüsseln“ verknüpft, weshalb sie **relational** sind.

Wenn deine WordPress-Seite auf diese Informationen zugreifen muss, sendet sie mithilfe von SQL eine Anfrage an den MySQL-Databaseserver (dies ist das **Client-Server-Modell**).

Kinsta verwendet MariaDB, nicht MySQL: MariaDB ist jedoch ein Fork von MySQL von einem der Mitbegründer von MySQL und bietet Drop-In-Ersatz-Interoperabilität sowie einige Leistungsverbesserungen. Daher gelten alle Kernkonzepte in diesem Artikel auch für MariaDB.

Wenn du bei Kinsta hostest, bieten wir sowohl [direkten Databasezugriff](#) als auch die Möglichkeit, Databaseverwaltungstools wie [phpMyAdmin](#) zu verwenden.

Spare Zeit, Kosten und maximiere die Leistung deiner Seite mit:

- Schnelle Hilfe von WordPress Hosting Experten, 24/7.
- Cloudflare Enterprise Integration.
- Globale Reichweite mit 28 Rechenzentren weltweit.
- Optimierung mit unserem integrierten Application Performance Monitoring.

All das und noch viel mehr, in einem Plan ohne langfristige Verträge, betreute Migrationen und eine 30-Tage-Geld-zurück-Garantie. [Schau Dir hier unsere Pakete an](#) oder [sprich mit dem Vertrieb](#), um den richtigen Tarif für dich zu finden.