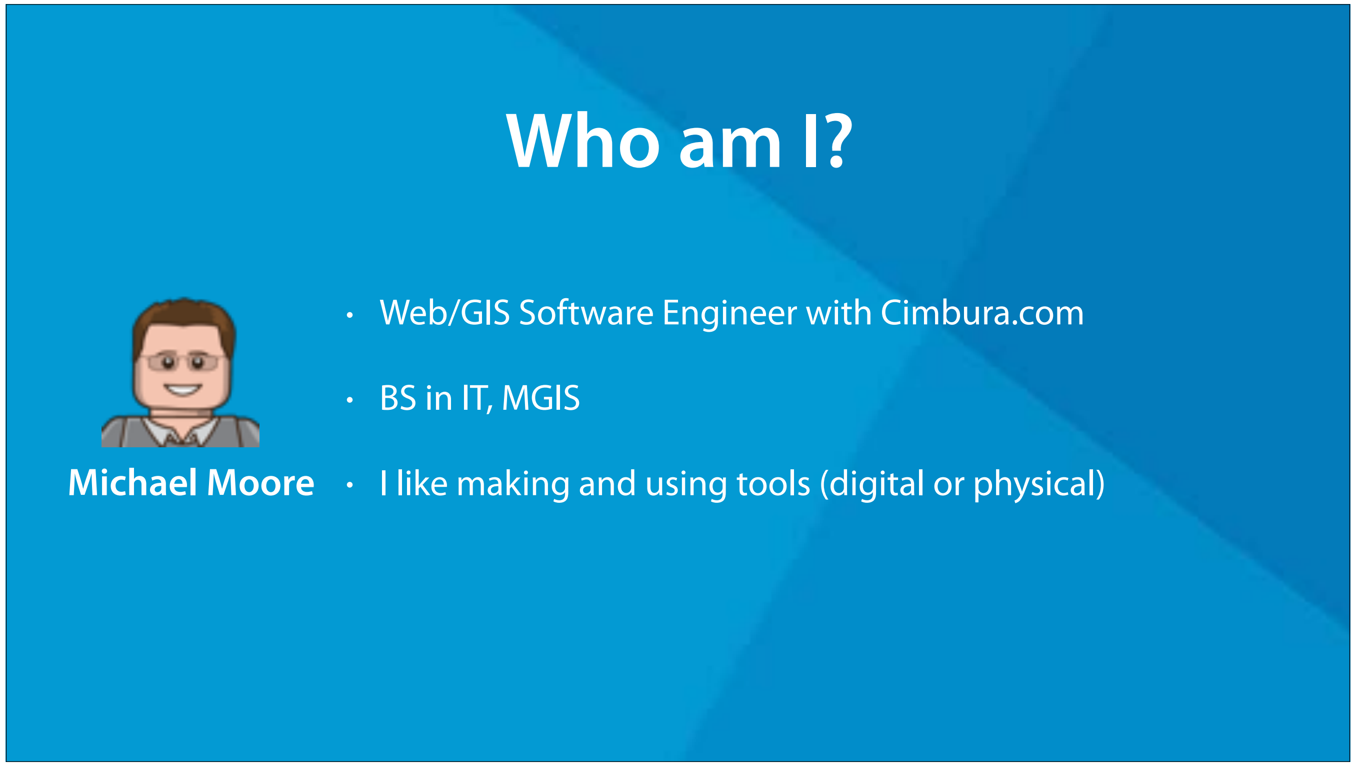
MySQL

The Right Database for GIS … Sometimes



Who am I?

Michael Moore

- Web/GIS Software Engineer with Cimbura.com
- BS in IT, MGIS
- I like making and using tools (digital or physical)

Cimbura.com

BRILLIANT SOLUTIONS

FileMaker
An Apple Subsidiary
Business Alliance
Platinum Reseller

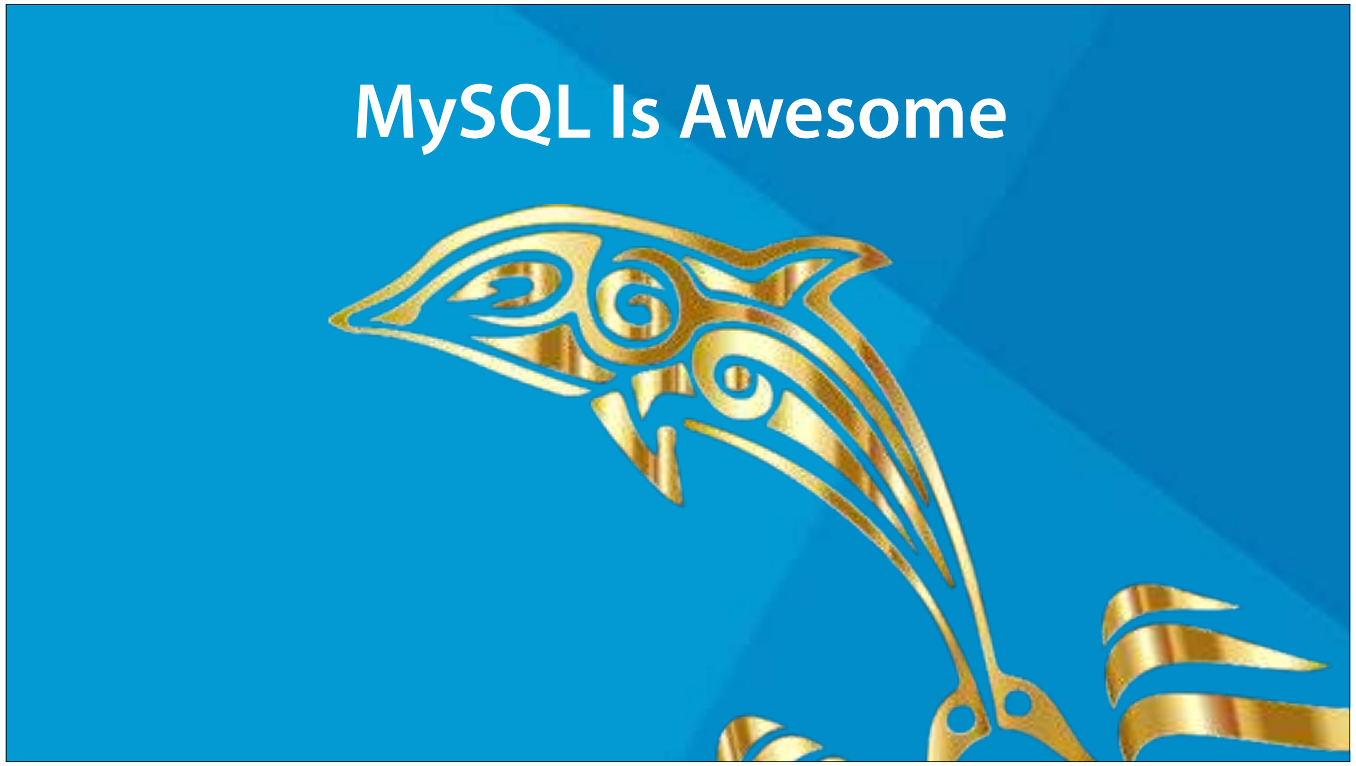
GIS Web Services

WordPress

I’m most interested in making GIS more accessible to more people. I think the way to do that is to (1) make GIS data more valuable (by finding value in existing data) and (2) making GIS tools simple enough for people to use

Todays Discussion

- MySQL Is Awesome
- MySQL Has Issues
- Why use MySQL
- How to do GIS in MySQL

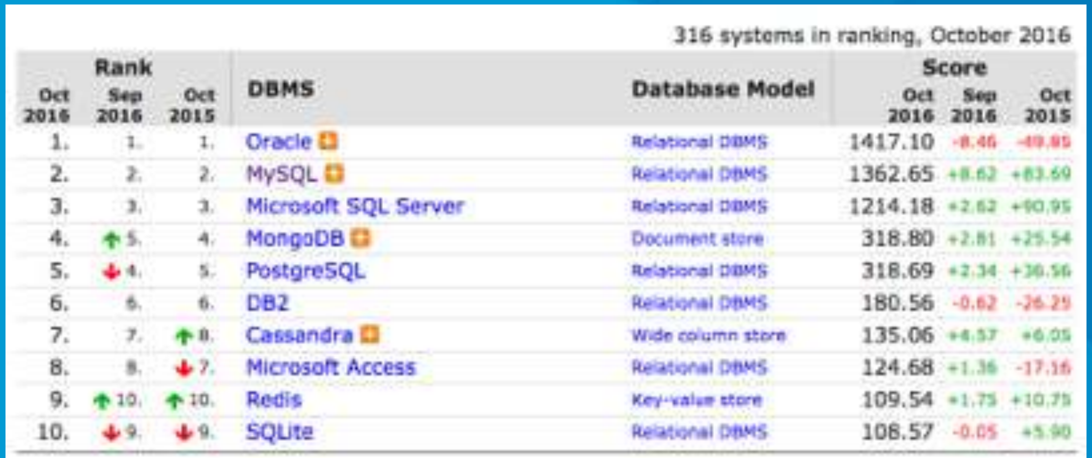


MySQL Is Awesome

It’s Everywhere

- The **M** in *LAMP*
- More than 30% of websites run MySQL
    - Including many huge sites
    - Maybe lots more — it’s hard to find hard numbers
- 7 of the 10 most popular content management systems use it

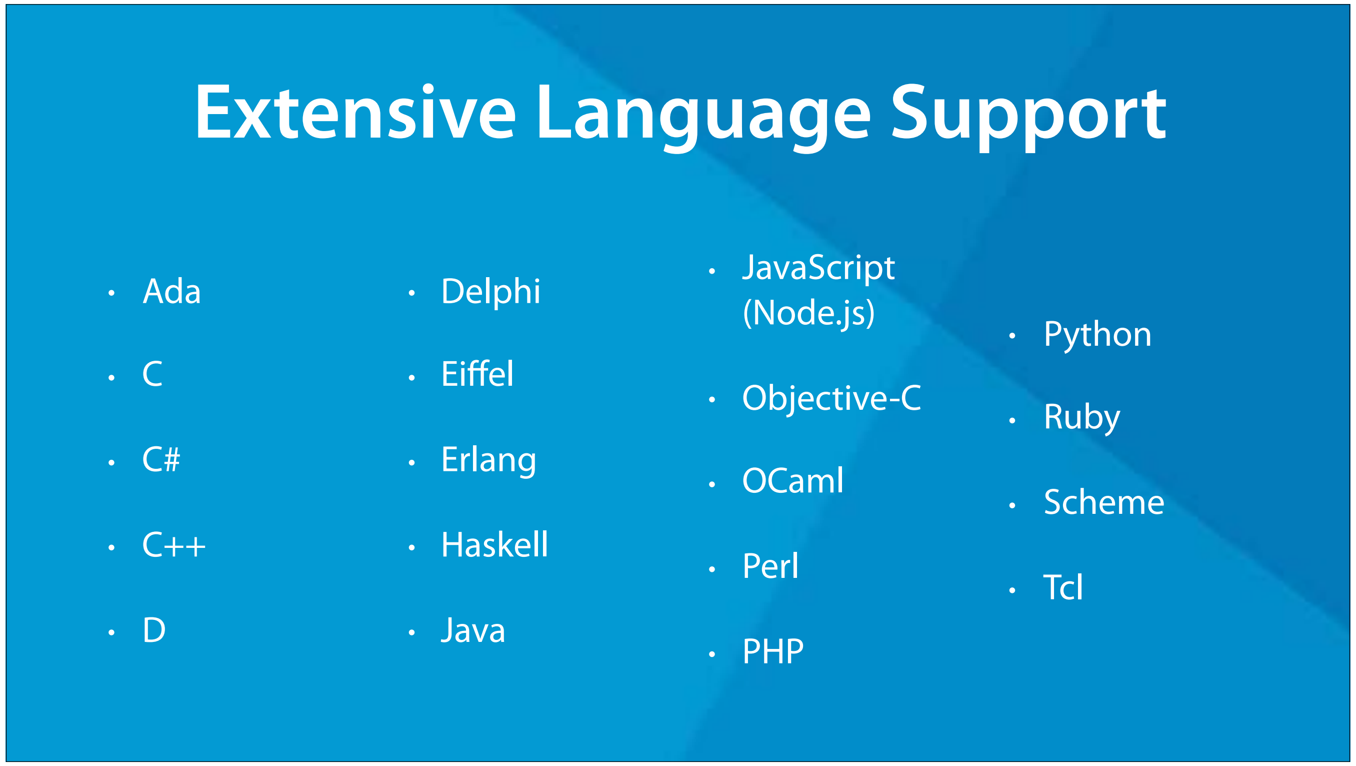
2nd Most Popular DB



316 systems in ranking, October 2016

Rank			DBMS	Database Model	Score		
Oct 2016	Sep 2016	Oct 2015			Oct 2016	Sep 2016	Oct 2015
1.	1.	1.	Oracle ⊞	Relational DBMS	1417.10	-8.46	-49.85
2.	2.	2.	MySQL ⊞	Relational DBMS	1362.65	+8.62	+83.69
3.	3.	3.	Microsoft SQL Server	Relational DBMS	1214.18	+2.62	+90.95
4.	↑ 5.	4.	MongoDB ⊞	Document store	318.80	+2.81	+25.54
5.	↓ 4.	5.	PostgreSQL	Relational DBMS	318.69	+2.34	+36.56
6.	6.	6.	DB2	Relational DBMS	180.56	-0.62	-26.29
7.	7.	↑ 8.	Cassandra ⊞	Wide column store	135.06	+4.57	+6.05
8.	8.	↓ 7.	Microsoft Access	Relational DBMS	124.68	+1.36	-17.16
9.	↑ 10.	↑ 10.	Redis	Key-value store	109.54	+1.75	+10.75
10.	↓ 9.	↓ 9.	SQLite	Relational DBMS	108.57	-0.05	+5.90

db-engines.com

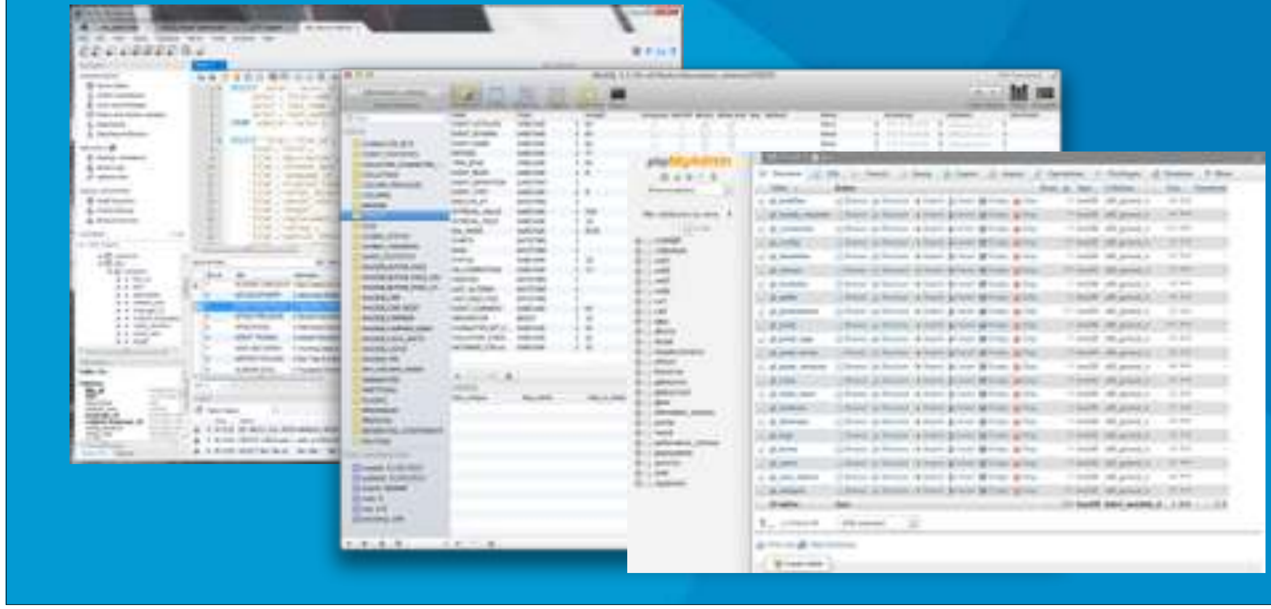


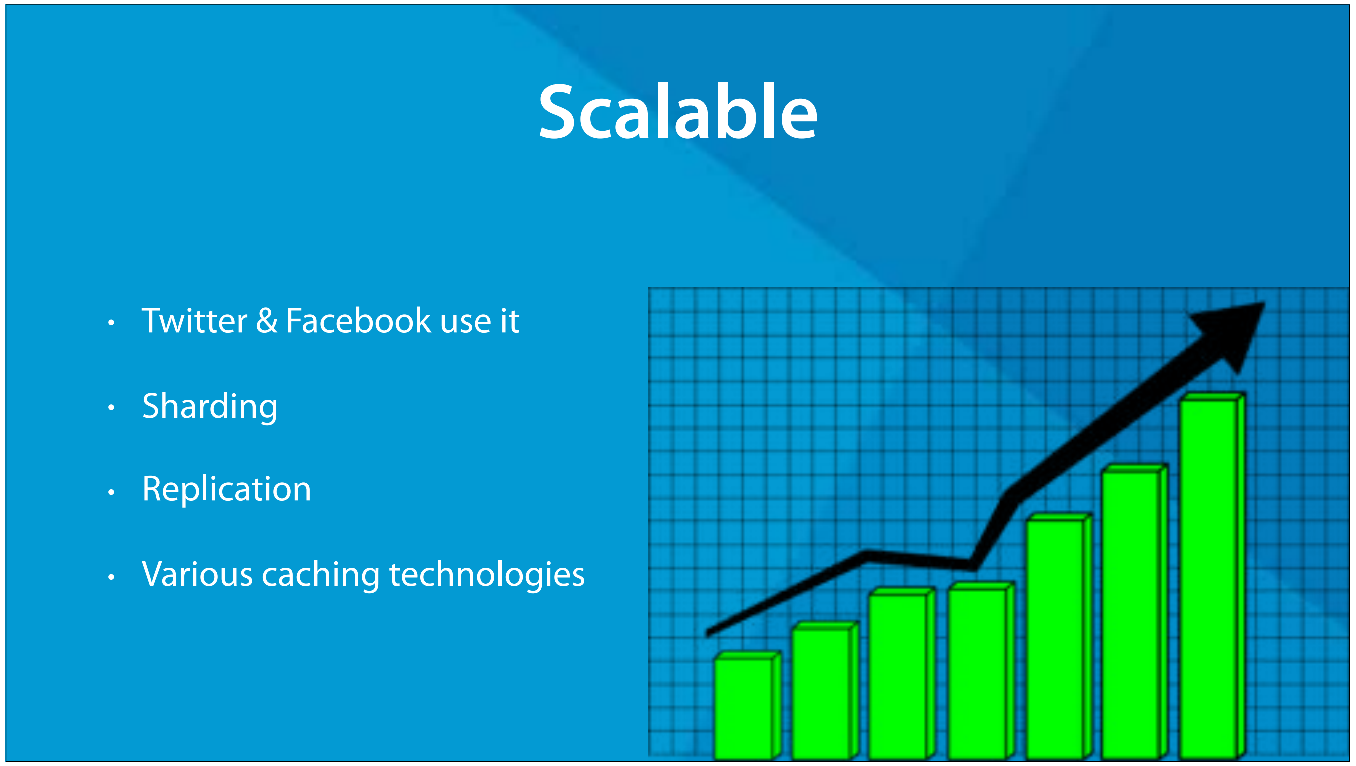
Extensive Language Support

- Ada
- C
- C#
- C++
- D
- Delphi
- Eiffel
- Erlang
- Haskell
- Java
- JavaScript (Node.js)
- Objective-C
- OCaml
- Perl
- PHP
- Python
- Ruby
- Scheme
- Tcl

and probably more!

Admin Tools Galore!





Scalable

- Twitter & Facebook use it
- Sharding
- Replication
- Various caching technologies

Memcached, Redis

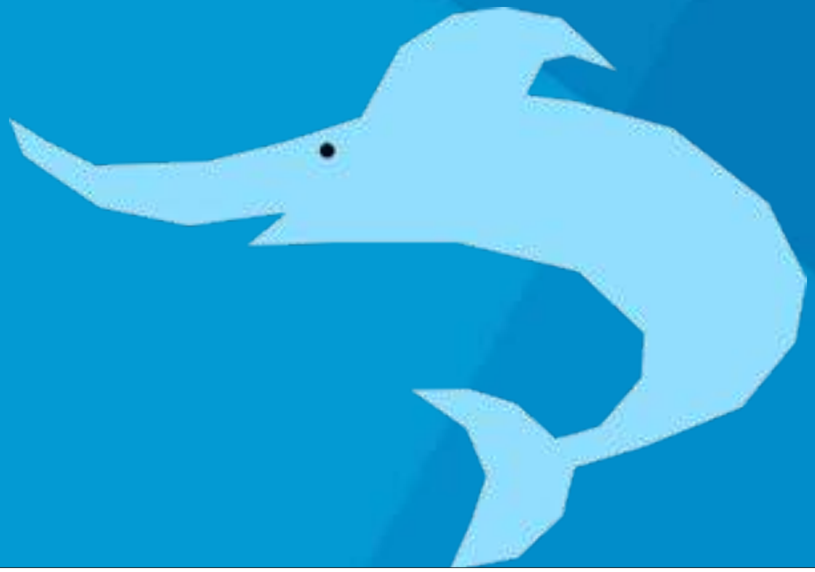
GIS Support

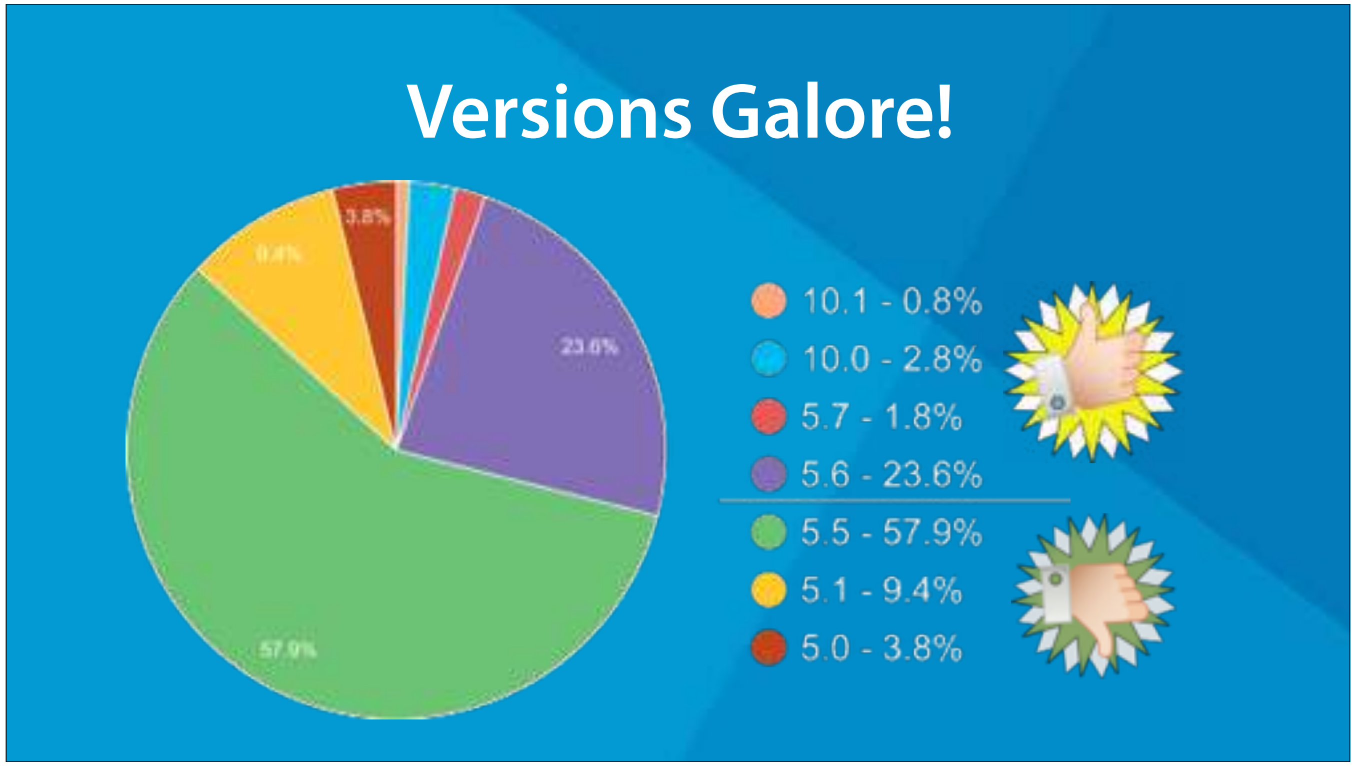
and of course, the reason you’re all here, GIS Support. Or actually…

GIS Support*

MySQL Has Issues

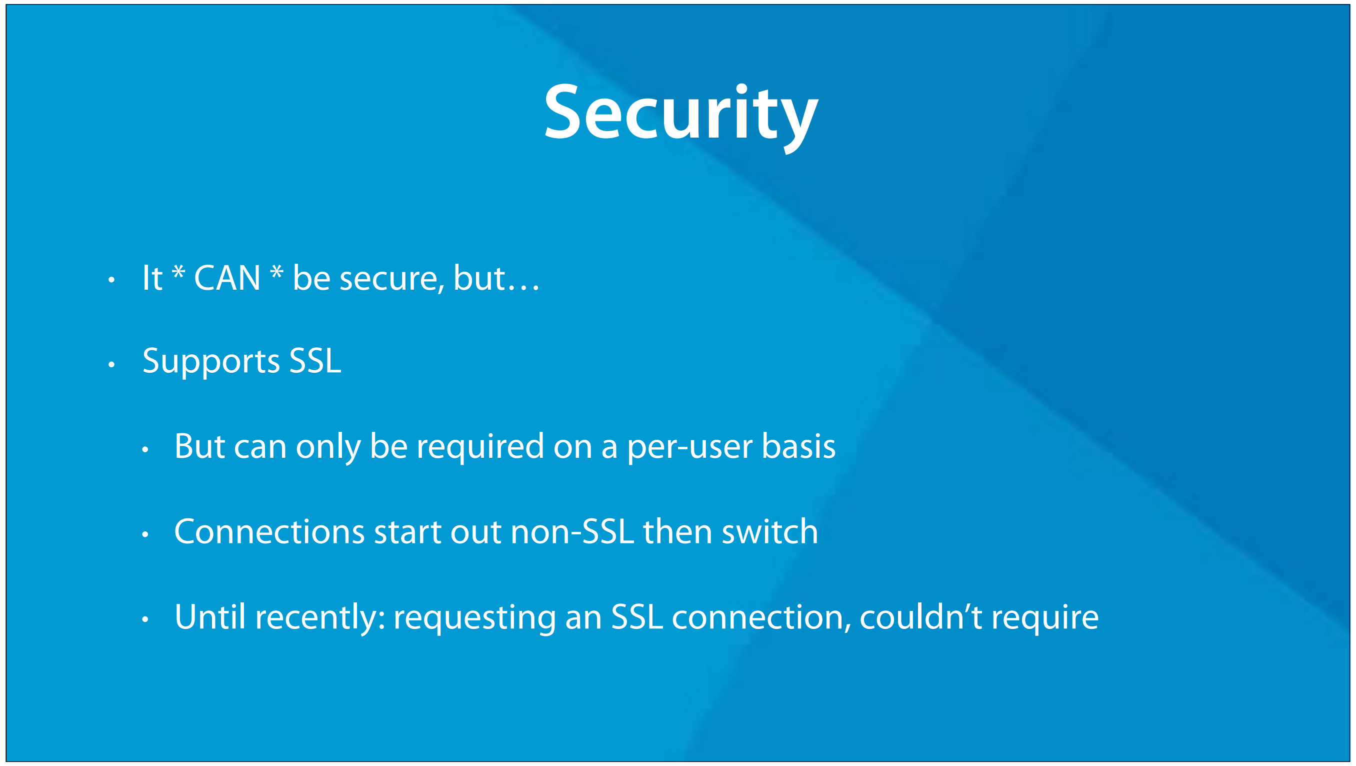
MySQL Has Issues





Versions Galore!

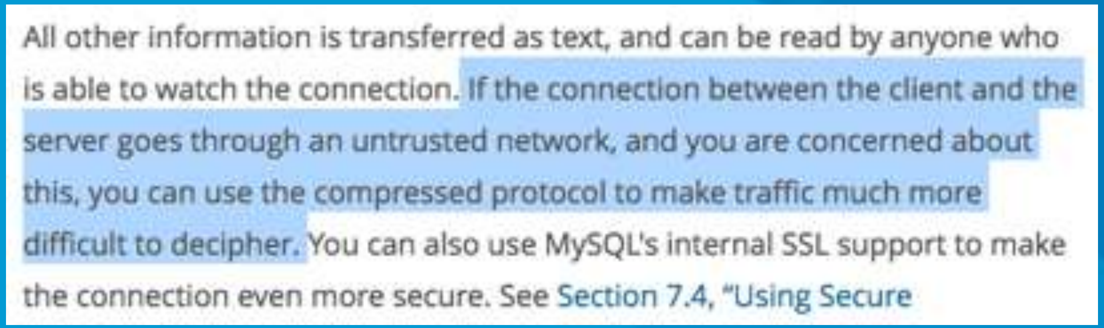
5.0 — 2005
5.1 — 2008
5.5 — 2010
5.6 — 2013
5.7 — 2015
10.0 — 2012
10.1 — 2014



Security

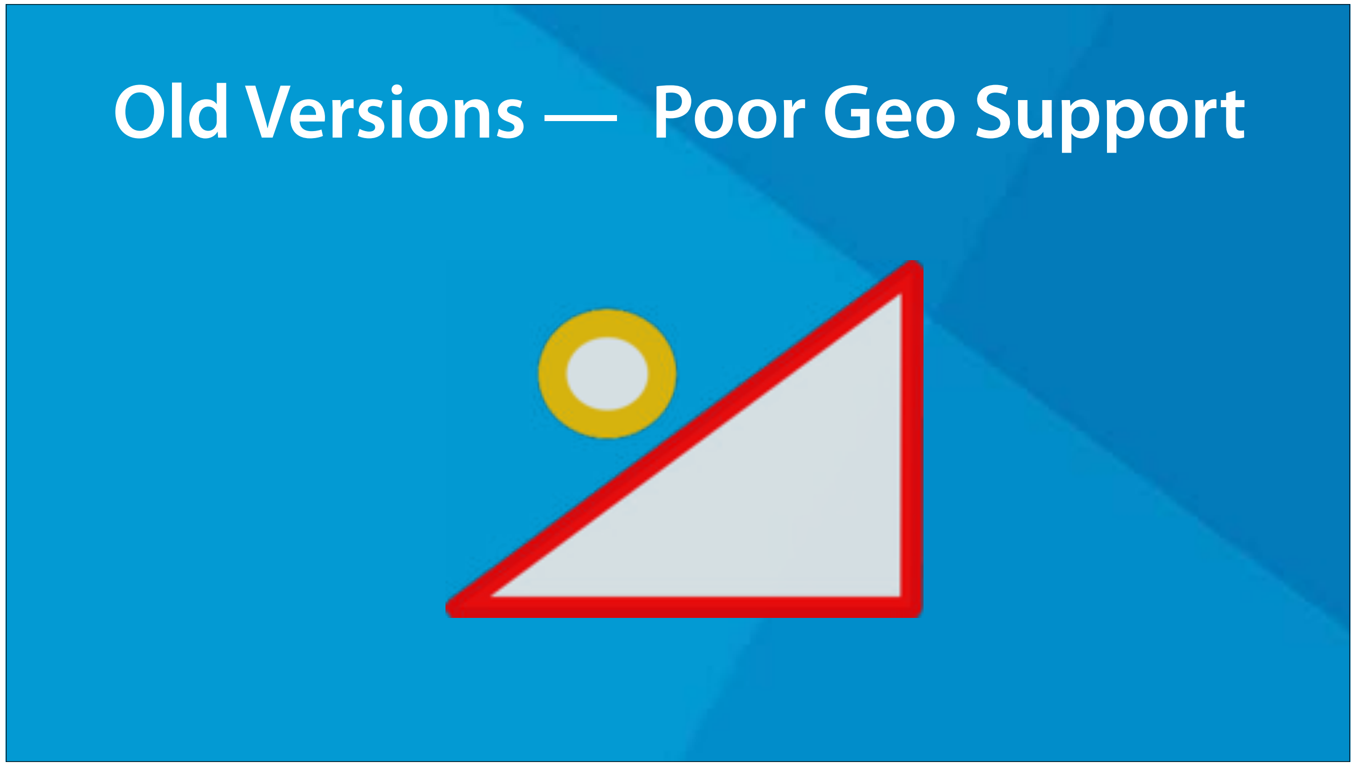
- It * CAN * be secure, but…
- Supports SSL
  - But can only be required on a per-user basis
  - Connections start out non-SSL then switch
  - Until recently: requesting an SSL connection, couldn’t require

Security



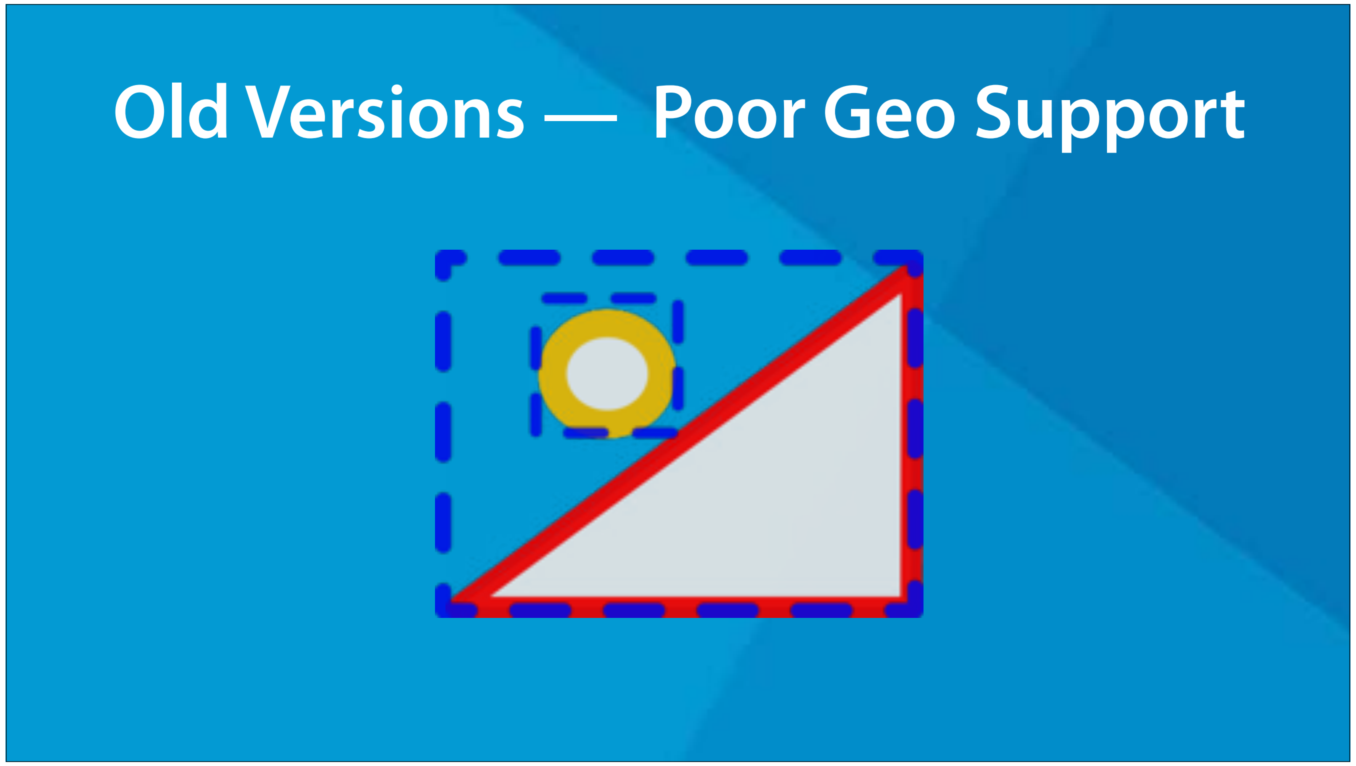
All other information is transferred as text, and can be read by anyone who is able to watch the connection. If the connection between the client and the server goes through an untrusted network, and you are concerned about this, you can use the compressed protocol to make traffic much more difficult to decipher. You can also use MySQL's internal SSL support to make the connection even more secure. See Section 7.4, "Using Secure

That’s like saying, if you want to make it hard for someone to read your postcard, you should just crumple it up in a ball before you mail it.



Old Versions — Poor Geo Support

Do these two shapes intersect?

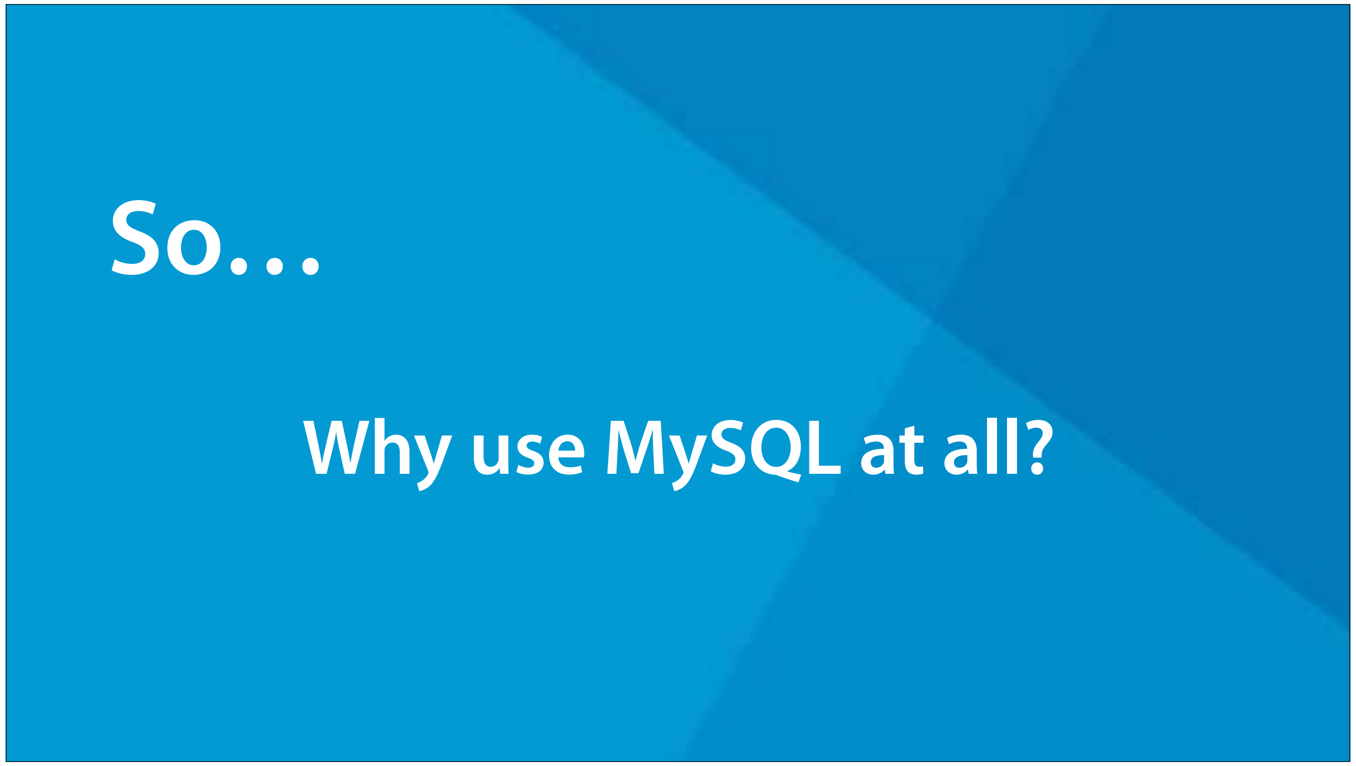


Old Versions — Poor Geo Support

Yes, if you’re using BBOX functions.

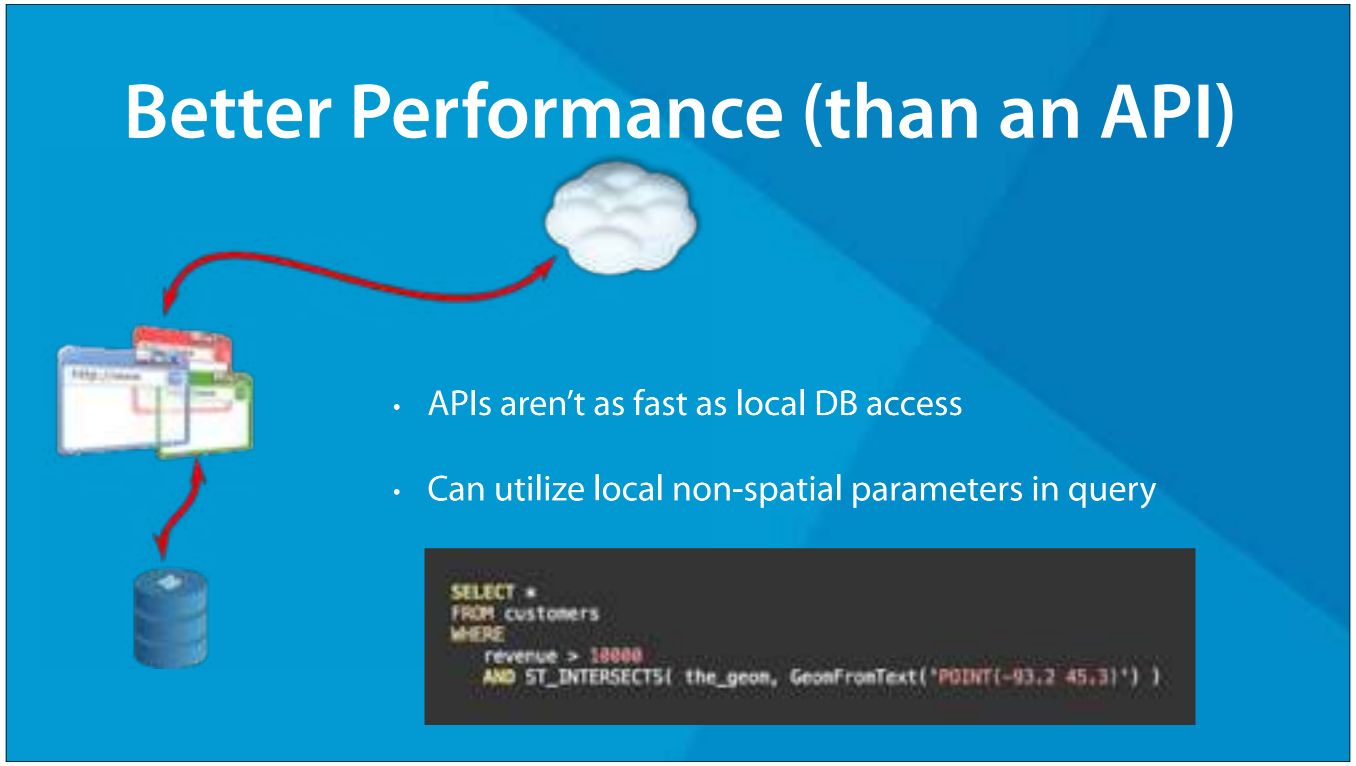
Old Versions — Poor Geo Support

- 71% — Bounding Box Queries only!
- 0% — RID support
- 0% — No reprojection support
- 0% — Geography support
- 8% — Write GeoJSON
- <8% — Read GeoJSON



So…

Why use MySQL at all?



Better Performance (than an API)

- APIs aren't as fast as local DB access
- Can utilize local non-spatial parameters in query

```
SELECT *
FROM customers
WHERE
   revenue > 10000
   AND ST_INTERSECTS( the_geom, GeomFromText('POINT(-93.2 45.3)') )
```

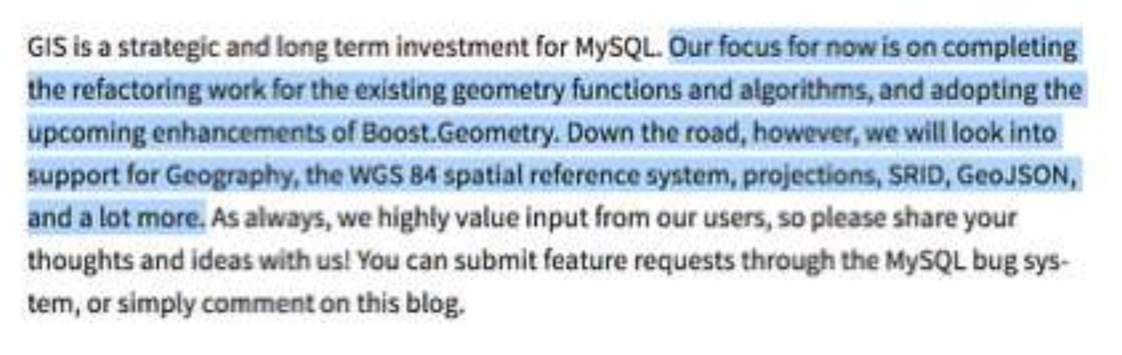
To utilize non-spatial parameters you need to either push data

Land of Opportunity?

Dwight Sipler

- Not many people doing GIS in MySQL
- Easier for a GIS dev to learn MySQL than vice-versa (?)
- Lots of products to integrate with
  - Some with established marketplaces

It’s Going To Get Better!

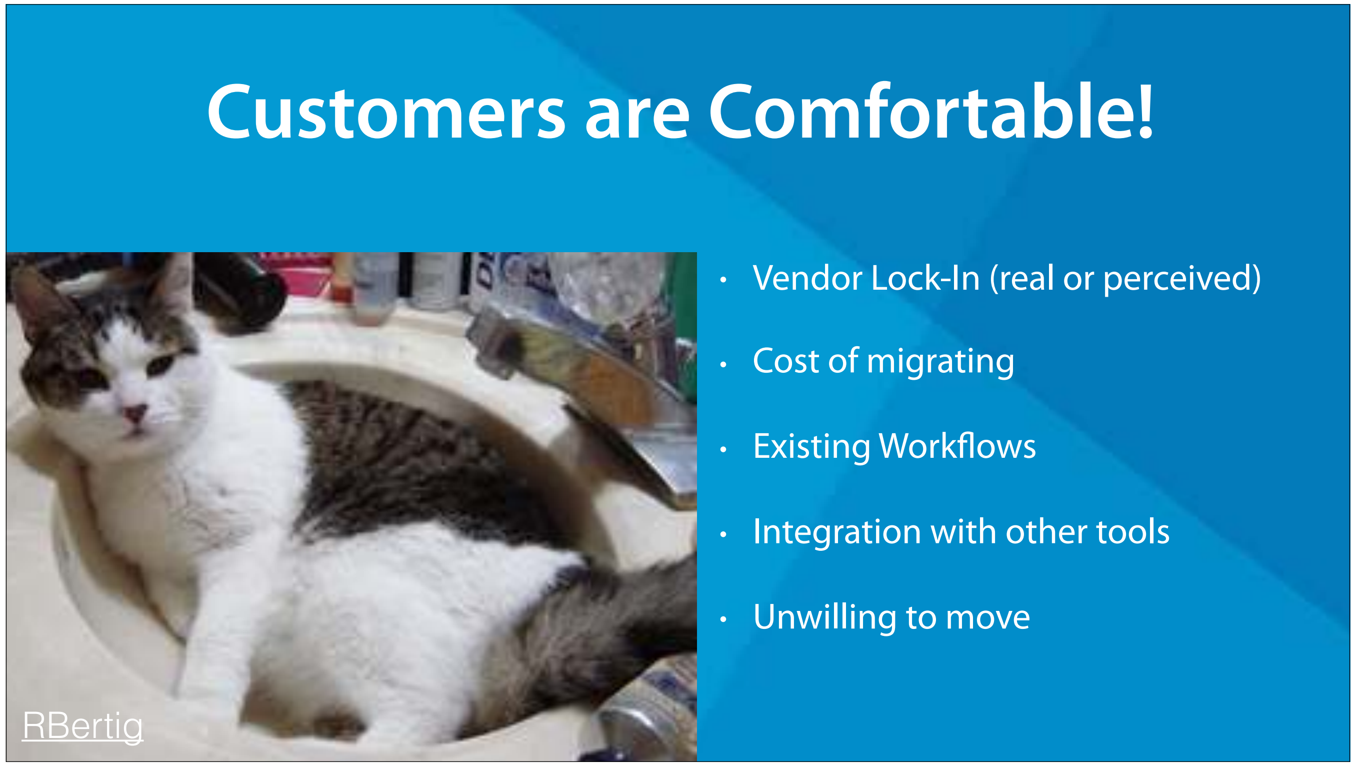


GIS is a strategic and long term investment for MySQL. Our focus for now is on completing the refactoring work for the existing geometry functions and algorithms, and adopting the upcoming enhancements of Boost.Geometry. Down the road, however, we will look into support for Geography, the WGS 84 spatial reference system, projections, SRID, GeoJSON, and a lot more. As always, we highly value input from our users, so please share your thoughts and ideas with us! You can submit feature requests through the MySQL bug system, or simply comment on this blog.

This was posted to the mysqlserverteam.com blog in 2014 and indeed during the 5.7 release cycle each release has had GIS improvements.

The next version, confusingly, is MySQL 8. The release notes for the beta versions indicates that they’re continuing the GIS code refactoring and their source code documentation indicates that they’ve started adding projection support to the code base.

However, possible future enhancements isn’t a great reason to start using a product now. The real reason to use MySQL for GIS today is because sometimes…



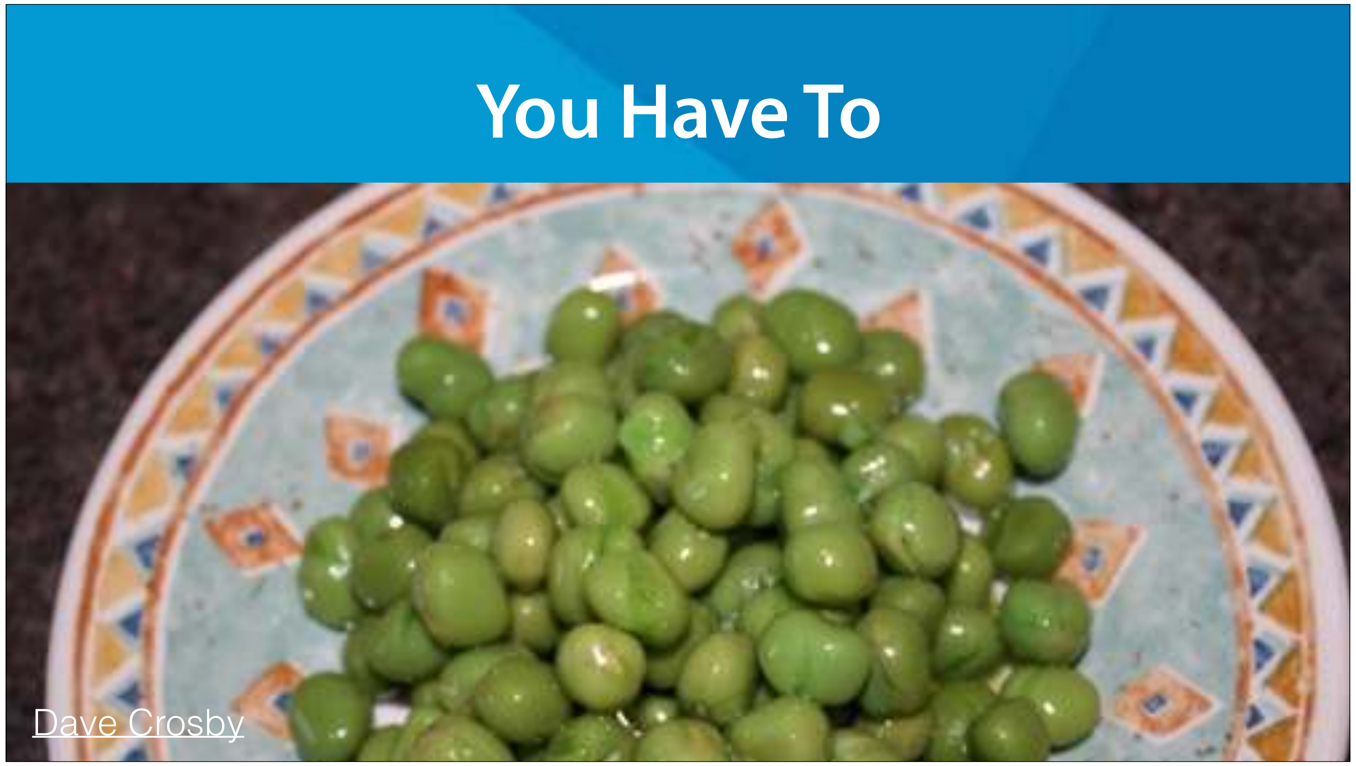
Customers are Comfortable!

- Vendor Lock-In (real or perceived)
- Cost of migrating
- Existing Workflows
- Integration with other tools
- Unwilling to move

RBertig

Customers may be comfortable, even if they know that there are better options out there.

Can you find a spatial content management system that does e-commerce, event ticketing, includes blogging tools, handles membership and event planning and plays well with search engines?



You Have To

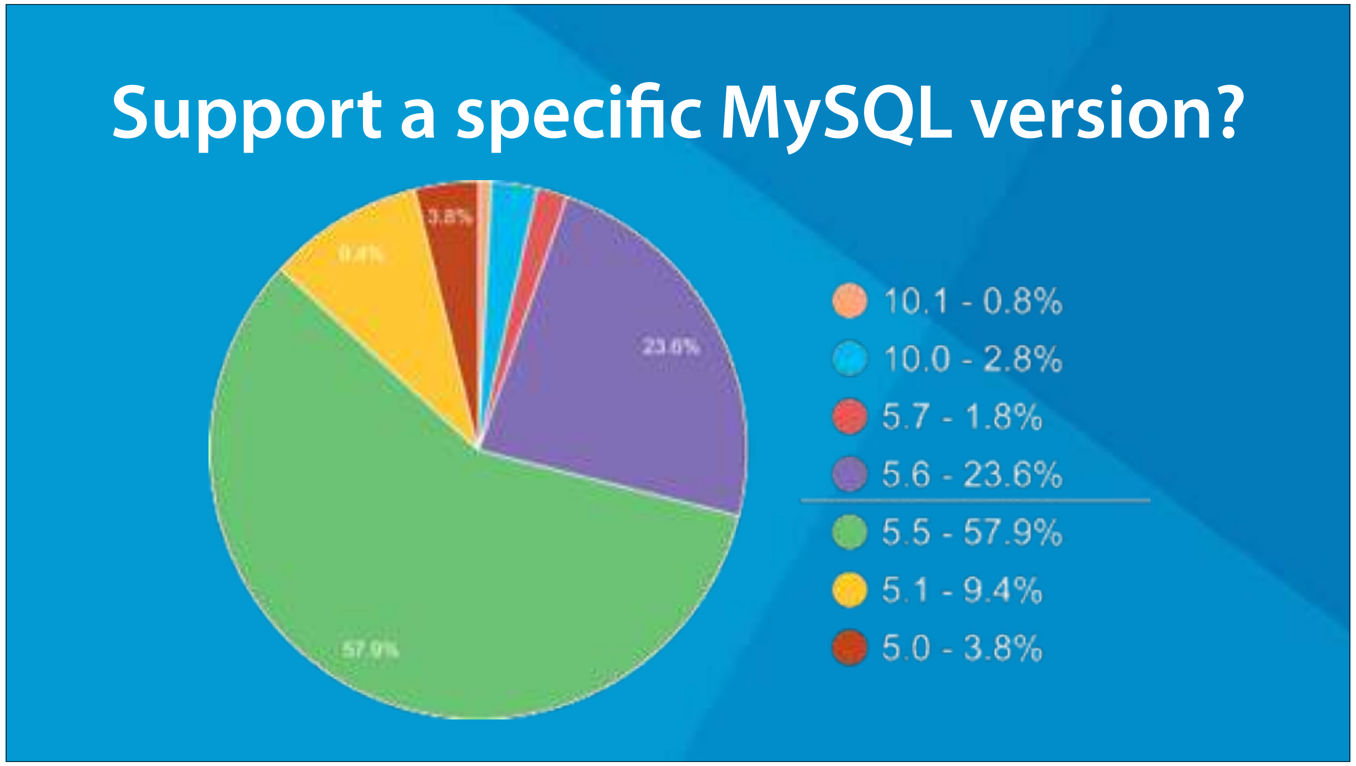
Dave Crosby

This is the real reason to use MySQL here. If you have the option, then you should use PostGIS, or SQLite, or SQL Server, or pretty much anything else.

But if you’re targeting customers that are already using MySQL, then you are going to have to eat your vegetables and use it, even though it’s not ideal.

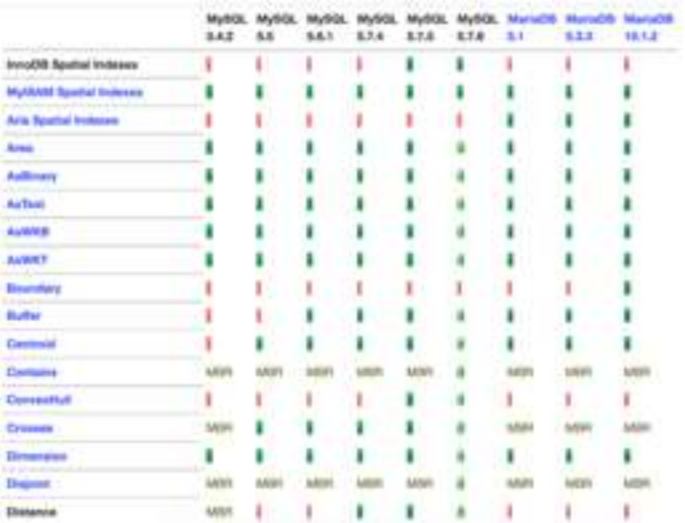
How to do GIS In MySQL

- Decide who you want to support
- Test for features, not versions
- Do as much as you can in code



Support a specific MySQL version?

Support a specific MySQL version?

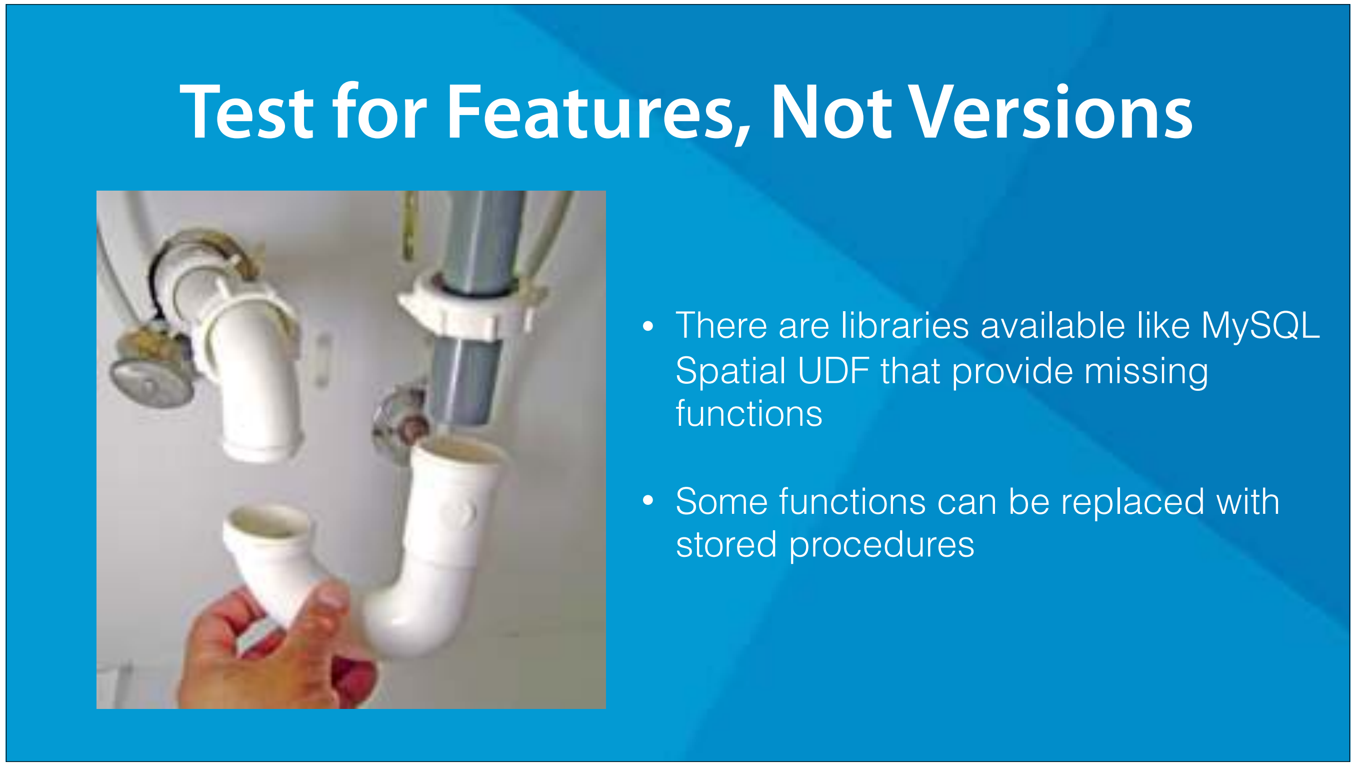


	MySQL 5.4.2	MySQL 5.5	MySQL 5.6.1	MySQL 5.7.4	MySQL 5.7.5	MySQL 5.7.6	MariaDB 5.1	MariaDB 5.3.3	MariaDB 10.1.2
InnoDB Spatial Indexes	✗	✗	✗	✗	✓	✓	✗	✗	✗
MyISAM Spatial Indexes	✓	✓	✓	✓	✓	✓	✓	✓	✓
Aria Spatial Indexes	✗	✗	✗	✗	✗	✗	✓	✓	✓
Area	✓	✓	✓	✓	✓	✓	✓	✓	✓
AsBinary	✓	✓	✓	✓	✓	✓	✓	✓	✓
AsText	✓	✓	✓	✓	✓	✓	✓	✓	✓
AsWKB	✓	✓	✓	✓	✓	✓	✓	✓	✓
AsWKT	✓	✓	✓	✓	✓	✓	✓	✓	✓
Boundary	✗	✗	✗	✗	✗	✗	✗	✗	✓
Buffer	✗	✗	✓	✓	✓	✓	✓	✓	✓
Centroid	✗	✓	✓	✓	✓	✓	✓	✓	✓
Contains	MBR	MBR	MBR	MBR	MBR	✓	MBR	MBR	MBR
ConvexHull	✗	✗	✗	✗	✓	✓	✗	✗	✗
Crosses	MBR	✓	✓	✓	✓	✓	MBR	MBR	MBR
Dimension	✓	✓	✓	✓	✓	✓	✓	✓	✓
Disjoint	MBR	MBR	MBR	MBR	MBR	✓	MBR	MBR	MBR
Distance	MBR	✗	✗	✓	✓	✓	✗	✗	✗

Just Google

“MySQL spatial support matrix”

https://mariadb.com/kb/en/mariadb/mysqlmariadb-spatial-support-matrix/



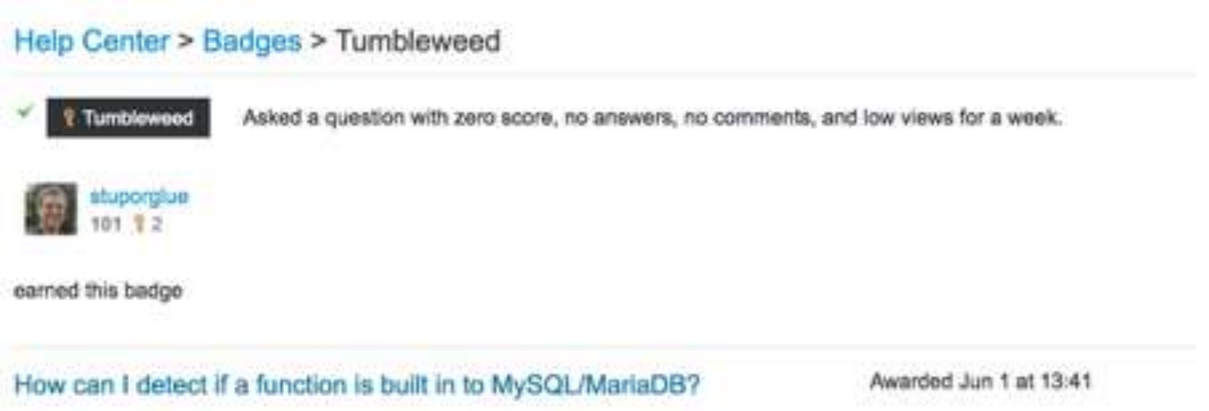
Test for Features, Not Versions

- There are libraries available like MySQL Spatial UDF that provide missing functions
- Some functions can be replaced with stored procedures

UDF, or User Defined Functions are C or C++ code and can link with Proj.4, GEOS or other libraries. Most users can’t install them because it’s too hard or not possible on their host.

Users or code should be able to create stored procedures. This is SQL code, stored in the database and is MUCH slower.

Test for Features, Not Versions



Help Center > Badges > Tumbleweed

✓ Tumbleweed Asked a question with zero score, no answers, no comments, and low views for a week.

stuporglue
101 2

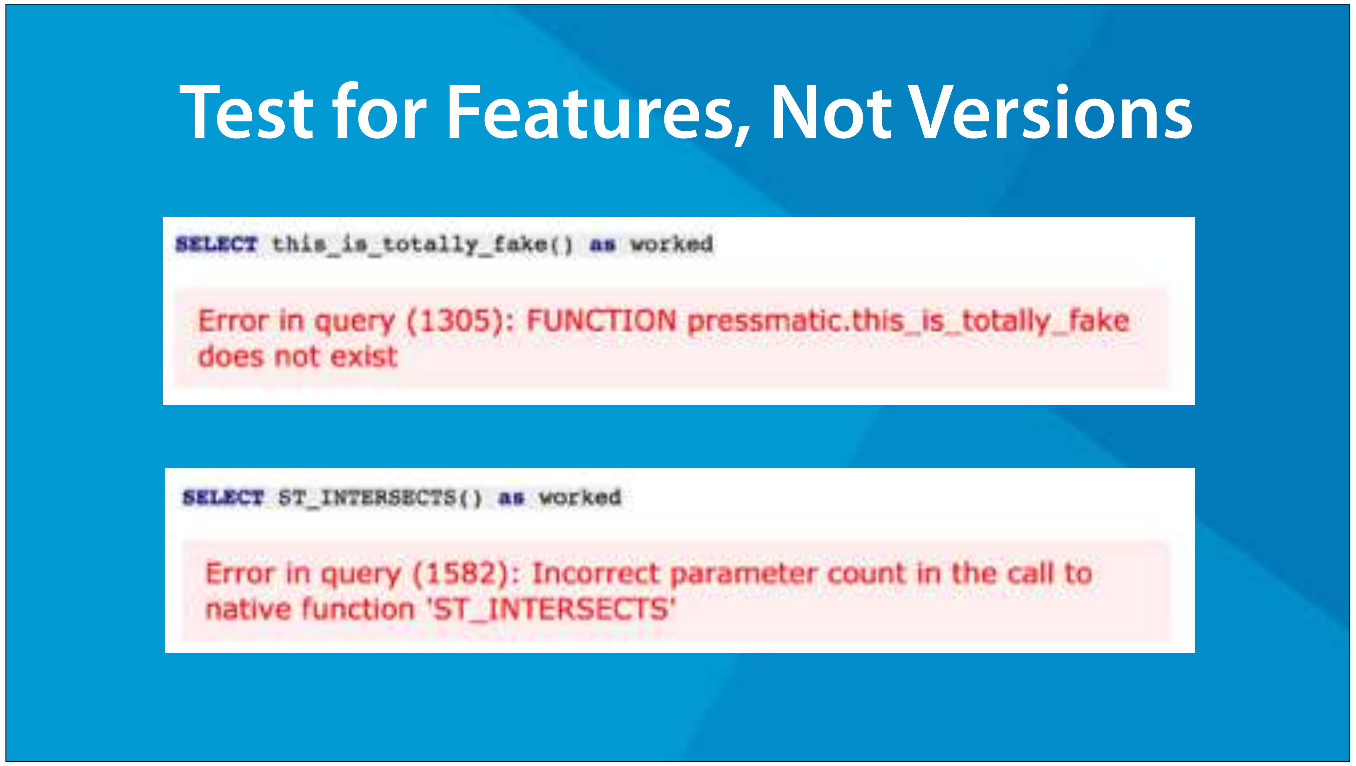
earned this badge

How can I detect if a function is built in to MySQL/MariaDB? Awarded Jun 1 at 13:41

It’s sometimes difficult to figure out *exactly* when a new feature appeared in MySQL. Even if spatial support was available in a certain version, it’s possible that someone compiled MySQL without spatial support (though support is the default, so it’d be kind of weird).

That along with the possibility of users or admins filling in missing functions means you should test for features, not for specific MySQL version numbers.

So, how do you do that? Well, MySQL doesn’t seem to have a way. I don’t use StackOverflow a ton, so it was exciting in a way to get the Tumbleweed badge while trying to figure out how to do this.



Test for Features, Not Versions

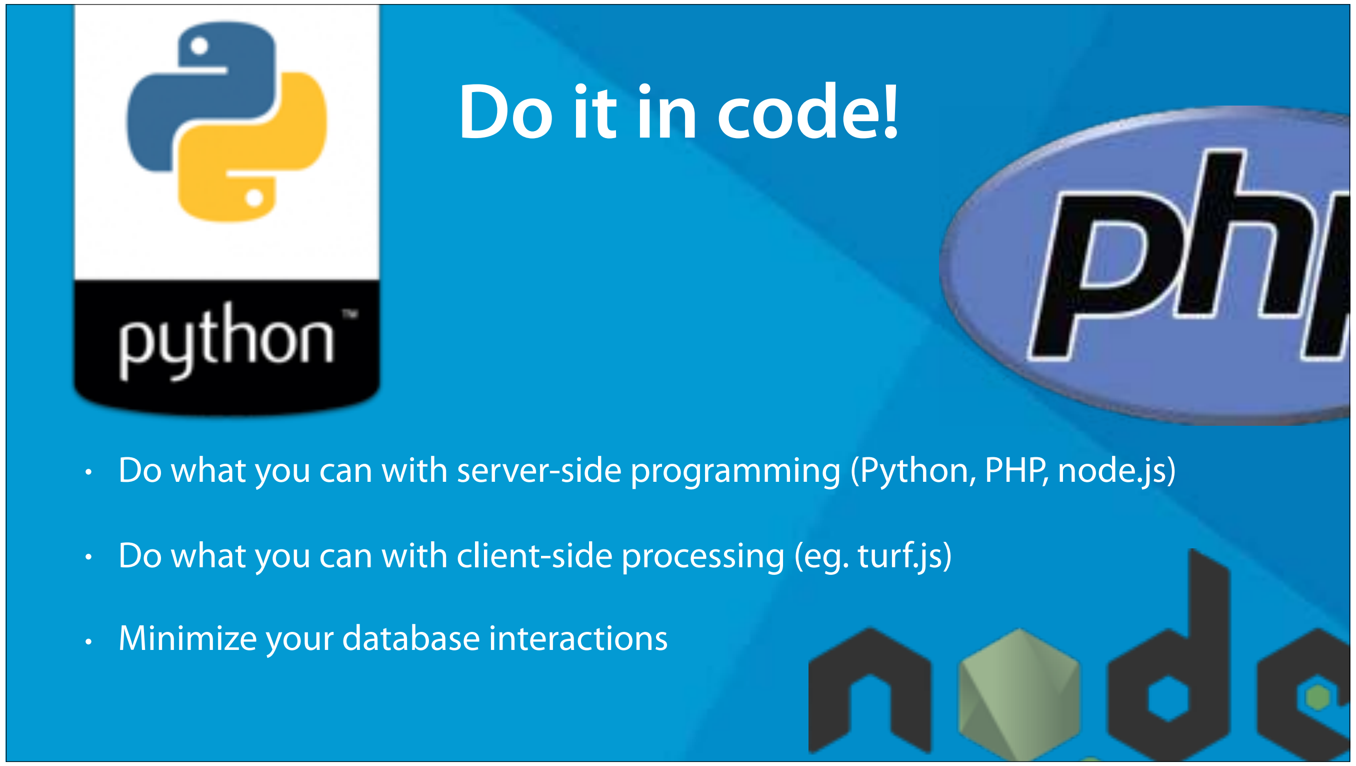
```
SELECT this_is_totally_fake() as worked
```

```
Error in query (1305): FUNCTION pressmatic.this_is_totally_fake
does not exist
```

```
SELECT ST_INTERSECTS() as worked
```

```
Error in query (1582): Incorrect parameter count in the call to
native function 'ST_INTERSECTS'
```

As far as I can tell, the best way to detect if a function is present is to run a query and check the error message.



Do it in code!

- Do what you can with server-side programming (Python, PHP, node.js)
- Do what you can with client-side processing (eg. turf.js)
- Minimize your database interactions

This is completely the opposite of what I like to do with Postgres. With Postgres I like to put everything I can in the database. That way the application code can focus on displaying data instead of working with it.

With MySQL I want to basically just store and retrieve data. Any data massaging or calculations, I'm going to do that with code.

Recap!

- MySQL is a pretty good database (and it’s getting better)
- Old versions are bad and are still out there
- **Right DB if you must integrate with an existing system**
- Rely on code if you can, and use MySQL mostly for storage

Cimbura.com
BRILLIANT SOLUTIONS

Questions