PERCONA LIVE 2023

FerretDB - A MongoDB Alternative Built on PostgreSQL

David Stokes

Technical Evangelist

📅 Wednesday, May 24th

🕒 3:30 PM MST

FerretDB - A MongoDB Alternative Built on PostgreSQL

MongoDB is a life-changing technology for many developers that lets them to build applications faster than using relational databases.

However MongoDB abandoned its Open-Source roots, changing the license to SSPL making it unusable for many Open Source and Commercial Projects.

FerretDB offers a stateless proxy, which converts MongoDB protocol queries to SQL, MongoDB protocol queries to SQL, and uses PostgreSQL as a database engine.

This will be compatible with MongoDB drivers, and should work as a drop-in replacement to MongoDB in many cases.

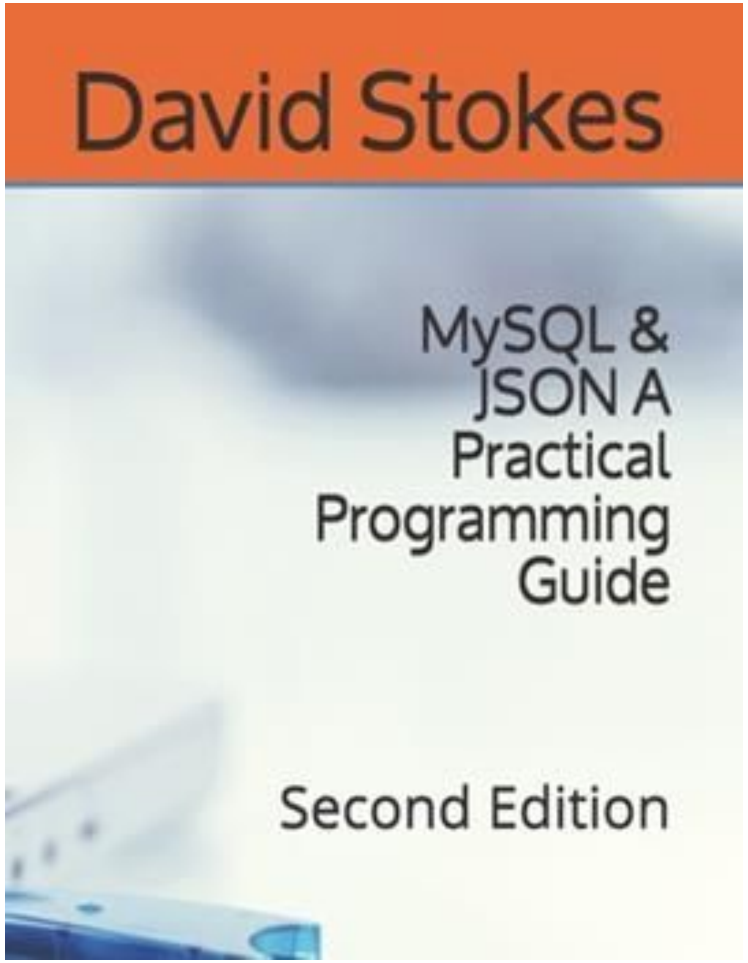
This session will look at this approach, its current status, and hopefully a glimpse of the future.

About me

Technology Evangelist at Percona

Long time open source advocate

Author



David Stokes

MySQL & JSON A Practical Programming Guide

Second Edition

@Stoker David.Stokes@Percona.com

MongoDB did a lot of things right

1. NoSQL wave
2. Lots of initial hand holding
3. No messy Structured Query Language (SQL)
4. Mental model easy for many developers to comprehend
5. Robust enough
6. Did most of the things needed at a basic level

License Change 2018

GNU AGPLv3 to Server Side Public License

https://www.mongodb.com/licensing/server-side-public-license/faq

The market is quickly moving to consume most software as a service. This is a time of incredible opportunity for open source projects, with the potential to foster a new wave of great open source server side software. The reality, however, is that once an open source project becomes interesting, it is too easy for large cloud vendors to capture all the value but contribute nothing back to the community.

Choices

pixtastock.com - 29182961

1. Run version before licence change
   a. The last versions licensed as AGPL version 3 are 4.0.3 (stable) and 4.1.4.
2. Run version with SSPL
   a. Hope you do not violate rules
3. Use something different
   a. FerretDB

FerretDB

https://www.ferretdb.io/

What is it?

FerretDB was founded to become the de-facto open-source substitute to MongoDB.

FerretDB is an open-source proxy, converting the MongoDB 6.0+ wire protocol queries to SQL - using PostgreSQL as a database engine.

Getting it running

```
$  docker run -d --rm --name ferretdb -p 27017:27017 ghcr.io/ferretdb/all-in-one
Unable to find image 'ghcr.io/ferretdb/all-in-one:latest' locally
latest: Pulling from ferretdb/all-in-one
f1f26f570256: Pull complete
1c04f8741265: Pull complete
dffc353b86eb: Pull complete
18c4a9e6c414: Pull complete
81f47e7b3852: Pull complete
5e26c947960d: Pull complete
a2c3dc85e8c3: Pull complete
17df73636f01: Pull complete
713535cdf17c: Pull complete
52278a39eea2: Pull complete
4ded87da67f6: Pull complete
05fae4678312: Pull complete
56b4f4aeea2d: Pull complete
68c486387c4f: Pull complete
5eb3eee800a9: Pull complete
8e5dd809e820: Pull complete
d3e85fce5b45: Pull complete
e6810cdbd43b: Pull complete
Digest: sha256:072312577c1daf469ac77d09284a638dea98b63f4f4334fd54959324847b93aa
Status: Downloaded newer image for ghcr.io/ferretdb/all-in-one:latest
58f00a86bad172674479f3663563af274e0dd3d15249029a403d0c85039b7ab5
```

Getting it running - connect with Mongo shell

```
$ docker exec -it ferretdb mongosh
Current Mongosh Log ID:     6435963392d12db06bdb7ecc
Connecting to:         mongodb://127.0.0.1:27017/?directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.8.0
Using MongoDB:         6.0.42
Using Mongosh:         1.8.0
```

Using Percona’s Mongosh mongodb://username:password@127.0.0.1/ferretdb?authMechanism=PLAIN

```
mongosh mongodb://username:password@127.0.0.1/ferretdb?authMechanism=PLAIN
Current Mongosh Log ID:     6464d4b9bbc0298bcdf87892
Connecting to:
mongodb://<credentials>@127.0.0.1/ferretdb?authMechanism=PLAIN&directConnection=true&serverSelectionTimeoutMS=2000&appName=mongosh+1.8.0
Using MongoDB:              6.0.42
Using Mongosh:              1.8.0

For mongosh info see: https://docs.mongodb.com/mongodb-shell/

------
   The server generated these startup warnings when booting
   2023-05-17T13:20:57.408Z: Powered by FerretDB v1.1.0 and PostgreSQL 15.3.
   2023-05-17T13:20:57.408Z: Please star us on GitHub: https://github.com/FerretDB/FerretDB.
   2023-05-17T13:20:57.408Z: The telemetry state is undecided.
   2023-05-17T13:20:57.408Z: Read more about FerretDB telemetry and how to opt out at https://beacon.ferretdb.io.
------

------
   Enable MongoDB's free cloud-based monitoring service, which will then receive and display
   metrics about your deployment (disk utilization, CPU, operation statistics, etc).

   The monitoring data will be available on a MongoDB website with a unique URL accessible to you
   and anyone you share the URL with. MongoDB may use this information to make product
   improvements and to suggest MongoDB products and deployment options to you.

   To enable free monitoring, run the following command: db.enableFreeMonitoring()
   To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
------

ferretdb>
```

Create a collection

```
ferretdb> show collections;

ferretdb> db.createCollection('test')
{ ok: 1 }
```

Create a collection & add data

```
ferretdb> db.test.insertOne({name: "Dave", state: "Texas"})
{
  acknowledged: true,
  insertedId: ObjectId("646631413e5dba560a774375")
}
ferretdb> db.test.insertOne({name: "Mac", state: "Texas"})
{
  acknowledged: true,
  insertedId: ObjectId("6466314d3e5dba560a774376")
}
```

Simple Find

```
ferretdb> db.test.find({name: "Dave"})
[
  {
    _id: ObjectId("646631413e5dba560a774375"),
    name: 'Dave',
    state: 'Texas'
  }
]
```

Find by state

```
ferretdb> db.test.find({state: "Texas"})
[
  {
    _id: ObjectId("646631413e5dba560a774375"),
    name: 'Dave',
    state: 'Texas'
  },
  {
    _id: ObjectId("6466314d3e5dba560a774376"),
    name: 'Mac',
    state: 'Texas'
  }
]
ferretdb>
```

$eq:

```
ferretdb> db.test.find({state:{$eq: "Texas"}})
[
  {
    _id: ObjectId("646631413e5dba560a774375"),
    name: 'Dave',
    state: 'Texas'
  },
  {
    _id: ObjectId("6466314d3e5dba560a774376"),
    name: 'Mac',
    state: 'Texas'
  }
]
ferretdb>
```

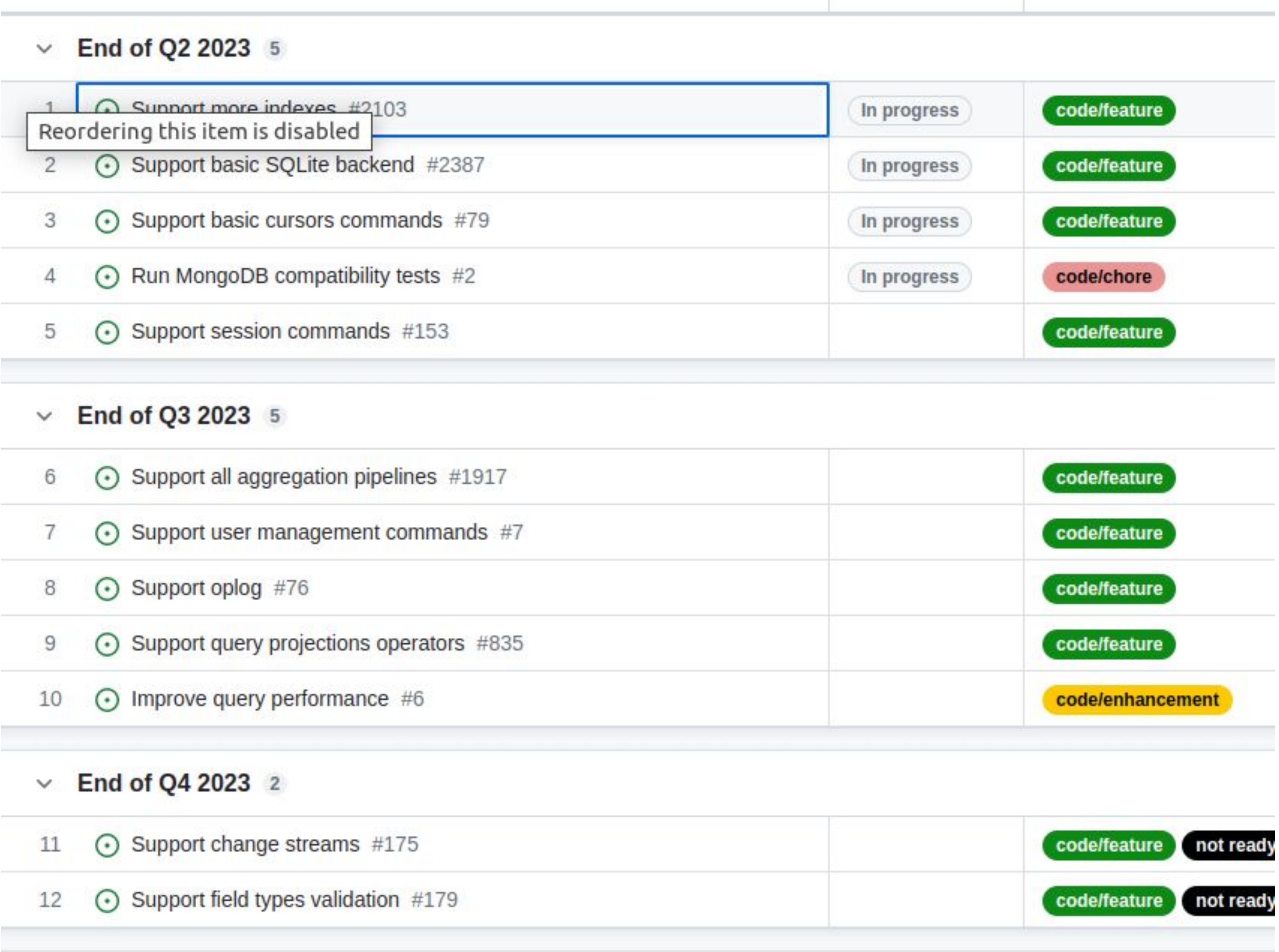
So it works?

What is it missing?

Known differences

1. FerretDB uses the same protocol error names and codes, but the exact error messages may be different in some cases.
2. FerretDB does not support NUL (`\0`) characters in strings.
3. FerretDB does not support nested arrays.
4. FerretDB converts `-0` (negative zero) to `0` (positive zero).
5. Document restrictions:
   - document keys must not contain `.` sign;
   - document keys must not start with `$` sign;
   - document fields of double type must not contain `Infinity`, `-Infinity`, or `NaN` values.
6. When insert command is called, insert documents must not have duplicate keys.
7. Update command restrictions:
   - update operations producing `Infinity`, `-Infinity`, or `NaN` are not supported.
8. Database and collection names restrictions:
   - name cannot start with the reserved prefix `_ferretdb_`;
   - database name must not include non-latin letters;
   - collection name must be valid UTF-8 characters;
   - database name must not start with a number;
9. For Tigris, FerretDB requires Tigris schema validation for `create` command: validator must be set as `$tigrisSchemaString`. The value must be a JSON string representing JSON schema in Tigris format.
10. FerretDB offers the same validation rules for the `scale` parameter in both the `collStats` and `dbStats` commands. If an invalid `scale` value is provided in the `dbStats` command, the same error codes will be triggered as with the `collStats` command.

Roadmap – https://github.com/orgs/FerretDB/projects/2/views/1



Reordering this item is disabled

End of Q2 2023 5

1	⊙ Support more indexes #2103	In progress	code/feature
2	⊙ Support basic SQLite backend #2387	In progress	code/feature
3	⊙ Support basic cursors commands #79	In progress	code/feature
4	⊙ Run MongoDB compatibility tests #2	In progress	code/chore
5	⊙ Support session commands #153		code/feature

End of Q3 2023 5

6	⊙ Support all aggregation pipelines #1917		code/feature
7	⊙ Support user management commands #7		code/feature
8	⊙ Support oplog #76		code/feature
9	⊙ Support query projections operators #835		code/feature
10	⊙ Improve query performance #6		code/enhancement

End of Q4 2023 2

11	⊙ Support change streams #175		code/feature not ready
12	⊙ Support field types validation #179		code/feature not ready

Beyond docker - https://github.com/FerretDB/FerretDB/releases/tag/v1.0.0

.DEB and .RPM packages are available

Documentation lacking

Backends

PostgreSQL
PostgreSQL backend is our main backend and is fully supported.

PostgreSQL should be configured with UTF8 encoding and one of the following locales: POSIX, C, C.UTF8, en_US.UTF8.

MongoDB databases are mapped to PostgreSQL schemas in a single PostgreSQL database that should be created in advance. MongoDB collections are mapped to PostgreSQL tables. MongoDB documents are mapped to rows with a single JSONB column. Those mappings might change as we work on improving compatibility and performance, but no breaking changes will be introduced without a major version bump.

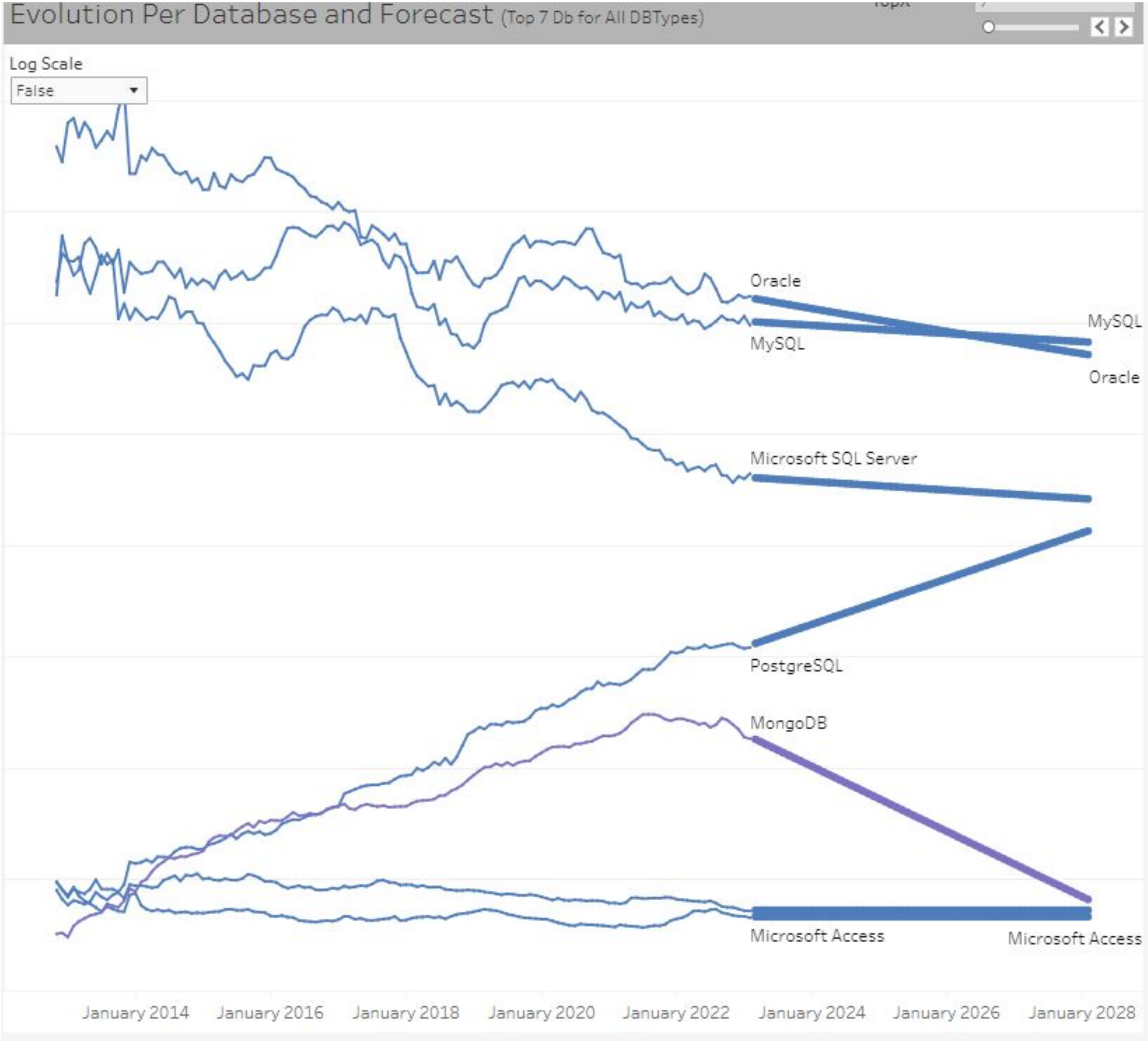
SQLite (alpha)
We are working on SQLite backend. It is not officially supported yet.

Tigris (beta)
We also support the Tigris backend on a beta level. Read more here.

SAP HANA (alpha)
Currently, we are also working with SAP on HANA compatibility. It is not officially supported yet.

Trends

Forcast



Evolution Per Database and Forecast (Top 7 Db for All DBTypes)

Log Scale: False

Thank You!

David.Stokes@Percona.com

@Stoker

Speakerdeck.com/Stoker