De la feuille à l'éléphant

aiven

Matthieu Cornillon
Senior Solution Architect

De la feuille à l'éléphant

Matthieu Cornillon
Senior Solution Architect

De la feuille à l'éléphant

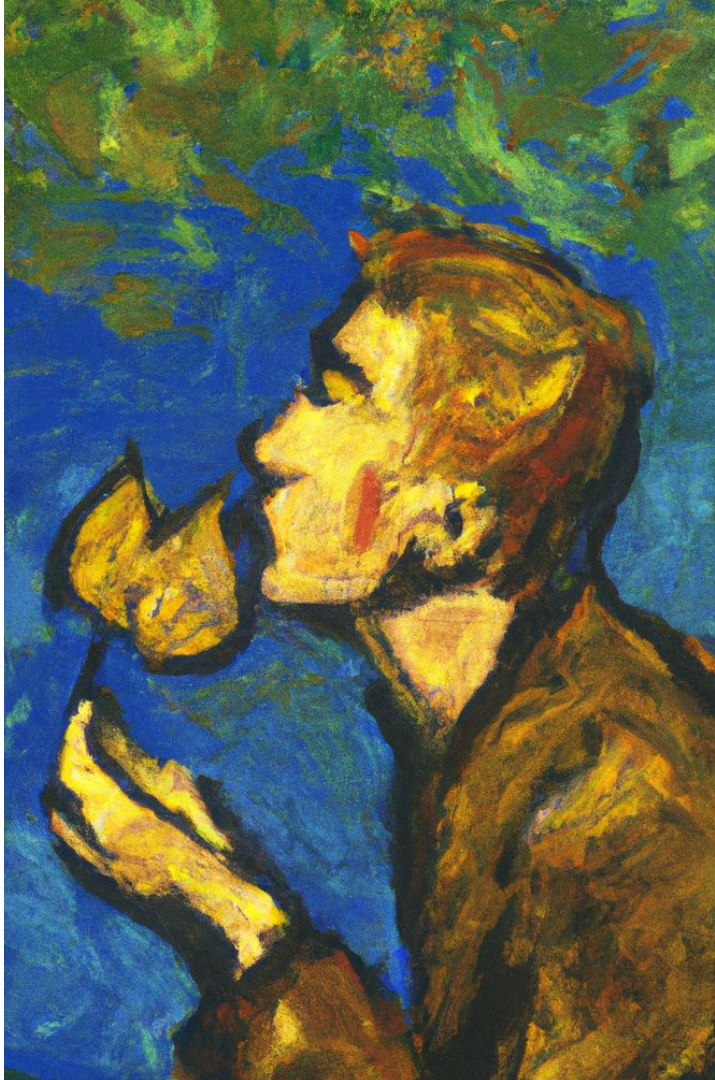
adeo

aiven

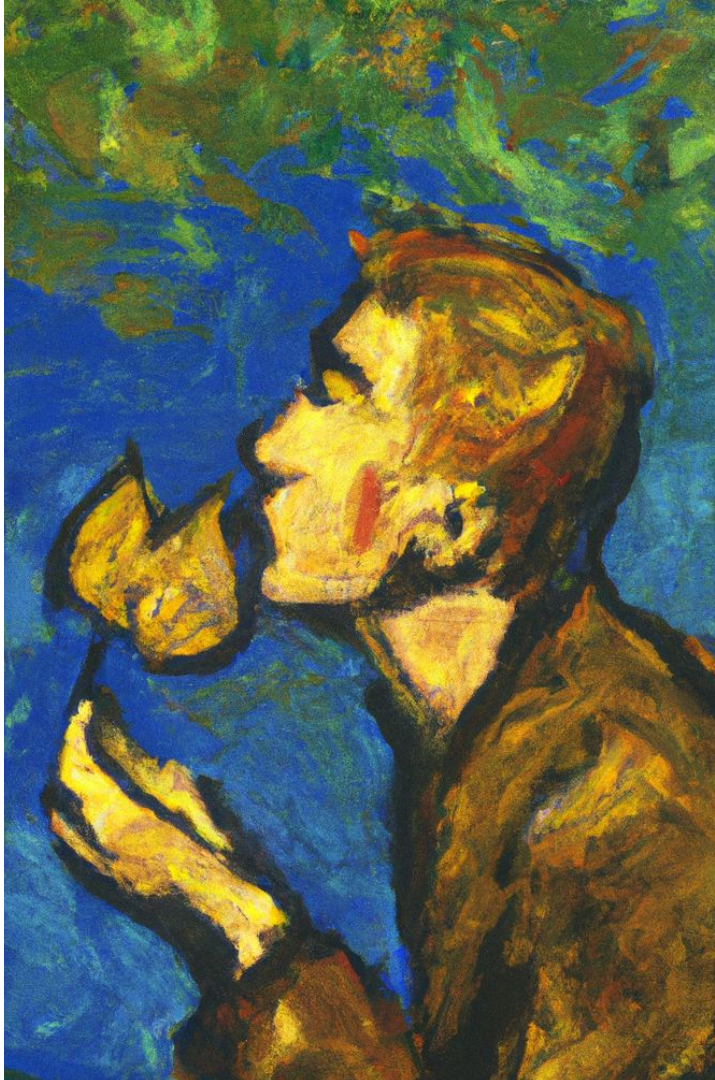
Matthieu Cornillon
Senior Solution Architect

De la feuille à l'éléphant

Matthieu Cornillon
Senior Solution Architect

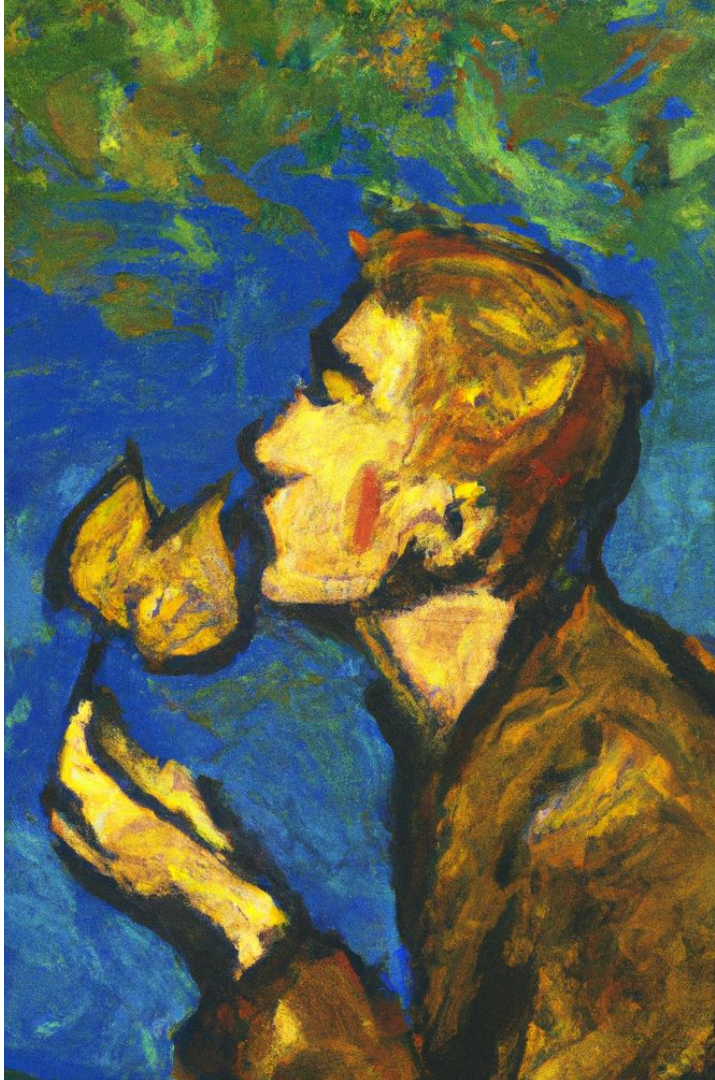


Les développeurs adorent MongoDB...

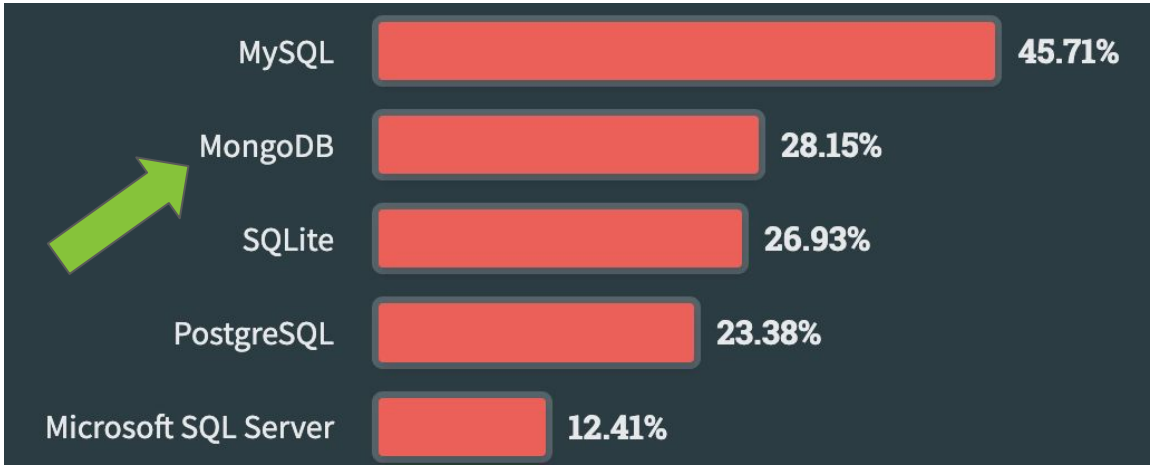


Les développeurs adorent MongoDB...

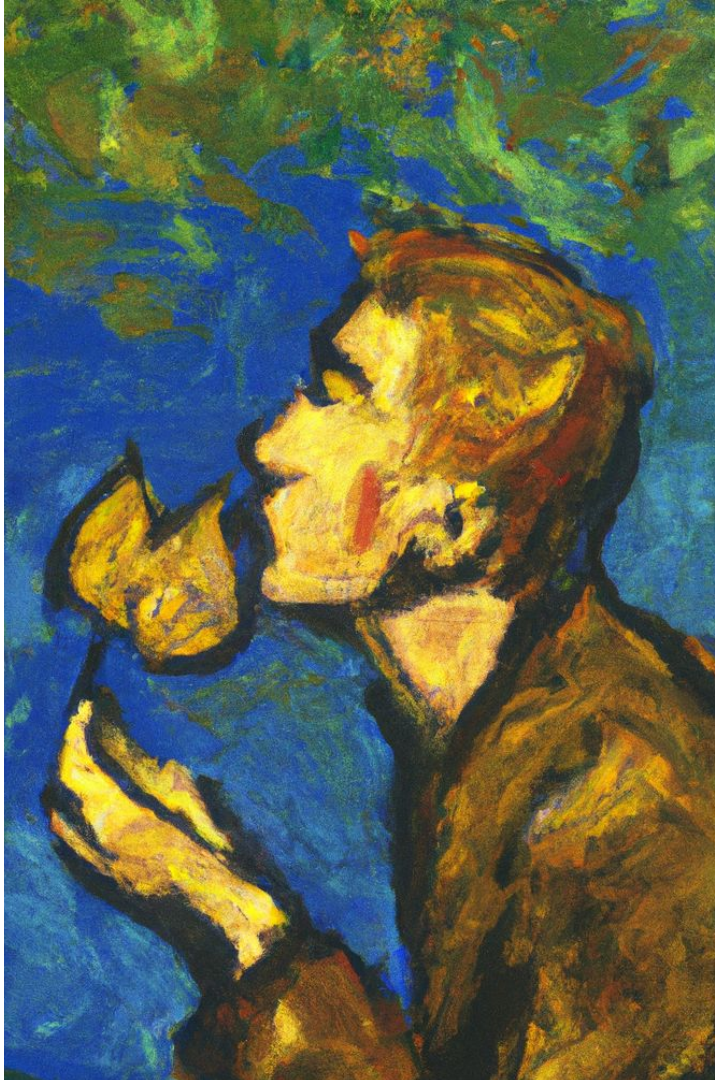
Et moi aussi.



Annual Stack Overflow developer survey (2023)



https://survey.stackoverflow.co/2023/#most-popular-technologies-database-learn




mongoDB

```
db.users.insertOne(
  {
    name: "sue",
    age: 26,
    status: "pending"
  }
)
```

collection

field: value

field: value

field: value

document


```
db.users.insertOne(          ← collection
  {
    name: "sue",             ← field: value
    age: 26,                 ← field: value
    status: "pending"        ← field: value
  }                                          } document
)
```

MongoDB University

```
db.users.insertOne(          <-- collection
  {
    name: "sue",             <-- field: value
    age: 26,                 <-- field: value
    status: "pending"        <-- field: value
  }                                  } document
)
```

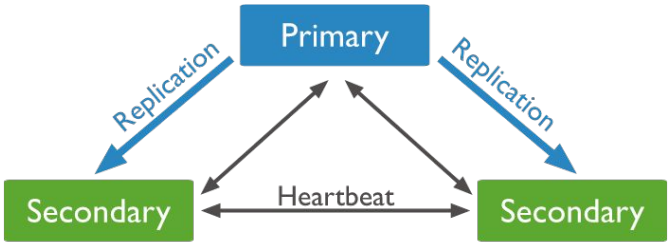

```
db.users.insertOne(          ← collection
  {
    name: "sue",             ← field: value
    age: 26,                 ← field: value
    status: "pending"        ← field: value
  }                          } document
)
```

MongoDB University

```
db.users.insertOne(          ← collection
  {
    name: "sue",             ← field: value
    age: 26,                 ← field: value
    status: "pending"        ← field: value
  }                          } document
)
```

node JS

mongoDB



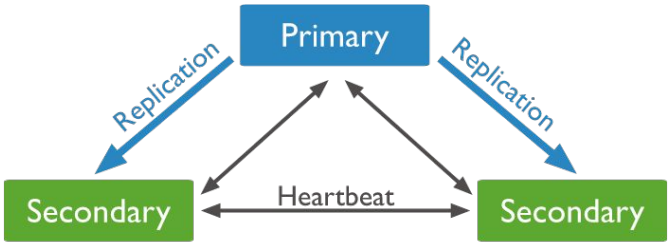
MongoDB University

```
db.users.insertOne(          <-- collection
  {
    name: "sue",             <-- field: value
    age: 26,                 <-- field: value
    status: "pending"        <-- field: value
  }                                              } document
)
```

node JS

mongoDB

mongoDB. Atlas



En 2018, MongoDB passe sous SSPL

(Server Side Public License)

En 2018, MongoDB passe sous SSPL

(Server Side Public License)

open source initiative®

https://blog.opensource.org/the-sspl-is-not-an-open-source-license/

Vendor lock-in

Vendor lock-in

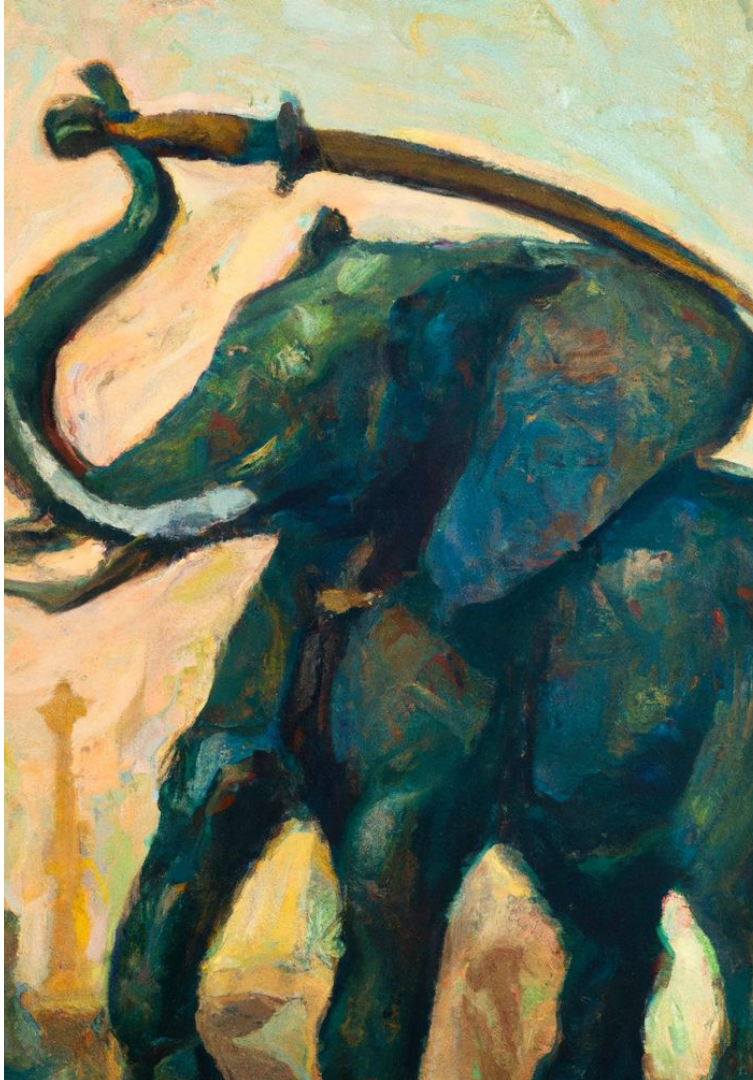
Développement dirigé par le profit

Vendor lock-in

Développement dirigé par le profit

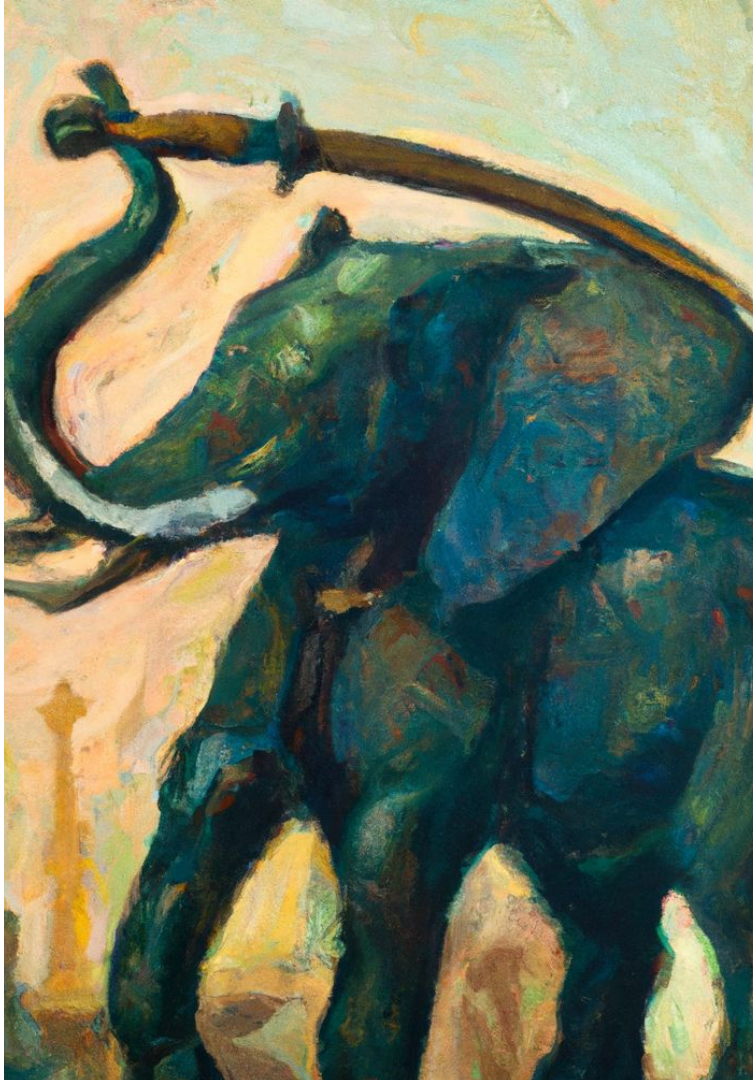
Perte d’équité

#1 : Mongo n’a pas de validation de schéma



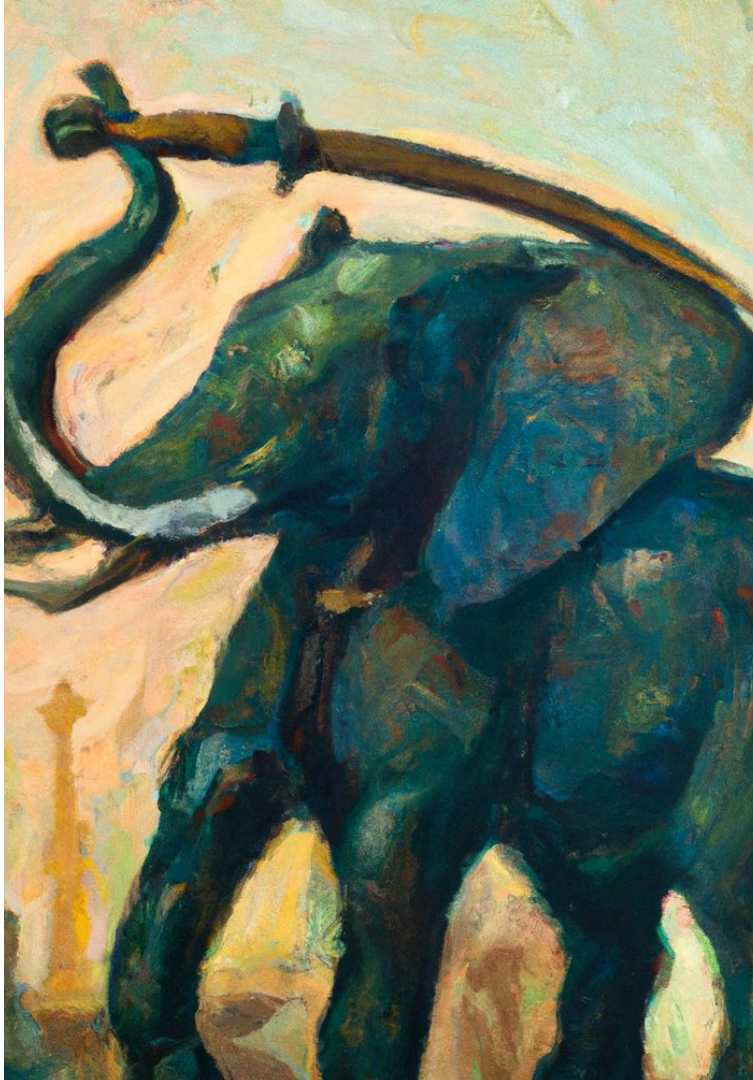
#1 : Mongo n’a pas de validation de schéma

- Cohérence des données



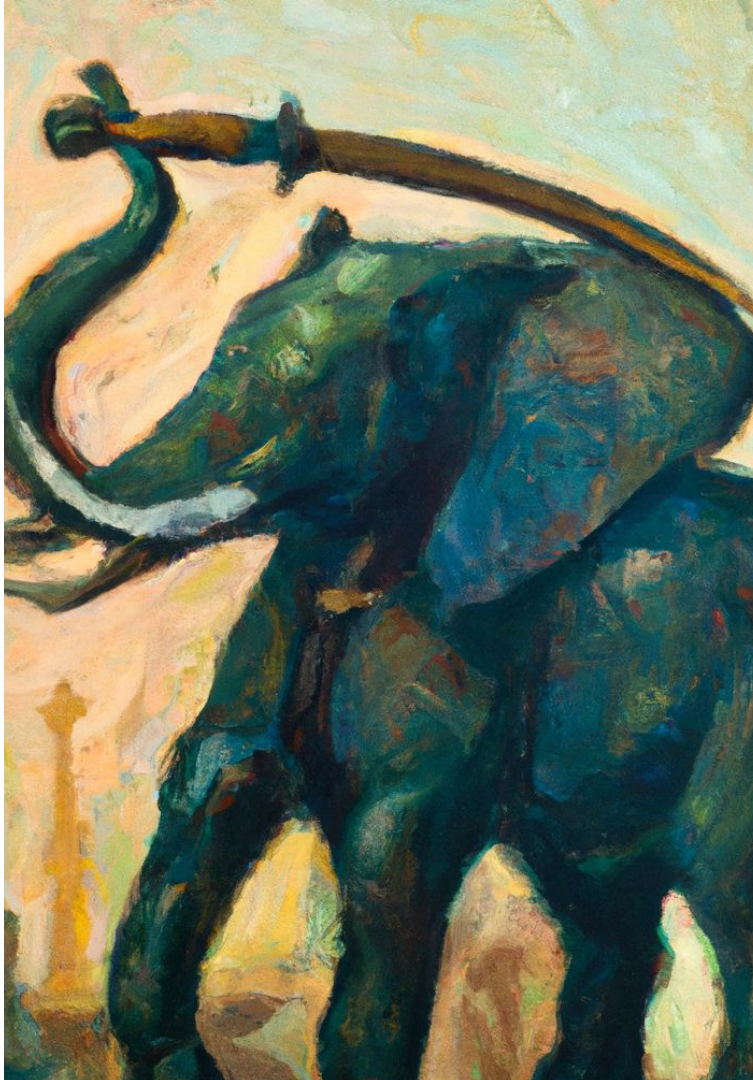
#1 : Mongo n’a pas de validation de schéma

- Cohérence des données
- Erreurs de runtime



#1 : Mongo n’a pas de validation de schéma

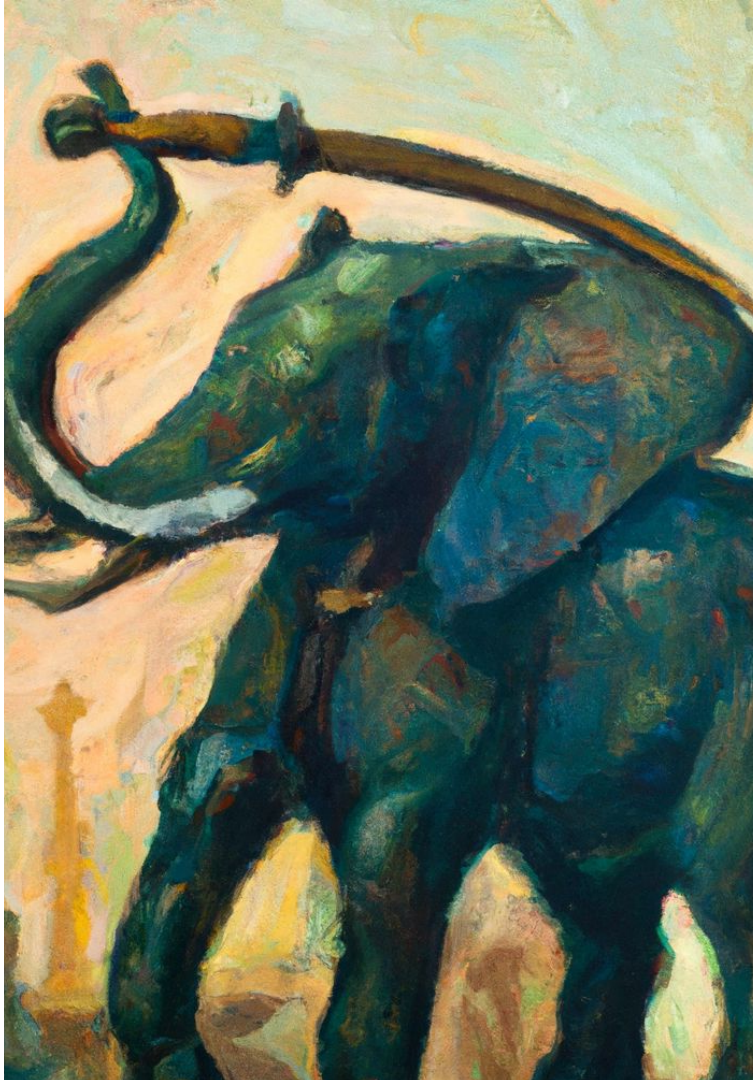
- Cohérence des données
- Erreurs de runtime
- Requêtes analytiques difficiles



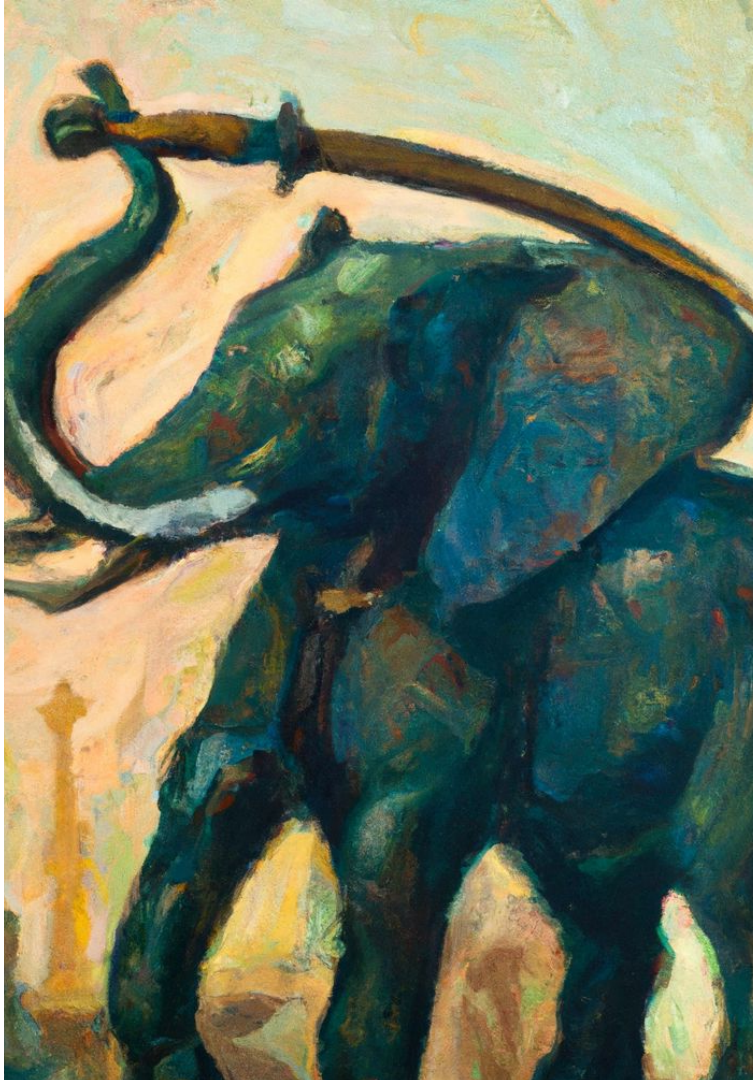
#1 : Mongo n'a pas de validation de schéma

- Cohérence des données
- Erreurs de runtime
- Requêtes analytiques difficiles

-> Depuis la 3.6 (2018), JSON Schema Validation

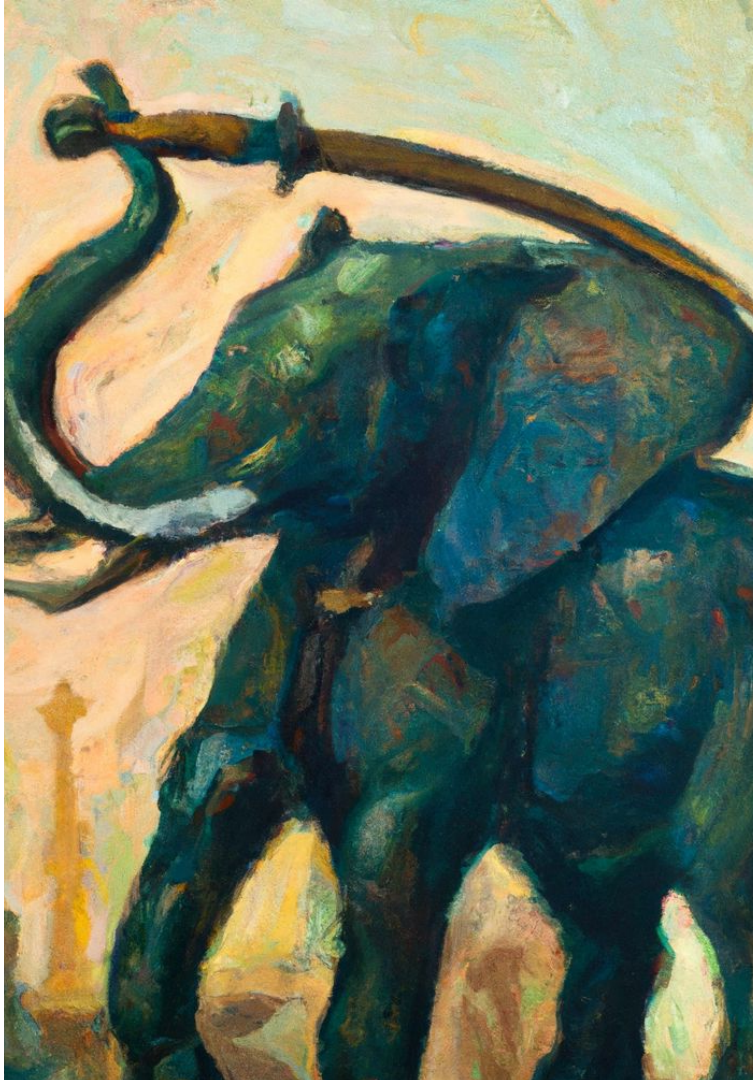


#2 : Mongo n’a pas de transactions



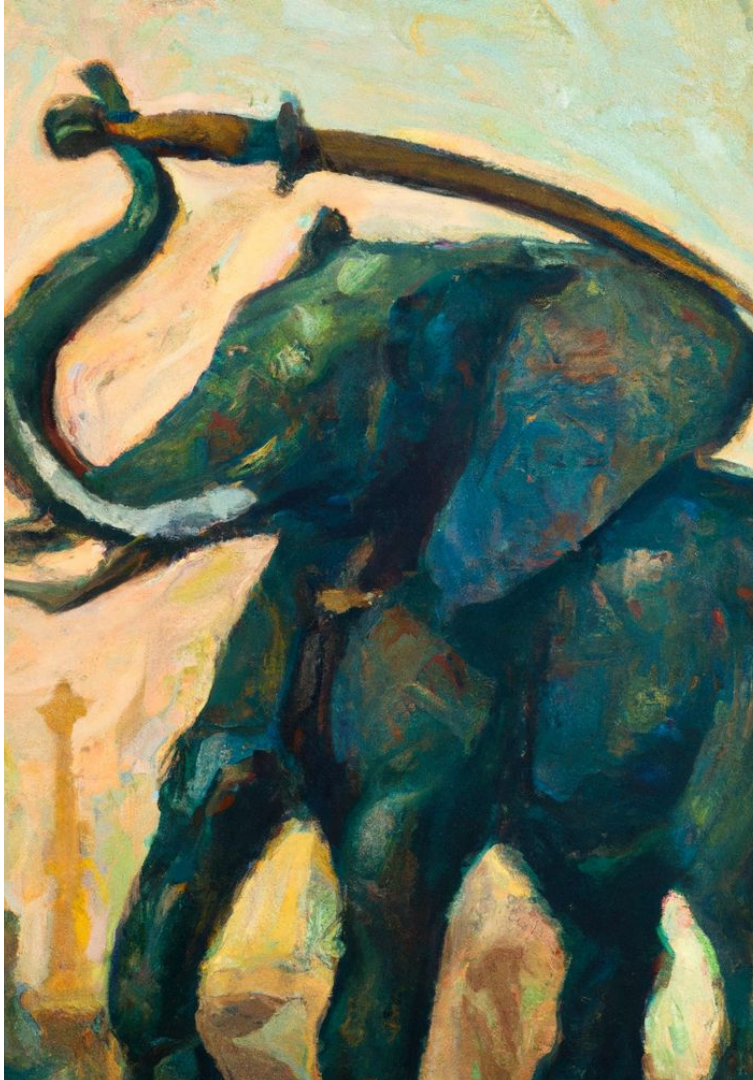
#2 : Mongo n’a pas de transactions

- Contrôles obligatoires par l’application



#2 : Mongo n’a pas de transactions

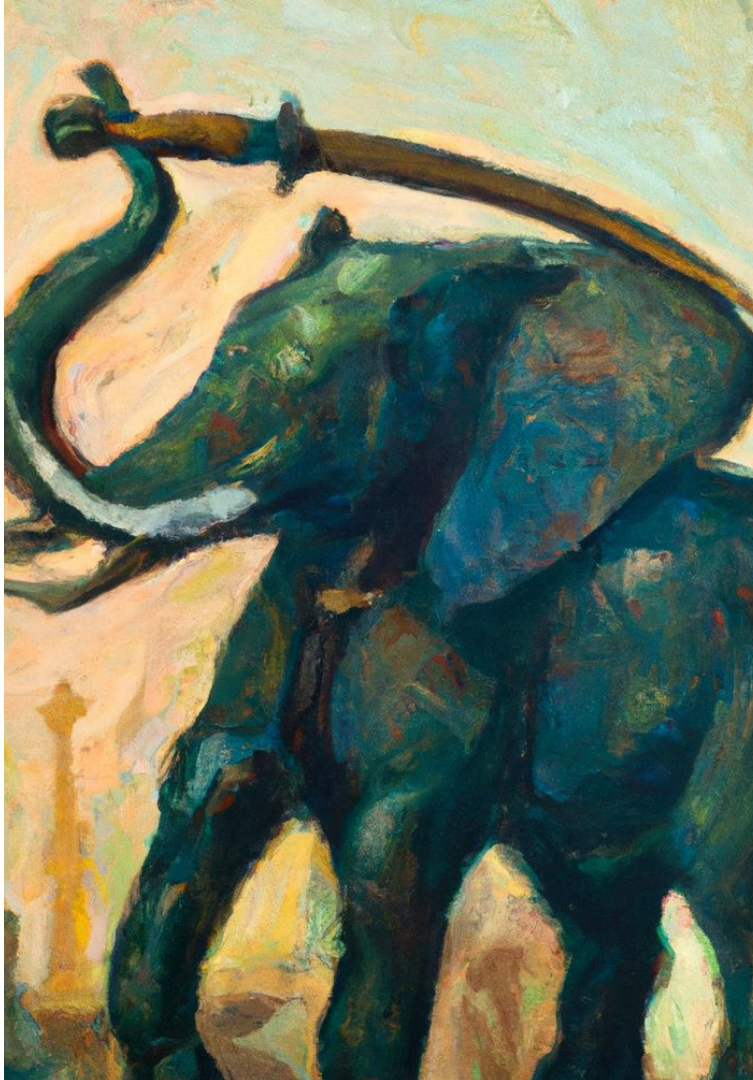
- Contrôles obligatoires par l’application
- Incohérences multi-documents



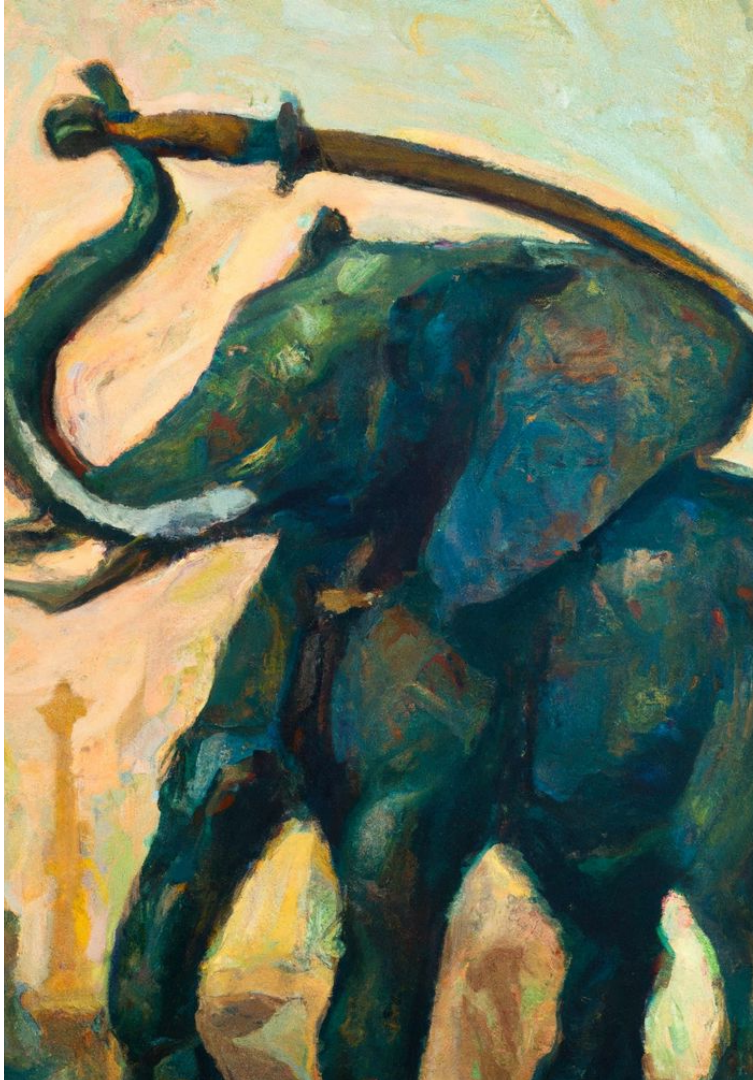
#2 : Mongo n'a pas de transactions

- Contrôles obligatoires par l'application
- Incohérences multi-documents

-> Depuis la 4.0 (2018), Multi-Document Transactions

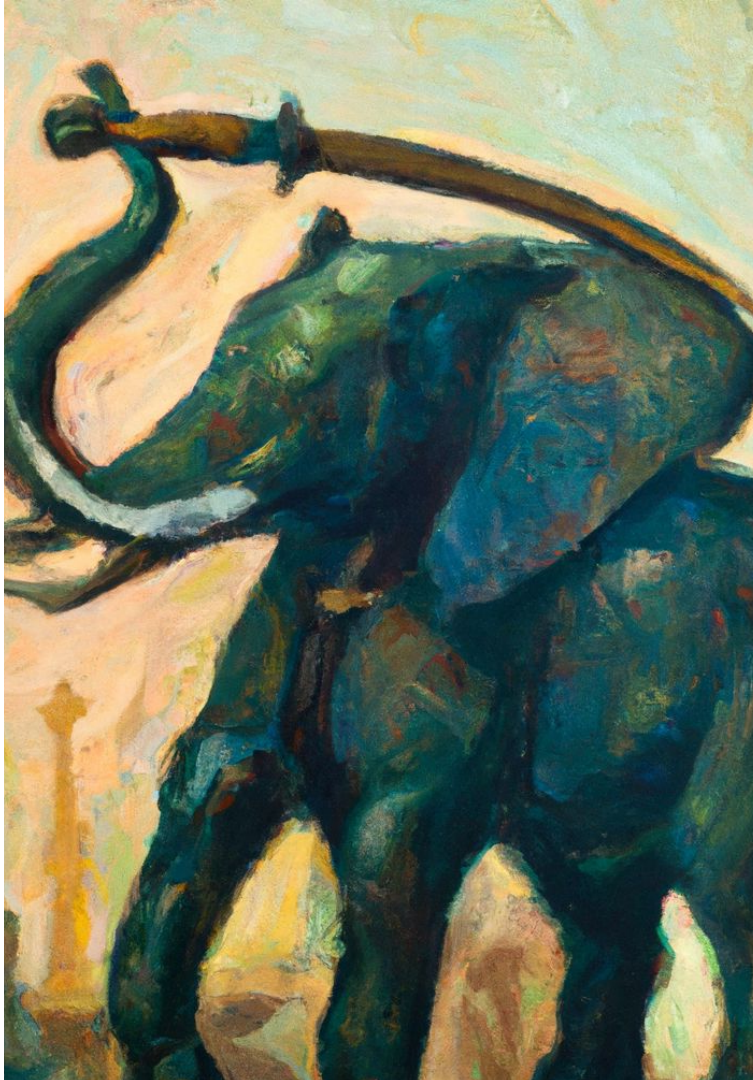


#3 : Mongo ne permet pas de Read Consistency



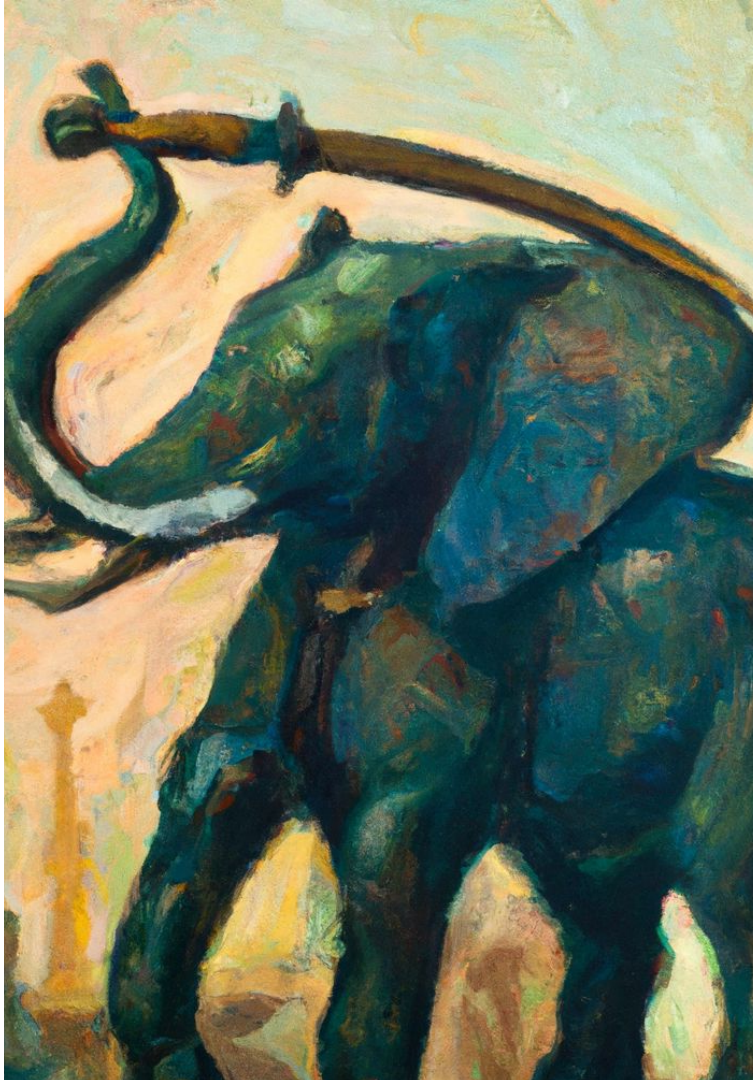
#3 : Mongo ne permet pas de Read Consistency

- Lectures inconsistentes (Eventual Consistency)



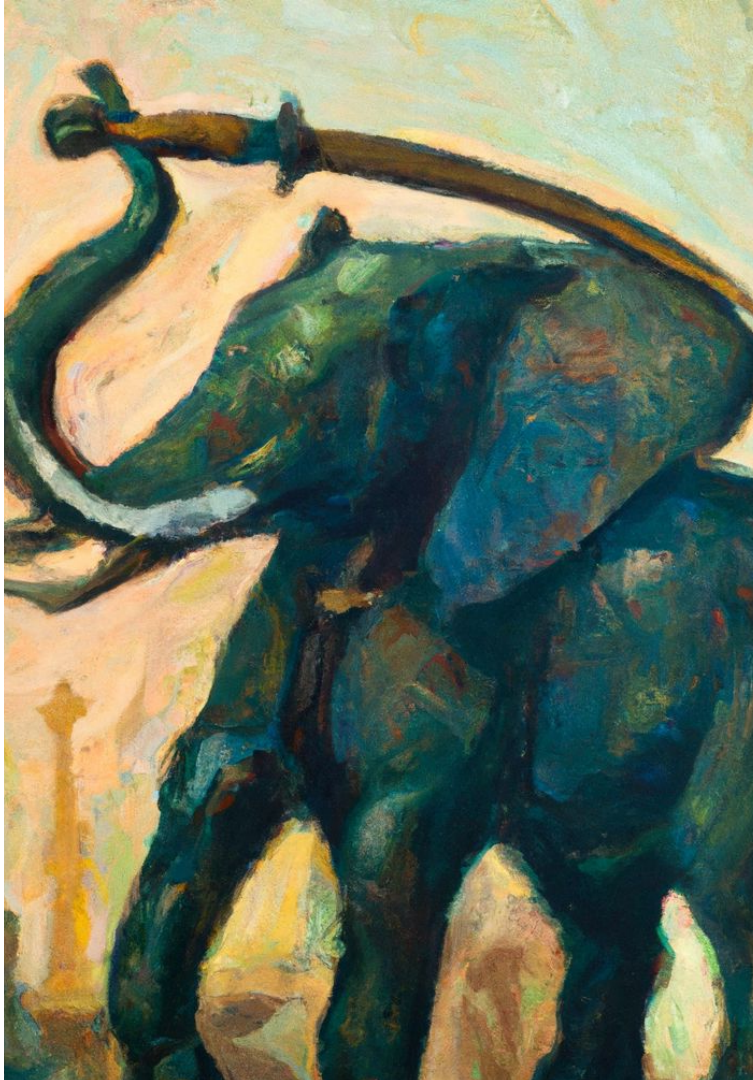
#3 : Mongo ne permet pas de Read Consistency

- Lectures inconsistentes (Eventual Consistency)
- Perte de données



#3 : Mongo ne permet pas de Read Consistency

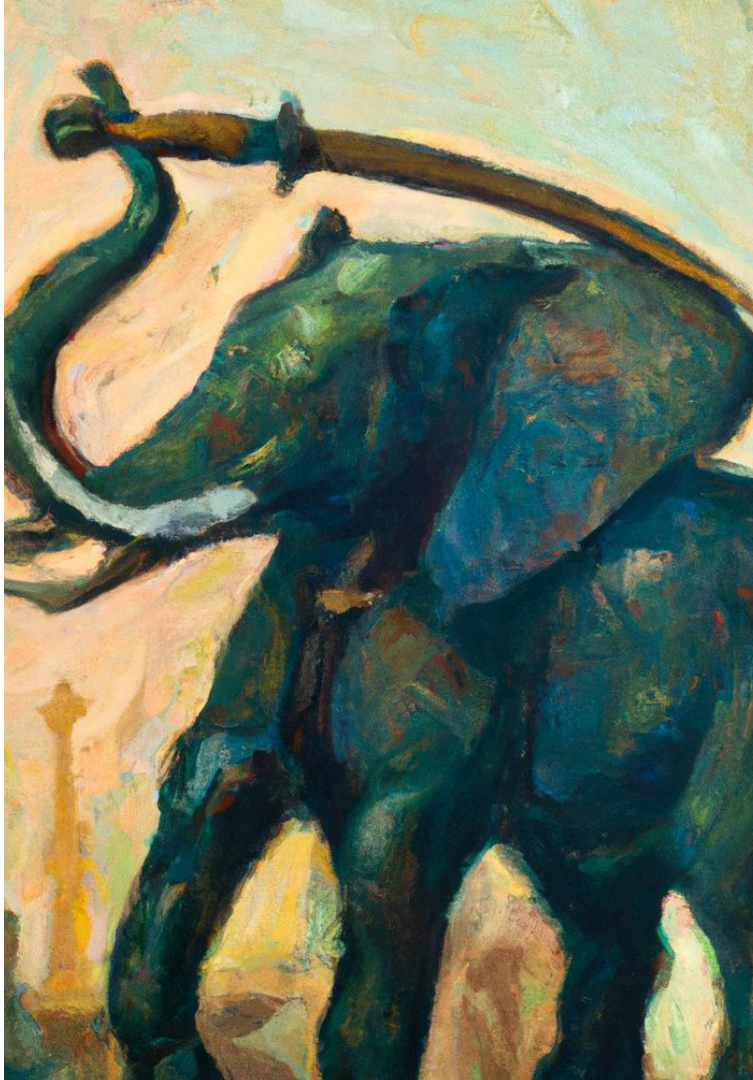
- Lectures inconsistentes (Eventual Consistency)
- Perte de données
- ~~ACID~~



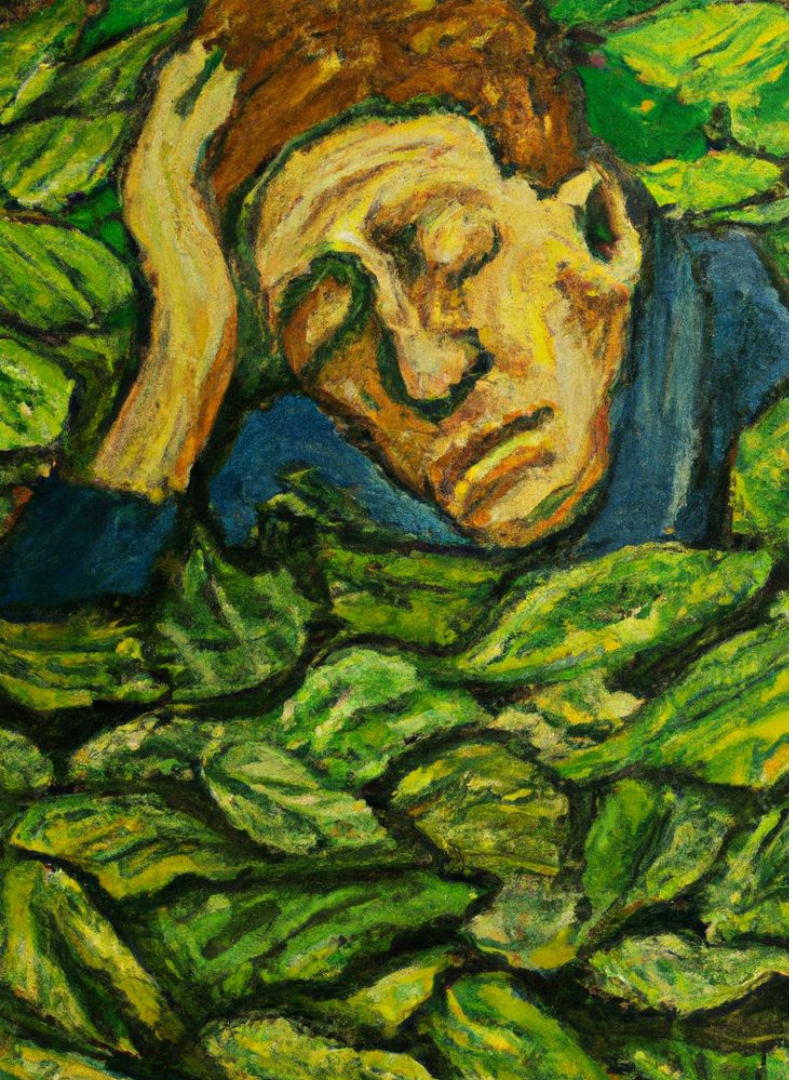
#3 : Mongo permet pas de Rea Consistency

- Lectures inconsistentes (Eventual Consistency)
- Perte de données
- ~~ACID~~

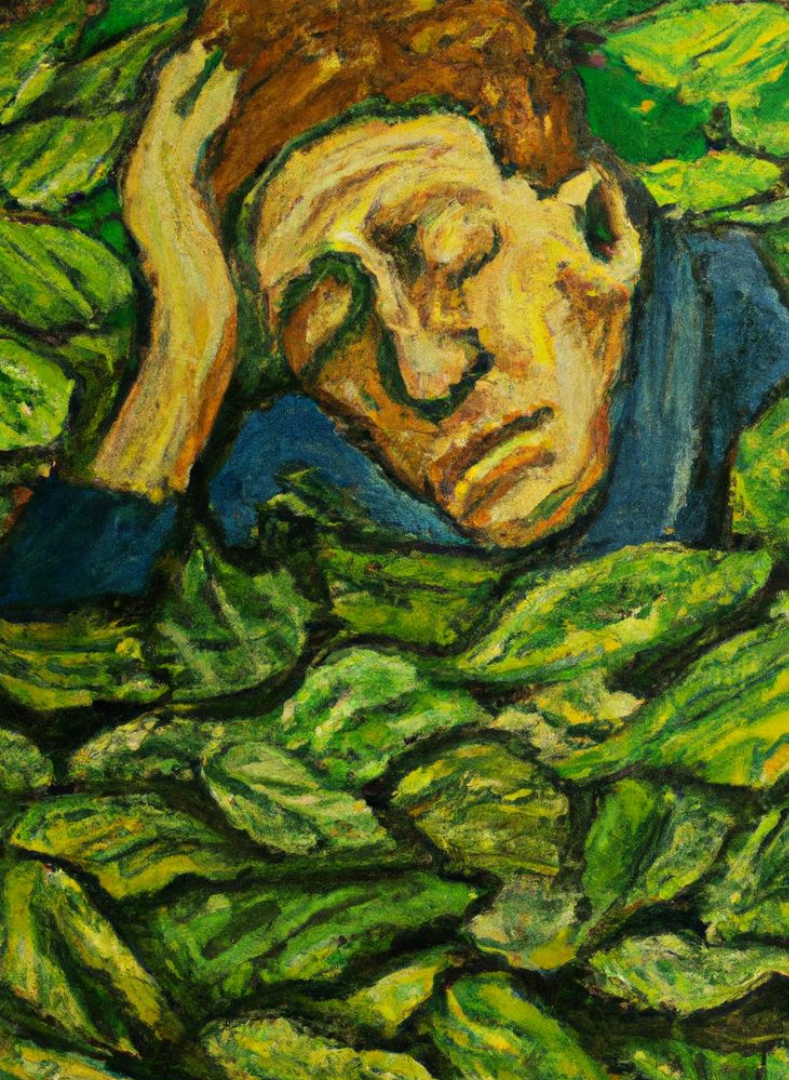
-> Write Concern + Read Concern + Snapshot Reads



Fin.

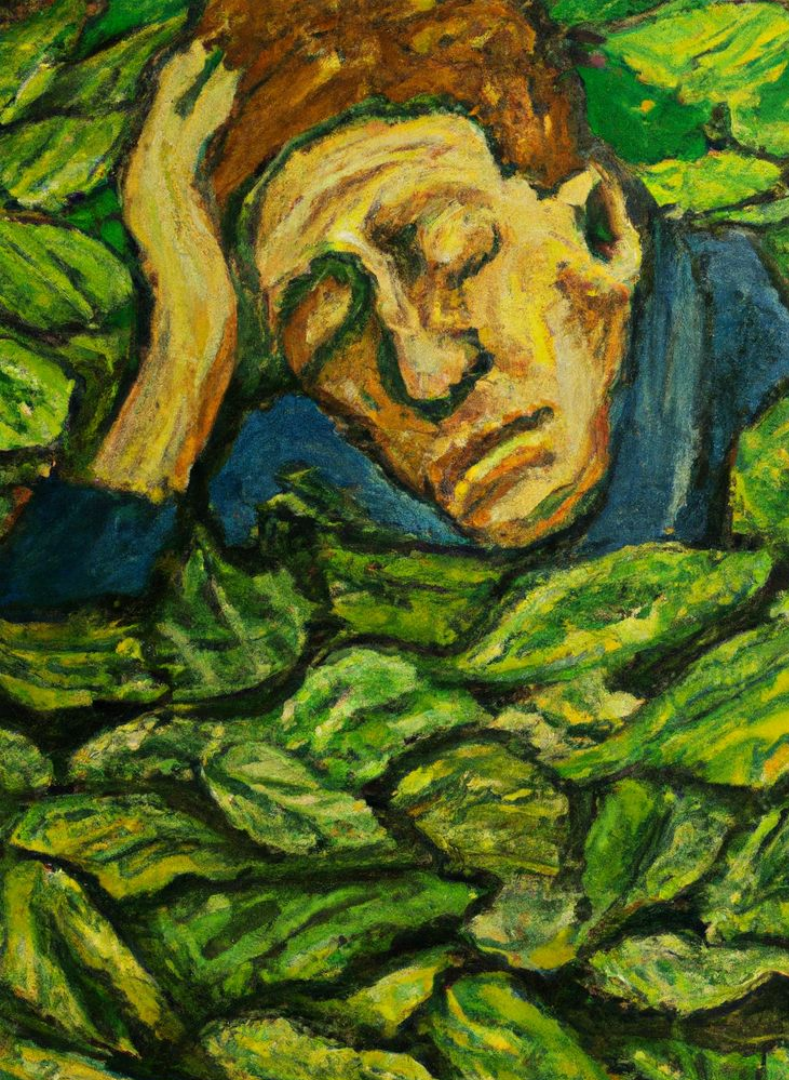


Oui, Mongo peut faire tout ça...



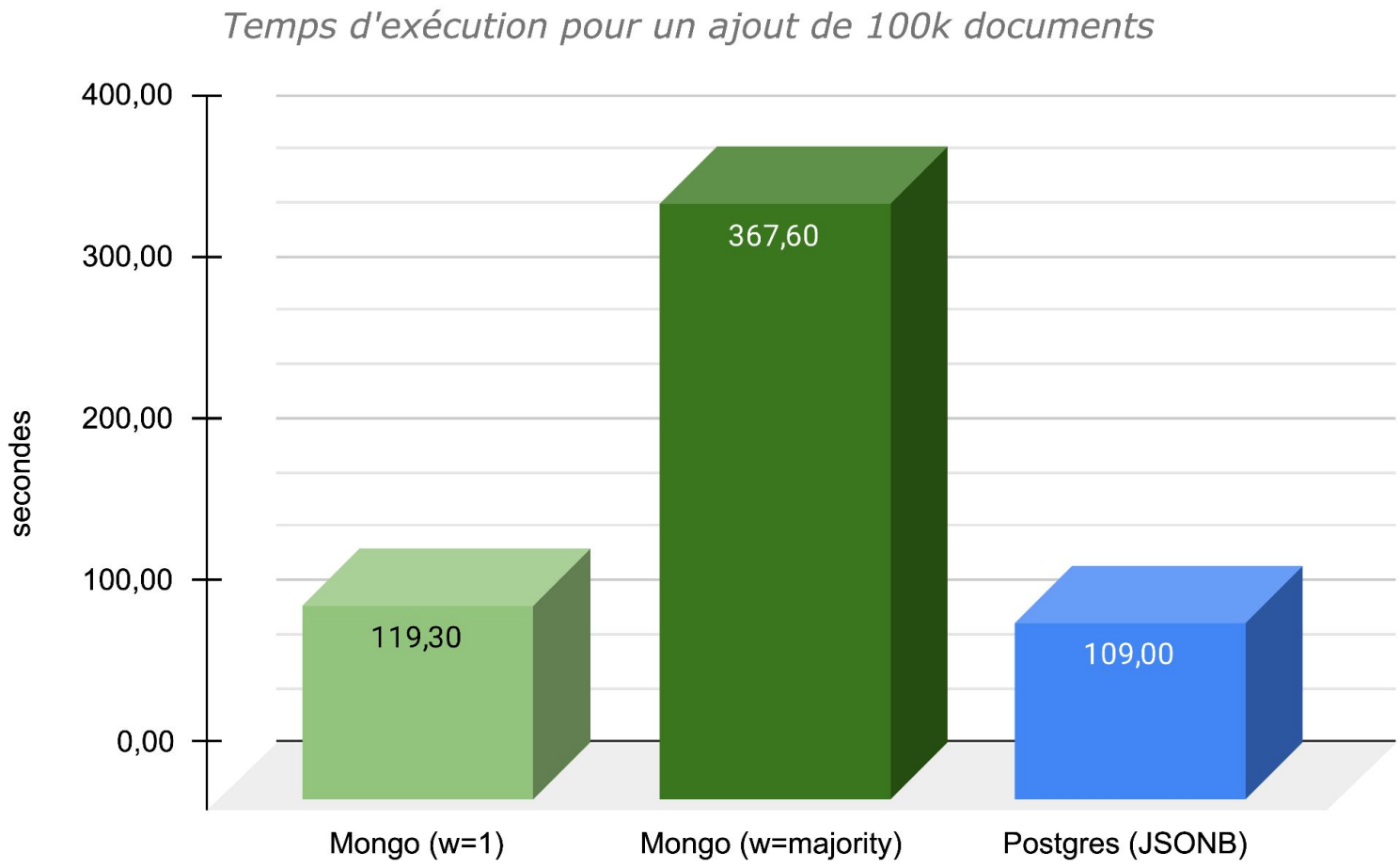
Oui, Mongo peut faire tout ça...

mais à quel prix ?



Oui, Mongo peut faire tout ça...

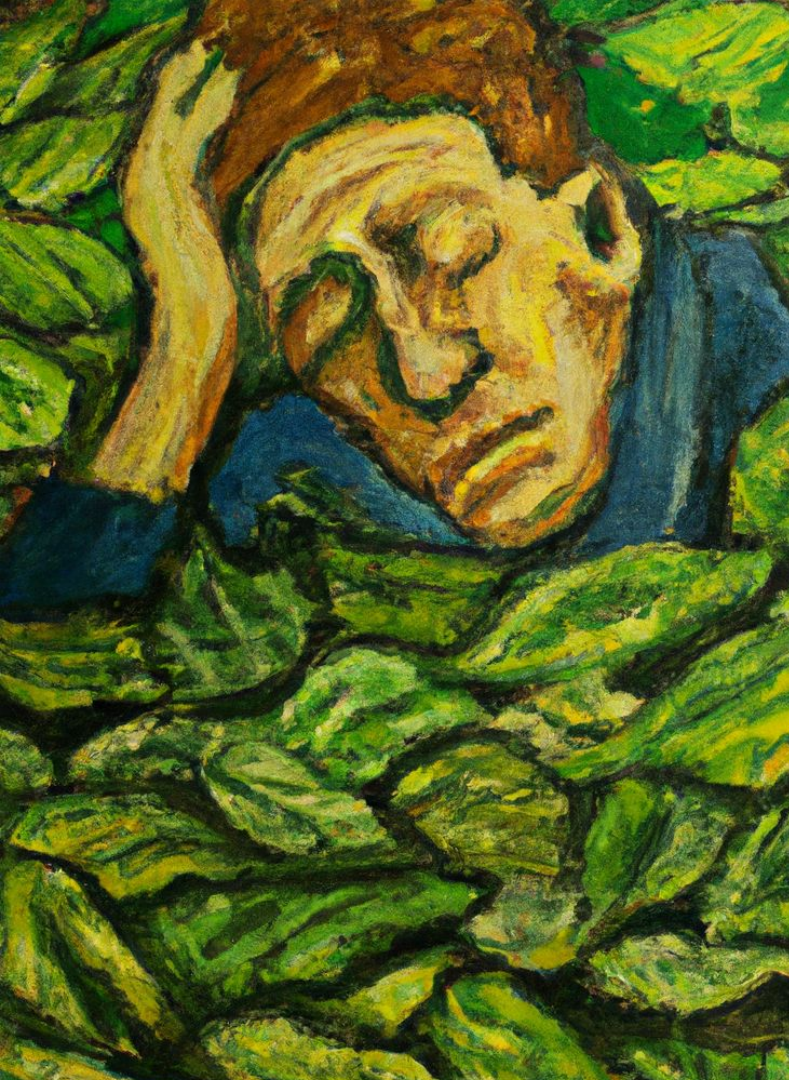
en sacrifiant les performances.


Temps d'exécution pour un ajout de 100k documents
400,00
300,00
200,00
100,00
0,00
secondes
119,30
367,60
109,00
Mongo (w=1)
Mongo (w=majority)
Postgres (JSONB)

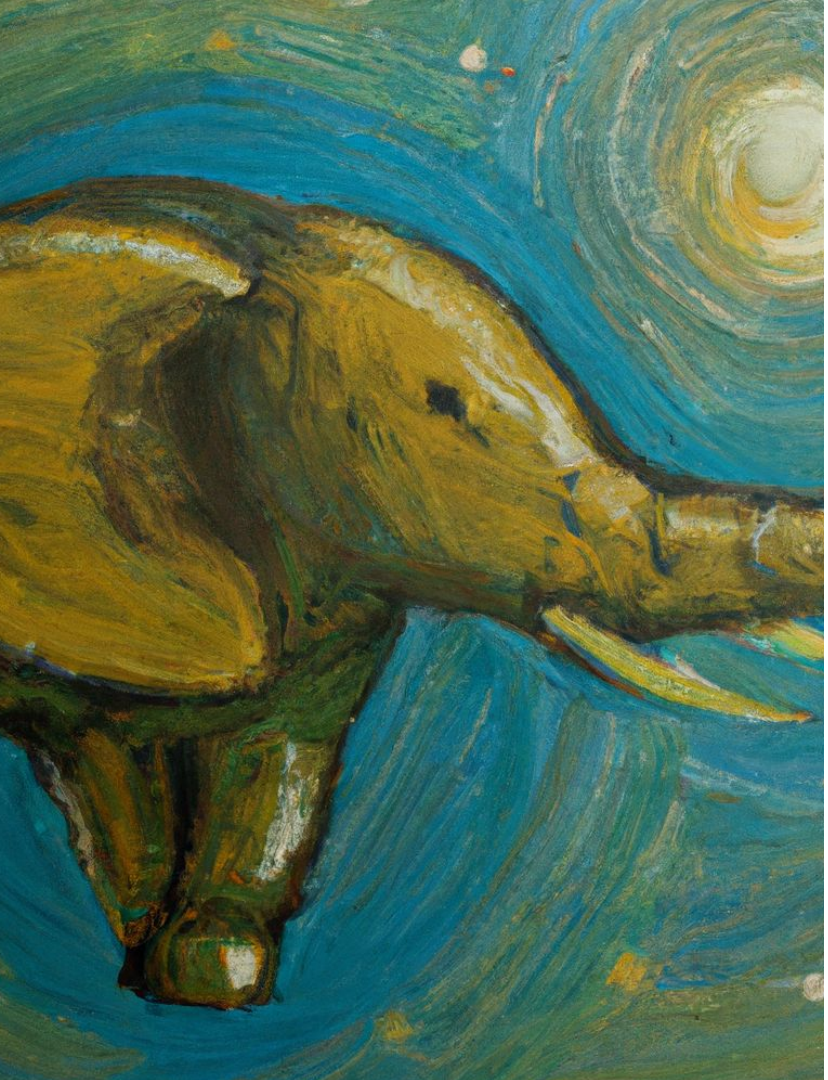
PostgreSQL

mongoDB®

	PostgreSQL	mongoDB
Open-Source	✓	✗
Schema validation	✓	✗
Transactions	✓	✗ (✓ avec de fortes dégradations des performances)
Consistance des données	✓	✗
Vendor lock-in	✓	✗
Coûts	✓	✗



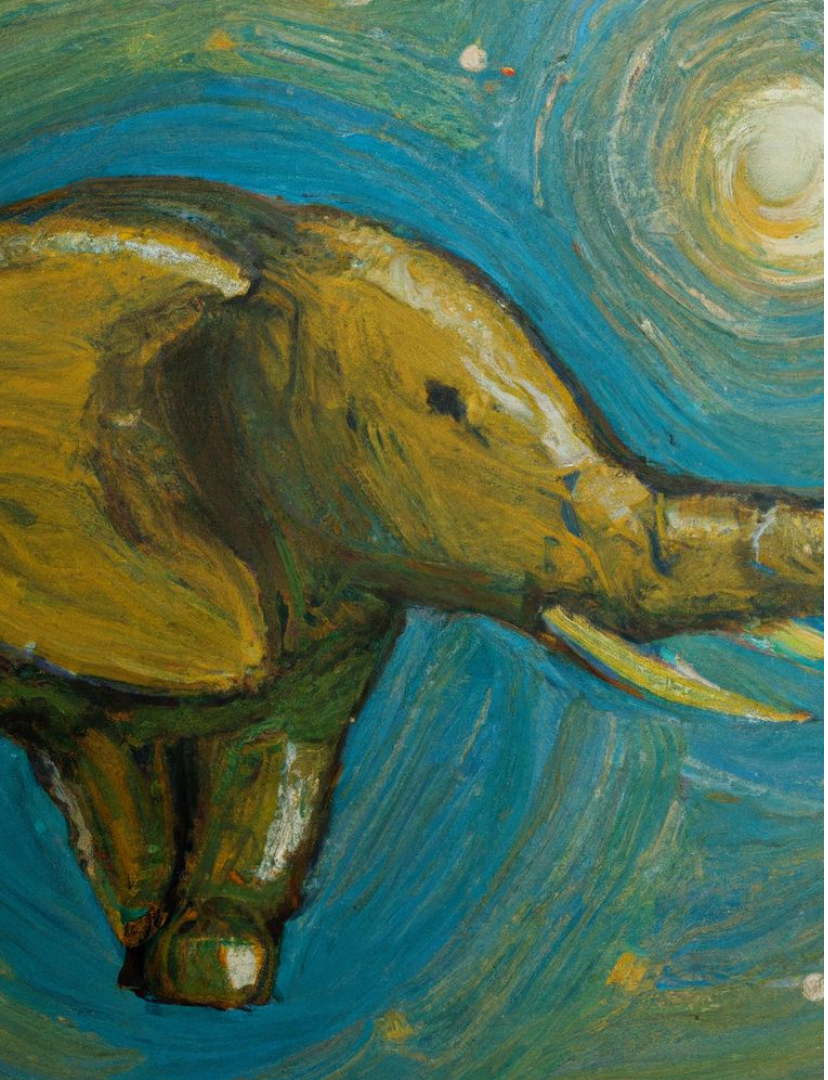
- Besoins d’intégrité de données
- Write Concern à “majority” + Read Concern à “linearizable”
- Jointures et requêtes complexes
- $lookup à profusion dans le code
- Transactions et respect de l’ACID



JSONB, œil pour œil, document pour document

- Natif et en constante amélioration
- Permet des usages hybrides
- Migration (presque) facile

https://ftisiot.net/postgresqljson/main/



Quelques défauts encore

- Un langage de requêtage difficile à lire et écrire

```
select
    jsonb_set(
        json_data -> 'pizzas',
        '{0,"pizzaName"}',
        to_jsonb('4 Stagioni'::text),
        false)
        as change_first_pizza_name
from test;
```

- TOAST peut être un soucis

Foreign-data wrapper pour MongoDB

- Similaire aux FDW comme Oracle ou MySQL
- Lecture / Ecriture
- Open-Source (GNU GPLv3)

https://github.com/EnterpriseDB/mongo_fdw

```
-- load extension first time after install
CREATE EXTENSION mongo_fdw;

-- create server object
CREATE SERVER mongo_server
        FOREIGN DATA WRAPPER mongo_fdw
        OPTIONS (address '127.0.0.1', port '27017');

-- create user mapping
CREATE USER MAPPING FOR postgres
        SERVER mongo_server
        OPTIONS (username 'mongo_user', password 'mongo_pass');
```

```
-- create foreign table
CREATE FOREIGN TABLE warehouse
        (
                _id name,
                warehouse_id int,
                warehouse_name text,
                warehouse_created timestamptz
        )
        SERVER mongo_server
        OPTIONS (database 'db', collection 'warehouse');

-- Note: first column of the table must be "_id" of type "name".

-- select from table
SELECT * FROM warehouse WHERE warehouse_id = 1;
           _id            | warehouse_id | warehouse_name |     warehouse_created
--------------------------+--------------+----------------+---------------------------
 53720b1904864dc1f5a571a0 |            1 | UPS            | 2014-12-12 12:42:10+05:30
(1 row)
```

Quelques contraintes:

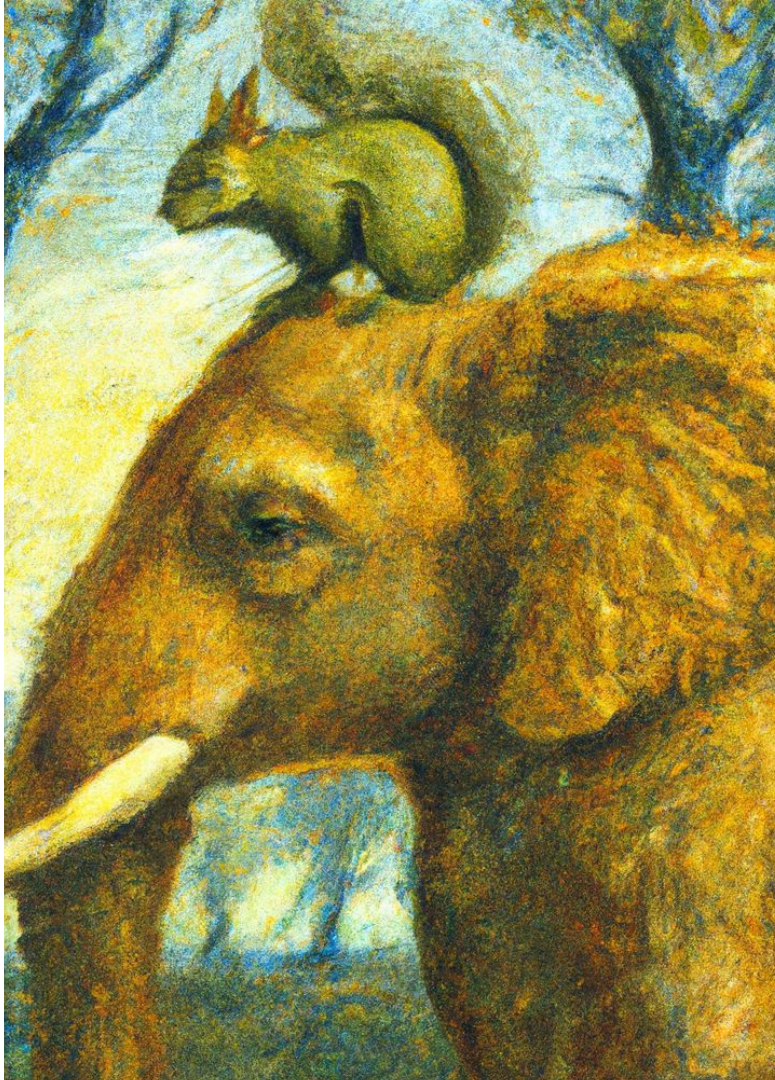
- Nécessaire de recréer le schéma manuellement
- Pas de migration au fil de l’eau
- Indisponible sur la majorité des DBaaS

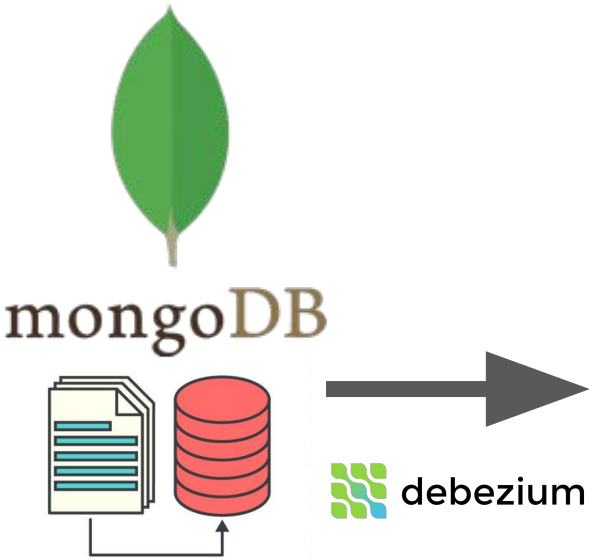
Apache Flink : un ETL en temps réel

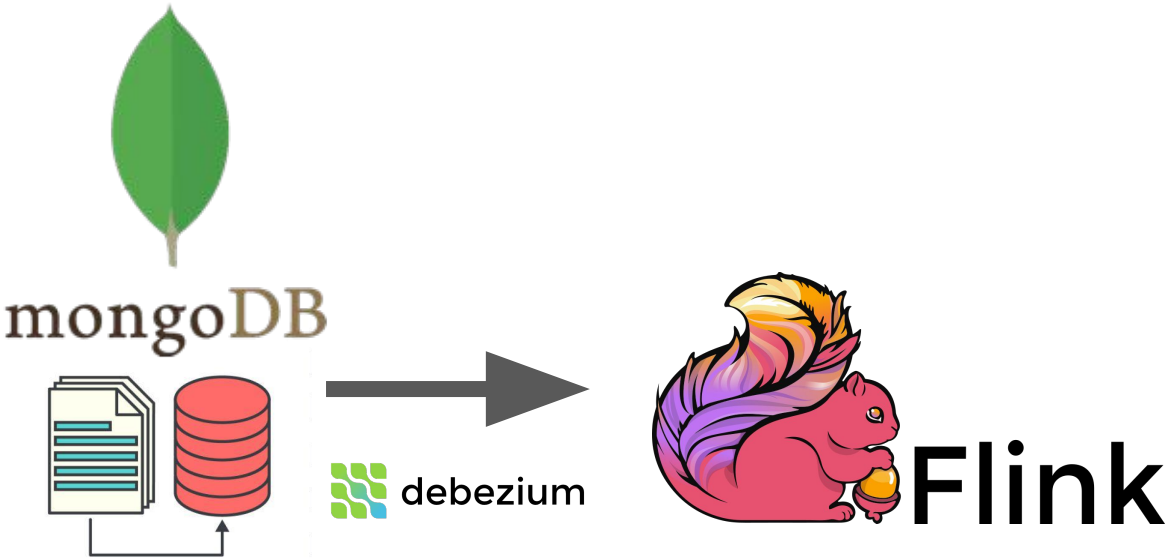
- Transformation de données à la volée
- Flink SQL
- Open-Source (Apache 2.0)

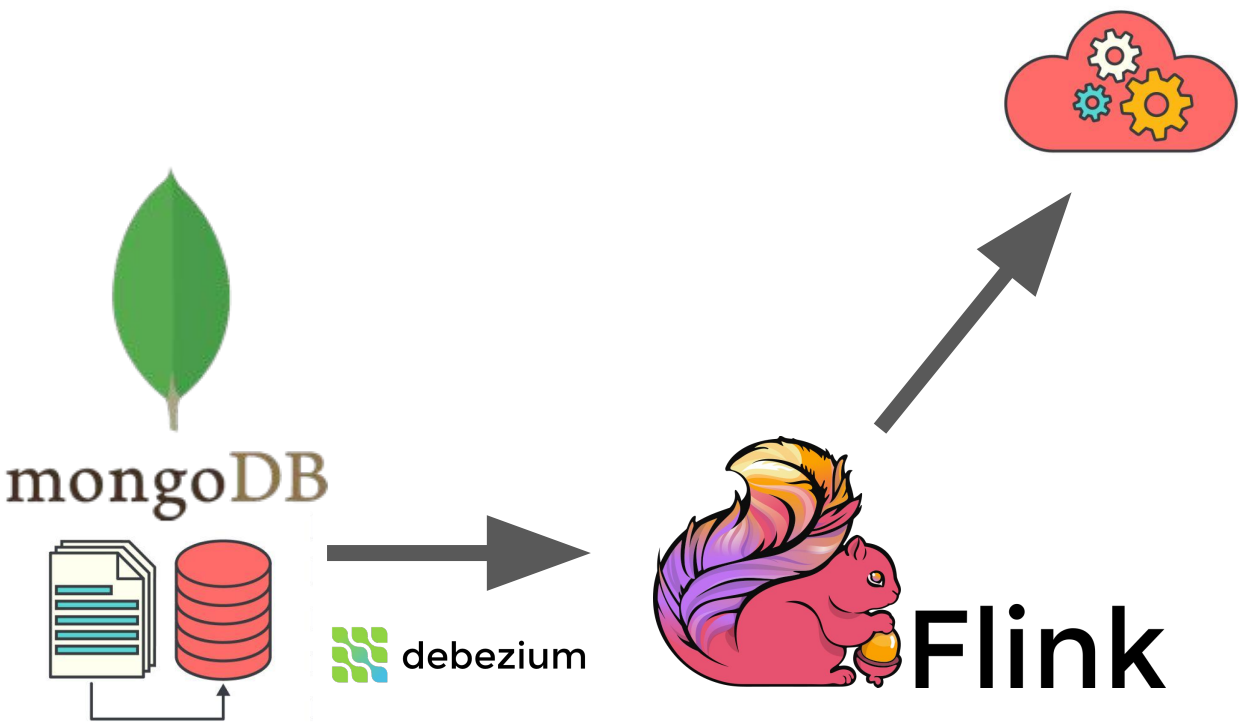
Flink

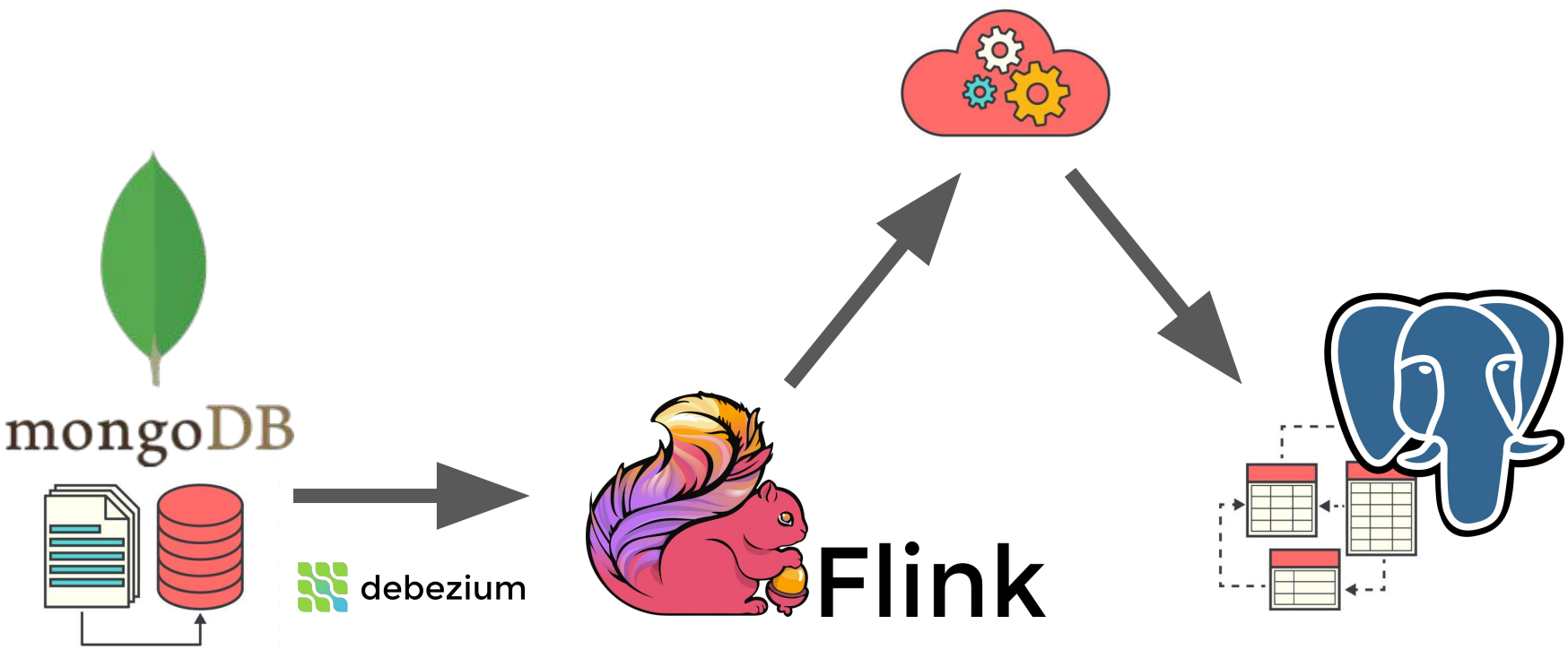
https://flink.apache.org/

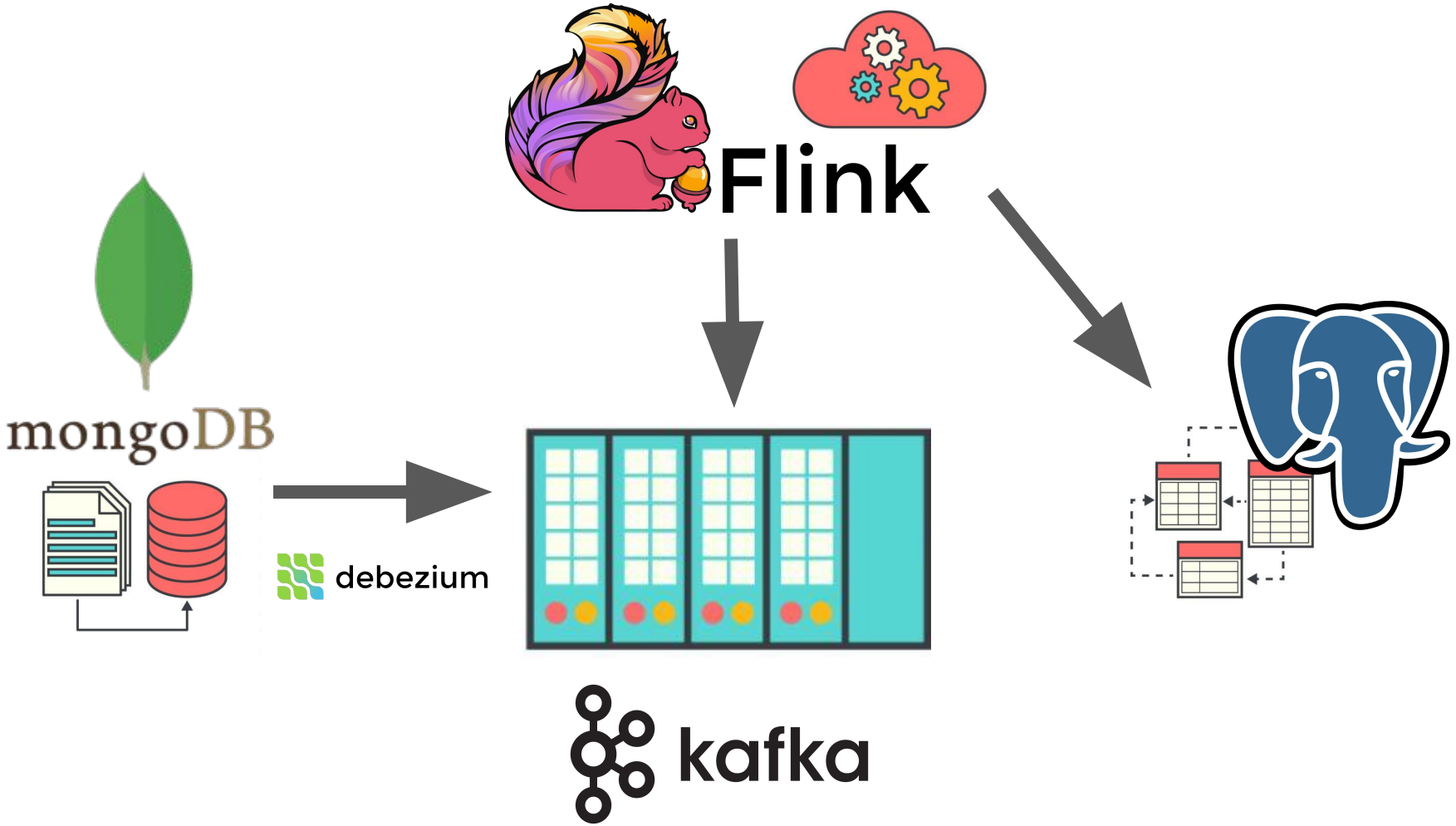



mongoDB
debezium


mongoDB
debezium
Flink


mongoDB
debezium
Flink


mongoDB
debezium
Flink


Flink
mongoDB
debezium
kafka

kafka-arkea
Aiven for Apache Kafka

```
CREATE TABLE mongo_nobels (
  fullDocument ROW(`firstname` STRING, `surname` STRING),
  operationType STRING
) WITH (
  'connector' = 'kafka',
  'properties.bootstrap.servers' = '',
  'scan.startup.mode' = 'earliest-offset',
  'topic' = 'ferretdb.nobels',
  'value.format' = 'json'
)
```

kafka-arkea
Aiven for Apache Kafka

```
CREATE TABLE mongo_nobels (
  fullDocument ROW(`firstname` STRING, `surname` STRING),
  operationType STRING
) WITH (
  'connector' = 'kafka',
  'properties.bootstrap.servers' = '',
  'scan.startup.mode' = 'earliest-offset',
  'topic' = 'ferretdb.nobels',
  'value.format' = 'json'
)
```

pg-master
Aiven for PostgreSQL

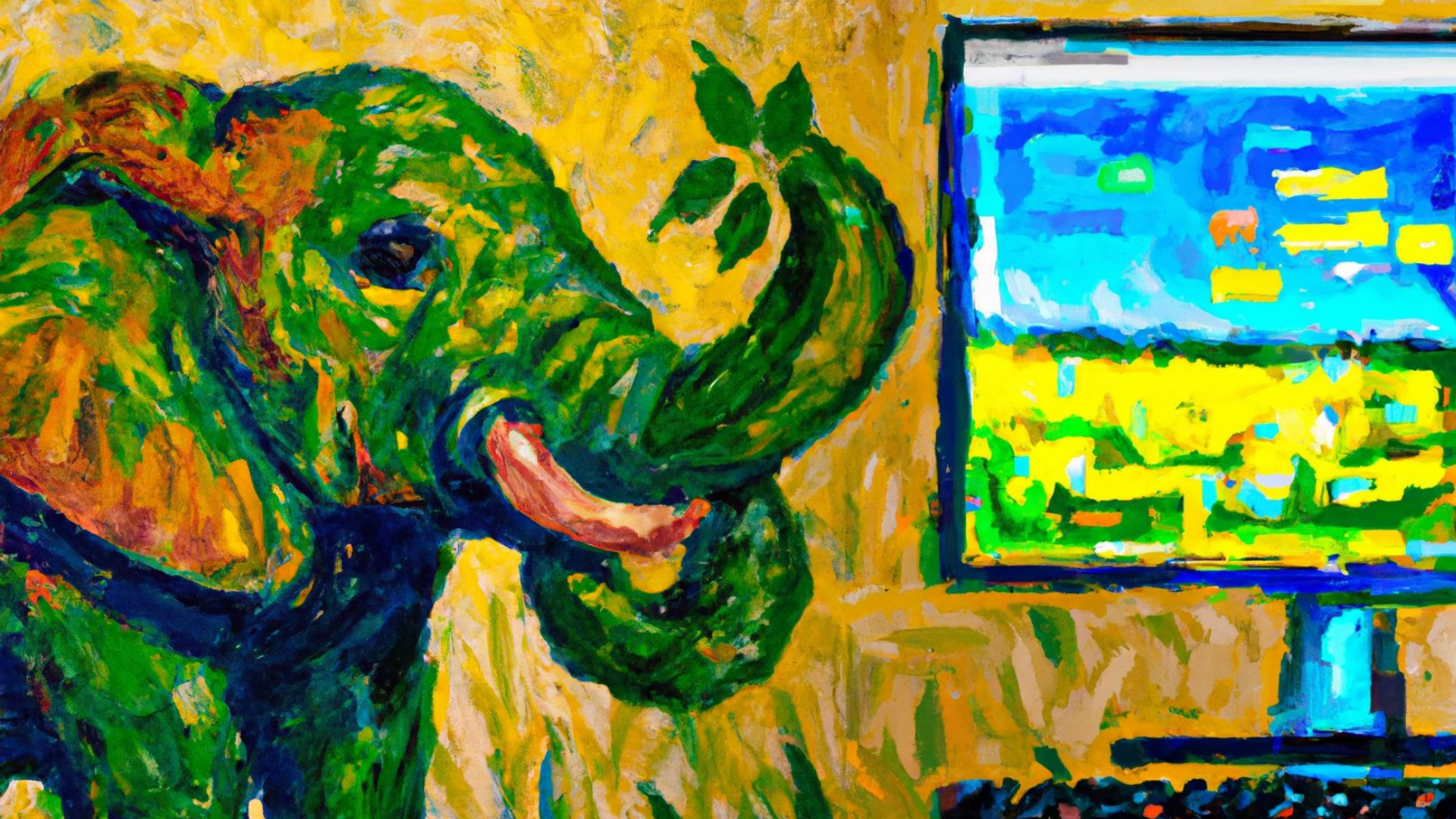
```
CREATE TABLE pg_nobels (
    firstname VARCHAR,
    lastname VARCHAR
) WITH (
    'connector' = 'jdbc',
    'url' = 'jdbc:postgresql://',
    'table-name' = 'nobels'
)
```

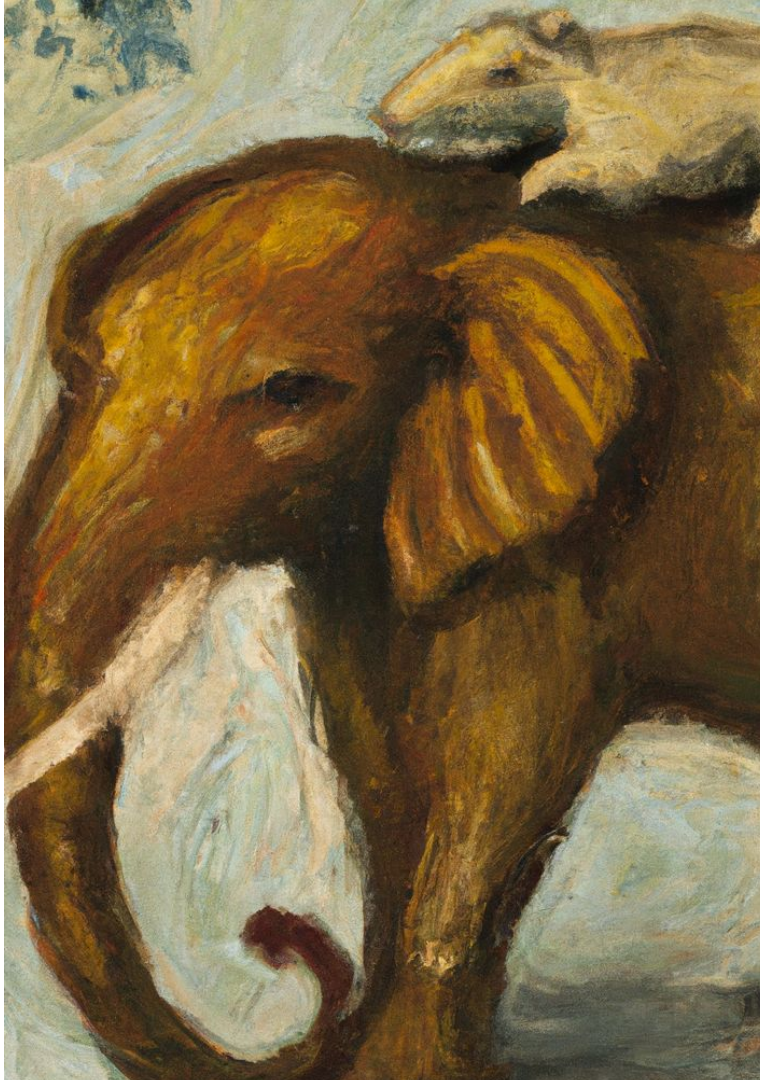
kafka-arkea
Aiven for Apache Kafka

```
CREATE TABLE mongo_nobels (
  fullDocument ROW(`firstname` STRING, `surname` STRING),
  operationType STRING
) WITH (
  'connector' = 'kafka',
  'properties.bootstrap.servers' = '',
  'scan.startup.mode' = 'earliest-offset',
  'topic' = 'ferretdb.nobels',
  'value.format' = 'json'
)
```

pg-master
Aiven for PostgreSQL

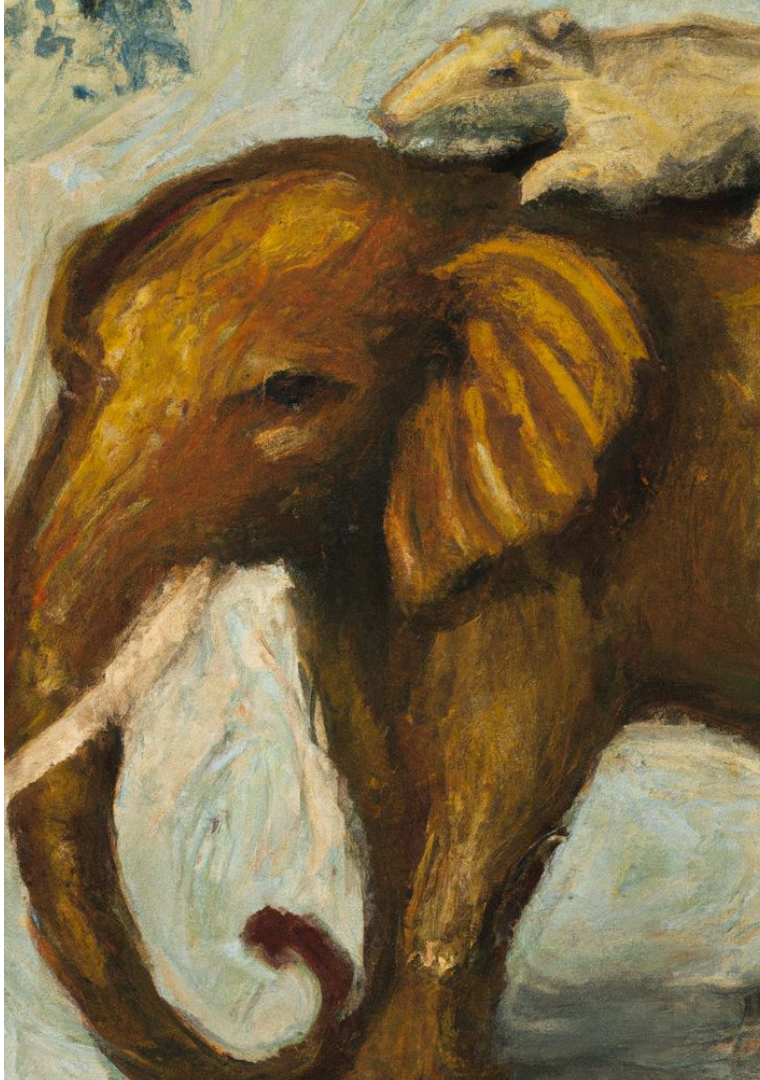
```
CREATE TABLE pg_nobels (
    firstname VARCHAR,
    lastname VARCHAR
) WITH (
    'connector' = 'jdbc',
    'url' = 'jdbc:postgresql://',
    'table-name' = 'nobels'
)
```





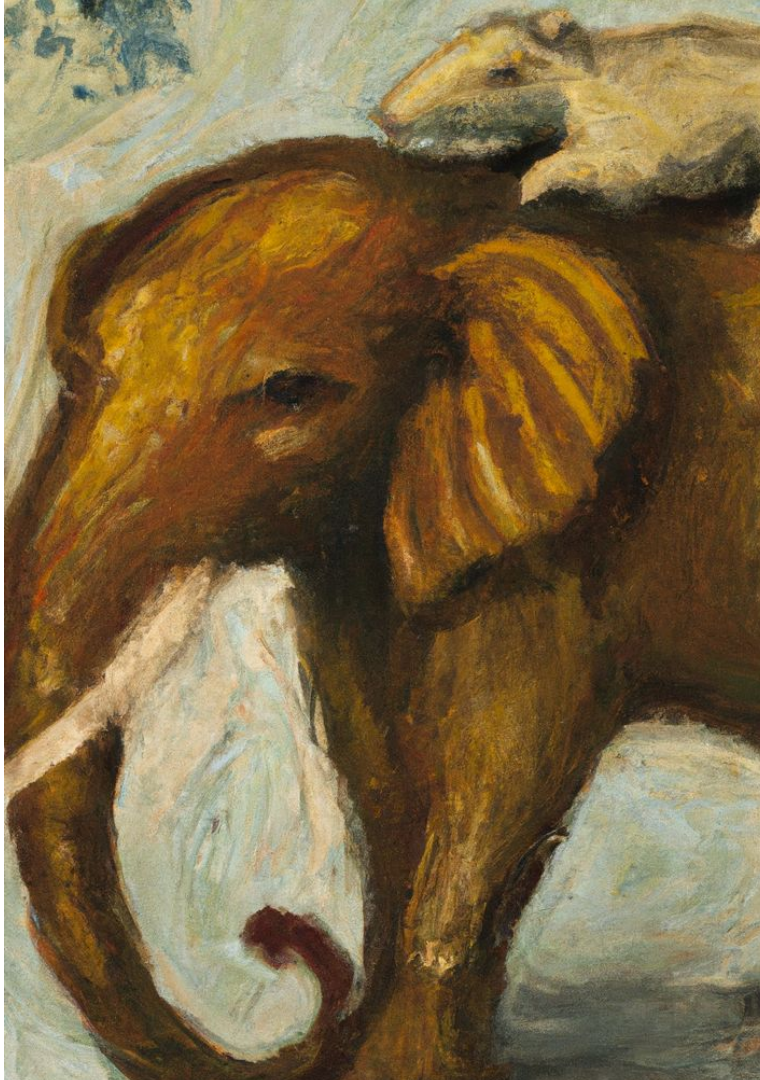
FerretDB : une vraie alternative Open Source à MongoDB

FerretDB



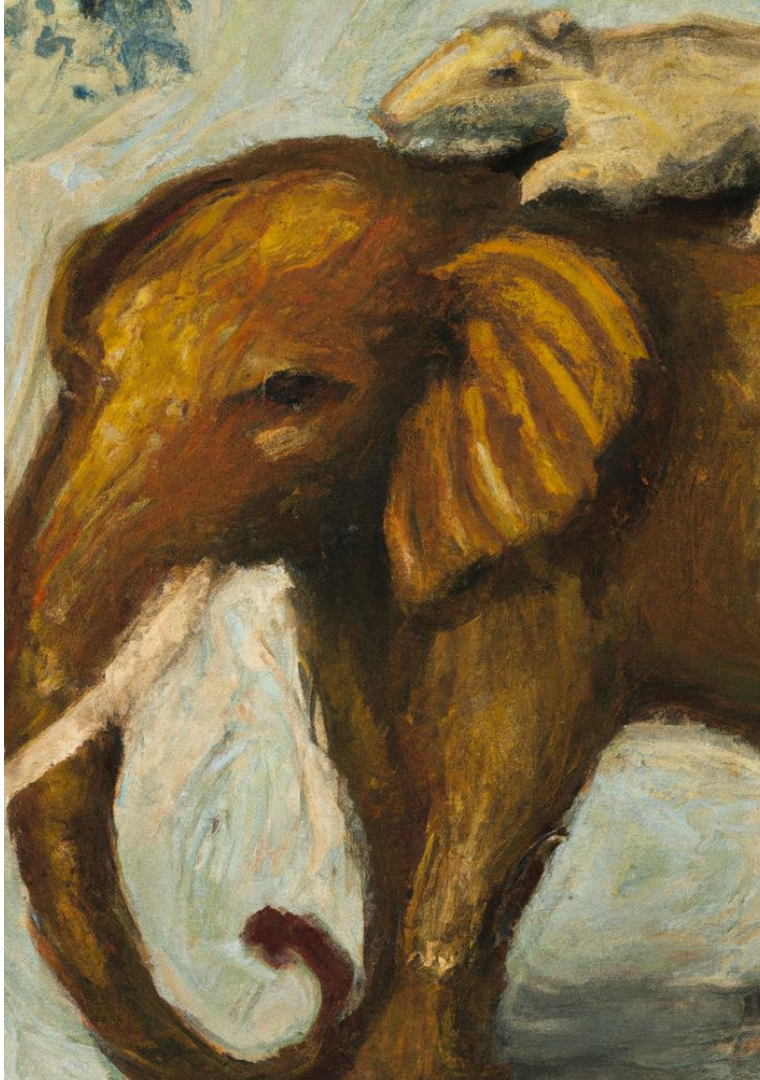
FerretDB : une vraie alternative Open Source à MongoDB

- Un proxy pour Postgres
- Ce n’est pas un énième moteur
- Communiquer avec votre base Postgres comme un vrai Mongo



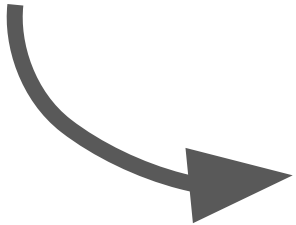
Pourquoi FerretDB ?

- Les bons côtés de Mongo avec la puissance de Postgres
- Conversion facilité des développeurs et applications
- Open source MQL



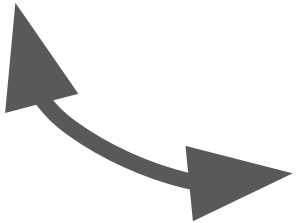
```
ferretdb> db.nobels.find(
  {
    "firstname": "Pierre",
    "surname": "Curie"
  }
);
```

```
ferretdb> db.nobels.find(
  {
    "firstname": "Pierre",
    "surname": "Curie"
  }
);
```

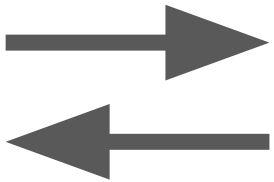


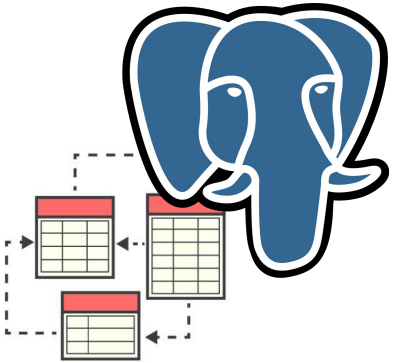
FerretDB

```
ferretdb> db.nobels.find(
  {
    "firstname": "Pierre",
    "surname": "Curie"
  }
);
```



FerretDB





JSONB

```
defaultdb=> \d ferretdb.nobels_7f990856
        Table "ferretdb.nobels_7f990856"
 Column | Type  | Collation | Nullable | Default
--------+-------+-----------+----------+---------
 _jsonb | jsonb |           |          |
Indexes:
    "nobels__id__76c2b24a_idx" UNIQUE, btree ((_jsonb -> '_id'::text))
    "nobels_surname___firstname___8aba495a_idx" btree ((_jsonb -> 'surname'::text)
```

```
defaultdb=> select * from ferretdb.nobels_7f990856 limit 1;
-[ RECORD 1 ]--------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
-------------------------------------------------------------------------------------------------------
_jsonb | {"$s": {"p": {"id": {"t": "string"}, "_id": {"t": "objectId"}, "born": {"t": "string"}, "
 {"p": {"year": {"t": "string"}, "share": {"t": "string"}, "category": {"t": "string"}, "motivatio
tring"}, "name": {"t": "string"}, "country": {"t": "string"}}, "$k": ["name", "city", "country"]}}
"t": "array"}, "surname": {"t": "string"}, "bornCity": {"t": "string"}, "diedCity": {"t": "string"
ring"}, "bornCountryCode": {"t": "string"}, "diedCountryCode": {"t": "string"}}, "$k": ["_id", "id
"diedCountry", "diedCountryCode", "diedCity", "gender", "prizes"]}, "id": "1", "_id": "6485ec0c5ed
{"year": "1901", "share": "1", "category": "physics", "motivation": "\"in recognition of the extra
med after him\"", "affiliations": [{"city": "Munich", "name": "Munich University", "country": "Ger
h", "firstname": "Wilhelm Conrad", "bornCountry": "Prussia (now Germany)", "diedCountry": "Germany
```

```
defaultdb=> select * from ferretdb.nobels_7f990856 limit 1;
-[ RECORD 1 ]-----------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
----------------------------------------------------------------------------------------------------
_jsonb | {"$s": {"p": {"id": {"t": "string"}, "_id": {"t": "objectId"}, "born": {"t": "string"}, "
 {"p": {"year": {"t": "string"}, "share": {"t": "string"}, "category": {"t": "string"}, "motivatio
tring"}, "name": {"t": "string"}, "country": {"t": "string"}}, "$k": ["name", "city", "country"]}}
"t": "array"}, "surname": {"t": "string"}, "bornCity": {"t": "string"}, "diedCity": {"t": "string"
ring"}, "bornCountryCode": {"t": "string"}, "diedCountryCode": {"t": "string"}}, "$k": ["_id", "id
"diedCountry", "diedCountryCode", "diedCity", "gender", "prizes"]}, "id": "1", "_id": "6485ec0c5ed
{"year": "1901", "share": "1", "category": "physics", "motivation": "\"in recognition of the extra
med after him\"", "affiliations": [{"city": "Munich", "name": "Munich University", "country": "Ger
h", "firstname": "Wilhelm Conrad", "bornCountry": "Prussia (now Germany)", "diedCountry": "Germany
```

```
defaultdb=> select * from ferretdb.nobels_7f990856 limit 1;
-[ RECORD 1 ]-----------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
_jsonb | {"$s": {"p": {"id": {"t": "string"}, "_id": {"t": "objectId"}, "born": {"t": "string"}, "
 {"p": {"year": {"t": "string"}, "share": {"t": "string"}, "category": {"t": "string"}, "motivatio
tring"}, "name": {"t": "string"}, "country": {"t": "string"}}, "$k": ["name", "city", "country"]}}
"t": "array"}, "surname": {"t": "string"}, "bornCity": {"t": "string"}, "diedCity": {"t": "string"
ring"}, "bornCountryCode": {"t": "string"}, "diedCountryCode": {"t": "string"}}, "$k": ["_id", "id
"diedCountry", "diedCountryCode", "diedCity", "gender", "prizes"]}, "id": "1", "_id": "6485ec0c5ed
{"year": "1901", "share": "1", "category": "physics", "motivation": "\"in recognition of the extra
med after him\"", "affiliations": [{"city": "Munich", "name": "Munich University", "country": "Ger
h", "firstname": "Wilhelm Conrad", "bornCountry": "Prussia (now Germany)", "diedCountry": "Germany"
```

```
defaultdb=> select * from ferretdb.nobels_7f990856 limit 1;
-[ RECORD 1 ]--------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
------------------------------------------------------------------------------------------------
_jsonb | {"$s": {"p": {"id": {"t": "string"}, "_id": {"t": "objectId"}, "born": {"t": "string"}, "
 {"p": {"year": {"t": "string"}, "share": {"t": "string"}, "category": {"t": "string"}, "motivatio
tring"}, "name": {"t": "string"}, "country": {"t": "string"}}, "$k": ["name", "city", "country"]}}
"t": "array"}, "surname": {"t": "string"}, "bornCity": {"t": "string"}, "diedCity": {"t": "string
ring"}, "bornCountryCode": {"t": "string"}, "diedCountryCode": {"t": "string"}}, "$k": ["_id", "id
"diedCountry", "diedCountryCode", "diedCity", "gender", "prizes"]}, "id": "1", "_id": "6485ec0c5ed
{"year": "1901", "share": "1", "category": "physics", "motivation": "\"in recognition of the extra
med after him\"", "affiliations": [{"city": "Munich", "name": "Munich University", "country": "Ger
h", "firstname": "Wilhelm Conrad", "bornCountry": "Prussia (now Germany)", "diedCountry": "Germany
```

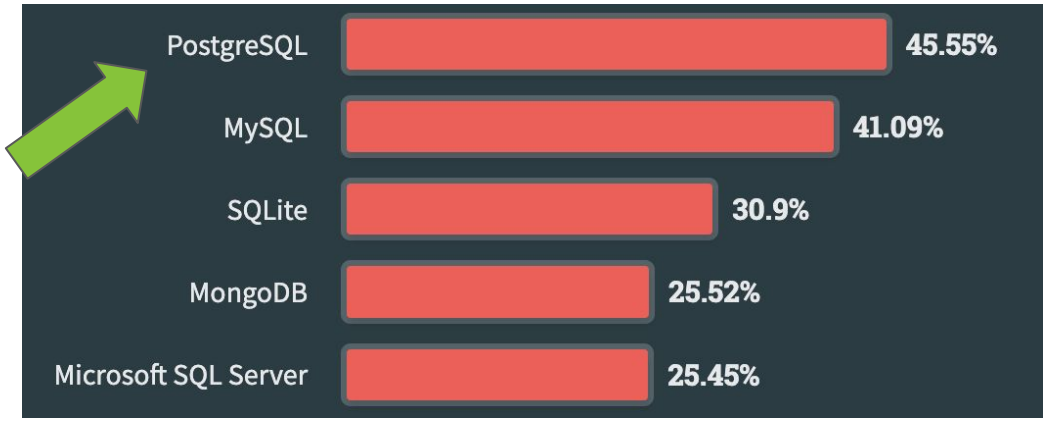
FerretDB

It's here.
FerretDB 1.0 GA

a truly Open Source MongoDB database alternative

www.ferretdb.io

Annual Stack Overflow developer survey (2023)



https://survey.stackoverflow.co/2023/#most-popular-technologies-database

Merci !

SCAN ME

	aiven	mongoDB. Atlas
Plan (ressources)	2 CPU - 8 GB RAM - 175 GB stockage	2 CPU - 8 GB RAM - 175 GB stockage
Haute disponibilité	✓	✓
PITR	✓	✓
Failover automatique	✓	✓
Multi-cloud	✓	✓
Tarif	**400 $/mois** *(tout inclus)*	**615 $/mois** *(474,5 + Backups (~10%) + Network (~10%) + Audit log (10%))*