ONLINE, AROUND-THE-WORLD OPEN SOURCE DATABASE CONFERENCE

PERCONA LIVEONLINE

MAY 12 - 13th

2021

MongoDB surviving after unclean shutdowns

About Me

- Senior Database Engineer
- 15 Years of experience with data
- Started with Relational Databases in Sql Server 7.0
- Today I work in Brazil helping DBACorp' customers get the best results.
- Over the Years have worked with:
  - MongoDB
  - SQL Server
  - Cassandra
  - Redis
  - Kafka

Alexandre Araujo

alexandre.araujo@dbacorp.com.br

@aleraraujo16

linkedin.com/in/alexandrearaujo16/

Agenda

- MongoDB internally has a powerful mechanism to face unexpected interruptions.
- We will talk about:
  - What WAL protocols means and how it was implemented to the MongoDB
  - How works the journal process
  - What kind of disasters impacts the MongoDB database
  - Why we have must concern about points of failure
  - When recover process takes action
  - And some configurations and strategies that you should use in your deploy.

Database Darwinism

After World War II

After Cold War

Cloud and Data Explosion

FLAT FILES

Magnetic Tape

Magnetic Disk

ORACLE DATABASE

Microsoft SQL Server

MySQL

PostgreSQL

mongoDB

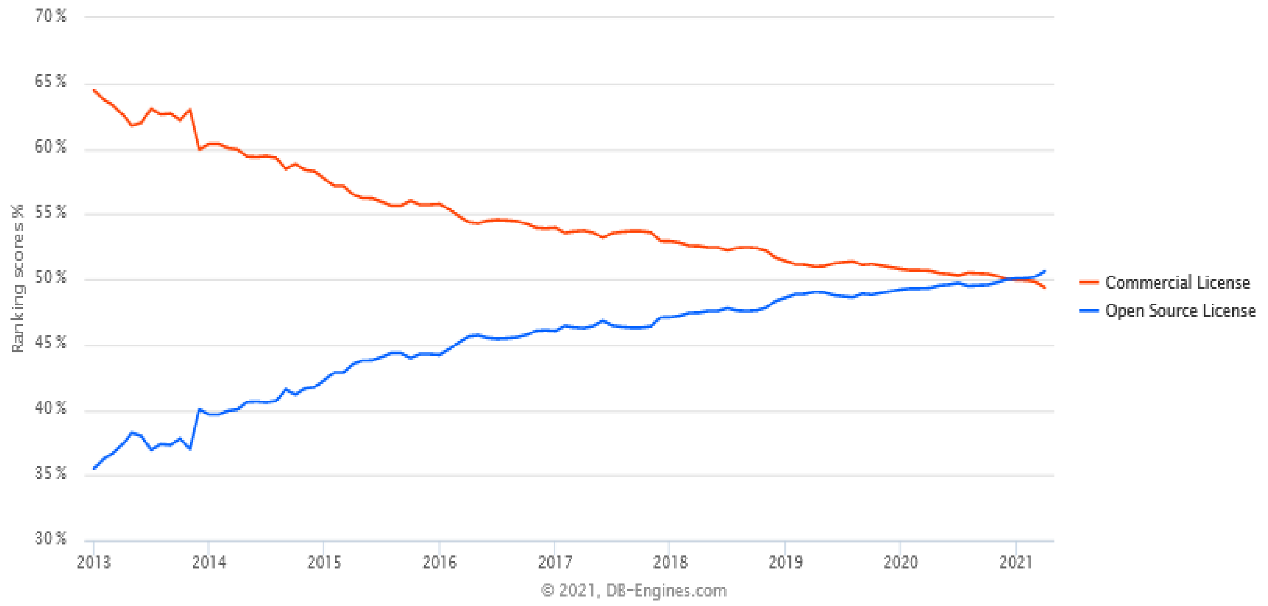
aws

Google Cloud Platform

APACHE kafka

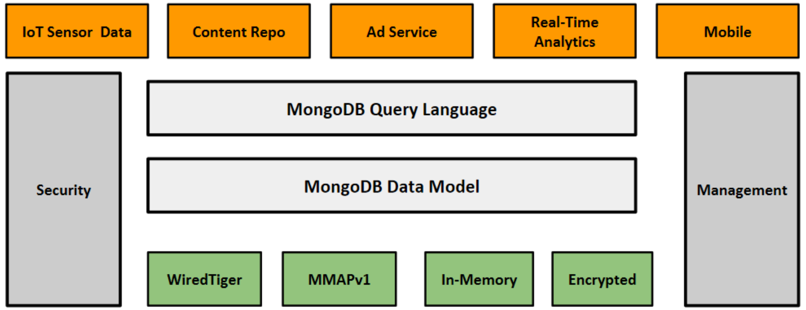
kubernetes

Open Source Database Trends 2021



MongoDB Storage Engines

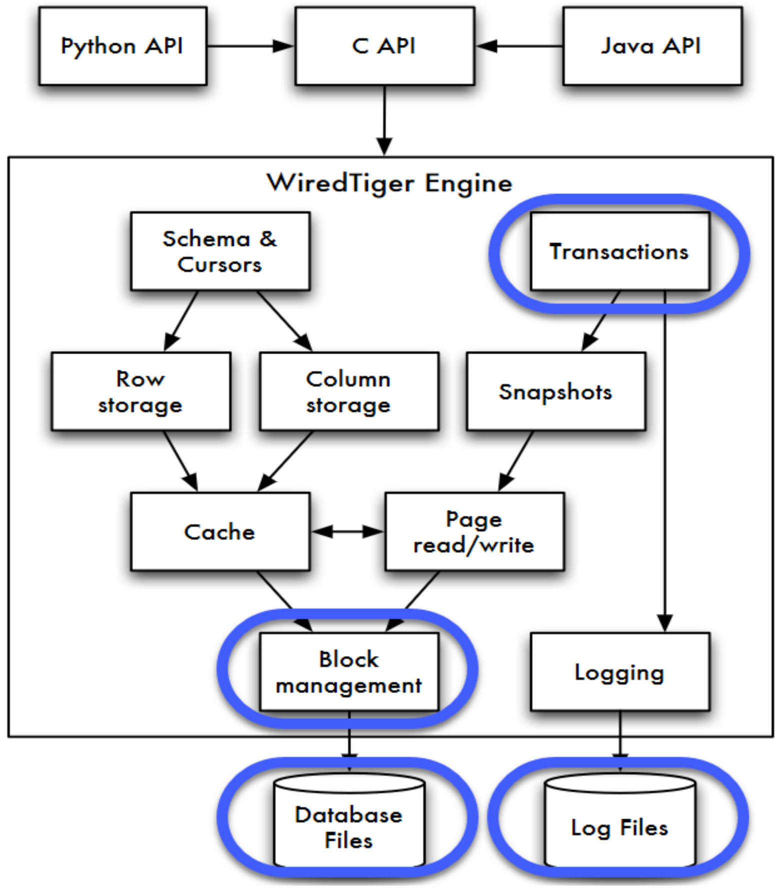
- Storage Engine is responsible for managing how data is stored.
- MongoDB provides a variety of Storage engines
- You can choose the appropriate Storage engine according your deploy.
- WiredTiger is the default Storage engine since MongoDB 3.2.



WiredTiger Storage Engine

- WiredTiger uses snappy compression by default with up to 80% compression
- The WiredTiger in MongoDB uses C-API although exists WiredTiger in Python and Java.
- WT takes advantage of modern hardware and more performance between threads
- Eliminate blocking due to concurrency control using MVCC
- Between 7 and 10x better write throughput from older storages
- Documentation: https://source.wiredtiger.com/10.0.0/index.html

WiredTiger - Architecture



WiredTiger Storage Engine

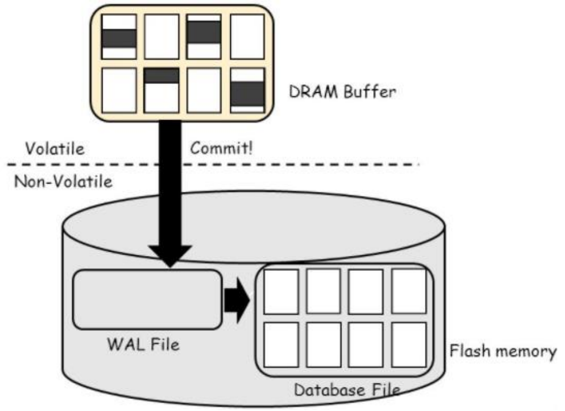
Durability Mode Three Modes	MongoD Crash	OS Crash
In-Memory	Potential data Loss	Potential data Loss
Write-No-Sync	Data Always Recoverable	Potential data Loss
FULL SYNC	Data Always Recoverable	Data Always Recoverable

Two modes of durability in MongoDB

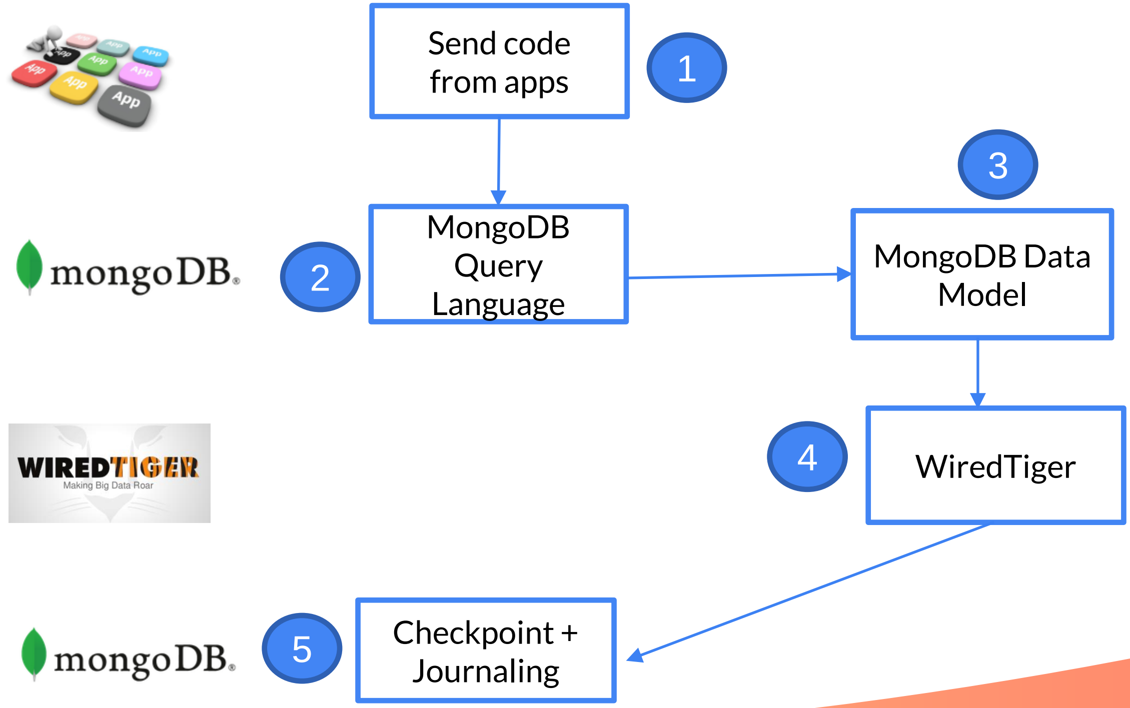
+

WAL Protocol

- Write Ahead Logging protocol, most well-know recovery method, is the standard industry ensures that a record of every change to the database is available while attempting to recover from a crash
- When a data is changed and committed it is forced do stable Storage
- Traditional databases use a write-ahead log for recovery
- In the ACID systems WAL generally referes to Durability



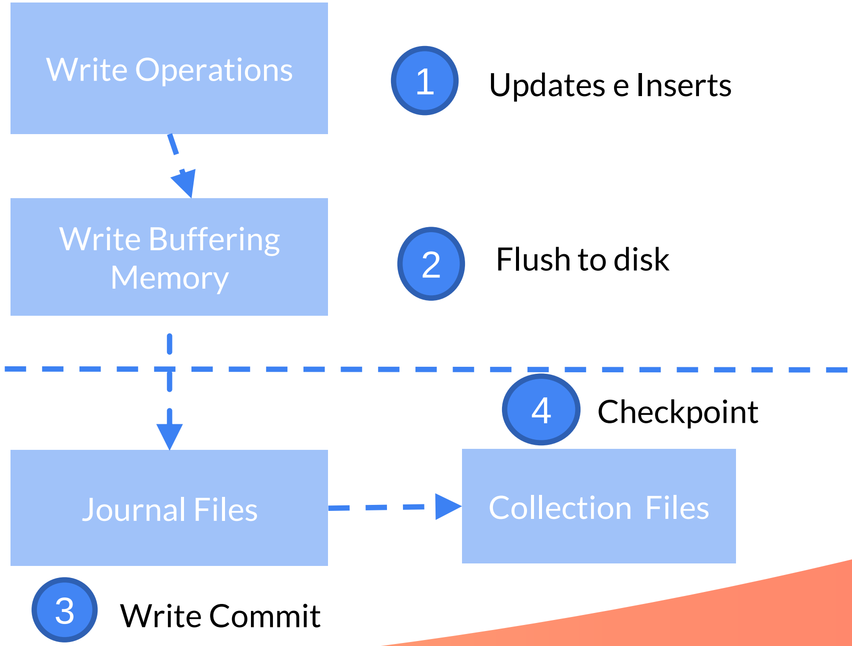
A data journey from user writes until commit



WAL – Write Ahead Logging

Volatile Memory

Non Volatile Memory



Journalism

- WT creates one journal record for each write and index operation
- Max size limit is 100 megabytes and the minimum jornal record size is 128 bytes
- This is not the Replication OpLog also it is not user-level transactions
- Automatically removes old journal files to maintain only the files needed to recover from last checkpoint

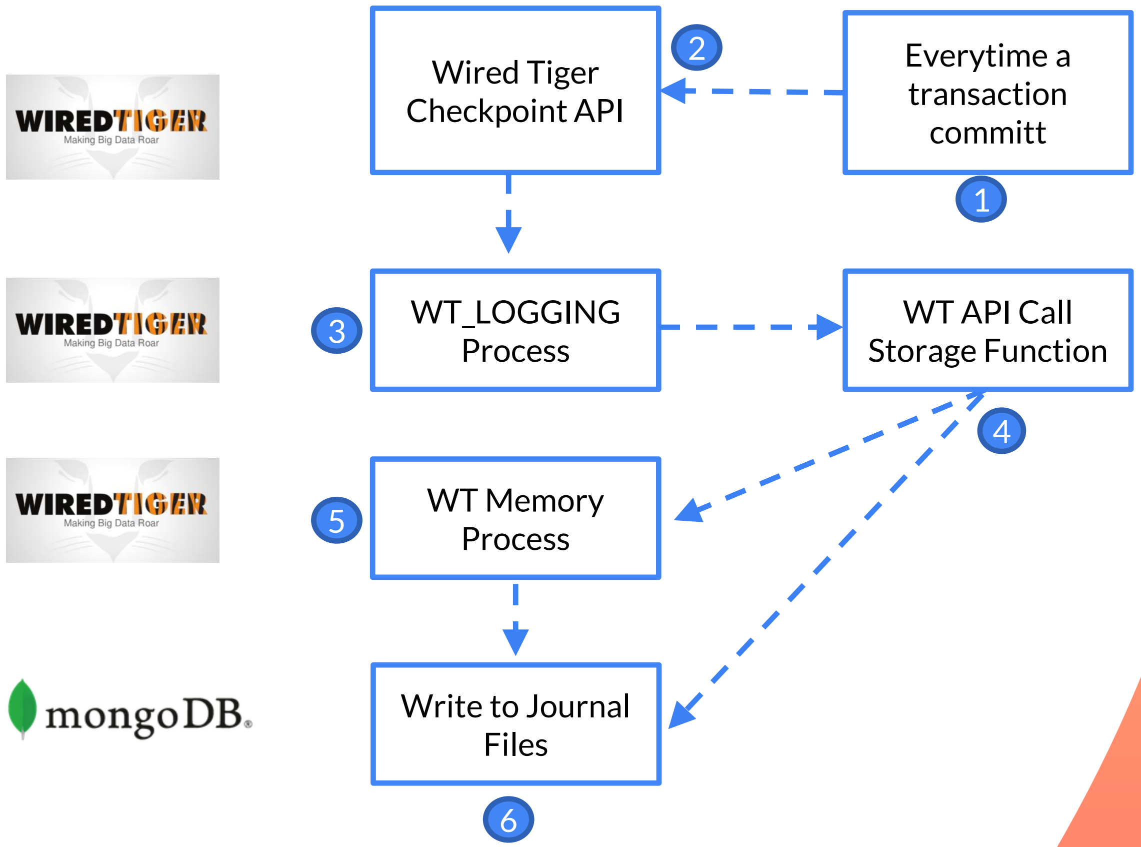
Journalism - Structure

- Location at: dbpath/jornal/WiredTigerLog.0000000001
- A record per each client initiated write operation.
- WiredTiger creates a single jornal record that includes both the update operation and it associated index modifications.
- Each record has a unique identifier.

Checkpoints

- Internal WiredTiger process
- Checkpoints flushs dirty data at every 60 seconds creating stable data
- The cosistent point-in-time snapshot of data writes all in memory to disk and ensures there is no point in time where data might be lost.
- Checkpoint will mark inside the jornal indicating the last checkpoint

Journalism occurs at every 100 ms



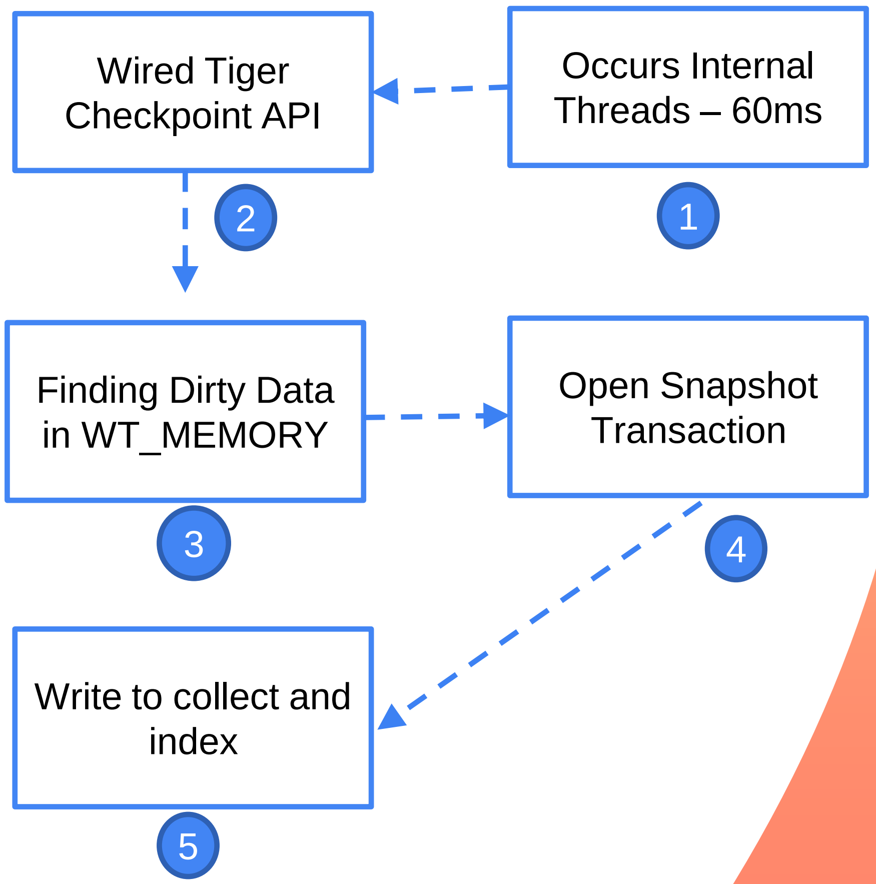
WIREDTIGER Making Big Data Roar
WIREDTIGER Making Big Data Roar
WIREDTIGER Making Big Data Roar
mongoDB.
Everytime a transaction committ
1
2
Wired Tiger Checkpoint API
3
WT_LOGGING Process
WT API Call Storage Function
4
5
WT Memory Process
Write to Journal Files
6

Checkpoints occurs at every 60 ms

WIREDTIGER
Making Big Data Roar

WIREDTIGER
Making Big Data Roar

mongoDB.



Occurs Internal Threads – 60ms
1
Wired Tiger Checkpoint API
2
Finding Dirty Data in WT_MEMORY
3
Open Snapshot Transaction
4
Write to collect and index
5

Failure Scenarios

- What happens after a crash:
  - The data is lost but the file system is consistent
  - The log has exactly the right information to fix the problem

- Kinds of failure:
  - Hardware Failure
  - Power Failure
  - Storage Volume Failure
  - Kill unxpected mongod process
  - Unexpected restart S.O.

Recovering – How mongod Recovers from crash

- Faster recovery = checkpoint + journal
- First check data files to find the id of the last checkpoint
- Then check the journal files for the record that matches of last checkpoint
- And finally applies the jornal files since the last checkpoint
- MongoDB does not guarantee storage failure after a checkpoint.

Recovery Process

WRITE OPERATION WT_MEMORY	COMMIT DATA JOURNAL	COLLECTION FILES
Seahawks	Seahawks	Seahawks
Panthers	Panthers	Panthers
Packers	Packers	Packers
	Checkpoint	Steelers
Steelers	Steelers	Eagles
	Checkpoint	
Eagles	Eagles	
	Crash	
	Recovery	

MongoDB InMemory

- Performance is a good indicator for use MongoDB in Memory
- It become durable when filling the memory buffer
- Become durable when a new jornal is created
- Idle systems can trigger the write jounal every 50ms
- Crashs does not guarantee data after last checkpoint

Tuning

- Change configuration option “commitIntervalMs”according your workloads
- The jornal default value is 100 ms and range can be between 1 and 500 ms
- Use symlink for journal to a different hubs increasing the concorrency of your write deploy
- Big checkpoints can cause painful performance and you can control the WiredTiger CacheSize limiting the amount of dirty pages and sizing eviction threads
- TRADE-OFFS
    - More frequent checkpoints means less record that you reply and more faster your recovery althoug more intense use of ou storage
    - Less frequent checkpoints means more record that you reply and more longer will be your recovery process

“

Open Source is the new oil.

Thanks!

Any questions?

You can find me at:

- @aleraraujo16 - Twitter
- alexandre.araujo@dbacorp.com.br

THANK YOU !

ONLINE, AROUND-THE-WORLD OPEN SOURCE DATABASE CONFERENCE

PERCONA

LIVEONLINE

MAY 12 - 13th

2021