Process

Indications   When you are aware of a `mongod` instance running without journaling that stops unexpectedly **and** you're not running with replication, you should always run the repair operation before starting MongoDB again. If you're using replication, then restore from a backup and allow replication to perform an initial *sync* (page 593) to restore data.

If the `mongod.lock` file in the data directory specified by `dbPath`, `/data/db` by default, is *not* a zero-byte file, then `mongod` will refuse to start, and you will find a message that contains the following line in your MongoDB log our output:

```
Unclean shutdown detected.
```

This indicates that you need to run `mongod` with the `--repair` option. If you run repair when the `mongodb.lock` file exists in your `dbPath`, or the optional `--repairpath`, you will see a message that contains the following line:

```
old lock file: /data/db/mongod.lock. probably means unclean shutdown
```

If you see this message, as a last resort you may remove the lockfile **and** run the repair operation before starting the database normally, as in the following procedure:

Overview

> **Warning:** Recovering a member of a replica set.
> Do not use this procedure to recover a member of a *replica set*. Instead you should either restore from a *backup* (page 184) or perform an initial sync using data from an intact member of the set, as described in *Resync a Member of a Replica Set* (page 633).

There are two processes to repair data files that result from an unexpected shutdown:

- Use the `--repair` option in conjunction with the `--repairpath` option. `mongod` will read the existing data files, and write the existing data to new data files.

  You do not need to remove the `mongod.lock` file before using this procedure.

- Use the `--repair` option. `mongod` will read the existing data files, write the existing data to new files and replace the existing, possibly corrupt, files with new files.

  You must remove the `mongod.lock` file before using this procedure.

Note: `--repair` functionality is also available in the shell with the `db.repairDatabase()` helper for the `repairDatabase` command.

Procedures
Important: Always Run `mongod` as the same user to avoid changing the permissions of the MongoDB data files.

Repair Data Files and Preserve Original Files   To repair your data files using the `--repairpath` option to preserve the original data files unmodified.

Repair Data Files without Preserving Original Files   To repair your data files without preserving the original files, do not use the `--repairpath` option, as in the following procedure:

> **Warning:**   After you remove the `mongod.lock` file you *must* run the `--repair` process before using your database.