MongoDB Standard Queries vs Aggregations

How they are differ and when to use each

Standard Queries

- Primary way to interact with your MongoDB instance when used in conjunction with built in query methods like “find” and “update”
- Simple to use, makes writing filters quick and easy
- Simple has its limits. Prevents it from being viable in some use cases but works for a vast majority of simple CRUD operations

Aggregations

- One of the most powerful features of MongoDB.
- Can create custom data pipelines to get data in just the right shape needed for your use case.
- Harder to use but is much more powerful and flexible.
- Complexity of some aggregations pipelines makes it more likely to introduce errors.
- Can be used to solve the same problems as standard queries and much more.

Why ever use standard queries over aggregations?

Let’s look at some use cases

Filtering documents based on its fields

```
db.users.find({email: “test@test.com”})
```

```
db.users.aggregate([{$match:{email: “test@test.com”}}])
```

Performing a SQL-like “Join”

Can’t do it!

```
db.users.aggregate([
    { $lookup: {
        from: "addresses",
        localField: "id",
        foreignField: "user_id",
        as: "addresses"
    }}
]);
```

Summary

- Standard queries are easier to work with and will suffice for a large majority of simple and intermediate use cases.
- Aggregations are the go-to for more complex processing and other advanced use cases that outside the scope of what can be done with simple queries.