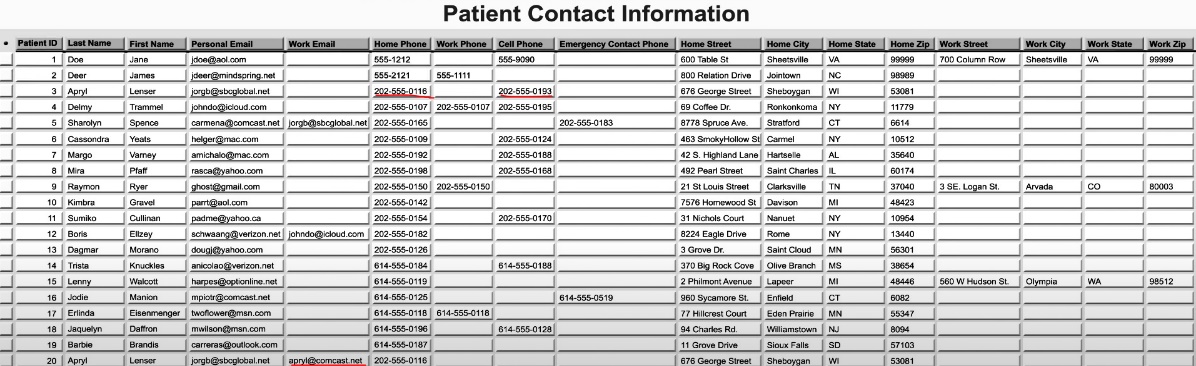
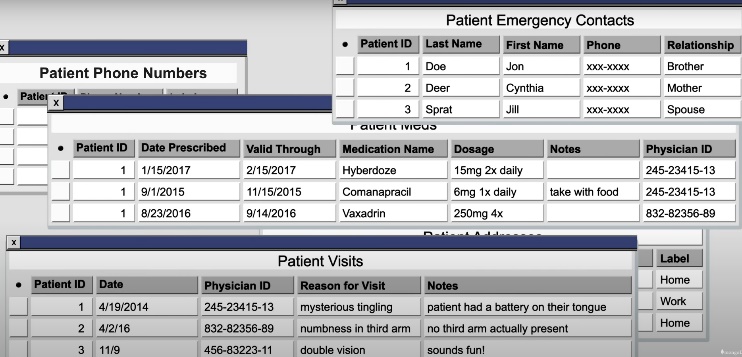
**The Fundamentals of MongoDB Aggregation**

**RDB**

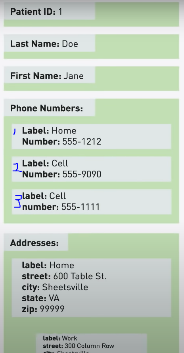
Databases are spreadsheets with metadata : For patients’ work email, entire new entry would be made

In RDB, data is organized with Primary keys (as index) in different tables : So, to look up at single patient, we traverse dozens of databases tables



* This complicates data understanding ability to write applications
* Hard to add features
* Fetching data from many sources [tables] is inefficient

**MongoDB**



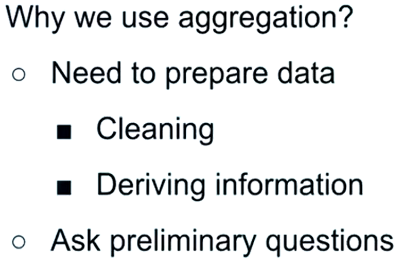
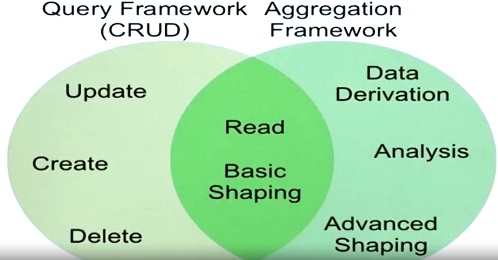
Document Model

**Structuring with flexible schema is:**

* Natural for programmers to read and code
* Easy for computers to process

Redundancy - Scalability - cloud native - **Fundamentally Distributed**

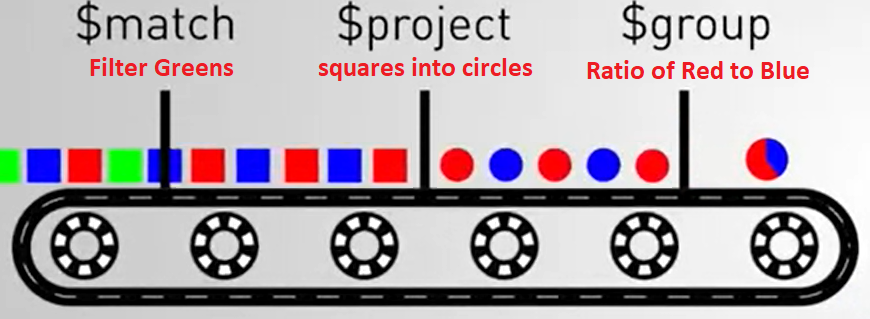
**MongoDB Query vs MongoDB Aggregation**

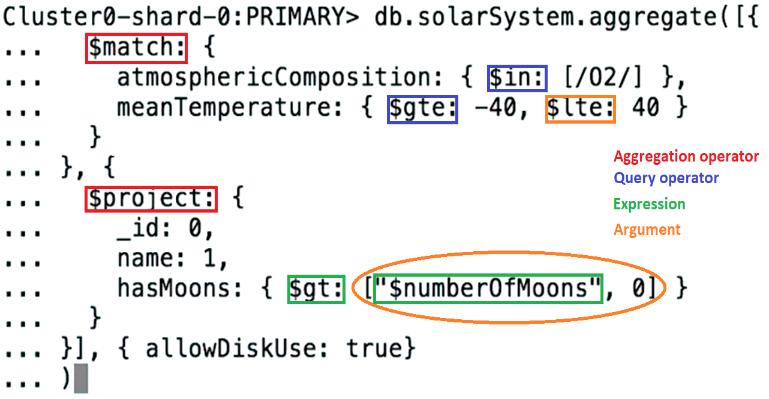


Aggregation framework enables developers to define FUNCTIONAL PIPELINES for data Preparation – Shaping – Analytics. Fundamental components are-

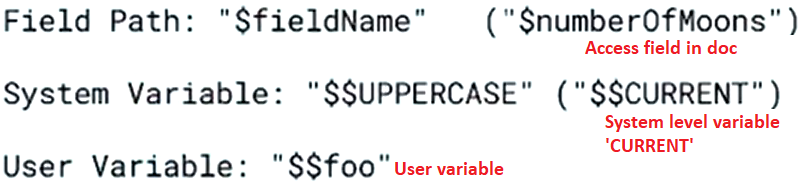
* **Stages** : Grouping-Sorting-Shaping etc.
* **Expressions** : Logic units of functionality

**Concept of Functional Pipeline**

****Assembly line of 1:N stages configurable for transformations using 1:N aggregation operators Or Expressions

**Aggregation framework: Structure and syntax**

[Aggregation Pipeline Quick Reference](https://docs.mongodb.com/manual/meta/aggregation-quick-reference/)

* **Operators** typically appear in **Key** position
* **Expressions [act like functions]** typically appear In **Value** position

|  |  |  |  |
| --- | --- | --- | --- |
| **Operator** | **Syntax** | **Description** | **Details** |
| **MATCH** | db.<DBname>.aggregate([{  $match : {<query>}  }]) | * Cannot use $where * Does not have projection * Match should be first operator if $text is used * First stage match increases query throughput as it can take adv of Indexing | **Filter operation** |
| **PROJECT** | db.<DBname>.aggregate([{  $project : {\_id:0, <specs>}  }]) | * Select, Remove, Reassign, derive new fields * Similar to **“map”** function in python * Mention all fields to retain; others are removed auto * Except “\_id” : needs explicit removal |  |
| **MULTIPLY** | db.<DBname>.aggregate([{  $multiply : [<expr1>, <expr2>,<expr3>, ...]  }]) | * Expr1\*Expr2\*…ExprN |  |
| **DIVIDE** | db.<DBname>.aggregate([{  $divide: [<expr1>,<expr2>]  }]) | * Expr1 / Expr2 |  |

**MongoDB expressions**

