

John Dohoney - Bio

John Dohoney has been in the High Tech community since graduation from College. He started out in the VAX/VMS world working on DOD projects with NASA. He holds 5 patents that range from Data Quality to Artificial Intelligence. John enjoys technology, and digging into a technical challenge. Johns work at 10-Gen has touched many areas of application usage across a very diverse set of vertical markets that mix real time, development, DevOps, and all issues data. As a developer, John stated his career with the Ada language, and uses Python, Closure, and Javascript with MongoDB users. John enjoys the outdoors with his wife whether it is on a Stand-Up Paddleboard, Skis, or on his two feet hiking the diverse micro-climates of California.





Javascript LA Meetup

MongoDB Atlas and Javascript

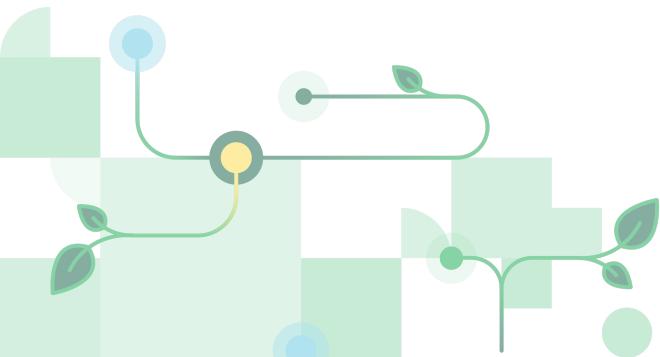
Javascript use in the Cloud



John Dohoney Jr.
Sr. Solutions Architect | MongoDB | @johndohoneyjr

Presentation Slides

Demo Repos



Meetup Agenda

As a Javascript Engineer, I want to learn:

1. How I can get a free Atlas Account
2. How Javascript is used in the Mongo Shell
3. How to access MongoDB Atlas using the MongoDB Javascript Driver
4. How I can use Javascript in Atlas to write:
 - a. Database Triggers
 - b. React and GraphQL



What tools do you need?

- **MongoDB Atlas Account**
Your cloud data platform
- **Node.js/npm**
- **Docker – On your Laptop**
- **REST Tools - optional**
Postman or curl clients



Set-Up a Free MongoDB Atlas Account



Go to: <https://www.mongodb.com/cloud/atlas/lp/try2>

Loading sample Data

SA-NA-WEST > JOHNDOHONEY-AWS

Clusters

Find a cluster...

Demo-Store
Version 4.2.6

[CONNECT](#) [METRICS](#) [COLLECTIONS](#) [...](#)

BACKUPS Active
CLUSTER TIER M30 (General)

REGION AWS / Oregon (us-west-2)
TYPE Replica Set - 3 nodes

LINKED STITCH APP Triggers_StitchApp
BI CONNECTOR Enabled

PAUSED for 12 more days

Disney-OCR
Version 4.0.18

[CONNECT](#) [METRICS](#) [COLLECTIONS](#)

BACKUPS Active
CLUSTER TIER M10 (General)

REGION AWS / Oregon (us-west-2)
TYPE Replica Set - 2 nodes

Operations R: 2.4 W: 0.4 • 2.4/s
Disk Usage 3.4 GB

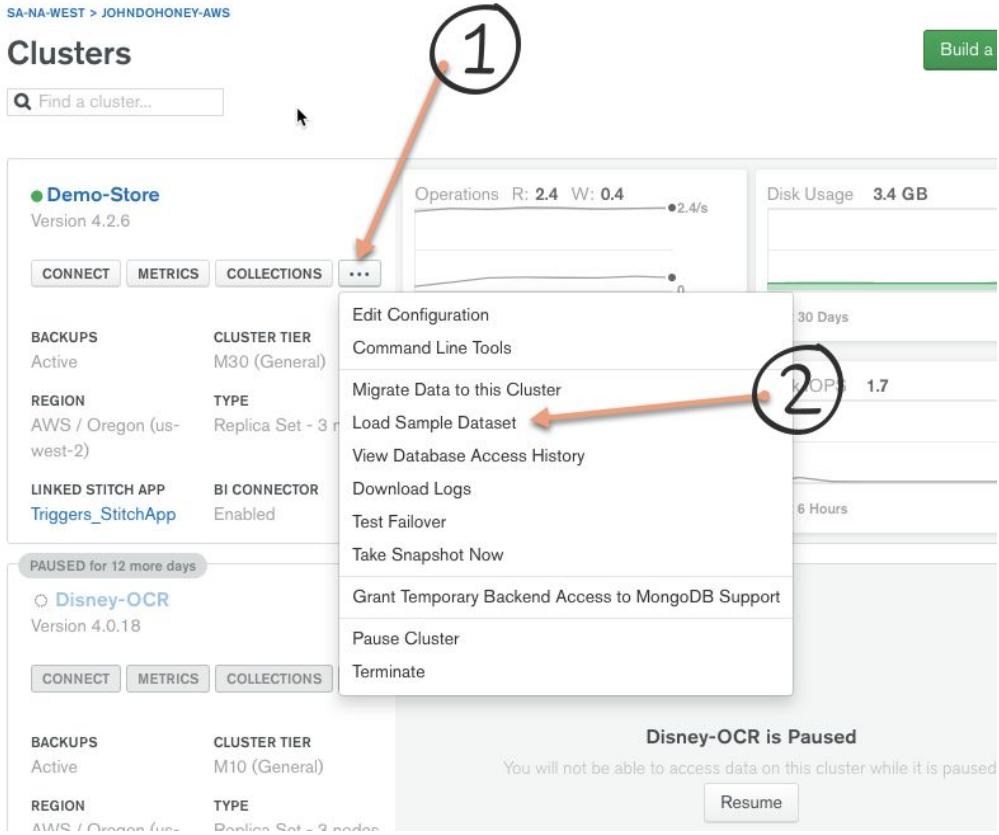
30 Days
1.7 OPS/s

6 Hours

[Edit Configuration](#)
[Command Line Tools](#)
[Migrate Data to this Cluster](#) **Load Sample Dataset** [View Database Access History](#)
[Download Logs](#)
[Test Failover](#)
[Take Snapshot Now](#)
[Grant Temporary Backend Access to MongoDB Support](#)
[Pause Cluster](#)
[Terminate](#)

Disney-OCR is Paused
You will not be able to access data on this cluster while it is paused

[Resume](#)



Whitelisting your laptop for Secure access

MongoDB Atlas has always on security

By "White-listing" your Laptops public IP now you can access Atlas from your local shell using:

- Mongo Shell
- MongoDB Javascript Driver
- MongoDB Compass

The screenshot shows the MongoDB Atlas web interface for managing network access. On the left, a sidebar lists project clusters (SA-NA-West, JohnDohoney-AWS) and various sections: DATA STORAGE (Clusters, Triggers, Data Lake BETA), SECURITY (Database Access), and NETWORK ACCESS (selected). In the main area, a modal dialog titled "Add IP Whitelist Entry" is open. It contains instructions about whitelisting, two buttons ("ADD CURRENT IP ADDRESS" and "ALLOW ACCESS FROM ANYWHERE", with the former being active), a "Whitelist Entry:" field containing "47.145.201.230", a "Comment:" field with placeholder text, a toggle switch for temporary entries (set to 6 hours), and "Cancel" and "Confirm" buttons. Below the modal, a larger section titled "Whitelist an IP address" with the sub-instruction "Configure which IP addresses can access your cluster" features a prominent green "Add IP Address" button and a "Learn more" link.

MongoDB Atlas

unlocks
agility and
reduces cost



Self-service
and elastic



Global and
cloud agnostic



Secure
by default



Comprehensive
monitoring

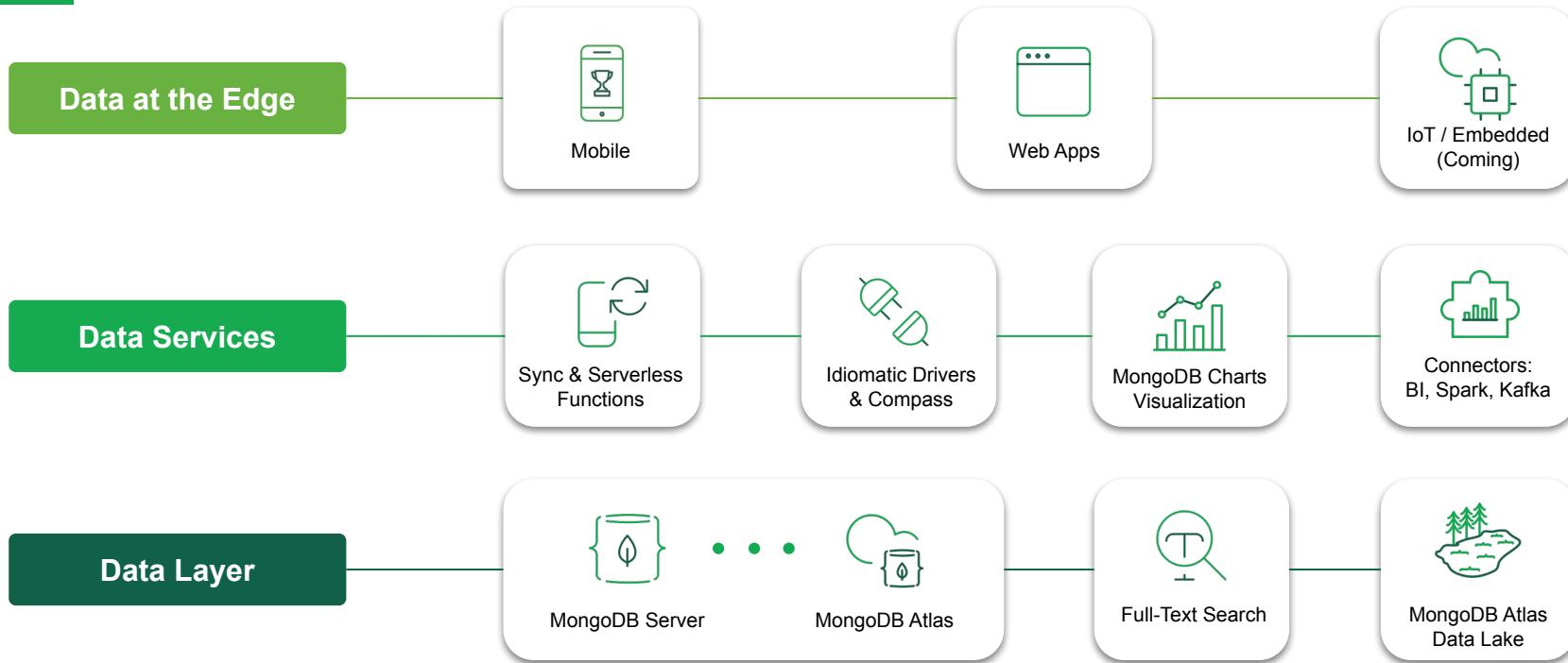


Managed
backup



Sync &
Serverless

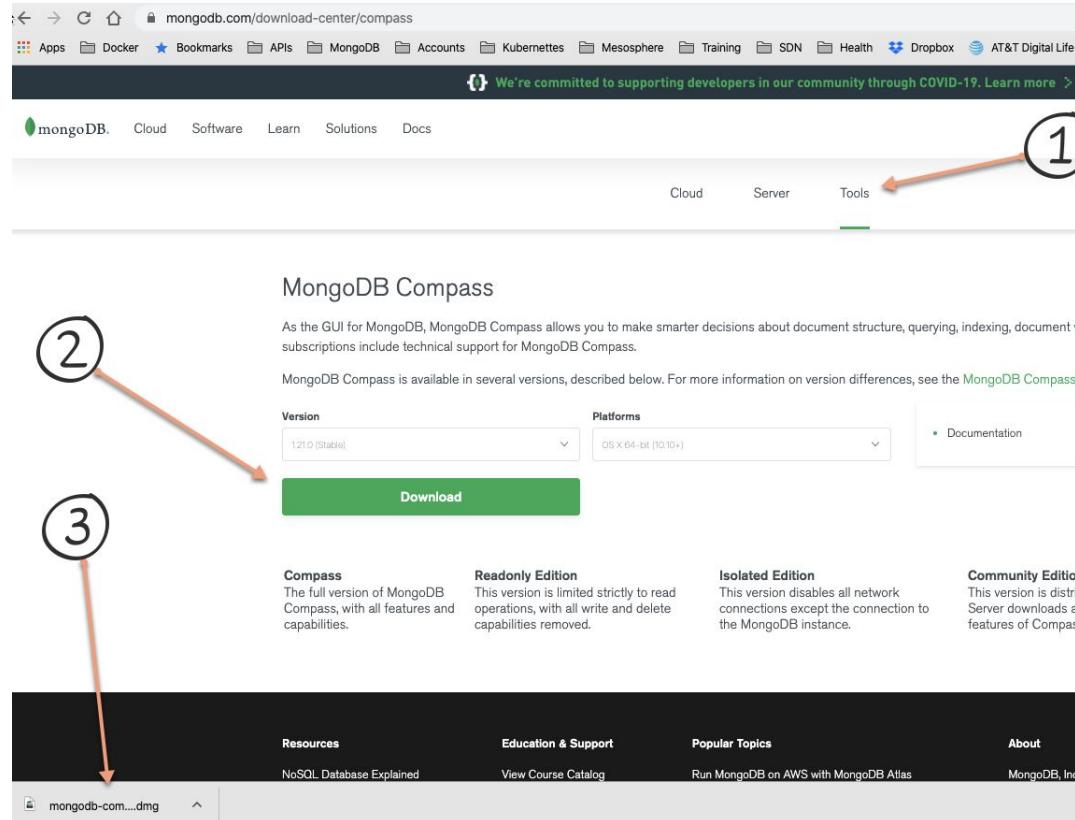
MongoDB Data Platform



Atlas Whitelisting and Walkthrough



Download MongoDB Compass



<https://www.mongodb.com/download-center/compass>

Mongo Compass Demo



VS Code - MongoDB Extension

The screenshot shows the VS Code interface with the MongoDB extension installed. On the left, the sidebar displays the 'EXTENSIONS: MARKETPLACE' section with the search term 'mongodb'. A red circle with the number '1' highlights the 'Azure Tools' extension. A red circle with the number '2' highlights the 'MongoDB for VS Code' extension in the main content area.

MongoDB for VS Code (Preview)

Details Feature Contributions Changelog

Azure Pipelines failed

MongoDB for VS Code makes it easy to work with MongoDB, whether your own instance or in [MongoDB Atlas](#).

Features

Navigate your MongoDB Data

- Navigate your database, collections and read-only views
- See the documents in your collections
- Get a quick overview of your schema

Results: sample_airbnb.listingsAndReviews.json

```
1 {
2   "_id": "10859872",
3   "listing_url": "https://www.airbnb.com/rooms/10859872",
4   "name": "Soho Cozy, Spacious & Convenient",
5   "summary": "Clean, fully furnish, Spacious 1 bedroom flat just off the escalator in Mid Levels. 2 minutes From Soho Bar and Restaurants. Located in a quite",
6   "space": "...",
7   "description": "Clean, fully furnish, Spacious 1 bedroom flat just off the escalator in Mid Levels. 2 minutes From Soho Bar and Restaurants. Located in a quite",
8   "neighborhood_overview": "...",
9   "notes": "...",
10  "travels": "...",
11  "access": "...",
12  "interaction": "...",
13  "house_rules": "...",
14  "property_type": "Apartment",
15 }
```

Node v10.13.0

VS Code - MongoDB Extension Playground

The screenshot shows the MongoDB extension's playground feature in VS Code. A red circle with the number '1' is highlighted on the 'MONGODB' sidebar icon.

MONGODB Sidebar:

- Connections
 - demo-store.hbxn.mongodb.net:27017 connected
 - JavascriptLA
 - sales
 - Documents

Code Editor:

```
1 // Select the database to use.  
2 use('JavascriptLA');  
3  
4 // The drop() command destroys all data from a collection.  
5 // Make sure you run it against proper database and collection.  
6 db.sales.drop();  
7  
8 // Insert a few documents into the sales collection.  
9 db.sales.insertMany([  
10   { '_id' : 1, 'item' : 'Little Widget', 'price' : 10, 'quantity' : 2, 'date' : new Date('2014-03-01T08:00:00Z') },  
11   { '_id' : 2, 'item' : 'Not A Widget', 'price' : 20, 'quantity' : 1, 'date' : new Date('2014-03-01T09:00:00Z') },  
12   { '_id' : 3, 'item' : 'Giant Widget', 'price' : 5, 'quantity' : 10, 'date' : new Date('2014-03-15T09:00:00Z') },  
13   { '_id' : 4, 'item' : 'Giant Widget', 'price' : 5, 'quantity' : 20, 'date' : new Date('2014-04-04T11:21:39.73') },  
14   { '_id' : 5, 'item' : 'Little Widget', 'price' : 10, 'quantity' : 10, 'date' : new Date('2014-04-04T21:23:13.3') },  
15   { '_id' : 6, 'item' : 'Medium Widget', 'price' : 7.5, 'quantity' : 5, 'date' : new Date('2015-06-04T05:08:13Z') },  
16   { '_id' : 7, 'item' : 'Medium Widget', 'price' : 7.5, 'quantity' : 10, 'date' : new Date('2015-09-10T08:43:00Z') },  
17   { '_id' : 8, 'item' : 'Little Widget', 'price' : 10, 'quantity' : 5, 'date' : new Date('2016-02-06T20:20:13Z') }  
18 ]);  
19  
20
```

Output Panel:

```
[  
  {  
    "_id": "Giant Widget",  
    "totalSaleAmount": 150  
  },  
  {  
    "_id": "Little Widget",  
    "totalSaleAmount": 120  
  },  
  {  
    "_id": "Not A Widget",  
    "totalSaleAmount": 20  
  }]
```

Bottom Status Bar:

Ln 13, Col 30 (5 selected) Spaces: 4 UTF-8 LF MongoDB Node v10.13.0

VS Code - MongoDB Extension - Login

The screenshot shows the MongoDB extension for VS Code. On the left is a dark sidebar with various icons for file operations like Open, Save, Find, and others. The main area has a title bar with tabs: "JSLA-widgets.mongodb" and "Connect to MongoDB". The "Connect to MongoDB" tab is active, showing a "Connect to MongoDB" dialog.

Connect to MongoDB

Enter your connection details below, or [connect with a connection string](#)

Hostname: localhost

Port: 27017

SRV Record:

Authentication: None

Replica Set Name:

Read Preference: Primary

SSL: None

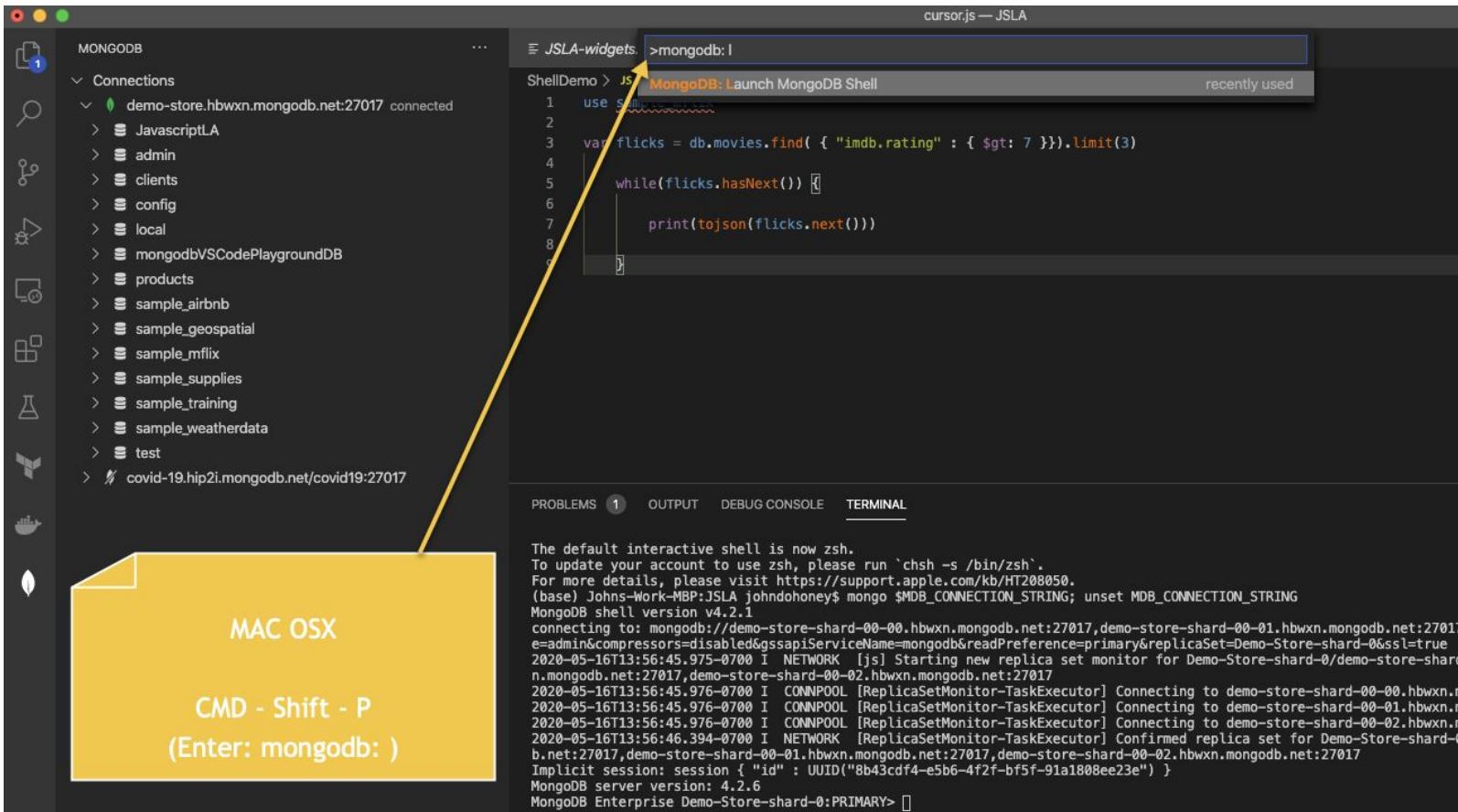
SSH Tunnel: None

Connect

A small white callout box is overlaid on the right side of the dialog, containing a green leaf icon and the text: "New to MongoDB and don't have a cluster? If you don't already have a cluster you can create one for free using [MongoDB Atlas](#)". Below this is a green button labeled "Create Free Cluster".



VS Code - MongoDB Shell



The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit <https://support.apple.com/kb/HT208050>.

```
(base) Johns-Work-MBP:JSLA johndhoney$ mongo $MDB_CONNECTION_STRING; unset MDB_CONNECTION_STRING
MongoDB shell version v4.2.1
connecting to: mongodb://demo-store-shard-00-00.hbwxn.mongodb.net:27017,demo-store-shard-00-01.hbwxn.mongodb.net:27017,
e=admin&compressors=disabled&gssapiServiceName=mongodb&readPreference=primary&replicaSet=Demo-Store-shard-0&ssl=true
2020-05-16T13:56:45.975-0700 I NETWORK  [js] Starting new replica set monitor for Demo-Store-shard-0/demo-store-shard-0.mongodb.net:27017,demo-store-shard-00-02.hbwxn.mongodb.net:27017
2020-05-16T13:56:45.976-0700 I CONNPOOL [ReplicaSetMonitor-TaskExecutor] Connecting to demo-store-shard-00-00.hbwxn.mongodb.net:27017
2020-05-16T13:56:45.976-0700 I CONNPOOL [ReplicaSetMonitor-TaskExecutor] Connecting to demo-store-shard-00-01.hbwxn.mongodb.net:27017
2020-05-16T13:56:45.976-0700 I CONNPOOL [ReplicaSetMonitor-TaskExecutor] Connecting to demo-store-shard-00-02.hbwxn.mongodb.net:27017
2020-05-16T13:56:46.394-0700 I NETWORK  [ReplicaSetMonitor-TaskExecutor] Confirmed replica set for Demo-Store-shard-0
Implicit session: session { "id" : UUID("8b43cdf4-e5b6-4f2f-bf5f-91a1808ee23e") }
MongoDB server version: 4.2.6
MongoDB Enterprise Demo-Store-shard-0:PRIMARY> 
```

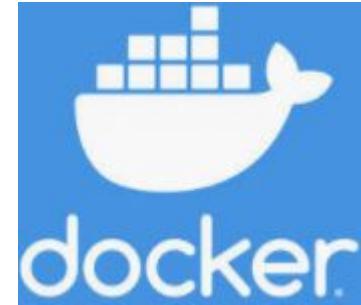
MAC OSX

CMD - Shift - P
(Enter: mongodb:)

Mongo VS Code Demo



Running a Mongo Client



```
docker run -d -p 3000:3000 mongoclient/mongoclient
```

```
$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
6e87e0835244	mongoclient/mongoclient	./entrypoint.sh nod...	4 hours ago	Up 4 hours	

```
$ sudo docker exec -it 6e87e0835244 /bin/bash
```

```
node@6e87e0835244:~$ mongo --version
MongoDB shell version v4.2.6-33-g7fd3c03
git version: 7fd3c03c548d0febfa1e871e16d638513c417c79
OpenSSL version: OpenSSL 1.1.0l 10 Sep 2019
allocator: tcmalloc
modules: none
build environment:
  distmod: debian92
  distarch: x86_64
  target_arch: x86_64
```

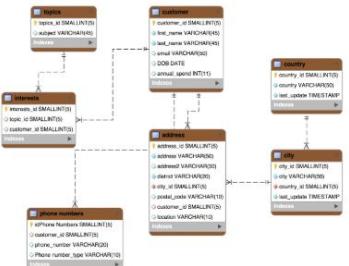


Mongo Shell Demo



MongoDB in a nutshell

Easy: Contrasting data models



Tabular (Relational) Data Model

Related data split across multiple records and tables

```
{
  "_id": ObjectId("5ad88534e3632e1a35a58d00"),
  "name": {
    "first": "John",
    "last": "Doe"
  },
  "address": [
    {
      "location": "work",
      "address": {
        "street": "16 Hatfields",
        "city": "London",
        "postal_code": "SE1 8D3"
      },
      "geo": {
        "type": "Point",
        "coord": [
          51.5065752,
          -0.109081
        ]
      }
    }
  ],
  "phone": [
    {
      "location": "work",
      "number": "+44-1234567890"
    }
  ],
  "dob": ISODate("1977-04-01T05:00:00Z"),
  "retirement_fund": NumberDecimal("1292815.75")
}
```

Document Data Model

Related data contained in a single, rich document



Versatile: Multiple data models, rich query functionality



JSON Documents



Tabular



Key-Value



Text



Geospatial



Graph

Rich Queries

Point | Range | Geospatial | Faceted Search | Aggregations | JOINs | Graph Traversals

HTAP (Hybrid Transactional/Analytical Processing)

Transactional



Primary

Secondary

Secondary

Secondary (use = analytics)

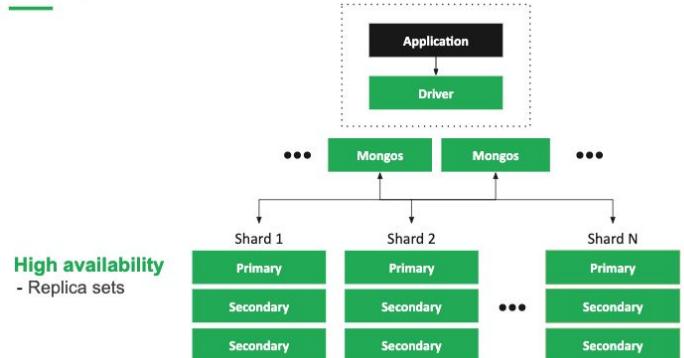
Secondary (use = analytics)

Analytical



Predictive Analytics & Data Science
Spark
R
python

Highly Available & Horizontally Scalable Architecture

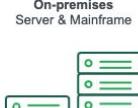


High availability
- Replica sets

Freedom to run anywhere



On-premises
Server & Mainframe



Private cloud

Hybrid cloud



Public cloud

Fully managed
cloud service

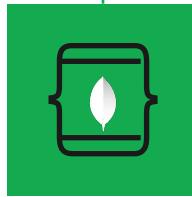


Thinking in Documents

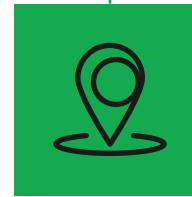


MongoDB: Built for Developer Productivity

Intelligent Data Platform



**Best way to work
with data**



**Intelligently put data
where you need it**



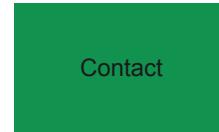
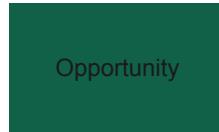
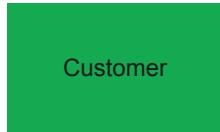
**Freedom
to run anywhere**

You probably have thousands of tables

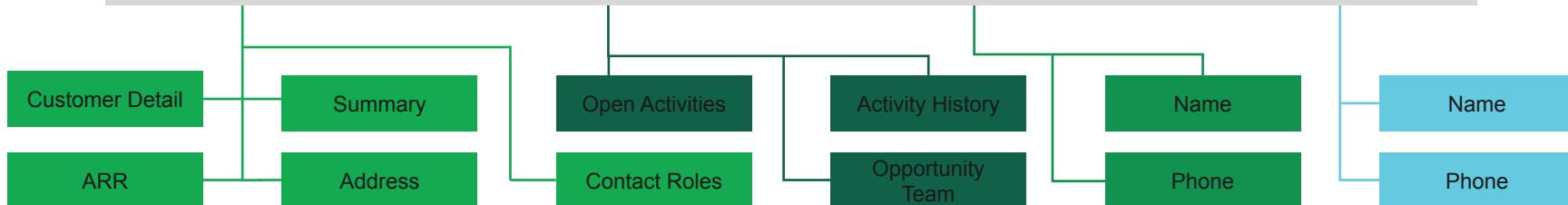


Go from this....

Objects



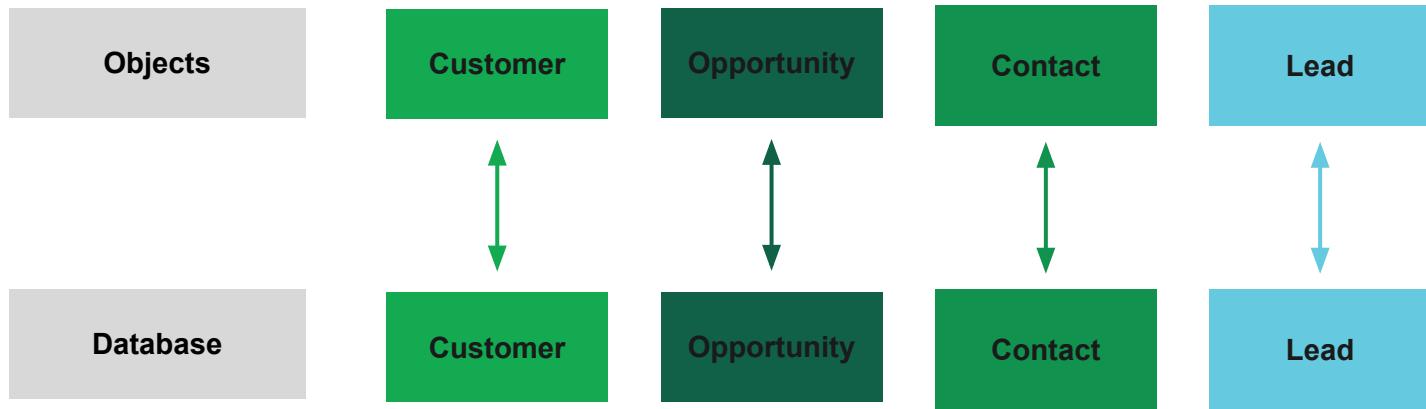
Object Relational Mapping Layer



Tables



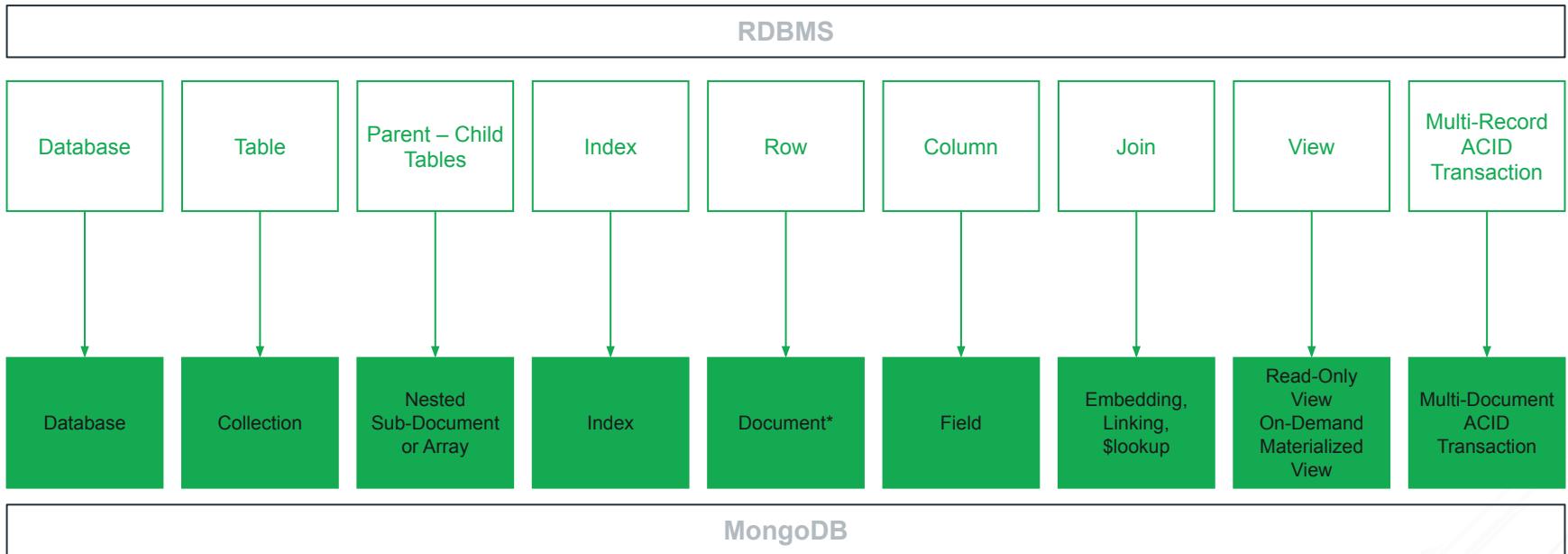
To this: store objects directly...





Some Terminology

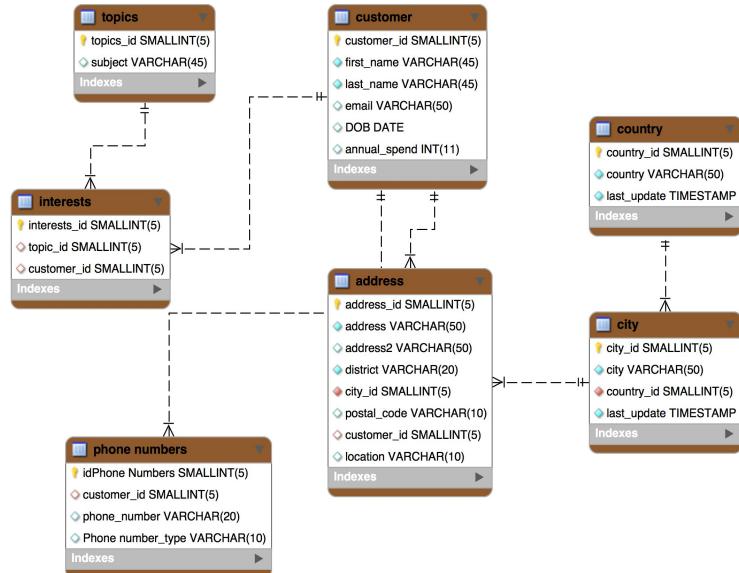
A comparison



* Proper document schema design yields more entity data per document than found in a relational database row



Easy: Contrasting data models



Tabular (Relational) Data Model

Related data split across multiple records and tables

```
{
  "_id" : ObjectId("5ad88534e3632e1a35a58d00"),
  "name" : {
    "first" : "John",
    "last" : "Doe"
  },
  "address" : [
    {
      "location" : "work",
      "address" : {
        "street" : "16 Hatfields",
        "city" : "London",
        "postal_code" : "SE1 8DJ"
      },
      "geo" : {
        "type" : "Point",
        "coord" : [
          51.5065752,
          -0.109081
        ]
      }
    },
    ...
  ],
  "phone" : [
    {
      "location" : "work",
      "number" : "+44-1234567890"
    }
  ],
  "dob" : ISODate("1977-04-01T05:00:00Z"),
  "retirement_fund" : NumberDecimal("1292815.75")
}
```

Document Data Model

Related data contained in a single, rich document



Easy: Document data model

- Naturally maps to objects in code
- Represent data of any structure
- Strongly typed for ease of processing
 - Over 20 binary encoded JSON data types
- Access by idiomatic drivers in all major programming language

```
{  
  "_id" : ObjectId("5ad88534e3632e1a35a58d00"),  
  "name" : {  
    "first" : "John",  
    "last" : "Doe" },  
  "address" : [  
    { "location" : "work",  
      "address" : {  
        "street" : "16 Hatfields",  
        "city" : "London",  
        "postal_code" : "SE1 8DJ"},  
        "geo" : { "type" : "Point", "coord" : [  
          51.5065752,-0.109081]}},  
    + { ... }  
  ],  
  "phone" : [  
    { "location" : "work",  
      "number" : "+44-1234567890"}],  
  + { ... }  
  ],  
  "dob" : ISODate("1977-04-01T05:00:00Z"),  
  "retirement_fund" : NumberDecimal("1292815.75")  
}
```

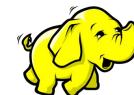


Easy: Drivers and Frameworks

Drivers



Frameworks



Stitch/GraphQL

API Demo



Stitch - GraphQL Resources

Stitch Overview - <https://www.mongodb.com/cloud/stitch>

Stitch Tutorials - <https://docs.mongodb.com/stitch/tutorials/>

Stitch GraphQL - <https://docs.mongodb.com/stitch/graphql/>

JavaScript SDK Demo



MongoDB Driver for Javascript Resources

Native Driver - <https://mongodb.github.io/node-mongodb-native/>

General Reference - <https://docs.mongodb.com/drivers/node>

NPM Package Reference - <https://www.npmjs.com/package/mongodb>

Driver - MongoDB Version compatibility - <https://docs.mongodb.com/drivers/driver-compatibility-reference>

MongoDB Developer Portal - <https://developer.mongodb.com/>

The screenshot shows the MongoDB Developer Portal homepage. At the top, there's a purple banner with the text "MongoDB.live, free & fully virtual | June 9th - 10th" and a "Register Now" button. Below the banner, the navigation bar includes the MongoDB logo, "Developer", "Learn", and "Community". The main content area features a large code snippet: `ideas.find({"attributes": ["fast", "innovative", "original"]})`. Below the code, the text "What will you create today?" is displayed. To the left of the code, there's a decorative graphic of colored virus-like particles (blue, green, red) on a grid. On the right, there are three cards: "Go" (orange background), "C#" (purple background), and "MongoDB + GraphQL" (pink background). At the bottom, there are four article cards: "How to work with Johns Hopkins University COVID-19 Data in...", "Multi-Document ACID Transactions in MongoDB with Go", "MongoDB & C Sharp: CRUD Operations Tutorial", and "Introducing GraphQL Support in MongoDB Atlas with Stitch". The footer contains the MongoDB logo.

{ } MongoDB.live, free & fully virtual | June 9th - 10th Register Now

mongoDB. | Developer Learn Community

ideas.find({ "attributes": ["fast", "innovative", "original"] })

What will you create today?



How to work with Johns Hopkins University COVID-19 Data in...

Multi-Document ACID Transactions in MongoDB with Go

MongoDB & C Sharp: CRUD Operations Tutorial

Introducing GraphQL Support in MongoDB Atlas with Stitch

mongoDB

MongoDB University- <https://university.mongodb.com/>

[My Courses](#) [All Courses](#)

Self-Paced • 5 Chapters • Online

M220JS: MongoDB for Javascript Developers

Learn the essentials of Node.js application development with MongoDB.

[Courses](#) > M220JS

Start Now:

[Register](#)



Instructor: Matt Javyly

Matt Javyly is a Curriculum Engineer at MongoDB. Matt is a recent graduate of Carleton College, where he majored in Computer Science. Before joining MongoDB, Matt worked at Lenddo, where he used MongoDB to build verification products for

What You'll Learn

This course will teach you how to use MongoDB as the database for a Node.js application.

You will play the role of a back-end developer for a Node.js application, where your job is to implement the application's communication with MongoDB. Using the Node.js driver you will read and write data to the database, use the aggregation framework, manage the configuration of the database client, and create a robust application by handling exceptions and timeouts.

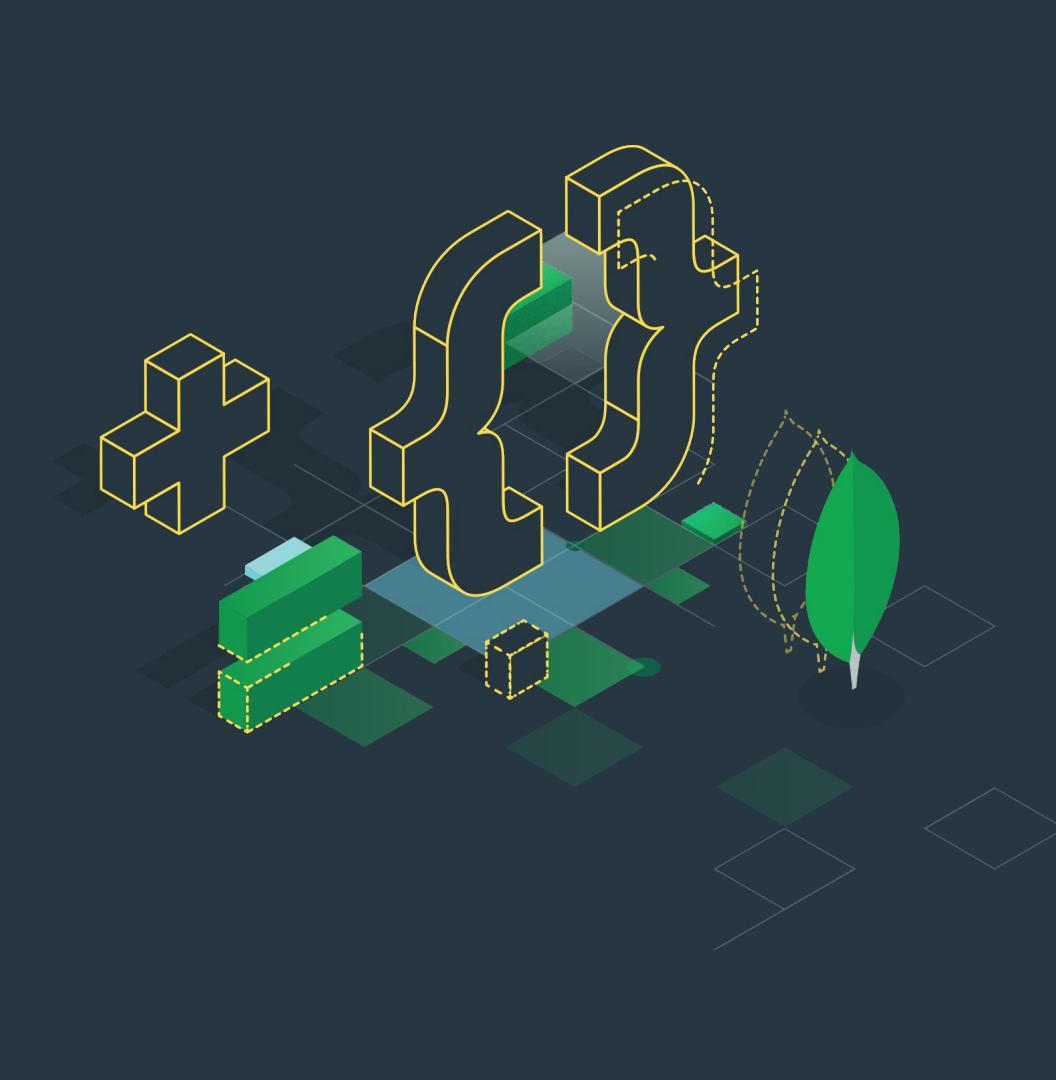
Prerequisites:

We highly recommend taking [M001](#) prior to taking this course. A basic understanding of MongoDB's document model will help you get the most out of this course.

What You'll Build

You'll build the back-end for a movie-browsing application called MFlix.

Using the Node.js driver for MongoDB, you will implement MFlix's basic functionality. This includes basic and complex



MONGODB .LIVE

June 9-10, 2020

A **free**, fully digital, two-day event featuring globally-accessible content including keynotes, breakout sessions, hands-on tutorials, digital sessions, an interactive Community Cafe, and more.

For more information and to register, visit the [event website](#).

Javascript Sessions:

Building Your First GraphQL Client in JS

We will answer the following questions. What is the philosophy behind GraphQL? How do you architect a scalable schema? How can GraphQL boost productivity? How can you avoid common pitfalls?

We will then get a GraphQL server up and running while focusing on exploring real-world patterns for architecting our schema. We will discuss and implement practical steps to improve query performance, error handling and caching.

Speakers



Joe Karlsson

Developer Advocate, MongoDB

Joe Karlsson is a software engineer turned Developer Advocate at MongoDB. He comes from the frozen tundra of Minneapolis, Minnesota (and yes, it does get really cold here, and no, not everyone here has the accent from the movie, Fargo)

[On-demand] Building Reports 10x Faster with Pipelines

At KazooHR, as our larger customer base increased, we ran into issues with large reports taking a long time and or consuming large amounts of memory. To solve this issue, we constructed reports inside the aggregation pipeline and then utilized NodeJS streams to consume the aggregation and stream all data into a file. Come hear more about how we reduced report run times and decreased container counts using MongoDB.

Speakers



Ezekiel Keator

Software Engineer, Palo Alto Networks
Texas



Thank you

That's all folks

Net- working topics

1. Name, Your background in JavaScript, Where do you live?
2. What would be your superhero name?
3. What is one thing you learned from a JS project that went wrong?
4. What is the next deep dive in JS?
5. What is your favorite database?

