BMW & MongoDB Architecture and Infrastructure Working Session
VSS-based Data Middleware with Tiered Sync (Edge Server)

Arnaldo Vera, Industry Solutions MongoDB
The Concept

Onboard Applications

Cloud

Mobile Applications

Device Sync

MongoDB Atlas

GraphQL API

Realm Database

Web Applications
Connected Car without Edge Server

Legend
- **Realm** - Object Oriented lightweight DB.
- **Device Sync** - Automatic Synchronization engine
- **MongoDB** - Document Model based DB

Vehicle

- ECU X
- ECU A (Bordcomputer)
- ECU B (Bordcomputer)
- ECU Y
- Ethernet
- ECU C (Bordcomputer)

MongoDB Atlas

- Operational Data Plane (HOT)
- Analytical Data Plane (COLD)

User App

12 October 2023 | COVES Fall AMM 2023Troy, MI
Atlas Device Sync: Edge Server (Tiered Sync)
Atlas Device Sync: Standard Architecture

1. Server-side app updates
   maintenance schedule on document

2. The maintenance update is
   pushed down to each Client

MongoDB Atlas

Lot One

Vehicle

Client

Lot Two

Vehicle

Client
Lot One is offline but local clients can still communicate with each other.

Server-side app updates maintenance schedule on document.

Lot One comes back online and data is replicated between the lot and cloud - resolving conflicts.

The maintenance update is pushed down to Lot two edge server.
Connected Car with Edge Server

ECU X
ECU A
Bordcomputer

ECU B
Bordcomputer

ECU Y

Vehicle State
Telemetry Stream

Either or

ECU C
Bordcomputer

Operational Data Plane
(HOT)

Analytical Data Plane
(COLD)

Realm - Object Oriented lightweight DB.

Device Sync - Automatic Synchronization engine

MongoDB - Document Model based DB

Legend

User App

Closed Door

12 October 2023 | COVESAMM 2023Troy, MI
Tree of Objects
Realm Data Model & VSS in the vehicle

The VSS data model maps perfectly to Realm’s object based data model

**Vehicle Model in Realm**

```java
Class Vehicle = {
    vehicleIdentification: {
        vin: "string",
        model: "string",
        brand: "string",
    },
    ambientAirTemperature: "int",
    driveTrain: "DriveTrain?"
    ...
}
```

**Vehicle Model in VSS**

[Diagram showing the VSS data model]
Realm Delta Change Notification

Vehicle Signal Specification
“VSS”

Realm Schema & Change Notifications

Create delta notifications containing only the changed attributes

Console Output:
1 properties have been changed.
Modified Attributes: {
  ambientairtemperature: "21.0"
  ...
}

Out of the box change notifications ->
MongoDB Atlas address a range of application use cases and removing complexity

- Flexible data model that maps to how developers think/code
- Strongly consistent, support for ACID transactions
- Able to support full-text search functionality for delivering a great user experience
- Able to support data on mobile and embedded devices w/o having to manually sync data
- Able to deliver real-time analytics on live data w/o having to move data back & forth
Sync mechanism between device & cloud

1. **User makes change**
   User logs into the app and makes an edit

2. **SDK writes transaction**
   The Realm SDK writes the transaction, committing the object to disk, and sends the changes to the server

3. **Object Conversion**
   Atlas Device Sync automatically converts the Realm object to a MongoDB document

4. **Insert into collection**
   MongoDB document inserted into the Atlas collection

5. **Change Detected**
   Document changed by another user is picked up by changeStreams and forwarded to Atlas Device Sync

6. **Object Conversion**
   Atlas Device Sync automatically translates MongoDB documents to Realm objects and syncs down to the device

7. **Notification**
   Realm commits the change to disk. A notification is fired and the UI is updated displaying the new change to the user
How Device Sync works: dynamic queries

Vehicle A - Firmware 1
- Product.type == 'snow tire'
- AND quantity >= 4
- AND status == 'Available'

Vehicle A - Firmware 2
- Car.VIN == 5FJPYF19494B395

Vehicle B
- Delivery.date == 01-01-2022

Atlas Device Sync

MongoDB Atlas

Atlas Device Sync
How Device Sync works: hierarchical permissions

12 October 2023 | COVESAM Fall AMM 2023 Troy, MI
How Device Sync works: document-level permissions

- **ECU A**
  - Read: All fields
  - Write: All fields

- **ECU B**
  - Read: All fields
  - Write: Some fields

- **App A**
  - Read: All fields
  - Write: Some fields

**MongoDB Atlas**

- User
- Auto
- Order
- Notes
How Device Sync works: field-level permissions

- **Customer/Driver**
  - **ID**
    - `_id`: `{ "$oid": "$60e51a64d0ff67f998bc1732" }`
  - **Address**
    - `name": "Jane Leaf",
    - `address": {
      "street": "Bedford Ave"
      "city": "New York"
    }
  - **Billing**
    - `creditCard": "1234123412341234"
    - `balance": "300"
  - **Booking**
    - `orderDate": "May62022",
      "auto": "sedan",
      "days": "5",
      "returned": false,
  - **Rental Notes**
    - `childSeat": true
      "description": "for my pug"

- **Billing**
  - `_id`: `{ "$oid": "$60e51a64d0ff67f998bc1732" }`
  - `name": "Jane Leaf",
    - `address": {
      "street": "Bedford Ave"
      "city": "New York"
    }
  - `creditCard": "1234123412341234"
    - `balance": "300"

- **Agent**
  - `_id`: `{ "$oid": "$60e51a64d0ff67f998bc1732" }`
  - `name": "Jane Leaf",
    - `address": {
      "street": "Bedford Ave"
      "city": "New York"
    }
  - `orders": [{
      "orderDate": "May62022",
      "auto": "sedan",
      "days": "5",
      "returned": false,
  ]
  - `rentalNotes":[
      "childSeat": true
      "description": "for my pug"
  ]

12 October 2023 | COVESAMM 2023Troy, MI
How Device Sync works: Data Ingest

**Sensors**
- Sensors on machines measure humidity, temperature, status, etc. and send this data to the cloud.

**Logs**
- Logs are generated and continuously sent in real-time to the cloud.

**Autos/Trucks**
- Autos moving shipments send GPS location and continuously send updated data to the cloud.

---

**Atlas Device Sync**

**MongoDB Atlas**

**Atlas Time Series**
- Aggregated sensor data optimized in time series collection, with built-in thresholds that can trigger alerts.

**Atlas Charts**
- Visualize and analyze time series data.
## End-to-end, security features

### On-device

**On-device encryption:** Easily encrypt database files with the encryption APIs

**Authorization:** Rich authorization methods including anonymous, email/password, API key, Custom Function, Custom JWT, and also Facebook, Google, and Apple.

### In-flight

**In-flight encryption:** All network traffic is encrypted using Transport Layer Security (TLS).

**Granular permissions:** Document and field-level access permissions are used to dynamically determine the data synced to the device.

### In the cloud

**In-cloud security:** Features that integrate with your existing protocols and compliance standards.

- Encrypted storage volumes
- Network isolation
- Role-based access management

### Compliance Standards

- ISO
- SOC
- PCI
- HIPAA
- CSA
- HITRUST
- VPAT
- GDPR
Observations (i.e., raw data)

Data in Context (i.e., Categorized)

Information in Context (i.e., Semantic)

Knowledge-based action

WISDOM

COVESAS VSS
SHACL, JSON-LD

COVESAS VSS
JSON, YAML, XML
Connected Car with Edge Server

Legend
- Realm - Object Oriented lightweight DB.
- Device Sync - Automatic Synchronization engine
- MongoDB - Document Model based DB

Vehicle
- ECU X
- ECU A
  - Bordcomputer
- ECU Y
- ECU B
  - Bordcomputer
- ECU C
  - Bordcomputer
- Ethernet

MongoDB Atlas
- Operational Data Plane (HOT)
- Analytical Data Plane (COLD)

User App

12 October 2023 | COVES A Fall AMM 2023 Troy, MI
Demo time!

Video Link: https://www.youtube.com/watch?v=6qodPDPNoSg
Open Source

MongoDB & AWS Connected Vehicle End to End Demo Repository

Code samples and demos around using the Realm database in combination with MongoDB Atlas, Device Sync and AWS Sagemaker.

Demo Video

Developer Data Platform

About

Code samples and demos around using the Realm database in combination with MongoDB Atlas and device sync.

Readme

Activity

0 stars

1 watching

4 forks

Report repository

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

Contributors

felixreichenbach Felix Reichenbach

amhosamujica

humza-akhtar

93arnaldo Arnaldo Vera
Curious? -> Reach out

arnaldo.vera@mongodb.com

humza.akhtar@mongodb.com