

DataStax

# Developers

Implement Java & Python Apps  
with Apache Cassandra™



# 01



Introduction

# 02

Microservices  
Why with Apache Cassandra ?

# 03

Rest APIs

# 04

Sensor Application  
Drivers; via command line

# 05

Sensor Rest API  
FastAPI, SpringBoot

# 06

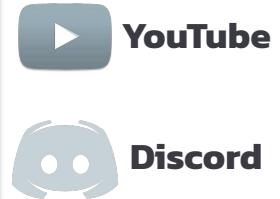
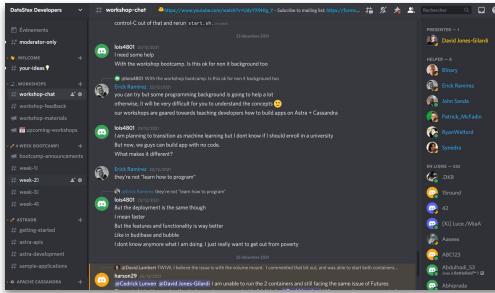
What's next?  
Quiz, Homework, Next week



**Livestream:** youtube.com/DataStaxDevs

**Questions:** <https://dtsx.io/discord>

#### Agenda



## Games and quizzes: [menti.com](https://menti.com)

How much experience do you have with the Spring Framework ?



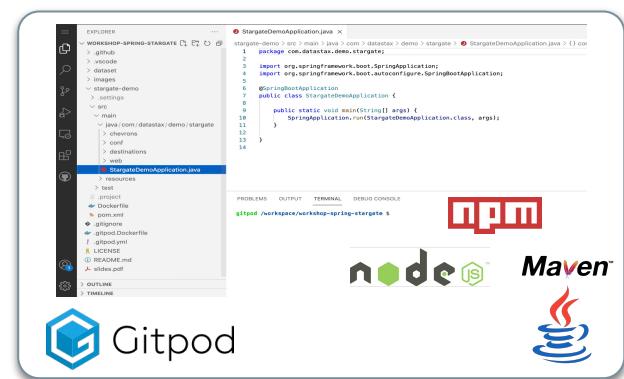
1

Attend the live sessions

## Database + GraphQL + PlayGround



DataStax  
**Astra DB**



The GitHub repository page for "DataStax-Examples / todo-astra-jamstack-notify" shows a list of branches (master, 5 branches, 5 tags), pull requests, and code. It includes sections for "Todos", "Releases", and "Actions". The "Todos" section lists items like "kidnerecursive readme updates" and "functions rfa todos". The "Releases" section shows a single release named "astra.datastax.com/register".

GitHub

2

## Complete Workshops Labs

The screenshot shows the DataStax Academy course page for DS220. At the top, it says "Course Content" and "Introduction". Below that, there are three sections: "Data Modeling Overview", "Data Modeling Overview Quiz", and "Relational Vs. Apache Cassandra". Each section has a "Start" button. To the right, there's a progress bar showing "Not Started 0/56" and a circular progress indicator at 0%. Below the progress bar are sections for "Badges" and "Competencies".

The screenshot shows the DataStax /dev website's "Learning Series Topics" page for "Cassandra Fundamentals". It features a blue header with the title "Cassandra Fundamentals" and a sub-section "Learning Series Topics". Below the header, there's a list of 11 topics: 01 Introduction to Apache Cassandra™, 02 Cassandra Query Language, 03 Keyspaces and Data Replication Strategies, 04 Tables with Single-Row Partitions, 05 Tables with Multi-Row Partitions, 06 Queries, 07 Advanced Data Types, 08 Tunable Consistency and Consistency Levels, 09 Linearizable Consistency and Lightweight Transactions, 10 Readme, and 11 Releases.

The screenshot shows a GitHub repository named "DataStax-Examples / todo-astra-jamstack-netlify". The repository has 177 commits ahead of "tjake/master". It includes branches like "master" and "astra". The repository details show 230 commits, 1 PR, 87 files, and 10 months ago. It also lists "astra" and "astra-datastax.com/register" as forks. The "Readme" section indicates no releases published. The GitHub logo is at the bottom.

3

Do your homework



# API and Microservices with Cassandra Homework

Welcome and thank you!

Here you can submit your homework for the DataStax Developers "Api and microservices with Cassandra" workshop.

In case of any questions please contact the organizers at <https://dtsx.io/aleks>, <https://dtsx.io/cedrick> or just send an email to [aleksandr.volochnev@datastax.com](mailto:aleksandr.volochnev@datastax.com).

- Workshop materials: <https://github.com/datastaxdevs/workshop-cassandra-application-development#readme>
- Discord chat: <https://dtsx.io/discord>

Email \*

Your email



4

Pass the Weekly Test



Go to [www.menti.com](http://www.menti.com) and use the code 3491 9972

## Leaderboard

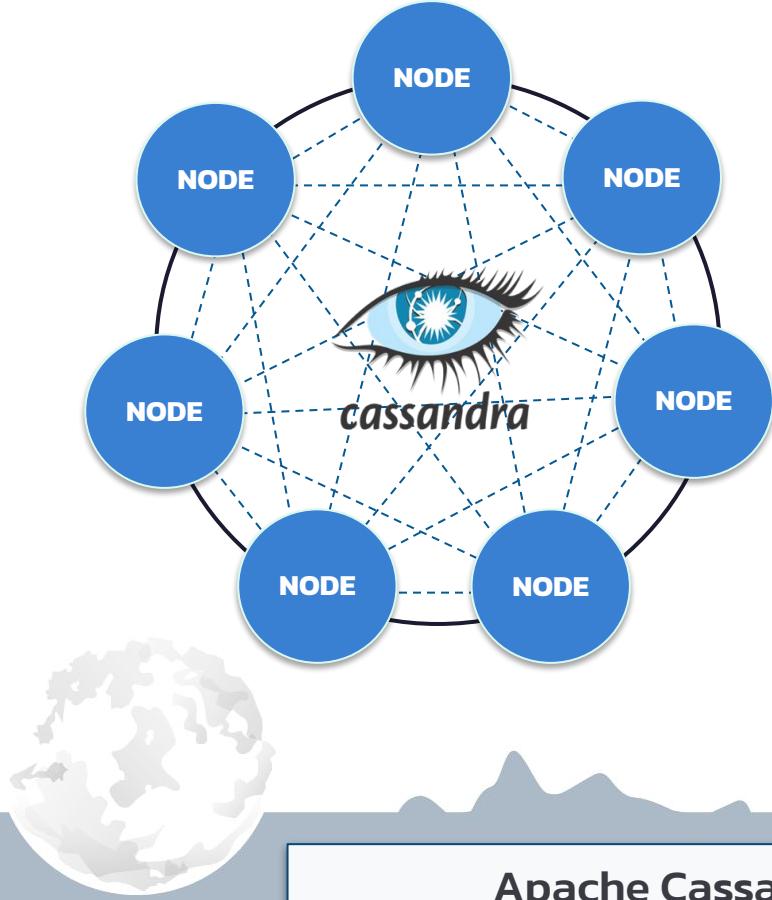
4821 p	spanda
4820 p	Agent X9
4775 p	fastest
4775 p	Sam
4711 p	CCedrickThePresenter
4468 p	shubham
4371 p	aaa
3895 p	vignesh
3877 p	adry
3861 p	Millie
3812 p	Puggie

2.11.07 / 2.26.05



Play with us with [Menti.com](http://Menti.com) (new TAB)

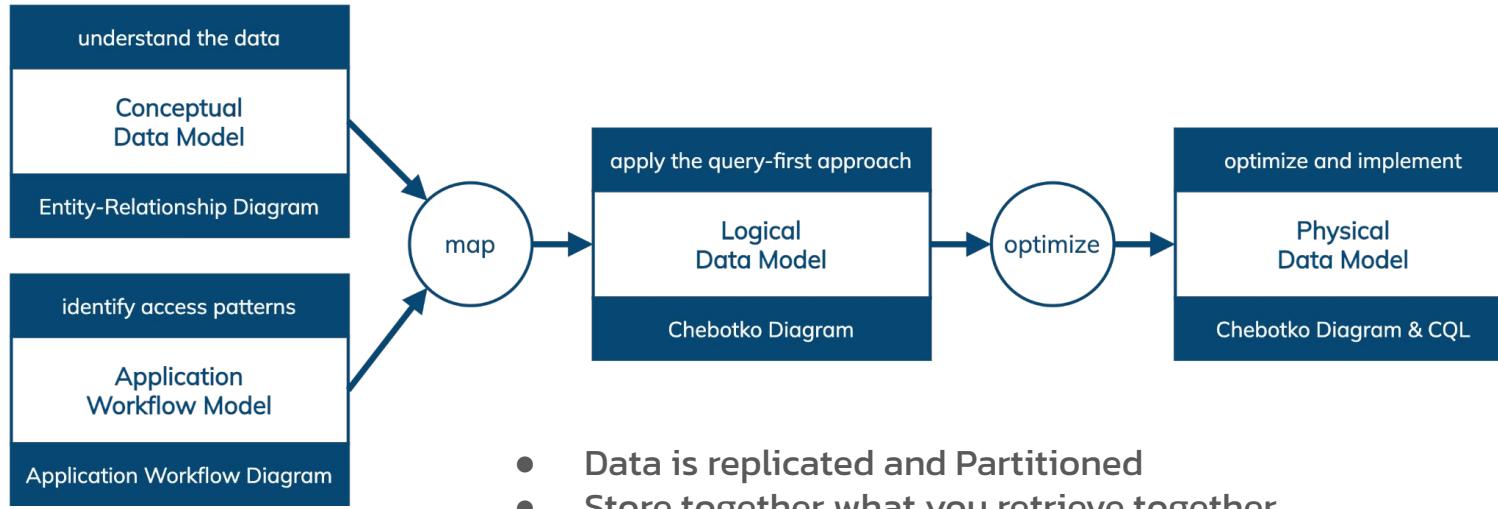
# Introduction to Apache Cassandra



- Big Data Ready
- Read / Write Performance
- Linear Scalability
- Highest Availability
- Self-Healing and Automation
- Geographical Distribution
- Platform Agnostic
- Vendor Independent

Apache Cassandra's Awesomeness

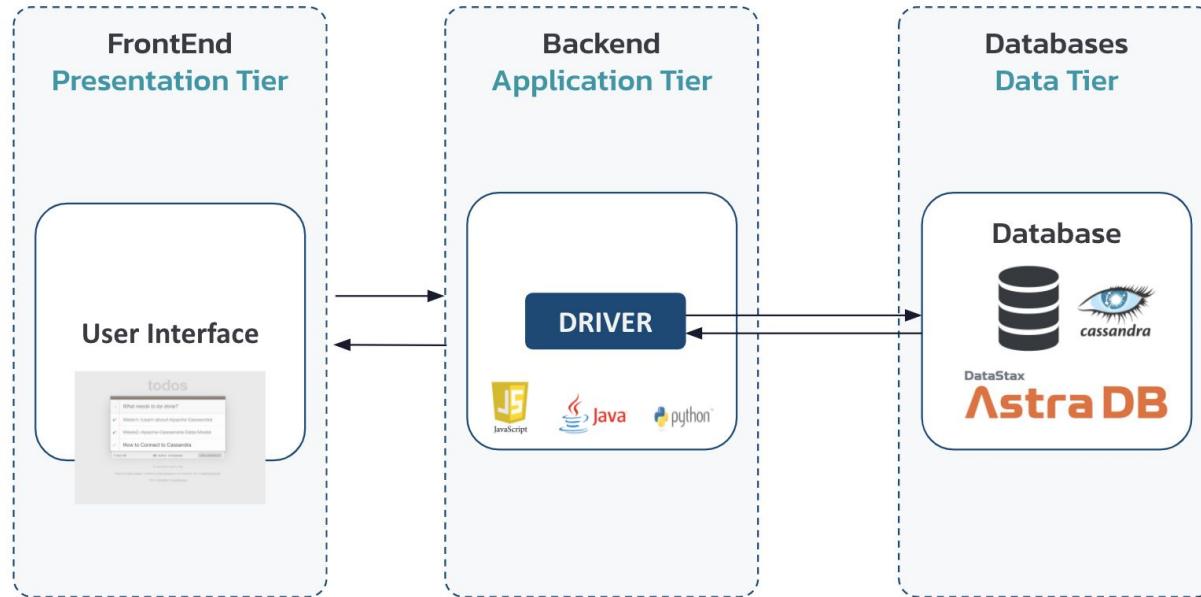
# Data Modeling with Apache Cassandra



- Data is replicated and Partitioned
- Store together what you retrieve together
- Avoid big or hot partitions

Cassandra Data Modeling Methodology

# Application Development with Apache Cassandra



High-level Application Development w/ Cassandra



Choose your language!





And today's badge for your pleasure

# 01



Introduction

# 02

Microservices  
Why with Apache Cassandra ?

# 03

Rest APIs

# 04

Sensor Application  
Drivers; via command line

# 05

Sensor Rest API  
FastAPI, SpringBoot

# 06

What's next?  
Quiz, Homework, Next week



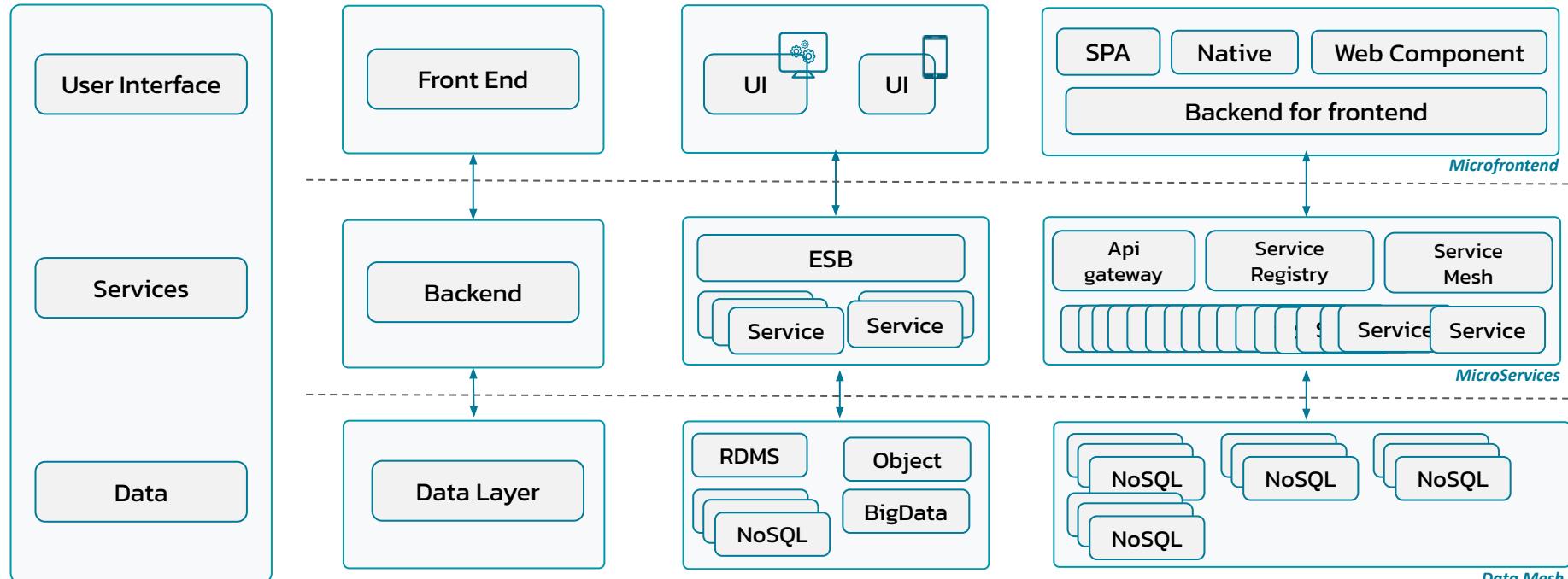
Agenda

Monolith 90'

Multi Tiers 2000

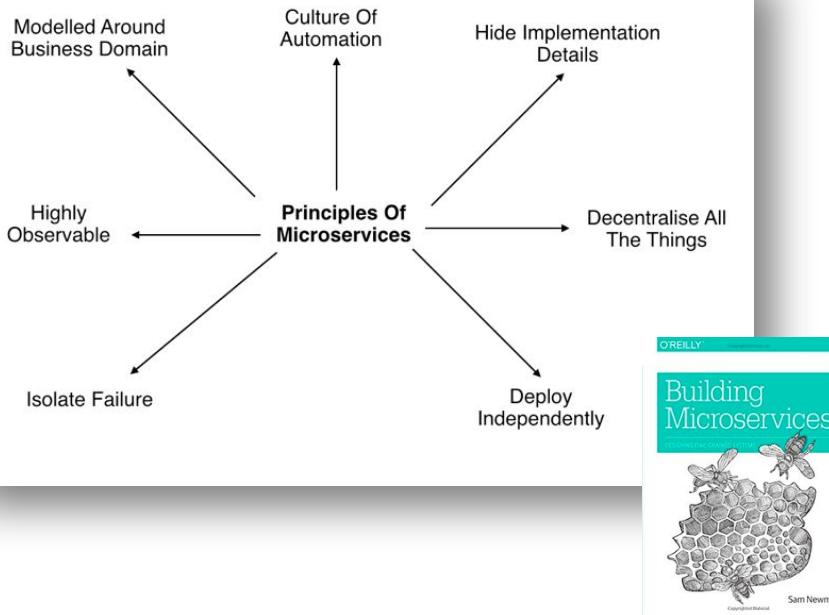
SOA (2005)

Microservices (2015)



To the Microservices Architecture





- ✓ Organized around Business Capabilities
- ✓ Products not Projects
- ✓ Smart endpoints and dumb pipes
- ✓ Decentralized Governance
- ✓ Decentralized Data Management
- ✓ Infrastructure Automation
- ✓ Design for failure
- ✓ Evolutionary Design

Martin Fowler



## MicroServices Principles

## ADVANTAGES



- Reduce Cost (Scaling, Design)
- Reduce Risk (resilience)
- Increase Release Speed
- Enable Visibility (security, monitoring)

## DISADVANTAGES



- Complexity (Security, Transaction, Orchestration)
- Cultural Changes
- Bigger RUN footprint



**Microservices should not share anything**  
**You install a Database for each service**

**Do you ?**  
**Does anyone do that ?**

**No. So WHY ?**



Atomicity

Consistency

Isolation

Durability

Basic

Availability

Soft State

Eventual Consistency

 No distributed transactions across services

Service A  
Order Management



Service B  
Customers



 AP System with Idempotence

 Event Sourcing, CQRS



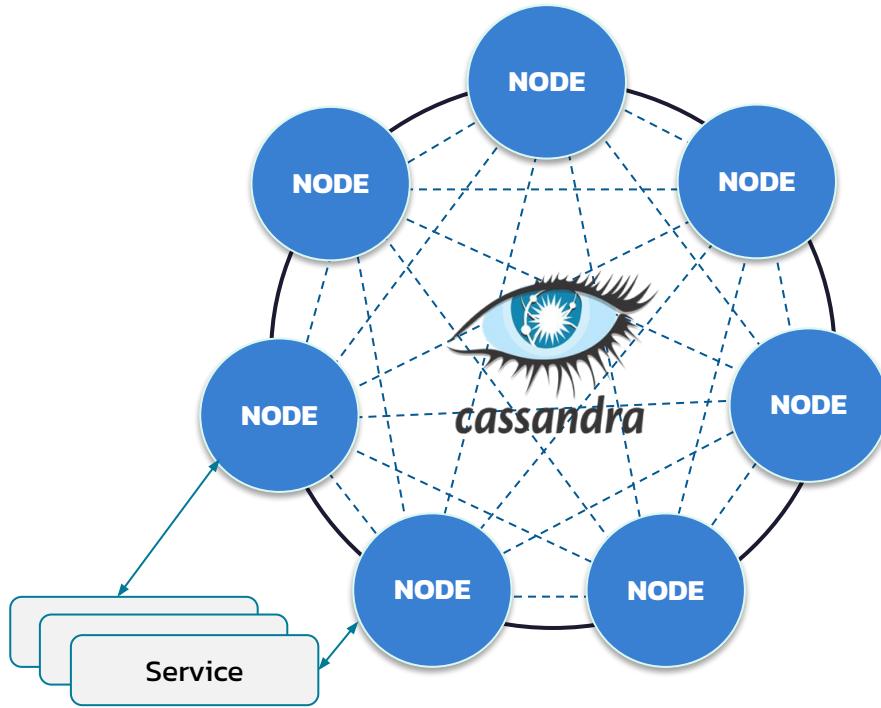
From "ACID" to "BASE"

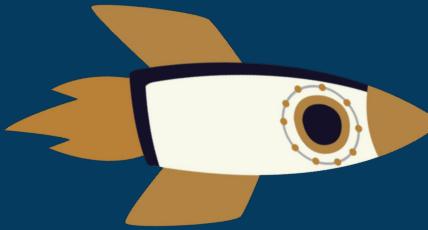
## Loose Coupling: Data resiliency

- Data Replicated on multiple Nodes
- Load Balancing at driver side
- Health Check at driver side
- Hinted HandOff

## Shared nothing: Data isolation

- Per Keyspace (with replications)
- Per Tables (1 query = 1 table)
- Per profile (RBAC)





# Hands-on (!github)

## #1 Database Setup

- ✓ Resume the database if "*hibernated*"
- ✓ Create Database if you have not
- ✓ Create Schema & Populate DB

# 01



Introduction

# 02

Microservices  
Why with Apache Cassandra ?

# 03

Rest APIs

# 04

Sensor Application  
Drivers; via command line

# 05

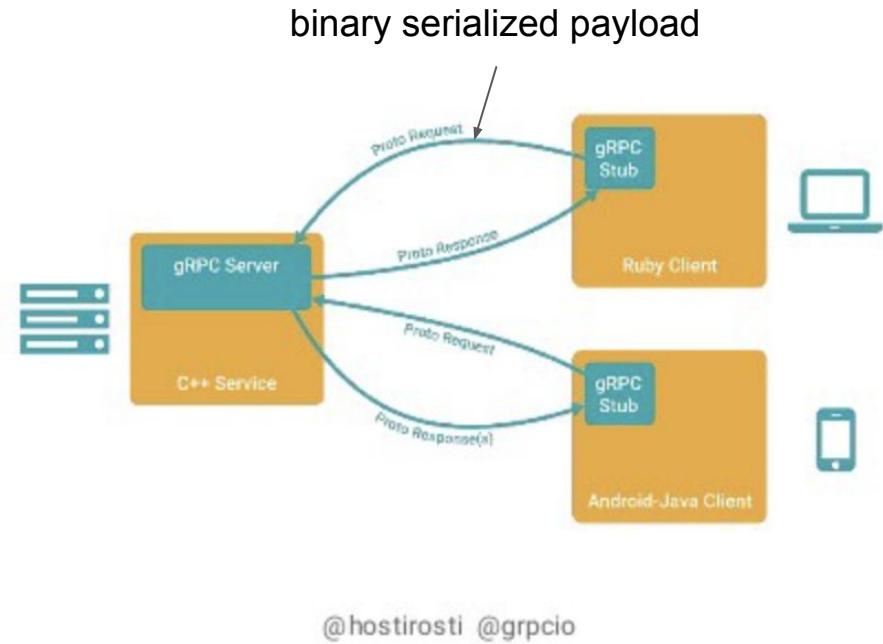
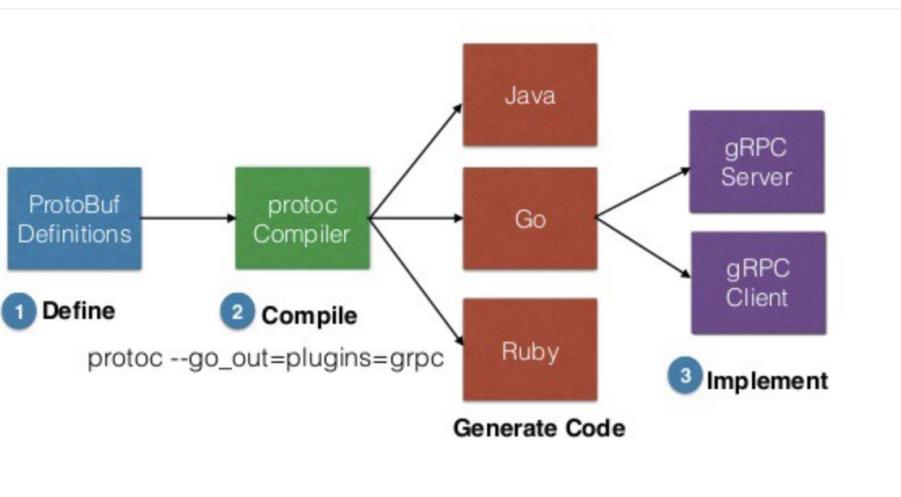
Sensor Rest API  
FastAPI, SpringBoot

# 06

What's next?  
Quiz, Homework, Next week



Agenda



Google Remote Procedure Call



## Describe your Data

```
type Workshop {  
    title: String!  
    abstract: String  
    speakers: [Speaker]  
    releaseYear: int  
}
```

## Ask what you need

```
incomingWorkshops {  
    title  
    abstract  
}
```

## Discoverability (demo)

The screenshot shows a GraphQL playground interface with a query editor and a results panel. The query is:

```
query {  
    incomingWorkshops {  
        title  
        abstract  
    }  
}
```

The results panel displays a nested JSON structure representing the workshop data. It includes fields like 'title', 'abstract', and lists of 'Speaker' objects. A red box highlights a specific part of the JSON output.



GraphQL



# { REST }

## Todos

Implement CRUD operations for Todo Tasks

**GET**

`/api/v1/todos/` Retrieve the complete list of Taskss

**POST**

`/api/v1/todos/` Create a new task

**DELETE**

`/api/v1/todos/` Delete all tasks in one go

**GET**

`/api/v1/todos/{taskId}` Get details of a task if exists

**DELETE**

`/api/v1/todos/{taskId}` Delete a task from its id if exists

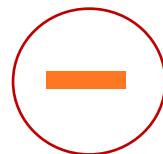
**PATCH**

`/api/v1/todos/{taskId}` Update an existing task

# { REST }



- Decoupling Client / Server (*Schema on read*)
- Api Lifecycle (*Versioning*)
- Tooling (*API Management, Serverless*)



- Verbose payloads (*json, xml*)
- No discoverability
- Not suitable for command-like (functions) API



- CRUD superstar
- Relevant for mutations (OLTP)
- Public and web APIs
- Limited Business Scope



Rest vs gRPC vs GraphQL ?

# 01



Introduction

# 02

Microservices  
Why with Apache Cassandra ?

# 03

Rest APIs

# 04

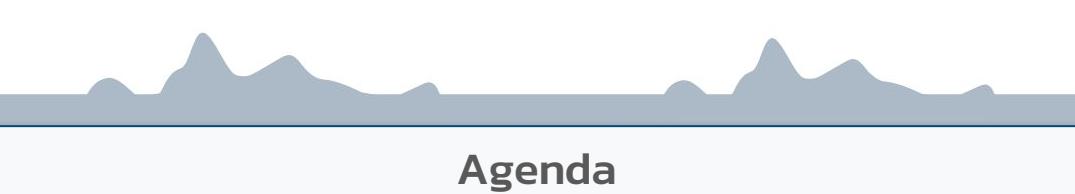
Sensor Application  
Drivers; via command line

# 05

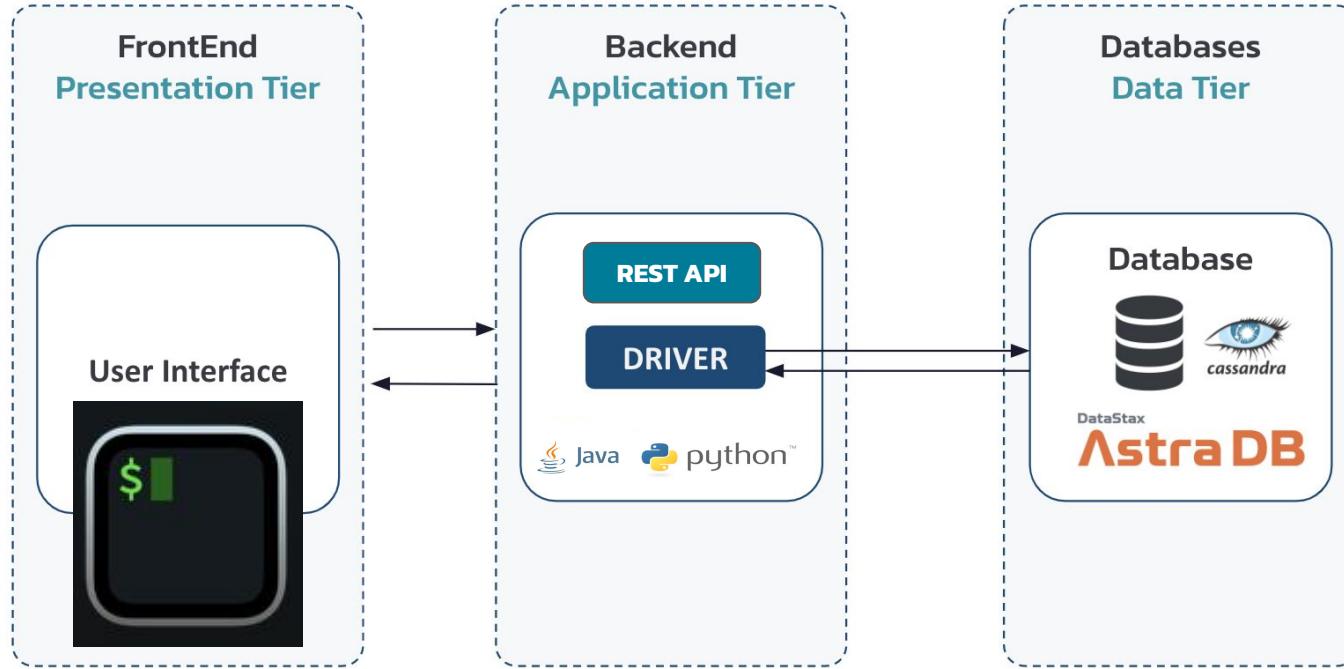
Sensor Rest API  
FastAPI, SpringBoot

# 06

What's next?  
Quiz, Homework, Next week



Agenda



Architecture of our Microservice Layer



### Connectivity

- ★ Token & Datacenter Aware
- ★ Load Balancing Policies
- ★ Retry Policies
- ★ Reconnection Policies
- ★ Connection Pooling
- ★ Health Checks
- ★ Authentication | Authorization
- ★ SSL

### Query

- ★ CQL Support
- ★ Schema Management
- ★ Sync/Async/Reactive API
- ★ Query Builder
- ★ Compression
- ★ Paging

### Parsing Results

- ★ Lazy Load
- ★ Object Mapper
- ★ Spring Support
- ★ Paging



Drivers



```
<dependency>
```

```
  <groupId>com.datastax.oss</groupId>
  <artifactId>java-driver-core</artifactId>
  <version>4.13.1</version>
</dependency>
```



```
pip install cassandra-driver
```



```
npm install cassandra-driver
```

4.6.3



```
{
  "dependencies": {
    "cassandra-driver": "^4.6.3"
  }
}
```



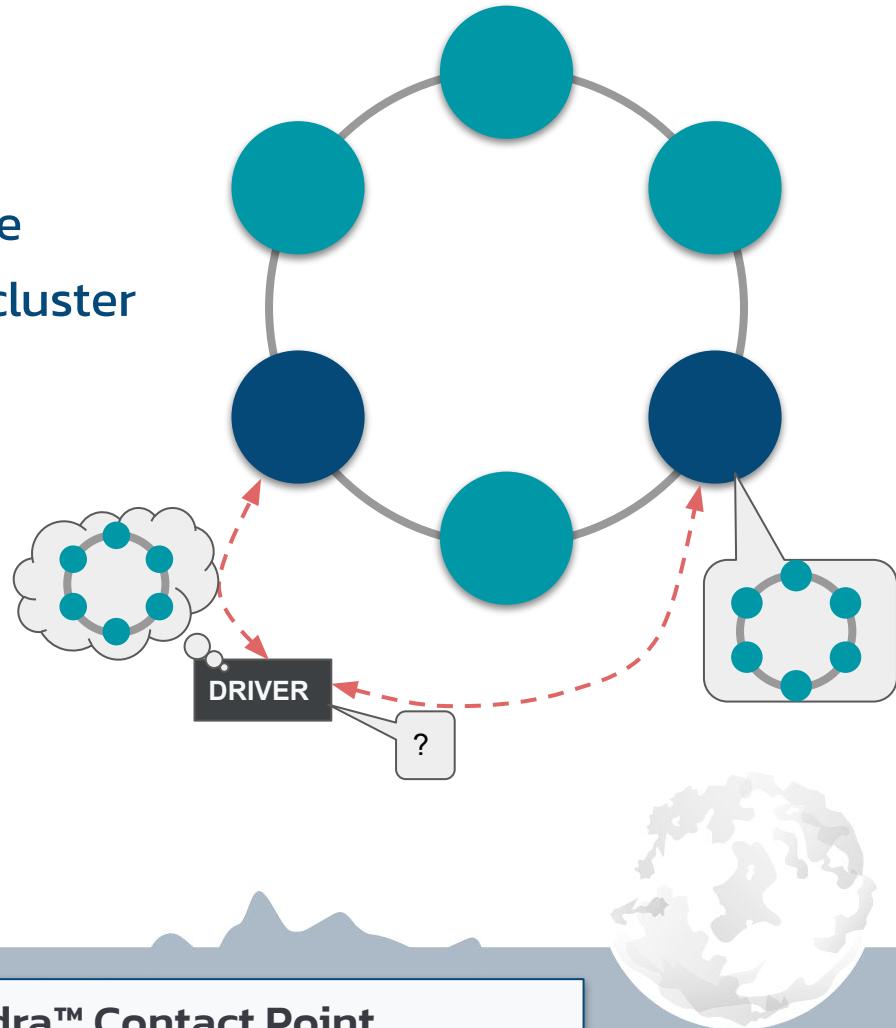
nuget v3.15.0

```
Install-Package CassandraCSharpDriver -Version 3.15.0
```



Installing the drivers

- Only one necessary
- Unless that node is down
- Better ~3 nodes per DC for resilience
- From there, drivers discover whole cluster
- Local Datacenter



Apache Cassandra™ Contact Point

```
CqlSession cqlSession = CqlSession.builder()  
    .addContactPoint(new InetSocketAddress("127.0.0.1", 9042))  
    .withKeyspace("killrvideo")  
    .withLocalDatacenter("dc1")  
    .withAuthCredentials("U", "P")  
    .build();
```



```
auth_provider = PlainTextAuthProvider(  
    username='U', password='P')  
  
cluster = Cluster(['127.0.0.1'],  
    auth_provider=auth_provider, protocol_version=2)  
  
session = cluster.connect('todos')
```



```
const client = new cassandra.Client({  
    contactPoints: ['127.0.0.1'],  
    localDataCenter: 'dc1',  
    keyspace: 'killrvideo',  
    credentials: { username: 'U', password: 'P' }  
});
```

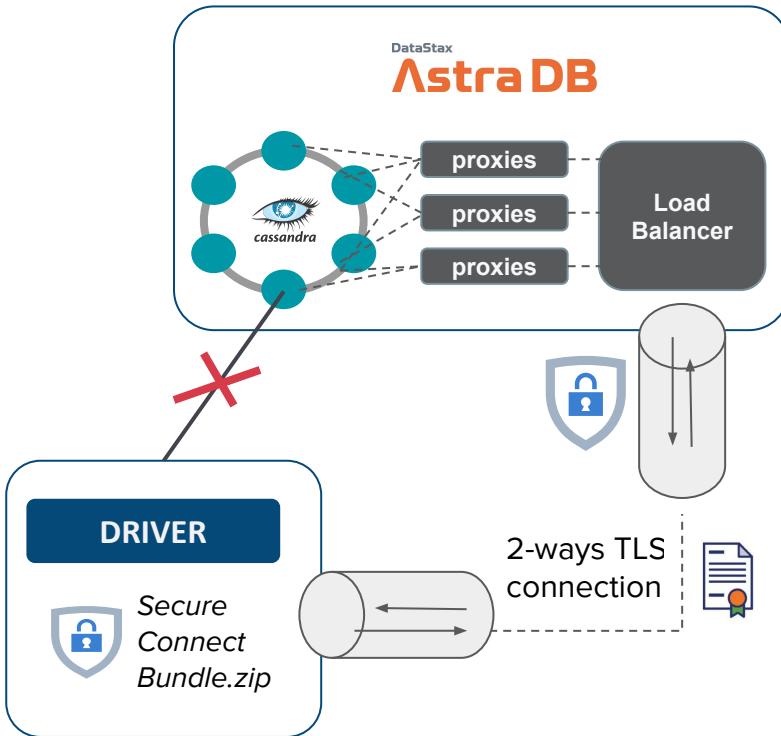


```
Cluster cluster = Cluster.Builder()  
    .AddContactPoint("127.0.0.1")  
    .WithCredentials("U", "P")  
    .Build();  
  
session = cluster.Connect("todos");
```



## Create Session/Client with Contact Points

- Secured connection over HTTP required.
- A *secureConnectBundle* ZIP is expected
- Your username is `clientId`
- Your Password is `clientSecret`
- No Single Point of failure (SPOF)



Contact Points with ASTRA

```
CqlSession cqlSession = CqlSession.builder()  
    .withCloudSecureConnectBundle(Paths.get("secure.zip"))  
    .withAuthCredentials("U", "P")  
    .withKeyspace("todos")  
    .build();
```



```
auth_provider = PlainTextAuthProvider(  
    username='U', password='P')  
  
cluster = Cluster(  
    Cloud ={ Secure_connect_bundle: 'secure.zip'},  
    auth_provider=auth_provider, protocol_version=2)  
session= cluster.connect('todos')
```



```
const client = new cassandra.Client({  
    cloud: { secureConnectBundle: 'secure.zip' },  
    credentials: { username: 'u', password: 'p' }  
});
```



```
var cluster = Cluster.Builder()  
    .WithCloudSecureConnectionBundle("secure.zip")  
    .WithCredentials("u", "p")  
    .Build();  
  
var session = cluster.Connect("todos");
```



- **Stateful** object handling communications with each node
- Should be unique in the Application (*Singleton*)
- Should be closed at application shutdown (*shutdown hook*) in order to free opened TCP sockets (*stateful*)

```
Java:      cqlSession.close();
```

```
Python:     session.shutdown();
```

```
Node:      client.shutdown();
```

```
CSharp:    IDisposable
```

# Executing CQL Queries



```
session.execute(  
    "SELECT * FROM sensors_by_network WHERE network = %s;",  
    (network,) )
```



```
cqlSession.execute(  
    "SELECT * FROM sensors_by_network WHERE network = '" + network  
    + "'");
```



Running queries from code

# Prepared Statements



```
q3_statement = session.prepare(  
    "SELECT * FROM sensors_by_network WHERE network = ?;")  
rows = session.execute(q3_statement, (network,) )
```



Prepared Statements in Python



# Prepared Statements



```
PreparedStatement q3Prepared = session.prepare(  
    "SELECT * FROM sensors_by_network WHERE network = ?");  
BoundStatement q3Bound = q3Prepared.bind(network);  
ResultSet rs = session.execute(q3Bound);
```



Prepared Statements in Java

# Advantages of CQL Prepared Statements

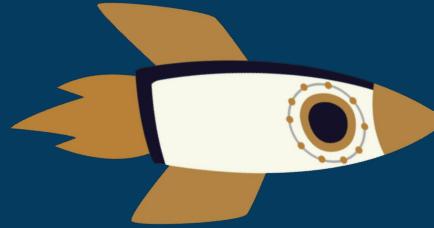


- Parse once, run many times.
- Saves network trips for result set metadata.
- Client-side type validation.
- Statements binding on partition keys compute their own cluster routing.



Advantages of Prepared Statements





# Hands-on (!github)

## #2 Sensor App

- ✓ Run existing Sensor Application
- ✓ Getting ready for Code !



# 01



Introduction

# 02

Microservices  
Why with Apache Cassandra ?

# 03

Rest APIs

# 04

Sensor Application  
Drivers; via command line

# 05

Sensor Rest API  
FastAPI, SpringBoot

# 06

What's next?  
Quiz, Homework, Next week



Agenda

Backend



**Spring Boot**

Spring MVC

JAVA Drivers

Backend



**FastAPI**

FastAPI

Python

PYTHON Drivers

## Technical Stacks Implementation

Backend



**Spring Boot**

SensorNetworkSpringApp

CassandraConnection

JAVA Drivers

Backend



**FastAPI**

api.py

db\_connection.py

PYTHON Drivers

# Sensor API

Two endpoints:

- Retrieve sensor data by network
- Retrieve measurements by date

GET	/sensors_by_network/{network}
POST	/measurements_by_sensor_date/

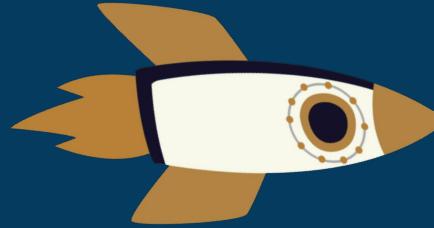
## Wait a minute...

## Why a POST?



Rest API Specifications





# Hands-on (!github)

## #3 Sensor API

- ✓ Run Restful Service
- ✓ Explore the endpoints



# 01



Introduction

# 02

Microservices  
Why with Apache Cassandra ?

# 03

Rest APIs

# 04

Sensor Application  
Drivers; via command line

# 05

Sensor Rest API  
FastAPI, SpringBoot

# 06

What's next?  
Quiz, Homework, Next week



Agenda



# Homework (!homework)



A couple of "theory" questions, plus ...

**Coding exercise:** *Enrich the API with a new  
GET endpoint for Q1 ("get all networks")*



# How to Get Help



Discord - [dtsx.io/discord](https://dtsx.io/discord)



stackoverflow - <https://stackoverflow.com/questions/tagged/cassandra>



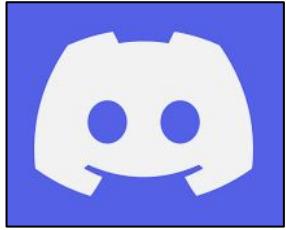
DBA Stack Exchange - <https://dba.stackexchange.com/questions/tagged/cassandra>

***Don't forget to "follow" the "cassandra" tag!***



Got a question? Get it answered!





DataStax Developers

# workshop-chat <https://www.youtube.com/watch?v=MuwT5xxFVVI> - Subscribe to mailing list: [http...](http://)

Événements  
moderator-only  
WELCOME  
start-here  
code-of-conduct  
introductions  
upcoming-events  
useful-resources  
memes  
your-ideas  
the-stage  
WORKSHOPS  
workshop-chat  
workshop-feedback  
workshop-materials  
upcoming-workshops  
ASTRADB  
getting-started  
astra-apis  
astra-development  
sample-applications  
APACHE CASSANDRA  
Cedrick Lun...

RIGGITYREKT Hier à 21:14  
I have a 5 node datacenter, 4 nodes are on dse version 5.1.20, one is on dse5.0.15. I am doing some mixed version testing for a class and the one node that is 5.0.15 is coming up as an analytics workload. I dont have /etc/default/dse, instead I am using /etc/init.d/dse-cassandra.  
how do I make that node start in cassandra workload, not in analytics?

RIGGITYREKT Hier à 23:39  
Okay I found out my issue, when i started DSE 5.0.15 it had endpointsnitch set to DseSimpleSnitch, the rest of my cluster is using PropertyFileSnitch, when i change it to PropertyFileSnitch, it still uses the simple snitch config. looking at the docs i see there is a way to go to GossipingPropertyFileSnitch, but i need the property file one. I can wipe this dbs, do anything with this node to get this done. how do i fix this?  
@here

19 novembre 2021  
Erick Ramirez Aujourd'hui à 02:19  
mixed versions isn't supported and you're guaranteed to run into weird issues that will cause further problems down the track

@RIGGITYREKT I have a 5 node datacenter, 4 nodes are on dse version 5.1.20, one is on dse5.0.15. I am doing some mixed v...  
Cedrick Lunen Aujourd'hui à 09:01  
When you start a node you have parameters -k for analytics, -g for graph and -s for search. To remove analytics check and remove -k

Envoyer un message dans #workshop-chat

PRESENTER — 1  
David Jones-Gilardi  
HELPER — 7  
012345  
AaronP  
Binary  
Chelsea Navo  
Jeremy Hanna  
John Sanda  
Patrick\_McFadin  
EN LIGNE — 560  
-samu-  
6304-42JB  
Aahlya  
Abdurahim  
abhi3pathi  
Abhiis.s  
Abhineet  
Abirsh



# !discord

[dtsx.io/discord](https://dtsx.io/discord)



DataStax Developers Discord (18k+)

# Subscribe



# Subscribe



How to create an Authentication Token in...

37 views • 4 weeks ago

How to use the Data Loader in Astra DB

62 views • 4 weeks ago

Astra DB Sample App Gallery

36 views • 4 weeks ago

How to use Secure Connect in Astra DB

42 views • 4 weeks ago

Cassandra Day India: CL Room (Workshops)

2.4K views • Streamed 4 weeks ago

Cassandra Day India: RF Room (Talks)

1.3K views • Streamed 1 month ago

# Thank You!

