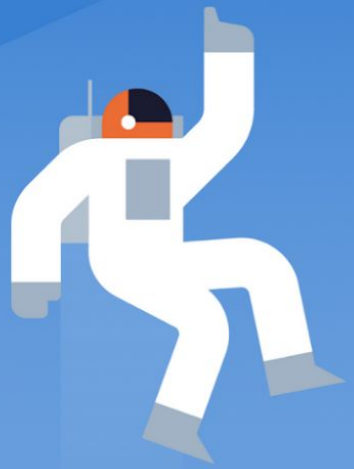
DataStax

Build an Appointment Scheduling application for a Petclinic



LEVEL UP with the DataStax Developers

Cédrick Lunven

Director Developer Relations

@clunven

- Trainer
- Public Speaker
- Developers Support
- Developer Applications
- Developer Tooling

- Creator of ff4j (ff4j.org)
- Maintainer for 8 years+

- Happy developer for 14 years
- Spring Petclinic Reactive & Starters
- Implementing APIs for 8 years

DataStax Developers

David Gilardi
Senior Developer Advocate

TO REVIEW BY DAVID

- Training
- Teaching
- Reference Applications
- Support
- Public Speaking

- Certified !

- Advanced User

Developer Advocate, Major and our HOST TODAY

@SonicDMG @SonicDMG

DataStax Developers

DatastaxDevs: Developer Advocates Team

Cedrick Lunven

Aleksandr Volochnev

Jack Fryer

Kirsten Hunter

Stefano Lottini

David Gilardi

Ryan Welford

Rags Srinivas

Sonia Siganporia

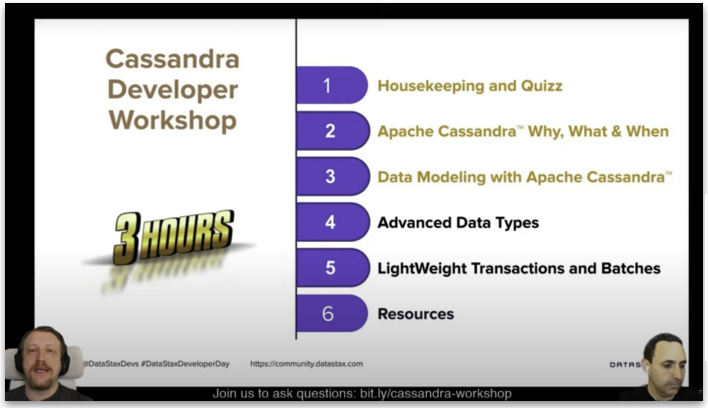
R

S

Housekeeping #1: Attending the workshop

Live and interactive

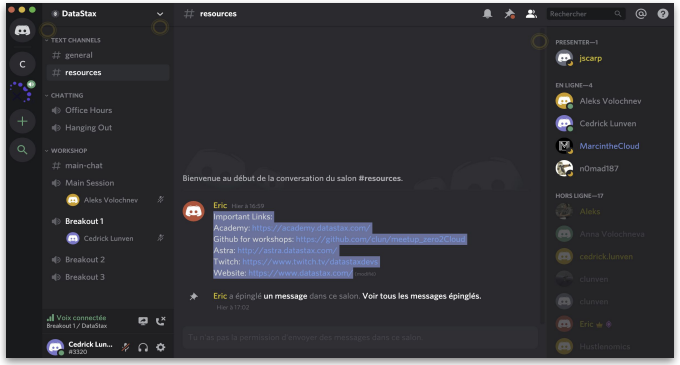
Livestream: youtube.com/DataStaxDevs



YouTube

Twitch

Questions: https://dtsx.io/discord

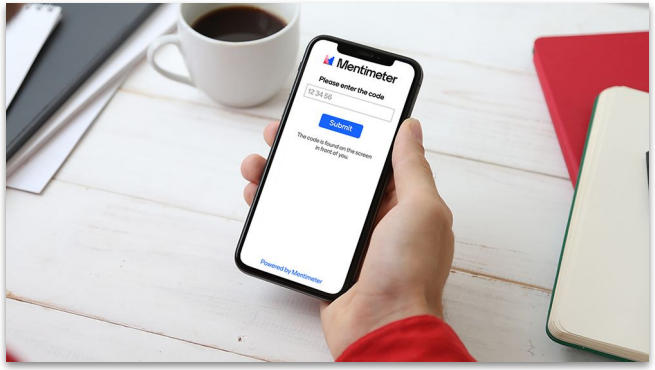


Discord

YouTube

Available on the iPhone App Store

Games menti.com



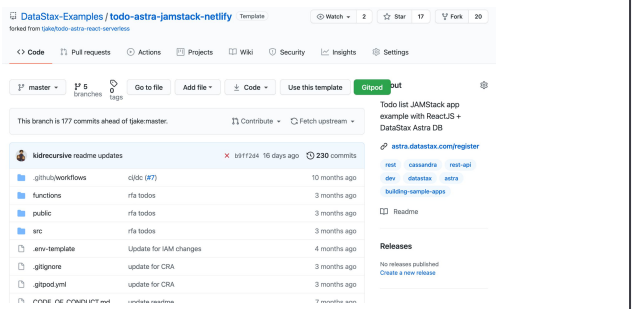
Mentimeter

GET IT ON Google play

Housekeeping #2: Doing Hands-On

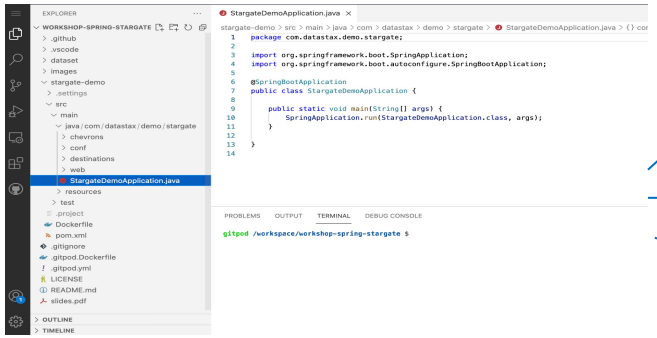
Nothing to install !

Source code + exercises + slides



GitHub

IDE



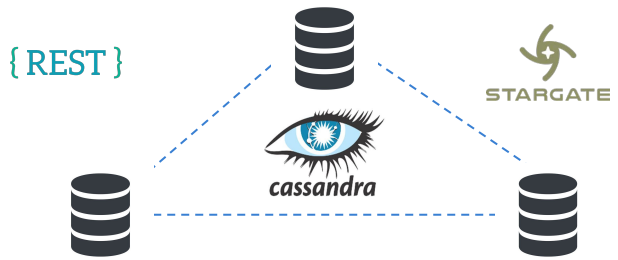
Gitpod

Maven

npm

node

Database + GraphQL + PlayGround



DataStax Astra DB

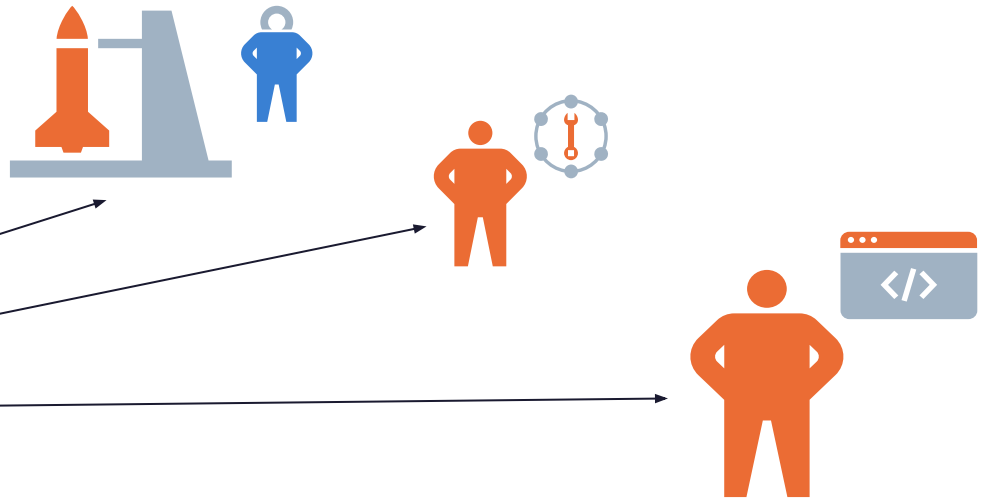
DataStax Developers

Get your Badge for today !

DataStax Developers

https://dtsx.io/staxgiving

500$ + 250$ + 250$ AMAZON gift cards to win



CONDITIONS

- #engage-with-us : Tasks that will earn you 1 point
- #work-with-us: Tasks that will earn you 5 points
- #build : Tasks that will earn you 10 points.

DUE DATE THANKSGIVING DAY => 25/11

Winners randomly chosen live on 12/8

DataStax Developers

Agenda

01

PetClinic
Architecture & Use Case

02

DataStax Astra

Cassandra Database
The Art of Data Modelling

03

Reactive Drivers
Reactive vs Async

04

Spring Reactive
Boot and Webflux

05

User Interface
Angular

06

Game &
Resources

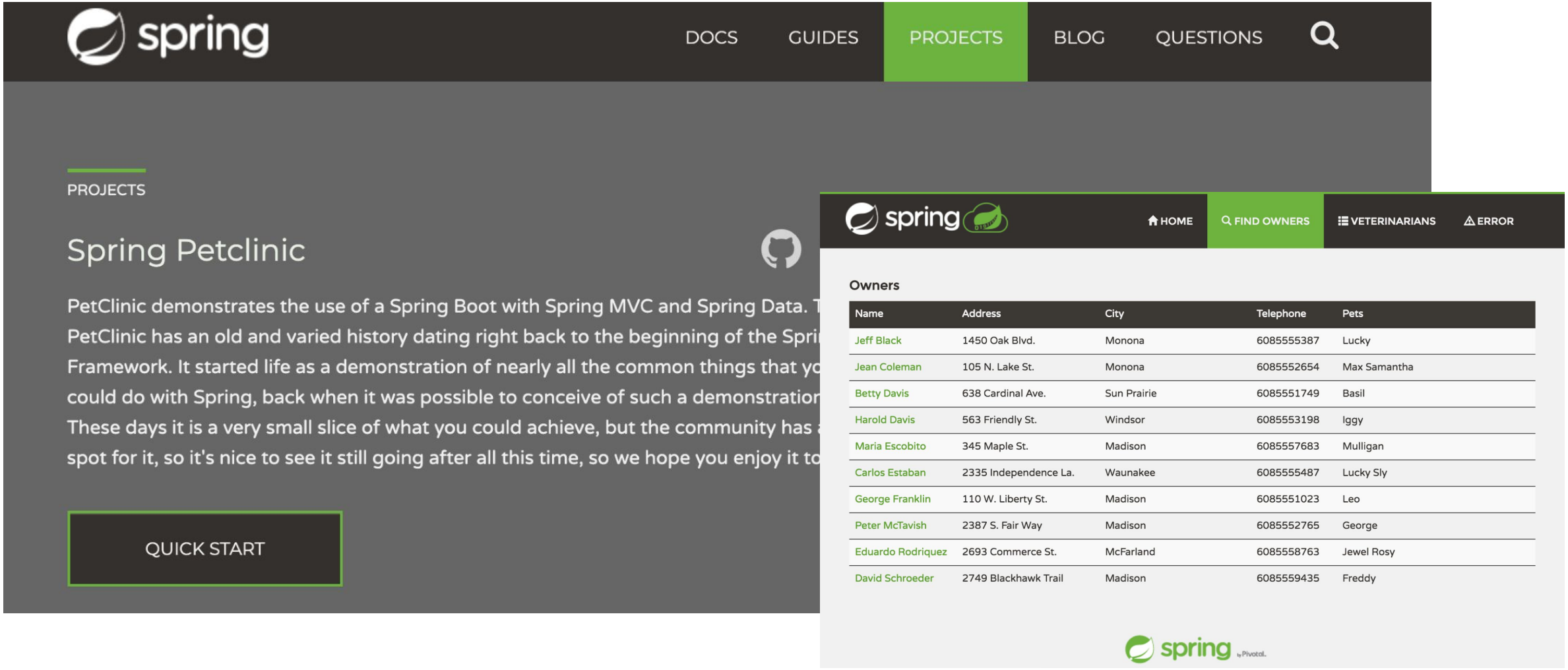
Welcome to PetClinic

Your mission is to
understand and run the
Spring reference application
with Cassandra as a backend
with Reactive Programming
with Angular user interface
and without installing anything

You have 2 hours
Good LUCK.

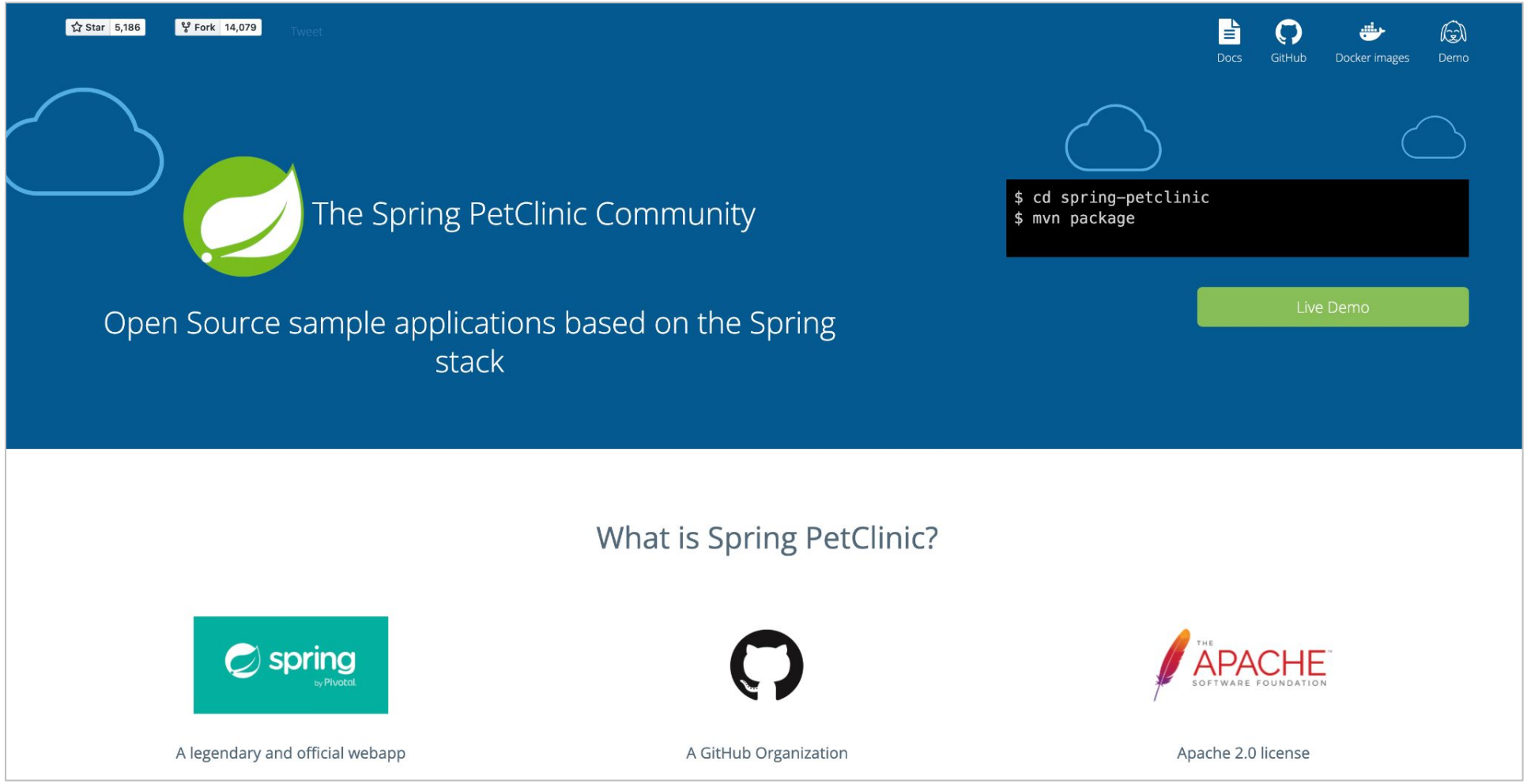
What is PetClinic ?

https://projects.spring.io/spring-petclinic/



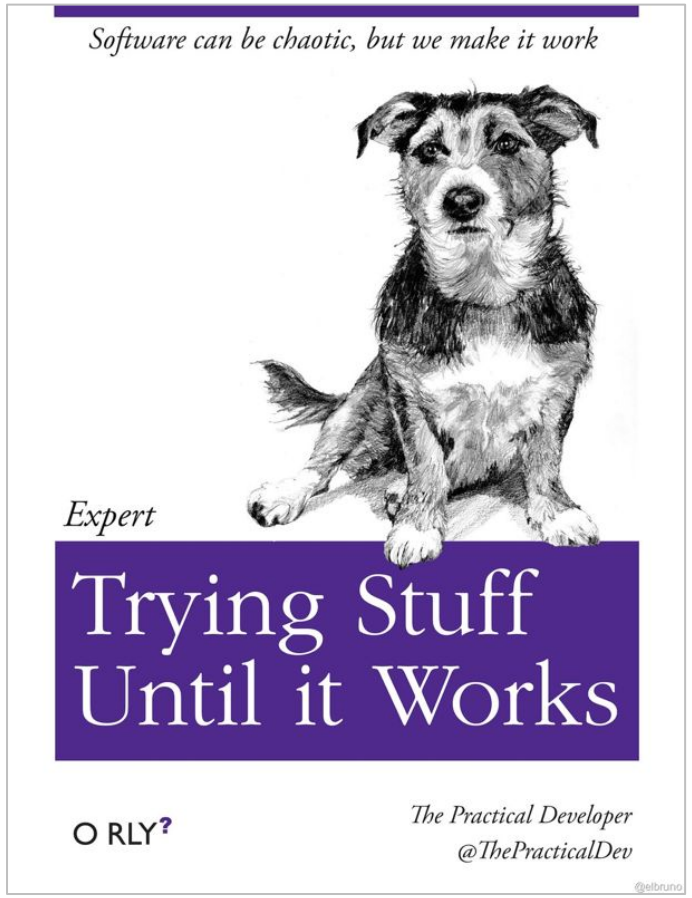
What is Spring Petclinic *Community ?*

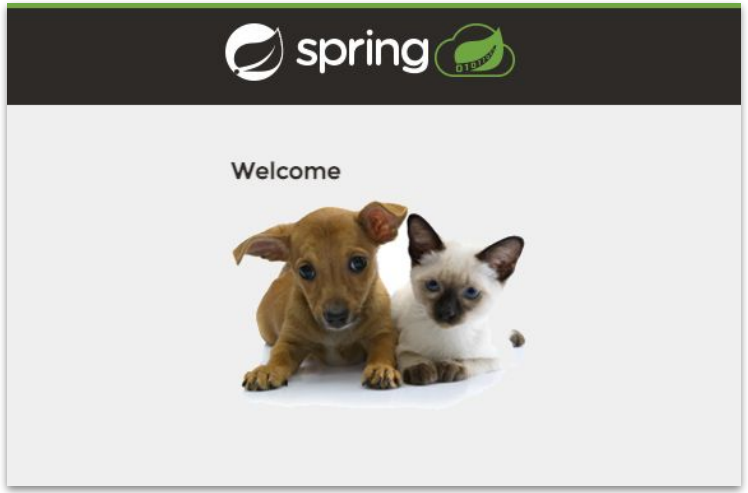
https://spring-petclinic.github.io/

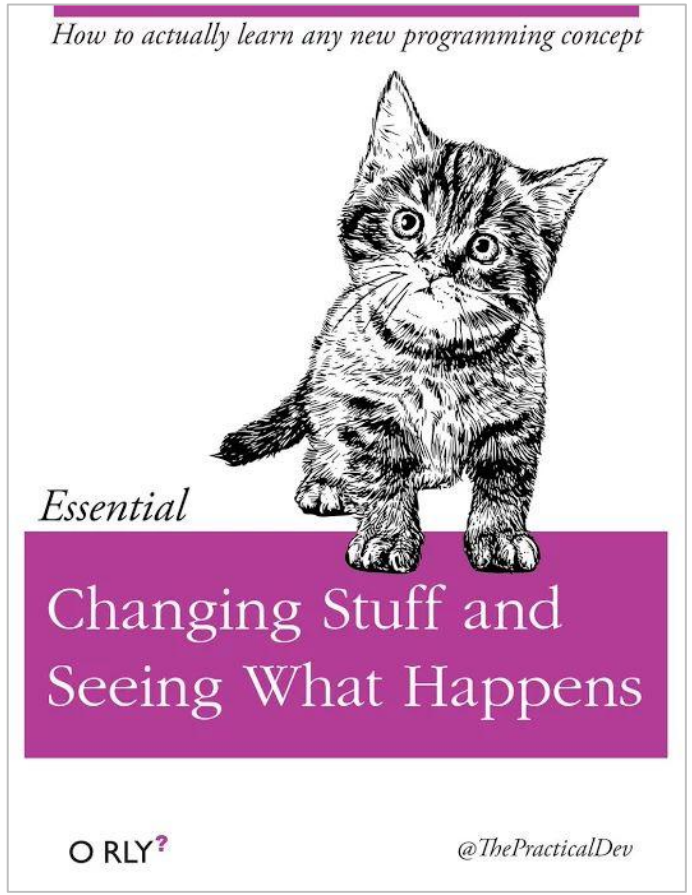


Spring Pet Clinic – Demo

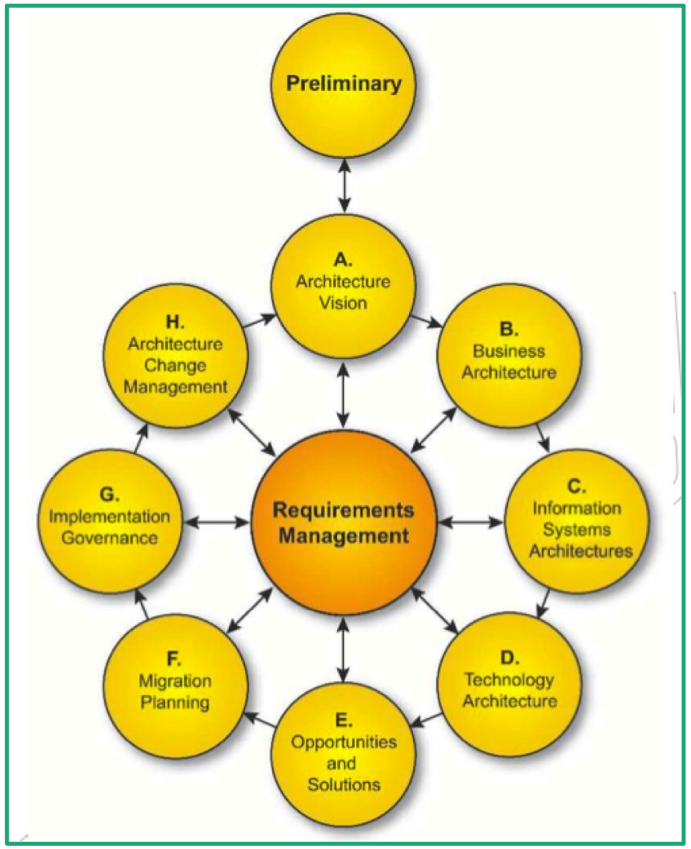
https://spring-petclinic-community.herokuapp.com/



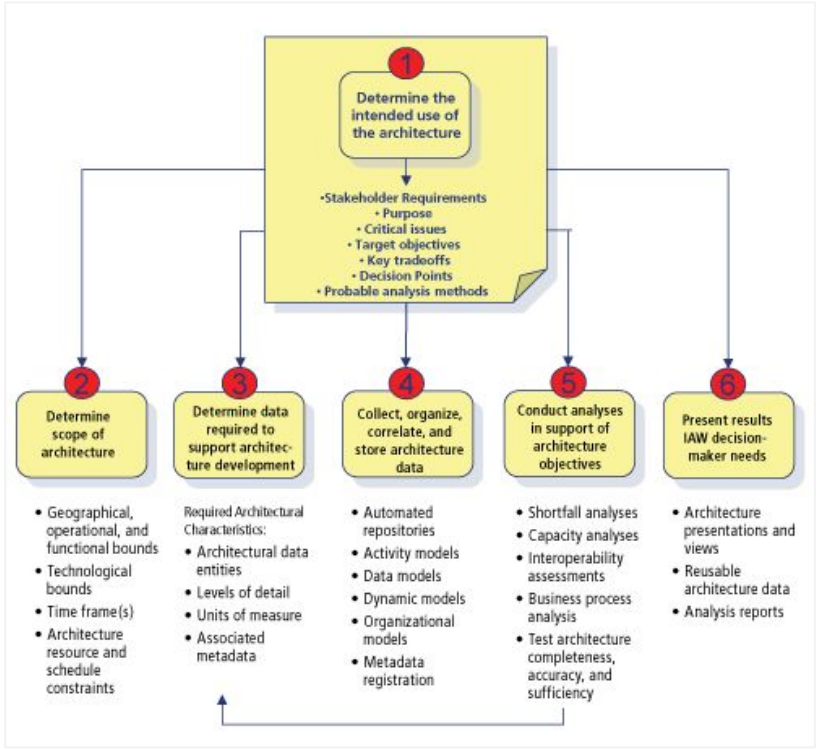




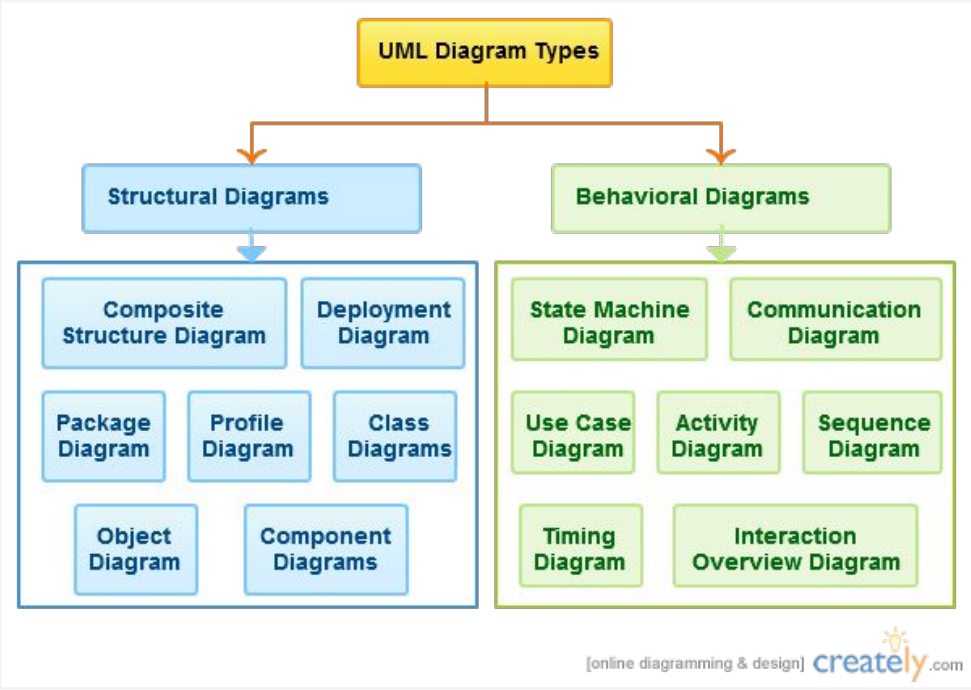
PetClinic :: Architecture



ToGAF (Archimate)



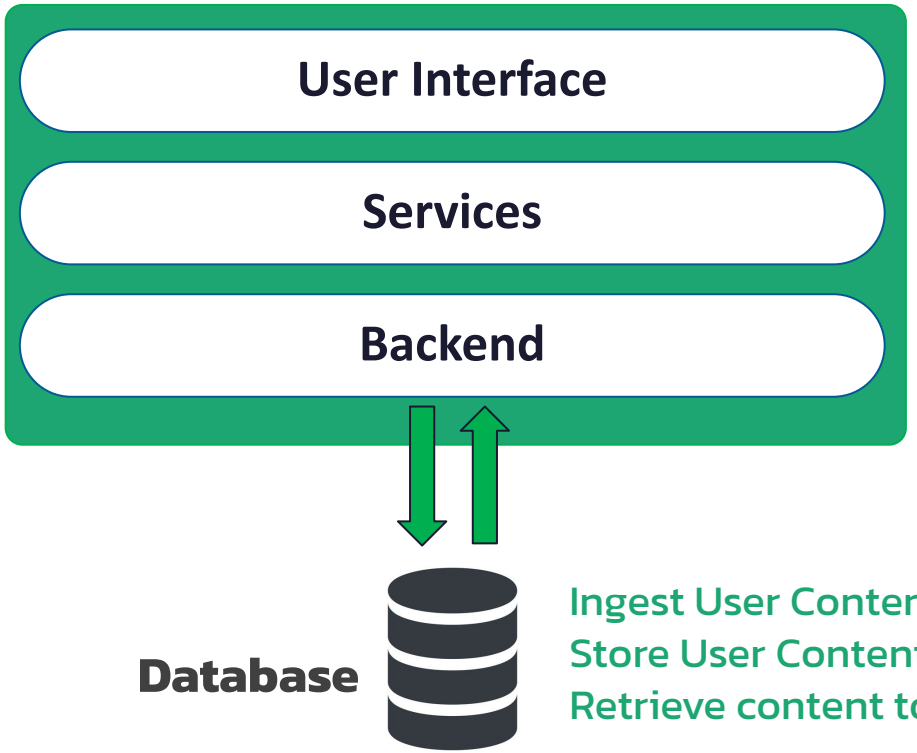
DoDAF (DM2)



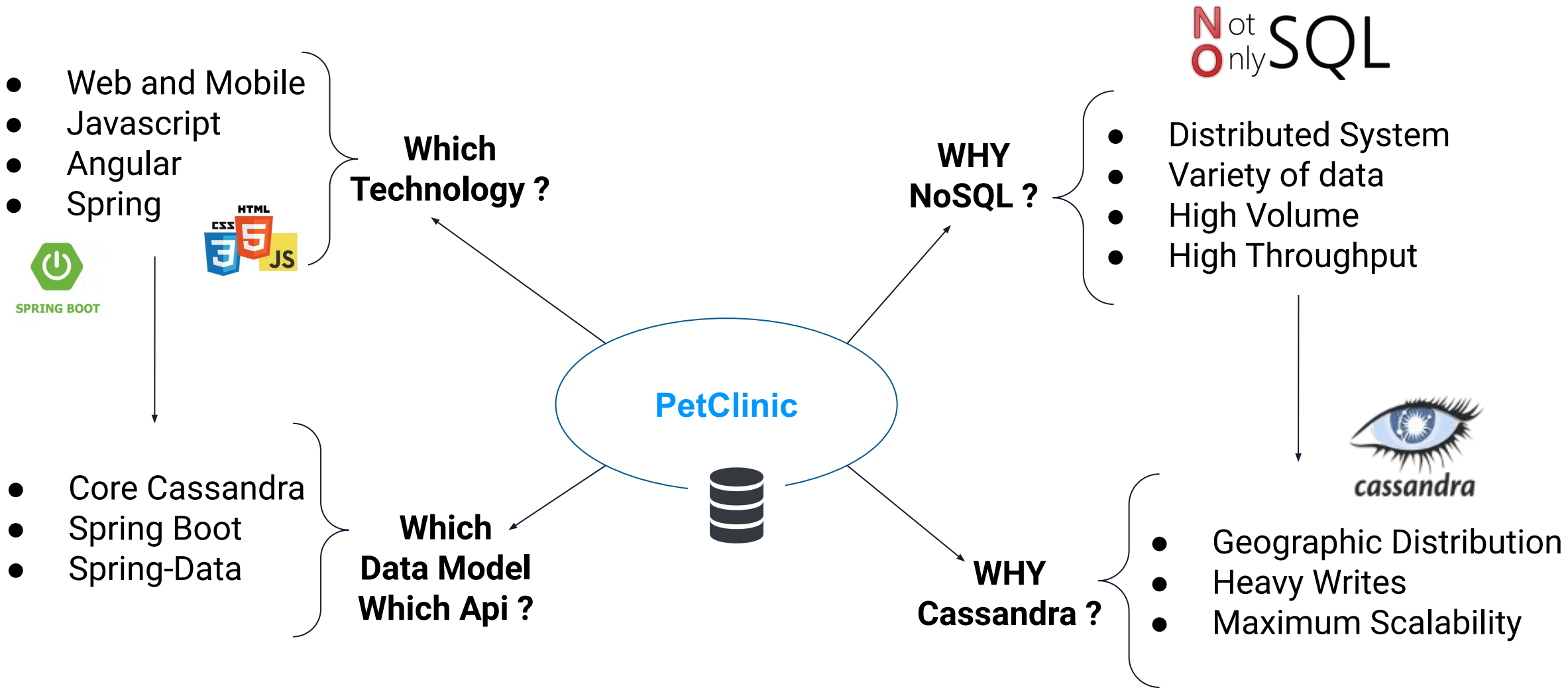
UML

Petclinic :: (A) Architecture Vision

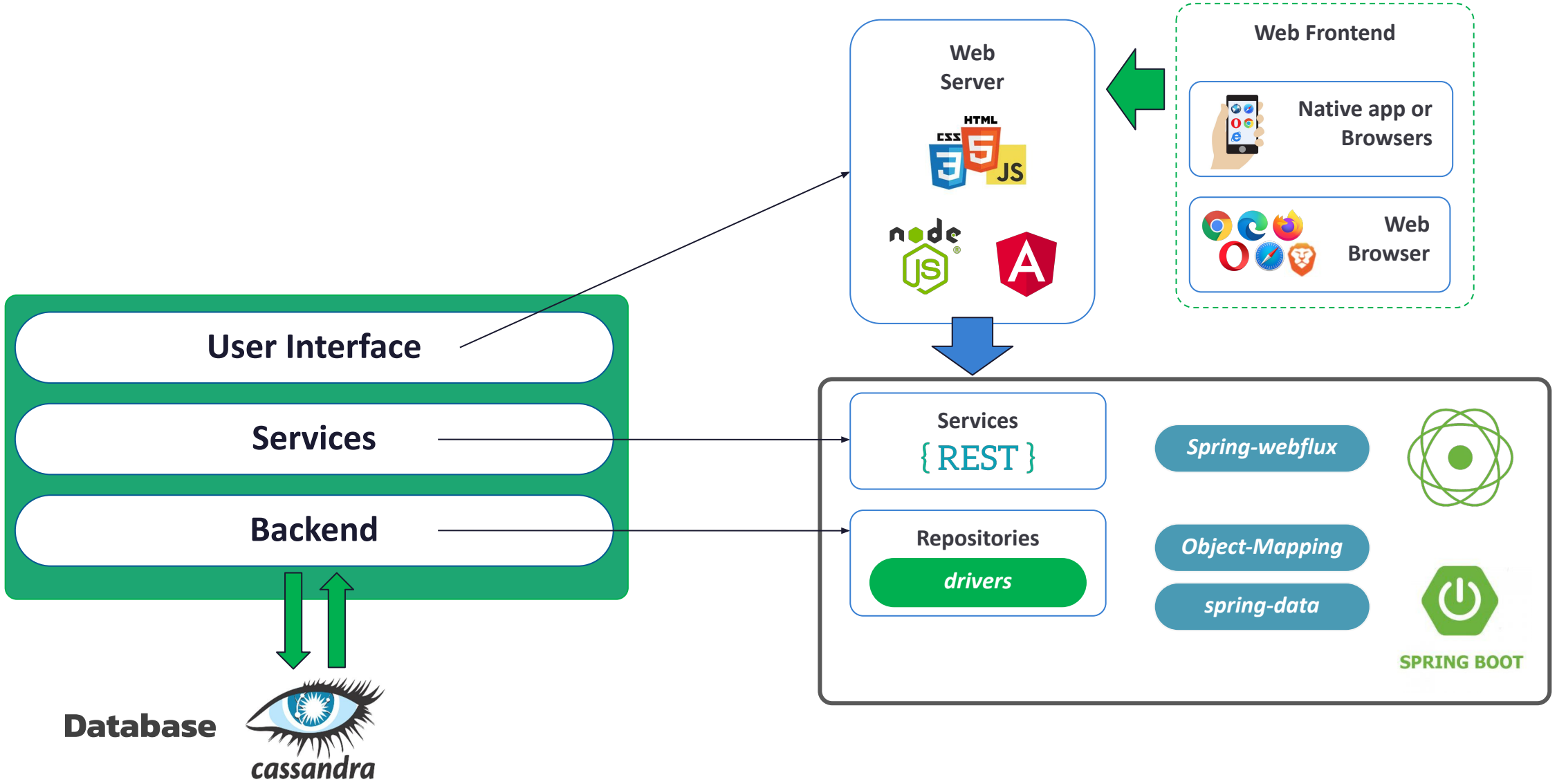
- **OLTP Create update delete operations** on multiple entities: Veterinarian, Owner, Pet, Visit with multiple **relationships**
- **Manage Reference lists :** Pet Family, Vet Specialties



Petclinic :: (D) Technology Architecture



Petclinic :: (E) Solution and Opportunity



Agenda

01

PetClinic
Architecture & Use Case

02

DataStax Astra

Cassandra Database
The Art of Data Modelling

03

Reactive Drivers
Reactive vs Async

04

Spring Reactive
Boot and Webflux

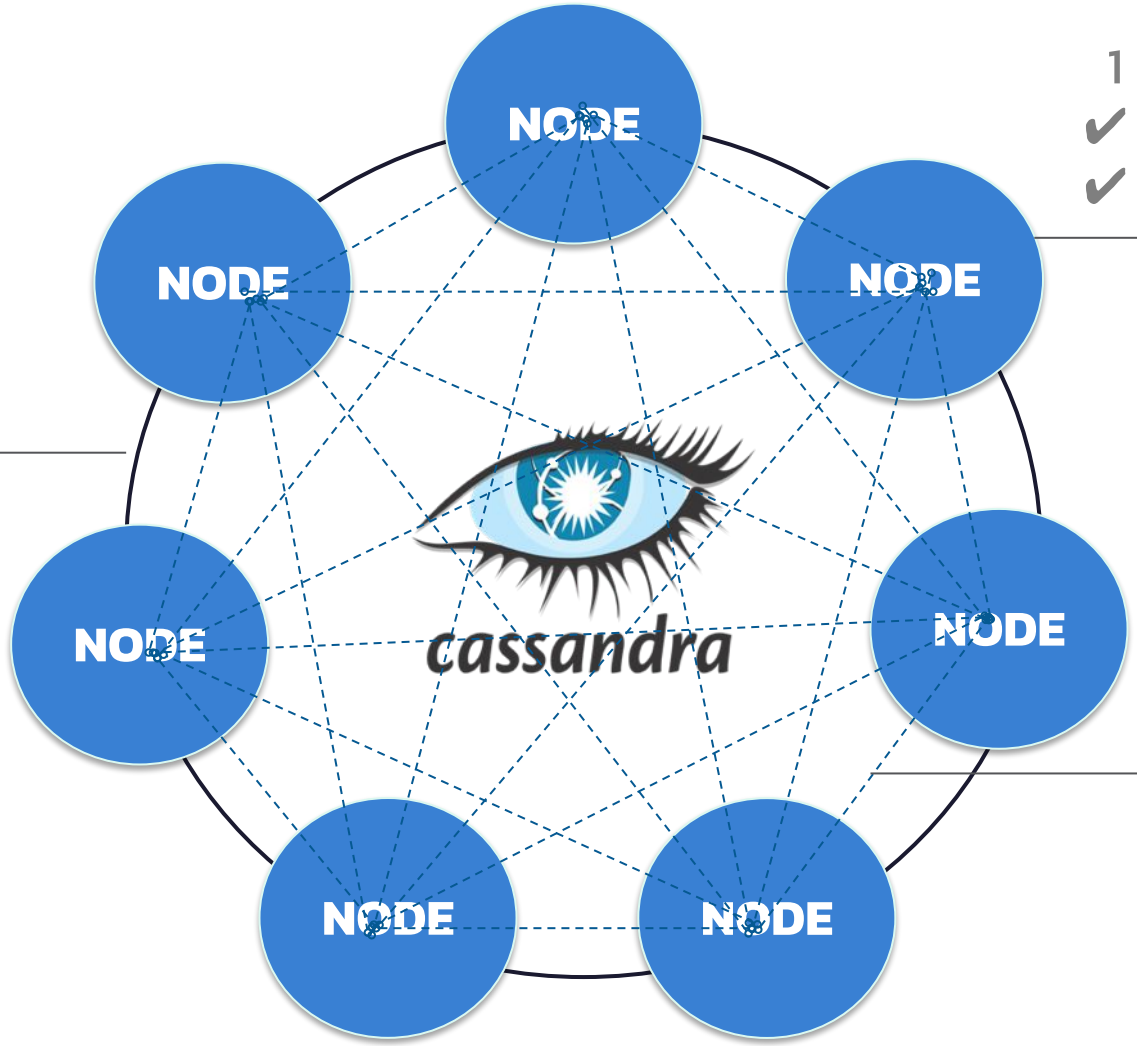
05

User Interface
Angular

06

Game &
Resources

Apache Cassandra™ = NoSQL Distributed Database



1 Installation = 1 NODE

✔ Capacity = ~ 2–4TB

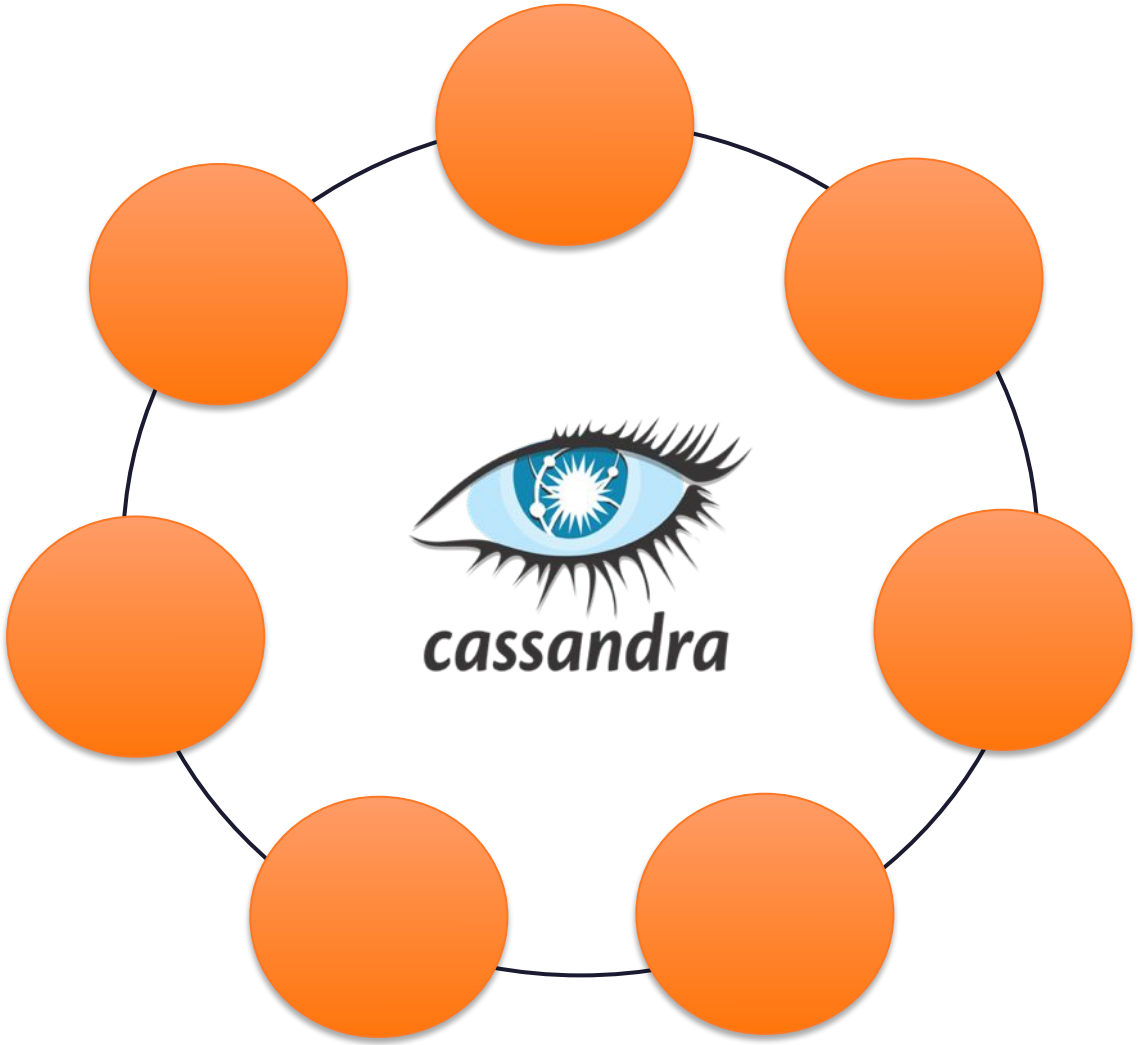
✔ Throughput = LOTS Tx/sec/core

DataCenter | Ring

Communication:

✔ Gossiping

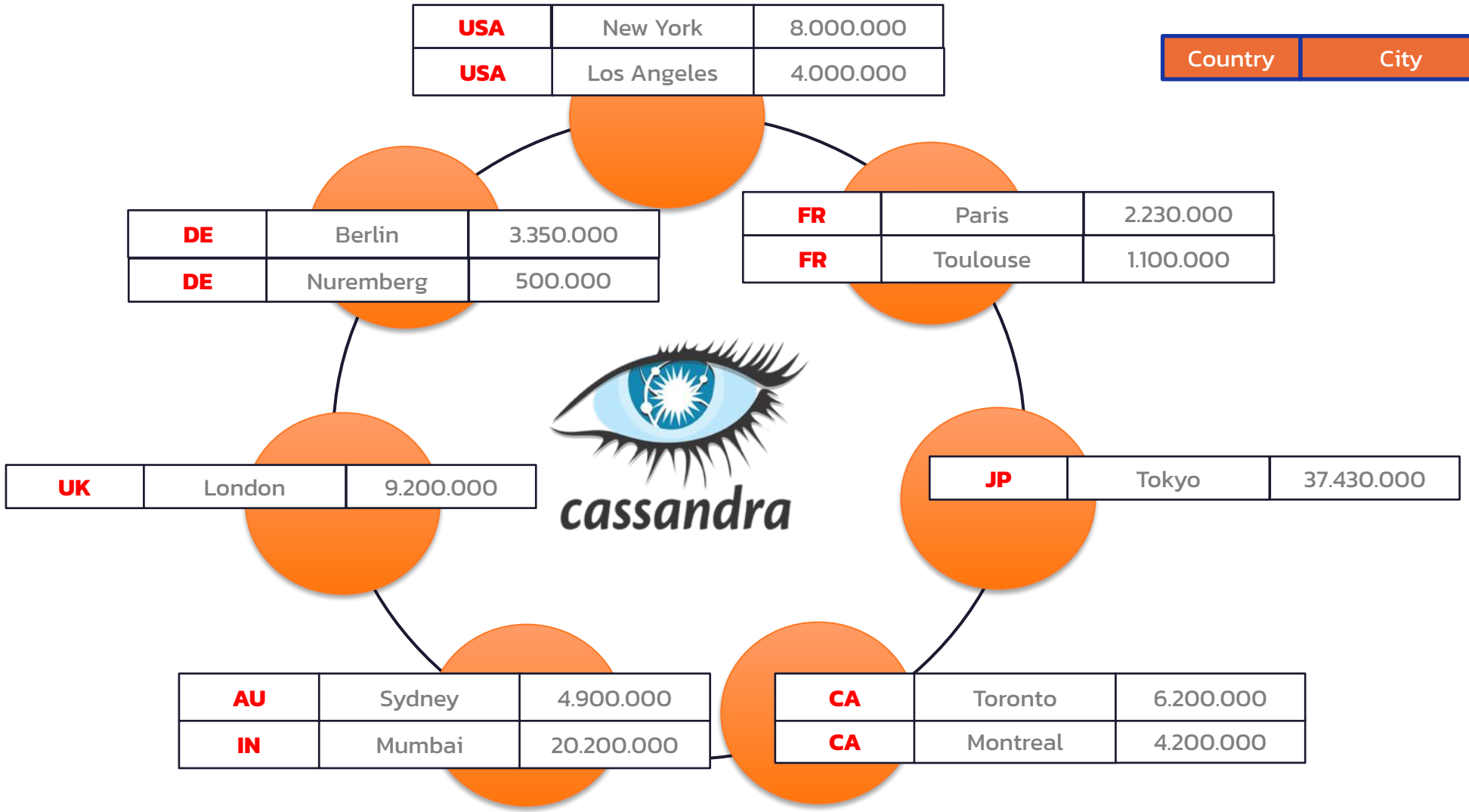
Apache Cassandra™ :: Data is Distributed



Country	City	Population
USA	New York	8.000.000
USA	Los Angeles	4.000.000
FR	Paris	2.230.000
DE	Berlin	3.350.000
UK	London	9.200.000
AU	Sydney	4.900.000
DE	Nuremberg	500.000
CA	Toronto	6.200.000
CA	Montreal	4.200.000
FR	Toulouse	1.100.000
JP	Tokyo	37.430.000
IN	Mumbai	20.200.000

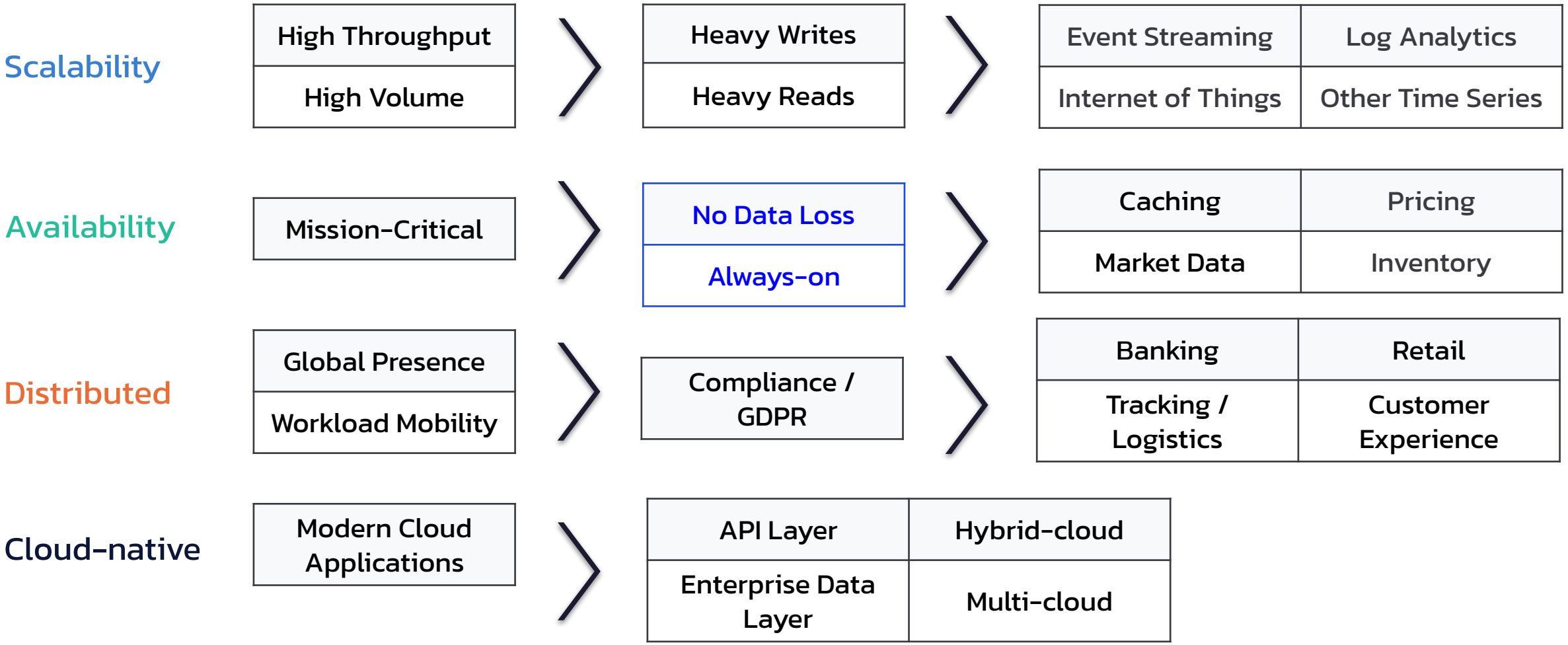
Partition Key

Apache Cassandra™ :: Data is Distributed

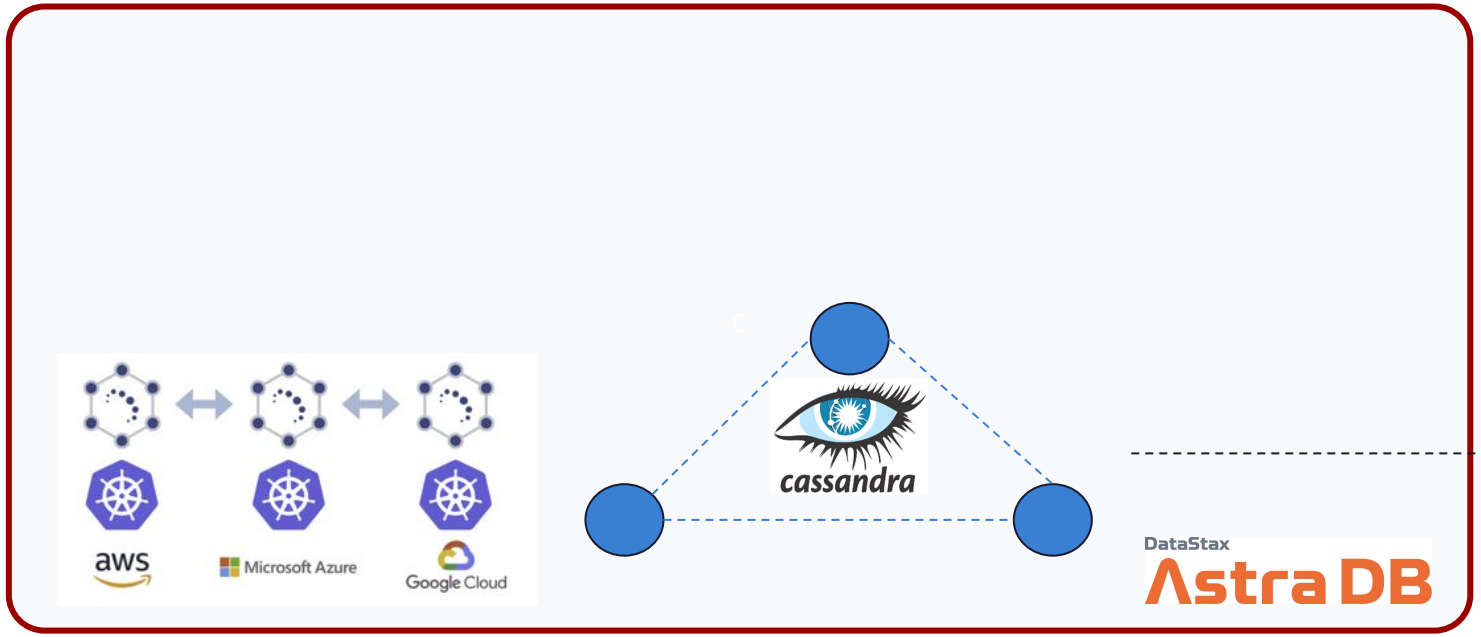


Country	City	Population

Apache Cassandra™ :: Use Cases



Initiating the Database



OSS Apache Cassandra
A Column oriented NoSQL Database

HandsOn: #4. Create Instance

!ASTRA

Astra DB

Get your instance here:

https://astra.dev/11-22

!GITHUB

GitHub

Repository:

github.com/datastaxdevs/workshop-spring-reactive

DataStax Developers

Normalization

"Database normalization is the process of structuring a relational database in accordance with a series of so-called normal forms in order to reduce data redundancy and improve data integrity. It was first proposed by Edgar F. Codd as part of his relational model."

PROS: Simple write, Data Integrity
CONS: Slow read, Complex Queries

Employees

userId	deptId	firstName	lastName
1	1	Edgar	Codd
2	1	Raymond	Boyce

Departments

departmentId	department
1	Engineering
2	Math

Denormalization

"Denormalization is a strategy used on a database to increase performance. In computing, denormalization is the process of trying to improve the read performance of a database, at the expense of losing some write performance, by adding redundant copies of data"

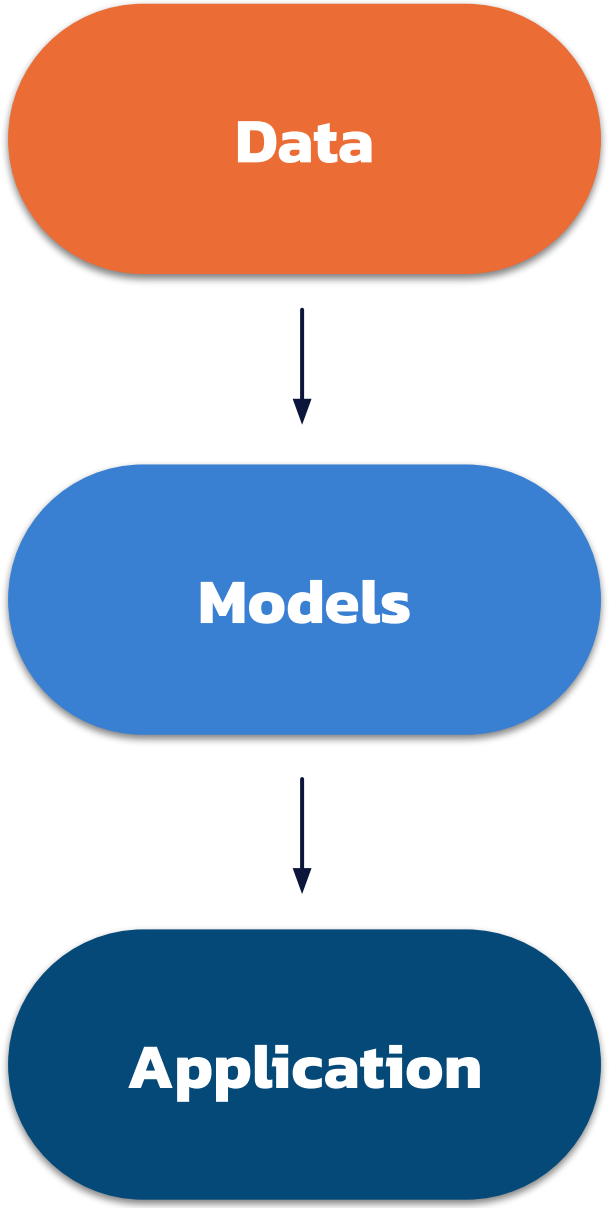
PROS: Quick Read, Simple Queries
CONS: Multiple Writes, Manual Integrity

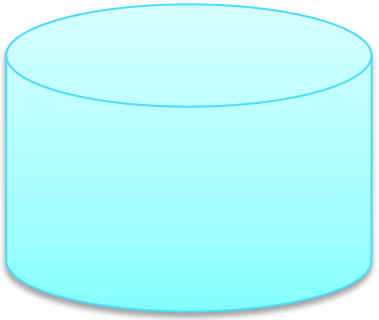
Employees

userId	firstName	lastName	department
1	Edgar	Codd	Engineering
2	Raymond	Boyce	Engineering
3	Sage	Lahja	Math
4	Juniper	Jones	Botany

Relational Data Model

1. Analyze raw data
2. Identify entities, their properties and relations
3. Design tables, using **normalization** and foreign keys.
4. Use JOIN when doing queries to join normalized data from multiple tables



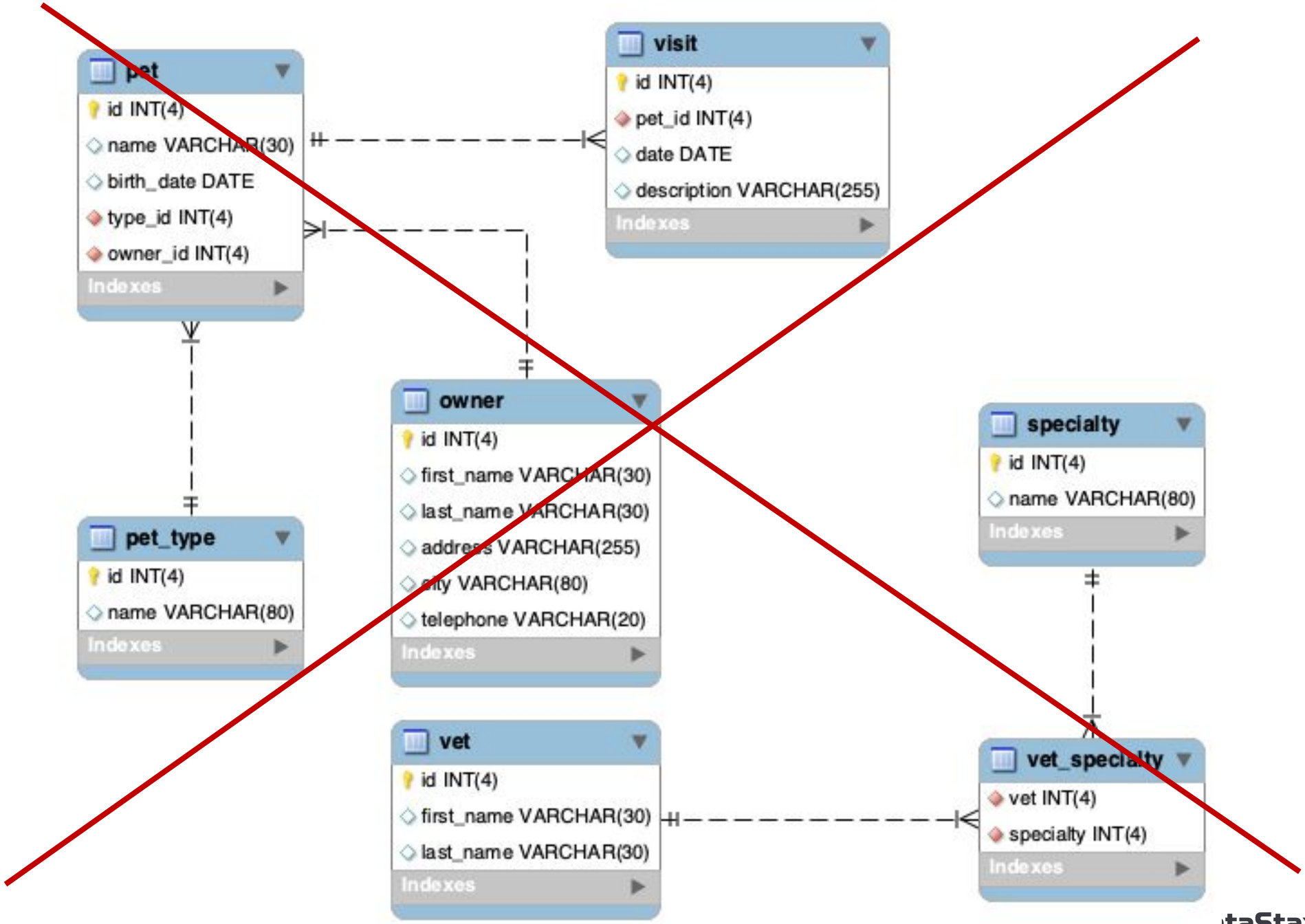


Employees

userId	firstName	lastName
1	Edgar	Codd
2	Raymond	Boyce

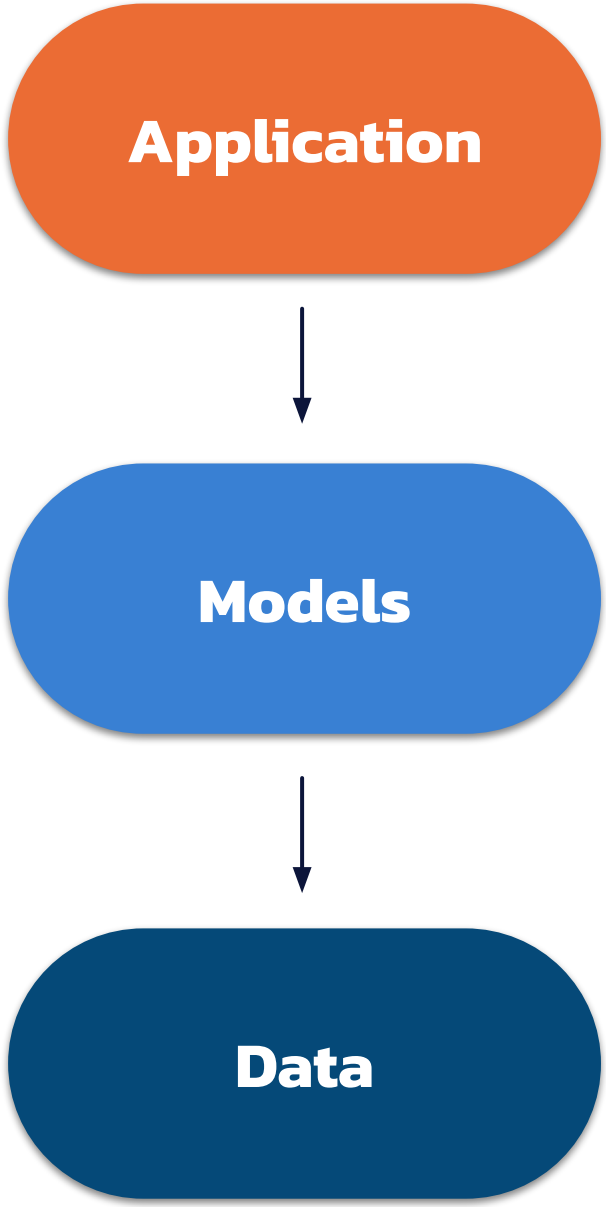
Departments

departmentId	department
1	Engineering
2	Math


pet
id INT(4)
name VARCHAR(30)
birth_date DATE
type_id INT(4)
owner_id INT(4)
Indexes
visit
id INT(4)
pet_id INT(4)
date DATE
description VARCHAR(255)
Indexes
owner
id INT(4)
first_name VARCHAR(30)
last_name VARCHAR(30)
address VARCHAR(255)
city VARCHAR(80)
telephone VARCHAR(20)
Indexes
pet_type
id INT(4)
name VARCHAR(80)
Indexes
specialty
id INT(4)
name VARCHAR(80)
Indexes
vet
id INT(4)
first_name VARCHAR(30)
last_name VARCHAR(30)
Indexes
vet_specialty
vet INT(4)
specialty INT(4)
Indexes

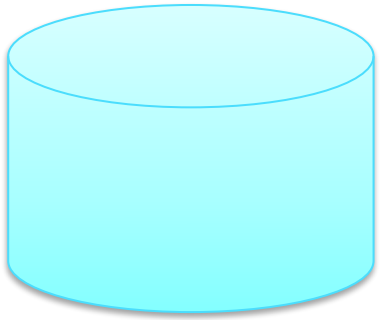
NoSQL Data Model

1. Analyze user behaviour (customer first!)
2. Identify workflows, their dependencies and needs
3. Define Queries to fulfill these workflows
4. Knowing the queries, design tables, using **denormalization**.
5. Insert and update multiple copies of data that may have resulted due to denormalization



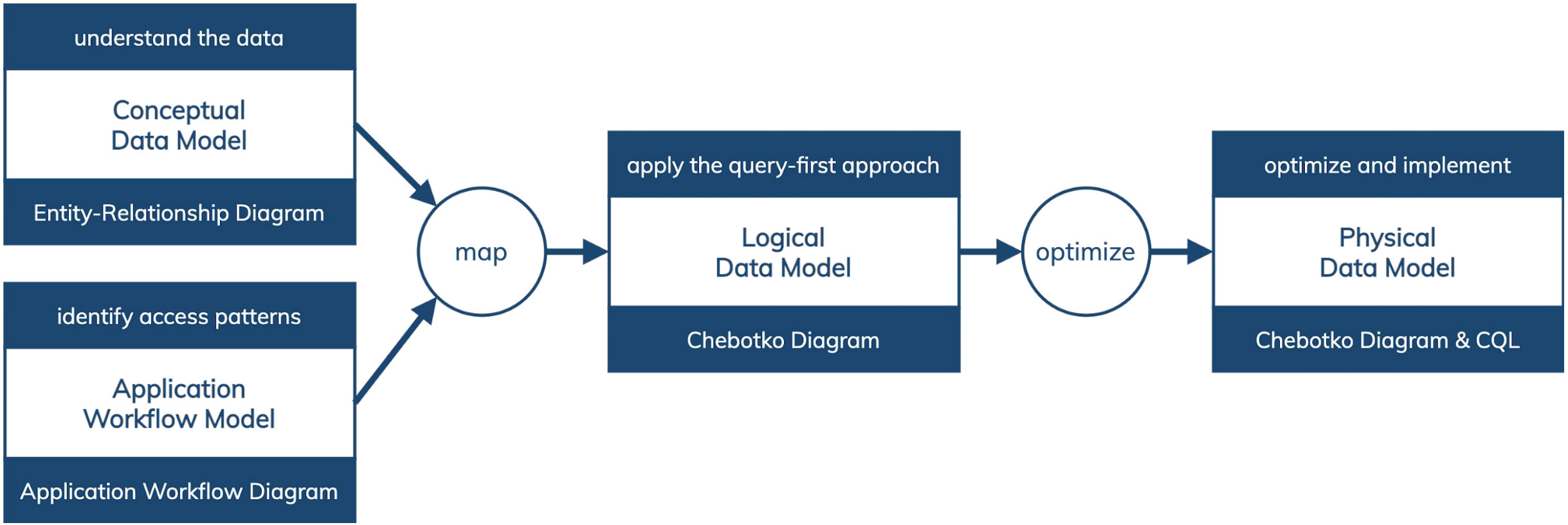
Employees

userId	firstName	lastName	department
1	Edgar	Codd	Engineering
2	Raymond	Boyce	Math
3	Sage	Lahja	Math
4	Juniper	Jones	Botany

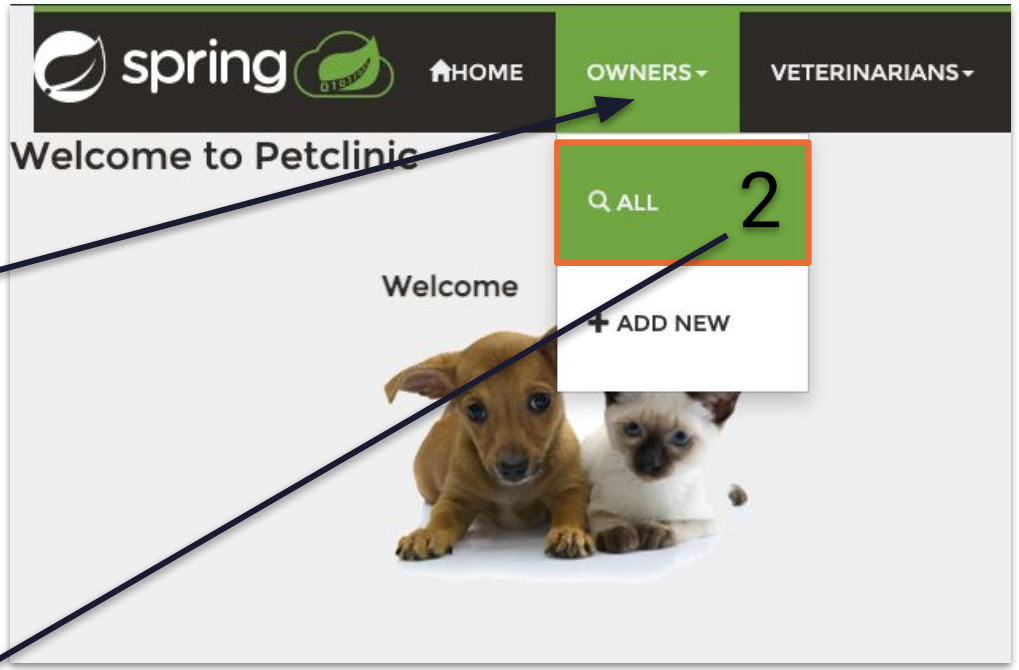


Cassandra Data Modeling Methodology

Four objectives, 4 models, 2 transitions



#1 – List Owners

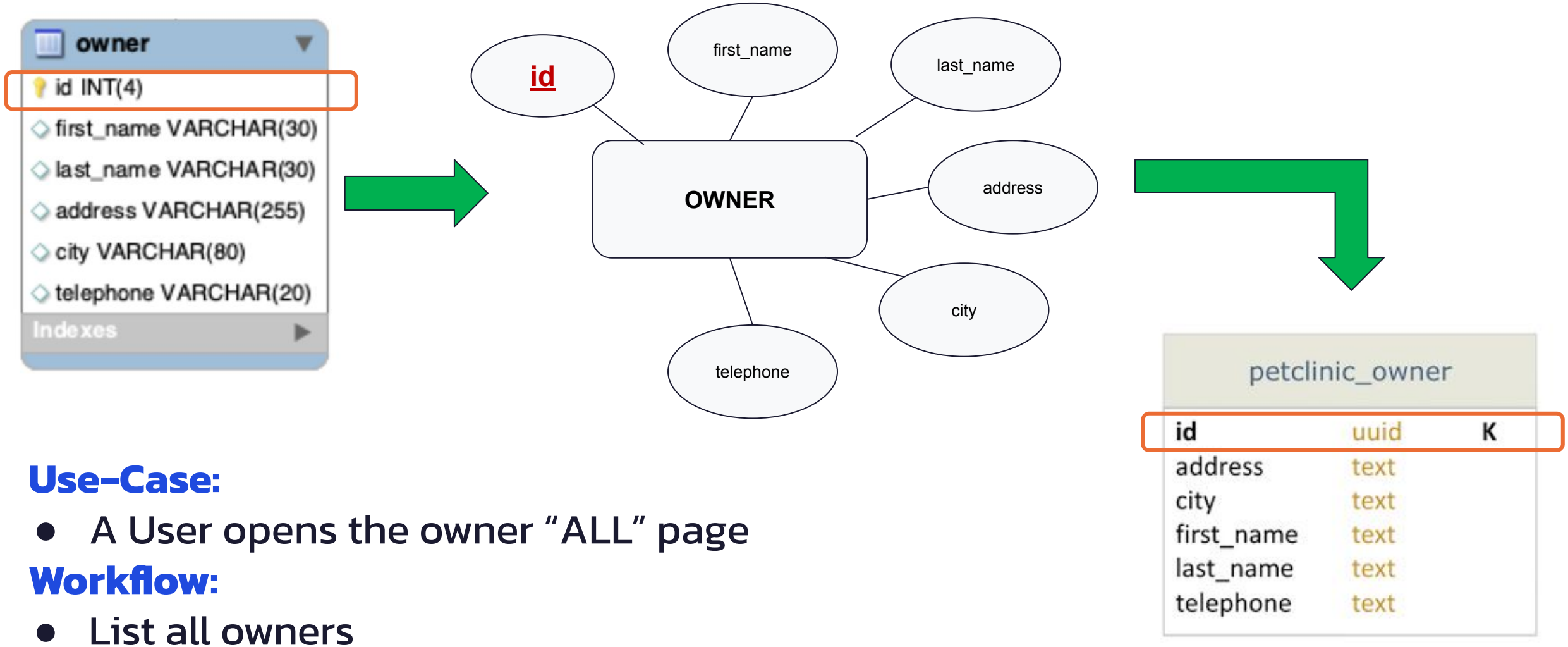


Owners

Name	Address	City	Telephone	Pets
Heather Jones-Gilardi	AnotherPlace Dr.	Orlando	987654321	
David Gilardi	Nowhere St.	Orlando	123456789	Carrabba Shocker

Add Owner

#1 – List Owners



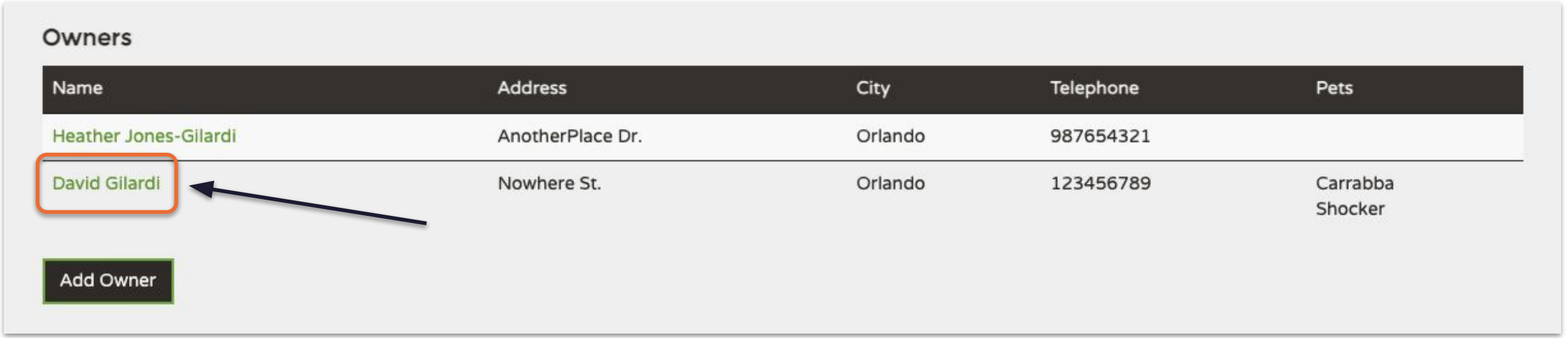
Use-Case:

- A User opens the owner "ALL" page

Workflow:

- List all owners

#2 – Get Pets of Owner (1 to many)



#2 – Get Pets of Owner (1 to many)

Owner Information

Name	David Gilardi
Address	Nowhere St.
City	Orlando
Telephone	123456789

< Back | Edit Owner | Add New Pet

Pets and Visits

Name Shocker
Birth Date 2021/03/29
Type dog

Edit Pet | Delete Pet | Add Visit

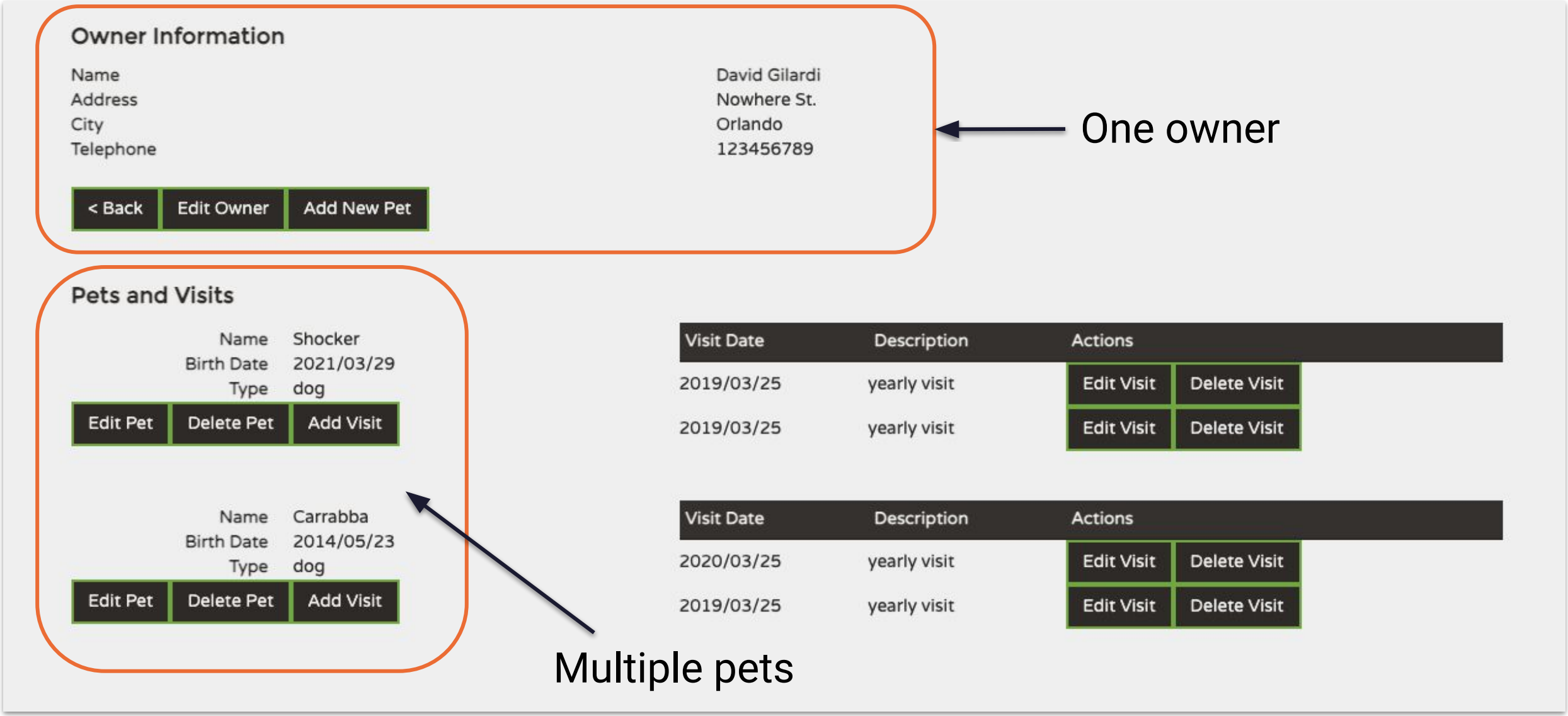
Visit Date	Description	Actions
2019/03/25	yearly visit	Edit Visit | Delete Visit
2019/03/25	yearly visit	Edit Visit | Delete Visit

Name Carrabba
Birth Date 2014/05/23
Type dog

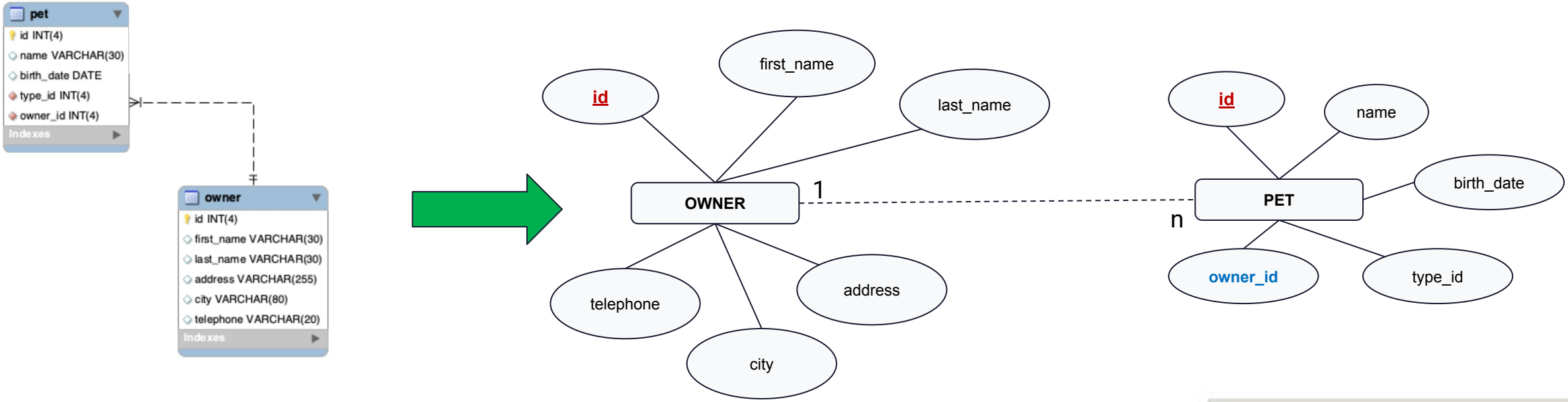
Edit Pet | Delete Pet | Add Visit

Visit Date	Description	Actions
2020/03/25	yearly visit	Edit Visit | Delete Visit
2019/03/25	yearly visit	Edit Visit | Delete Visit

#2 – Get Pets of Owner (1 to many)



#2 - Get Pets of Owner (1 to many)

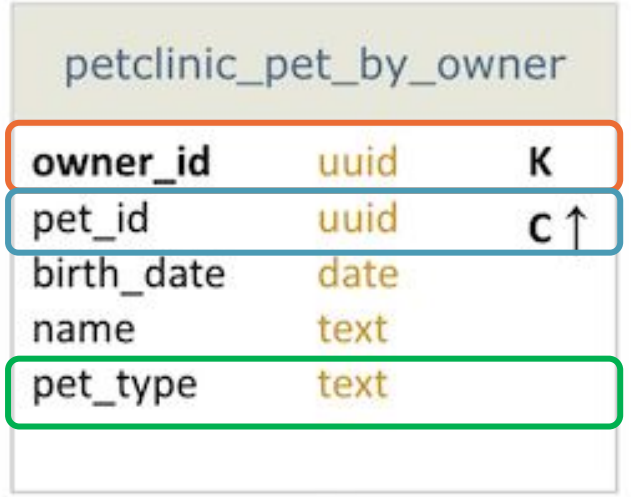


Use-Case:

- A User opens the owner detail page

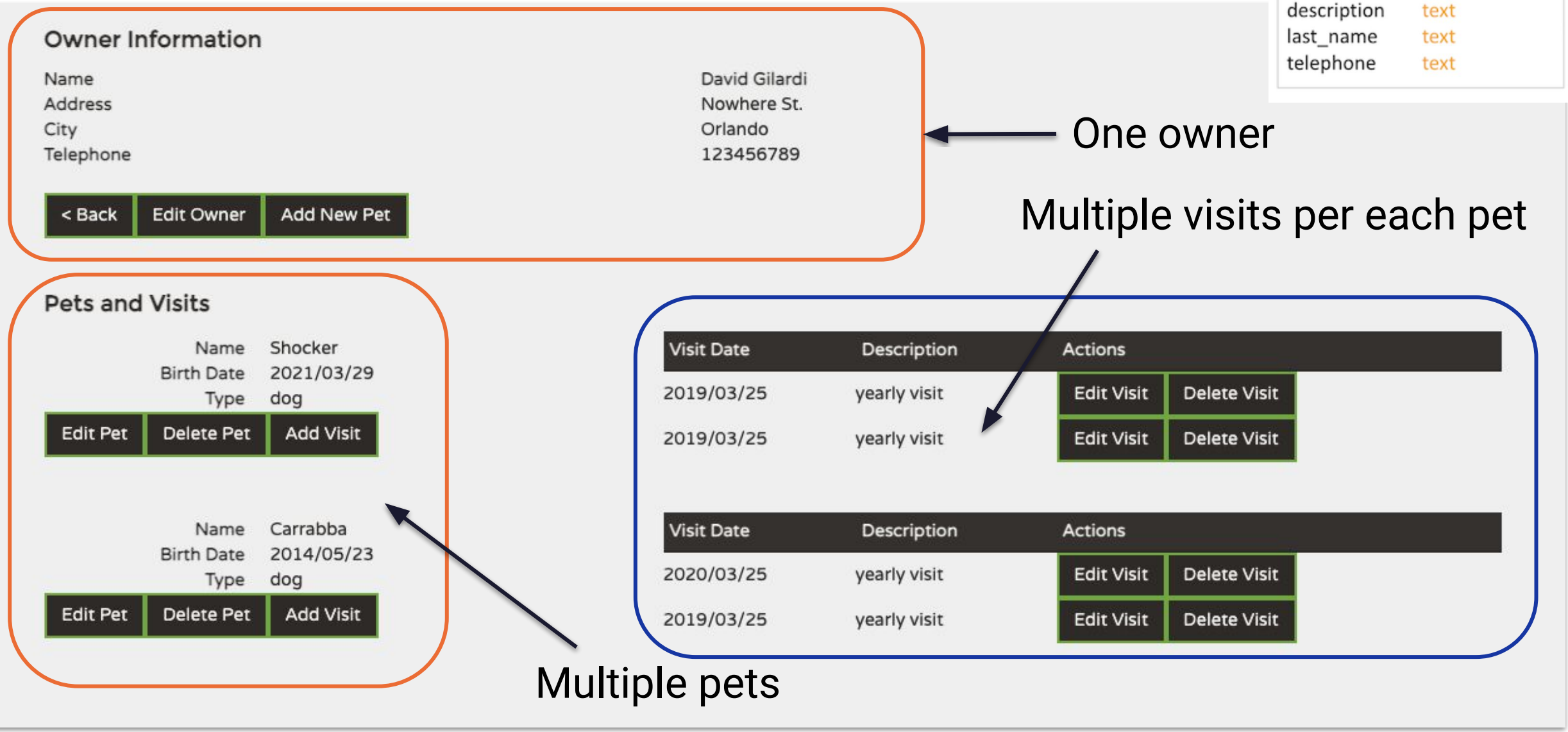
Workflow:

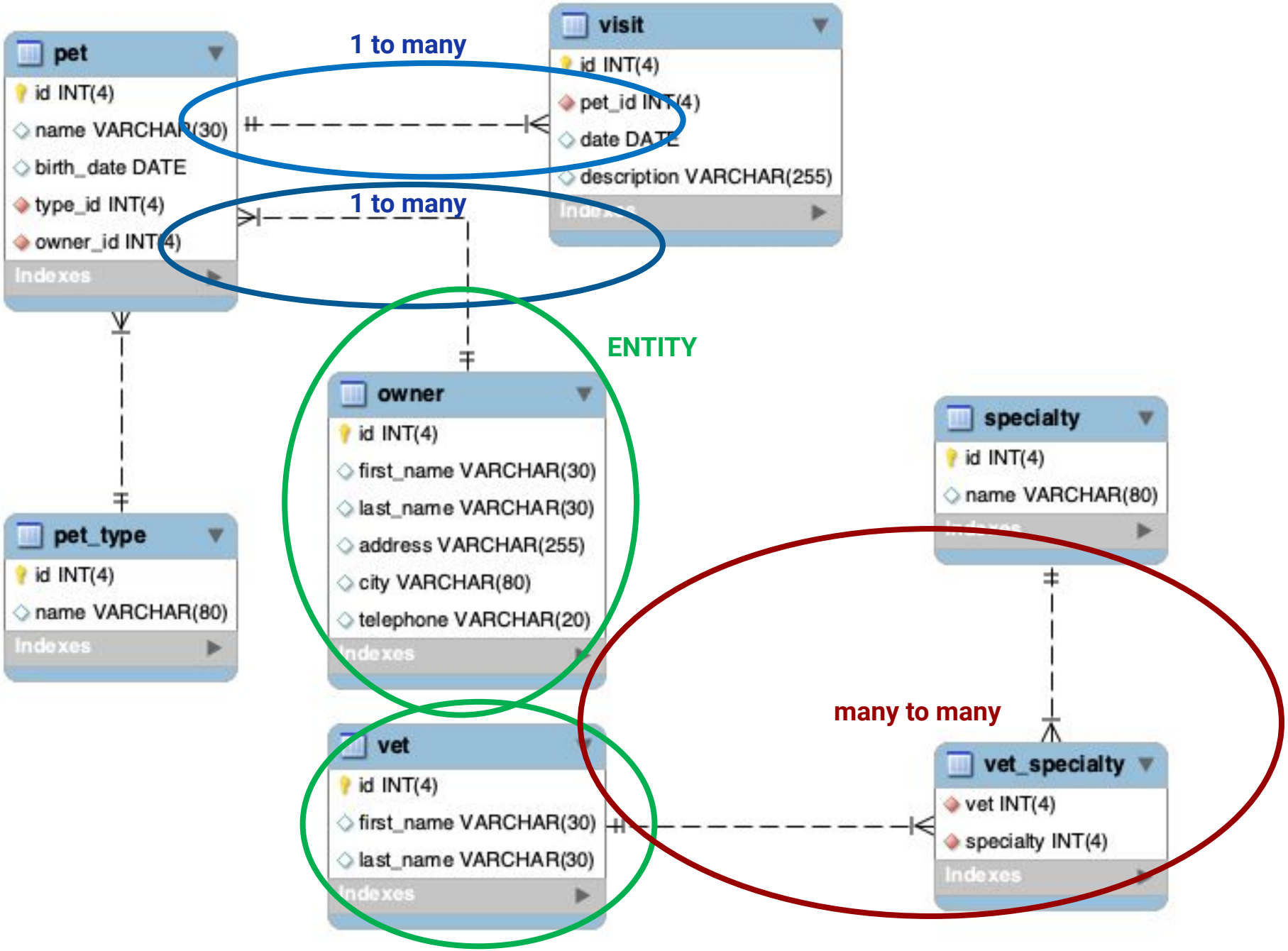
- Find owner and any pets related to target owner using its identifier (owner_id)



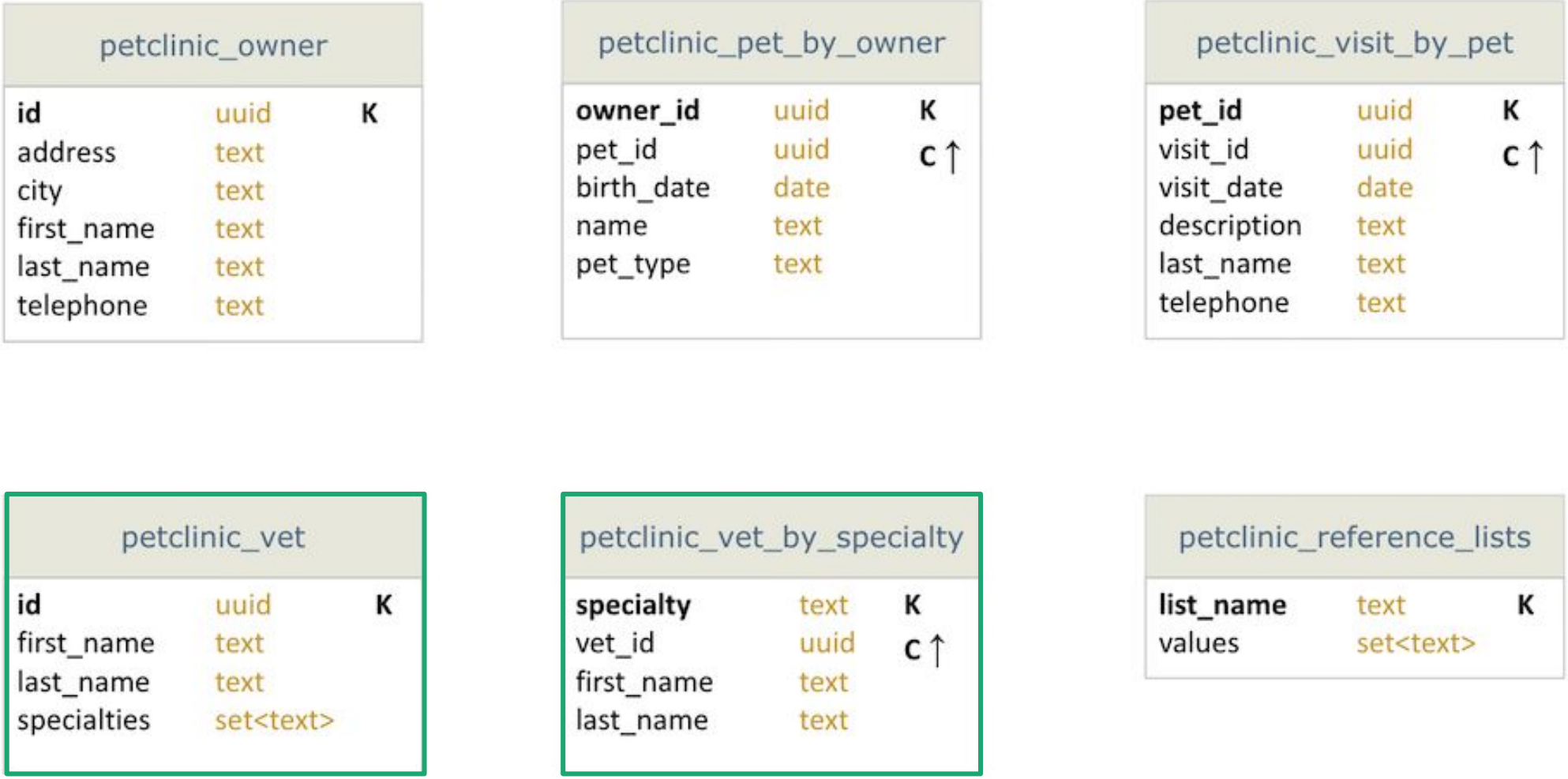
#3 – Get Visits of Pets (1 to many)

petclinic_visit_by_pet		
pet_id	uuid	K
visit_id	uuid	C ↑
visit_date	date	
description	text	
last_name	text	
telephone	text	




pet
id INT(4)
name VARCHAR(30)
birth_date DATE
type_id INT(4)
owner_id INT(4)
Indexes
1 to many
visit
id INT(4)
pet_id INT(4)
date DATE
description VARCHAR(255)
1 to many
ENTITY
owner
id INT(4)
first_name VARCHAR(30)
last_name VARCHAR(30)
address VARCHAR(255)
city VARCHAR(80)
telephone VARCHAR(20)
pet_type
id INT(4)
name VARCHAR(80)
specialty
id INT(4)
name VARCHAR(80)
many to many
vet
id INT(4)
first_name VARCHAR(30)
last_name VARCHAR(30)
vet_specialty
vet INT(4)
specialty INT(4)

Complete Model



HandsOn: 5. Create your schema

!ASTRA

Astra DB

Get your instance here:

https://astra.dev/11-22

!GITHUB

GitHub

Repository:

github.com/datastaxdevs/workshop-spring-reactive

DataStax Developers

Agenda

01

PetClinic
Architecture & Use Case

02

DataStax Astra

Cassandra Database
The Art of Data Modelling

03

Reactive Drivers
Reactive vs Async

04

Spring Reactive
Boot and WebFlux

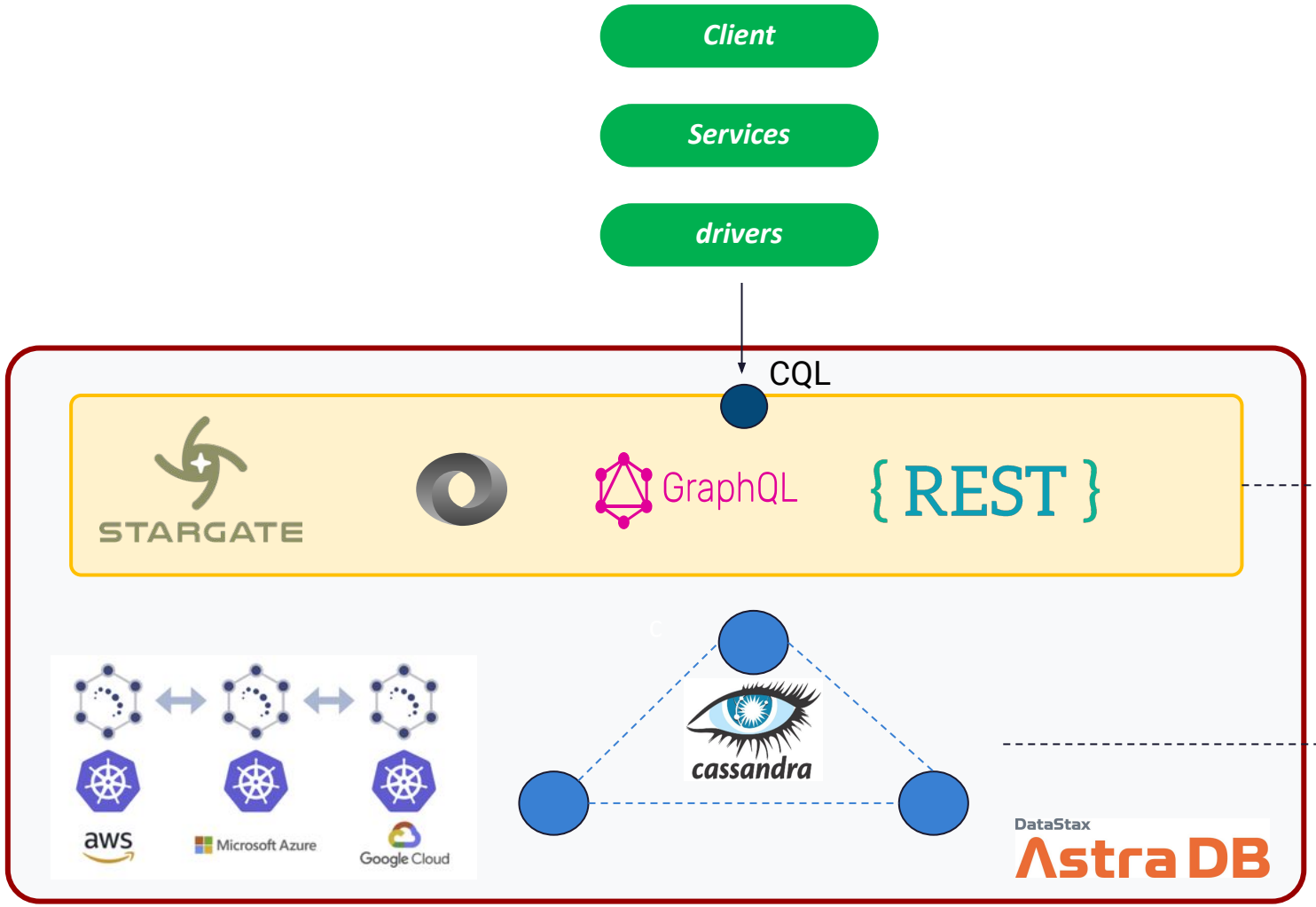
05

User Interface
Angular

06

Game &
Resources

Accessing the Database



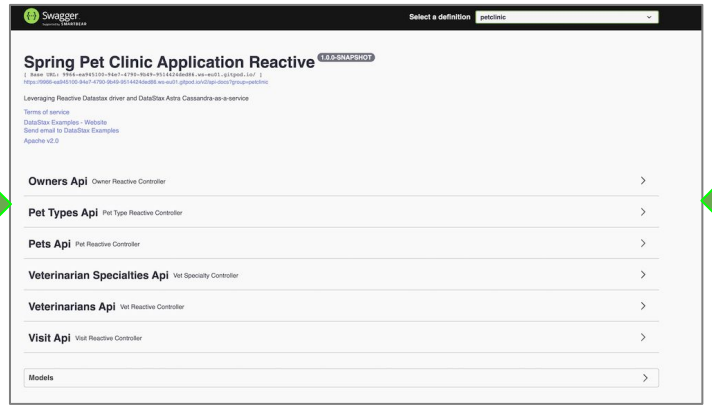
OSS Stargate.io
A data gateway to allow multiple usages

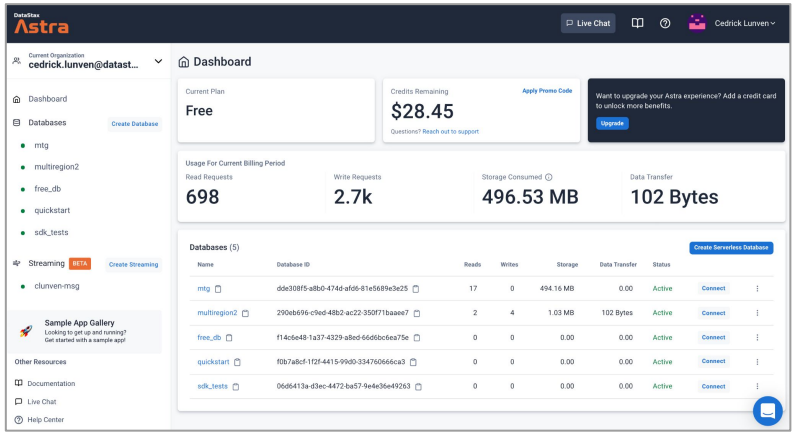
STARGATE

OSS Apache Cassandra
A Column oriented NoSQL Database

cassandra

Sequence Diagram (1)





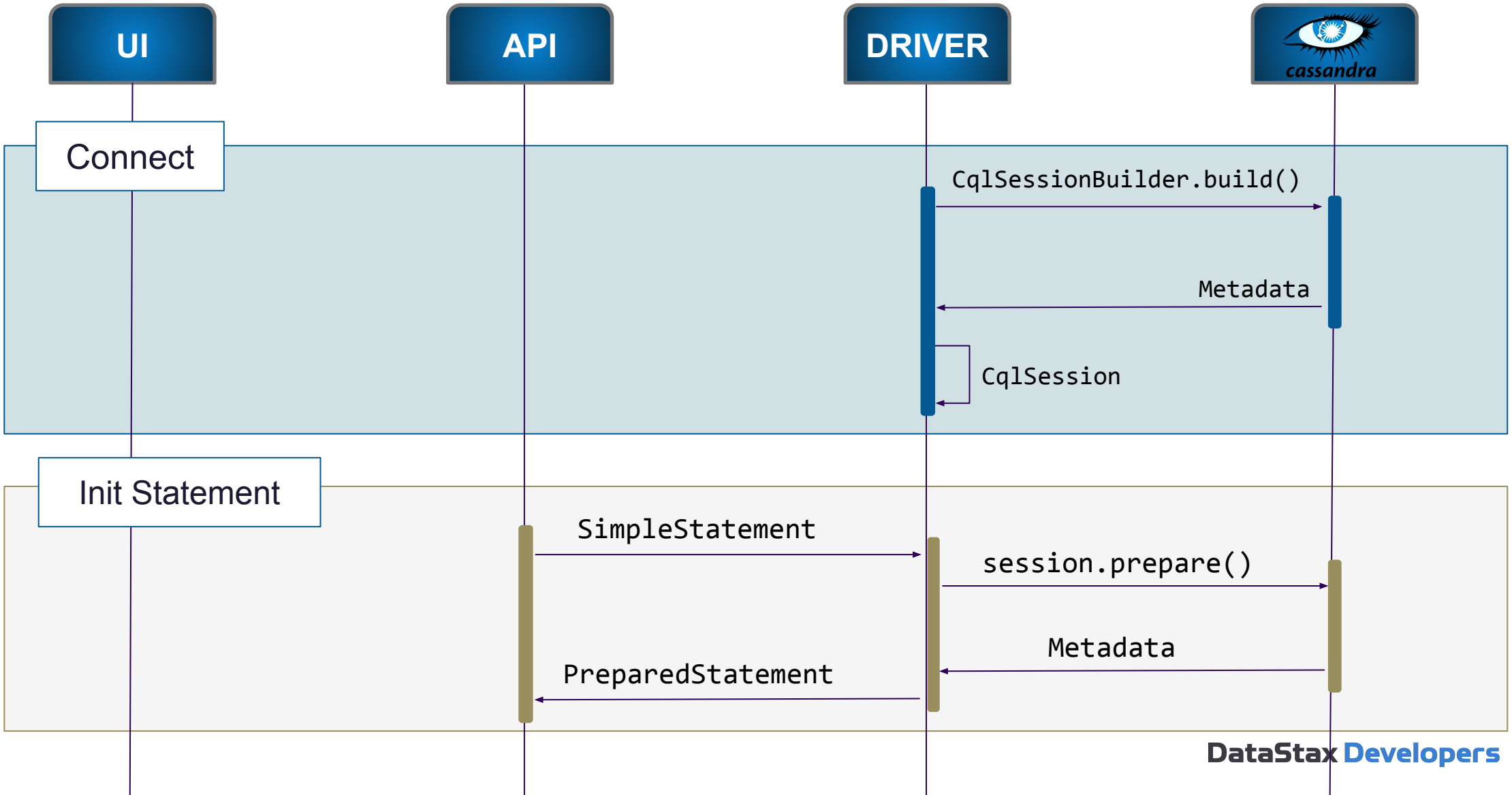
Client

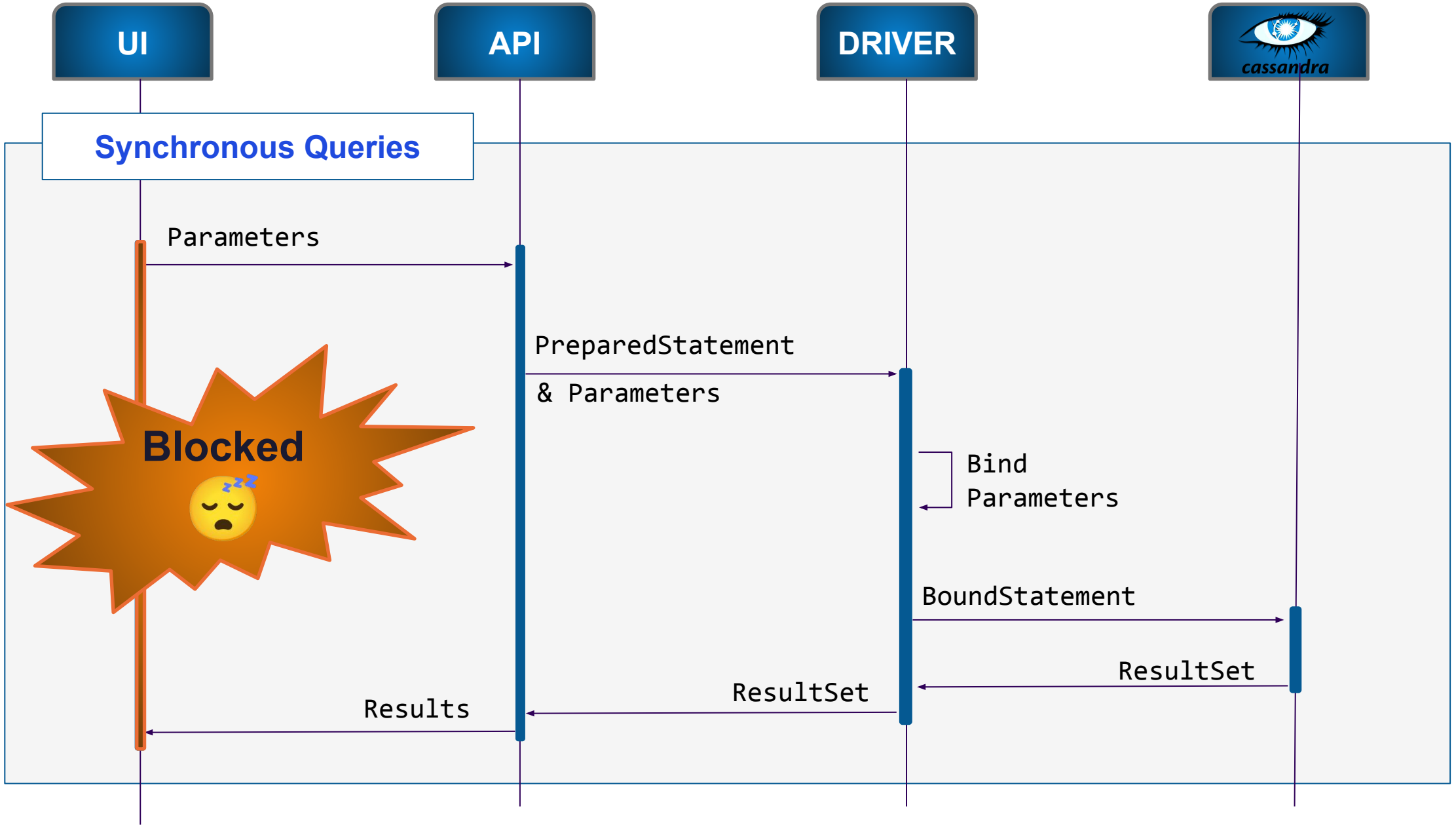
API

DRIVER

DB

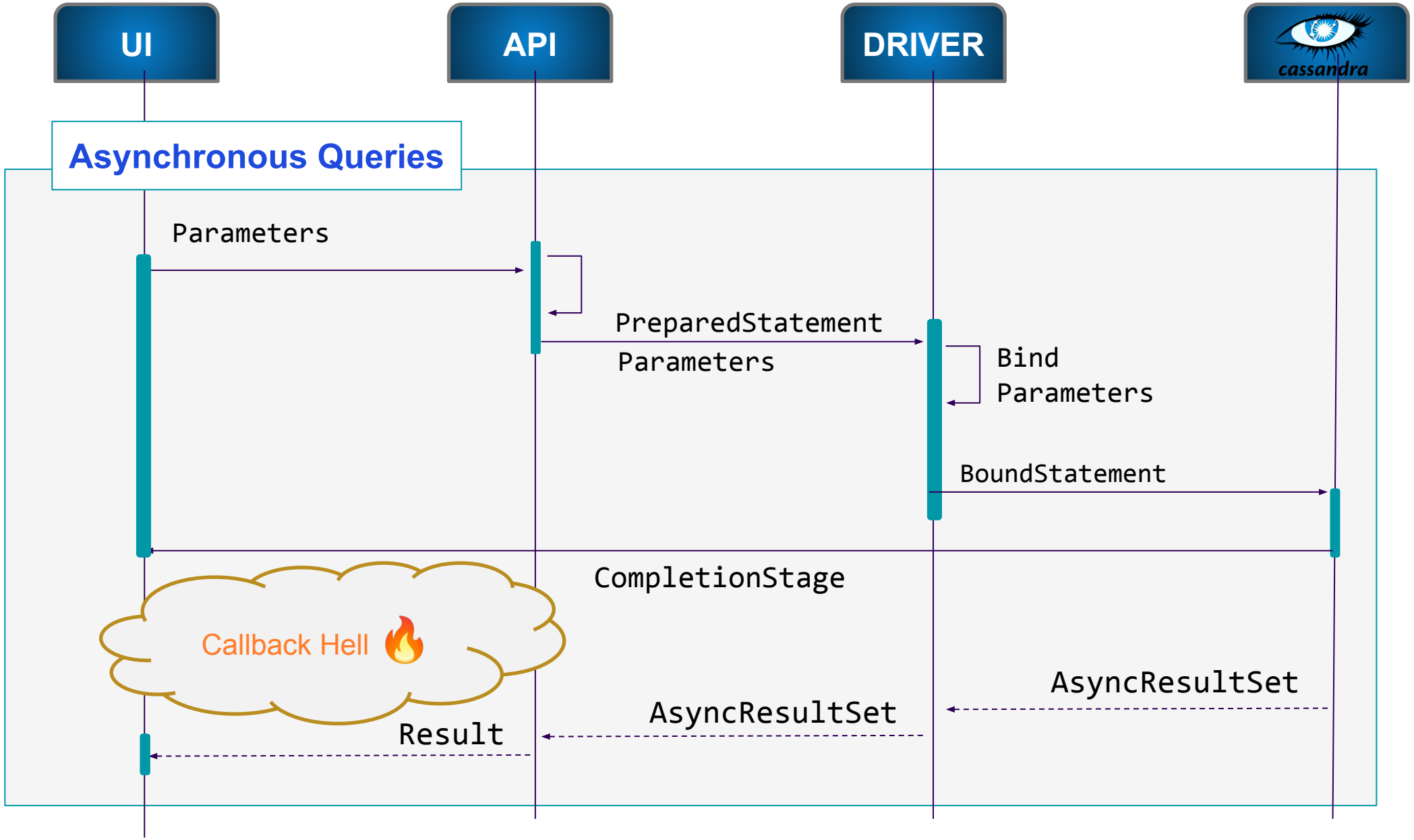
Initializations




UI
API
DRIVER
cassandra
Synchronous Queries
Parameters
PreparedStatement
& Parameters
Blocked
Bind
Parameters
BoundStatement
ResultSet
ResultSet
Results

Synchronous

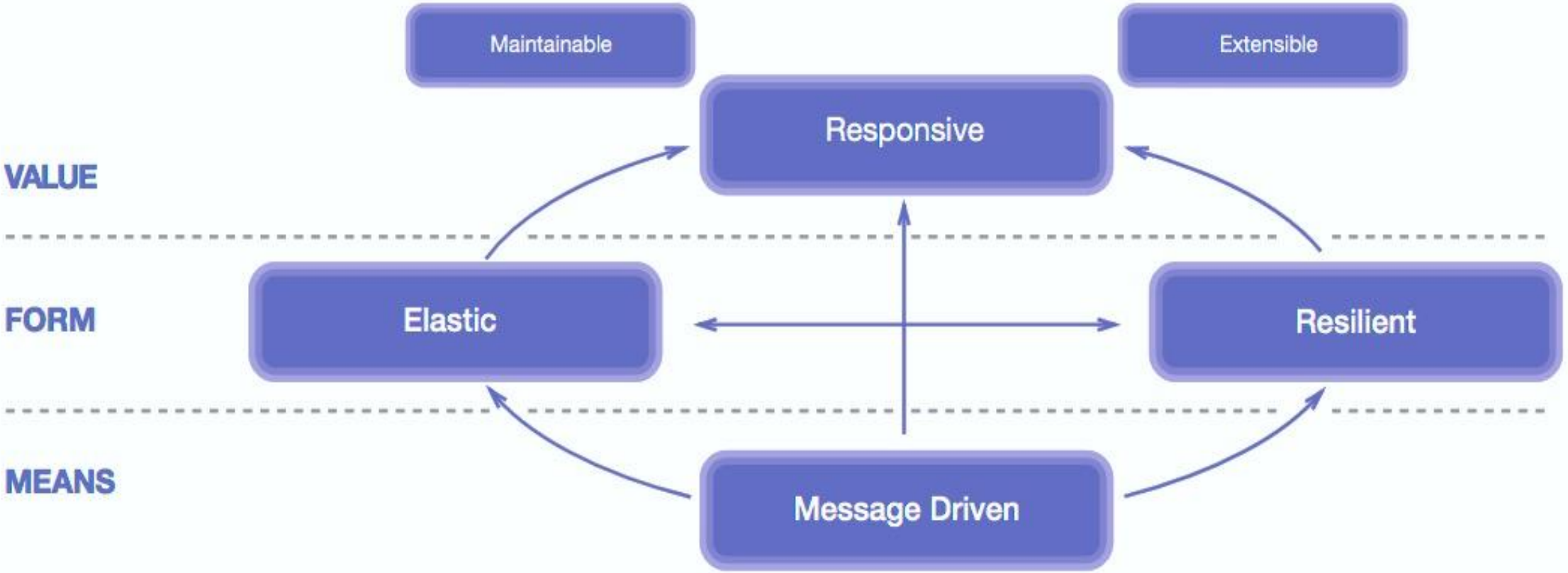
Strengths	Weaknesses
Simple	Blocking 😴
Opportunities	**Threats**
Easy to Test & Maintain	Scalability ⚠️


UI
API
DRIVER
cassandra
Asynchronous Queries
Parameters
PreparedStatement
Parameters
Bind
Parameters
BoundStatement
CompletionStage
Callback Hell
AsyncResultSet
AsyncResultSet
Result

Asynchronous

Strengths	Weaknesses
Non Blocking	Callback Hell 🔥
Opportunities	**Threats**
Scalability	Maintainability 🤯

Reactive Manifesto

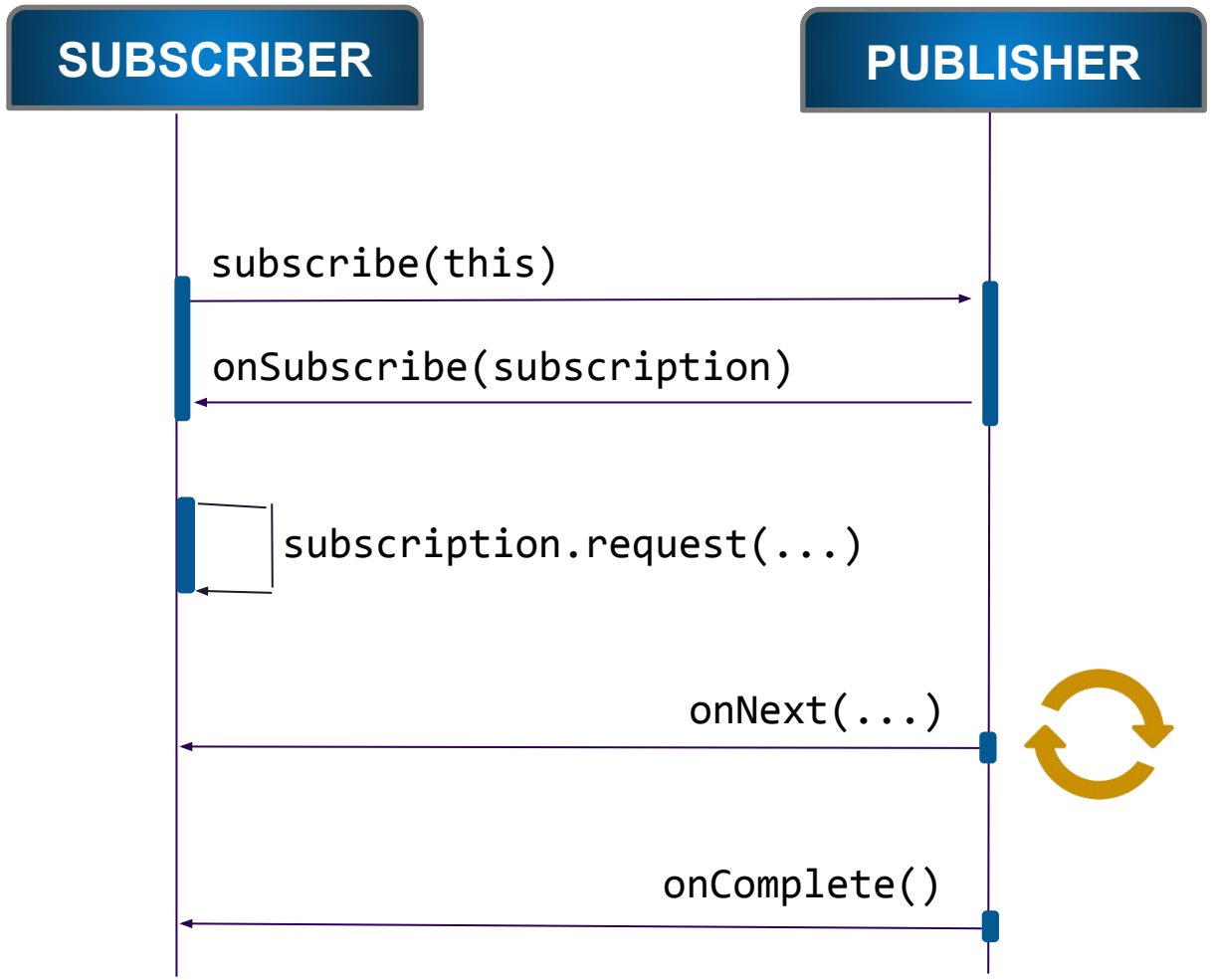


https://www.reactivemanifesto.org/

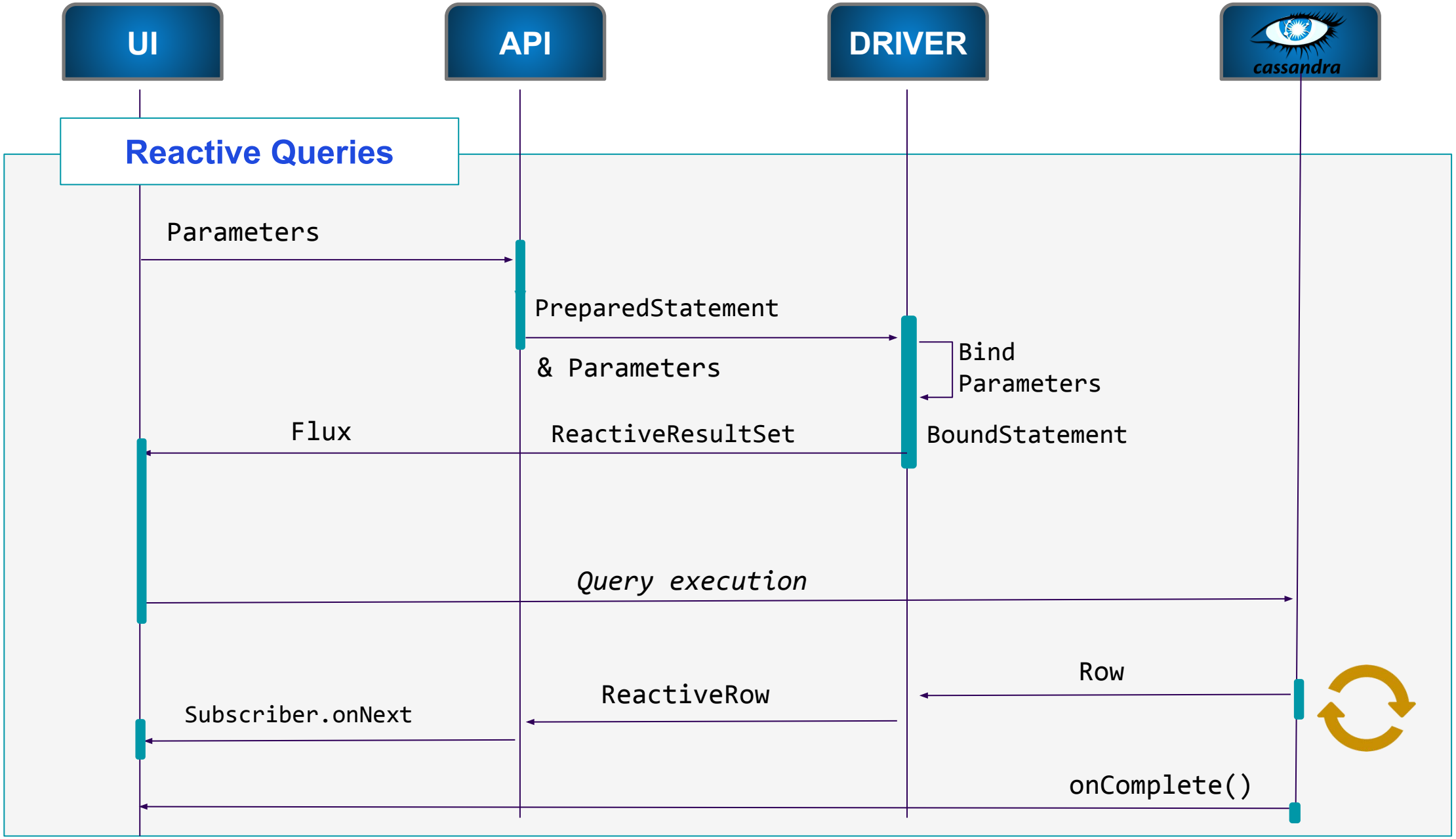
Reactive Stream Api

VERT.X

akka

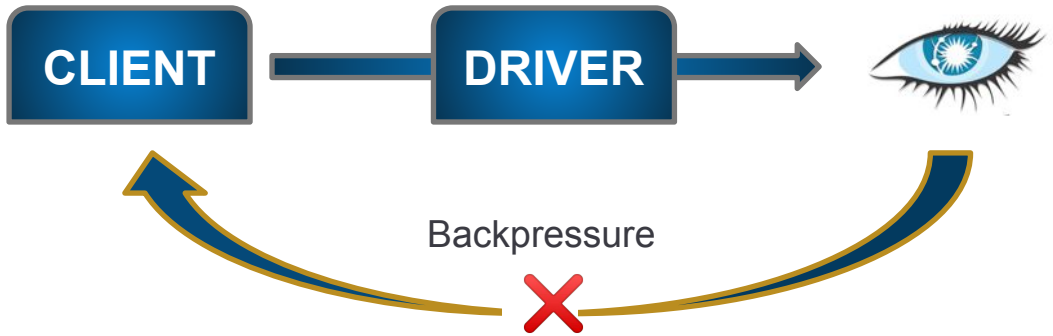


http://www.reactive-streams.org/


UI
API
DRIVER
cassandra
Reactive Queries
Parameters
PreparedStatement
& Parameters
Bind
Parameters
Flux
ReactiveResultSet
BoundStatement
Query execution
Row
ReactiveRow
Subscriber.onNext
onComplete()

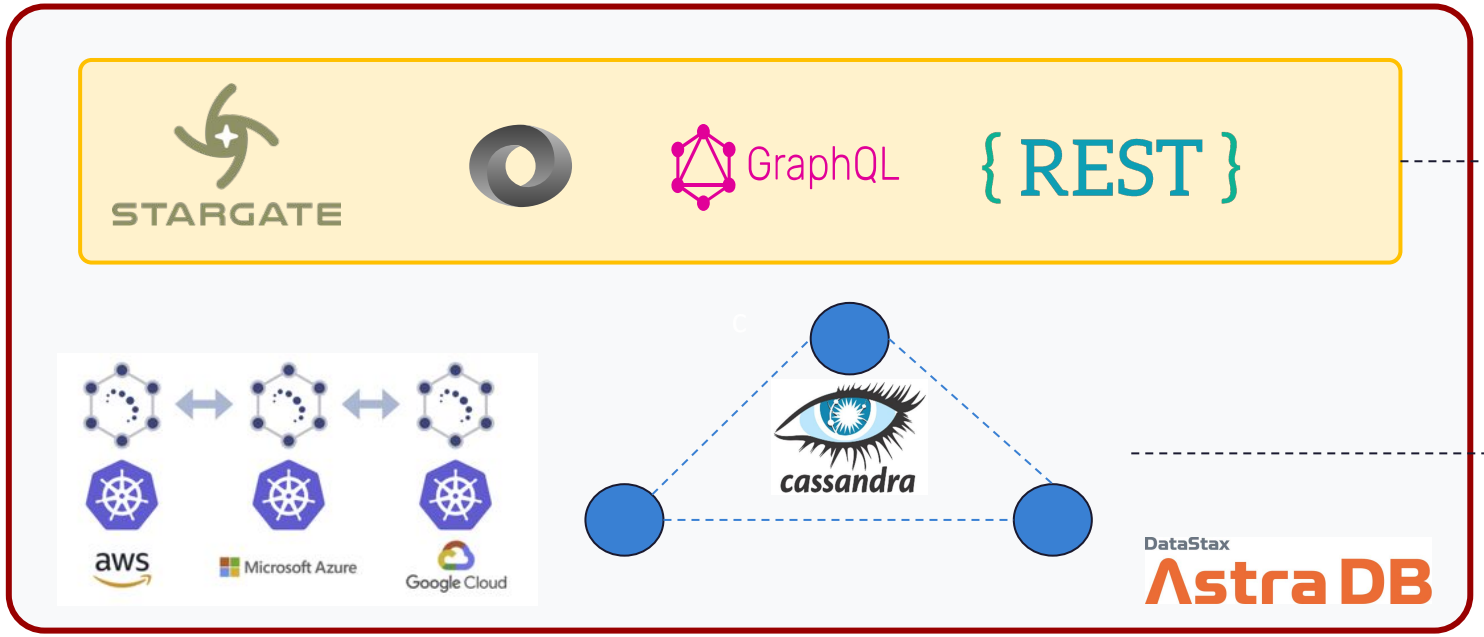
Back Pressure

READ



WRITE

Astra Connectivity



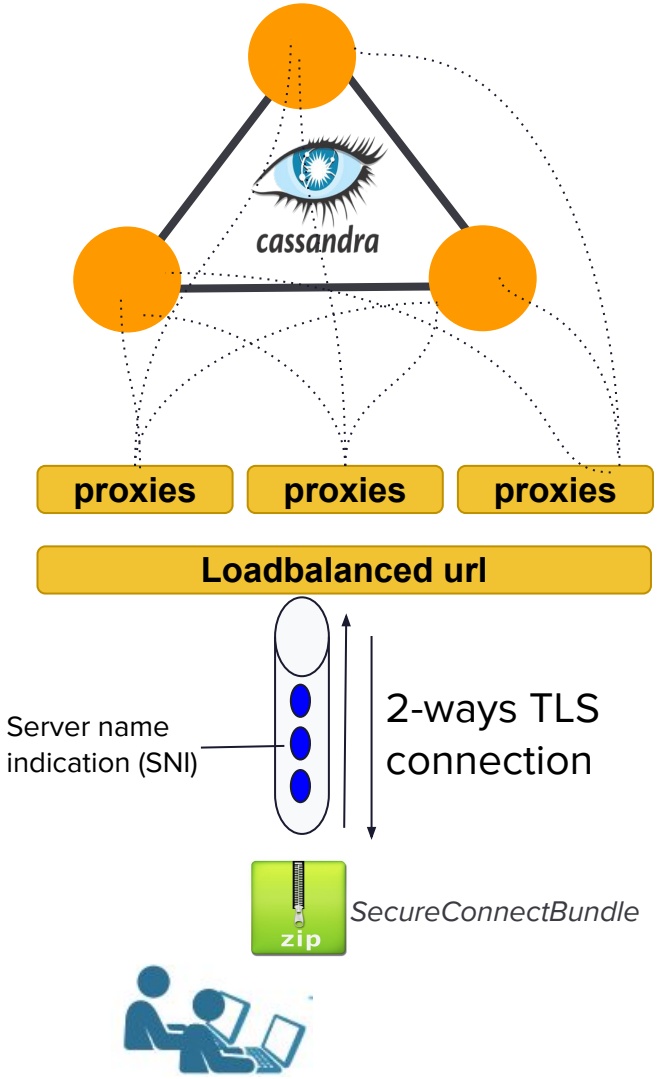
OSS Stargate.io
A data gateway to allow multiple usages

STARGATE

OSS Apache Cassandra
A Column oriented NoSQL Database

cassandra

Astra Connecticvity



HandsOn: 6, 7 and 8

6. Create your Token
7. Start Gitpod and setup your application
8. Working with Cassandra Drivers

!GITHUB

GitHub

Repository:

github.com/datastaxdevs/workshop-spring-reactive

DataStax Developers

Agenda

01

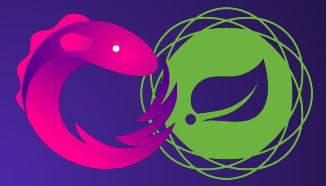
PetClinic
Architecture & Use Case

02

DataStax Astra

Cassandra Database
The Art of Data Modelling

03



Spring Reactive
Reactive vs Async

04

Spring Reactive
Boot and WebFlux

05

User Interface
Angular

06

Game &
Resources

Spring Framework : Dependency Injection

Spring Framework

Provides core support for dependency injection, transaction management, web apps, data access, messaging, and more.

```
public interface GreetingServices {
    String sayHello();
}
```

```
@Component("greeting.fr")
public class GreetingServicesFR implements GreetingServices {
  public String sayHello() { return "Bonjour !"; }
}
```

```
@Component("greeting.en")
public class GreetingServicesEN implements GreetingServices {
 public String sayHello() { return "Hello !"; }
}
```

```
@RestController
public class MyAPI {

    @Autowired
    @Qualifier("greeting.en") // <--
    private GreetingServices greetingService;

    @GetMapping
    public String sayHello() {
        return greetingService.sayHello();
    }
}
```

What Spring Framework Can do

spring.io

Microservices

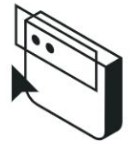
Quickly deliver production-grade features with independently evolvable microservices.

Reactive

Spring's asynchronous, nonblocking architecture means you can get more from your computing resources.

Cloud

Your code, any cloud—we've got you covered. Connect and scale your services, whatever your platform.



Web apps

Frameworks for fast, secure, and responsive web applications connected to any data store.

Serverless

The ultimate flexibility. Scale up on demand and scale to zero when there's no demand.

Event Driven

Integrate with your enterprise. React to business events. Act on your streaming data in realtime.

Batch

Automated tasks. Offline processing of data at a time to suit you.

Spring Framework Projects

https://spring.io/projects

Spring Boot

Takes an opinionated view of building Spring applications and gets you up and running as quickly as possible.

Spring Data

Provides a consistent approach to data access – relational, non-relational, map-reduce, and beyond.

Spring Security

Protects your application with comprehensive and extensible authentication and authorization support.

Spring Batch

Simplifies and optimizes the work of processing high-volume batch operations.

Spring Cloud

Provides a set of tools for common patterns in distributed systems. Useful for building and deploying microservices.

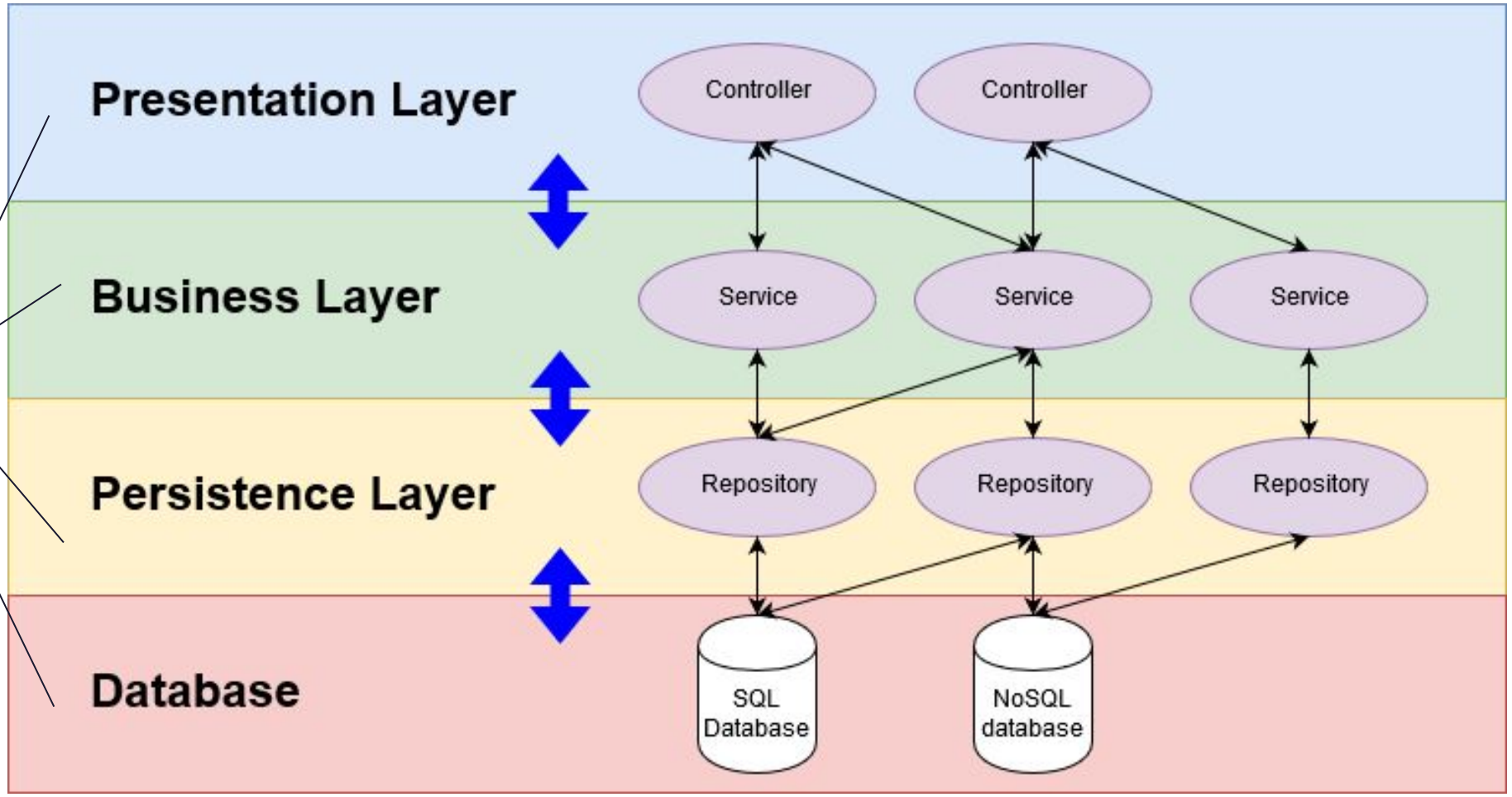
Spring REST Docs

Lets you document RESTful services by combining hand-written documentation with auto-generated snippets produced with Spring MVC Test or REST Assured.

... and many more

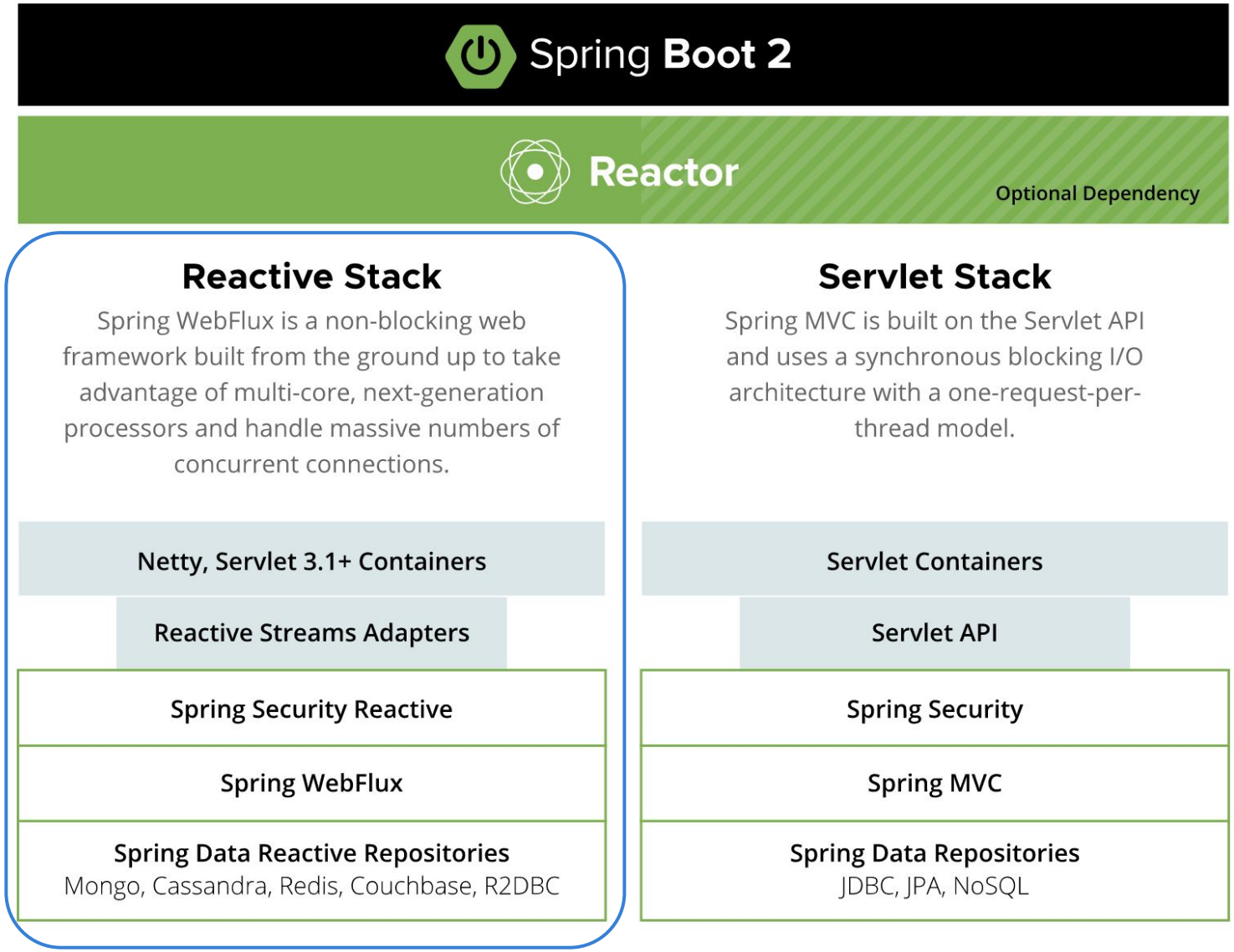
Spring Boot Application

spring boot



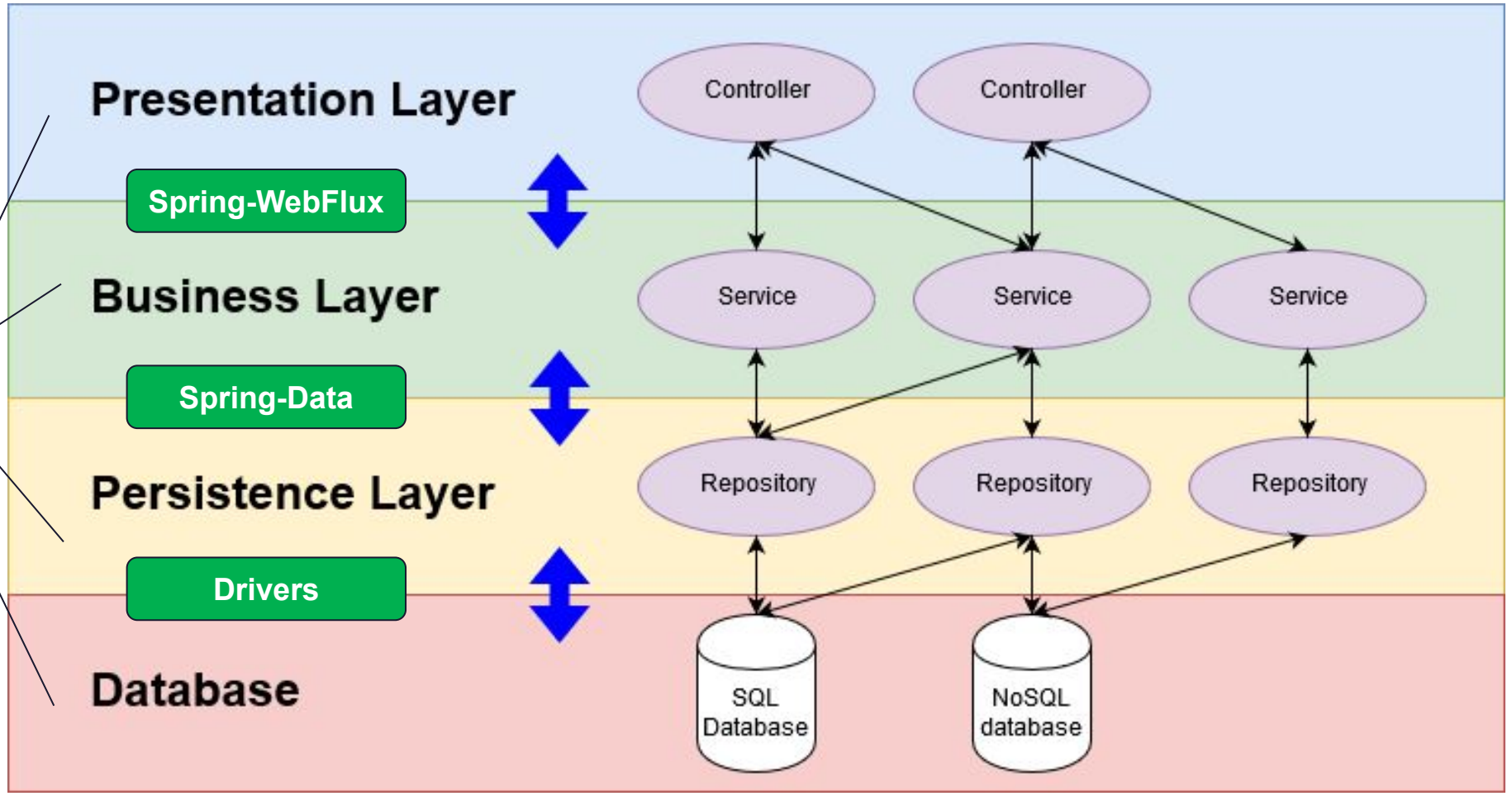
Spring Reactive

https://spring.io/reactive



Spring Boot Application

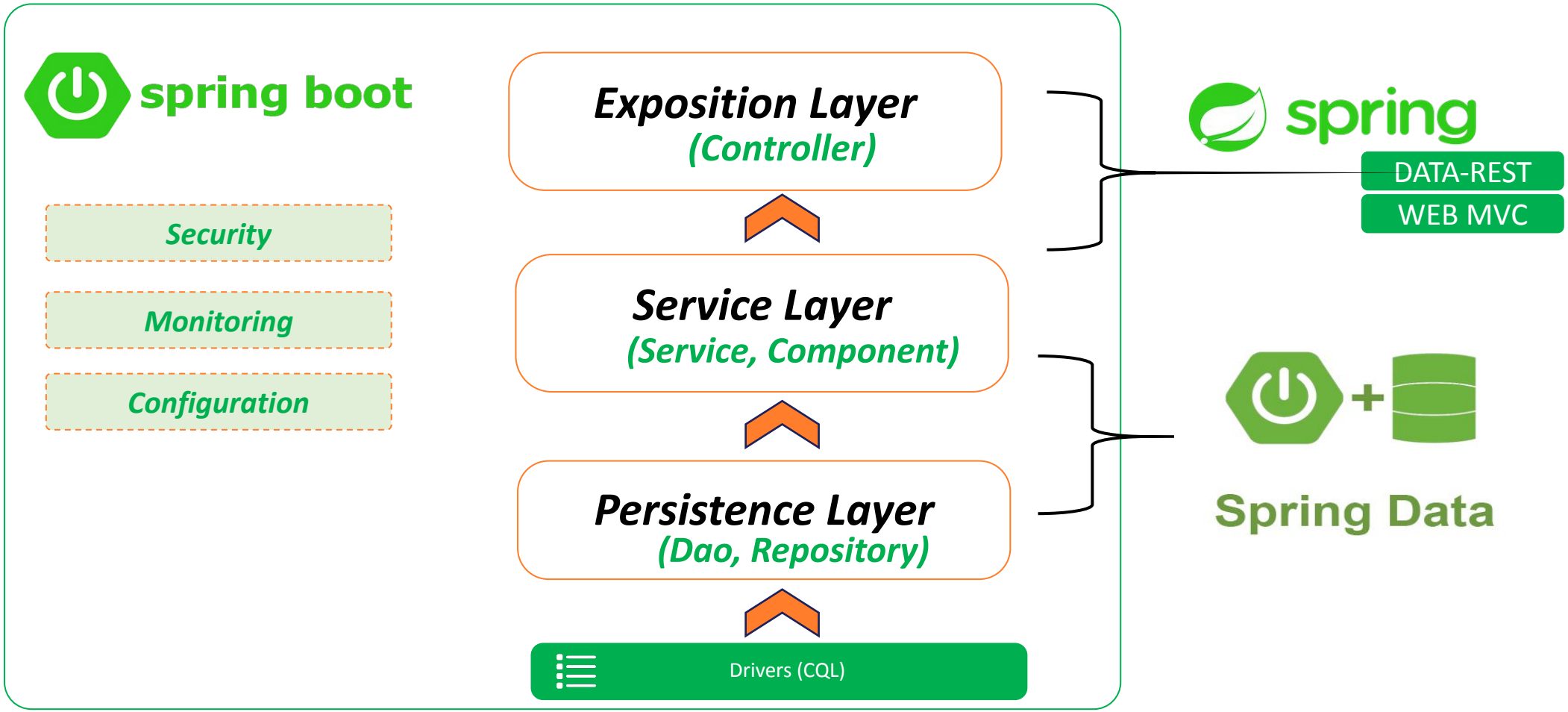
spring boot



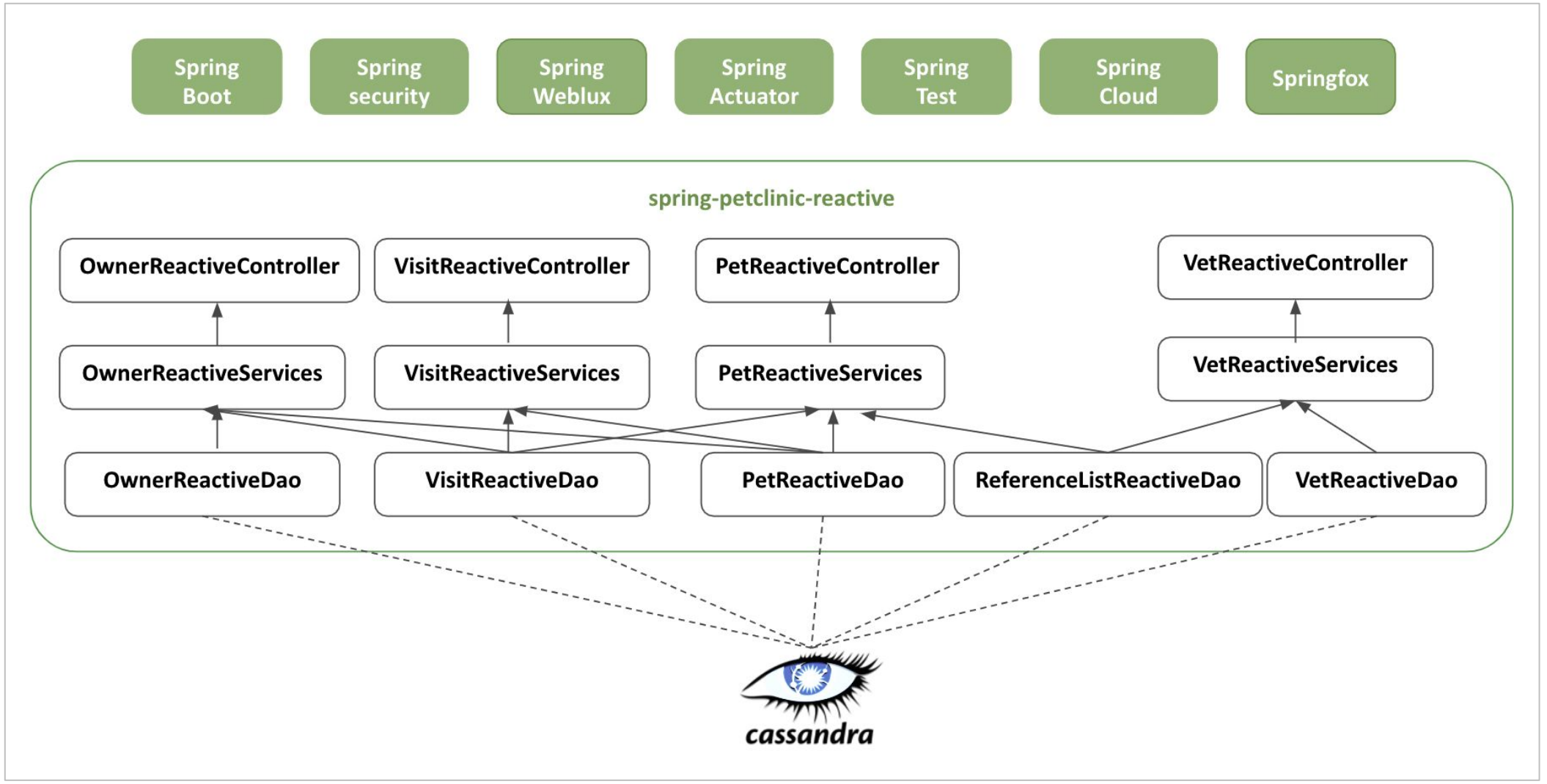
Spring boot Starters

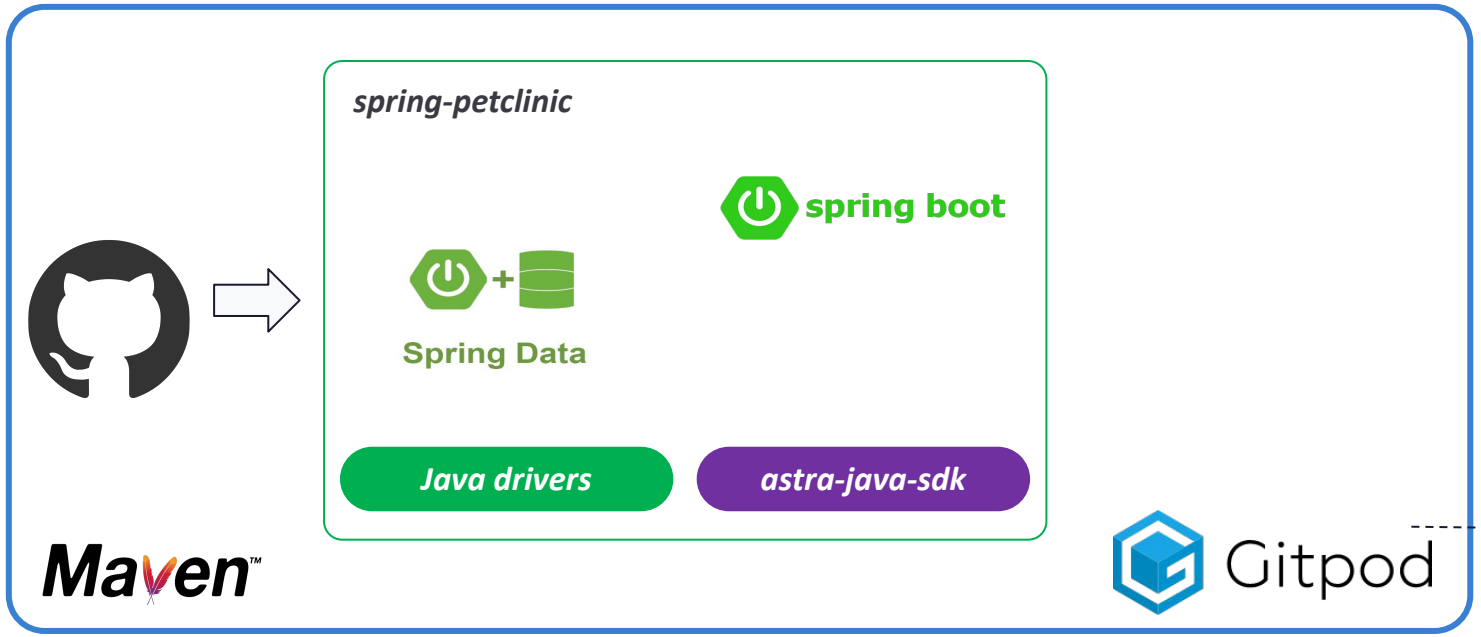
spring boot

https://spring.io/reactive

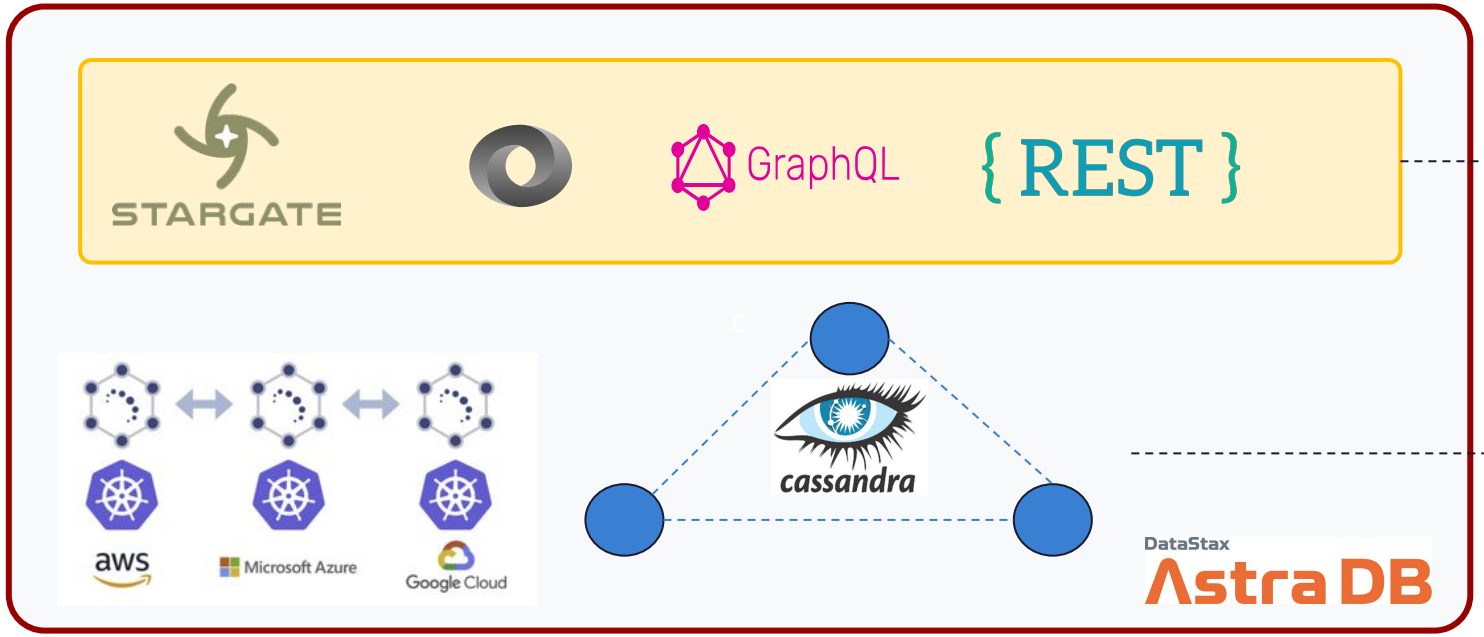


Spring Boot Application





IDE Gitpod.io
A free cloud-based IDE



OSS Stargate.io
A data gateway to allow multiple usages

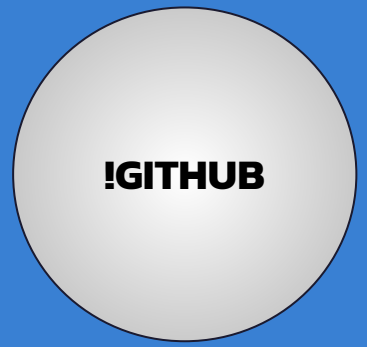
STARGATE

OSS Apache Cassandra
A Column oriented NoSQL Database

cassandra

HandsOn: 9 and 10

9. Working with Spring Data
10. Working with Spring WebFlux



Repository:

github.com/datastaxdevs/workshop-spring-reactive

DataStax Developers

Agenda

01

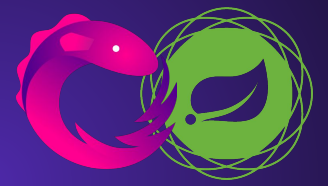
PetClinic
Architecture & Use Case

02

DataStax Astra

Cassandra Database
The Art of Data Modelling

03



Spring Reactive
Reactive vs Async

04

Spring Reactive
Boot and WebFlux

05

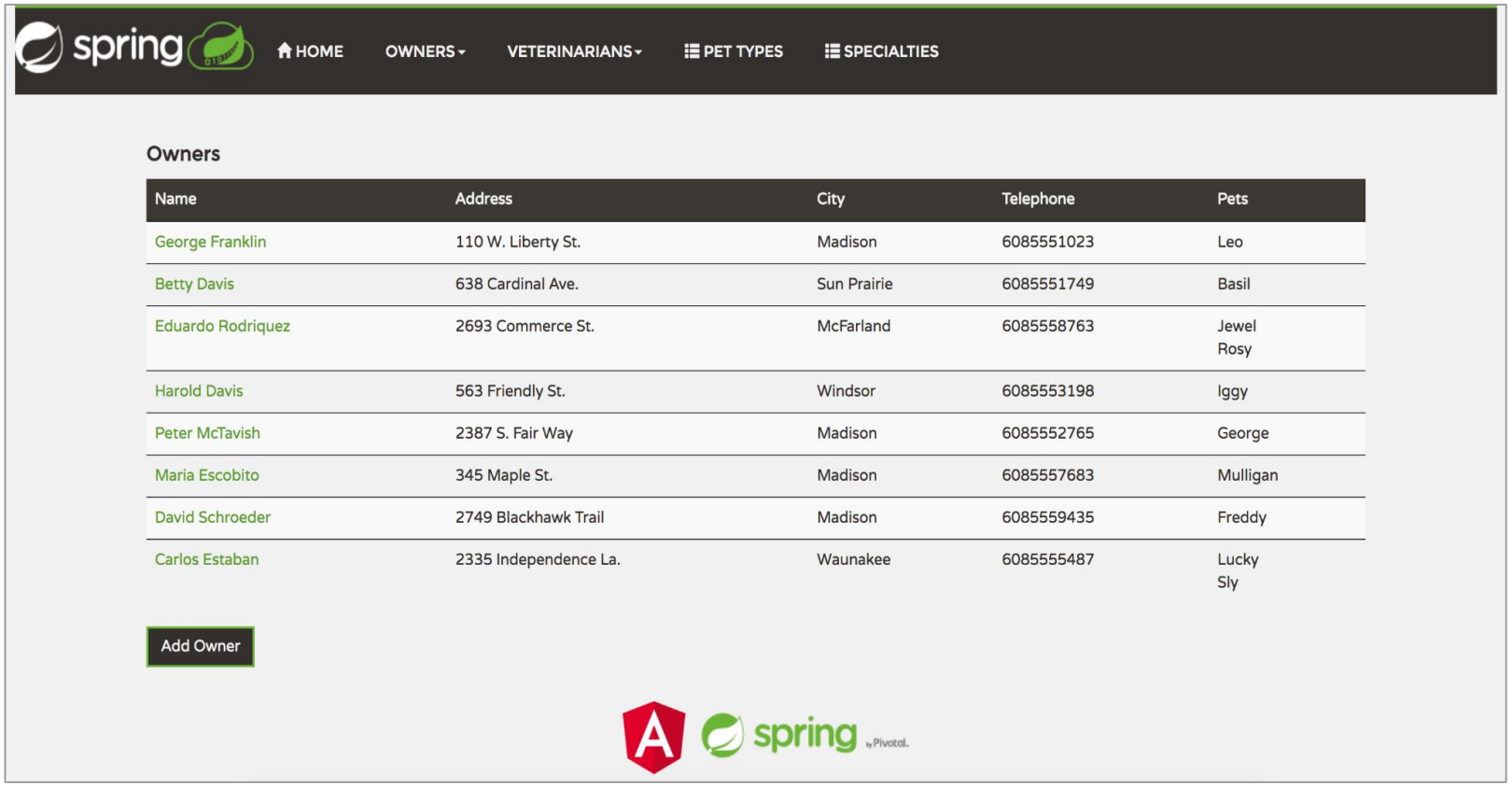
User Interface
Angular

06

Game &
Resources

Spring Pet Clinic Angular

https://github.com/spring-petclinic/spring-petclinic-angular



KARMA

Protractor
end to end testing for Angular

https://github.com/spring-petclinic/spring-petclinic-reactjs exists

HandsOn: 11

11. Working with Angular UI

!GITHUB

GitHub

Repository:

github.com/datastaxdevs/workshop-spring-reactive

DataStax Developers

Agenda

01

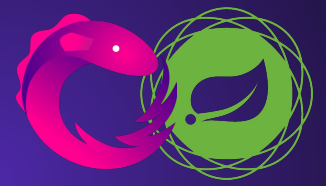
PetClinic
Architecture & Use Case

02

DataStax Astra

Cassandra Database
The Art of Data Modelling

03



Spring Reactive
Reactive vs Async

04

Spring Reactive
Boot and WebFlux

05

User Interface
Angular

06

Game &
Resources

menti.com

Mentimeter

Claim your FREE Certification Voucher 🎉

Vouchers (normally 145$ each)

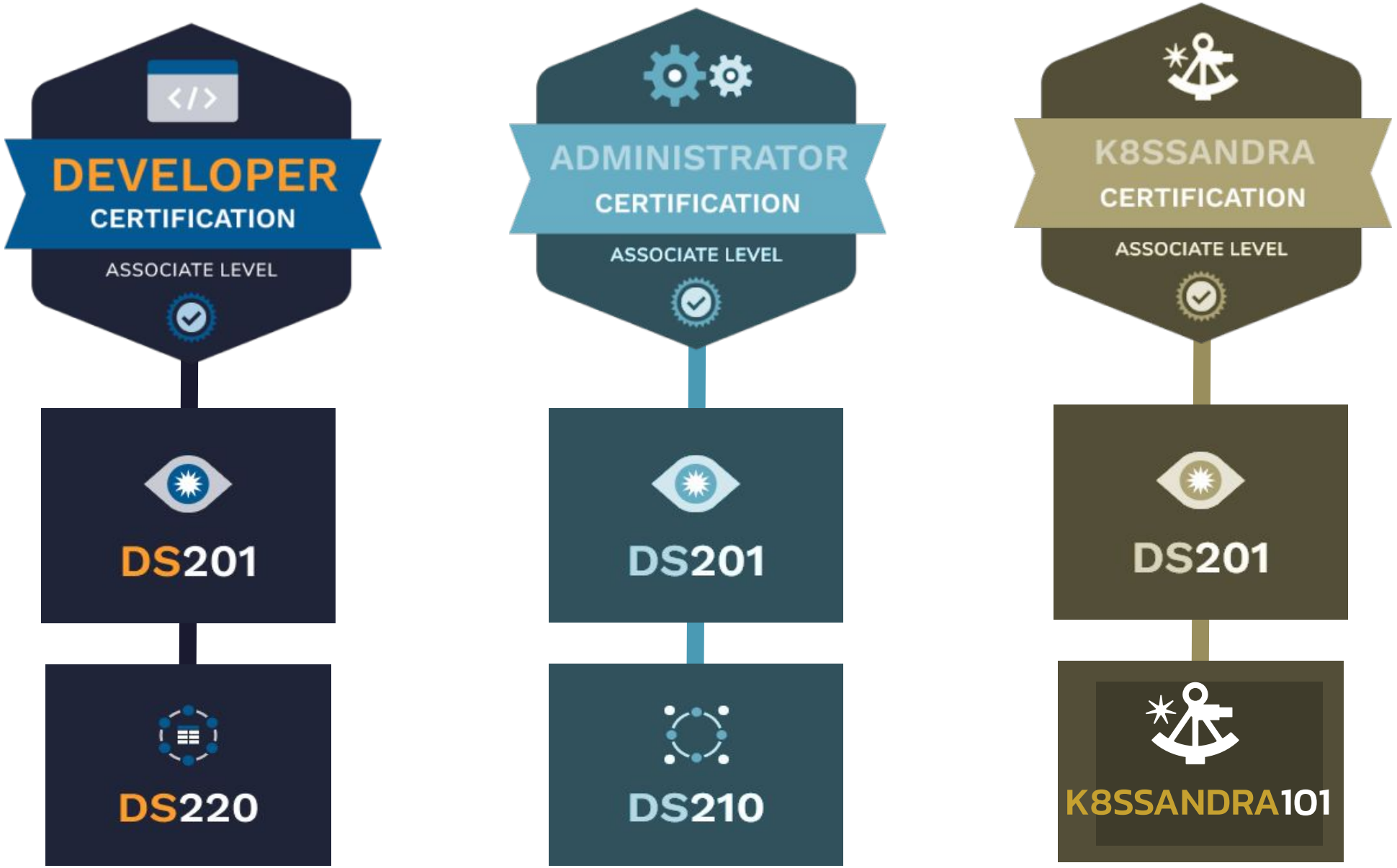
- valid for 3 months,
- valid for 2 attempts

https://www.datastax.com/dev/certifications

Claim using the link:

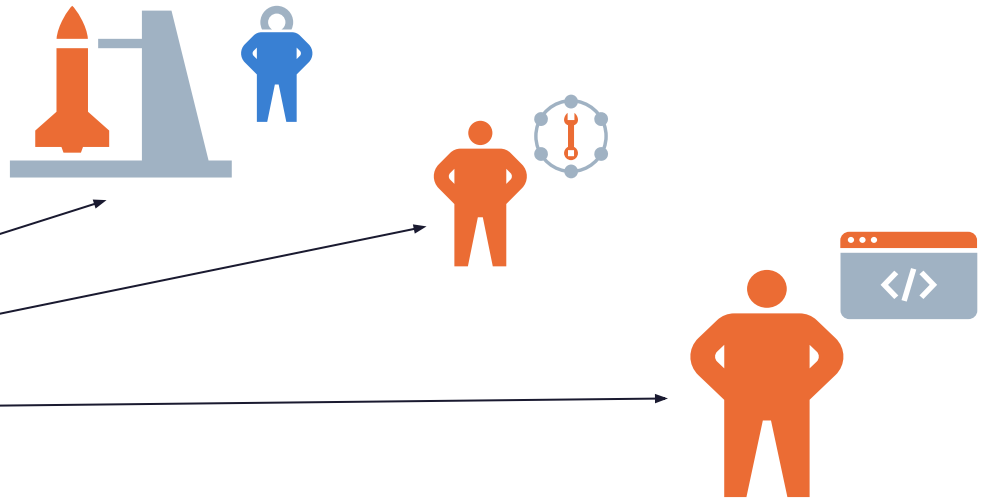
http://dtsx.io/workshop-voucher

LEARNING PATHS at the academy.datastax.com



https://dtsx.io/staxgiving

500$ + 250$ + 250$ AMAZON gift cards to win



CONDITIONS

- #engage-with-us : Tasks that will earn you 1 point
- #work-with-us: Tasks that will earn you 5 points
- #build : Tasks that will earn you 10 points.

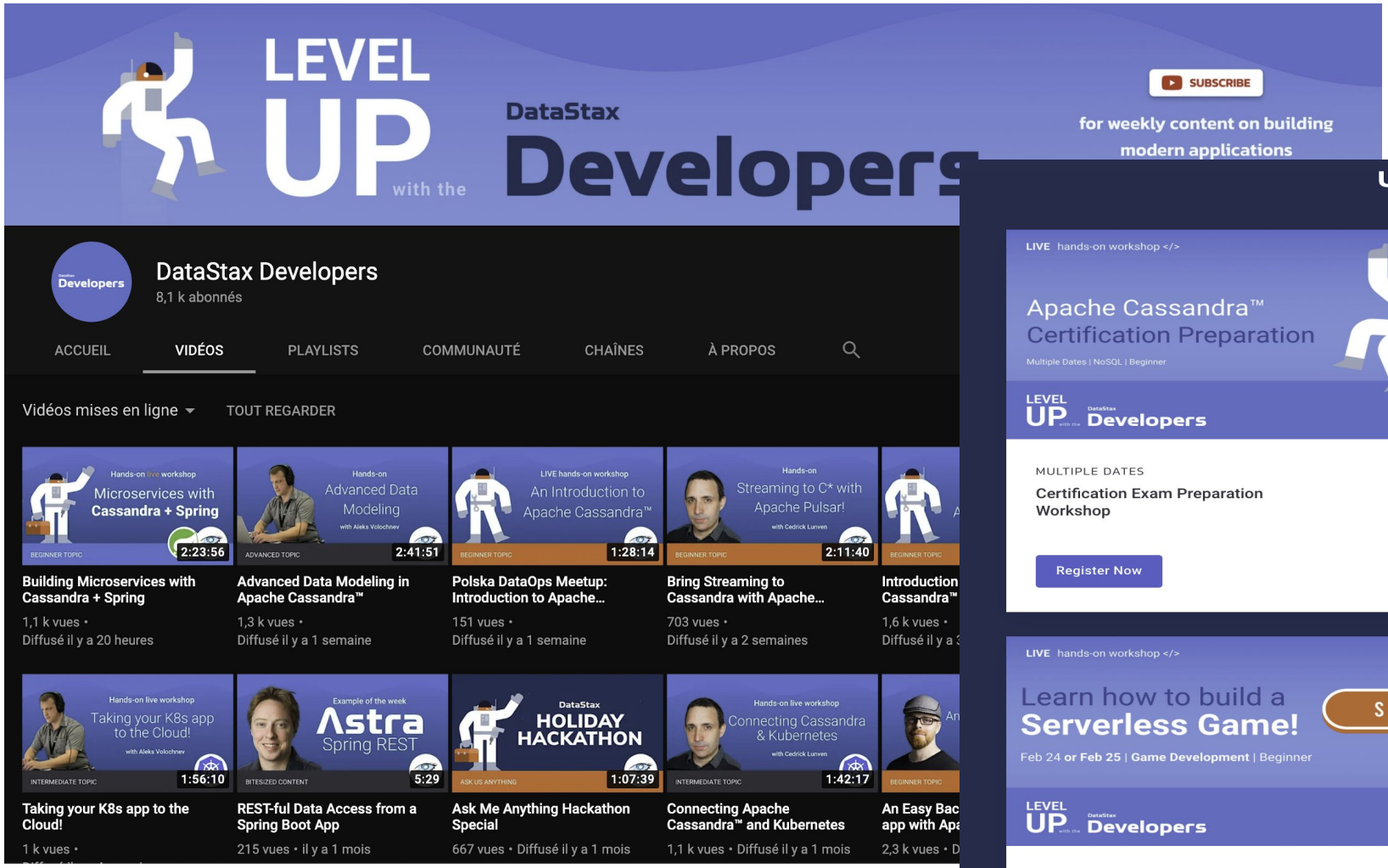
DUE DATE THANKSGIVING DAY => 25/11

Winners randomly chosen live on 12/8

DataStax Developers

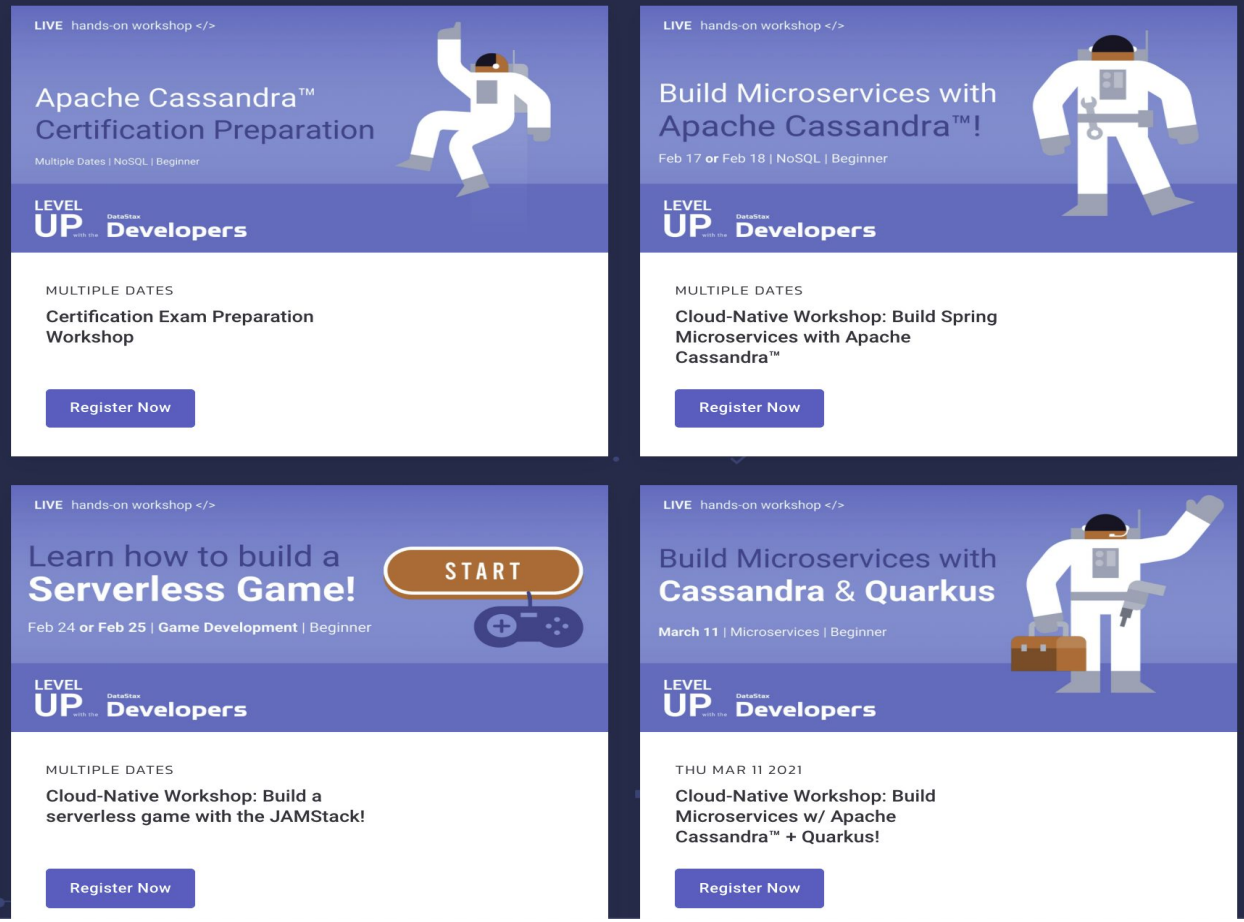
Weekly Workshops

datastax.com/workshops

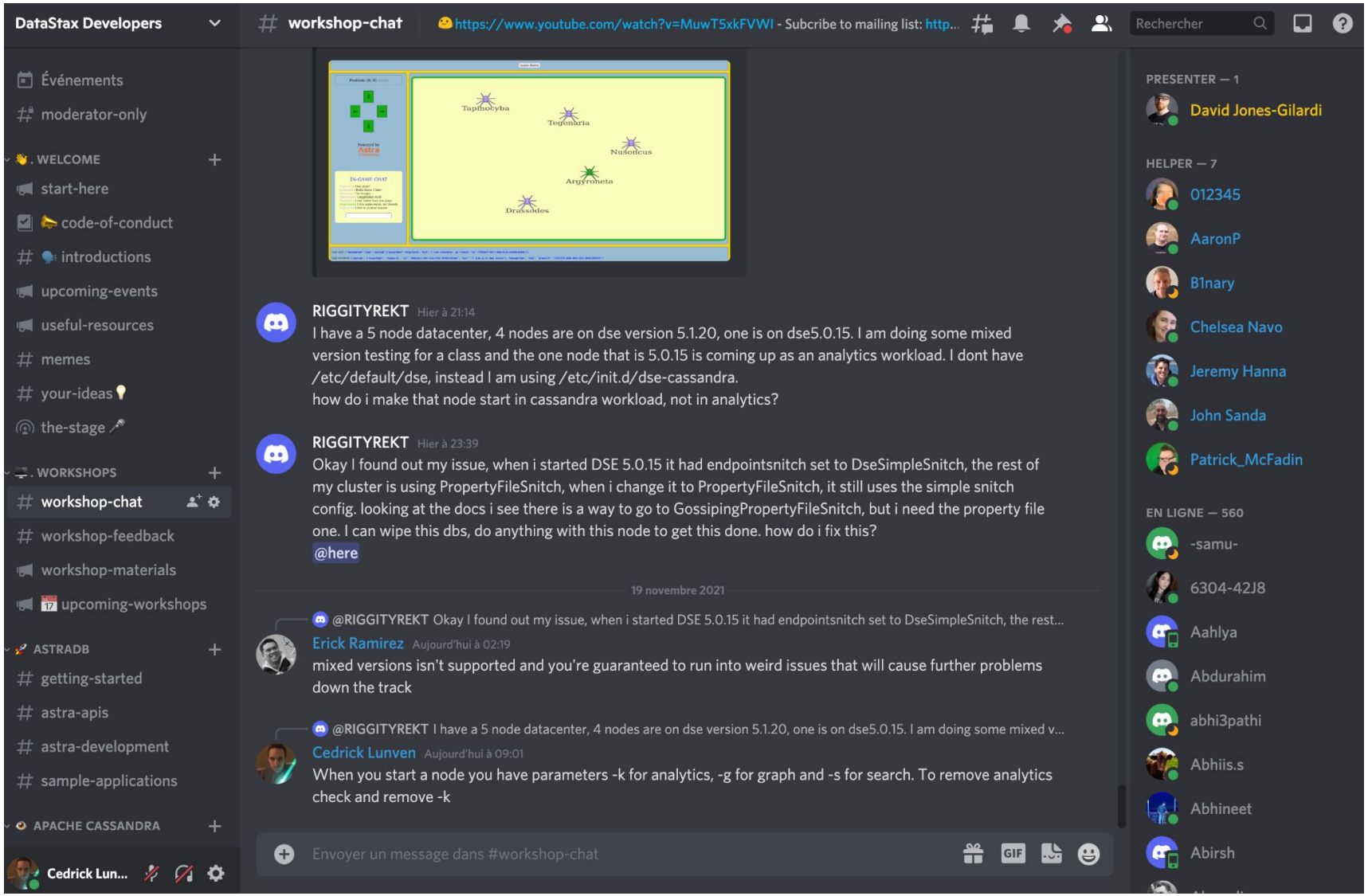


Subscribe

Upcoming Live Events



Join our 17k Discord Community
DataStax Developers

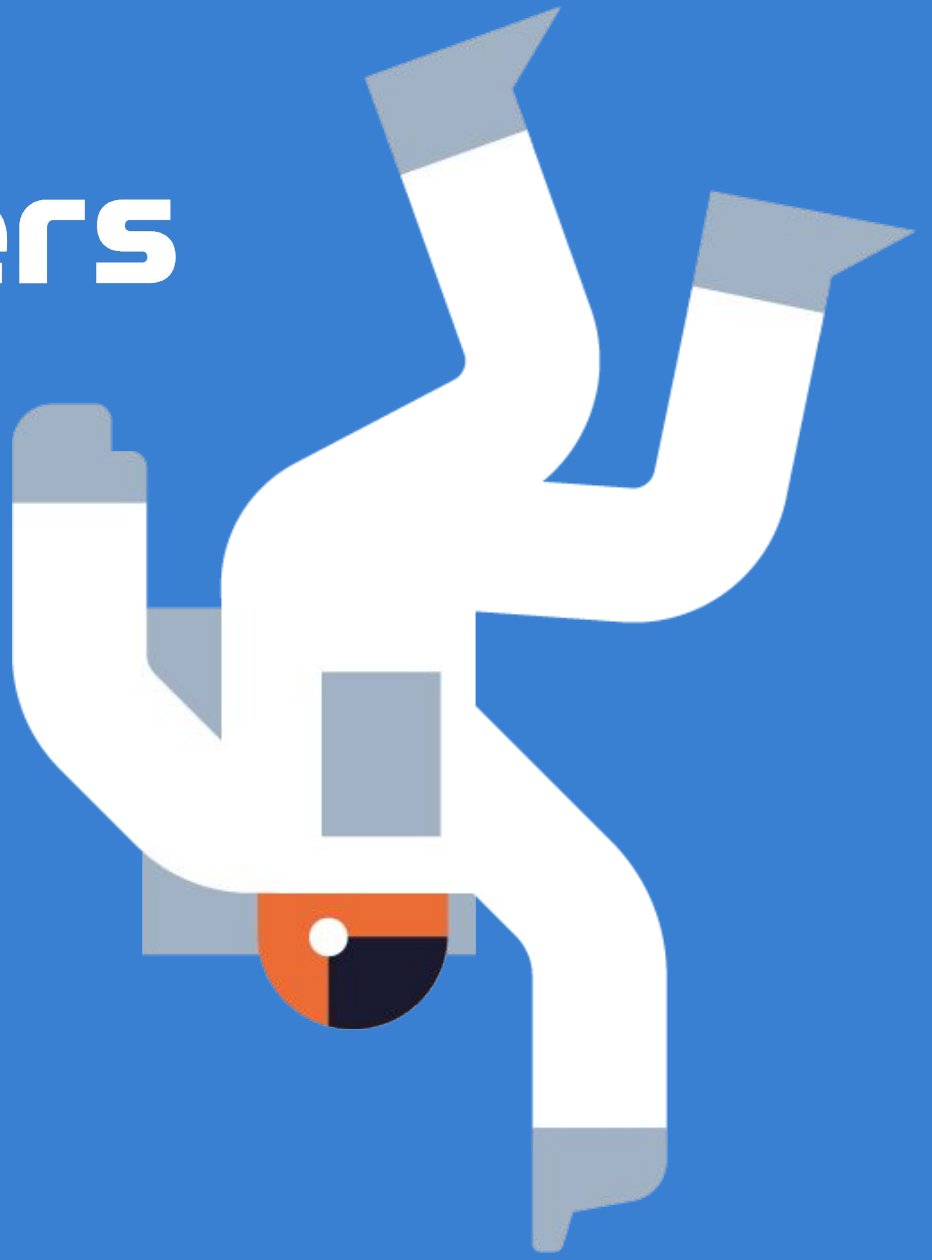


dtsx.io/discord

DataStax Developers

DataStax Developers

Thank you!



Subscribe

Streaming Engineer

Event Streaming Learning Series

Introduction to Pulsar

Event Streaming WORKSHOP

Data Engineer

Introduction to NoSQL WORKSHOP

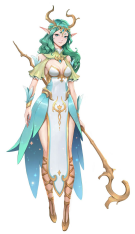
INTRODUCTION TO cassandra WORKSHOP

cassandra SQL to NoSQL WORKSHOP

Preparing Certification Workshop

cassandra DATA MODELING WORKSHOP

Machine Learning with C*



DEV FRONTEND

ReactNative WORKSHOP

Building a ToDO App WORKSHOP

GraphQL WORKSHOP

Netflix Clone WORKSHOP

Vue.JS WORKSHOP

TikTok Clone WORKSHOP

Advanced Doc. API Usage

K8SSANDRA WORKSHOP

Docker Kubernetes

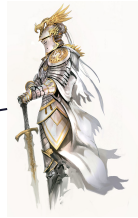
SRE

Twitter Clone

Python Flask

cassandra Quarkus + Cassandra WORKSHOP

Application DEV



DEV BACKEND

PetClinic Reactive

cassandra Spring Data + Cassandra WORKSHOP