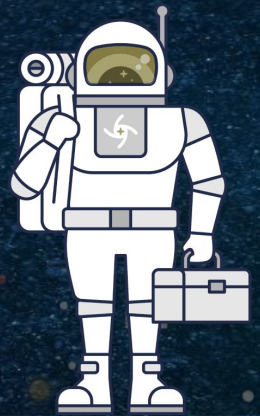
An OSS Api Layer for your Apache Cassandra™

STARGATE



DataStax Developers

Cédrick Lunven

Director Developer Relations

@clunven

@clun

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

SDK

- 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 Advocates Team

Cedrick Lunven

Aleksandr Volochnev

Jack Fryer

Kirsten Hunter

Stefano Lottini

David Gilardi

Ryan Welford

Rags Srinivas

Sonia Siganporia

R

S

Agenda (40 min)

01 STARGATE

Stargate Data gateway
What, Why and How

02 {REST}

Exploring Apis
Rest DML and DDL

03

Exploring Apis
Document oriented

04

Exploring Apis
GraphQL and Federation

05

Tooling
SDK, K8ssandra

05

What's NEXT ?
gRPC, CDC, sql...

Agenda (40 min)

01 STARGATE

Stargate Data gateway
What, Why and How

02 {REST}

Exploring Apis
Rest DML and DDL

03

Exploring Apis
Document oriented

04

Exploring Apis
GraphQL and Federation

05

Tooling
SDK, K8ssandra

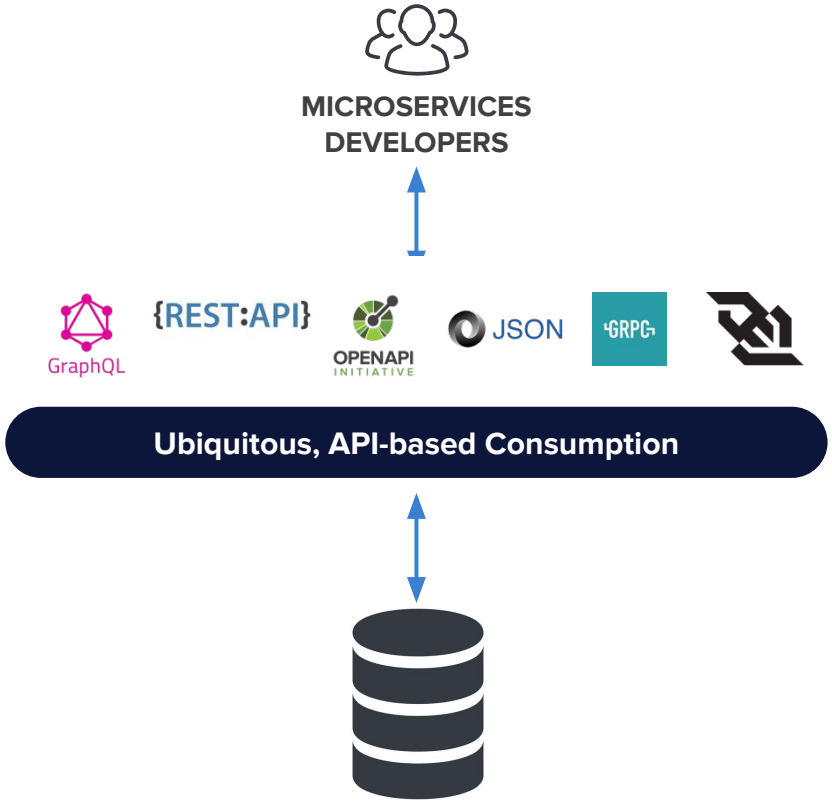
05

What's NEXT ?
gRPC, CDC, sql...

Data Gateway

Developers want the option to use modern APIs and development gateways.

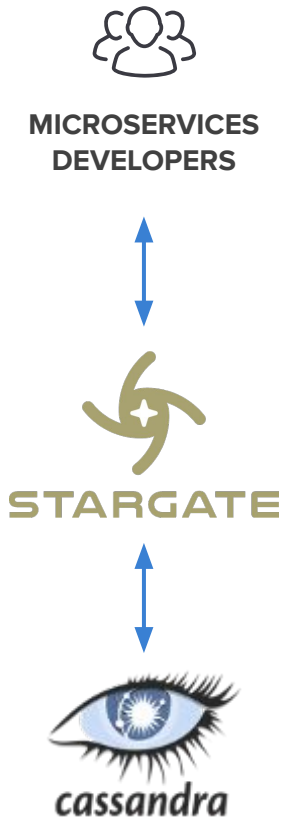
Cassandra is a database designed for the new standard and the associated APIs that facilitate the first choice of developers.



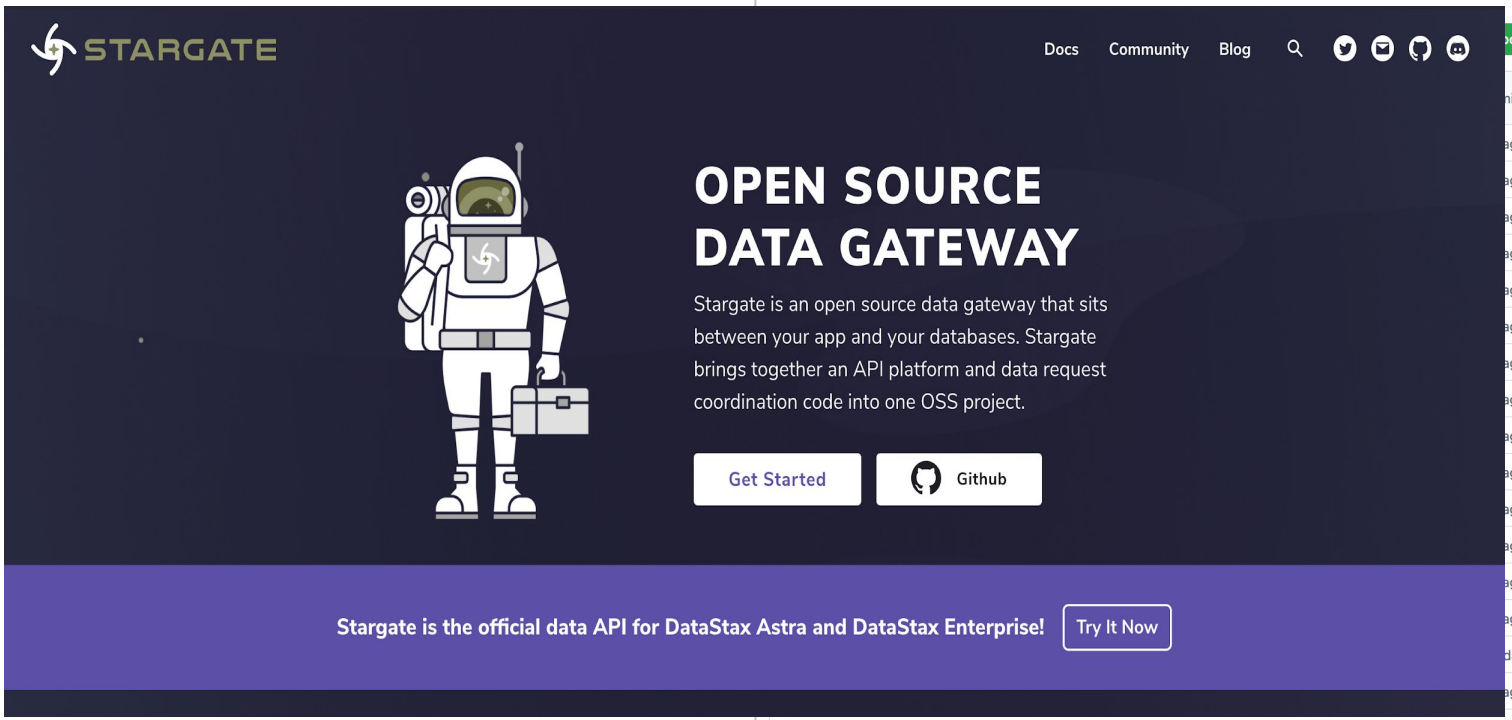
Stargate Overview

An open source API framework for data

Stargate makes it easy to use a database for any application workload by adding plugin support for new APIs, data types, and access methods

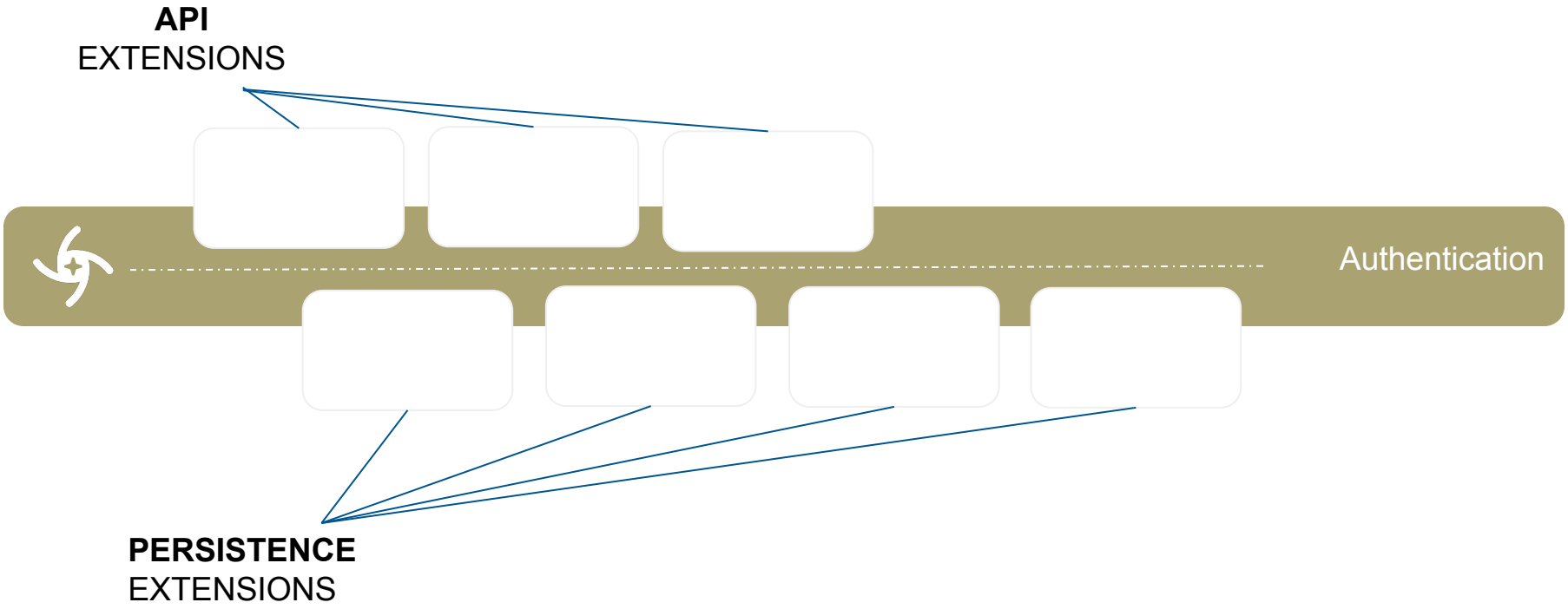


Stargate.io

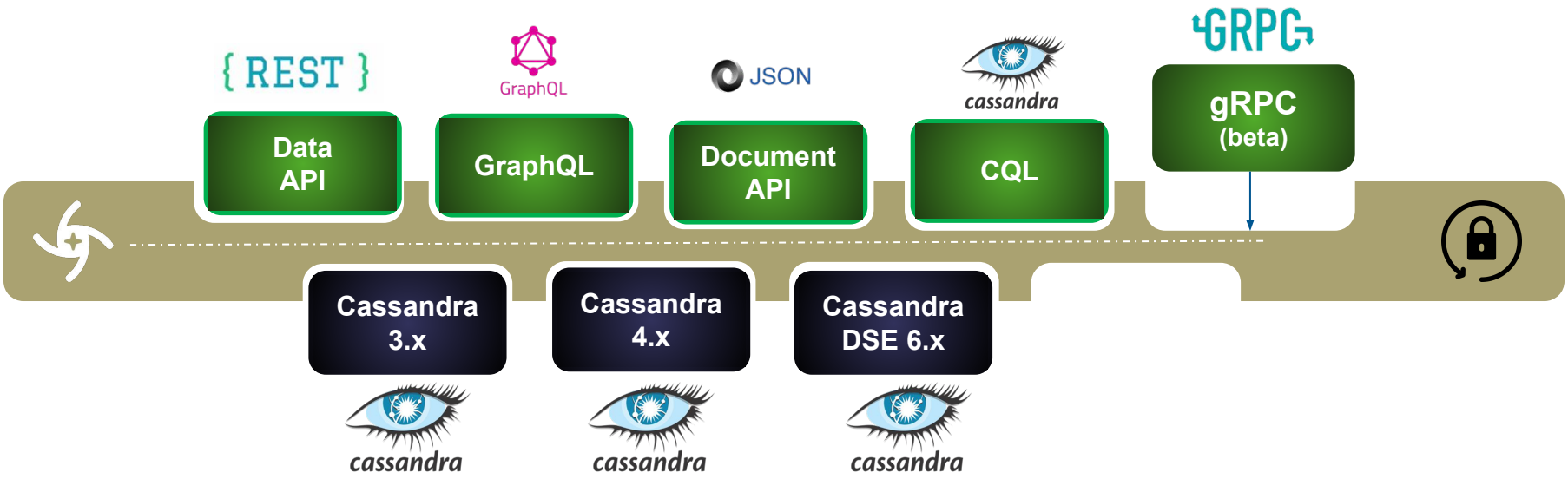

STARGATE
Docs
Community
Blog
OPEN SOURCE
DATA GATEWAY
Stargate is an open source data gateway that sits between your app and your databases. Stargate brings together an API platform and data request coordination code into one OSS project.
Get Started
Github
Stargate is the official data API for DataStax Astra and DataStax Enterprise!
Try It Now

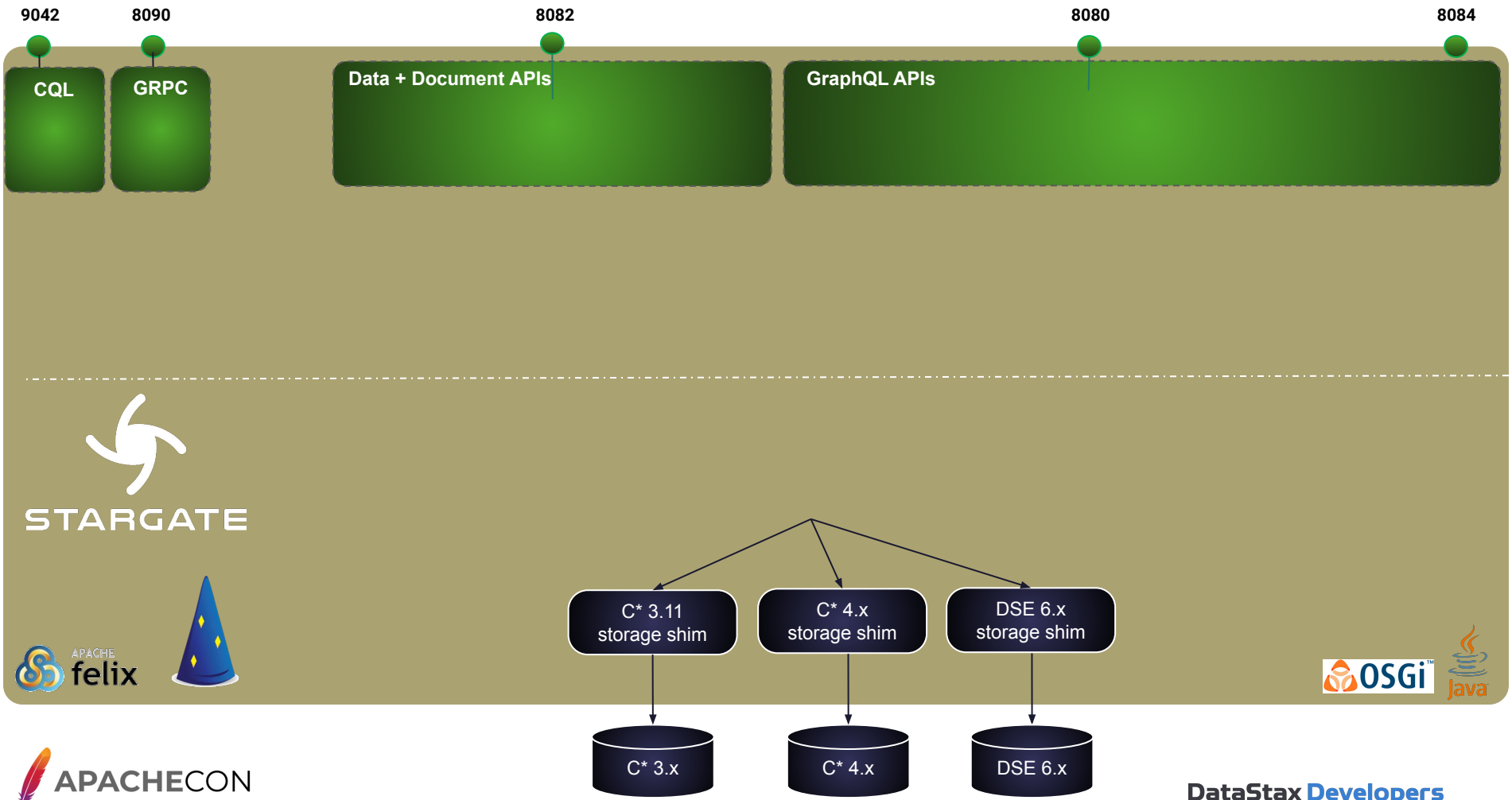
stargate / stargate
Public
Watch 22
Unstar 506
Fork 65
Code
Issues 246
Pull requests 13
Discussions
Actions
Projects
Wiki
About
An open source data gateway
stargate.io
java
graphql
rest
cassandra
cql
Readme
Apache-2.0 License
Releases 62
Release v1.0.32
Latest
25 days ago
+ 61 releases
Packages 13
io.stargate.stargate
io.stargate.db.persistence-api
io.stargate.db.persistence-common
+ 10 packages
Contributors 30
grpc
Reference example for gRPC java client (#1236)
yesterday

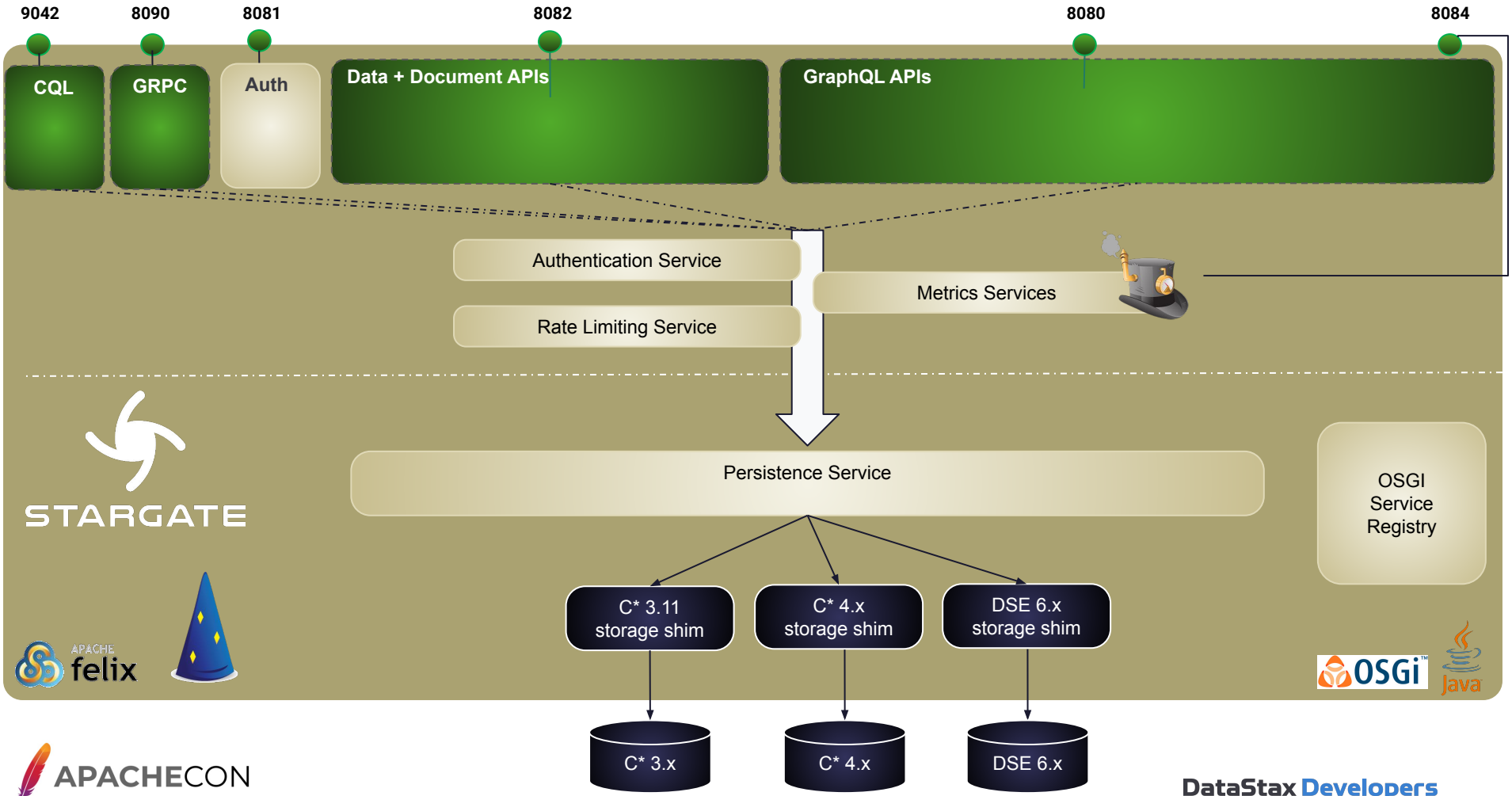
API and Persistence Extensions

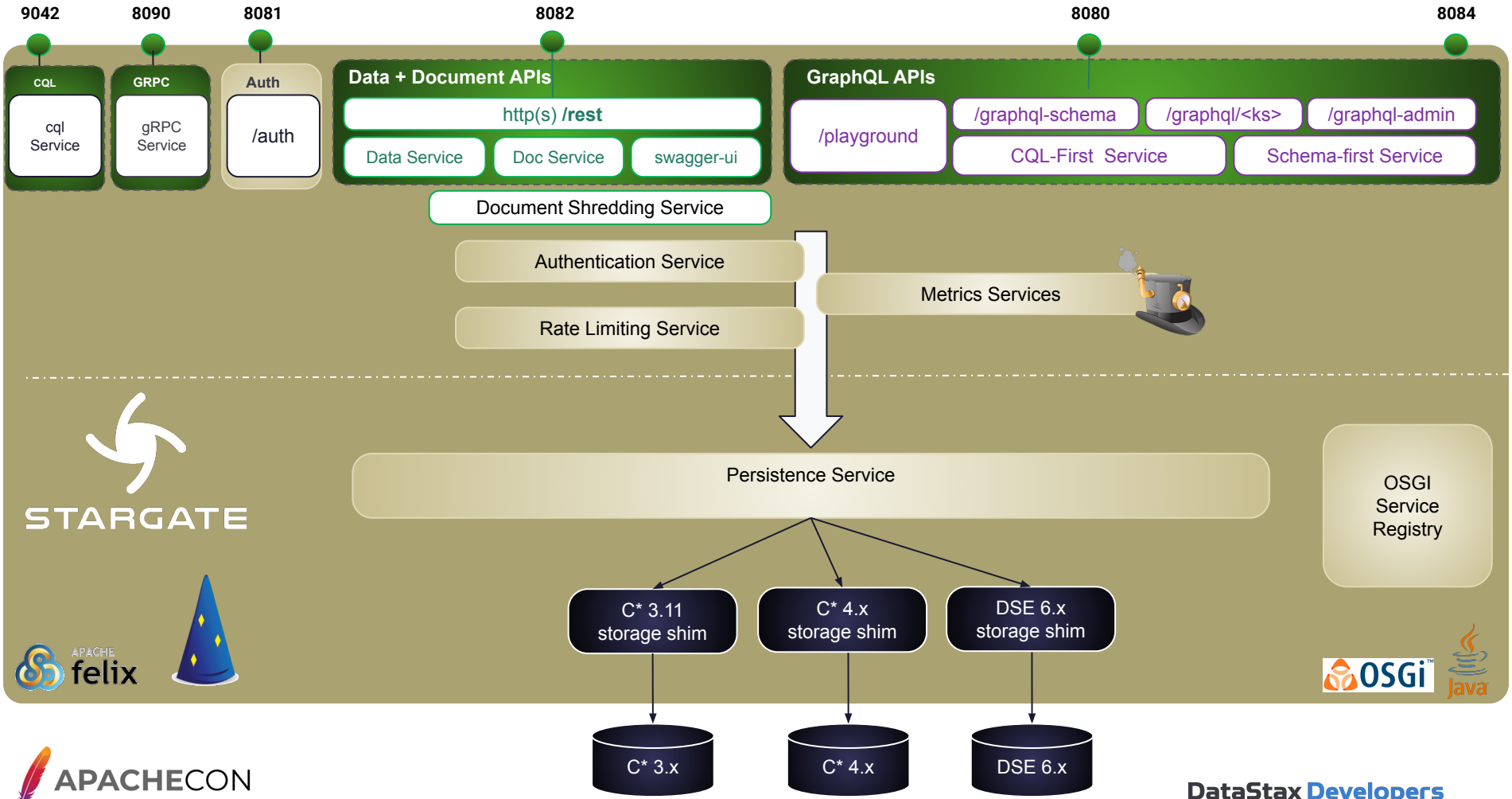


API Extensions and Persistence Extensions




9042
8090
8082
8080
8084
CQL
GRPC
Data + Document APIs
GraphQL APIs
STARGATE
C* 3.11
storage shim
C* 4.x
storage shim
DSE 6.x
storage shim
C* 3.x
C* 4.x
DSE 6.x
APACHE
felix
OSGi
Java

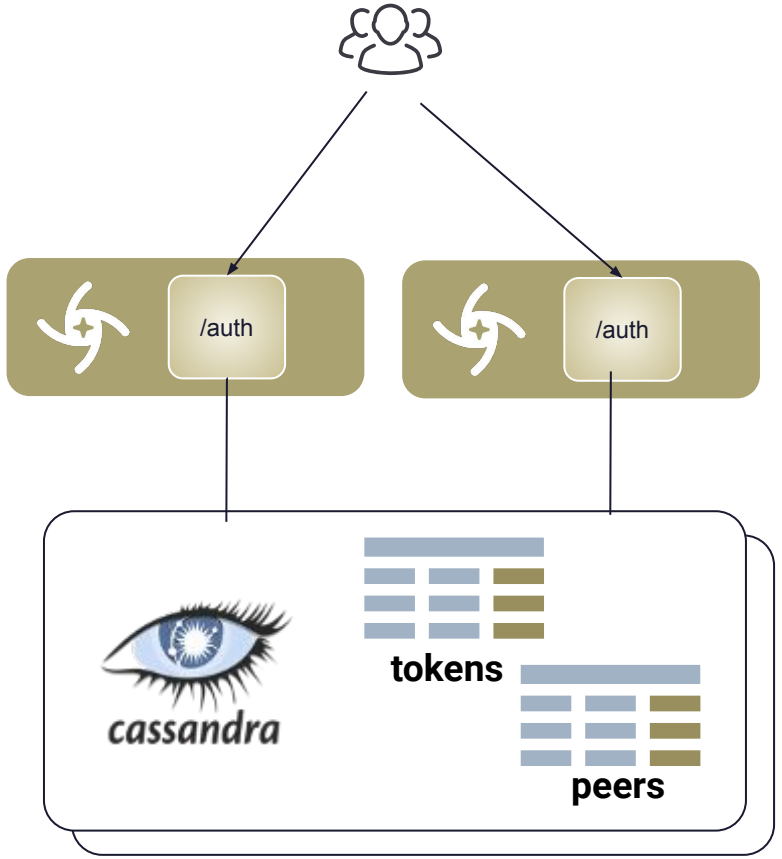

9042
8090
8081
8082
8080
8084
CQL
GRPC
Auth
Data + Document APIs
GraphQL APIs
Authentication Service
Rate Limiting Service
Metrics Services
STARGATE
Persistence Service
OSGI
Service
Registry
C* 3.11
storage shim
C* 4.x
storage shim
DSE 6.x
storage shim
APACHE
felix
OSGi
Java
C* 3.x
C* 4.x
DSE 6.x


9042
8090
8081
8082
8080
8084
CQL
cql Service
GRPC
gRPC Service
Auth
/auth
Data + Document APIs
http(s) /rest
Data Service
Doc Service
swagger-ui
GraphQL APIs
/playground
/graphql-schema
/graphql/<ks>
/graphql-admin
CQL-First Service
Schema-first Service
Document Shredding Service
Authentication Service
Rate Limiting Service
Metrics Services
STARGATE
Persistence Service
OSGI Service Registry
C* 3.11 storage shim
C* 4.x storage shim
DSE 6.x storage shim
C* 3.x
C* 4.x
DSE 6.x
APACHE felix
OSGi
Java
APACHECON
DataStax Developers

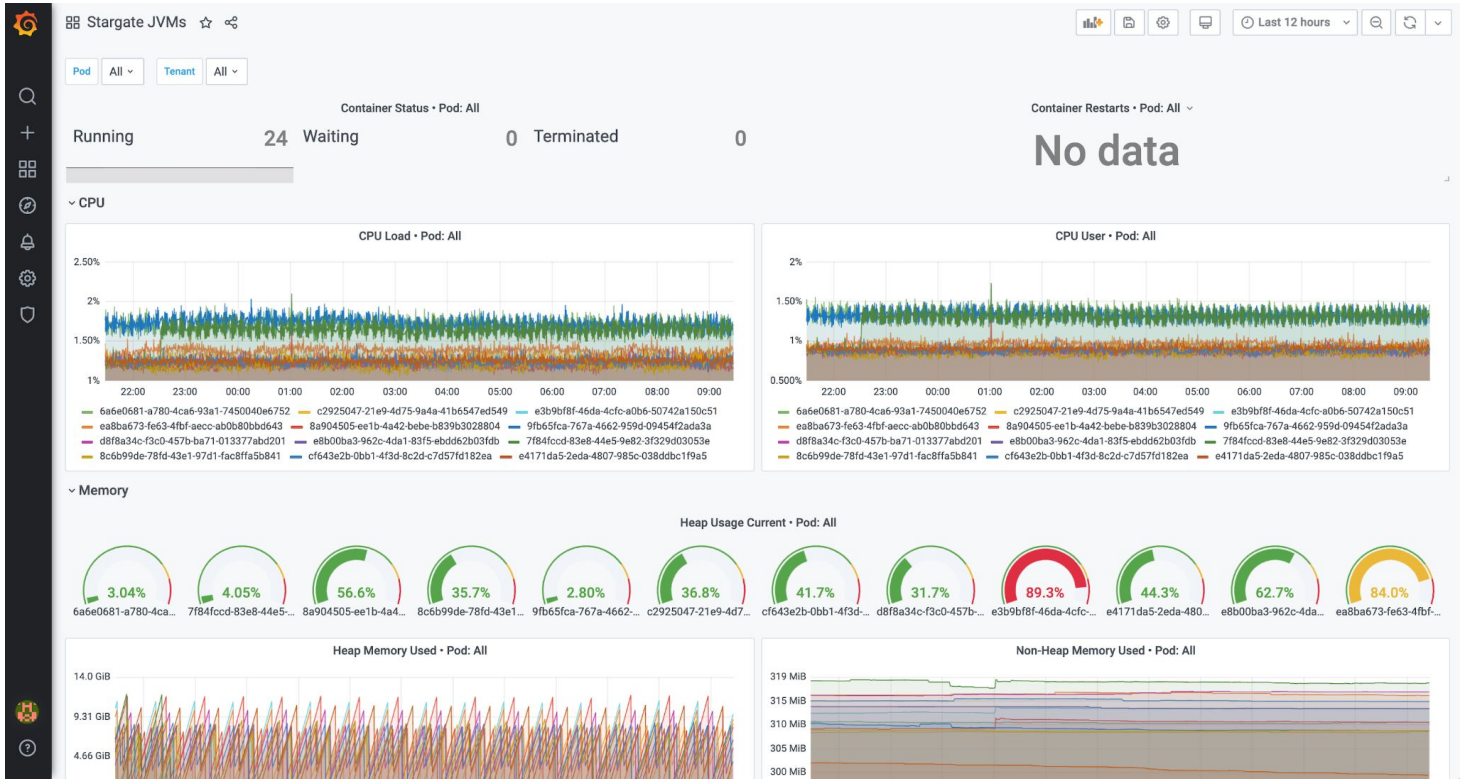
Authentication / Authorization

- Use Auth and get tokens (user+pwd)
- Tokens shared across instances
- Leverage on Cassandra roles (RBAC)
- `-Dstargate.auth_tokenttl` =X

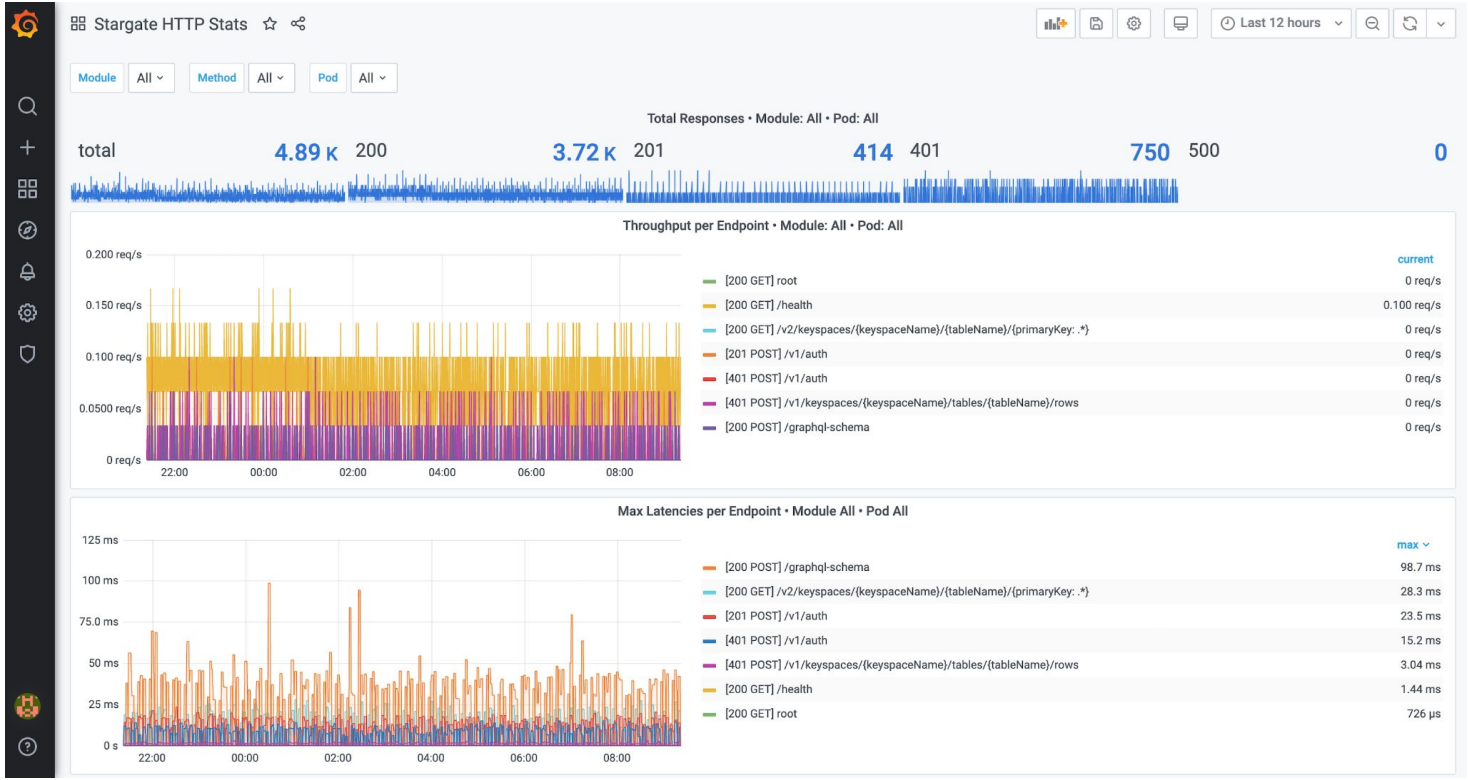
```
curl -L -X POST 'http://localhost:8081/v1/auth' \
  -H 'Content-Type: application/json' \
  --data-raw '{
    "username": "cassandra",
    "password": "cassandra"
}'
```



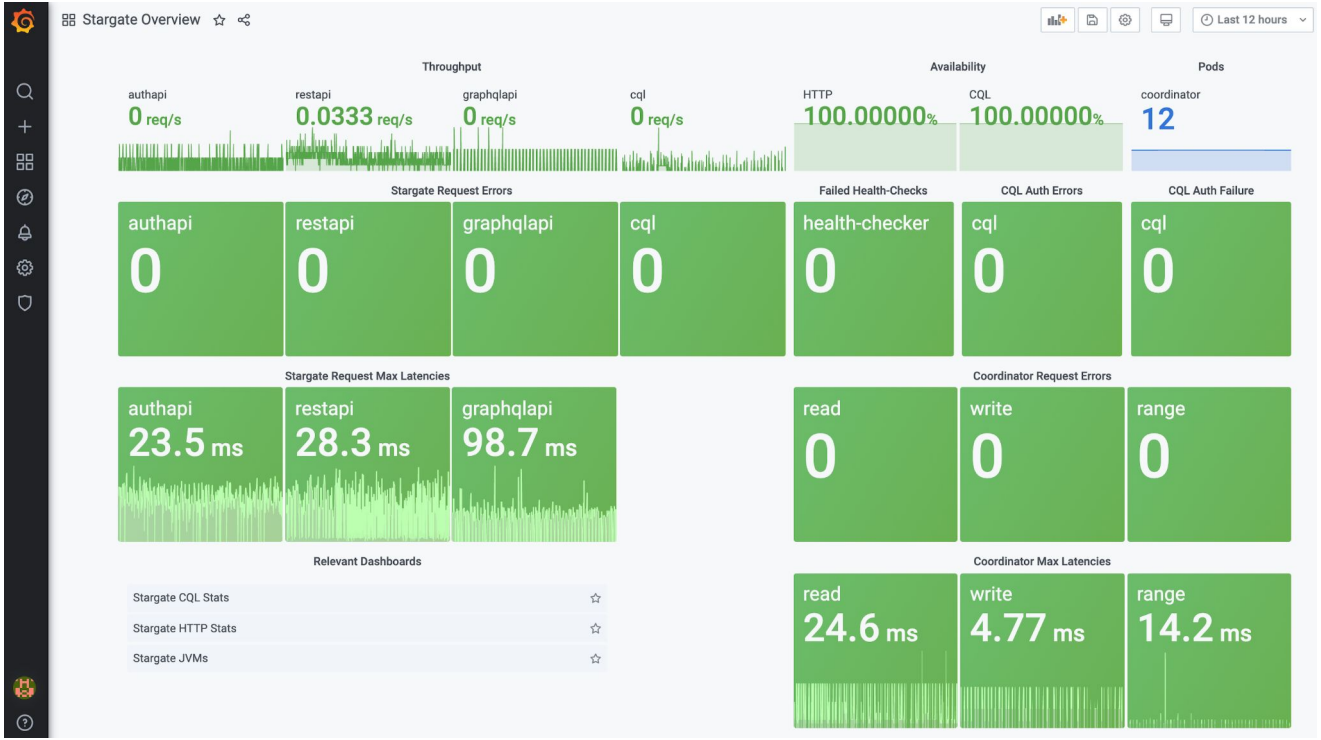
Monitoring (JVM)



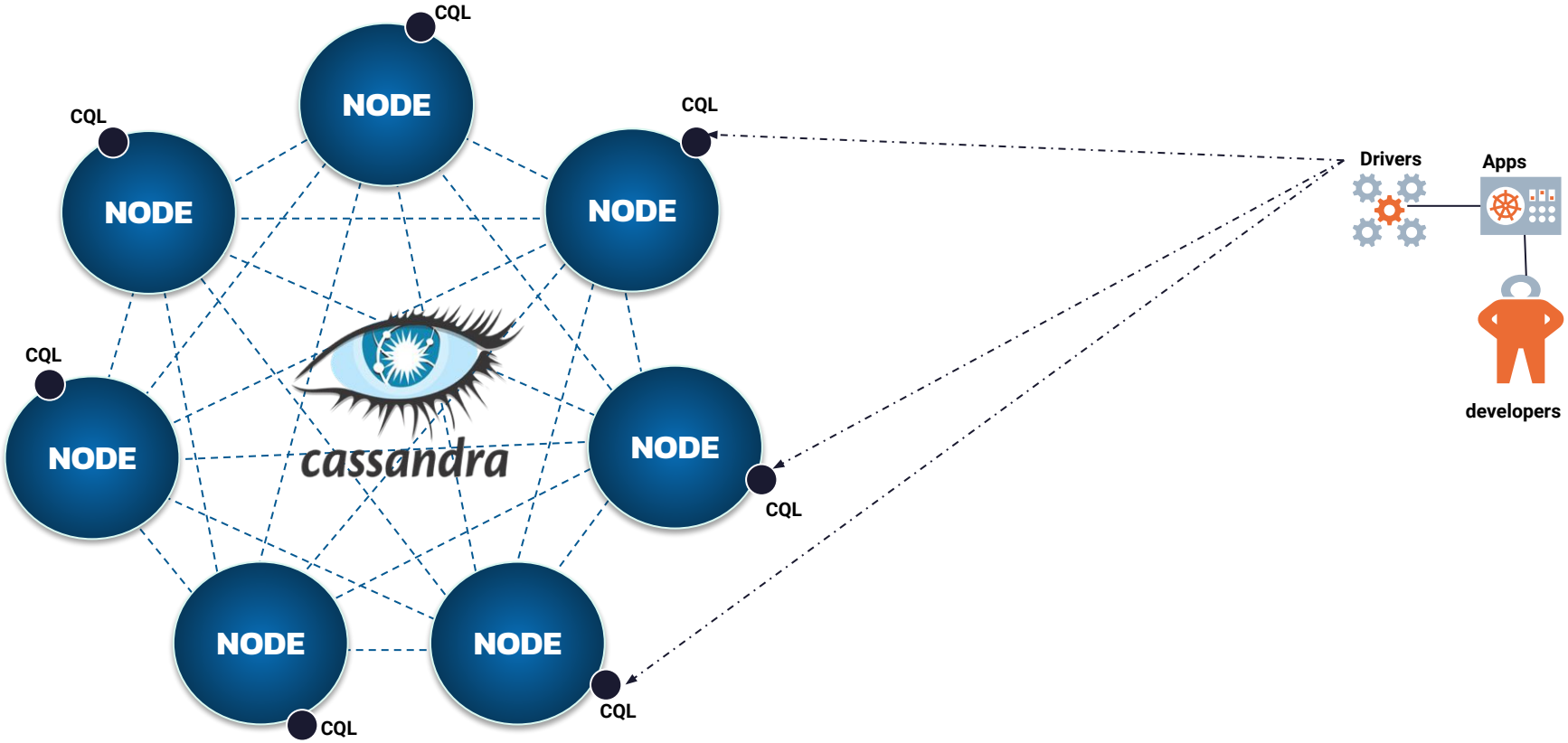
Monitoring (HTTP Traffic)



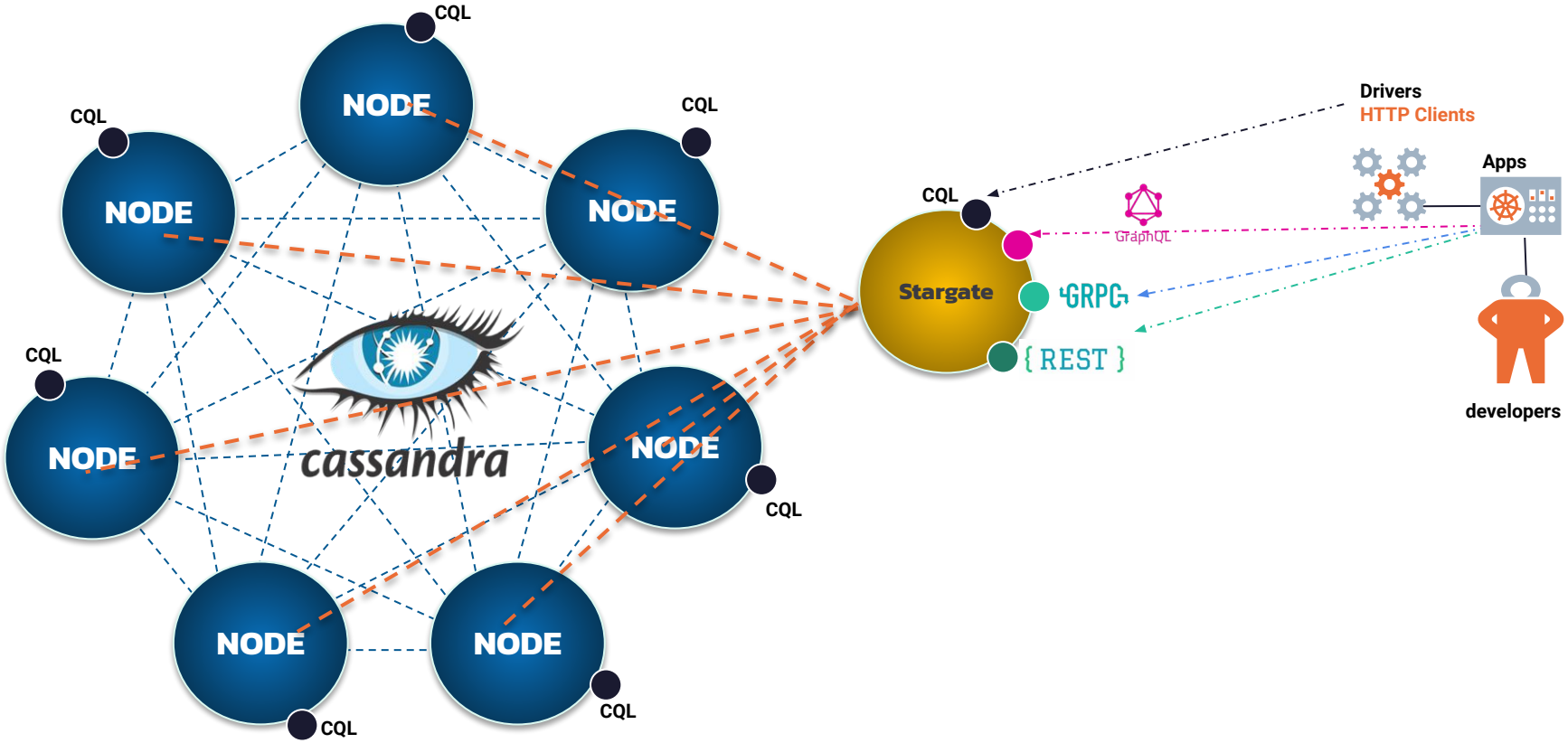
Monitoring (Services)



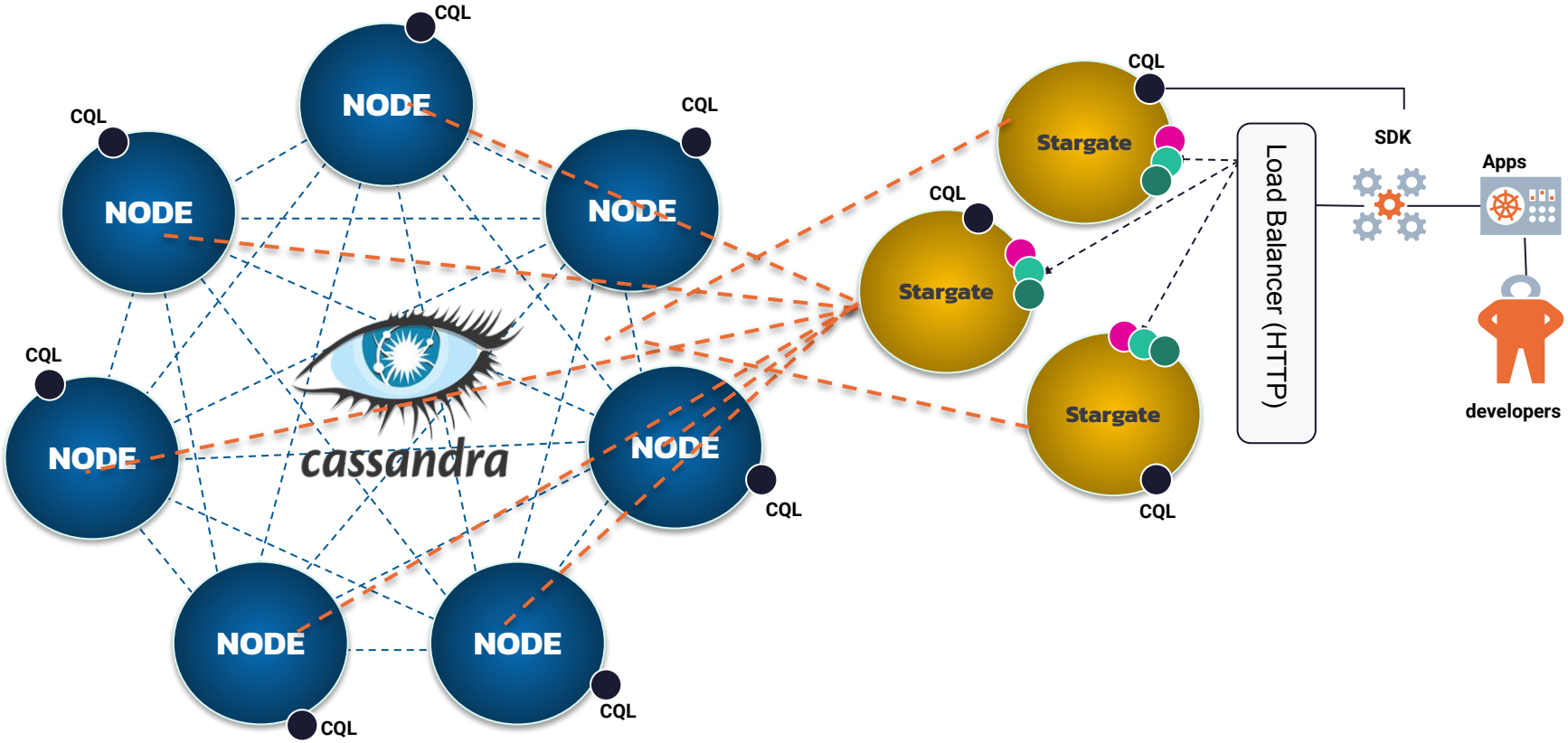
Connecting to your cluster (Before)



Connecting to your cluster (with Stargate)



Connecting to your cluster (with Stargates)



Agenda (40 min)

01 STARGATE

Stargate Data gateway
What, Why and How

02 {REST}

Exploring Apis
Rest DML and DDL

03

Exploring Apis
Document oriented

04

Exploring Apis
GraphQL and Federation

05

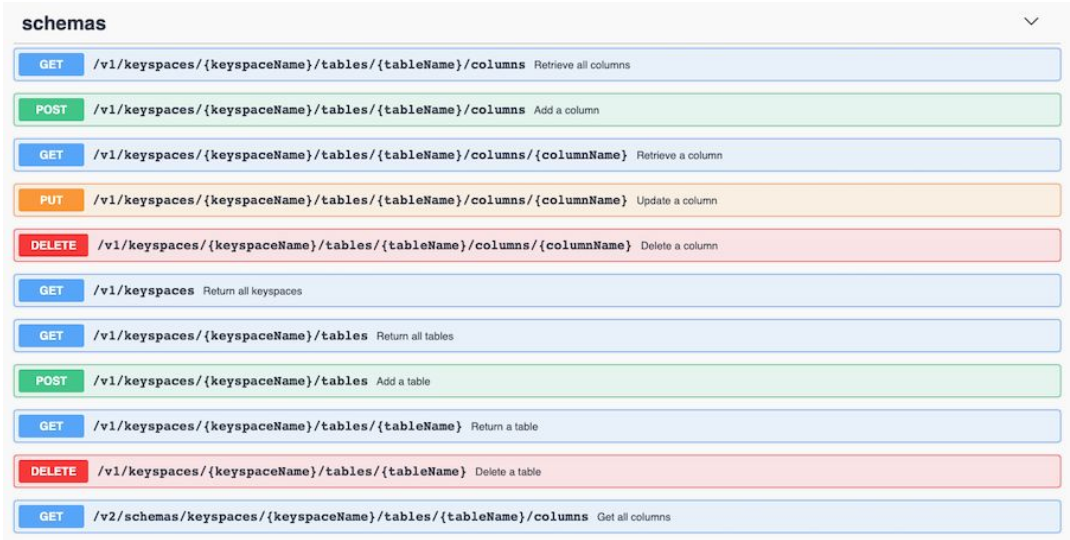
Tooling
SDK, K8ssandra

05

What's NEXT ?
gRPC, CDC, sql...

Stargate “REST” Api(s)

- Expose existing CQL schema as a RESTful endpoint, or create new CQL schema
- Most familiar API style for most teams
- Swagger integration



Quick start: https://stargate.io/docs/stargate/1.0/quickstart/quick_start-rest.html

REST Api(s) exposed DDL and best practices DML

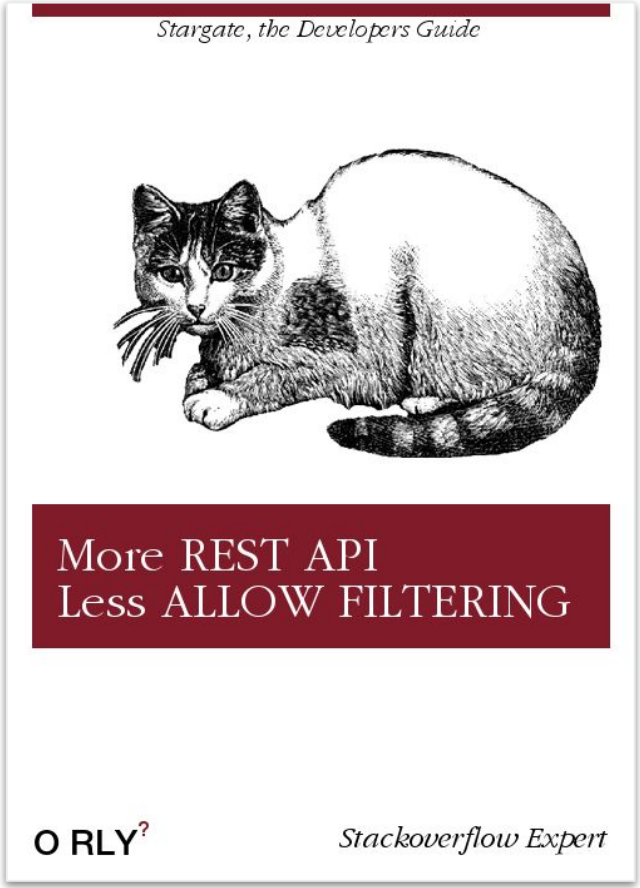
GET	/v2/schemas/keyspaces/{keyspaceName}/tables/{tableName}/columns	Get all columns
POST	/v2/schemas/keyspaces/{keyspaceName}/tables/{tableName}/columns	Create a column
GET	/v2/schemas/keyspaces/{keyspaceName}/tables/{tableName}/columns/{columnName}	Get a column
PUT	/v2/schemas/keyspaces/{keyspaceName}/tables/{tableName}/columns/{columnName}	Update a column
DELETE	/v2/schemas/keyspaces/{keyspaceName}/tables/{tableName}/columns/{columnName}	Delete a column
DELETE	/v2/schemas/keyspaces/{keyspaceName}/tables/{tableName}/indexes/{indexName}	Drop an index from keyspace
GET	/v2/schemas/keyspaces/{keyspaceName}/tables/{tableName}/indexes	Get all indexes for a given table
POST	/v2/schemas/keyspaces/{keyspaceName}/tables/{tableName}/indexes	Add an index to a table's column
GET	/v2/schemas/keyspaces/{keyspaceName}	Get a keyspace
DELETE	/v2/schemas/keyspaces/{keyspaceName}	Delete a keyspace
GET	/v2/schemas/keyspaces	Get all keyspaces
POST	/v2/schemas/keyspaces	Create a keyspace
GET	/v2/schemas/keyspaces/{keyspaceName}/tables/{tableName}	Get a table
PUT	/v2/schemas/keyspaces/{keyspaceName}/tables/{tableName}	Replace a table definition

DDL (Schema)

DML (data)

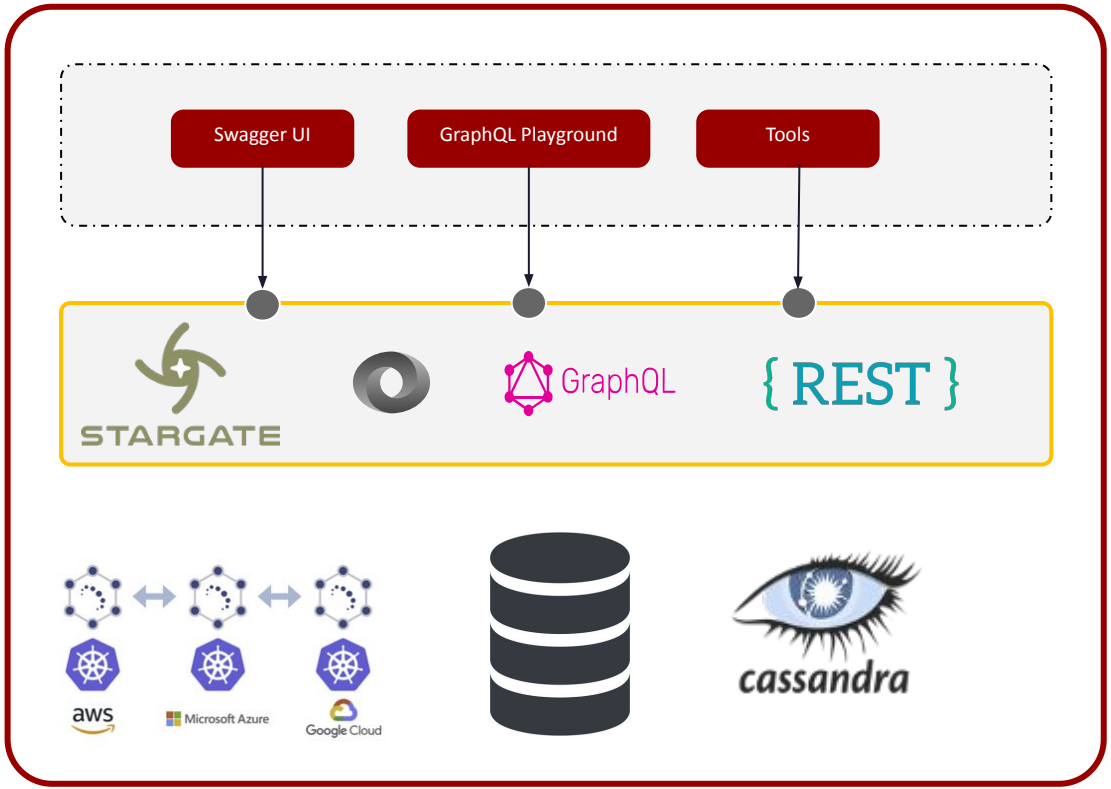
GET	/v2/keyspaces/{keyspaceName}/{tableName}/rows	Retrieve all rows
GET	/v2/keyspaces/{keyspaceName}/{tableName}	Search a table
POST	/v2/keyspaces/{keyspaceName}/{tableName}	Add row
GET	/v2/keyspaces/{keyspaceName}/{tableName}/{primaryKey}	Get row(s)
PUT	/v2/keyspaces/{keyspaceName}/{tableName}/{primaryKey}	Replace row(s)
DELETE	/v2/keyspaces/{keyspaceName}/{tableName}/{primaryKey}	Delete row(s)
PATCH	/v2/keyspaces/{keyspaceName}/{tableName}/{primaryKey}	Update part of a row(s)

Demo


Stargate, the Developers Guide
More REST API
Less ALLOW FILTERING
O RLY?
Stackoverflow Expert

DataStax

Astra DB



Agenda (40 min)

01 STARGATE

Stargate Data gateway
What, Why and How

02 {REST}

Exploring Apis
Rest DML and DDL

03

Exploring Apis
Document oriented

04

Exploring Apis
GraphQL and Federation

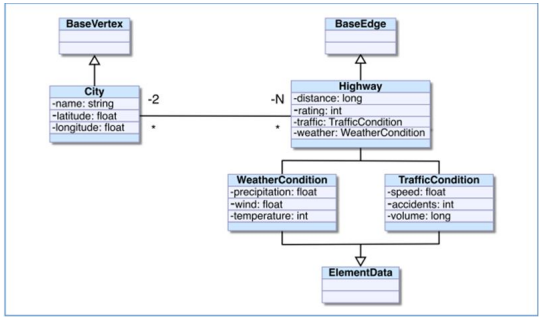
05

Tooling
SDK, K8ssandra

05

What's NEXT ?
gRPC, CDC, sql...

Working with JSON and Cassandra

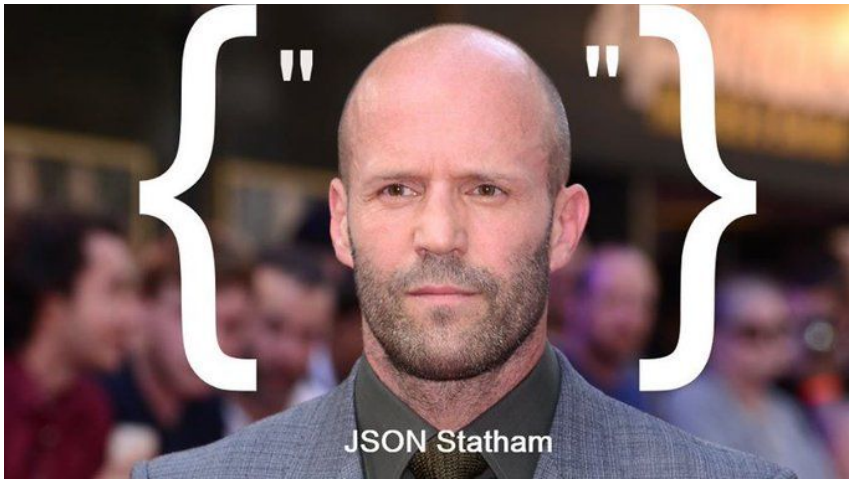


- Cassandra already handles both **JSON** and **nested objects**
  - INSERT JSON, SELECT JSON
  - Set<>, List<>,Map<>, and User Defined type (UDT) even nested
- But...
  - Strongly coupled with a SCHEMA VALIDATION
  - Dangerous with tombstones on updates

```
select json title,url,tags from videos;
```

```
INSERT INTO videos JSON '{
   "videoid":"e466f561-4ea4-4eb7-8dcc-126e0fbfd578",
     "email":"clunven@sample.com",
     "title":"A JSON videos",
     "upload":"2020-02-26 15:09:22 +00:00",
     "url": "http://google.fr",
     "frames": [1,2,3,4],
     "tags":   [ "cassandra","accelerate", "2020"]
}';
```

“Schemaless”

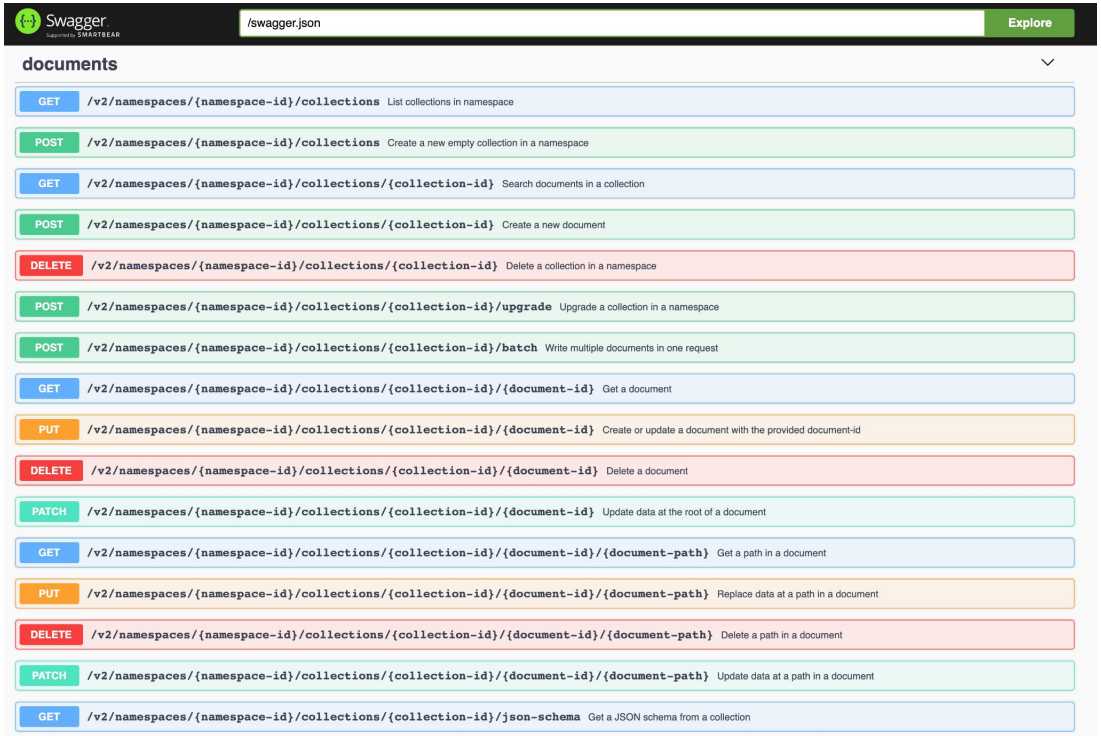


“What rules ?”

- You want to insert and retrieve any JSON documents efficiently
- Allow “schemaless” (Validation less)
- Write to a single document is a single batch of statements
- Read from a single document is a single SELECT statement.
- Limit Tombstones with range deletes

Stargate Document API

- Schemaless! (validationless)
- Add any JSON document to a collection
- Uses “Document shredding” approach



Quick start: https://stargate.io/docs/stargate/1.0/quickstart/quick_start-document.html

Document Shredding (1 / 3)

```
create table <name> (
  key text,
  p0 text,
  … p[N] text,
  bool_value boolean,
  txt_value text, d
  bl_value double, leaf text
)
```

Document Shredding (2 / 3)

```
{"a": { "b": 1 }, "c": 2}
```

The document would be “shredded” into rows looking like this:

key	p0	p1	dbl_value
x	a	b	1
x	c	null	2

Document Shredding (3 / 3)

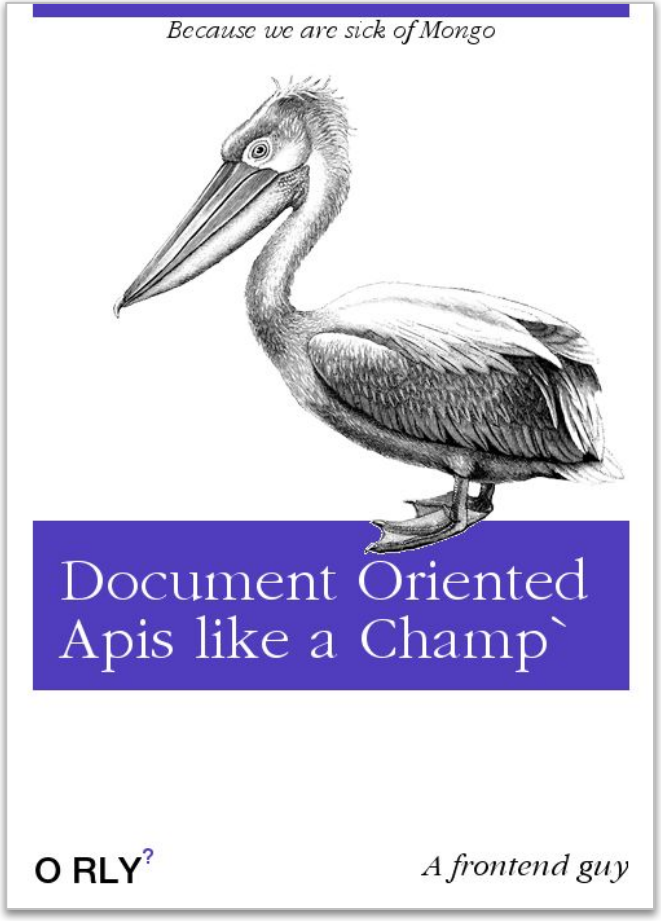
For data with an array, such as:

```
{"a": { "b": 1 }, "c": [{"d": 2}]}
```

there would be two rows, like so:

key	p0	p1	p2	dbl_value
x	a	b	null	1
x	c	[0]	d	2

Demo


Because we are sick of Mongo
Document Oriented Apis like a Champ`
O RLY?
A frontend guy

Agenda (40 min)

01 STARGATE

Stargate Data gateway
What, Why and How

02 {REST}

Exploring Apis
Rest DML and DDL

03

Exploring Apis
Document oriented

04

Exploring Apis
GraphQL and Federation

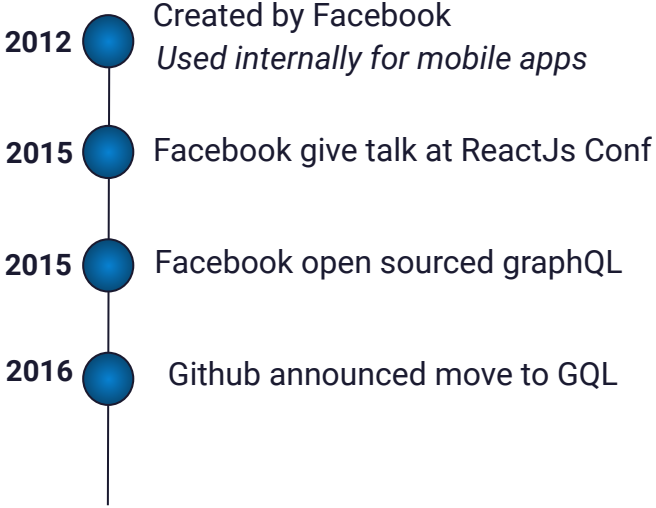
05

Tooling
SDK, K8ssandra

05

What's NEXT ?
gRPC, CDC, sql...

History



GraphQL

Definition

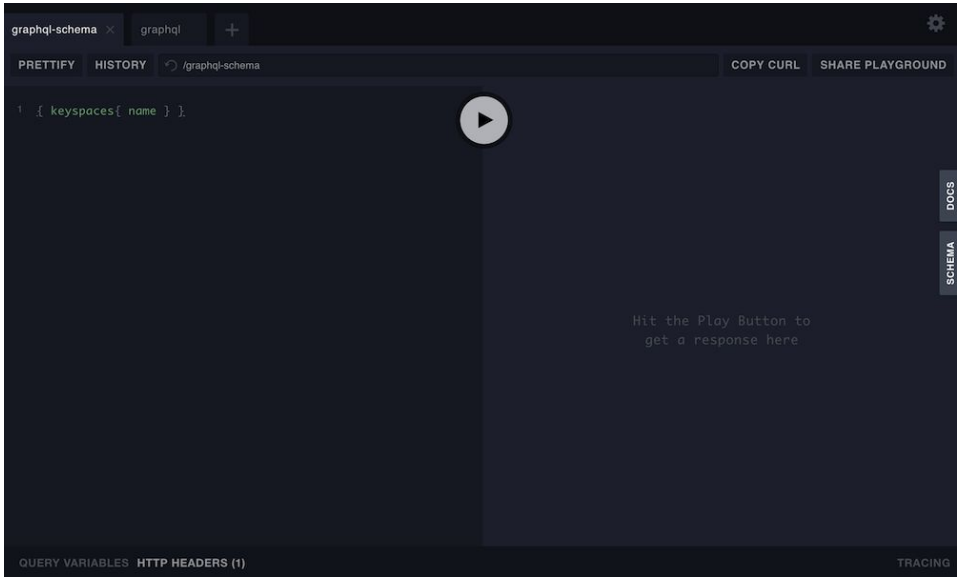
GraphQL is an application programming interface (API) query language and server-side runtime that prioritises giving customers precisely the data they request.

Why ?


GraphQL
Describe your data
type Project {
name: String
tagline: String
contributors: [User]
}
Ask for what you want
{
project(name: "GraphQL") {
tagline
}
}
Get predictable results
{
"project": {
"tagline": "A query language for APIs"
}
}

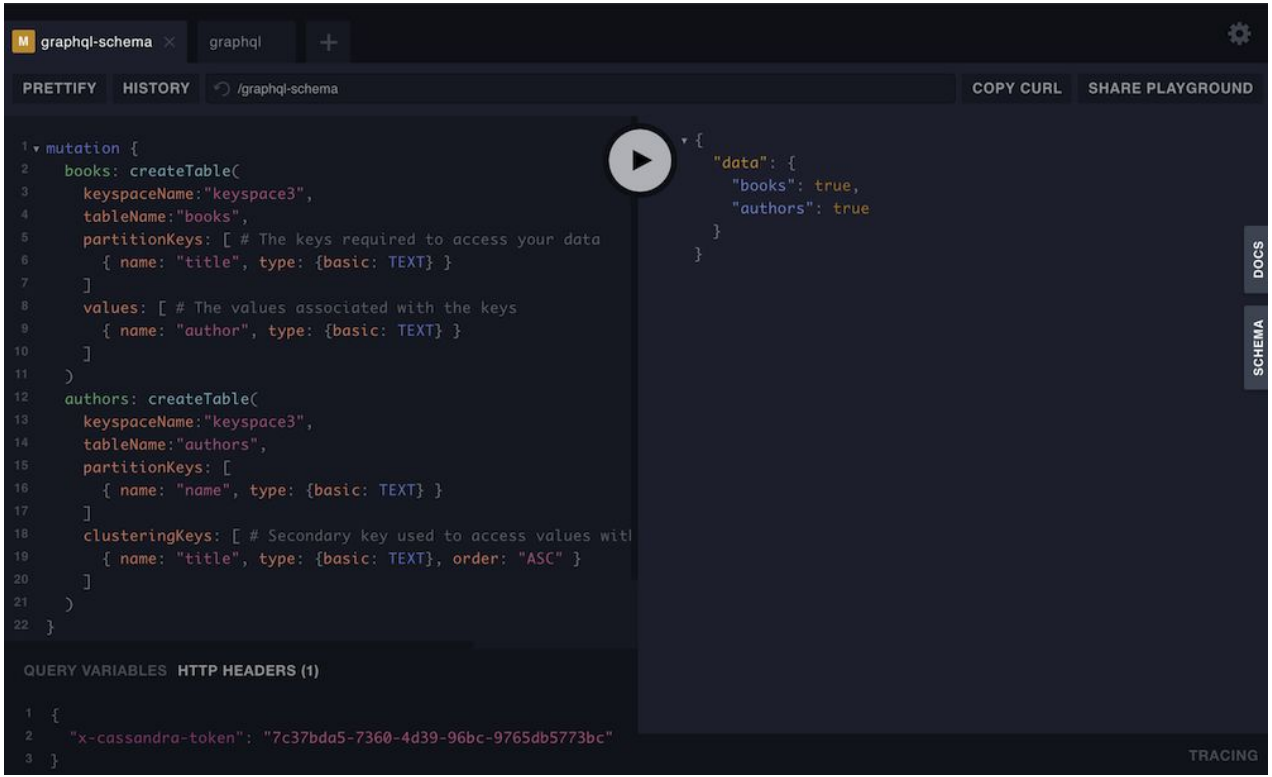
Stargate GraphQL API

- Expose existing CQL schema as a GraphQL endpoint
- Allows more fine-grained control of what fields returned
- In progress: federation across multiple services to join data
- Easy experimentation in GraphQL playground

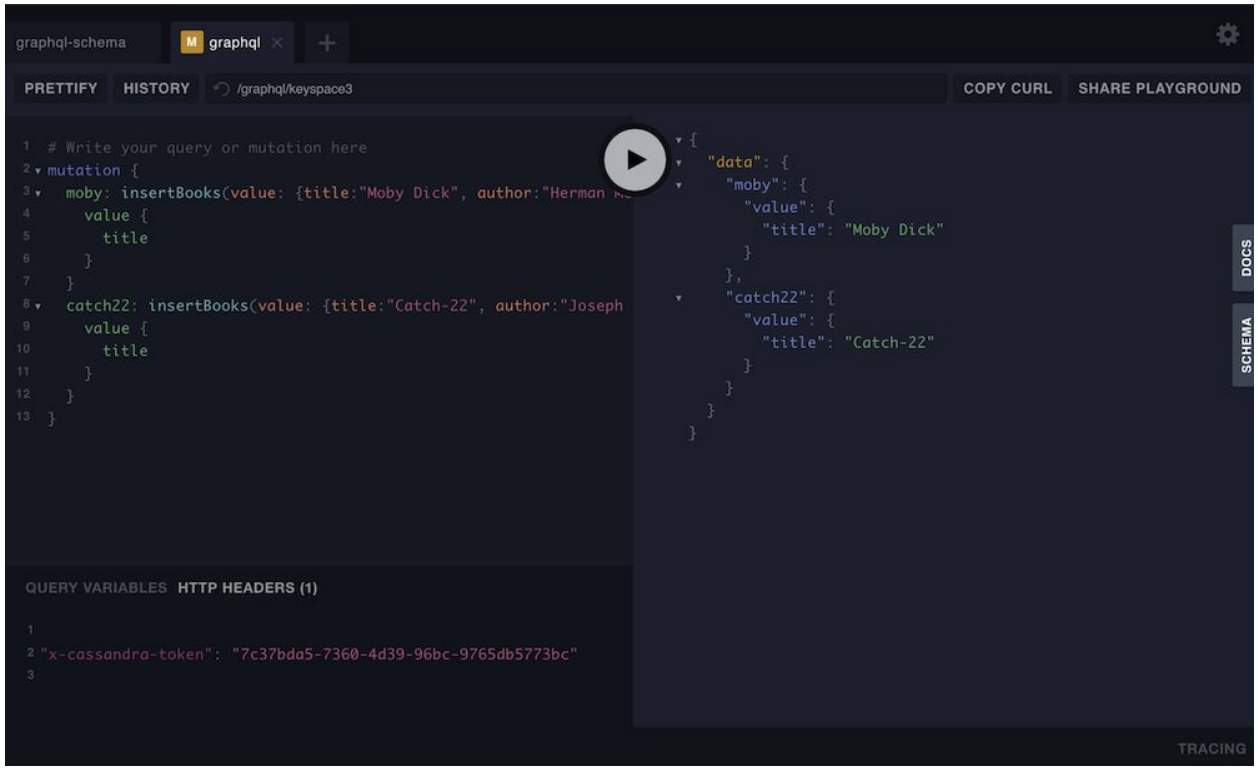


Quick start: https://stargate.io/docs/stargate/1.0/quickstart/quick_start-graphql.html

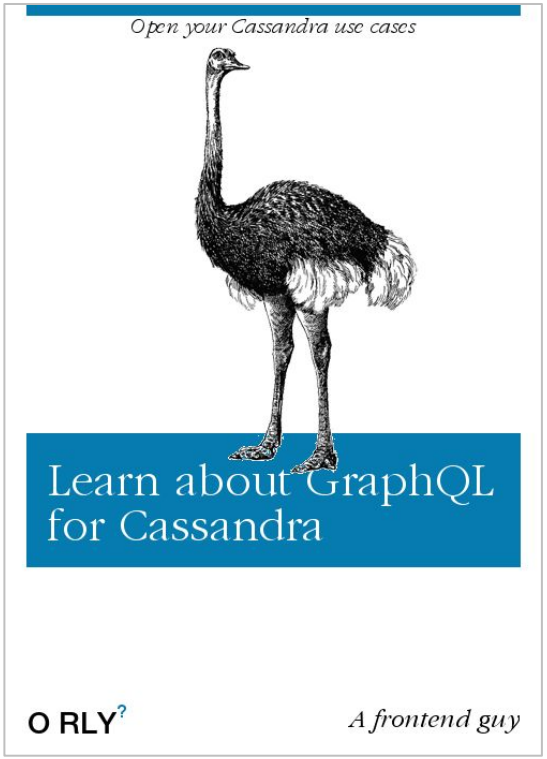
GraphQL “CQL First”

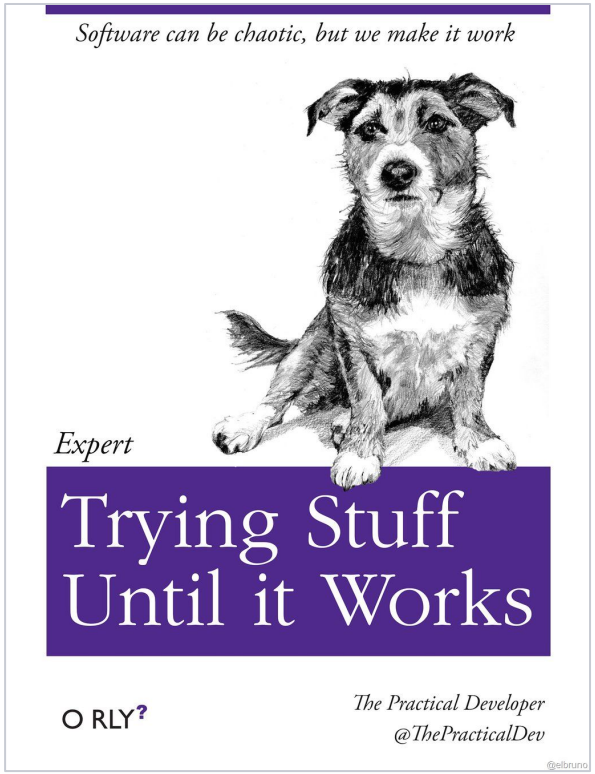


GraphQL “CQL First”

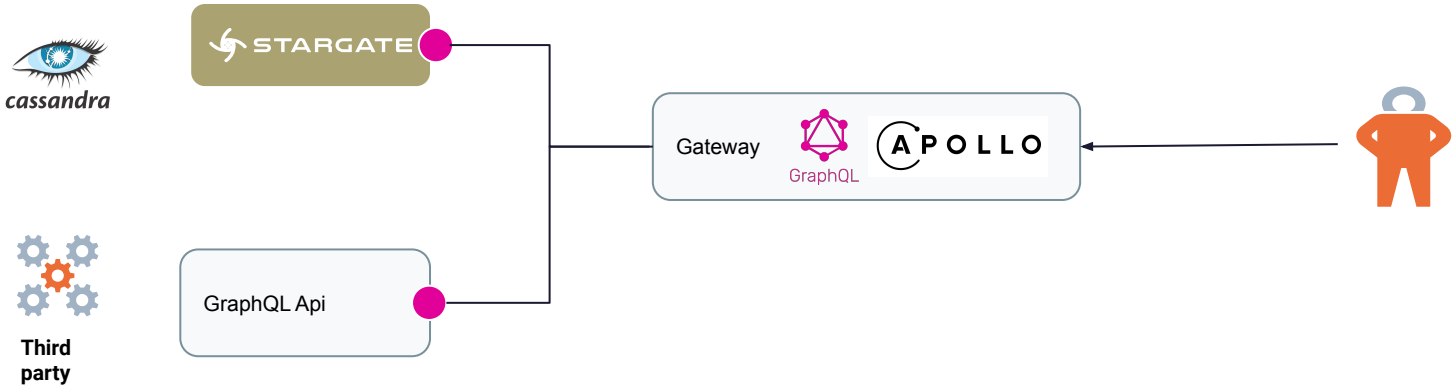


Demo


Open your Cassandra use cases
Learn about GraphQL for Cassandra
O RLY?
A frontend guy


Software can be chaotic, but we make it work
Expert
Trying Stuff Until it Works
O RLY?
The Practical Developer
@ThePracticalDev

GraphQL Federation



GraphQL “Schema First”

```
type Book @key @cql_entity(name: "book") @cql_input {
  title: String! @cql_column(partitionKey: true, name: "book_title")
  isbn: String @cql_column(clusteringOrder: ASC)
  author: [String] @cql_index(name: "author_idx", target: VALUES)
}

type Address @cql_entity(target: UDT) @cql_input {
  street: String
  city: String
  state: String
  zipCode: String @cql_column(name: "zip_code")
}

#...

type Query {
   bookByTitleAndIsbn(title:String!, isbn:String): [Book]
   readerByNameAndUserid(name:String!, user_id:Uuid): [Reader]
}

type Mutation {
  insertBook(book:BookInput!): Book
  updateBook(book:BookInput!): Boolean @cql_update
  deleteBook(book:BookInput!): Boolean
  insertReader(reader:ReaderInput!): Reader
  deleteReader(reader:ReaderInput!): Boolean
  insertLibCollection(libColl: LibCollectionInput!): LibCollection
  deleteLibCollection(libColl: LibCollectionInput!): Boolean
}
```

GraphQL “Schema First”

```
mutation {
  deploySchema(
    keyspace: "library"
    expectedVersion: "1da4f190-b7fd-11eb-8258-1ff1380eaff5"
    schema: """
    # Stargate does not require definition of fields in @key,
    # it uses the primary key
    type Book @key @cql_entity(name: "book") @cql_input {
      title: String! @cql_column(partitionKey: true, name: "book_title")
      isbn: String @cql_column(clusteringOrder: ASC)
      author: [String] @cql_index(name: "author_idx", target: VALUES)
    }

    ...
  }
```

GraphQL “Schema First”

```
query fetchBook {
  book(title: "Native Son") {
    title
    author
  }
}

type Query {
bookGT(
  title: String
  isbn: String @cql_where(field: "isbn", predicate: GT)
): [Book]
}

type Query {
  readerCONTAINS(
    reviews: ReviewInput! @cql_where(field: "reviews", predicate: CONTAINS)
  ): [Reader]
}

type Query {
  booksIn(
    title: [String!] @cql_where(field: "title", predicate: IN)
  ): [Book]
}
```

Agenda (40 min)

01 STARGATE

Stargate Data gateway
What, Why and How

02 {REST}

Exploring Apis
Rest DML and DDL

03

Exploring Apis
Document oriented

04

Exploring Apis
GraphQL and Federation

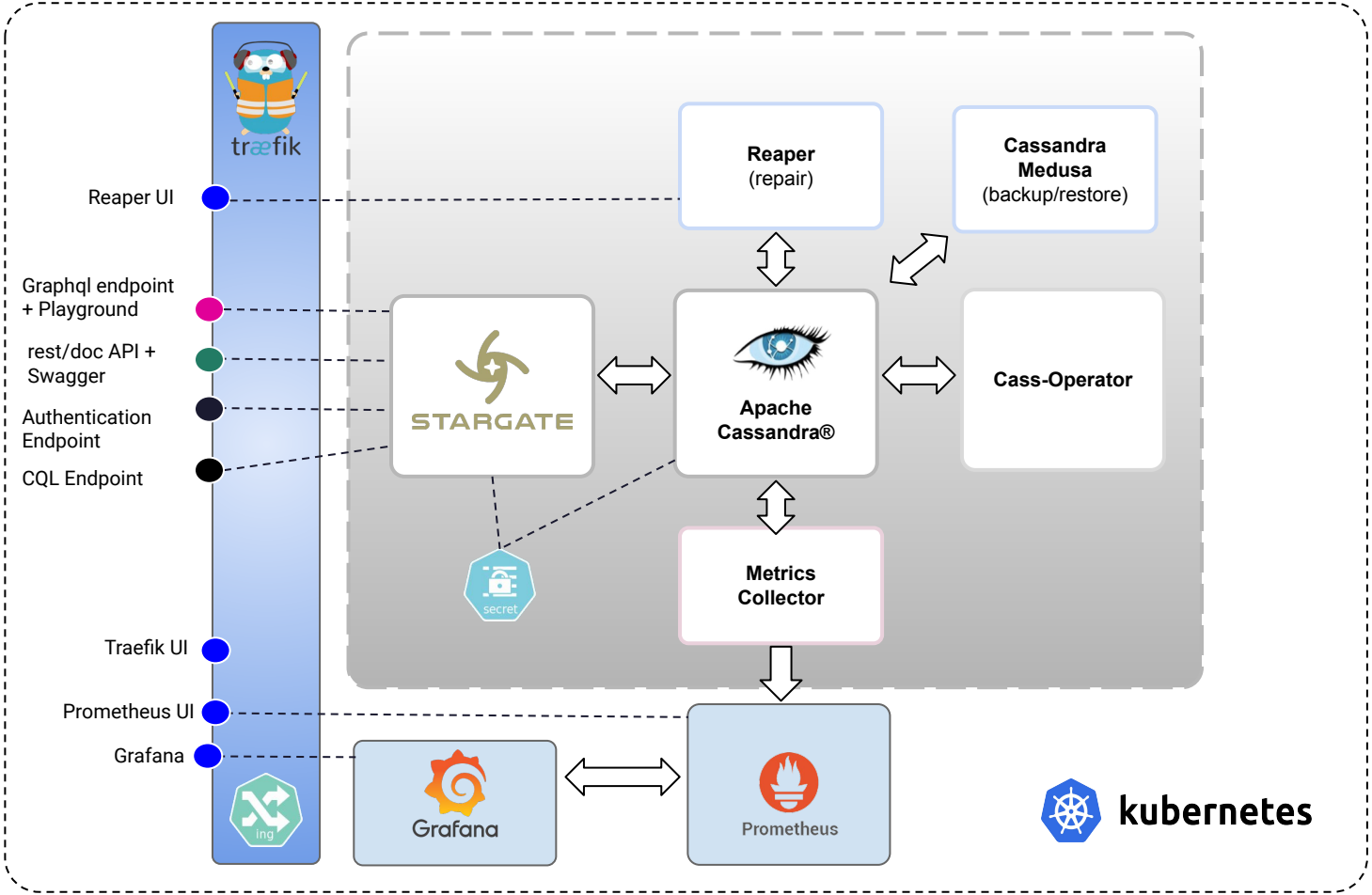
05

Tooling
SDK, K8ssandra

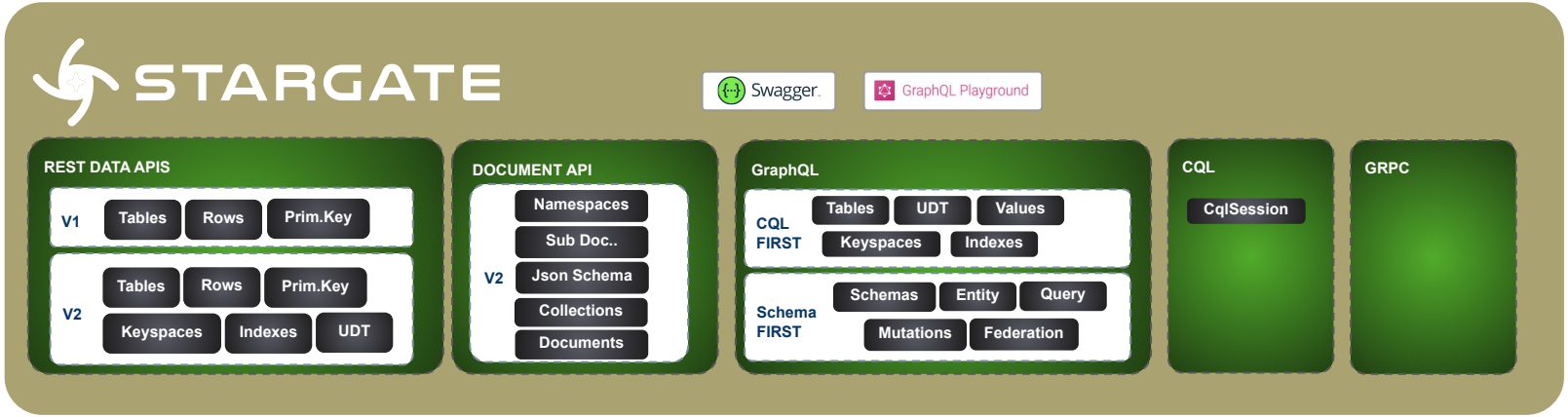
05

What's NEXT ?
gRPC, CDC, sql...

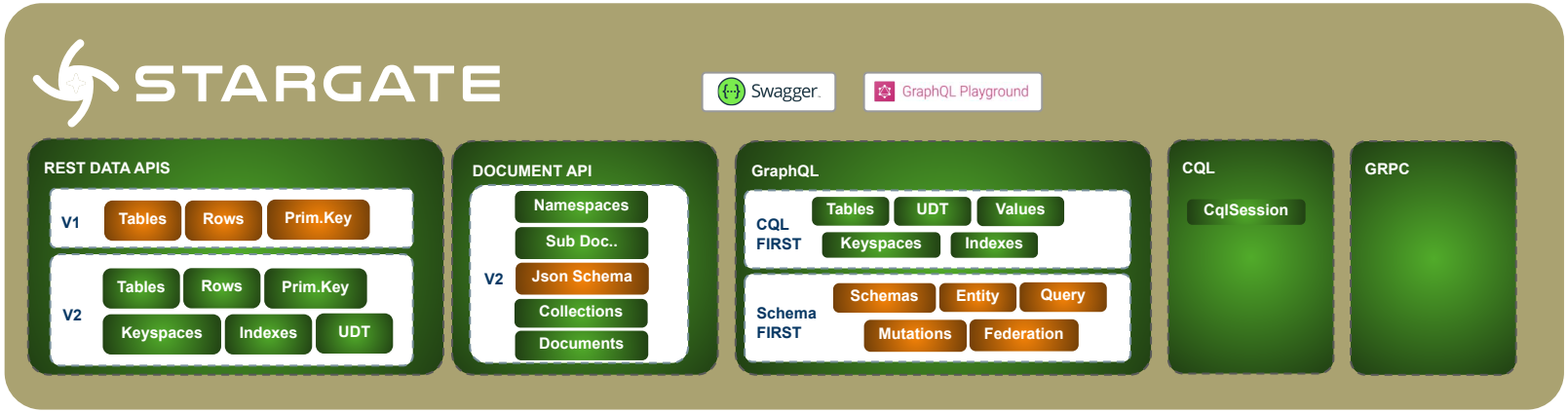
K8SSANDRA


traefik
Reaper UI
Graphql endpoint + Playground
rest/doc API + Swagger
Authentication Endpoint
CQL Endpoint
Traefik UI
Prometheus UI
Grafana
ing
Reaper (repair)
Cassandra Medusa (backup/restore)
STARGATE
Apache Cassandra®
Cass-Operator
secret
Metrics Collector
Grafana
Prometheus
kubernetes

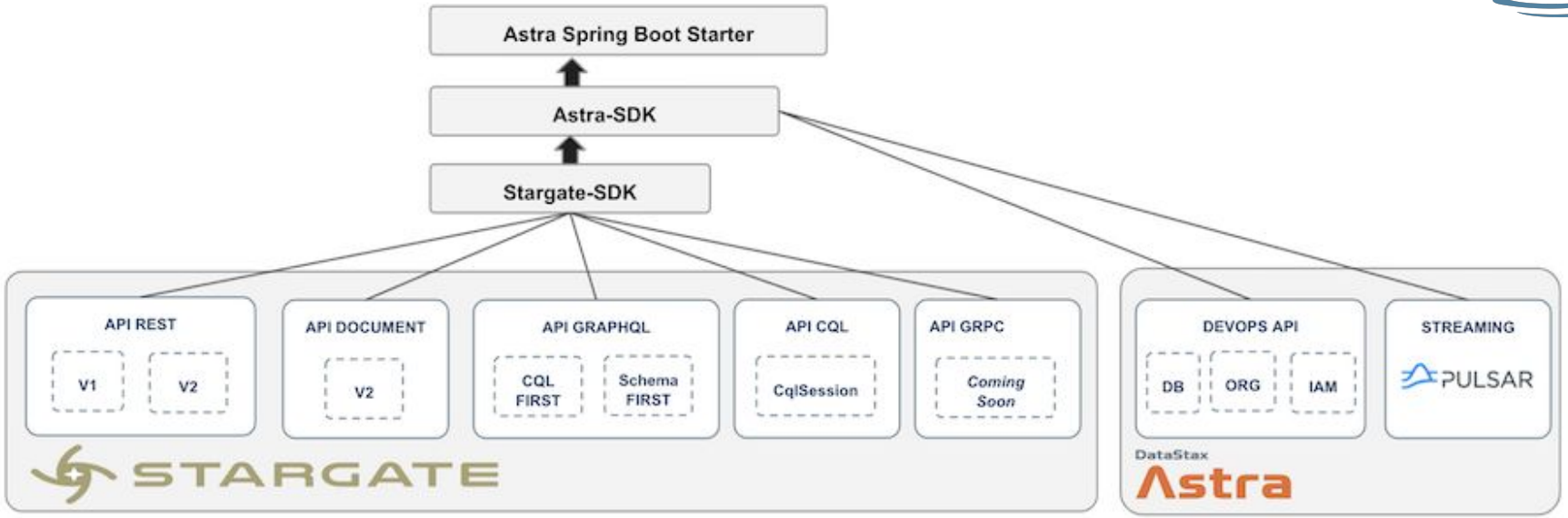
Features Map



SDK in Java, Javascript, Python



Focus on Java SDK



https://github.com/datastax/astra-sdk-java/wiki

Sample Codes

```
StargateClient client = StargateClient.builder()
  .username("k8ssandra-superuser")           // Mandatory username
  .password("JxzrPOnvDGqfEOQ0EySQ")          // Mandatory password
  .endPointAuth("http://localhost:8081")     // Mandatory authencation url, defaulting to http://localhost
  .endPointRest("http://localhost:8082")     // Rest and Document Apis
  .endPointGraphQL("http://localhost:8080")  // GraphQL Api

  // Cqlsession Only
  .addCqlContactPoint("127.0.0.", 9042)      // Contact Point
  .localDc("dc1")                            // Local Datacenter is mandatory driver 4x+
  .keypace("ks1")                            // (optional) Set your keyspace
  .build();
```

Sample Code

```java
// Retrieve an object and marshall
Optional<Address> address = colPersonClient
   .document("e8c5021b-2c91-4015-aec6-14a16e449818")
   .findSubDocument("address", Address.class);
```

```java
// Building query {"age": {"$gte":30},"lastname": {"$eq":"PersonAstra2"}}
SearchDocumentQuery query = SearchDocumentQuery.builder()
                    .where("age").isGreaterOrEqualsThan(30)      // First filter to use where()
                    .and("lastname").isEqualsTo("PersonAstra2")  // Any extra filter to use and()
                    .withPageSize(10)                            // Default and max pageSize are 20
                    .build();

// Retrieve PAGE 1
DocumentResultPage<Person> currentPage = colPersonClient.findPage(query,  Person.class);

// Retrieve PAGE 2 (if any)
if (currentPage.getPageState().isPresent()) {
  query.setPageState(currentPage.getPageState().get());
}
DocumentResultPage<Person> nextPage = colPersonClient.findPage(query,  Person.class);

// Retrieve all documents in one call (with warning in RED ABOVE)
Stream<Person> allPersons = colPersonClient.findAll(query, Person.class)
```

Agenda (40 min)

01 STARGATE

Stargate Data gateway
What, Why and How

02 {REST}

Exploring Apis
Rest DML and DDL

03

Exploring Apis
Document oriented

04

Exploring Apis
GraphQL and Federation

05

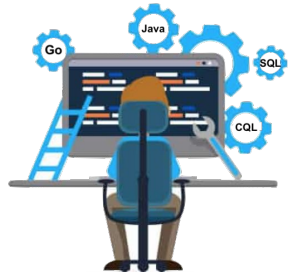
Tooling
SDK, K8ssandra

05

What's NEXT ?
gRPC, CDC, sql...

What's NEXT ?

- **Apis**
  - GRPC (GA next month)
  - Relational API
  - Dynamo API

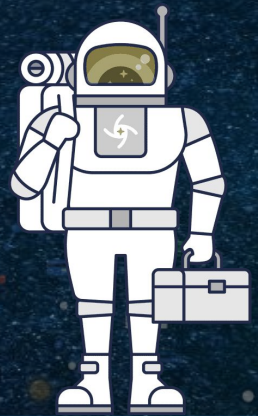


- **Architecture**
  - SSO, KMS
  - Architecture redesigned
  - ZDD
  - CDC

THANK YOU

GitHub @clun

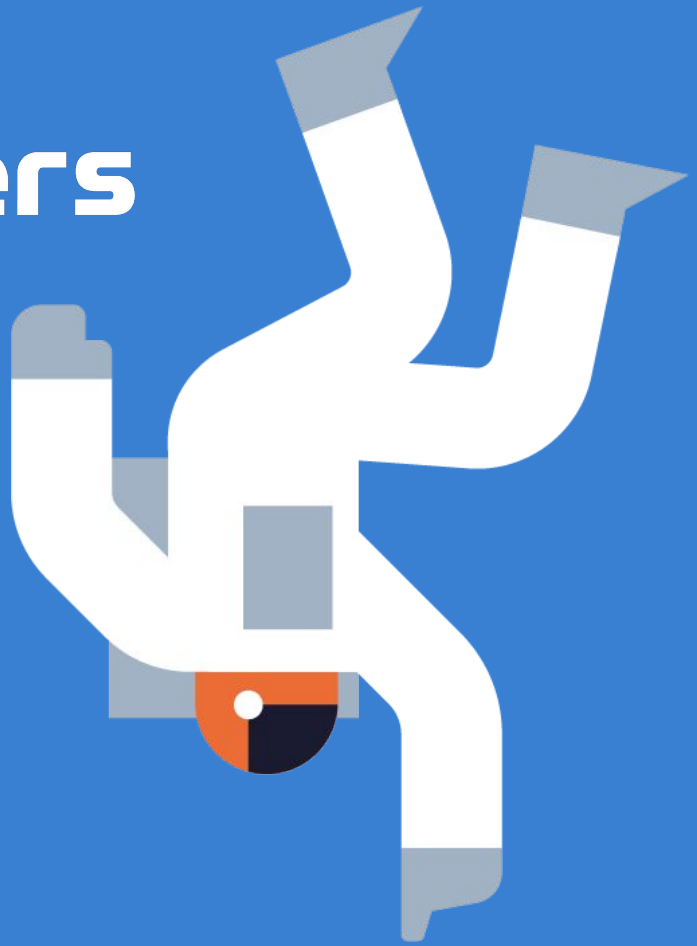
@clunven



DataStax Developers

DataStax Developers

Thank you!



GitHub @clun

@clunven