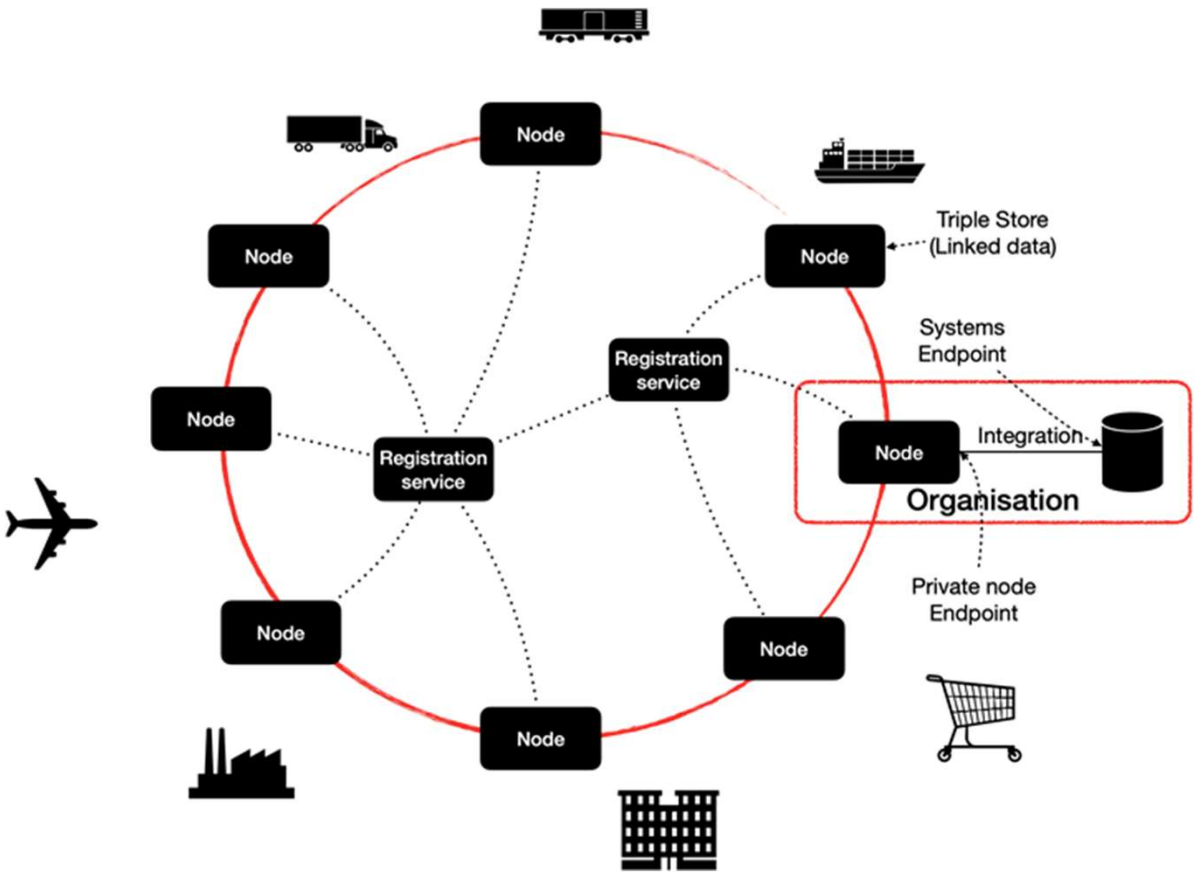
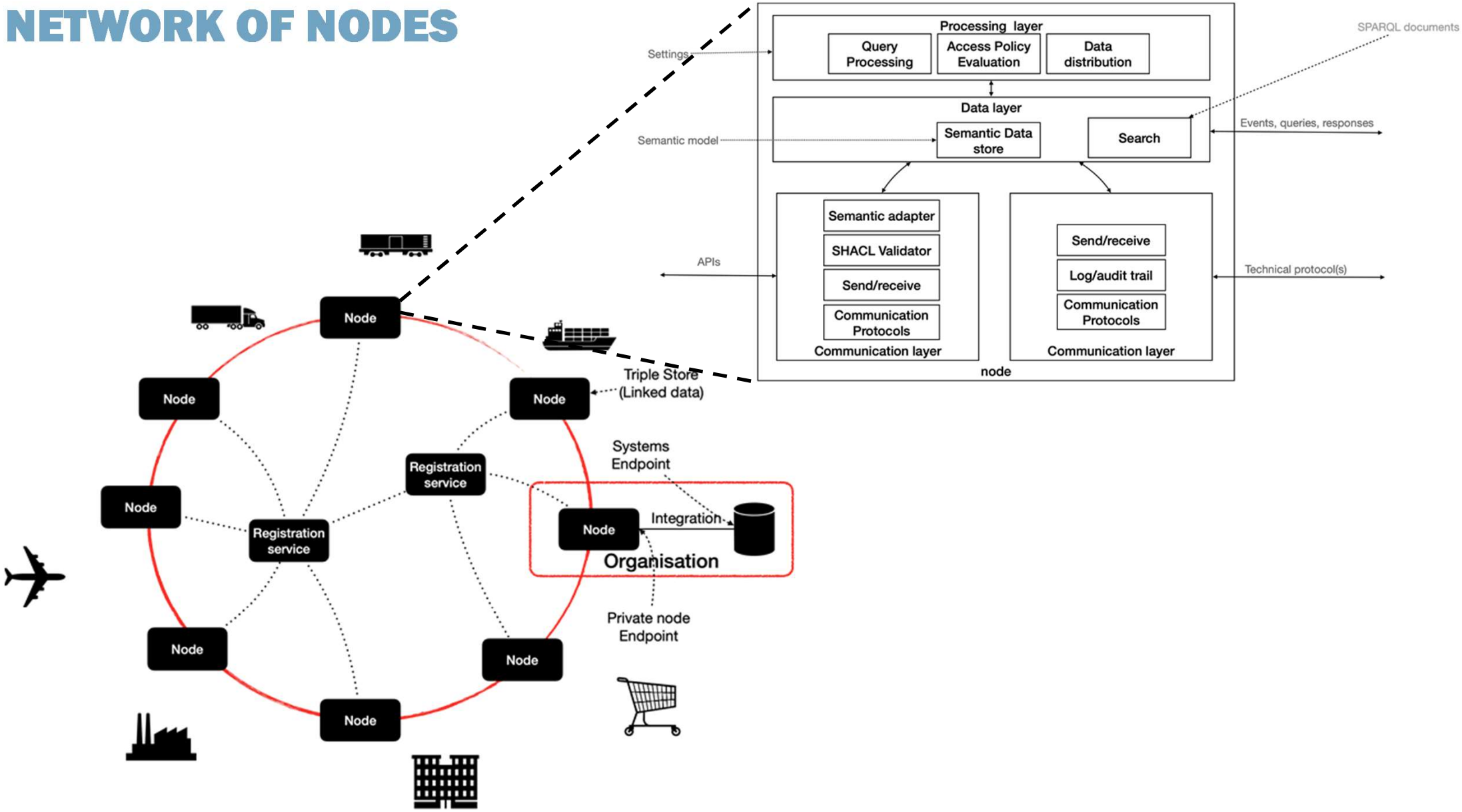

TNO innovation for life
BDI PROTOTYPE DEMO (NOV 2022)

BASIS DATA INFRASTRUCTURE



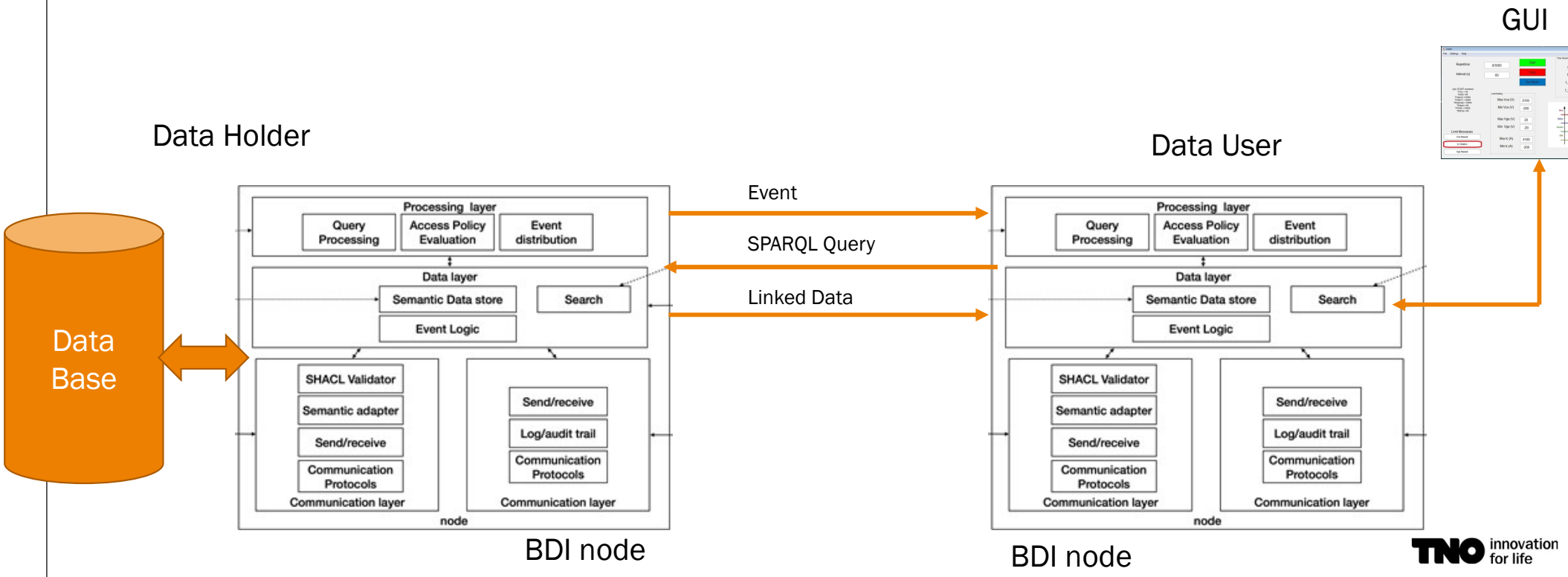
- Same language:
  - Data model
  - Digital Twins
  - Event based (e.g. departure, arrival, ...)
- Data at the source

NETWORK OF NODES



1STE PROTOTYPE (DOUANE-TRADELENS CASE)
FIRST STEP

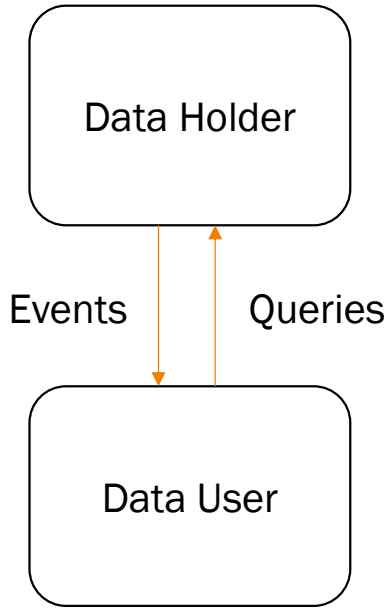
- PoC: event based communcation with data at the source using het same data model
- two node – one event implementation



BDI PROTOTYPE SCOPE
V0.1

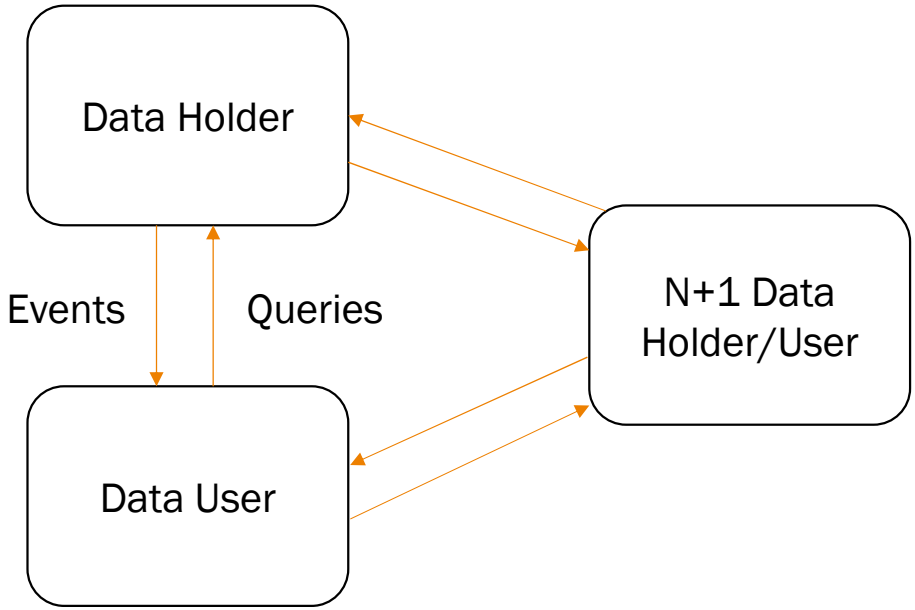
- Implement a proof-of-concept for a Basic Data-sharing Infrastructure (BDI)
  - **Event** based **semantic** data-sharing between **two** BDI nodes
  - **Querying** data at **the source** using **SPARQL** queries
- By creating a working prototype
  - Demonstrate that it is **technically possible** to implement
  - Reference implementation
- Using Tradelens and Douane as an example use-case
  - We **subscribe** to Tradelens events like: *"Actual Loaded On Vessel"* using a **WebHook**
  - Data is queried **by Douane** using a **SPARQL** query

DATA HOLDER & DATA USER
V0.1: TWO BDI NODES

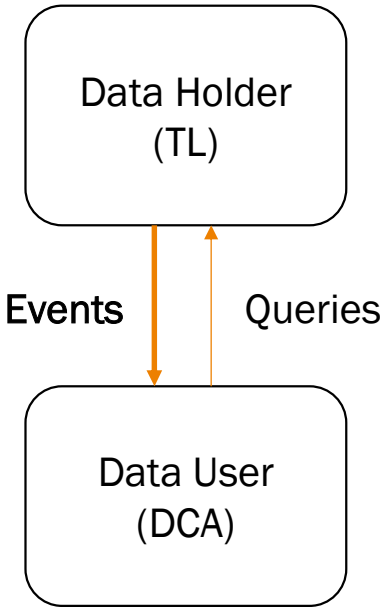


DATA HOLDER & DATA USER

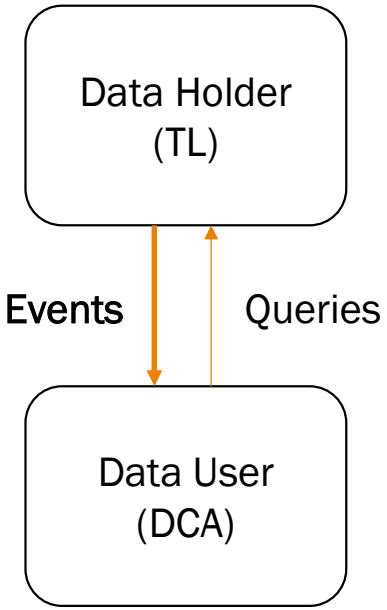
N+1 BDI NODES



DATA HOLDER & DATA USER
TRADELENS & DCA



DATA HOLDER & DATA USER
TRADELENS & DCA

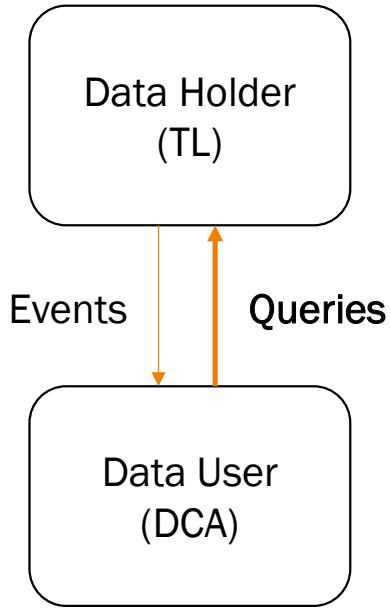


Events (RDF in turtle)

@prefix data: <https://ontology.tno.nl/logistics/federated/tradelens#> .
....
data:DigitalTwin-c2cb84ce-87e9-4465-bb03-79e34c4d54fe a owl:NamedIndividual, dt:Container .
data:DigitalTwin-d2cbf432-6dff-4f55-ad0a-63fccdaf5c69 a owl:NamedIndividual, dt:TransportMeans .
data:PhysicalInfrastructure-AEJEA a owl:NamedIndividual, pi:Location .

data:event-746dd77c-2b71-4b05-b928-7fcd7663e8df a owl:NamedIndividual, event:Event,
event:LoadEvent;
....

DATA HOLDER & DATA USER
TRADELENS & DCA



Events (RDF in turtle)

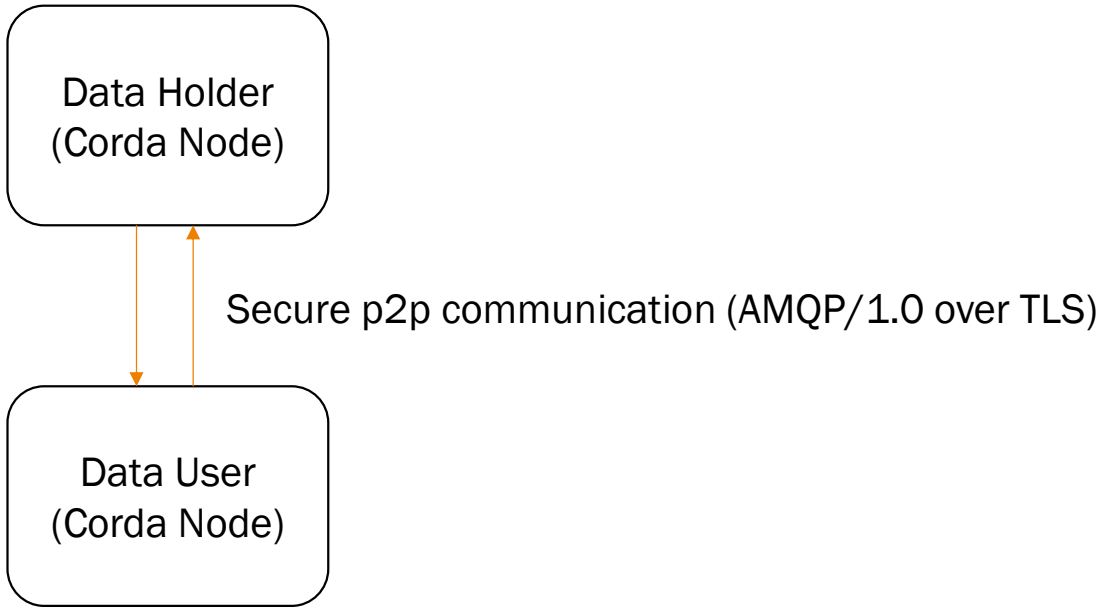
@prefix data: <https://ontology.tno.nl/logistics/federated/tradelens#> .
....
data:DigitalTwin-**c2cb84ce-87e9-4465-bb03-79e34c4d54fe** a owl:NamedIndividual, dt:Container .
data:DigitalTwin-d2cbf432-6dff-4f55-ad0a-63fccdaf5c69 a owl:NamedIndividual, dt:TransportMeans .
data:PhysicalInfrastructure-AEJEA a owl:NamedIndividual, pi:Location .

data:event-746dd77c-2b71-4b05-b928-7fcd7663e8df a owl:NamedIndividual, event:Event,
event:LoadEvent;
....

Queries (SPARQL)

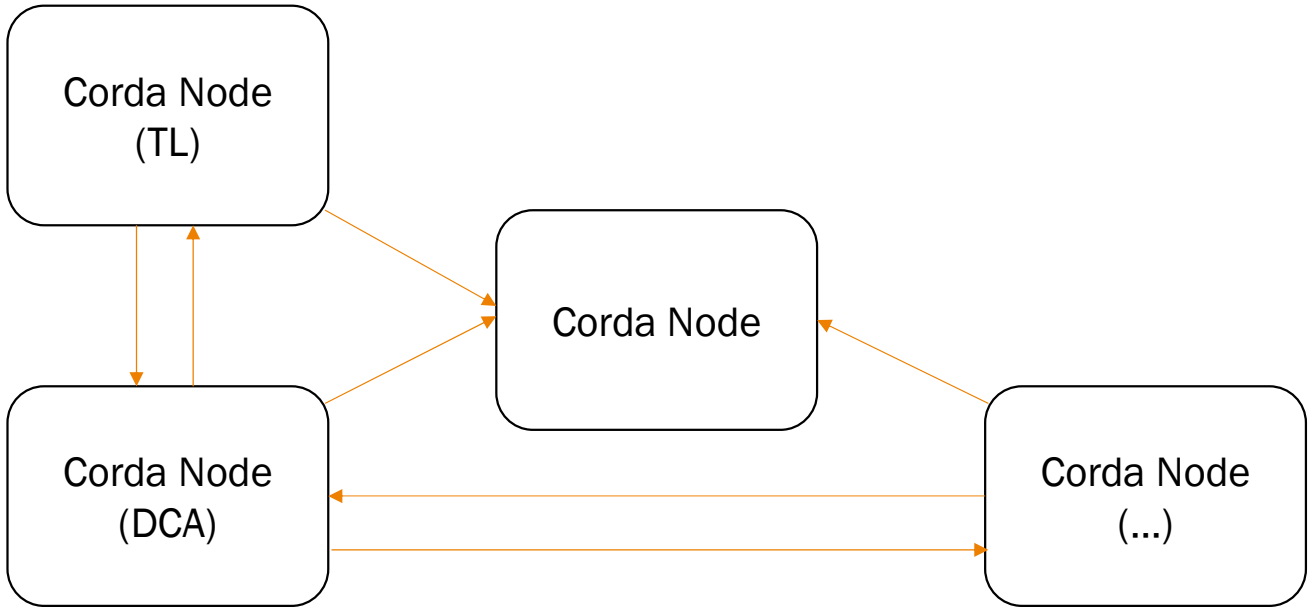
PREFIX data: <https://ontology.tno.nl/logistics/federated/tradelens#>
SELECT * WHERE {
data:DigitalTwin-**c2cb84ce-87e9-4465-bb03-79e34c4d54fe** ?p ?o .
}

COMMUNICATION BETWEEN BDI NODES
CORDA NETWORK



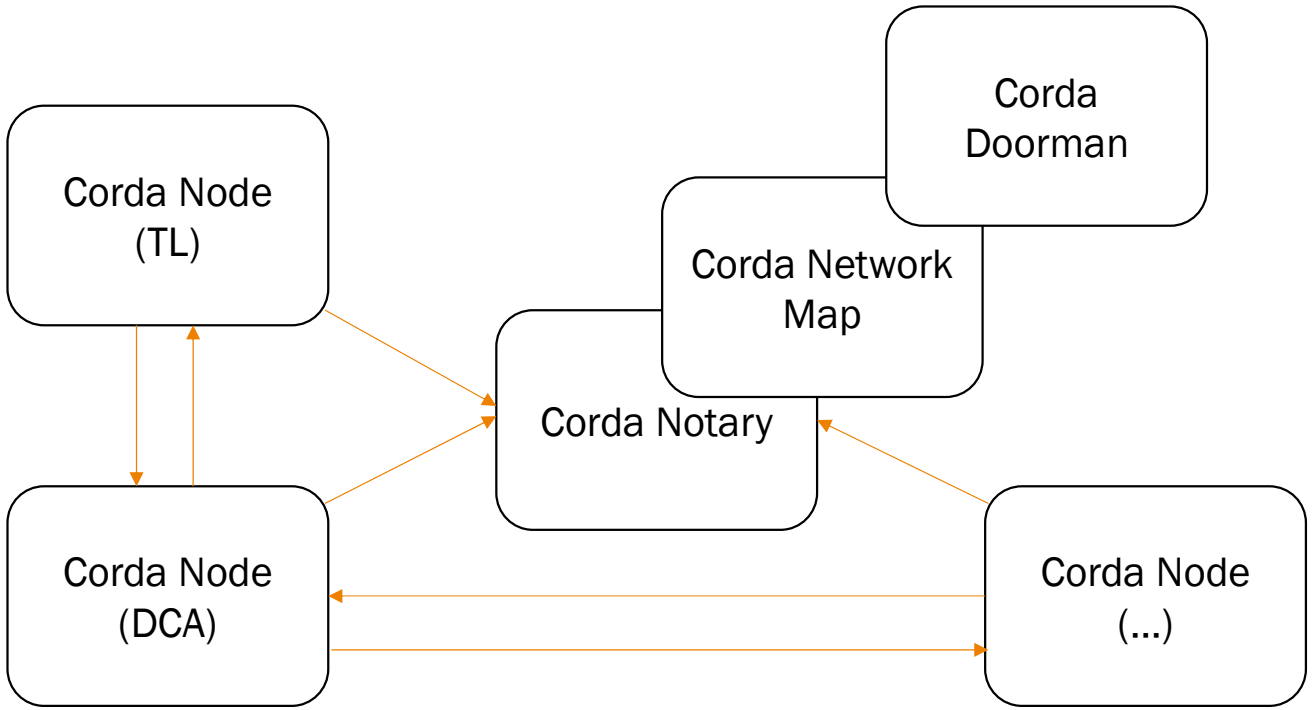
CORDA NETWORK (EXTENSIBLE)

N+1 BDI NODES = N+1 CORDA NODES



CORDA NETWORK

LEDGER, DISCOVERABILITY, PERMISSIONING



SEMANTIC ADAPTER
FROM NON-SEMANTIC TO SEMANTIC DATA

Events (RDF in turtle)

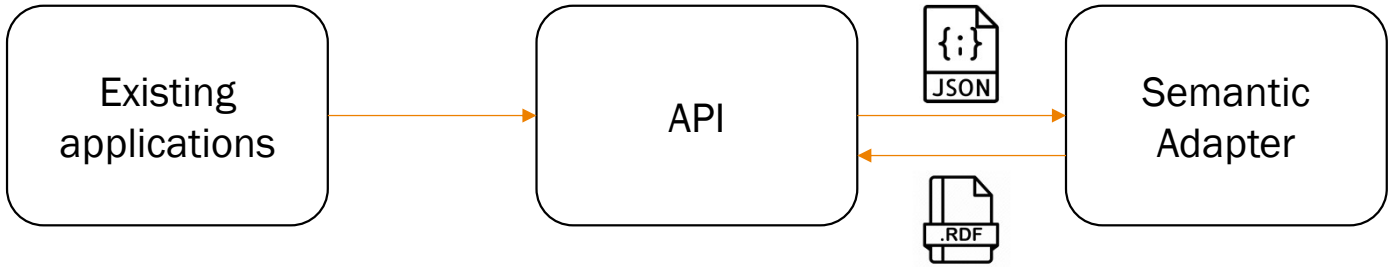
```
@prefix data: <https://ontology.tno.nl/logistics/federated/tradelens#> .
....
data:DigitalTwin-c2cb84ce-87e9-4465-bb03-79e34c4d54fe a owl:NamedIndividual, dt:Container .
data:DigitalTwin-d2cbf432-6dff-4f55-ad0a-63fccdaf5c69 a owl:NamedIndividual, dt:TransportMeans .
data:PhysicalInfrastructure-AEJEA a owl:NamedIndividual, pi:Location .

data:event-746dd77c-2b71-4b05-b928-7fcd7663e8df a owl:NamedIndividual, event:Event,
  event:LoadEvent;
  ....
```

Events from Tradelens are in JSON format

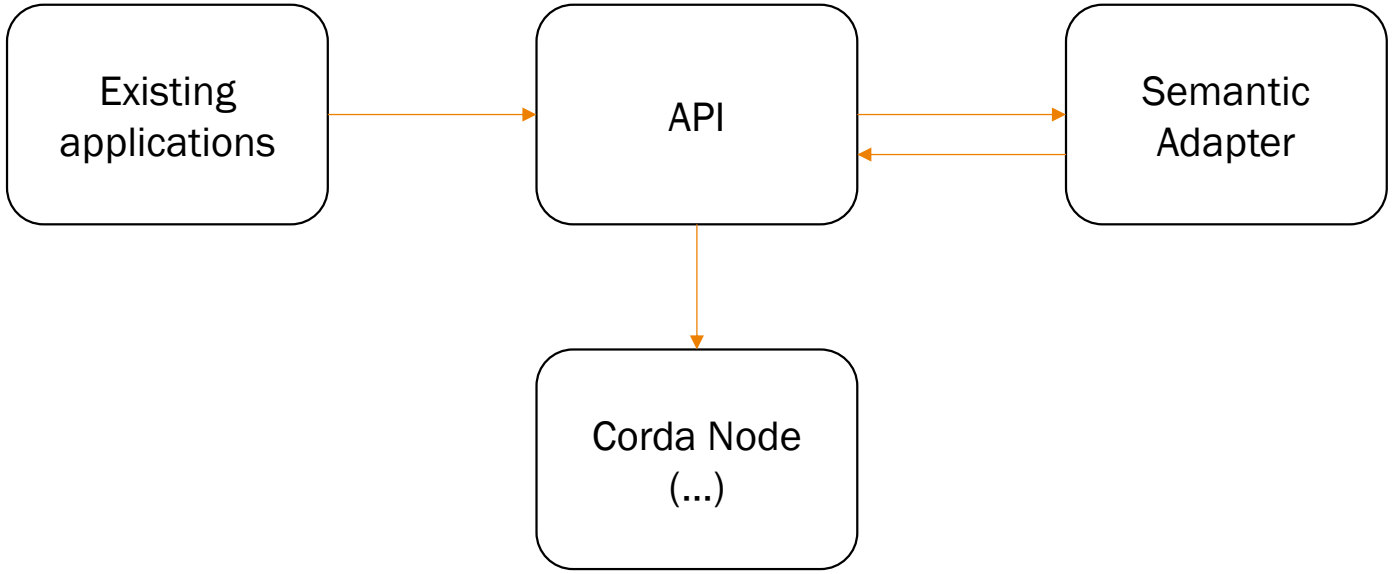
```
{
      "eventType": "actualLoadedOnVessel",
      "eventName": "Actual loaded on vessel",
      "consignmentId": "9c0a54d9-b152-4019-8994-283353ffdaaa",
      "transportEquipmentId": "c2cb84ce-87e9-4465-bb03-79e34c4d54fe",
      "billOfLadingNumber": "BL429905",
      "eventSubmissionTime8601": "2022-09-23T12:35:01.307Z",
      .....
}
```

SEMANTIC ADAPTER

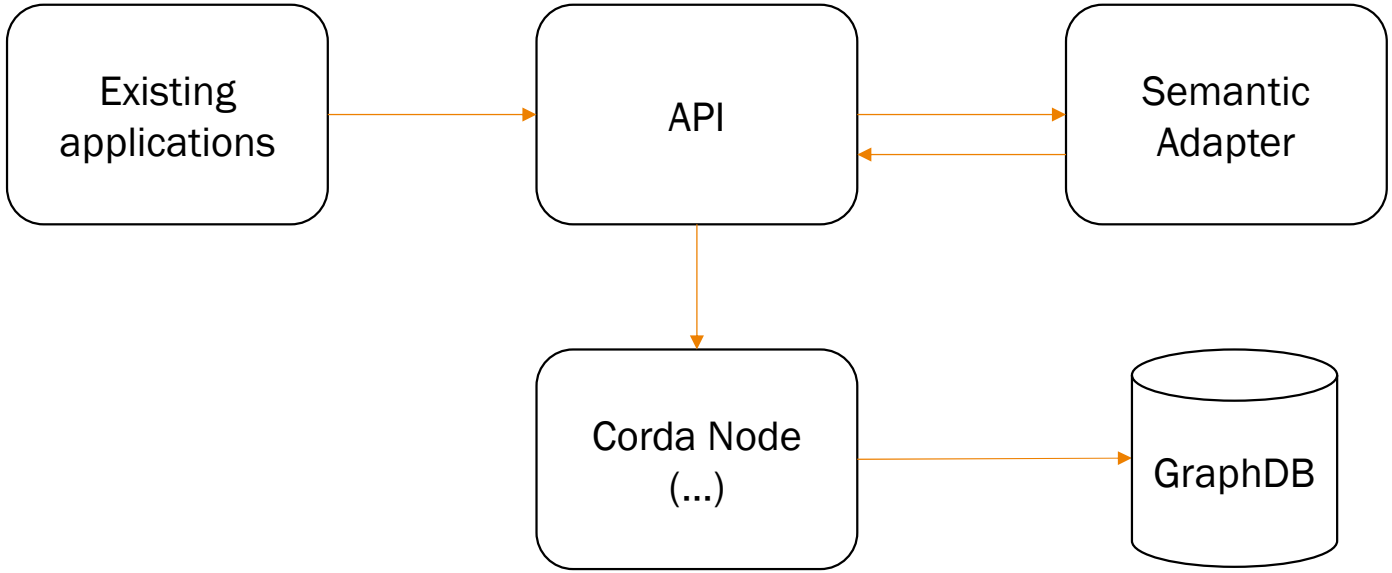
MAPPER BETWEEN NON-SEMANTIC AND SEMANTIC WORLD



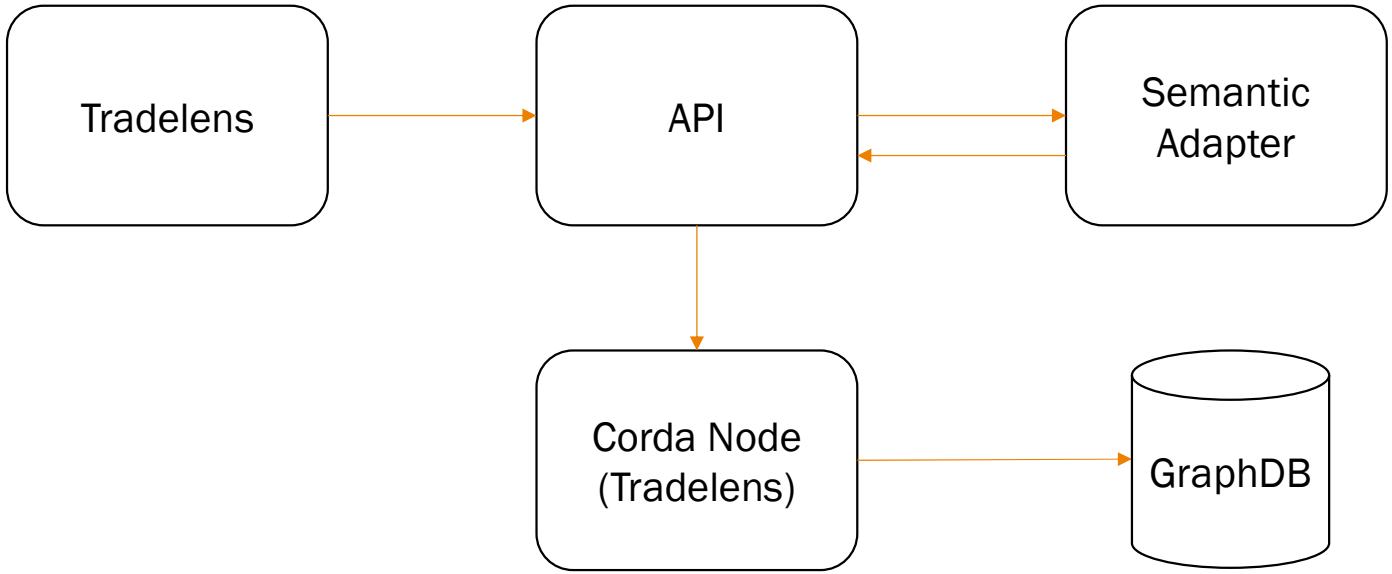
API
BRIDGES THE NON-BDI AND BDI WORLD



STORING EVENTS
GRAPHDB

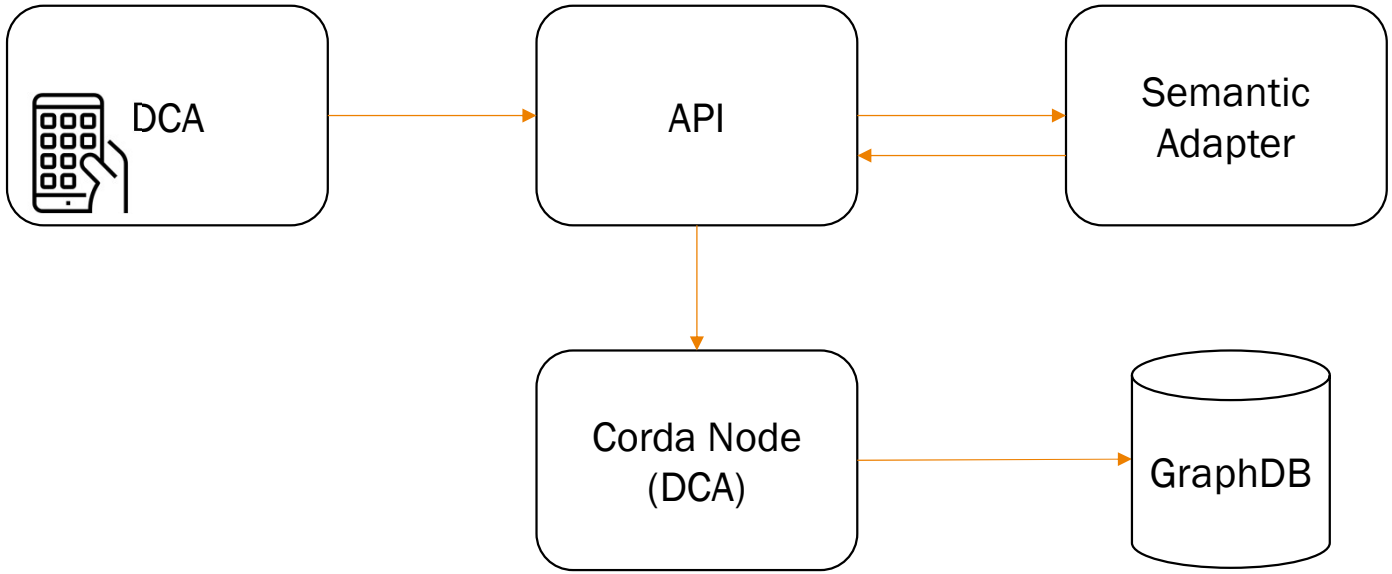


TRADELENS
BDI

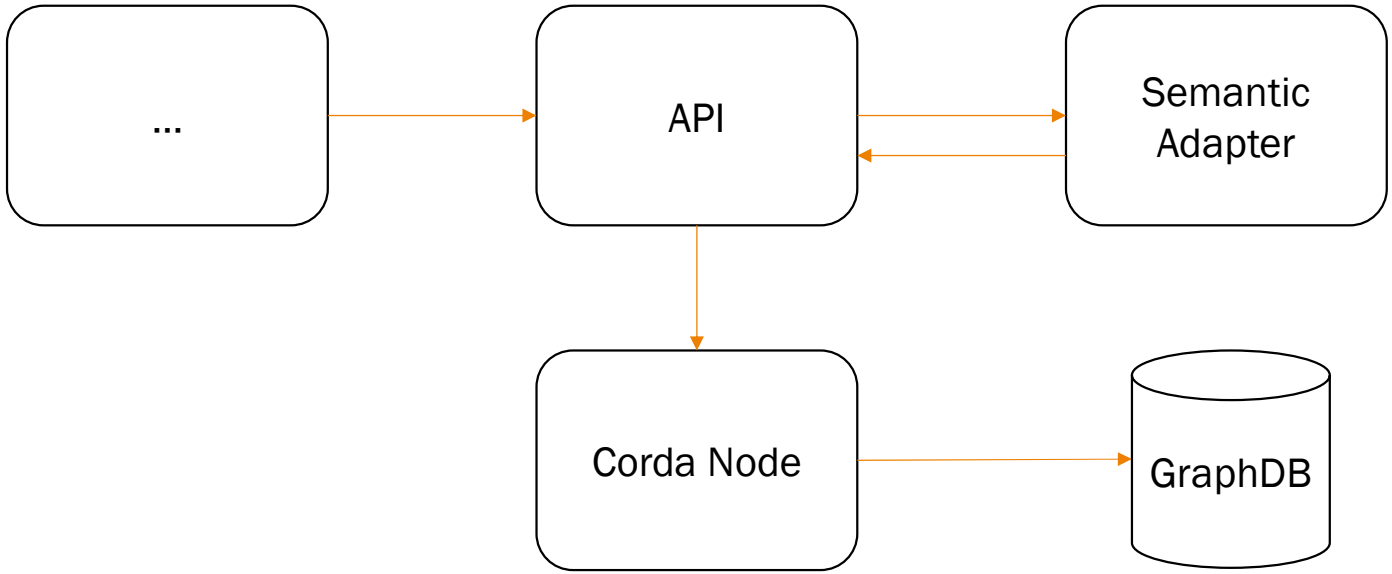


DUTCH CUSTOMS AUTHORITY

BDI



ANY
BDI

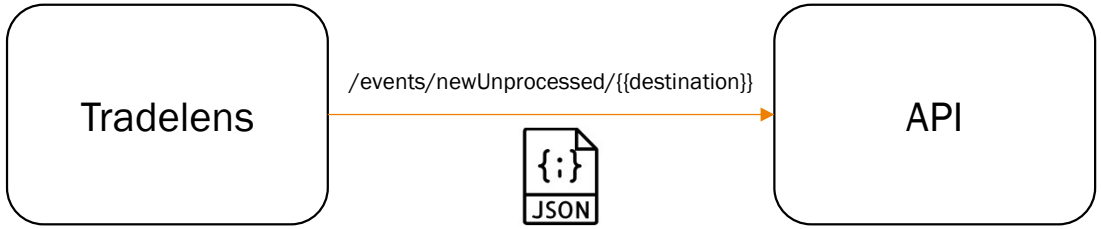


PROTOTYPE V0.1
COMPONENTS

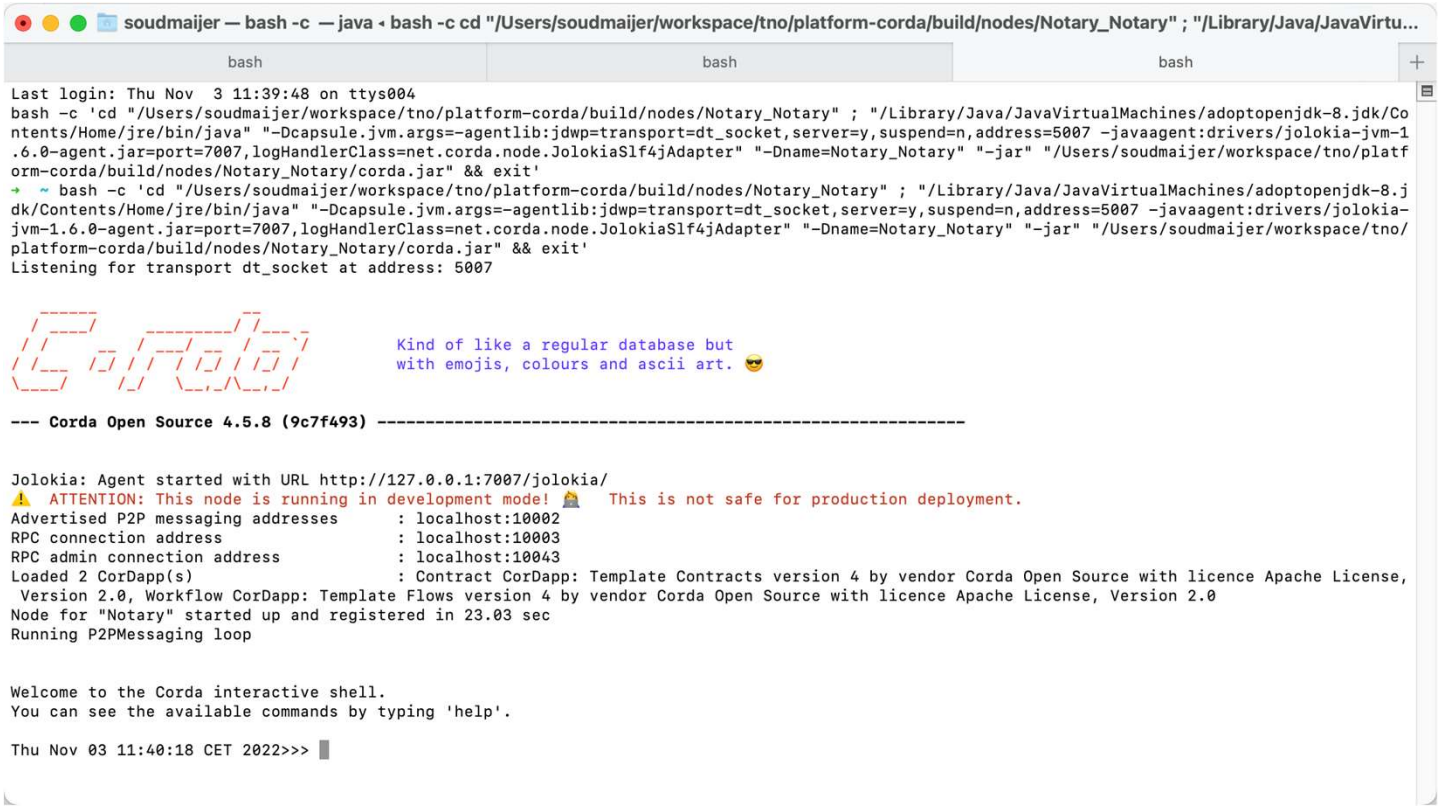
- Corda 4.x network
- Corda workflows written in Kotlin
- REST API using Spring Boot and Kotlin
- Semantic Adapter written in Python
- Running on JVM technology
- GraphDB 10
- Runs ons Docker and Kubernetes

USE-CASE: TRADELENS EVENT

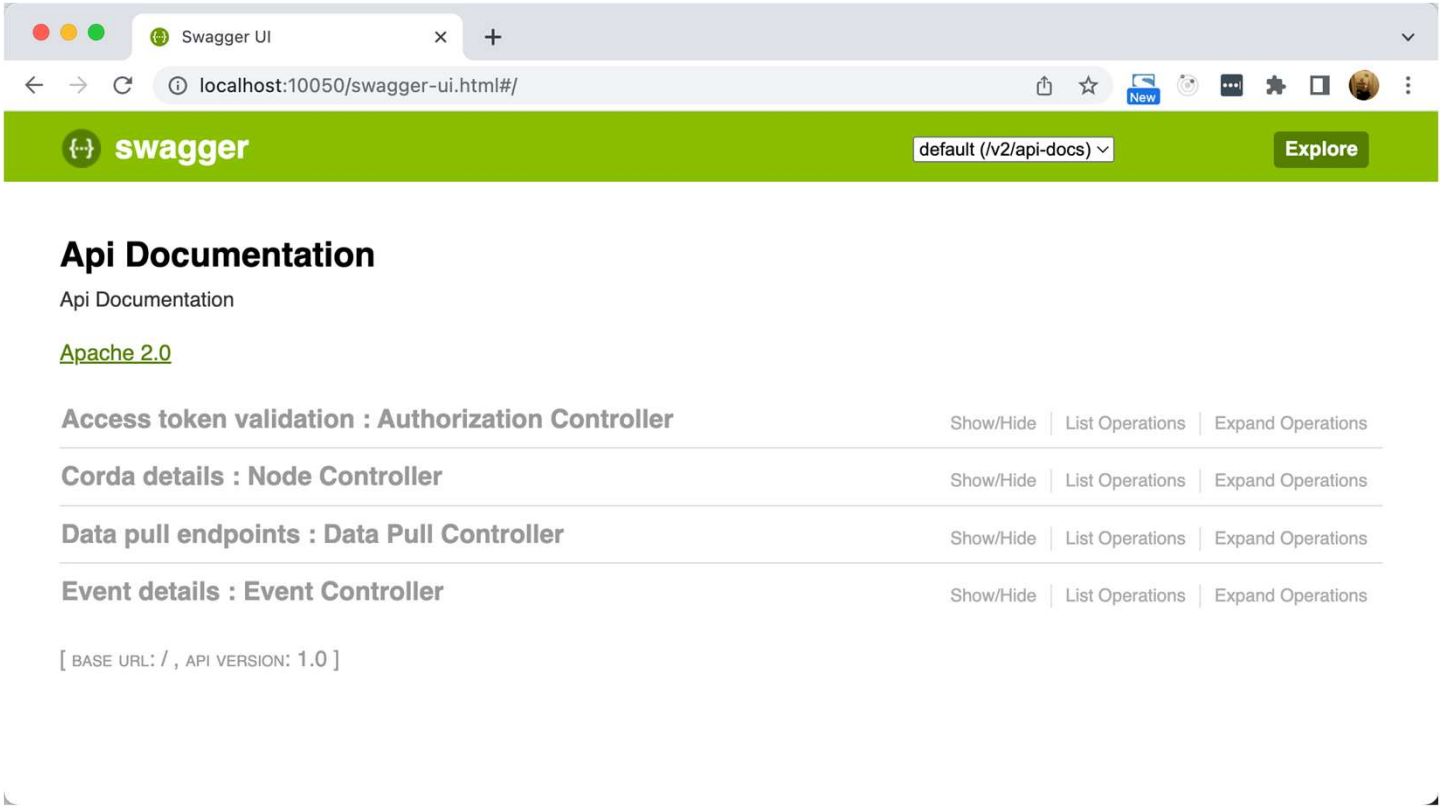
DEMO

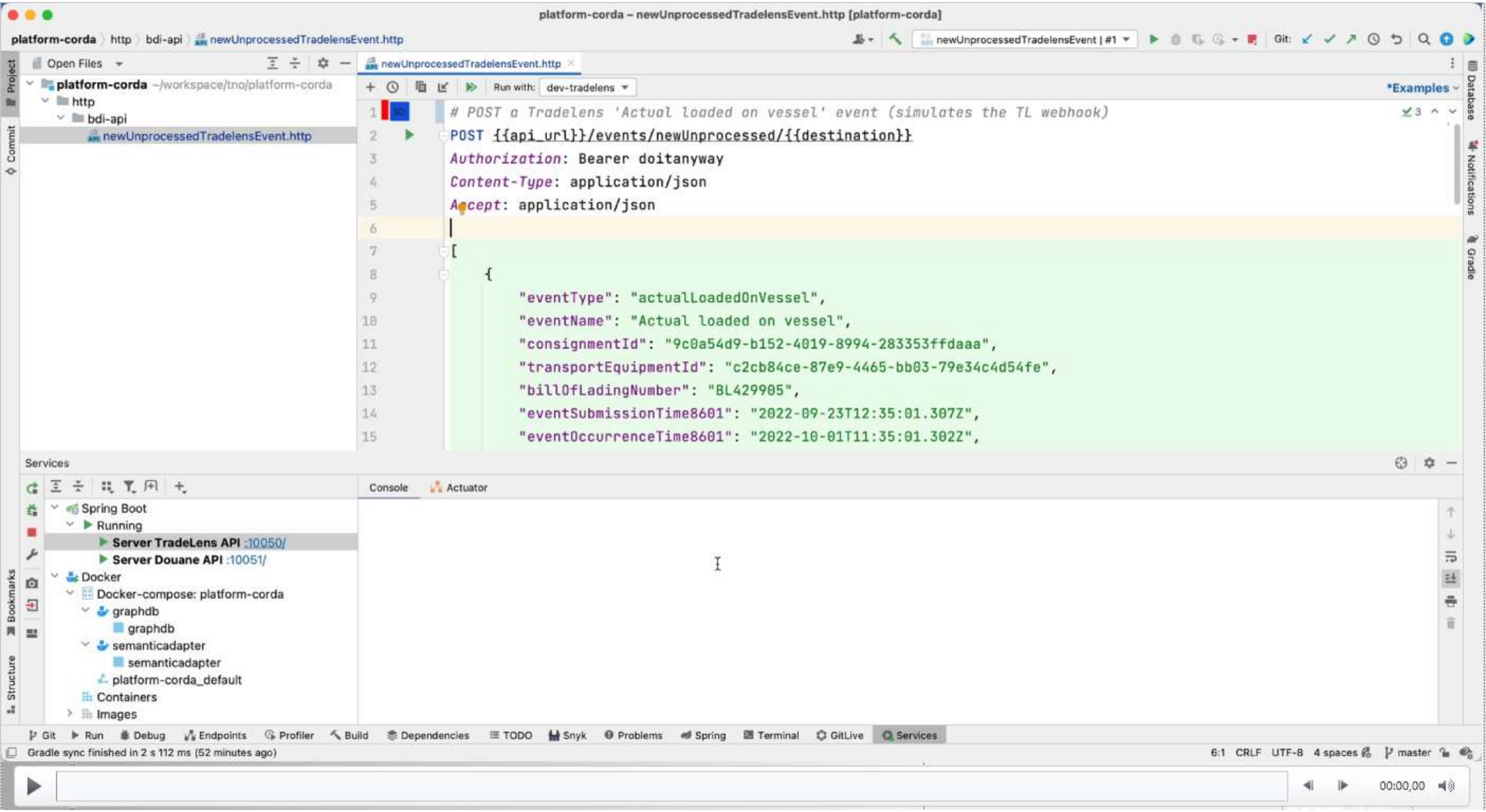


CORDA NETWORK
NODES RUNNING



API
SWAGGER UI




platform-corda – newUnprocessedTradelensEvent.http [platform-corda]
platform-corda 〉 http 〉 bdi-api 〉 newUnprocessedTradelensEvent.http
newUnprocessedTradelensEvent | #1
Git:
Open Files
platform-corda ~/workspace/tno/platform-corda
http
bdi-api
newUnprocessedTradelensEvent.http
Run with: dev-tradelens
*Examples
POST a Tradelens 'Actual loaded on vessel' event (simulates the TL webhook)
POST {{api_url}}/events/newUnprocessed/{{destination}}
Authorization: Bearer doitanyway
Content-Type: application/json
Accept: application/json
[
{
"eventType": "actualLoadedOnVessel",
"eventName": "Actual loaded on vessel",
"consignmentId": "9c0a54d9-b152-4019-8994-283353ffdaaa",
"transportEquipmentId": "c2cb84ce-87e9-4465-bb03-79e34c4d54fe",
"billOfLadingNumber": "BL429905",
"eventSubmissionTime8601": "2022-09-23T12:35:01.307Z",
"eventOccurrenceTime8601": "2022-10-01T11:35:01.302Z",
Services
Console
Actuator
Spring Boot
Running
Server TradeLens API :10050/
Server Douane API :10051/
Docker
Docker-compose: platform-corda
graphdb
graphdb
semanticadapter
semanticadapter
platform-corda_default
Containers
Images
Git
Run
Debug
Endpoints
Profiler
Build
Dependencies
TODO
Snyk
Problems
Spring
Terminal
GitLive
Services
Gradle sync finished in 2 s 112 ms (52 minutes ago)
6:1
CRLF
UTF-8
4 spaces
master
00:00,00

TRADELENS: NEW EVENT
DEMO

newUnprocessedTradelensEvent.http

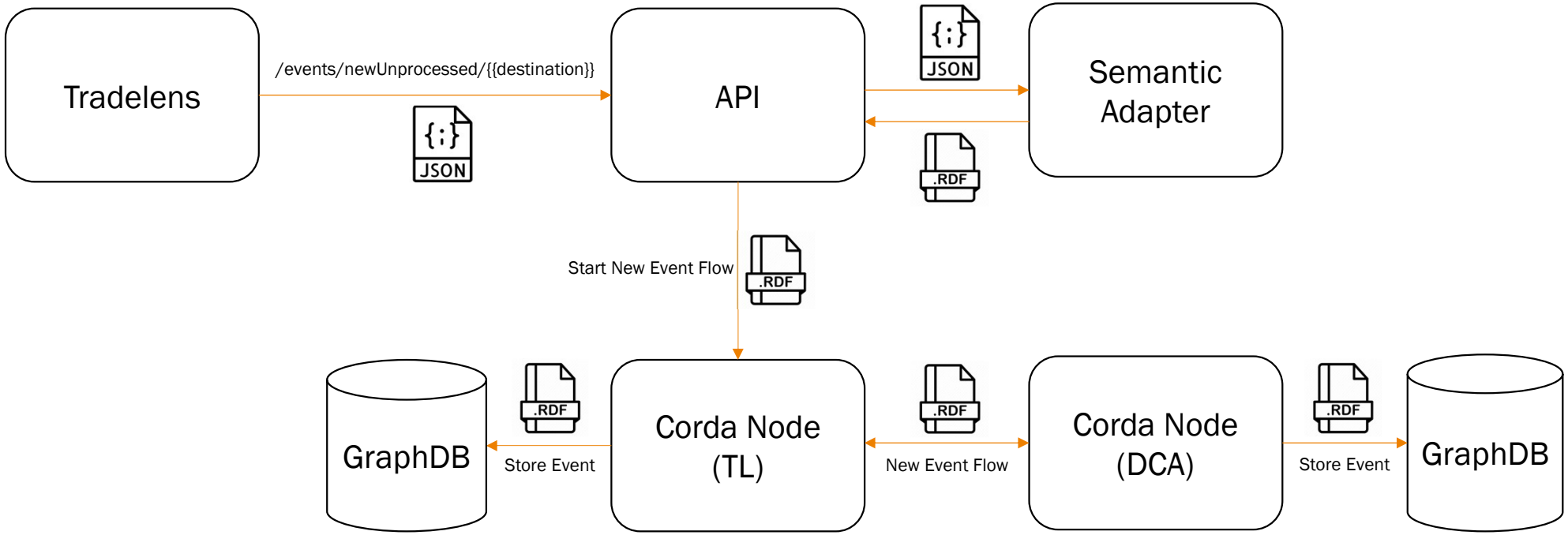
Run with: dev

```
# incoming tradelens event that would be submitted via a webhook
POST http://localhost:10050/events/newUnprocessed/{{destination}}
Authorization: Bearer doitanyway
Content-Type: application/json
Accept: application/json

[
    {
        "eventPublishTime": 1663936501514,
        "eventSubmissionTime": 1663936501307,
        "eventSubmissionTime8601": "2022-09-23T12:35:01.307Z",
        "eventName": "Actual loaded on vessel"
```

```
http://localhost:10050/events/newUnprocessed/BE

HTTP/1.1 201
Content-Type: application/json;charset=UTF-8
Content-Length: 51
Date: Thu, 03 Nov 2022 12:47:29 GMT

Event created: 12f828d8-329c-4578-8d87-ac62e2b8b7b4
Response file saved.
> 2022-11-03T134729.201.json

Response code: 201; Time: 12768ms (12 s 768 ms); Content length: 51 bytes (51 B)
```

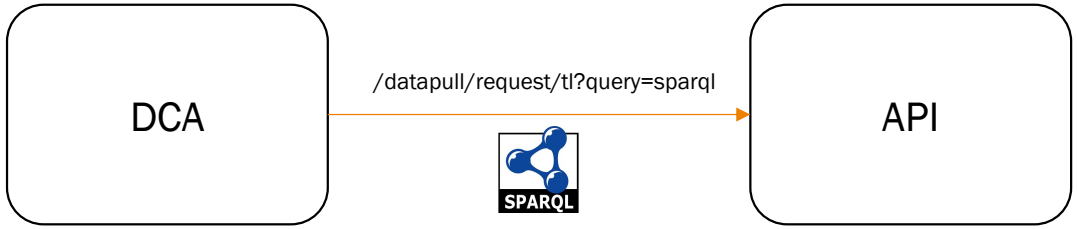
TNO innovation for life

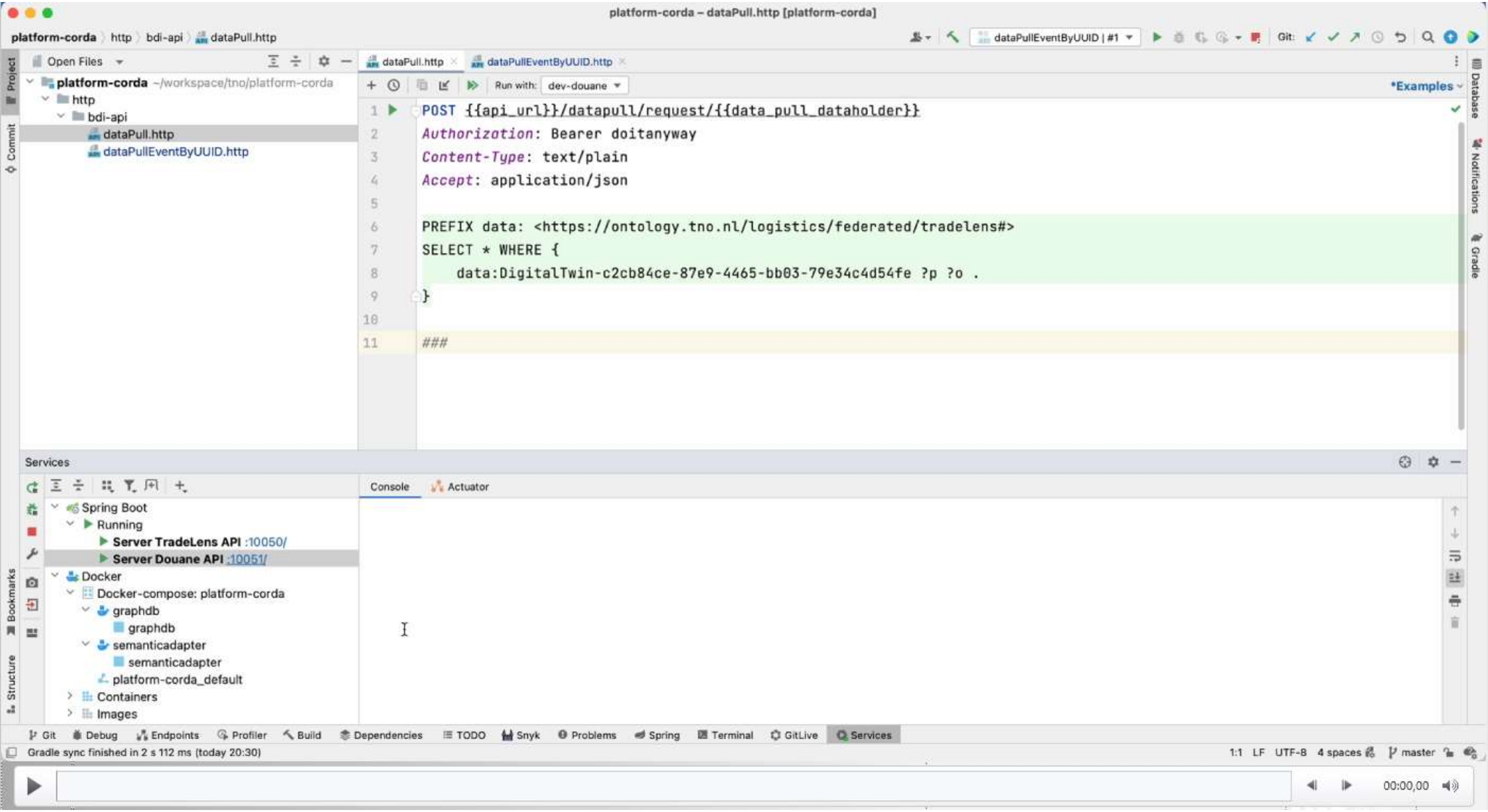
USE-CASE: TRADELENS EVENT
INTERNAL WORKING



USE-CASE: DCA QUERY (DATA PULL)

DEMO




platform-corda – dataPull.http [platform-corda]
platform-corda 〉 http 〉 bdi-api 〉 dataPull.http
dataPullEventByUUID | #1
Git:
Open Files
platform-corda ~/workspace/tno/platform-corda
http
bdi-api
dataPull.http
dataPullEventByUUID.http
dataPull.http
dataPullEventByUUID.http
Run with: dev-douane
*Examples
1 POST {{api_url}}/datapull/request/{{data_pull_dataholder}}
2 Authorization: Bearer doitanyway
3 Content-Type: text/plain
4 Accept: application/json
5
6 PREFIX data: <https://ontology.tno.nl/logistics/federated/tradelens#>
7 SELECT * WHERE {
8 data:DigitalTwin-c2cb84ce-87e9-4465-bb03-79e34c4d54fe ?p ?o .
9 }
10
11 ###
Services
Console
Actuator
Spring Boot
Running
Server TradeLens API :10050/
Server Douane API :10051/
Docker
Docker-compose: platform-corda
graphdb
graphdb
semanticadapter
semanticadapter
platform-corda_default
Containers
Images
Git
Debug
Endpoints
Profiler
Build
Dependencies
TODO
Snyk
Problems
Spring
Terminal
GitLive
Services
Gradle sync finished in 2 s 112 ms (today 20:30)
1:1
LF
UTF-8
4 spaces
master
00:00,00

USE-CASE: DCA QUERY (DATA PULL)

DEMO

dataPull.http

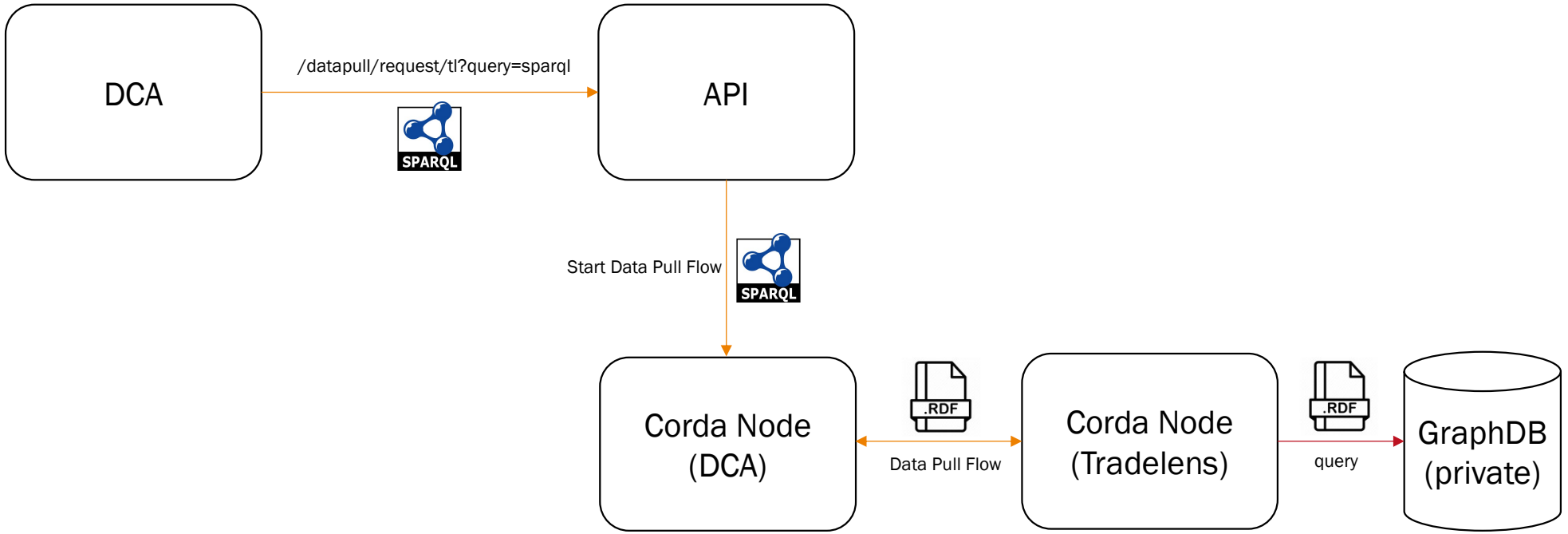
Run with: dev

```
POST http://localhost:10051/datapull/request/{{dataholder}}
Authorization: Bearer doitanyway
Content-Type: text/plain
Accept: application/json

PREFIX data: <https://ontology.tno.nl/logistics/federated/tradelens#>
SELECT * WHERE {
    data:DigitalTwin-c2cb84ce-87e9-4465-bb03-79e34c4d54fe ?p ?o .
}

###
```

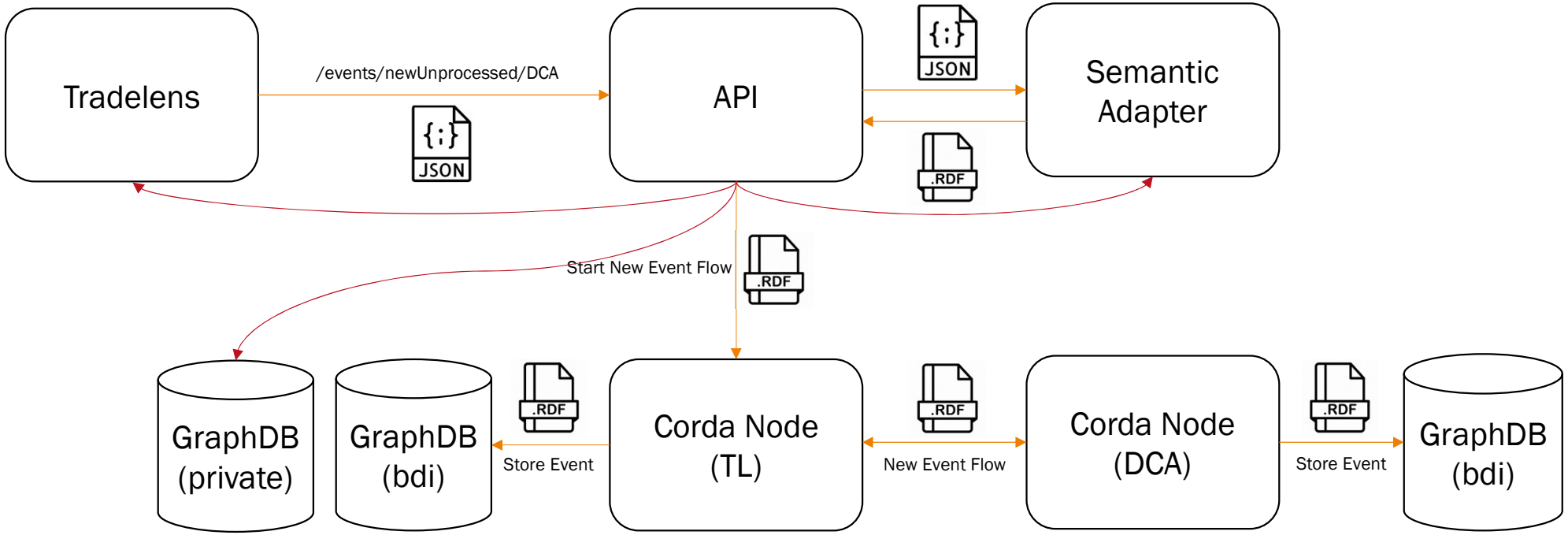
```
http://localhost:10051/datapull/request/Netherlands

HTTP/1.1 202
Content-Type: application/json;charset=UTF-8
Content-Length: 55
Date: Thu, 03 Nov 2022 13:37:35 GMT

State with result: 13351127-79a3-4ab1-b395-9d7eb0a3278f
Response file saved.
> 2022-11-03T143735.202.json

Response code: 202; Time: 6687ms (6 s 687 ms); Content length: 55 bytes (55 B)
```

TNO innovation for life

USE-CASE: DCA QUERY (DATA PULL)
INTERNAL WORKING

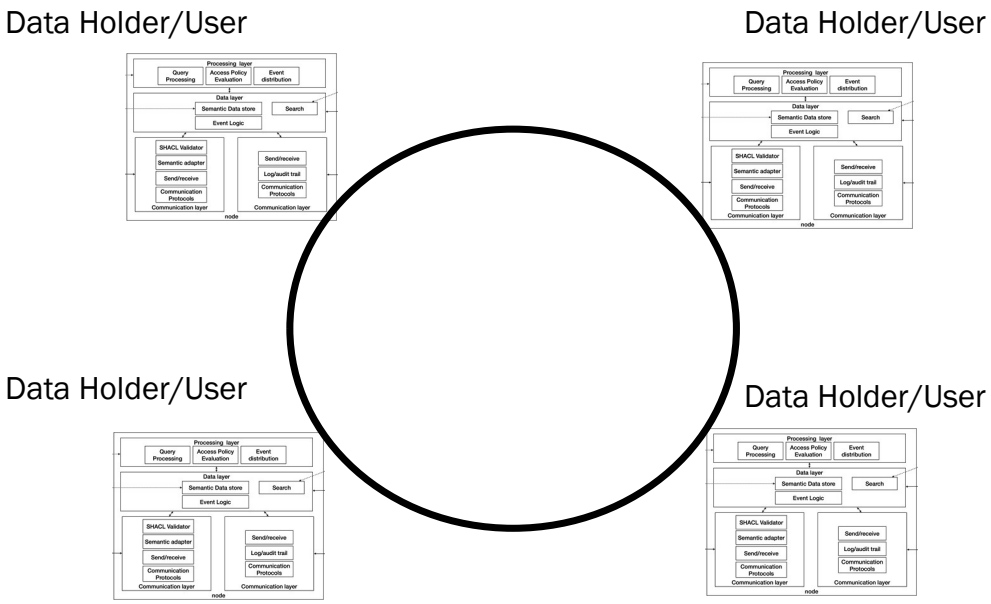


USE-CASE: TRADELENS EVENT
STORING EXTRA SEMANTIC DATA



2ND PROTOTYPE (MULTIPLE EVERYTHING)
NEXT STEPS

- PoC: multiple nodes, multiple events, multiple queries
  - Distribution algorithm(s)



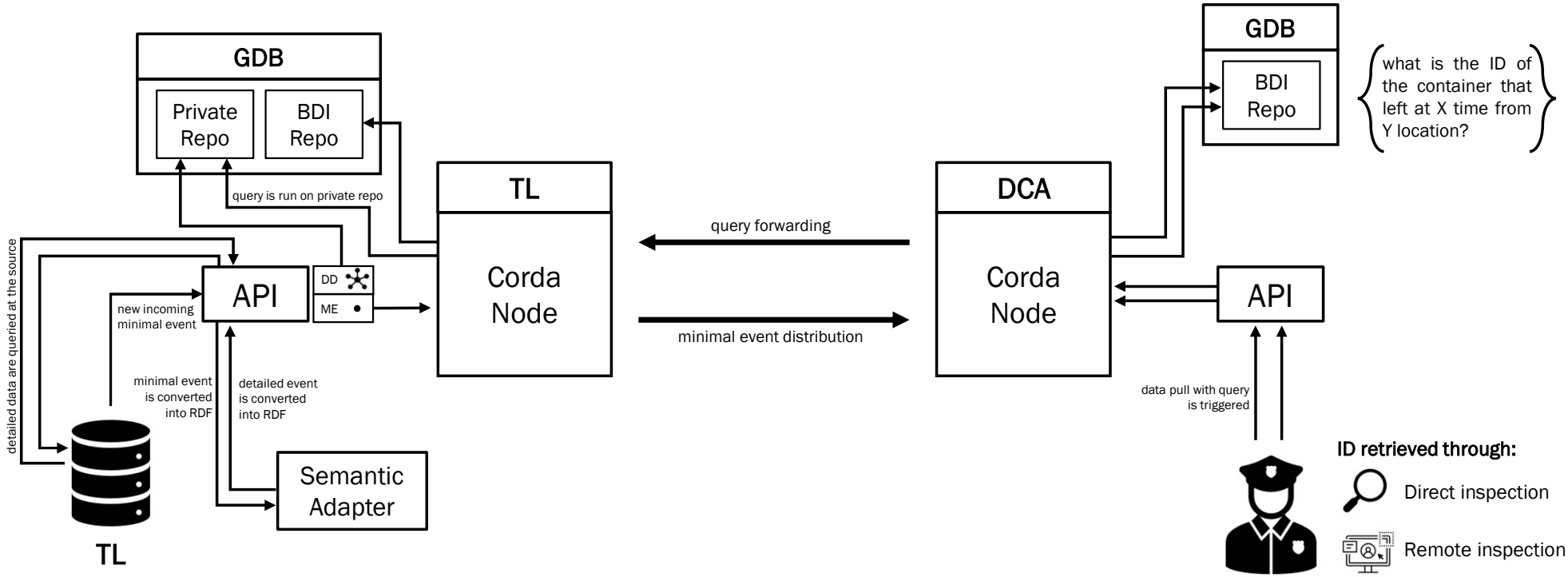
AFTER V0.2

- Event based access control
- Technology independed: Corda, IDSA, Peppol, Kafka, …
- Data at the source implementation: SPARQL, Triple store, API’s, …
- …

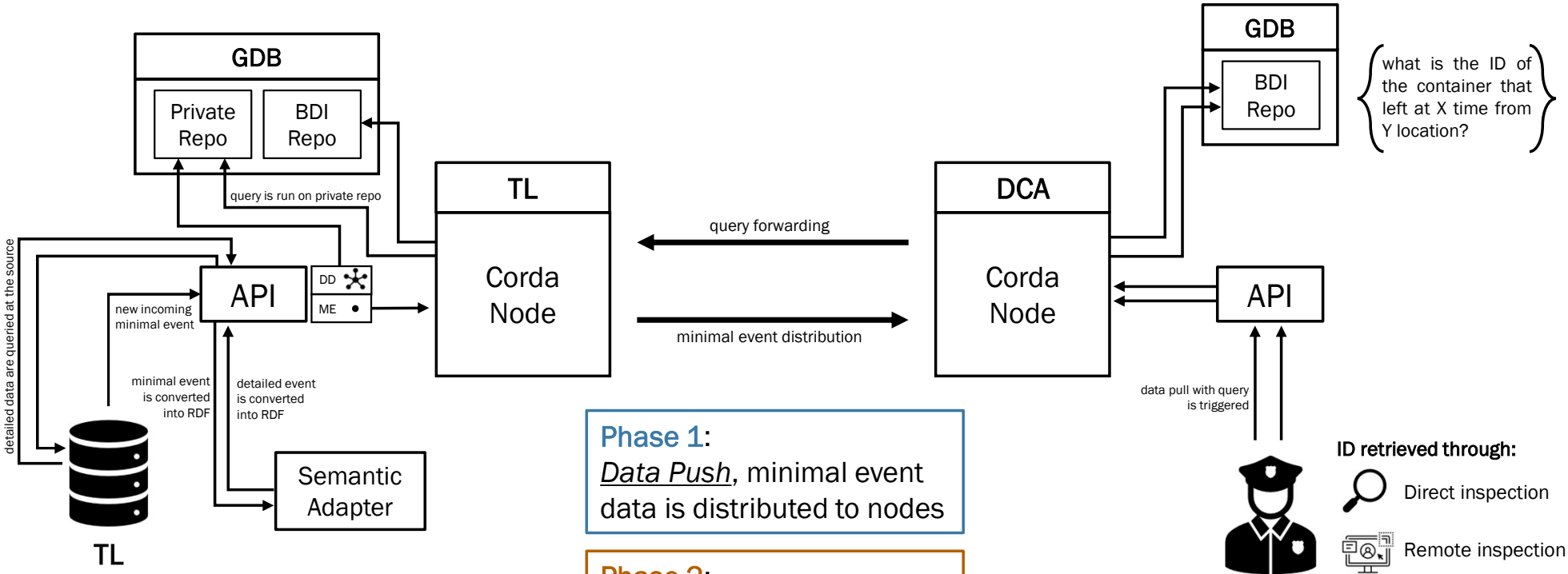
QUESTIONS?
THANK YOU FOR LISTENING

TNO innovation for life

FROM EVENT-SHARING TO EVENT-QUERYING



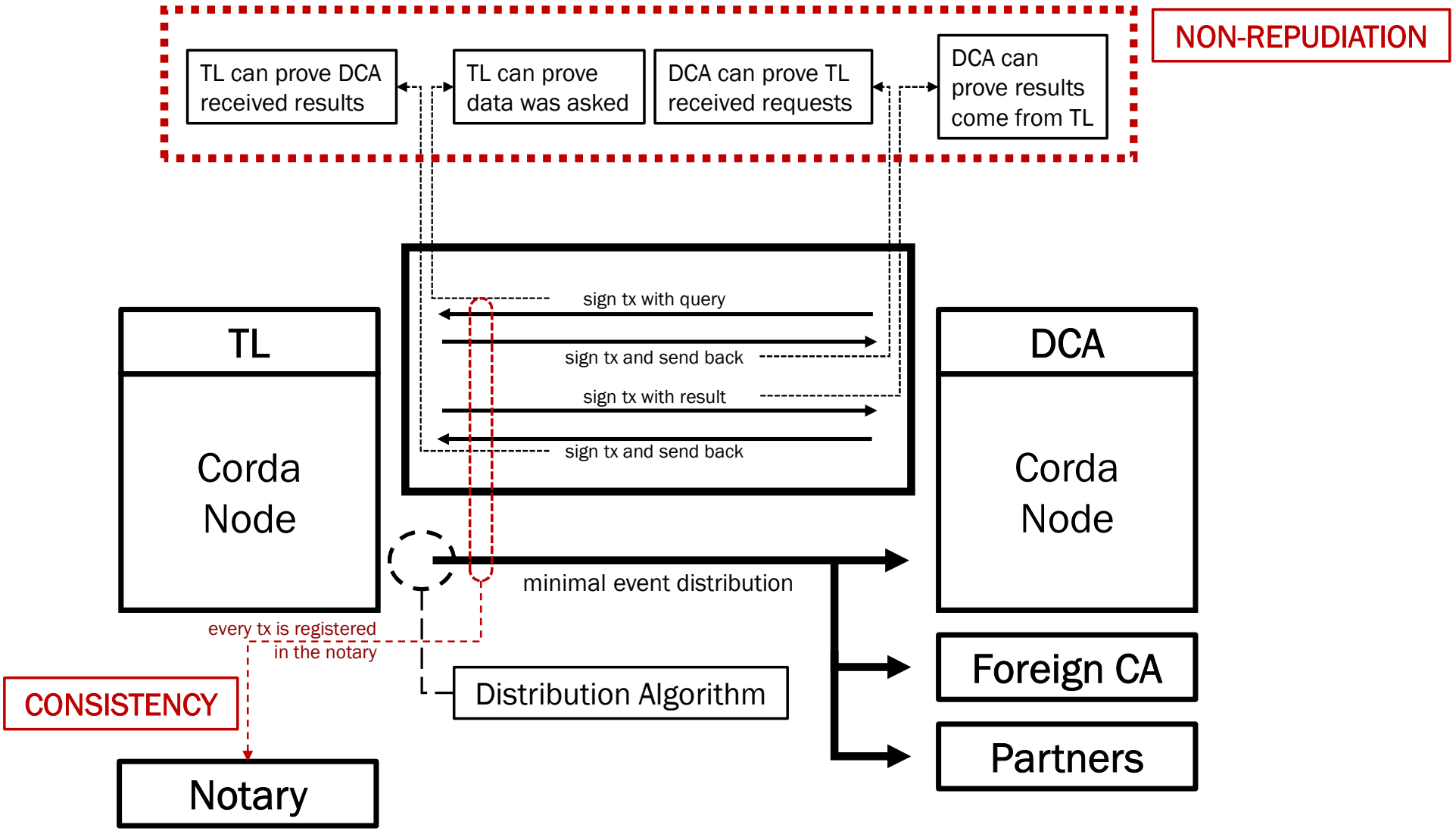
FROM EVENT-SHARING TO EVENT-QUERYING



Phase 1:
Data Push, minimal event data is distributed to nodes

Phase 2:
Data Pull, detailed data are queried at the source

TNO innovation for life


NON-REPUDIATION
TL can prove DCA received results
TL can prove data was asked
DCA can prove TL received requests
DCA can prove results come from TL
sign tx with query
sign tx and send back
sign tx with result
sign tx and send back
TL
Corda Node
DCA
Corda Node
minimal event distribution
every tx is registered in the notary
CONSISTENCY
Distribution Algorithm
Foreign CA
Partners
Notary