V O L V O

Volvo Trucks

Bojan Alempijevic

Product Owner for team Product Metrics

VSF (Vehicle Software Factory)

VOLVO

Applying Eiffel to drive data-driven software releases for trucks

1: Our software domain

2: What do we try to solve?

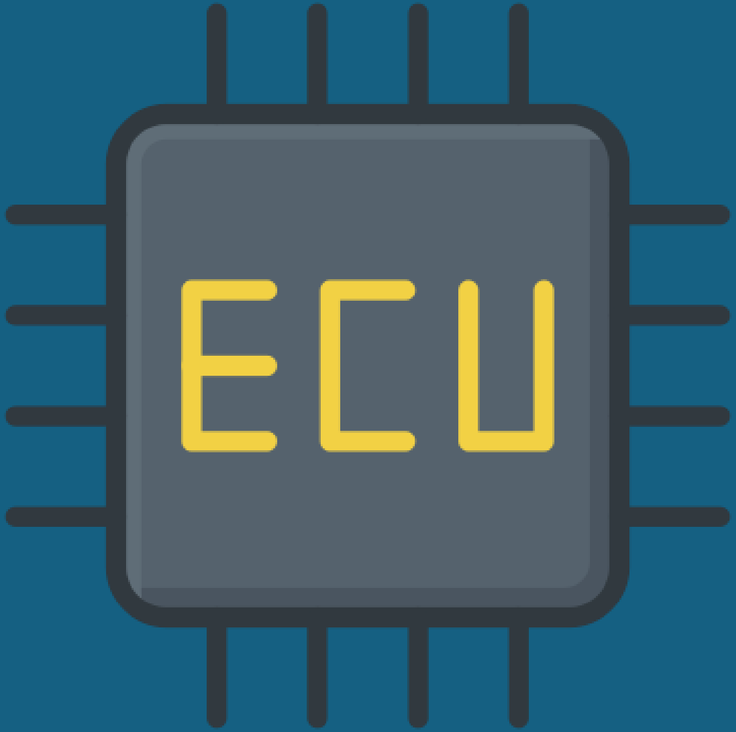
3: How do we map our software domain to Eiffel?

4: Using Neo4j for Eiffel data

Our software domain

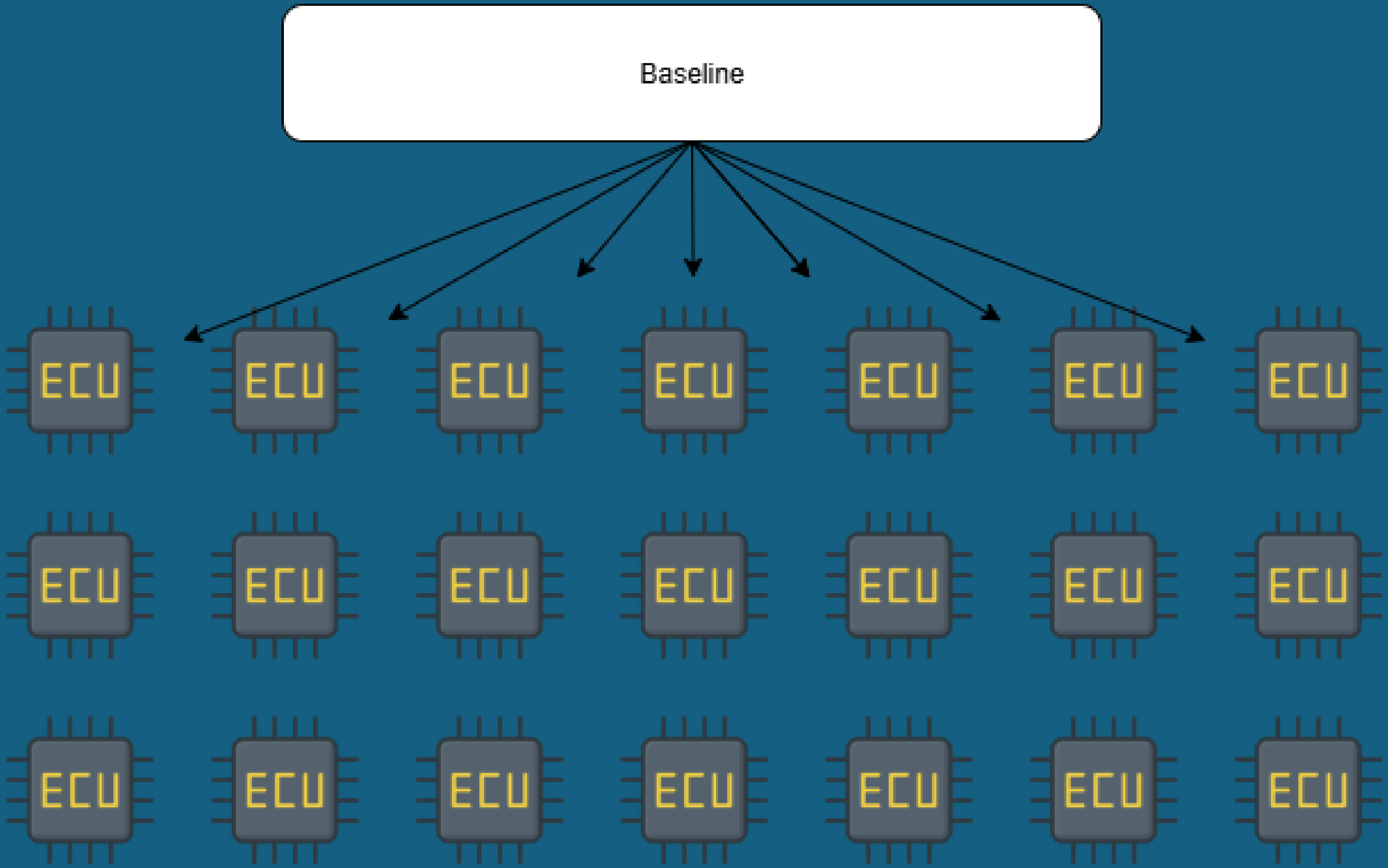
VOLVO

~20-25 ECU’s (Electronic Control Unit)

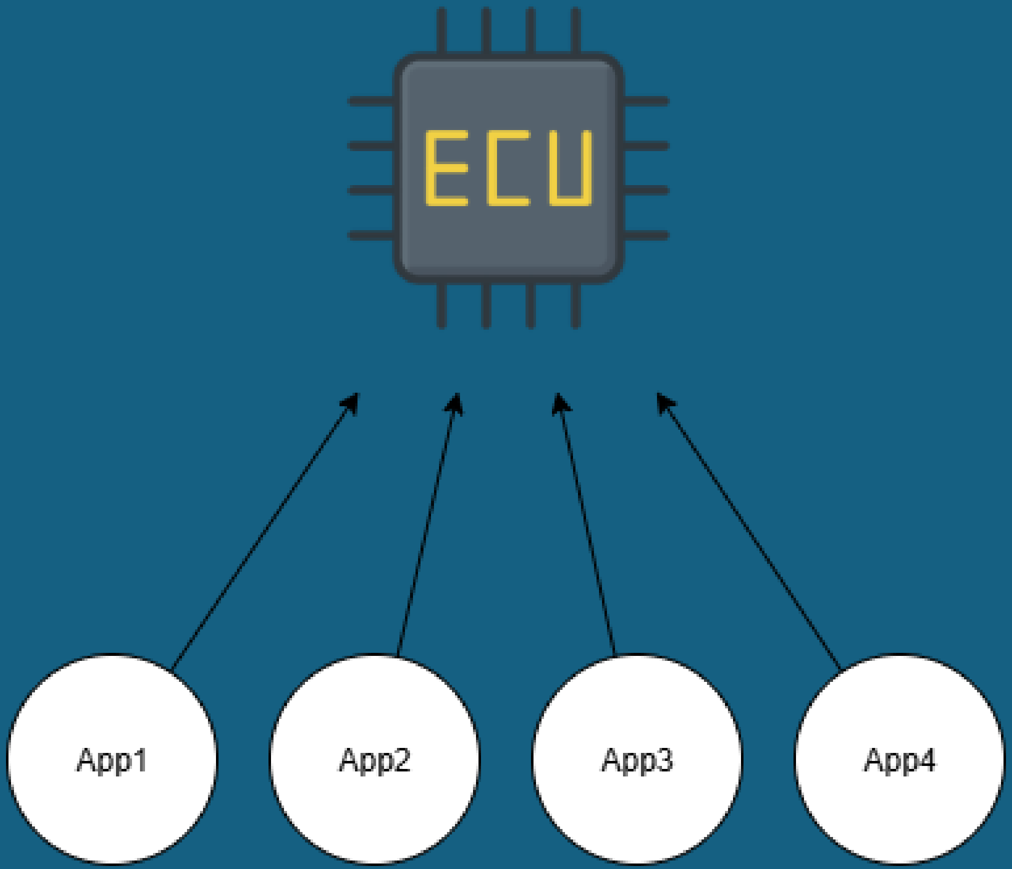


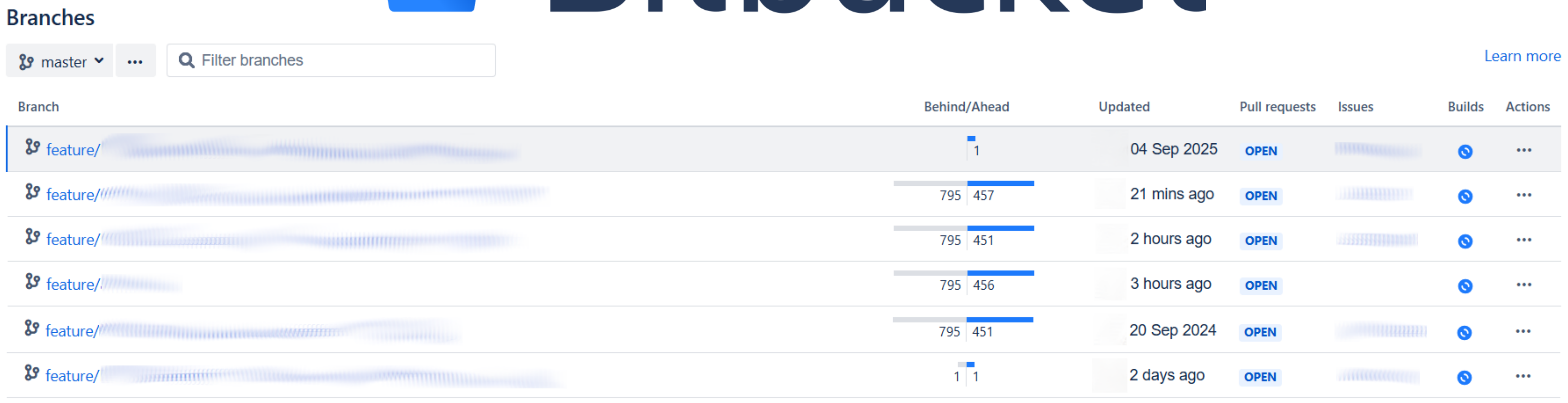
Baseline (Truck configuration manifest over the software on each ECU)

VOLVO



Different types of ECU deployments





Branches

master ••• Filter branches

Learn more

Branch	Behind/Ahead	Updated	Pull requests	Issues	Builds	Actions
feature/[redacted]	1	04 Sep 2025	OPEN	[redacted]	⊘	•••
feature/[redacted]	795 | 457	21 mins ago	OPEN	[redacted]	⊘	•••
feature/[redacted]	795 | 451	2 hours ago	OPEN	[redacted]	⊘	•••
feature/[redacted]	795 | 456	3 hours ago	OPEN		⊘	•••
feature/[redacted]	795 | 451	20 Sep 2024	OPEN	[redacted]	⊘	•••
feature/[redacted]	1 | 1	2 days ago	OPEN	[redacted]	⊘	•••

VOLVO

Bitbucket

Git submodules - .gitmodules file
BitBake - .bb files

VOLVO

What do we try to solve?

* Extensive release processes

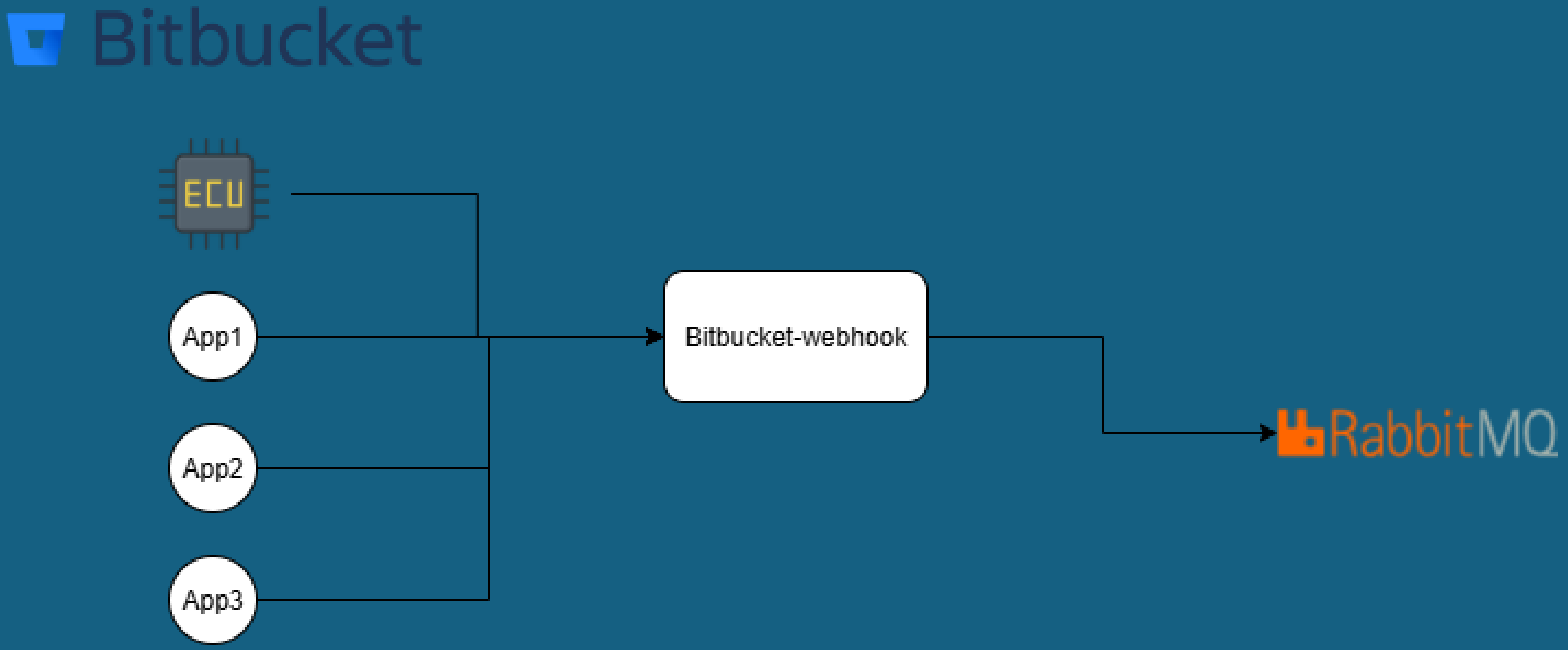
* Testing (SIL, HIL, Signal testing, Driving)

* Gathering of data from many systems and streams

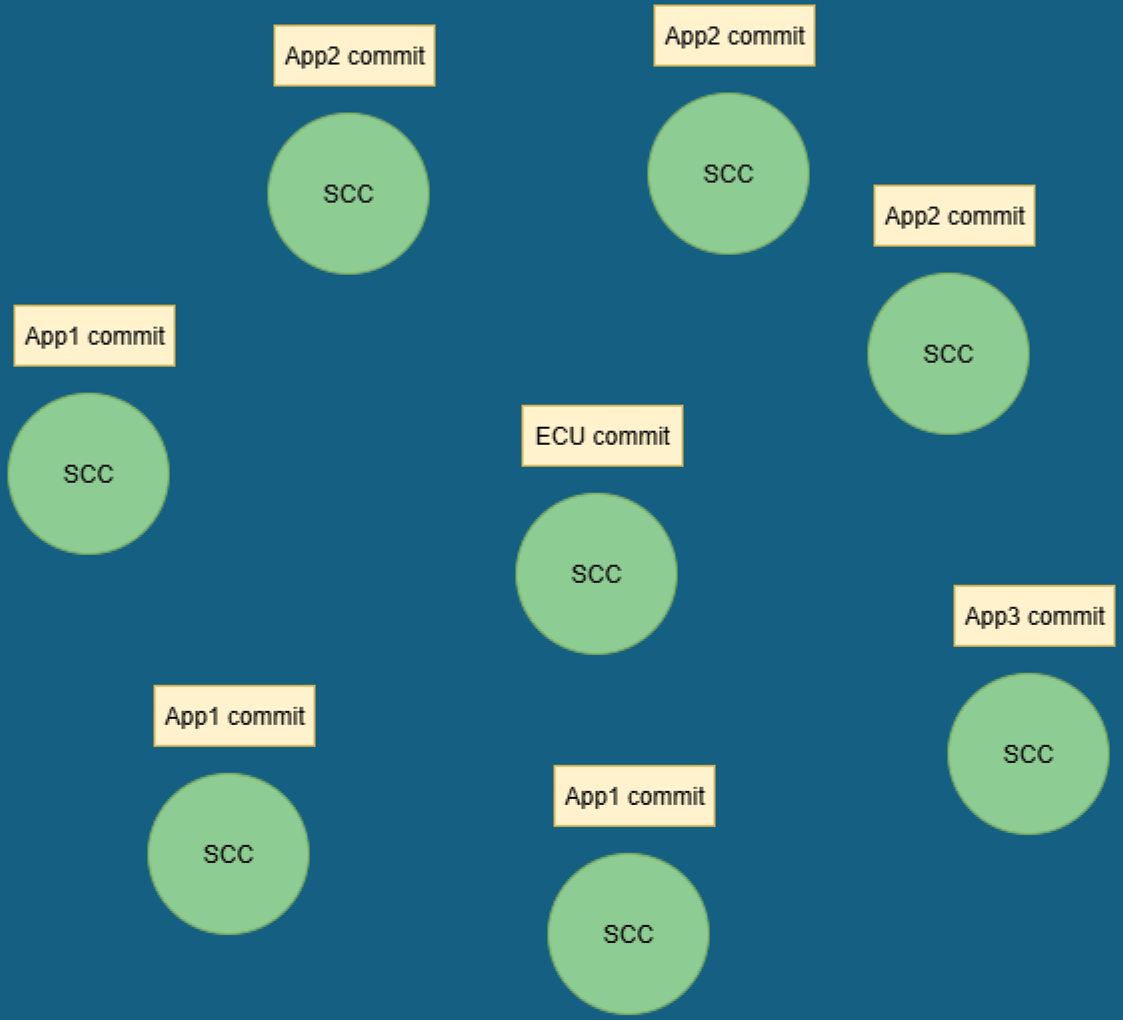
* Take decisions – **Release Board**

VOLVO

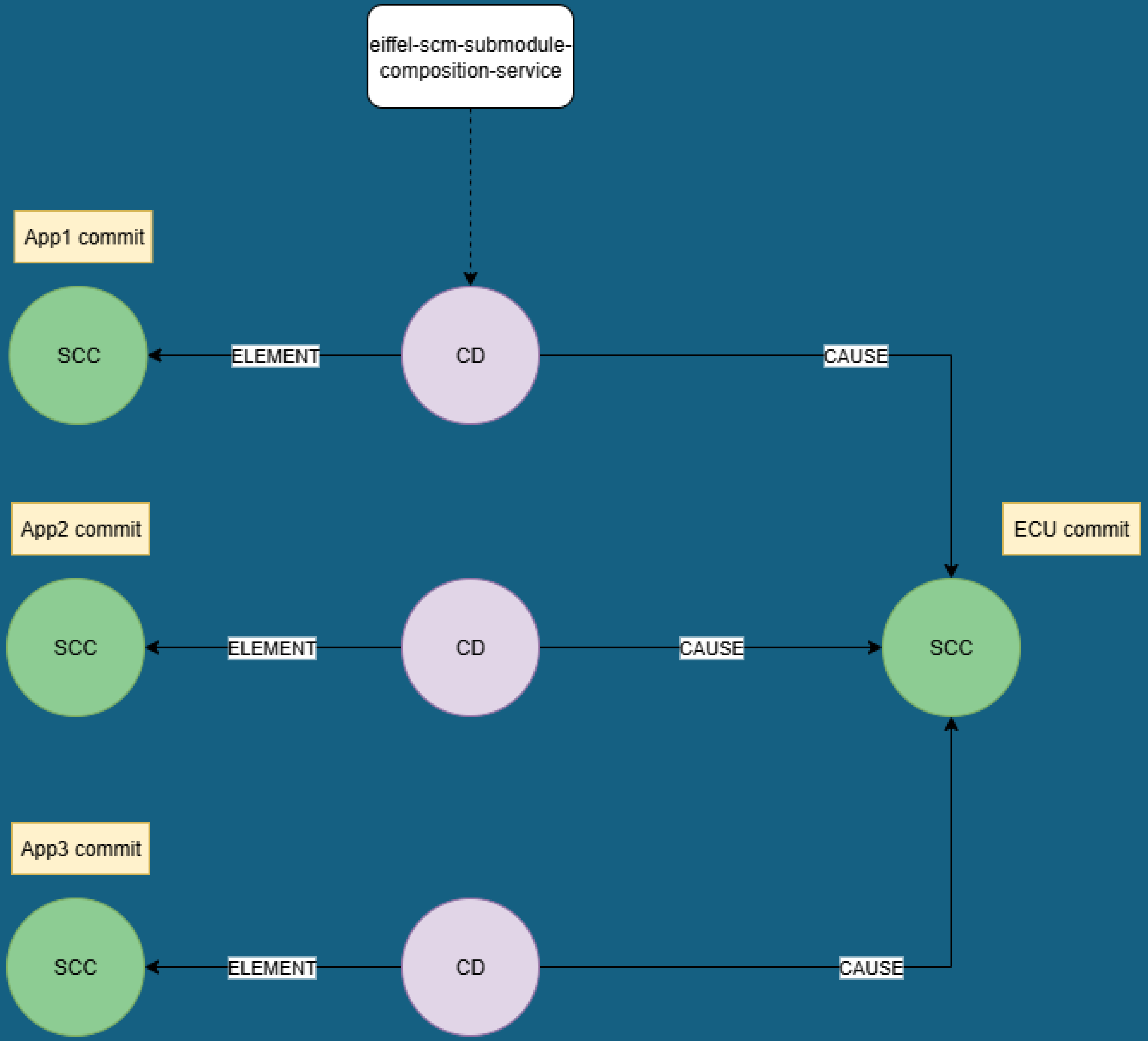
Eiffel


Bitbucket
ECU
App1
App2
App3
Bitbucket-webhook
RabbitMQ

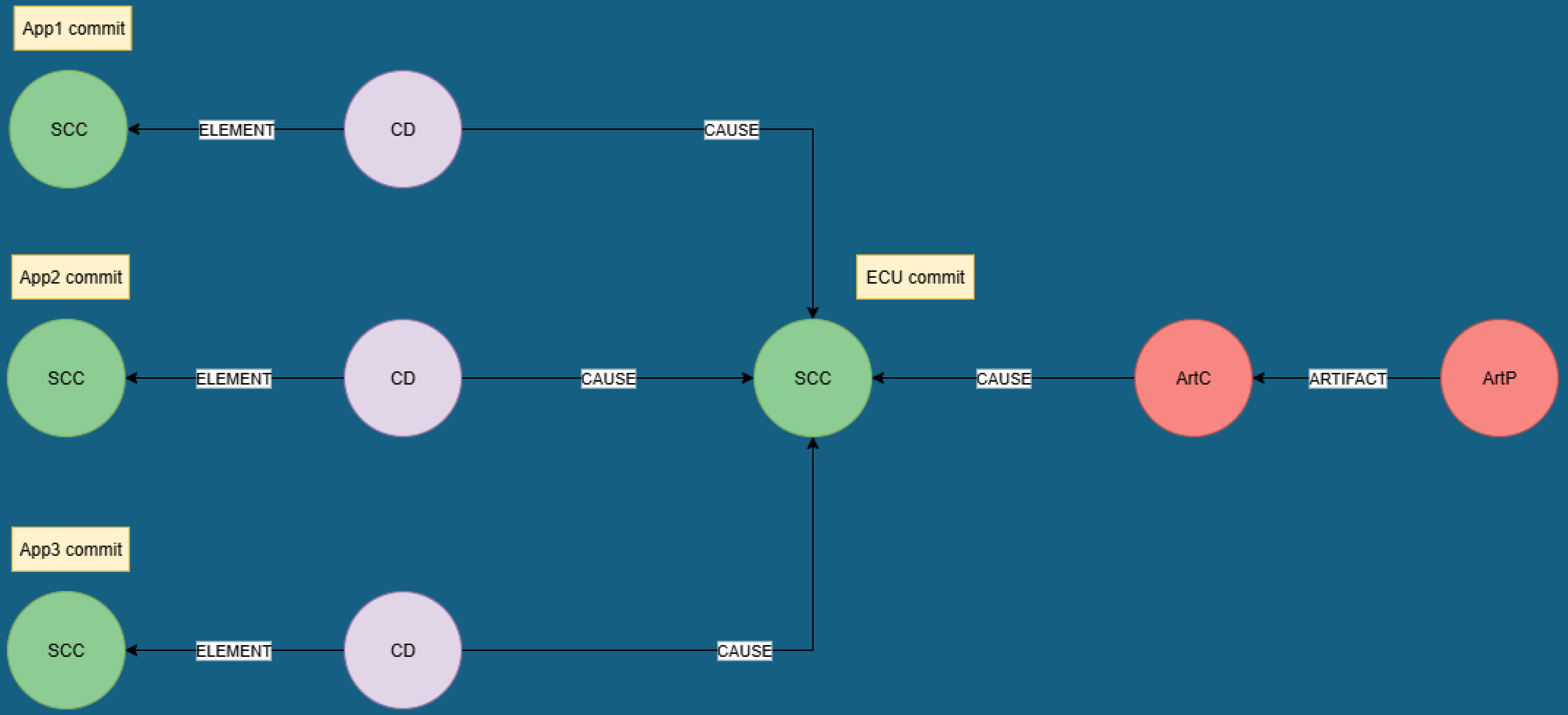
VOLVO


App2 commit
App2 commit
SCC
SCC
App2 commit
SCC
App1 commit
SCC
ECU commit
SCC
App3 commit
SCC
App1 commit
SCC
App1 commit
SCC

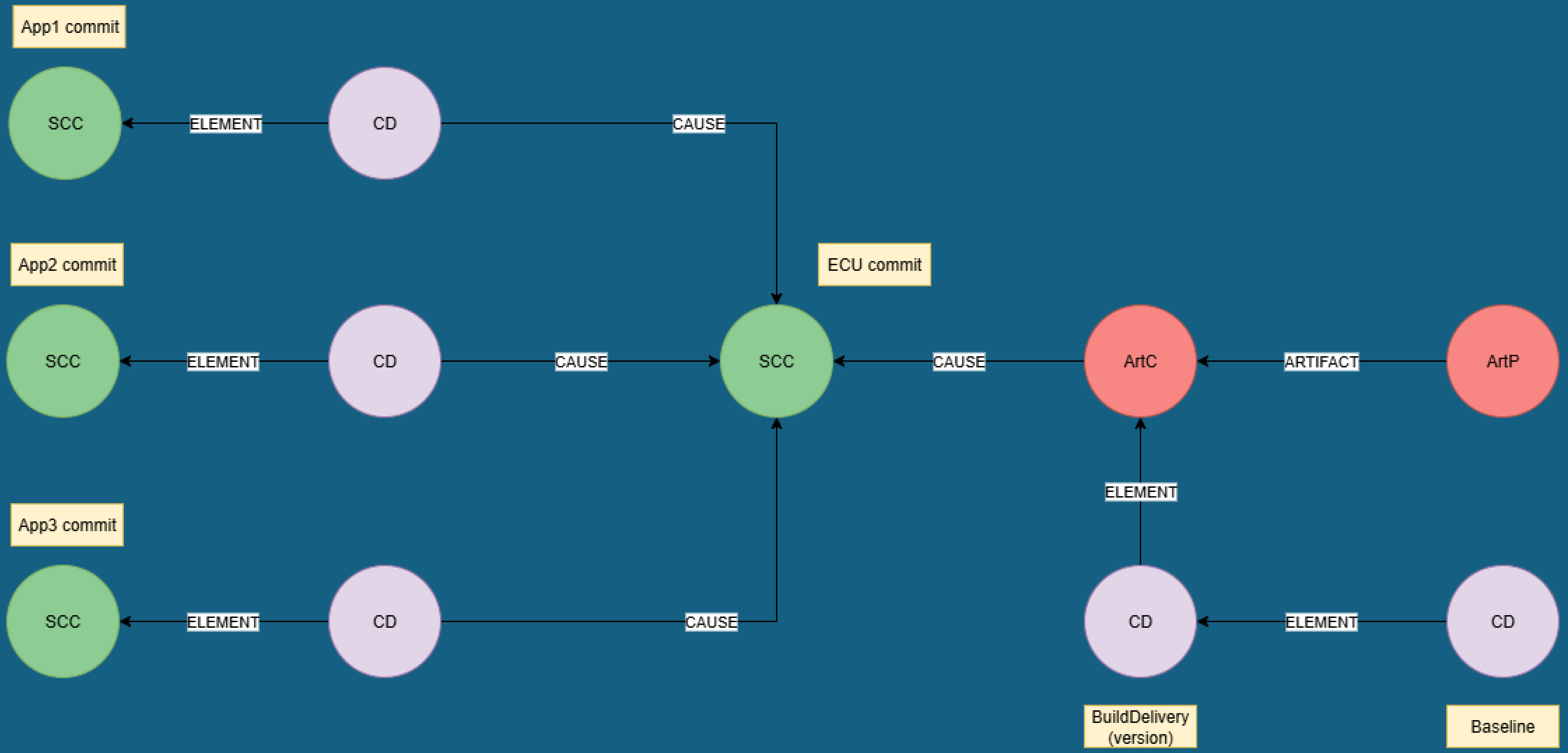
VOLVO


eiffel-scm-submodule-composition-service
App1 commit
App2 commit
App3 commit
ECU commit
SCC
CD
SCC
CD
SCC
CD
SCC
ELEMENT
ELEMENT
ELEMENT
CAUSE
CAUSE
CAUSE

VOLVO

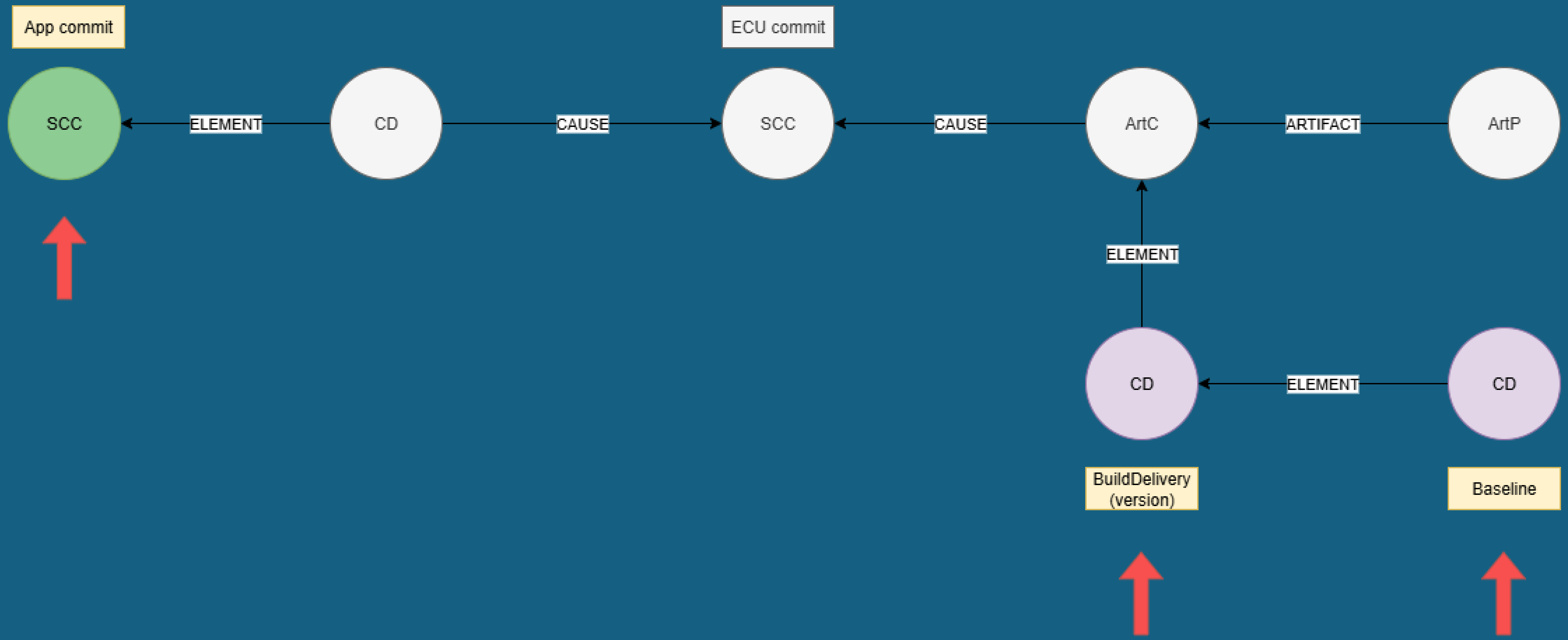


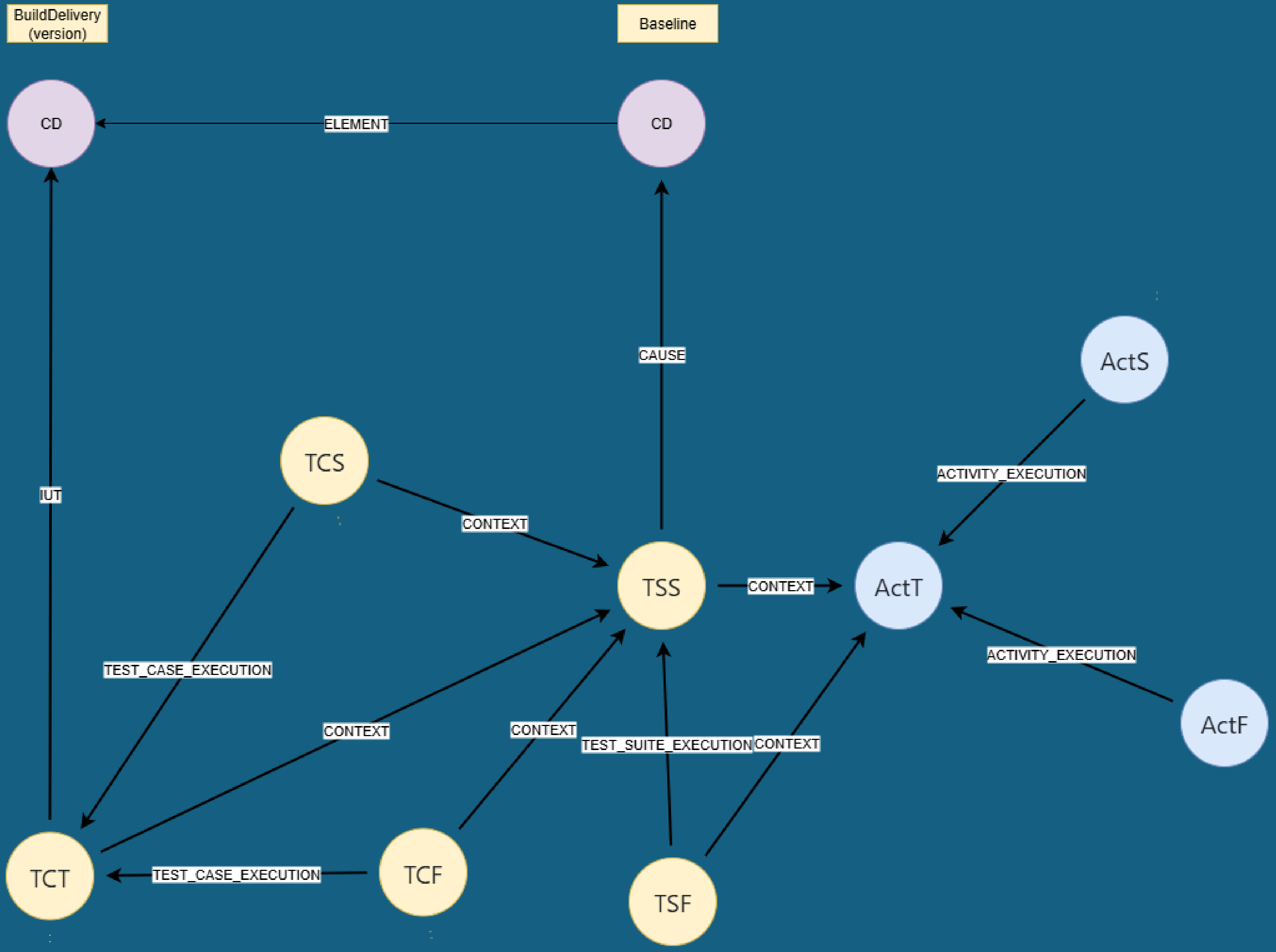
App1 commit
SCC
ELEMENT
CD
CAUSE
App2 commit
SCC
ELEMENT
CD
CAUSE
ECU commit
SCC
CAUSE
ArtC
ARTIFACT
ArtP
App3 commit
SCC
ELEMENT
CD
CAUSE

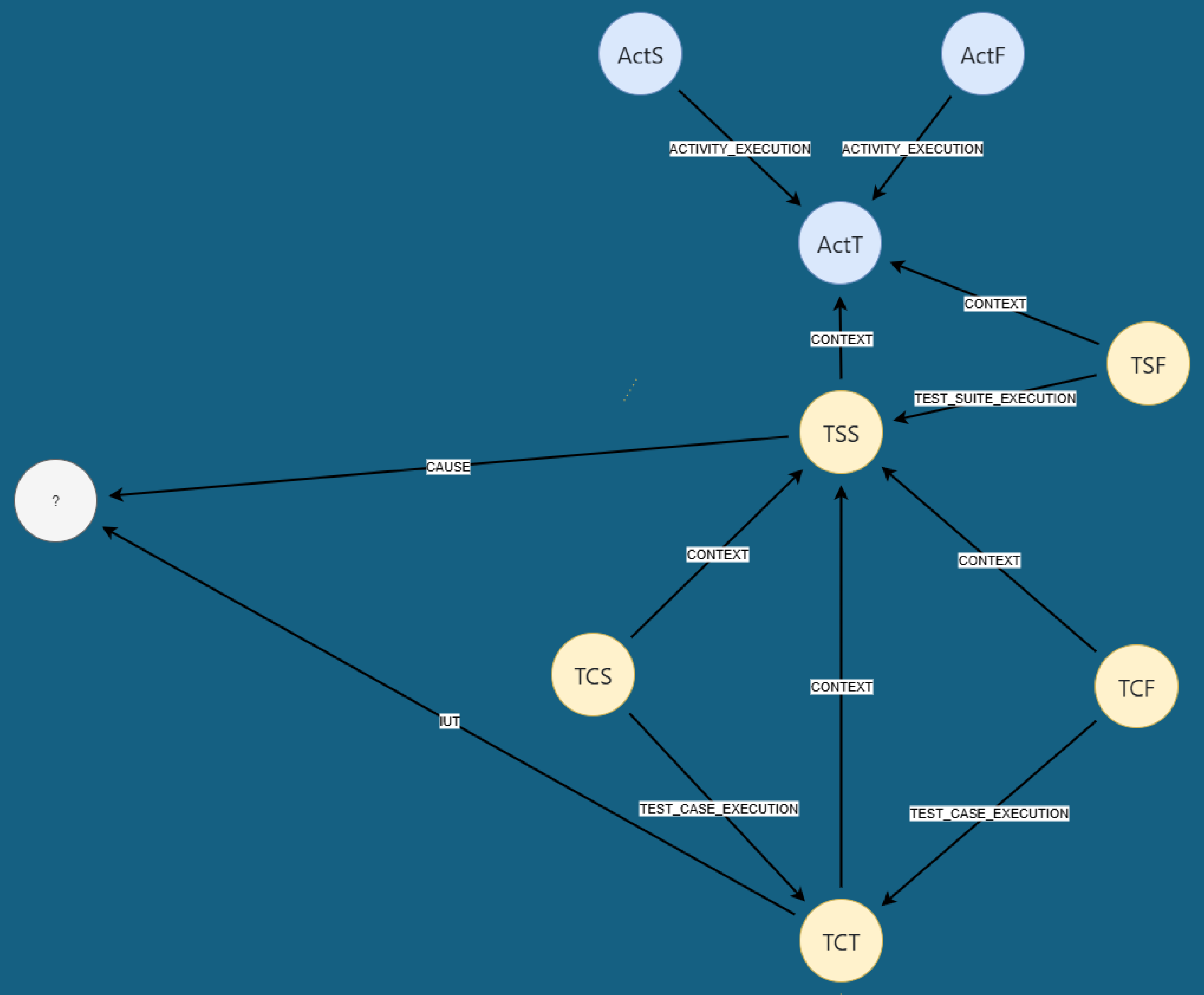

App1 commit
SCC
ELEMENT
CD
CAUSE
App2 commit
SCC
ELEMENT
CD
CAUSE
ECU commit
SCC
CAUSE
ArtC
ARTIFACT
ArtP
ELEMENT
App3 commit
SCC
ELEMENT
CD
CAUSE
CD
ELEMENT
CD
BuildDelivery (version)
Baseline

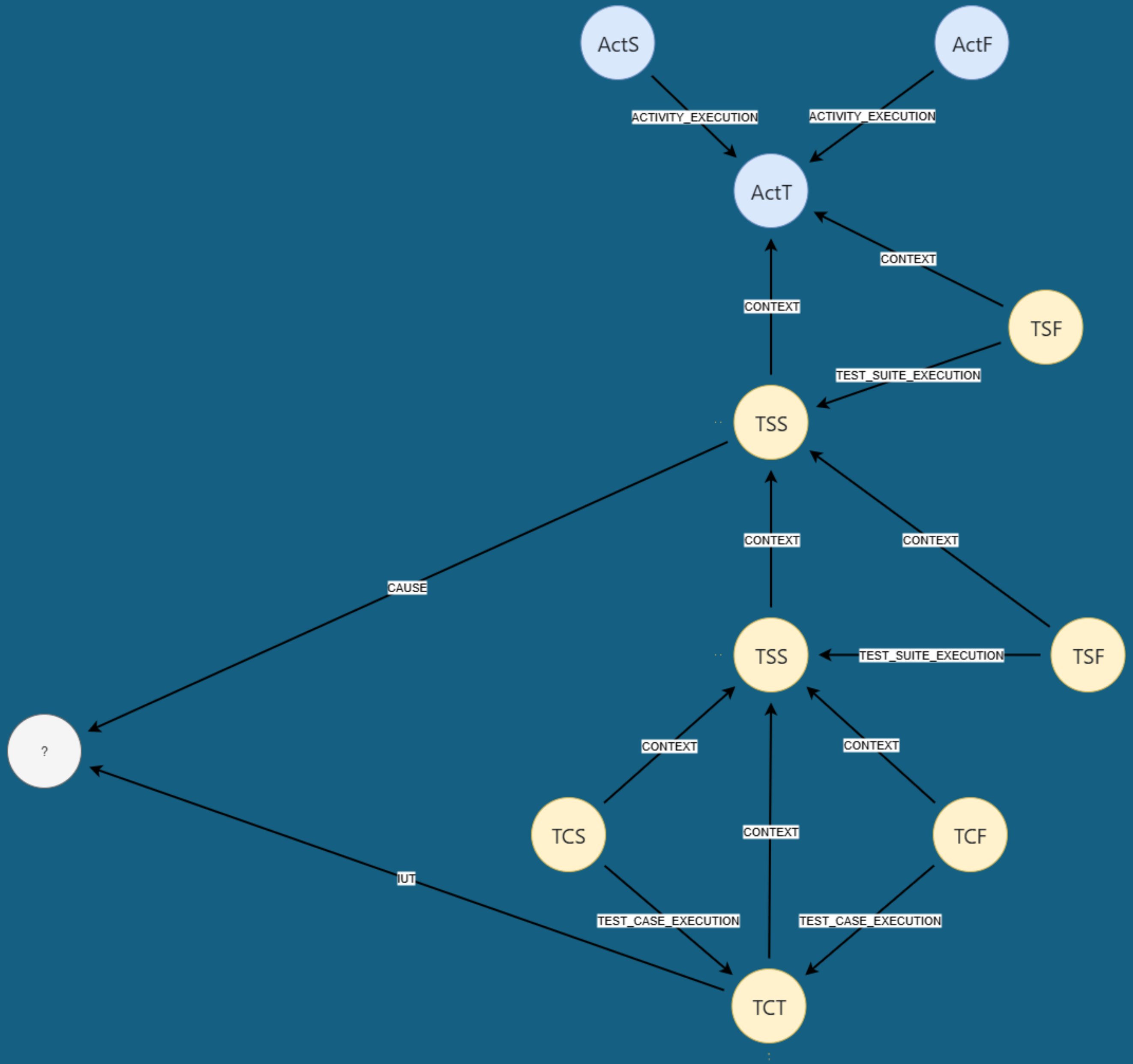
VOLVO

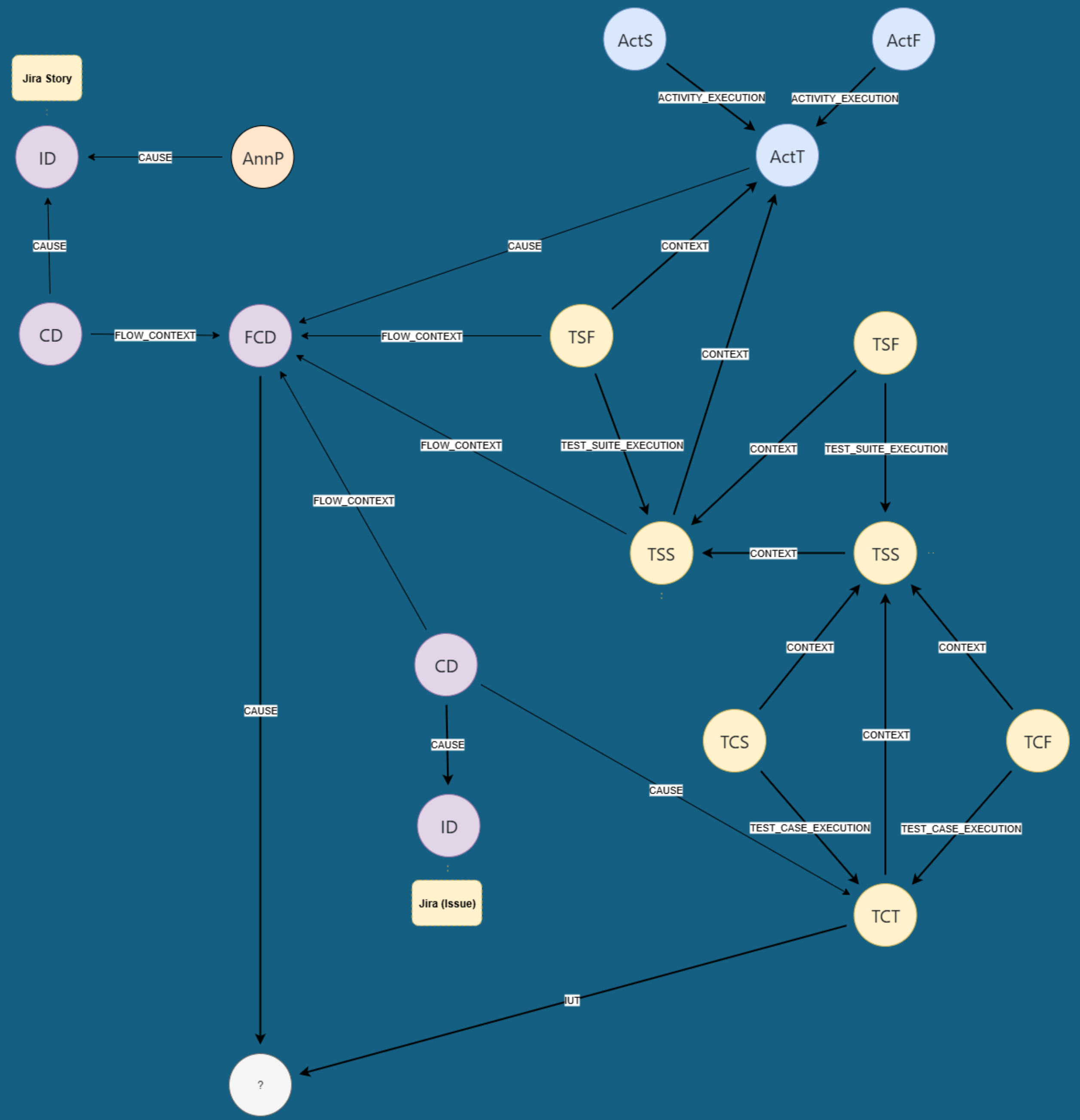
Add Tests


App commit
SCC
ELEMENT
CD
CAUSE
ECU commit
SCC
CAUSE
ArtC
ARTIFACT
ArtP
ELEMENT
CD
ELEMENT
CD
BuildDelivery (version)
Baseline


BuildDelivery (version)
Baseline
CD
ELEMENT
CD
CAUSE
IUT
ActS
ACTIVITY_EXECUTION
TCS
CONTEXT
TSS
CONTEXT
ActT
TEST_CASE_EXECUTION
ACTIVITY_EXECUTION
CONTEXT
CONTEXT
TEST_SUITE_EXECUTION
CONTEXT
ActF
TCT
TEST_CASE_EXECUTION
TCF
TSF


ActS
ActF
ACTIVITY_EXECUTION
ACTIVITY_EXECUTION
ActT
CONTEXT
CONTEXT
TSF
TEST_SUITE_EXECUTION
TSS
CAUSE
?
CONTEXT
CONTEXT
CONTEXT
TCS
TCF
IUT
TEST_CASE_EXECUTION
TEST_CASE_EXECUTION
TCT


ActS
ActF
ACTIVITY_EXECUTION
ACTIVITY_EXECUTION
ActT
CONTEXT
CONTEXT
TSF
TEST_SUITE_EXECUTION
TSS
CAUSE
CONTEXT
CONTEXT
TSS
TEST_SUITE_EXECUTION
TSF
?
CONTEXT
CONTEXT
TCS
CONTEXT
TCF
IUT
TEST_CASE_EXECUTION
TEST_CASE_EXECUTION
TCT


Jira Story
ID
CAUSE
AnnP
CAUSE
CD
FLOW_CONTEXT
FCD
ActS
ActF
ACTIVITY_EXECUTION
ACTIVITY_EXECUTION
ActT
CAUSE
CONTEXT
FLOW_CONTEXT
TSF
CONTEXT
TSF
FLOW_CONTEXT
TEST_SUITE_EXECUTION
CONTEXT
TEST_SUITE_EXECUTION
FLOW_CONTEXT
TSS
CONTEXT
TSS
CD
CONTEXT
CONTEXT
CAUSE
CAUSE
TCS
CONTEXT
TCF
ID
CAUSE
Jira (Issue)
TEST_CASE_EXECUTION
TEST_CASE_EXECUTION
TCT
IUT
?

VOLVO

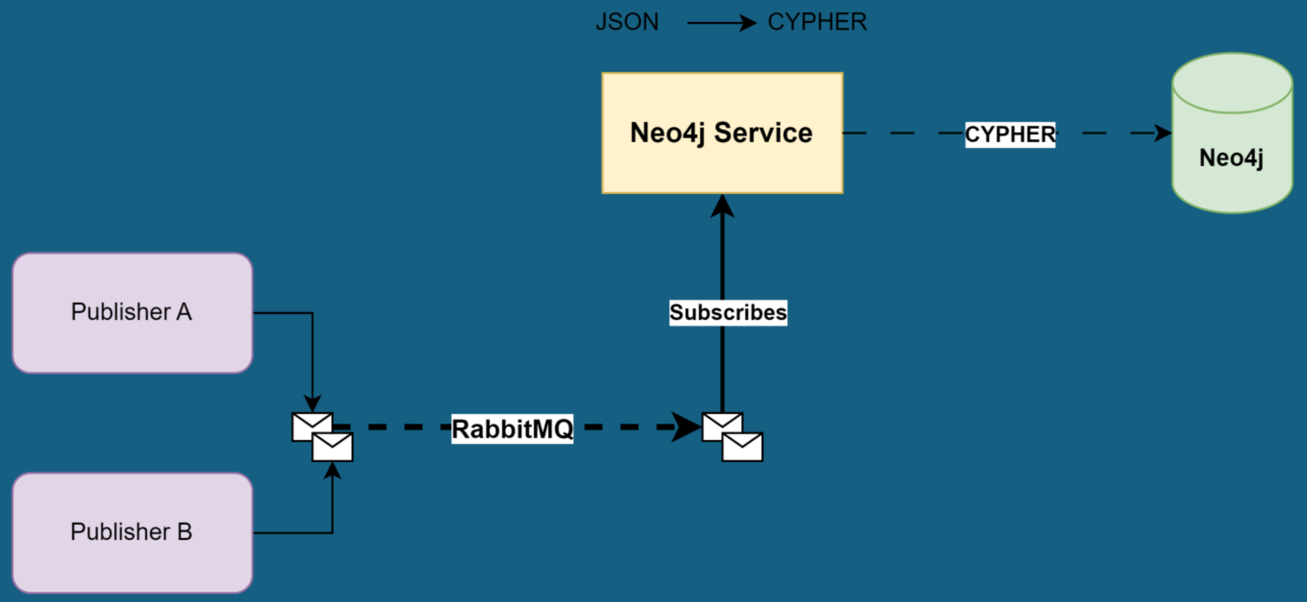
Working together with the teams who has data to define the model

VOLVO

neo4j

VOLVO

The **Neo4j Data Importer**



VOLVO

Simple example

RabbitMQ - JSON format

```
{
 "meta": {
  "id": "1111-2222-3333-4444",
  "type": "EiffelSourceChangeCreatedEvent"
 },
 "data": {
  "commitId": "abc123"
 }
}
```

Neo4j - Cypher format

```
MERGE (scc:Eiffel {id: "1111-2222-3333-4444"})

ON CREATE SET
 scc.type = "EiffelSourceChangeCreatedEvent",
 scc.commitId = "abc123”

RETURN e;
```

*Additional cyphers for **tags** and **links** follow similar pattern*

VOLVO

Abbreviation

Neo4j - Cypher format

```
case 'EiffelSourceChangeCreatedEvent':
    abbreviation = 'SCC';
    break;
```

Indexing

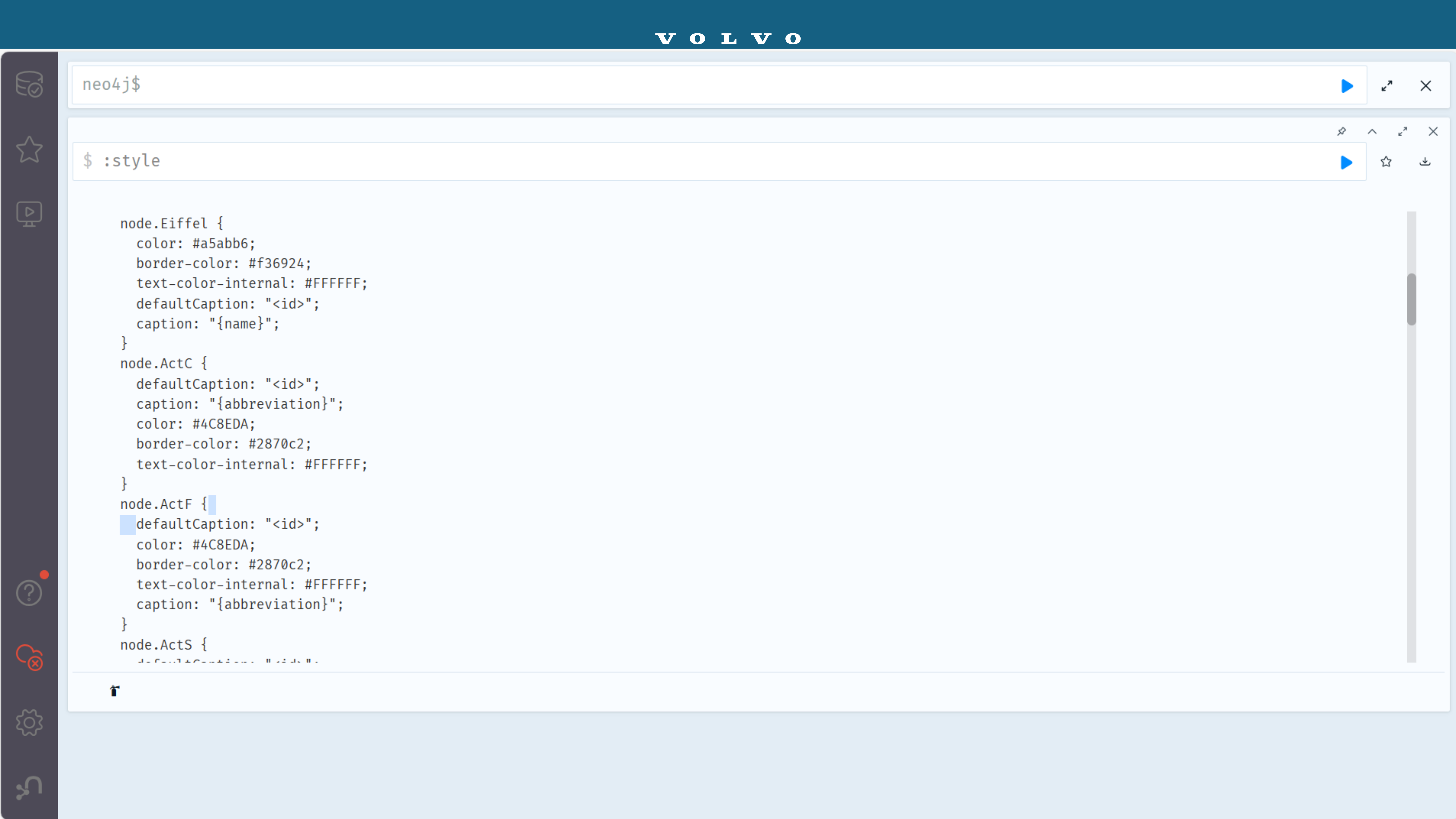
Neo4j - Cypher format

```
CREATE INDEX SCC_commitId IF NOT EXISTS FOR (n:SCC) ON (n.commitId);
```

Theme neo4j browser

eiffel_style.grass

```
node.Eiffel {
  color: #a5abb6;
  border-color: #f36924;
  text-color-internal: #FFFFFF;
  defaultCaption: "<id>";
  caption: "{name}";
}
node.ActC {
  defaultCaption: "<id>";
  caption: "{abbreviation}";
  color: #4C8EDA;
  border-color: #2870c2;
  text-color-internal: #FFFFFF;
}
```



```
$ :style
```

```
node.Eiffel {
  color: #a5abb6;
  border-color: #f36924;
  text-color-internal: #FFFFFF;
  defaultCaption: "<id>";
  caption: "{name}";
}
node.ActC {
  defaultCaption: "<id>";
  caption: "{abbreviation}";
  color: #4C8EDA;
  border-color: #2870c2;
  text-color-internal: #FFFFFF;
}
node.ActF {
  defaultCaption: "<id>";
  color: #4C8EDA;
  border-color: #2870c2;
  text-color-internal: #FFFFFF;
  caption: "{abbreviation}";
}
node.ActS {
```

Queries

Neo4j - MATCH

```
MATCH (<var>:<abbreviation>)
RETURN <var>

MATCH (scc:SCC)

MATCH (app_scc:SCC)
or
MATCH (ecu_scc:SCC)
```

VOLVO

Queries

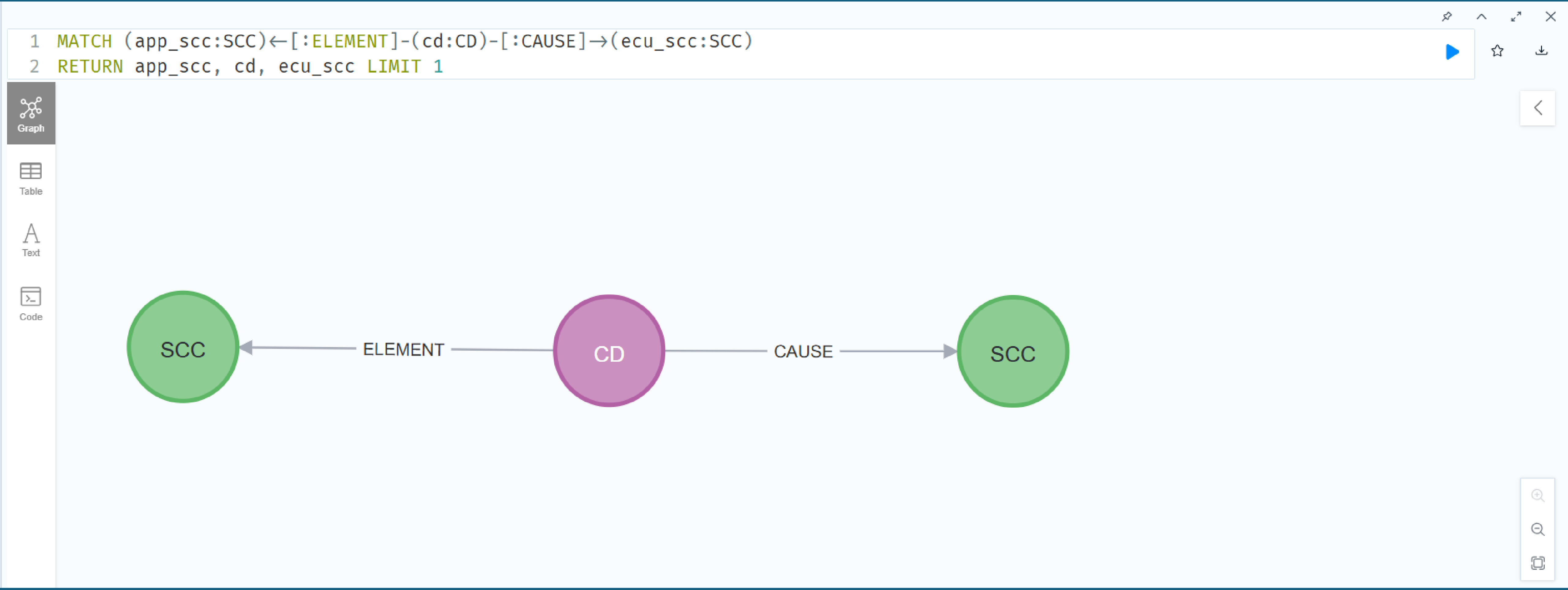
Neo4j - Links

<- [:<type>] -

```
MATCH (app_scc:SCC)<-[:ELEMENT]-(cd:CD)-[:CAUSE]->(ecu_scc:SCC)
```

VOLVO

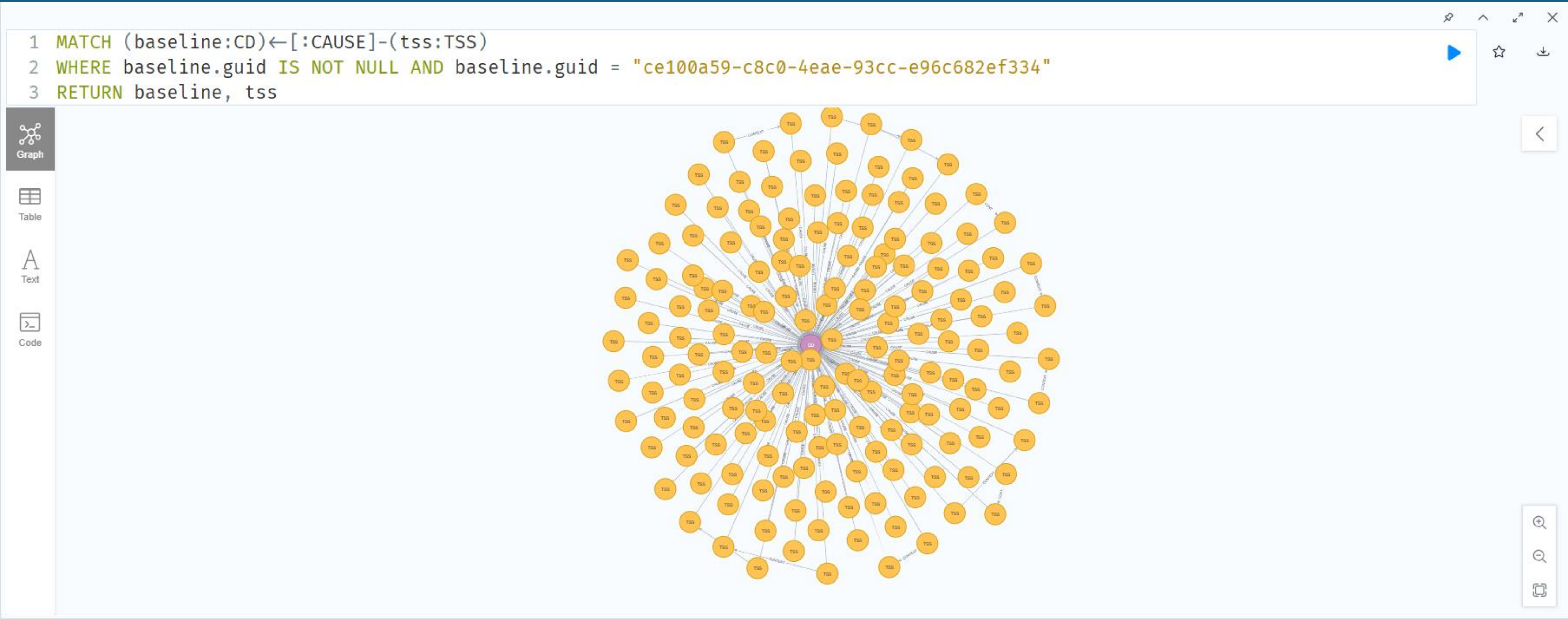
Queries



```
MATCH (app_scc:SCC)←[:ELEMENT]-(cd:CD)-[:CAUSE]→(ecu_scc:SCC)
RETURN app_scc, cd, ecu_scc LIMIT 1
```

VOLVO

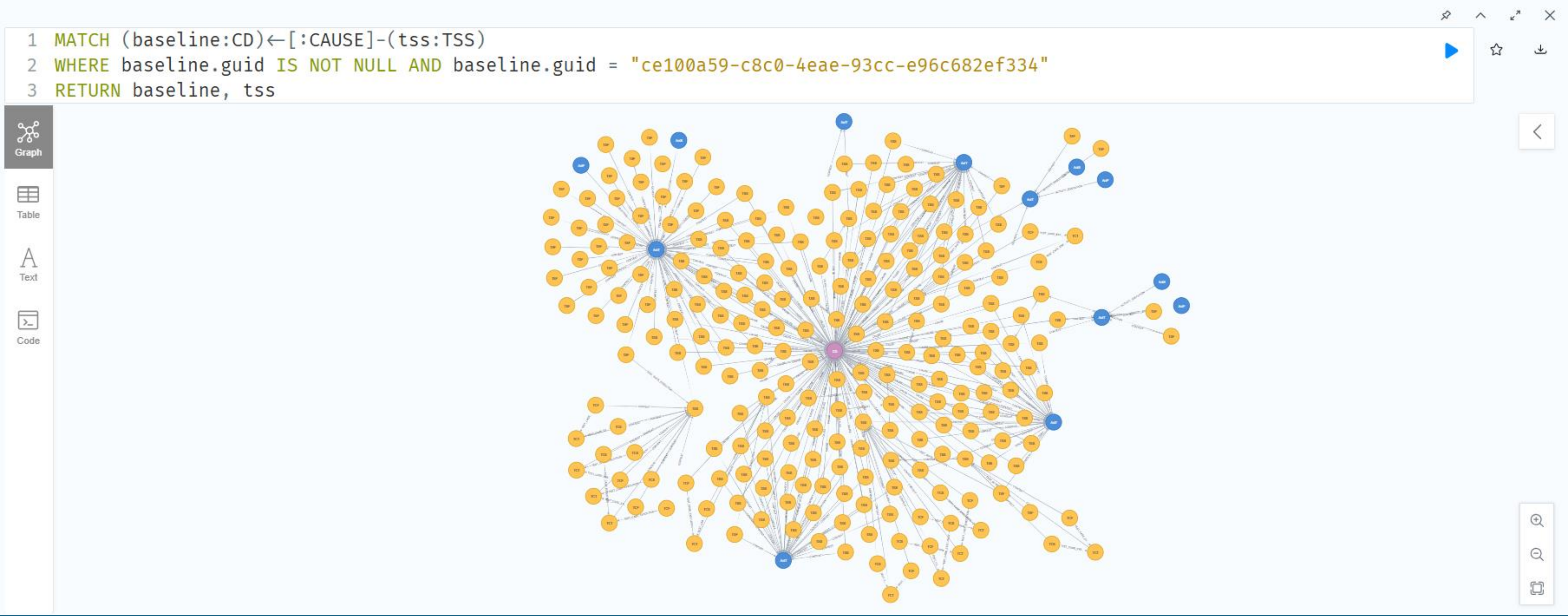
Queries



```
MATCH (baseline:CD)←[:CAUSE]-(tss:TSS)
WHERE baseline.guid IS NOT NULL AND baseline.guid = "ce100a59-c8c0-4eae-93cc-e96c682ef334"
RETURN baseline, tss
```

VOLVO

Queries



```
MATCH (baseline:CD)←[:CAUSE]-(tss:TSS)
WHERE baseline.guid IS NOT NULL AND baseline.guid = "ce100a59-c8c0-4eae-93cc-e96c682ef334"
RETURN baseline, tss
```

VOLVO

Q&A