MODULARITY AND DOMAIN DRIVEN DESIGN

a killer combination?

Tom De Wolf
Architect
tom.dewolf@aca-it.be
@tomdw

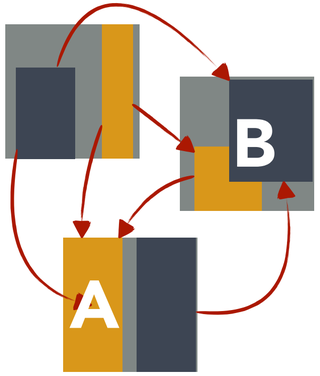
www.aca-it.be

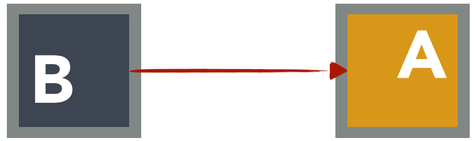
Stijn Van den Enden
CTO
stijn.vandenenden@aca-it.be
@stieno

Software Design

Separation of Concerns

Low coupling ↙ ↘ High Cohesion





- Change A impacts all modules = costly
- Change B requires split of module = costly

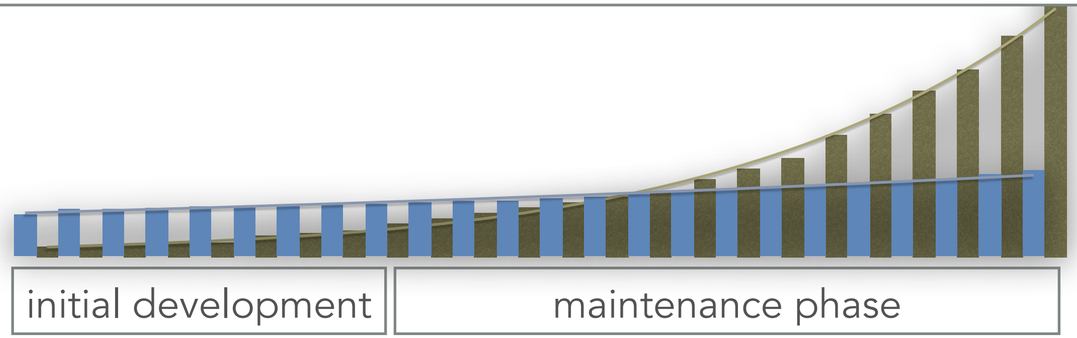
- Change A only impacts other module if api change
- Change B limited to module

Encapsulate Source of Change

Customer Satisfaction

Predictable Cost of Change

Constant change ↙ ↘ Business Driven



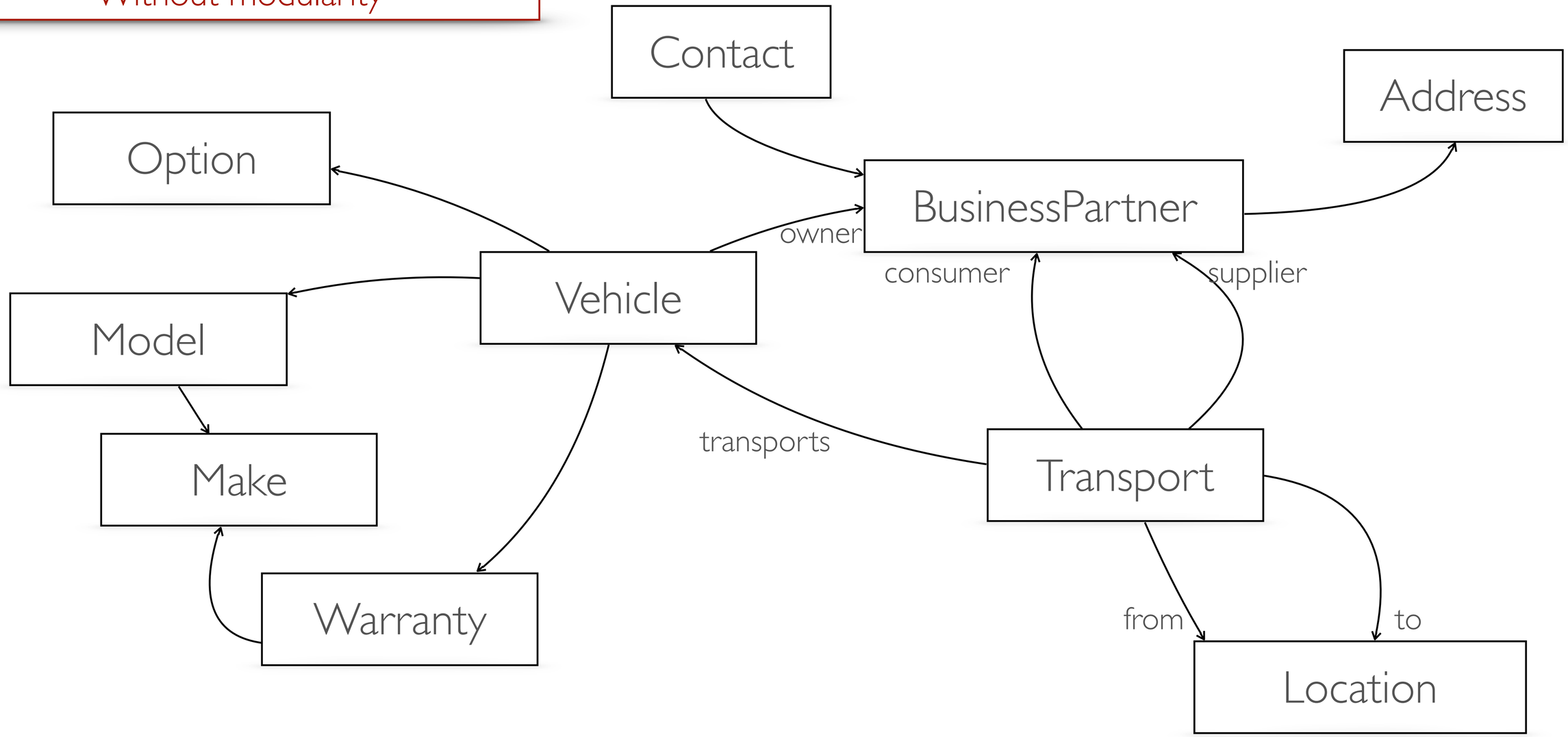
Aim for 1-on-1 mapping from business changes onto software constructs

Source of Change = Business

Functional Modularisation

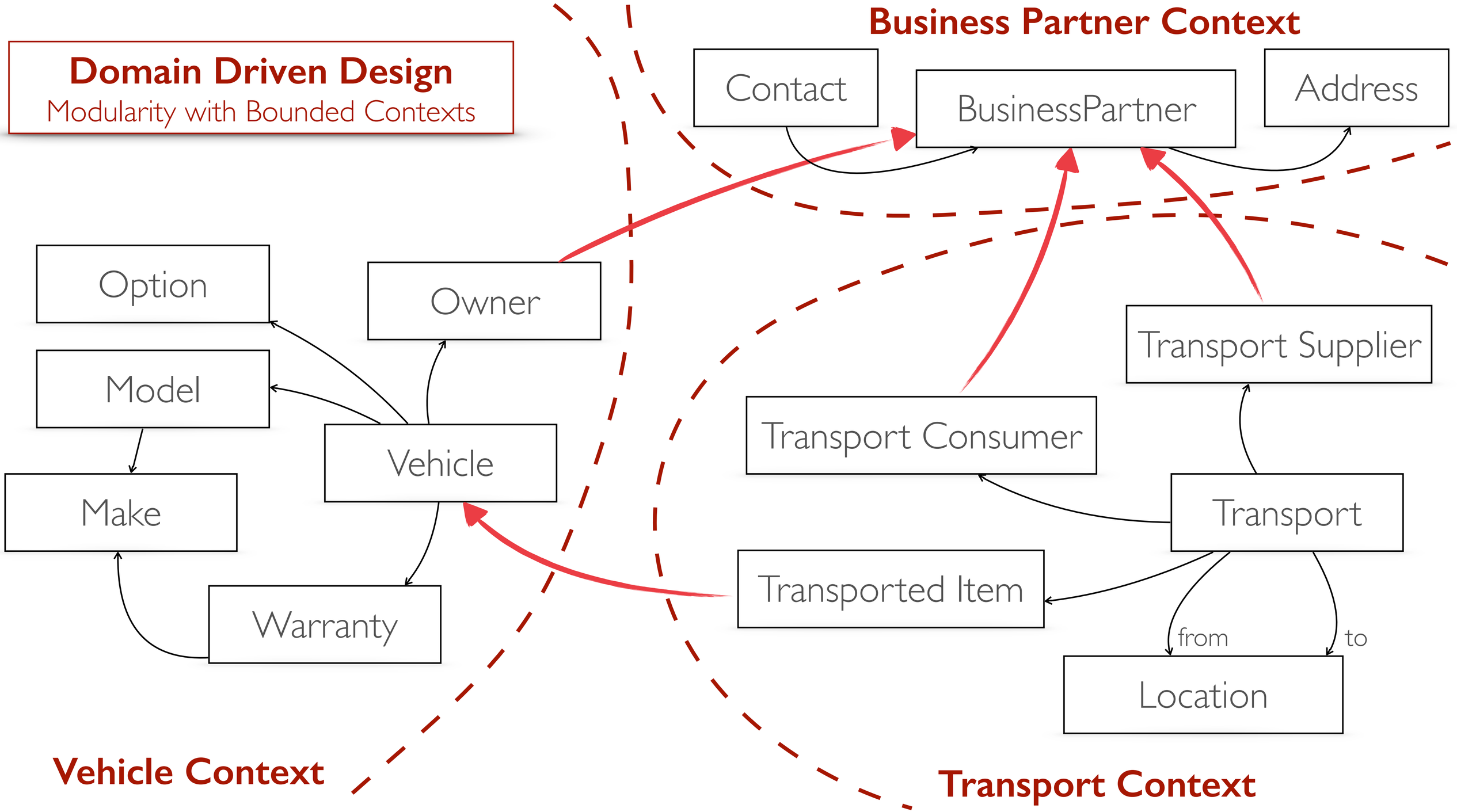
Domain Driven Design

Without modularity

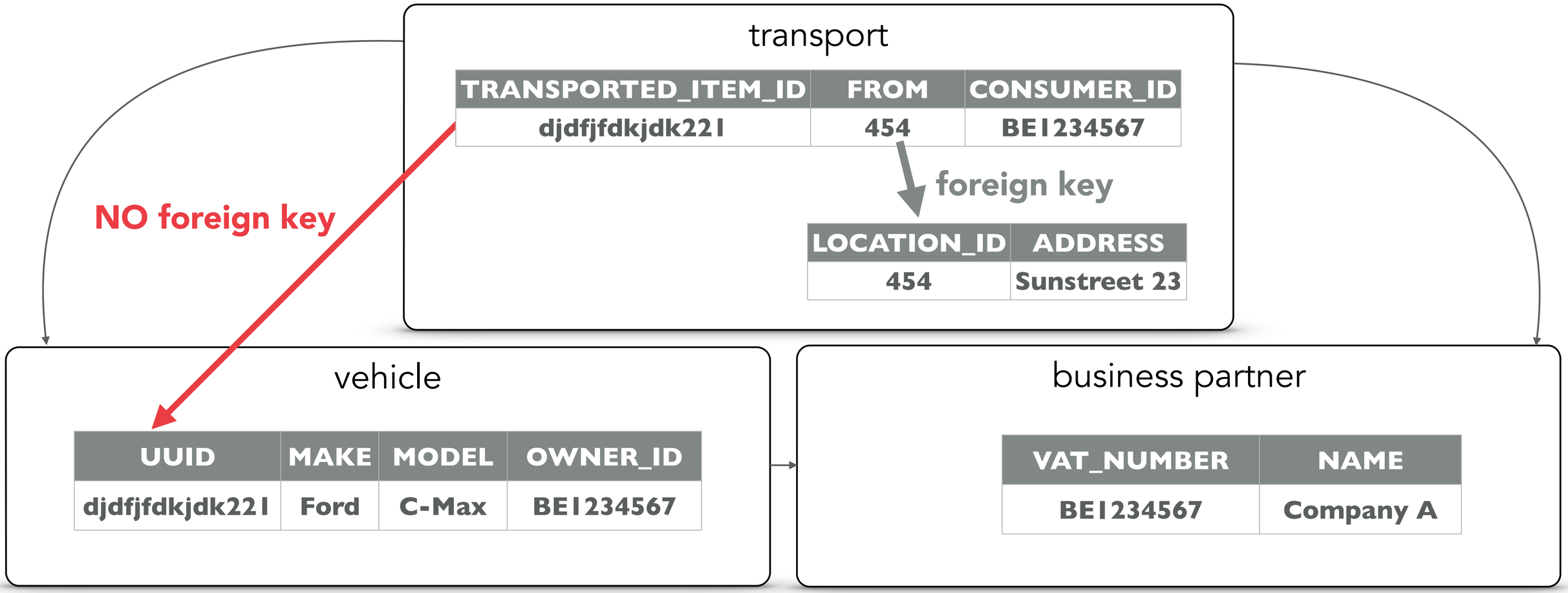


Domain Driven Design

Modularity with Bounded Contexts



MODULAR DATABASE SCHEMA



- cross domain database structures not allowed
- queries cannot cross domain boundaries

BENEFITS …

- Domain modules can **migrate independently** to next version!
- **Database schema** is **internal** to the domain bundle, i.e. **NOT part of the API**!
- Persistence and/or database **technology can differ between modules** in one system!

CHALLENGES

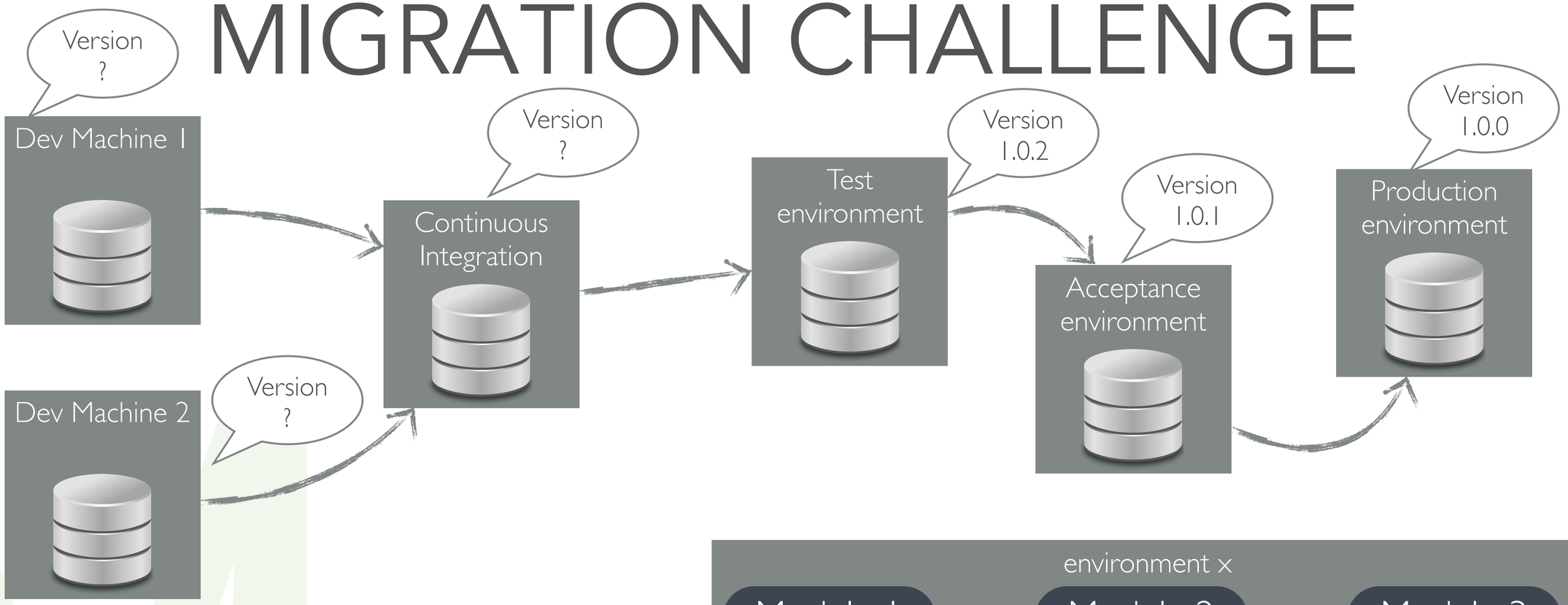
modular database migration

cross domain transactions

cross domain search and reporting

modular database migration

MIGRATION CHALLENGE



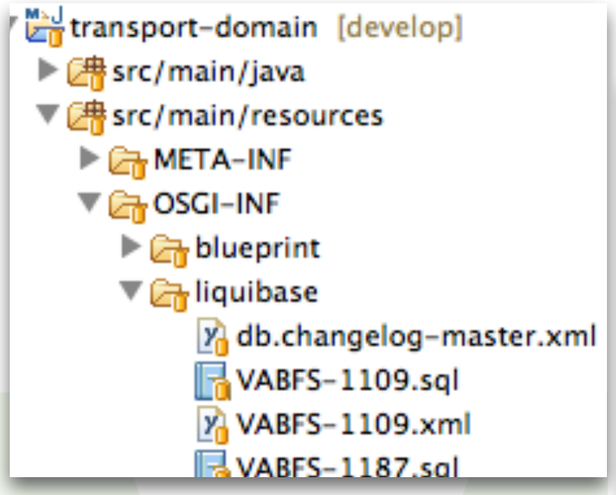
Manually track ~~which~~ scripts have run

on ~~which environment~~ for which module ?

LIQUIBASE

www.liquibase.org

In **versioned** source code



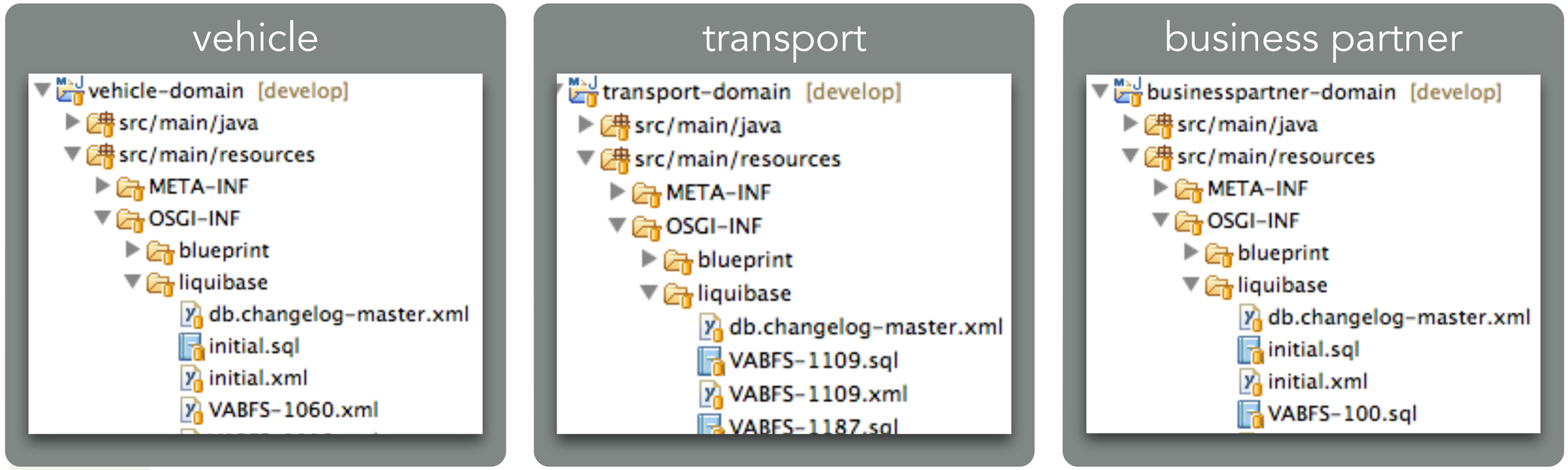
XML based **DSL** for changeSets

```xml
<databaseChangeLog xmlns="http://www.liquibase.org/xml/ns/dbchangelog" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://www.liquibase.org/xml/ns/dbchangelog http://www.liquibase.org/xml/ns/dbchangelog/dbchangelog-
    <include relativeToChangelogFile="true" file="VABFS-488.xml" />
    <changeSet id="VABFS-1109-remove-serviceitemtype-column-from-transporteditem" author="ward">
        <dropColumn tableName="TRANSPORTEDITEM" columnName="SERVICEITEMTYPE"/>
    </changeSet>
    <changeSet id="VABFS-1109-fill-new-serviceitemid-for-transporteditem" author="timv">
        <sqlFile path="VABFS-1109.sql" relativeToChangelogFile="true" stripComments="true" splitStatements="false"/>
    </changeSet>
    <changeSet id="VABFS-1109-populate-readableserviceitemid-column-for-transitem" author="ward">
        <sql>UPDATE TRANSPORTEDITEM SET READABLESERVICEITEMID = OLDSERVICEITEMIDENTIFICATION</sql>
    </changeSet>
</databaseChangeLog>
```

DATABASECHANGELOG table to **track executed changesets** for each environment

	ID	AUTHOR	FILENAME	DATEEXECU...	O...	EXECT...	MD5SUM
61	VABFS-1109-fill-new-serviceitemid-for-phot...	timv	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14...	301	EXECUTED	3:722ccac8df9c34efe4..
62	VABFS-1109-fill-new-serviceitemid-for-reta...	timv	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14...	372	EXECUTED	3:afe5368dd074c6c621..
63	VABFS-1109-fill-new-serviceitemid-for-tran...	timv	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14...	333	EXECUTED	3:fa505fb8dbe9901067..
64	VABFS-1109-notnull-readableaccounteditemid...	tomdw	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14...	316	EXECUTED	3:4bbc96b1970104b132..
65	VABFS-1109-notnull-readableserviceitemid-c...	ward	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14...	326	EXECUTED	3:21ad4a72ca57d057f1..
66	VABFS-1109-notnull-readableserviceitemid-c...	tomdw	/OSGI-INF/liquibase/VABFS-1109.xml	30-JAN-14...	479	EXECUTED	3:a893a786fe14c218ab..
67	VABFS-1109-notnull-readableserviceitemid-c...	timv	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14...	305	EXECUTED	3:ff1e7eca7fbcc88bc8..
68	VABFS-1109-notnull-readableserviceitemid-c...	ward	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14...	376	EXECUTED	3:f0b4aef8654da8e1f4..
69	VABFS-1109-notnull-readableserviceitemid-c...	ward	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14...	337	EXECUTED	3:a1e02fe4203c8151de..
70	VABFS-1109-populate-readableaccounteditemi...	tomdw	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14...	315	EXECUTED	3:afeeec830a9ddf1231..
71	VABFS-1109-populate-readableserviceitemid-...	ward	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14...	325	EXECUTED	3:ed8e873a37d07f1cfa..
72	VABFS-1109-populate-readableserviceitemid-...	tomdw	/OSGI-INF/liquibase/VABFS-1109.xml	30-JAN-14...	478	EXECUTED	3:06c8b2fdd3bcbc94b9..
73	VABFS-1109-populate-readableserviceitemid-	timv	/OSGI-INF/liquibase/VABFS-1109.xml	10-JAN-14	304	EXECUTED	3:8e0f3e3105103b403

MODULAR LIQUIBASE

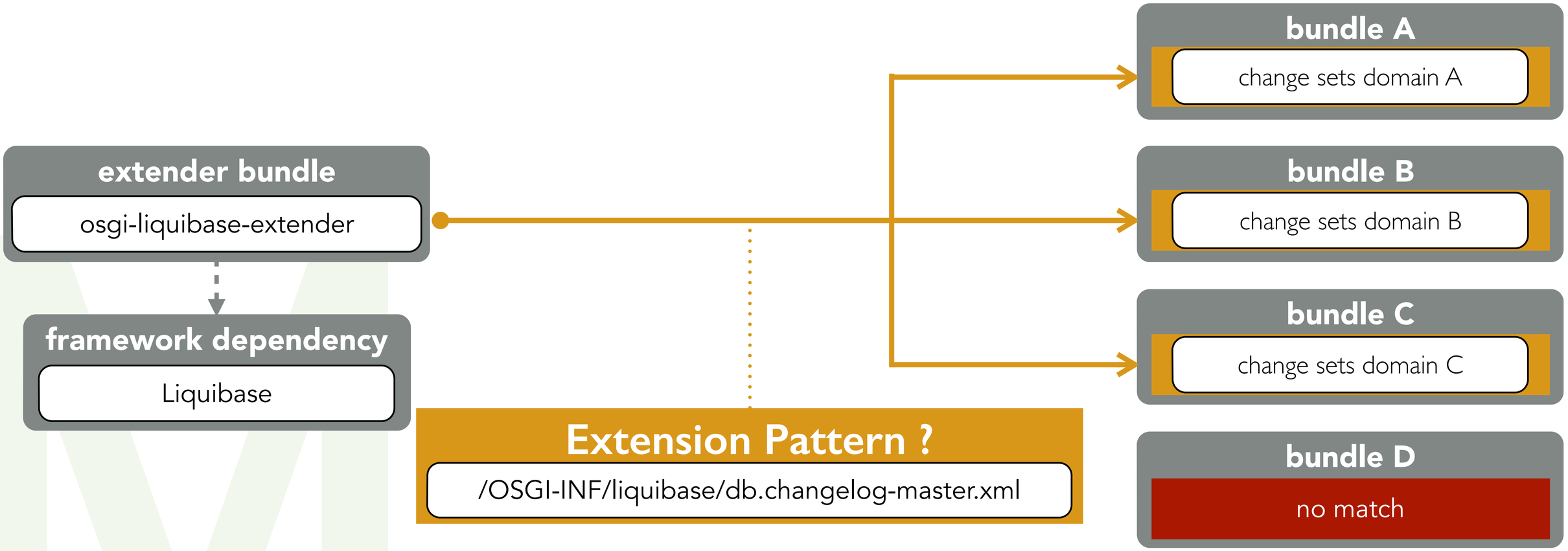


deployment 1	migrate to initial version	migrate to initial version	migrate to initial version
deployment 2	no db changes, no migration	migrate to version 2	migrate to version 2
deployment 3		migrate to version 3	

Migrate modules separately

Migrate @deploy of module

THE EXTENDER PATTERN



- Only extender depends on Liquibase framework
- New Liquibase process for each matching bundle
- Update of single bundle triggers Liquibase update

cross domain transactions

JTA OR NOT?

1 transaction spanning multiple modular domains

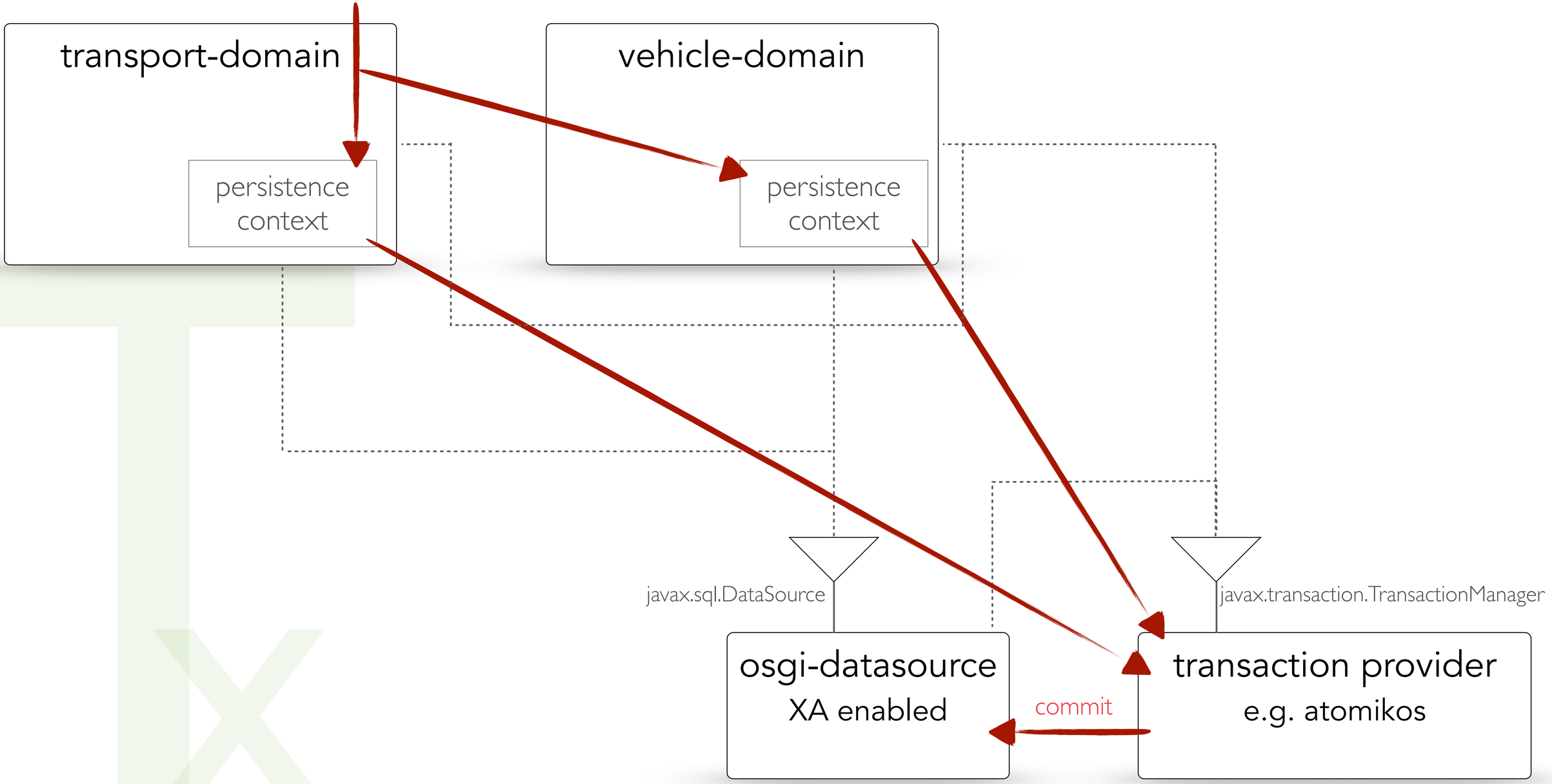
JTA not needed

- No persistence layer — direct jdbc/sql access — 1 datasource

JTA required

- Multiple datasources
- Different types of persistence layers (JPA, NoSQL, …)
- JPA persistence layer in each domain bundle
  - instead of one big persistence unit
  - even on 1 datasource

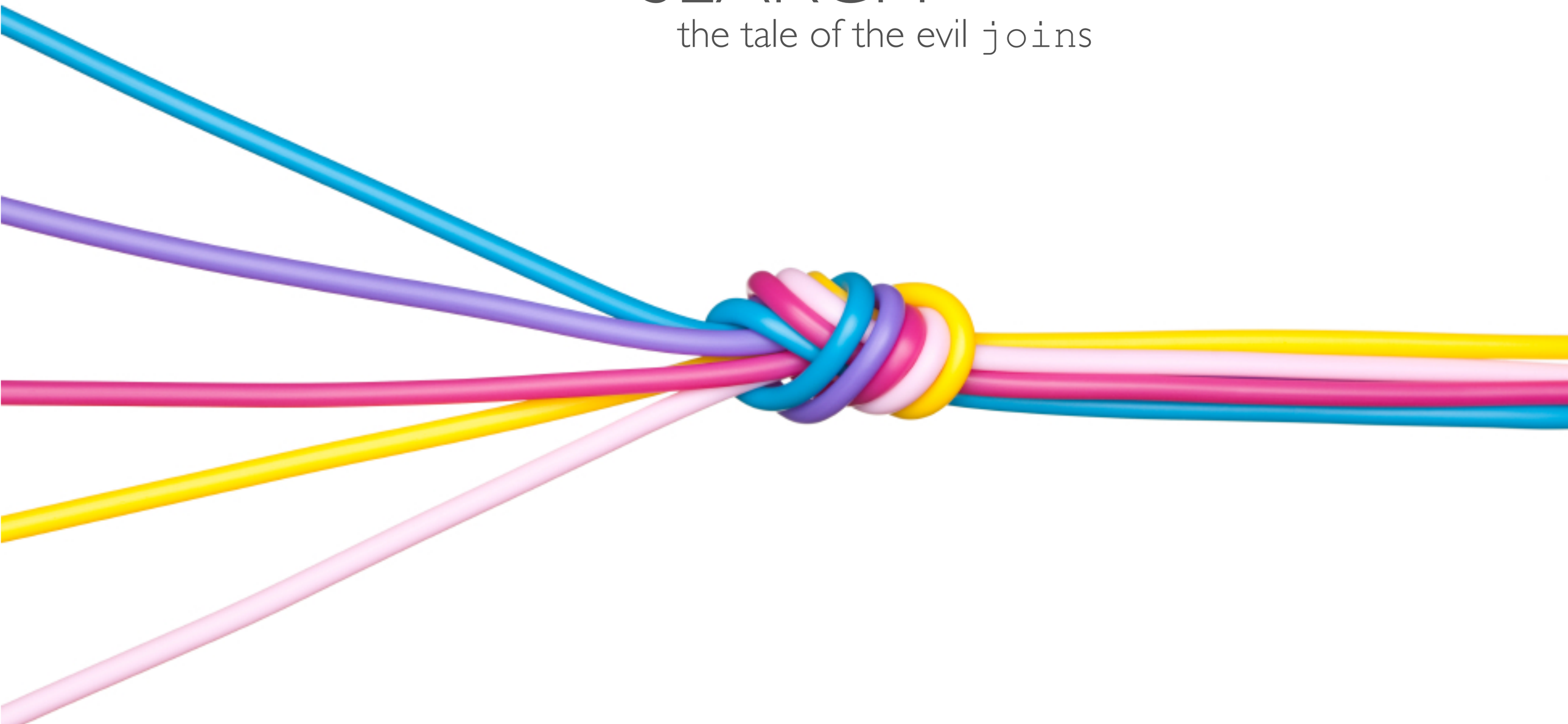
MODULAR PERSISTENCE

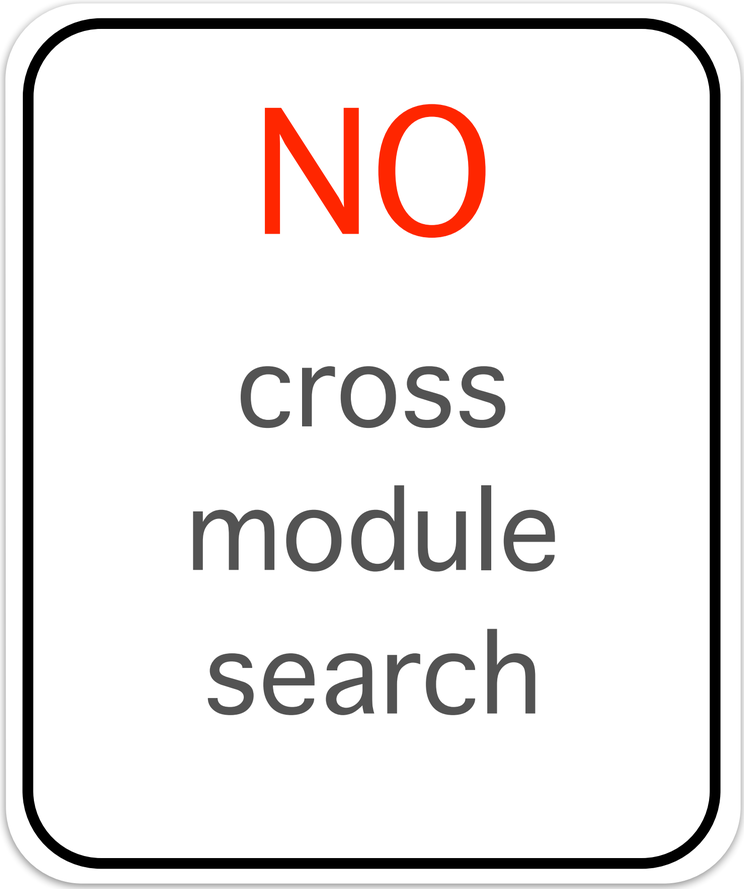


cross domain search

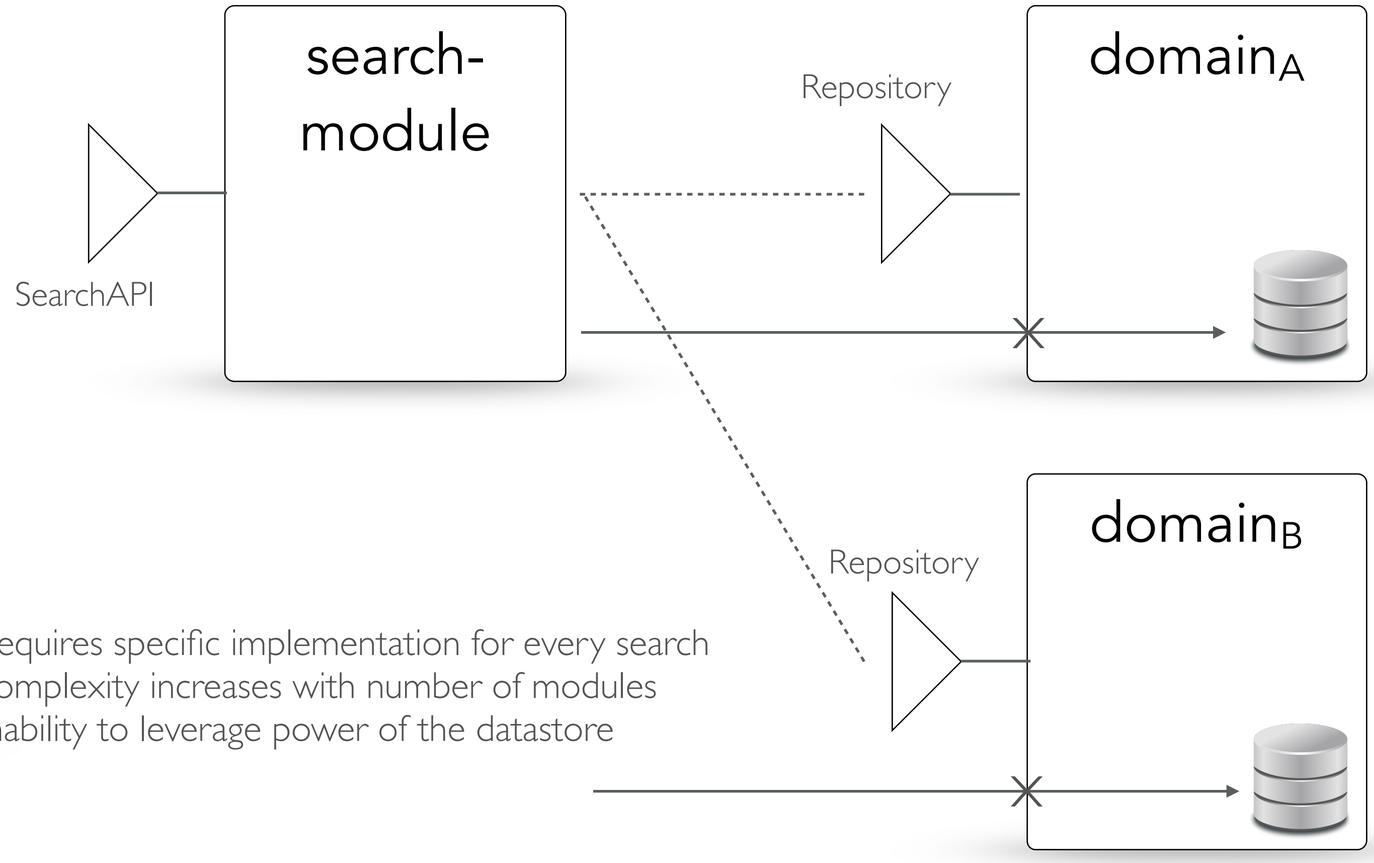
SEARCH

the tale of the evil `joins`




NO
cross
module
search

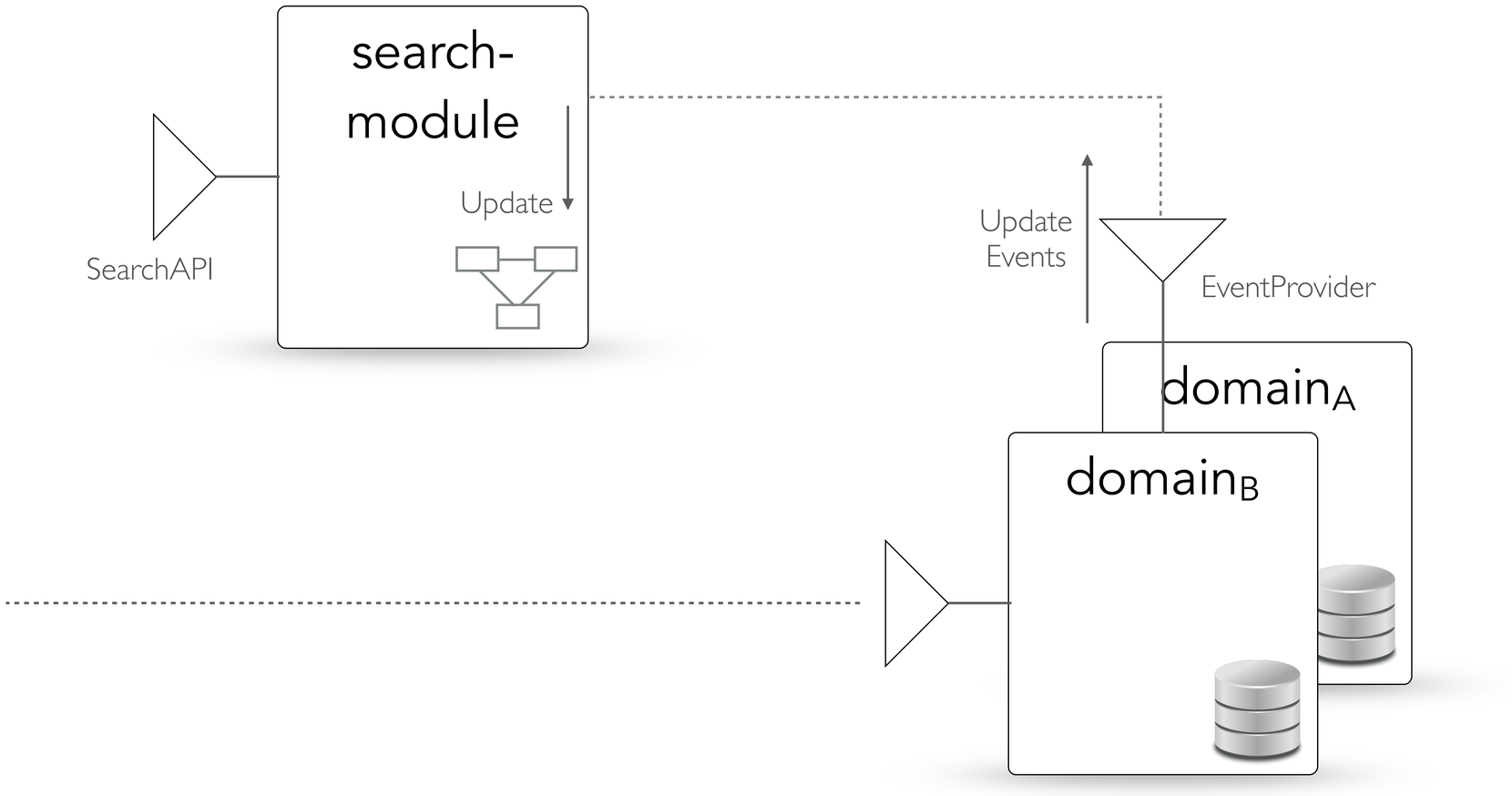
A. Search encapsulated in a separate module leveraging the functionality of other domain modules



Stop

- requires specific implementation for every search
- complexity increases with number of modules
- inability to leverage power of the datastore

B. Search is using a separate query model




elasticsearch

Index a document

index

document_type

```
$ curl -XPUT ‘localhost:9200/vehicle/external/1’ -d ‘{
  “make”: “BMW”,
  “VIN”: “30203232012102”
}’
```

id

Update a document

index document_type

```
$ curl -XPOST ‘localhost:9200/vehicle/external/1/
_update’ -d ‘{
  “doc”: {“make”: “Alpina”}
}’
```

id

Querying an index

index

```
$ curl -XPOST localhost:9200/vehicle/_search?pretty -d
‘{"query":{"match":{"make":"BMW"}}}'
```

Querying an index

elasticsearch

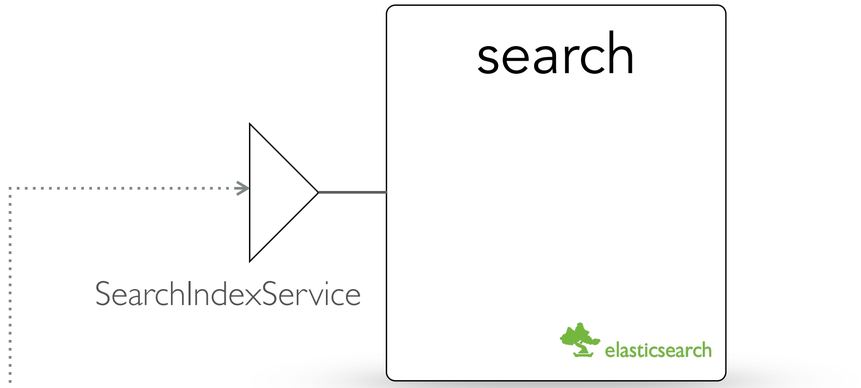
```
$ curl -XPOST localhost:9200/vehicle/_search?pretty -d
‘{"query":{"match":{"mark":"BMW"}}}'
{
  "took" : 3,
  "timed_out" : false,
  "_shards" : {
    "total" : 5,
    "successful" : 5,
    "failed" : 0
  },
  "hits" : {
    "total" : 2,
    "max_score" : 0.30685282,
    "hits" : [ {
      "_index" : "vehicle",
      "_type" : "external",
      "_id" : "1",
      "_score" : 0.30685282, "_source" : {"make": "BMW", "VIN": "30203232012102"}
    }, {
      "_index" : "vehicle",
      "_type" : "external",
      "_id" : "2",
      "_score" : 0.30685282, "_source" : {"makes": "BMW", "VIN": "30203232012112"}
    } ]}}
```

elasticsearch

- Distributed (shards/replicas)
- Advanced Quering (lucene/geo/aggregation)
- Facets (terms/ranges/histogram)
- API (HTTP/Java/Java Testing)
- …

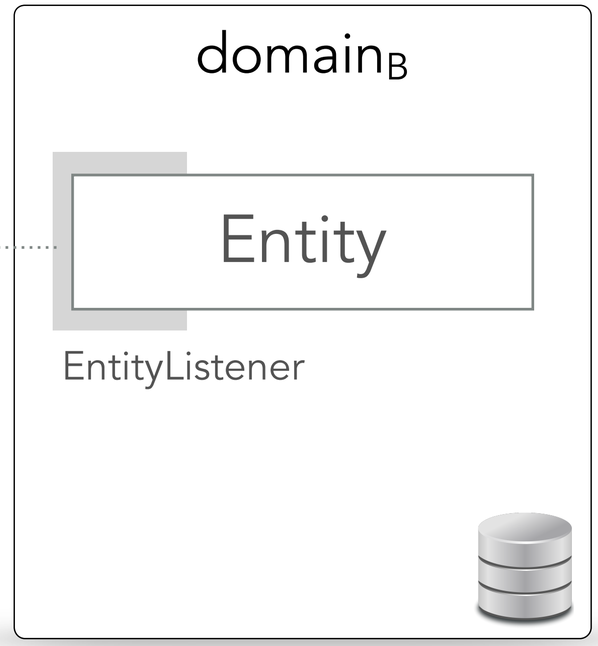
1

Populate the index



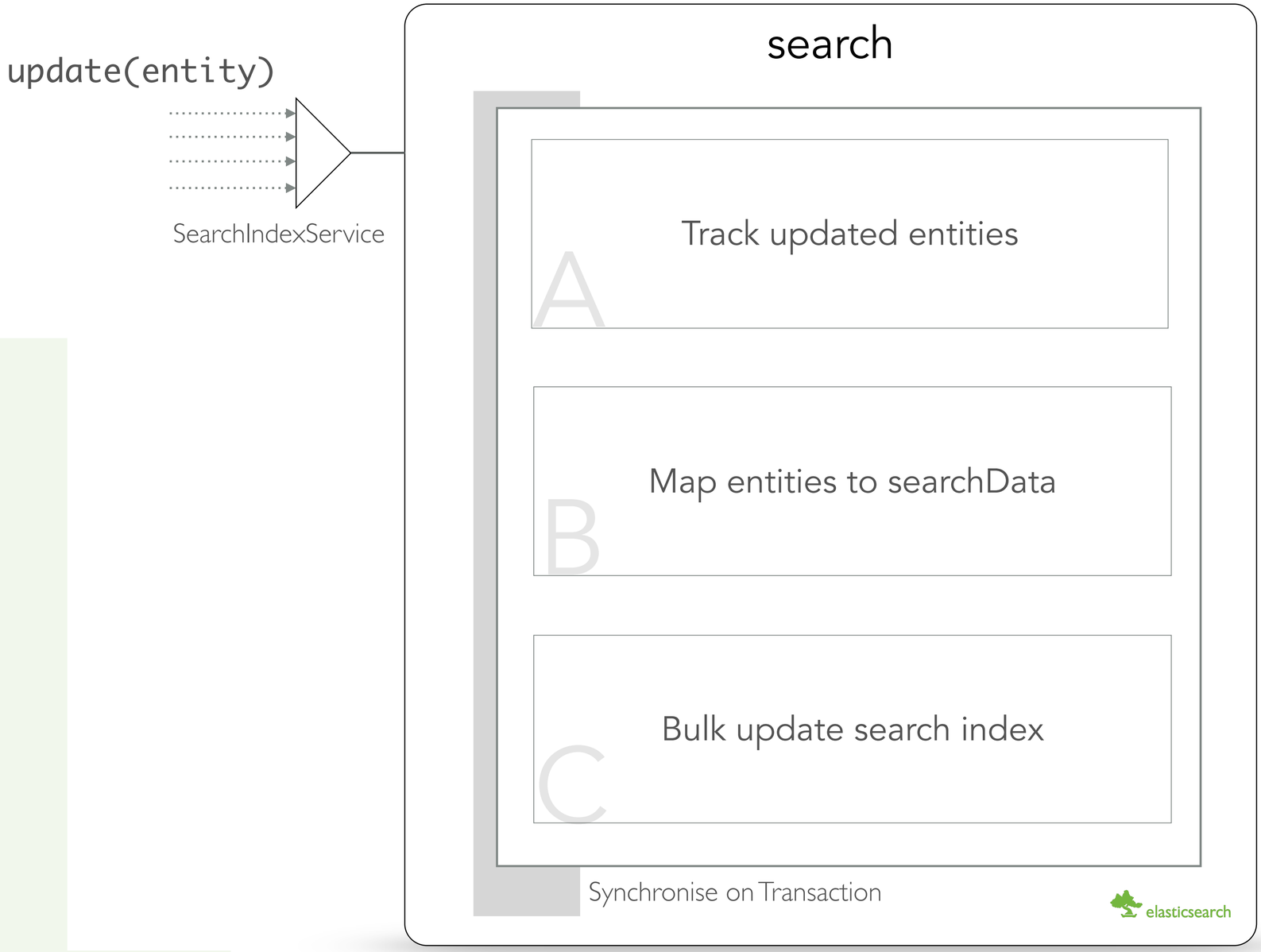

```
package be.vabfs.search.api;

public interface SearchIndexService {

   void update(Object entityToUpdate);

   void delete(Object entityToRemove);

   void populate();

   void clear();

}
```

@PrePersist
@PostUpdate
@PostRemove



1

Populate the index

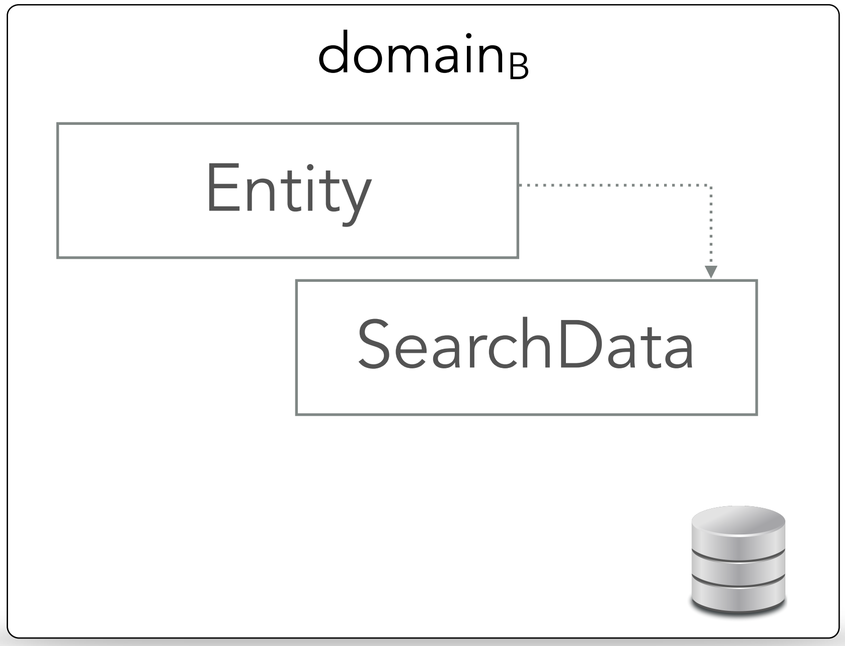


1

Populate the index

B

Map entities to searchData



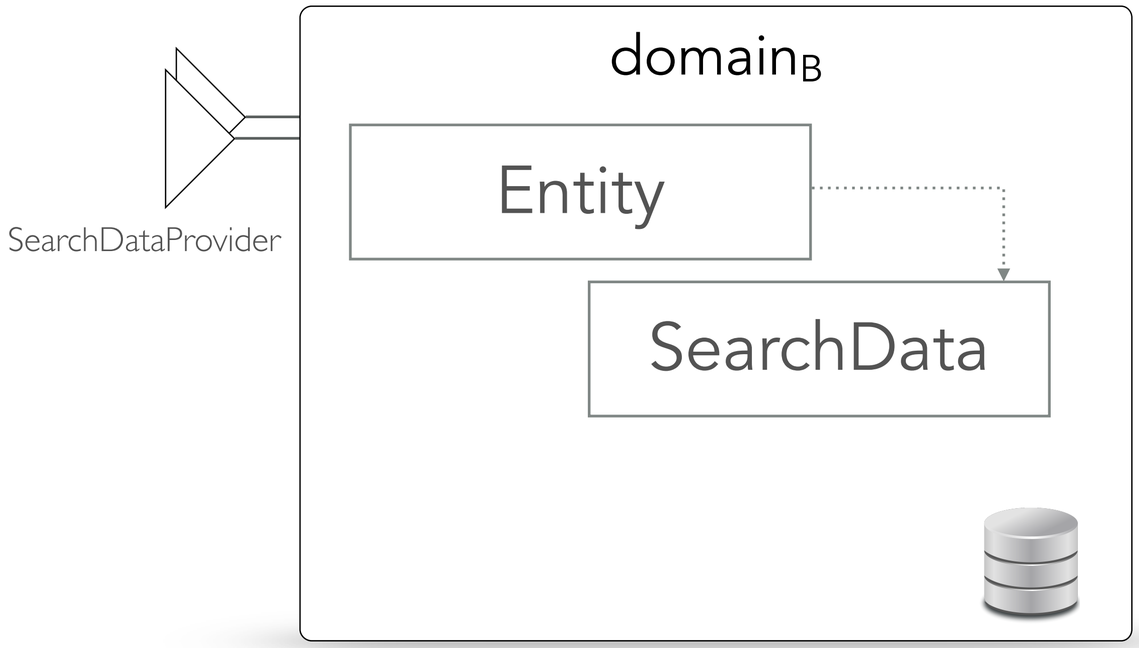

```
package be.vabfs.search.api.data;

import java.util.List;

public interface SearchDataProvider {

   Class<? extends Object> getEntityClass();

   List<SearchData> index(Object entity);

}
```

1

Populate the index

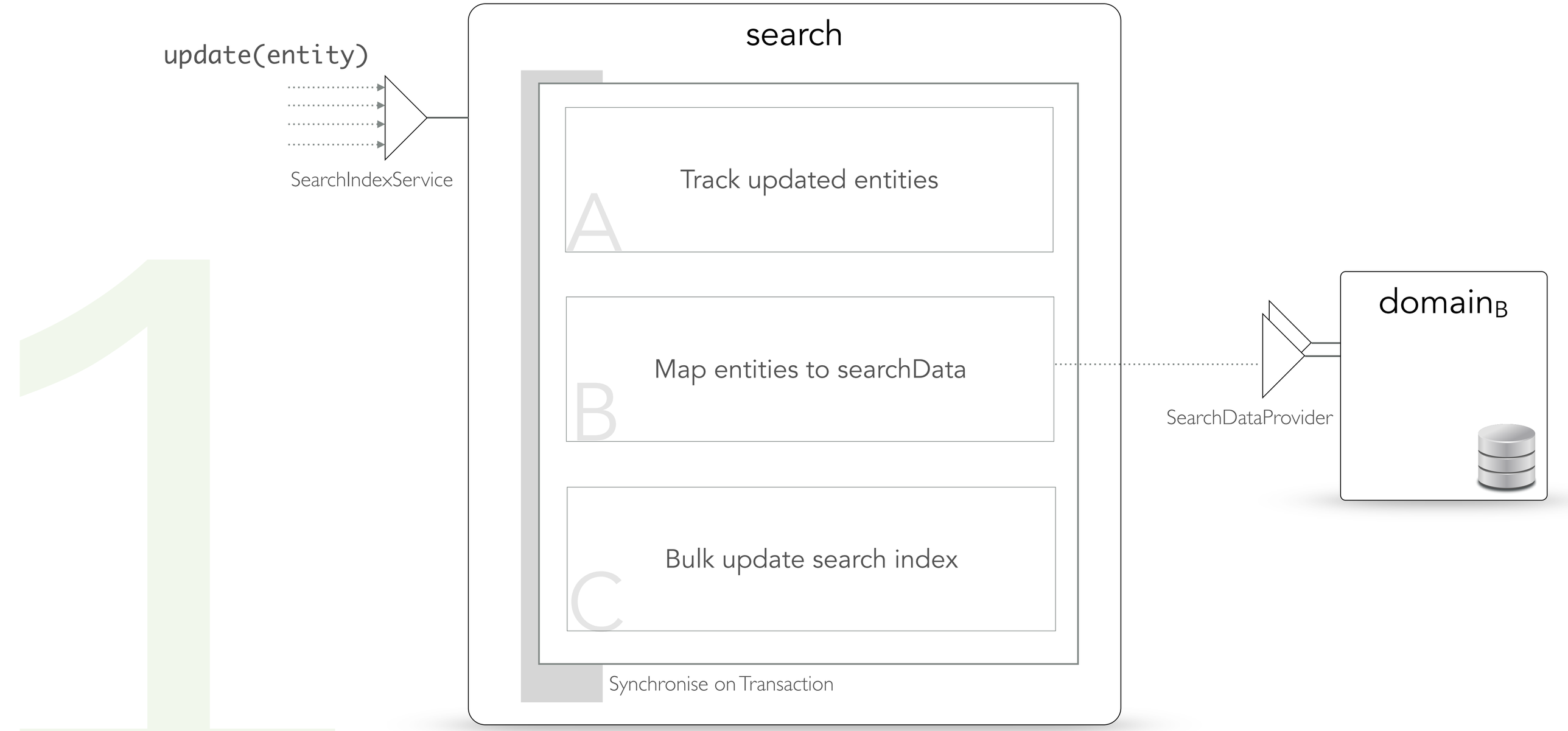
B

Map entities to searchData



1

Populate the index



1

Populate the index

2

Enhancing Search

automotive services

Vehicles | Sales | Operations | Accounting | Admin

Bart Mets
Garage De dijle

Overview | Archive

English

Vehicle description overview

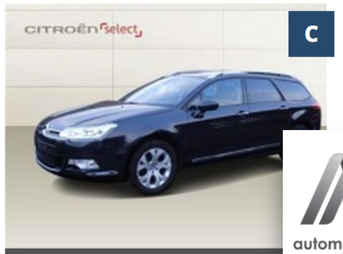
+ Add

All services

Identification: | Your reference: | Make: | Model:

Alfa Romeo

Audi A2
5454 km 2011



BMW 3
156564 km

Aro 24
255656 km 2011

Hyundai Getz
545646 km

Renault Laguna Break
6000 km 2010

Mercedes-Benz B
5000 km

automotive services

Vehicles | Sales | Operations | Accounting | Admin

Bart Mets
Garage De dijle

Retail | Community publications | Auction | Virtual showroom

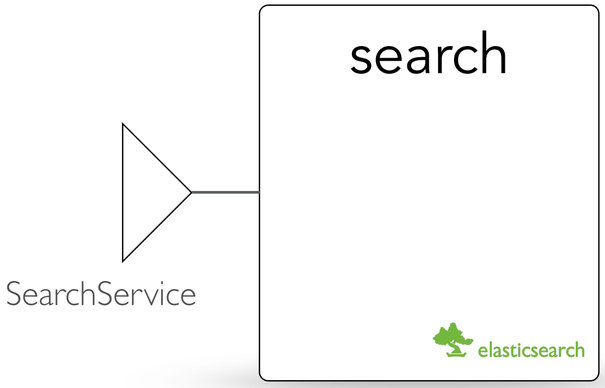
English

Retail overview

Retail channel: | Price inclusive VAT: | Make: | Model:

- [x] Retail channel
- [ ] Price exclusive VAT
- [x] Price inclusive VAT
- [ ] VAT deductable
- [ ] State
- [ ] State date
- [ ] Identification
- [ ] VIN
- [ ] License plate
- [ ] Your reference
- [x] Make

Retail channel	Identification	Description	Price inclusive VAT	State	
autolive	AZERWSDEZE9988899	Audi A6 2.0 TDI, 88000 km, 6/2010, Diesel, Anthracite	29000,00 €	Published	28
autolive	00000000000000001	Alfa Romeo 145 , 10000 km, 1/2009, Diesel, Green	5000,00 €	Published	28
AUTO SCOUT 24	00000000000000001	Alfa Romeo 145 , 10000 km, 1/2009, Diesel, Green	5000,00 €	Published	28
AUTO SCOUT 24	45879ZEDSFG659654	Hyundai i40 CW 1.9 cdi, 25600 km, 7/2011, Petrol, Grey	6000,00 €	Published	28
VROOM.be	45879ZEDSFG659654	Hyundai i40 CW 1.9 cdi, 25600 km, 7/2011, Petrol, Grey	6000,00 €	Published	28
AUTO SCOUT 24	AZERWSDEZE9988899	Audi A6 2.0 TDI, 88000 km, 6/2010, Diesel, Anthracite	29000,00 €	Unpublished	27/05/2014




```
package be.vabfs.search.api;

import …

public interface SearchService {

…

  SearchResults query(String query, int start, int rows,
                      String sort, String sortOrder,
                      String documentType);

   List<String> options(String field,
                        String... documentTypes);

   List<SearchCriterion> availableCriteria(
                        String... criteriaCategories);

}
```

2

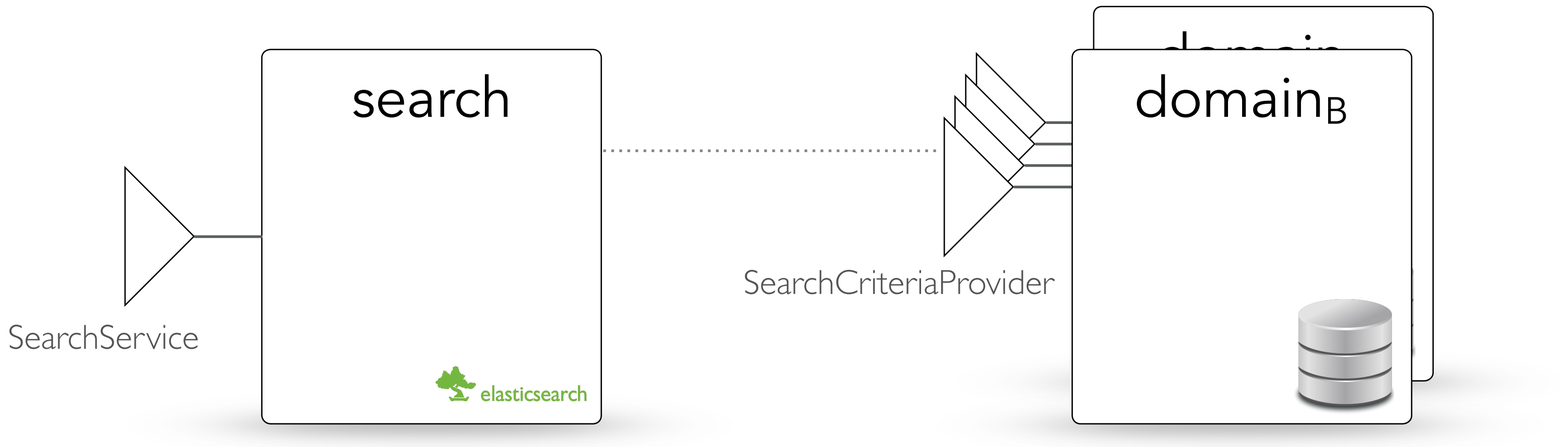
Enhancing Search

```
package be.vabfs.search.api.criteria;

import java.util.List;

public interface SearchCriteriaProvider {

   String getProviderCategory();

   List<SearchCriterion> getAvailableCriteria();
}
```

```
"vehicle.mileage"
"mileage"
"km"
SearchCriterionType.NUMBER
SearchCriterionQueryType.RANGE_NUMBER
"SERVICE_ITEM"
```

```
package be.vabfs.search.api.criteria;

public interface SearchCriterion {

   String getName();
   String getKey();
   String getUnitName();
   SearchCriterionType getType();
   SearchCriterionQueryType getQueryType();
   String getDocumentType();

}
```

2



Enhancing Search

LESSONS LEARNED

Migration

@deployment time migration = **RISK**

- have CI build continuously migrate current production data

Refactor wrong modularisation requires **unwanted migration dependencies**

- reset/flatten migration change sets to restore modularisation

Cross domain search and reporting

Effort to integrate search is limited

- due to dynamic osgi service model

Advanced search **functionality** is possible

WHAT IS NEXT?

Migration

Multiple module **versions** active? Multiple schema versions active?

- e.g. feature try-out to limited customer base

Downgrade to older module version?

- Previous schema with new data?

Hot deploy while users are firing requests?

- Migration still busy = failure

Cross domain search and reporting

Domain **S**pecific Query **L**anguage

Exploiting elastic search capabilities **beyond search**, e.g. reporting

Modular migration

Liquibase extender @ Deploy

Cross domain transactions

Distributed JTA transactions

Cross domain search

Elasticsearch view

Functionally

not limited by domain module boundaries to answer business questions

Non-functionally

future-proof platform with impact of change contained in loosely coupled domain modules

Tom De Wolf

Architect

tom.dewolf@aca-it.be

@tomdw

www.aca-it.be

Stijn Van den Enden

CTO

stijn.vandenenden@aca-it.be

@stieno