SQL
KONFERENZ
2025

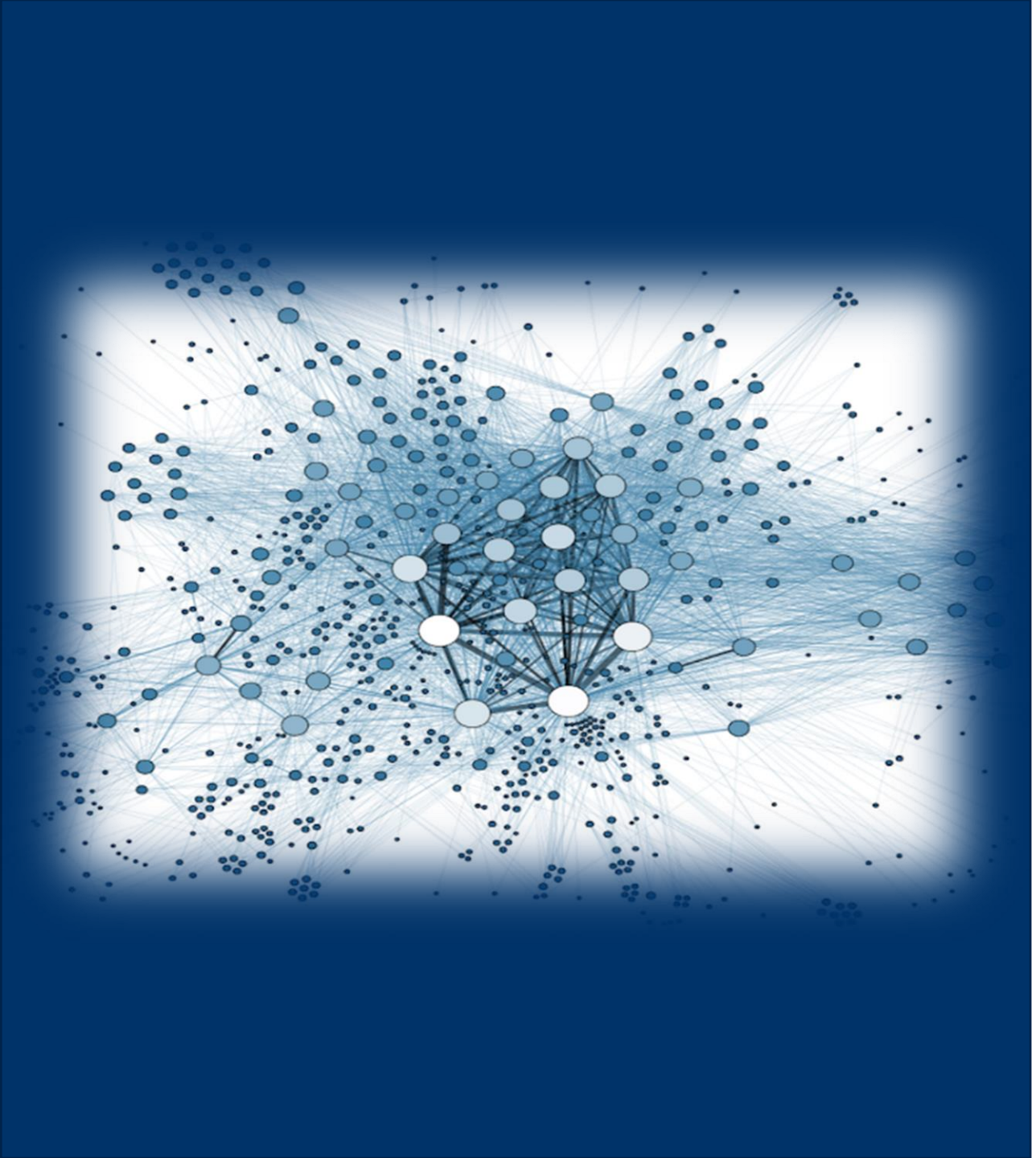




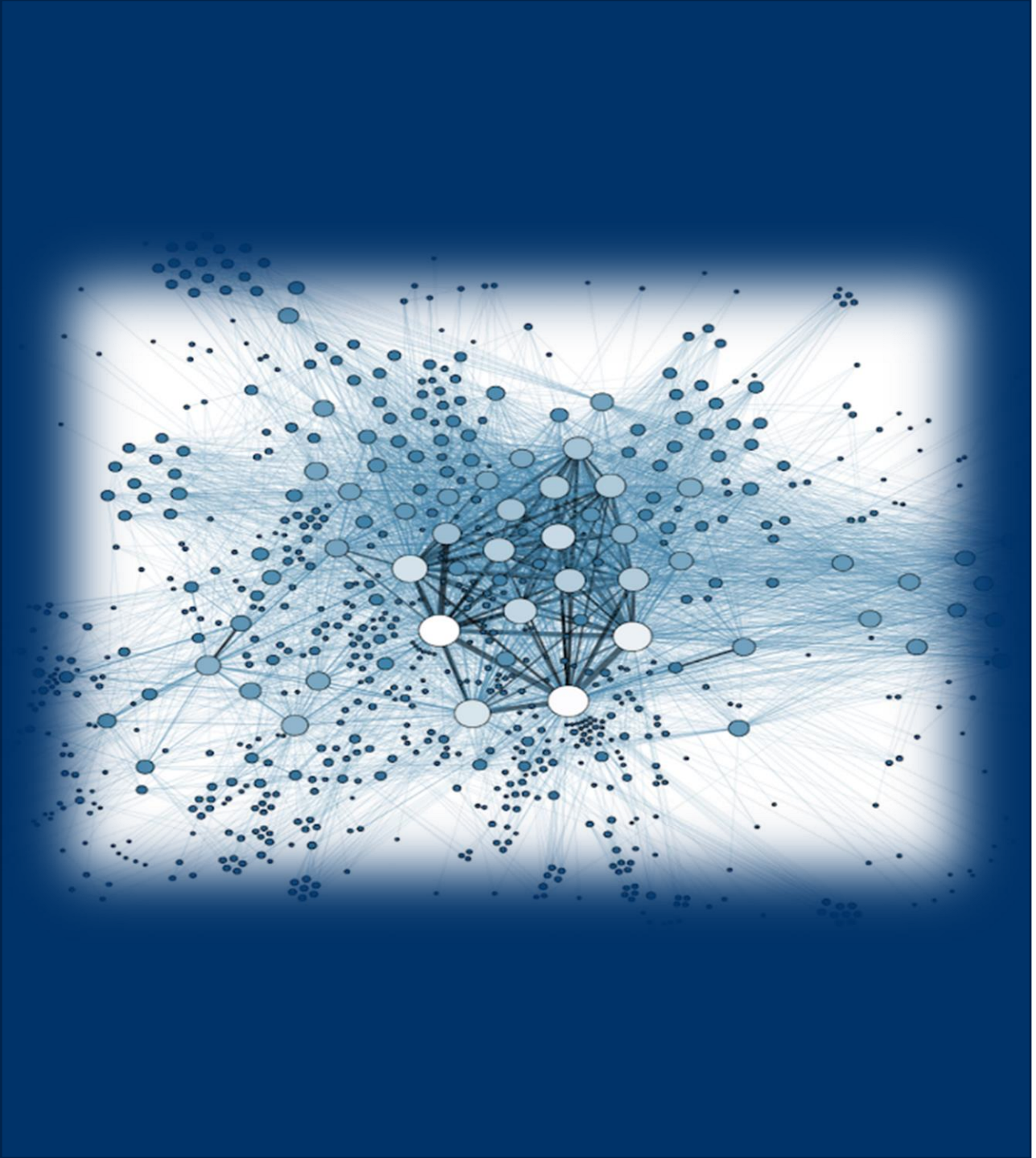
Knowledge Graphs

A Crash Course

Christoph Seck | chrseck@gmail.com | Christoph.seck@kuehne-nagel.com

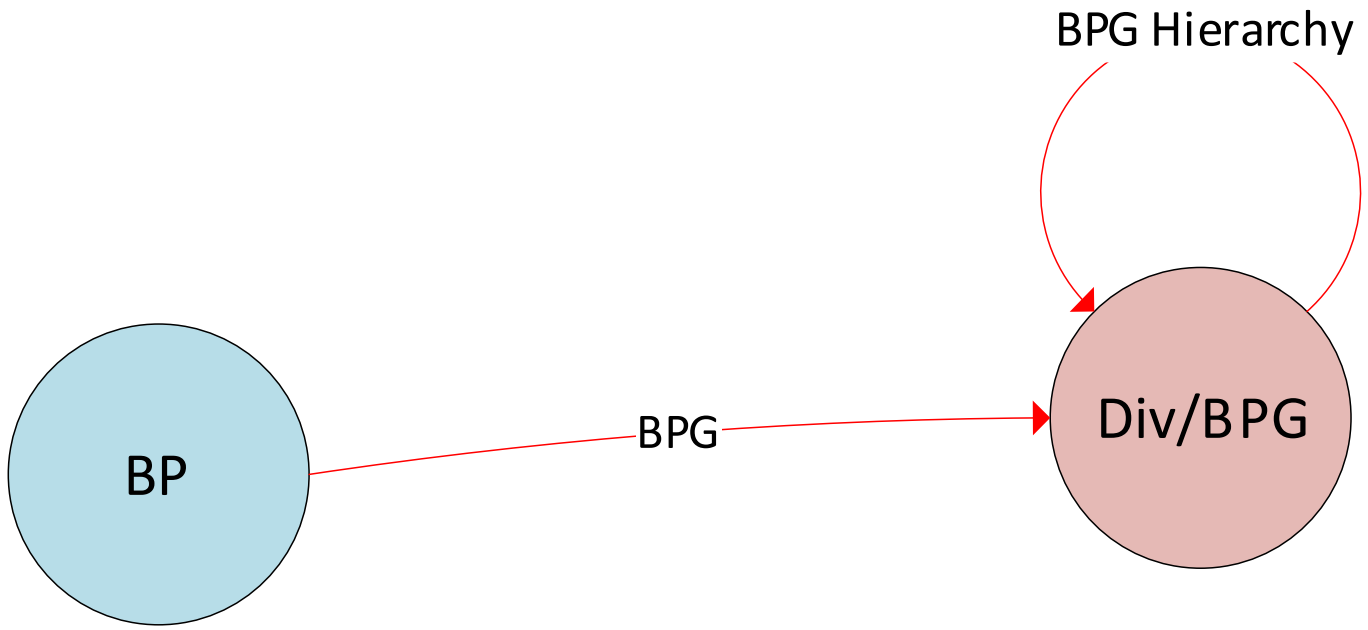


1. Intro
2. Graphs
3. Knowledge Graphs
4. Implementation: RDF & SPARQL
5. Structure with RDFS & OWL
6. The closed world of SHACL
7. Wrap Up

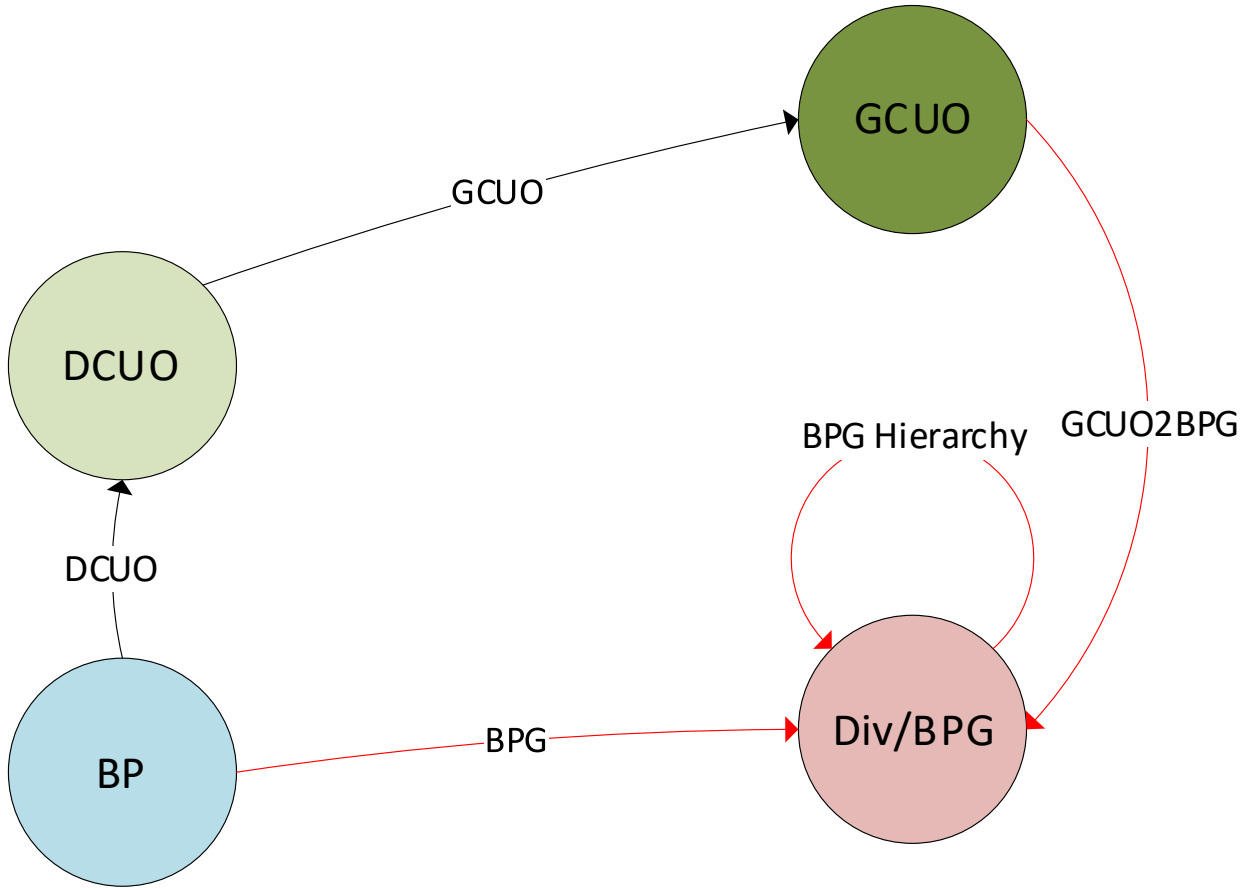


1. **Intro**
2. Graphs
3. Knowledge Graphs
4. Implementation: RDF & SPARQL
5. Structure with RDFS & OWL
6. The closed world of SHACL
7. Wrap Up

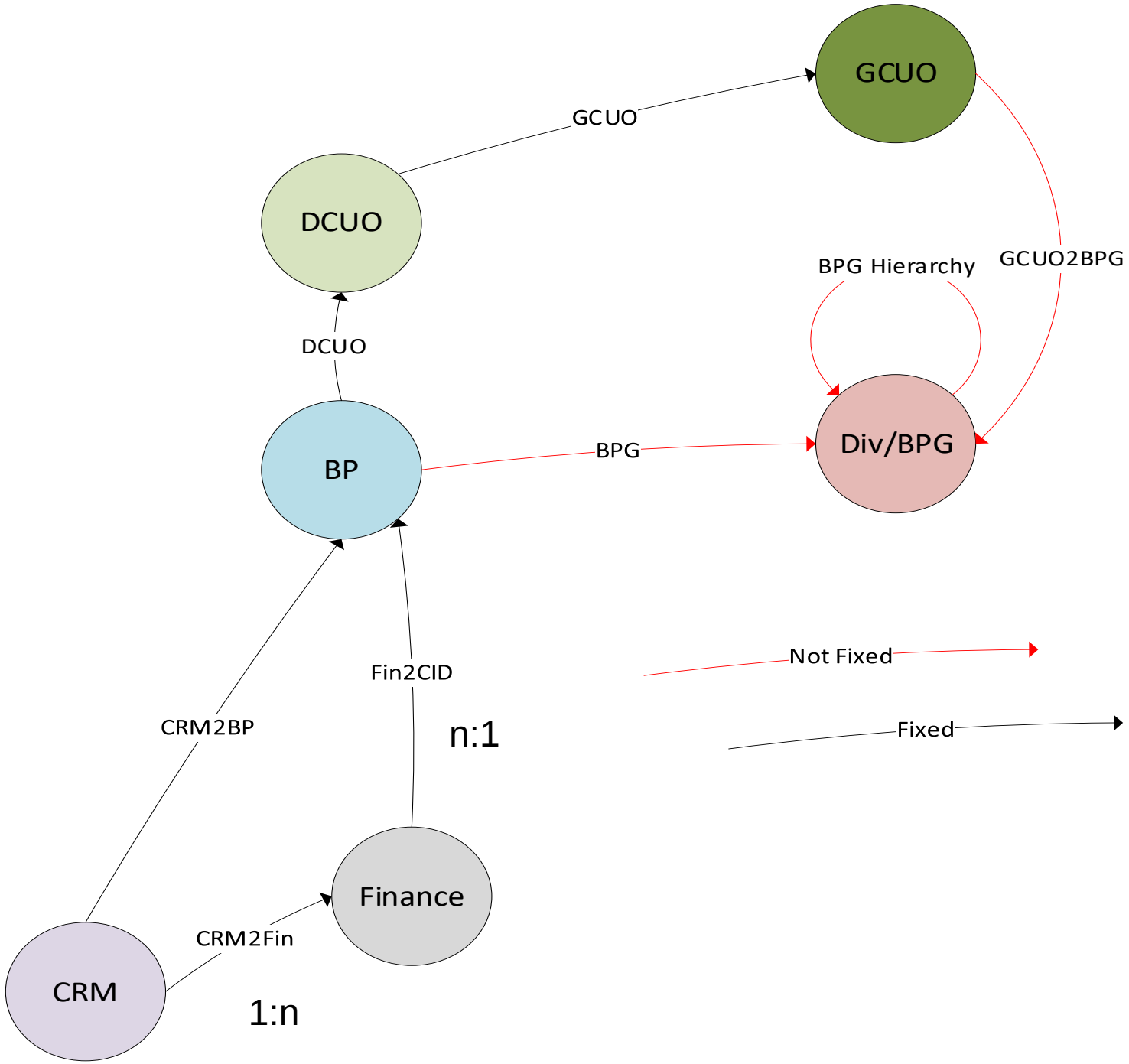
“Demo” KN Grouping



Grouping Business Partners

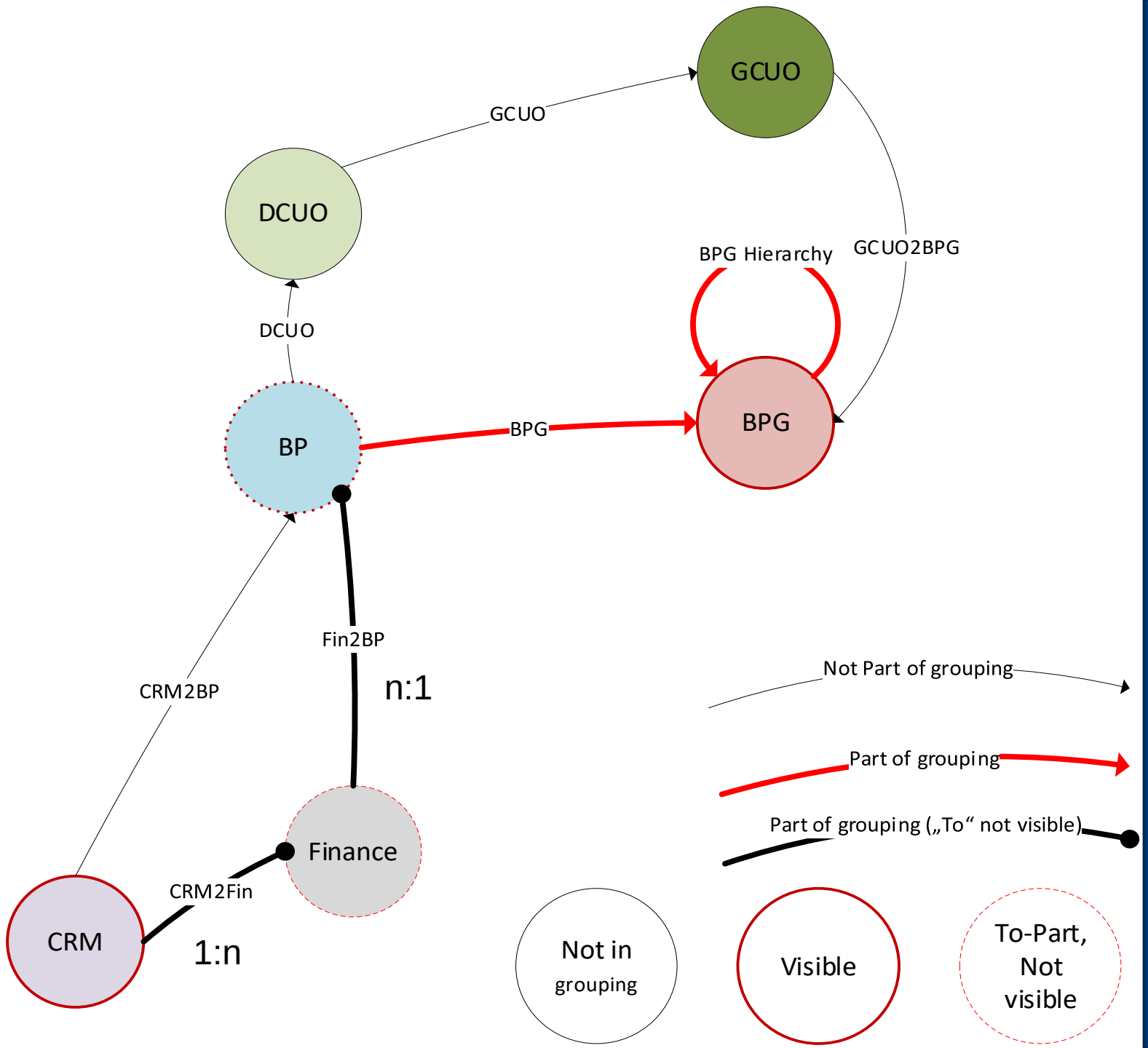


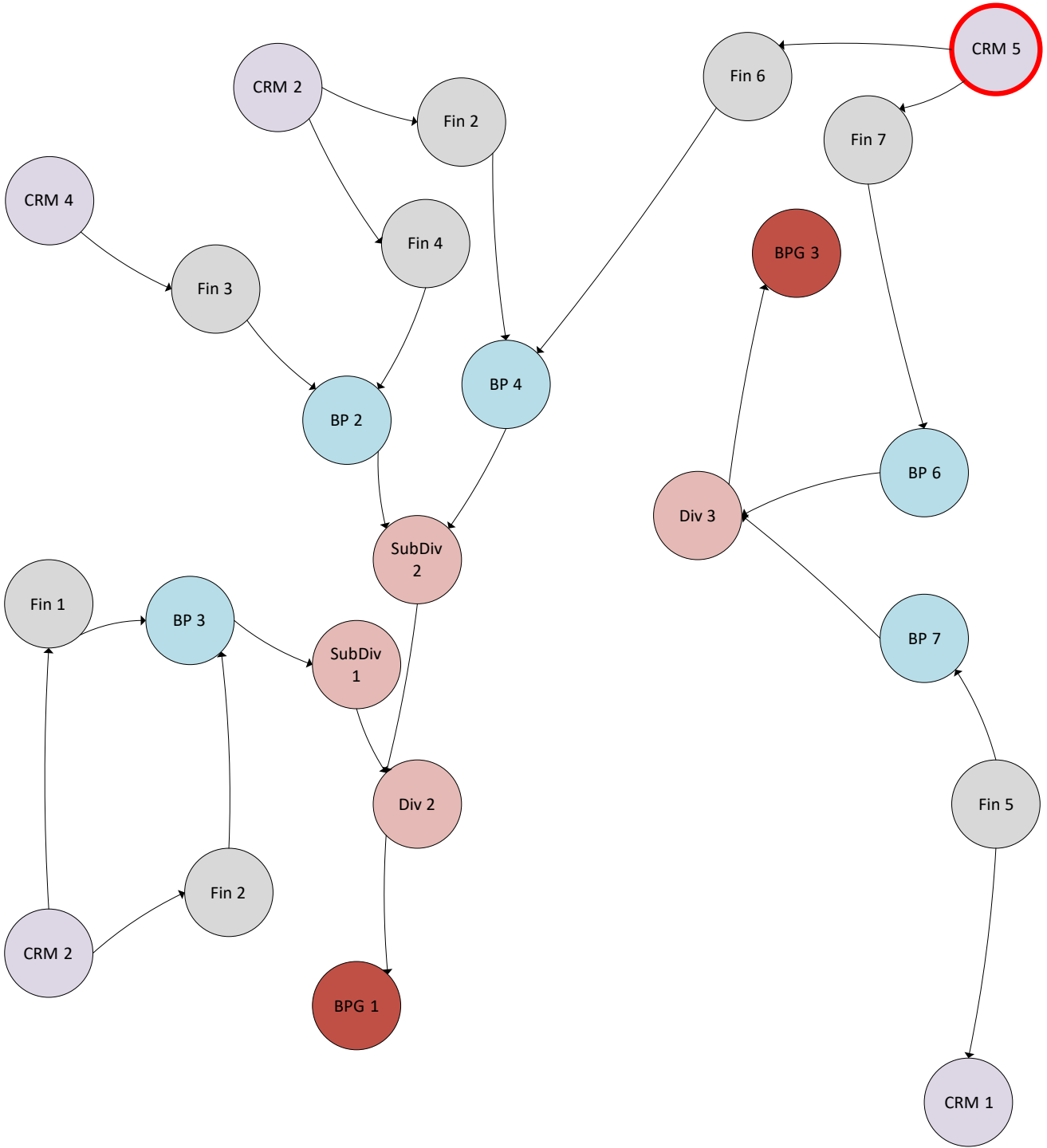
Grouping Business Partners



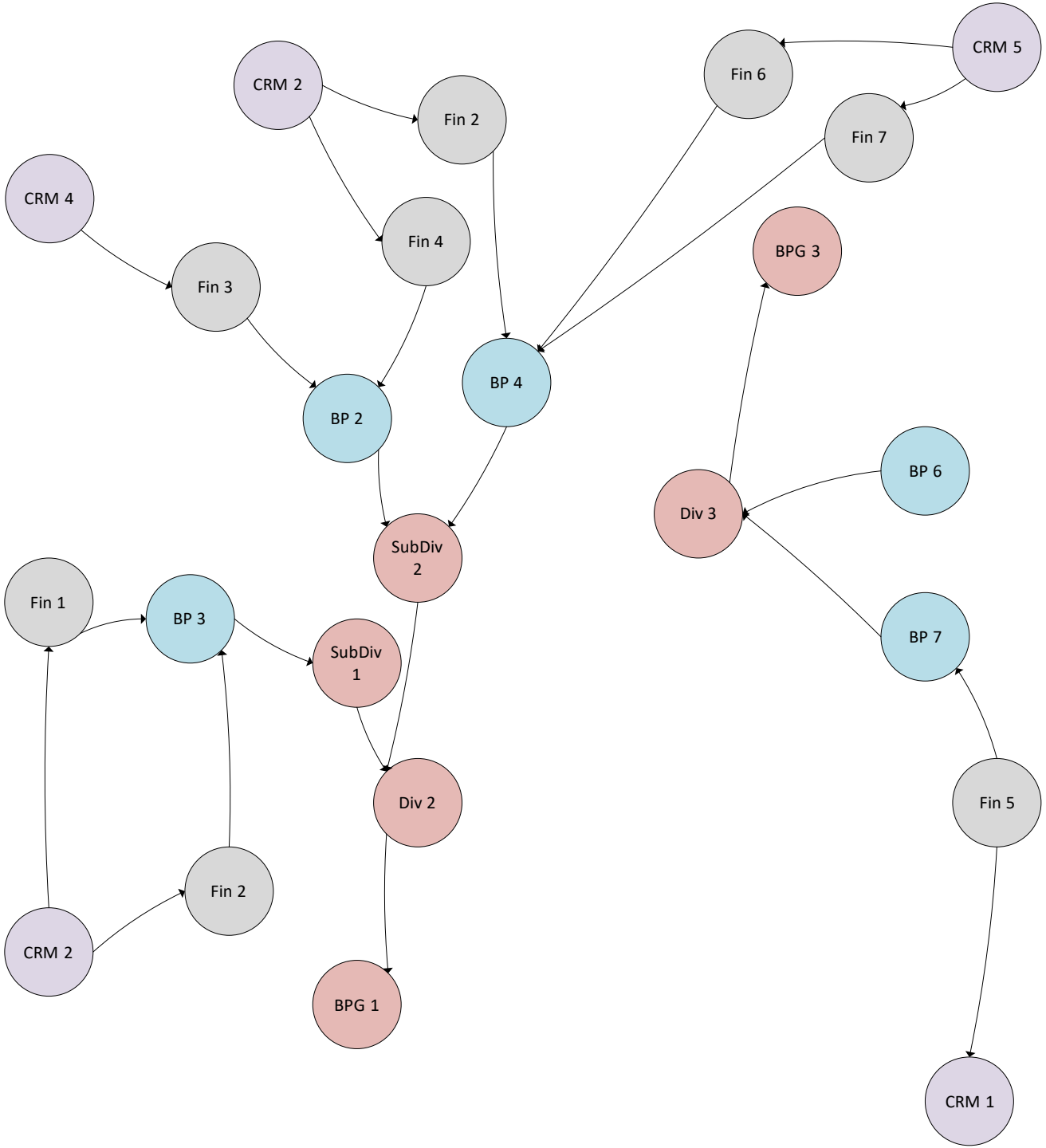
Grouping Business Partners

Grouping
Business Partners

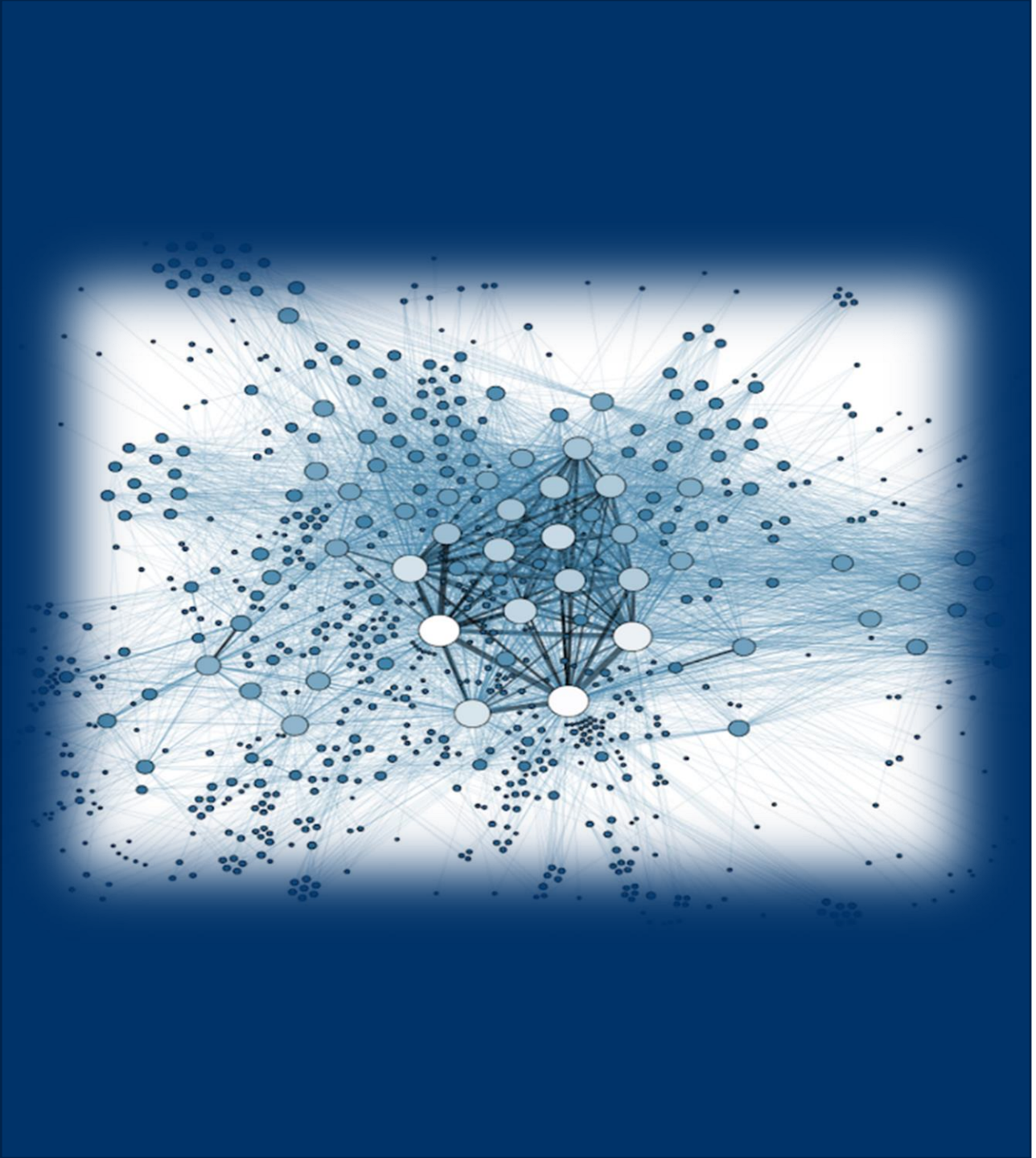



CRM 5
CRM 2
Fin 6
Fin 2
Fin 7
CRM 4
Fin 4
BPG 3
Fin 3
BP 4
BP 2
BP 6
Div 3
SubDiv 2
Fin 1
BP 3
BP 7
SubDiv 1
Fin 5
Div 2
Fin 2
CRM 2
BPG 1
CRM 1

Not allowed

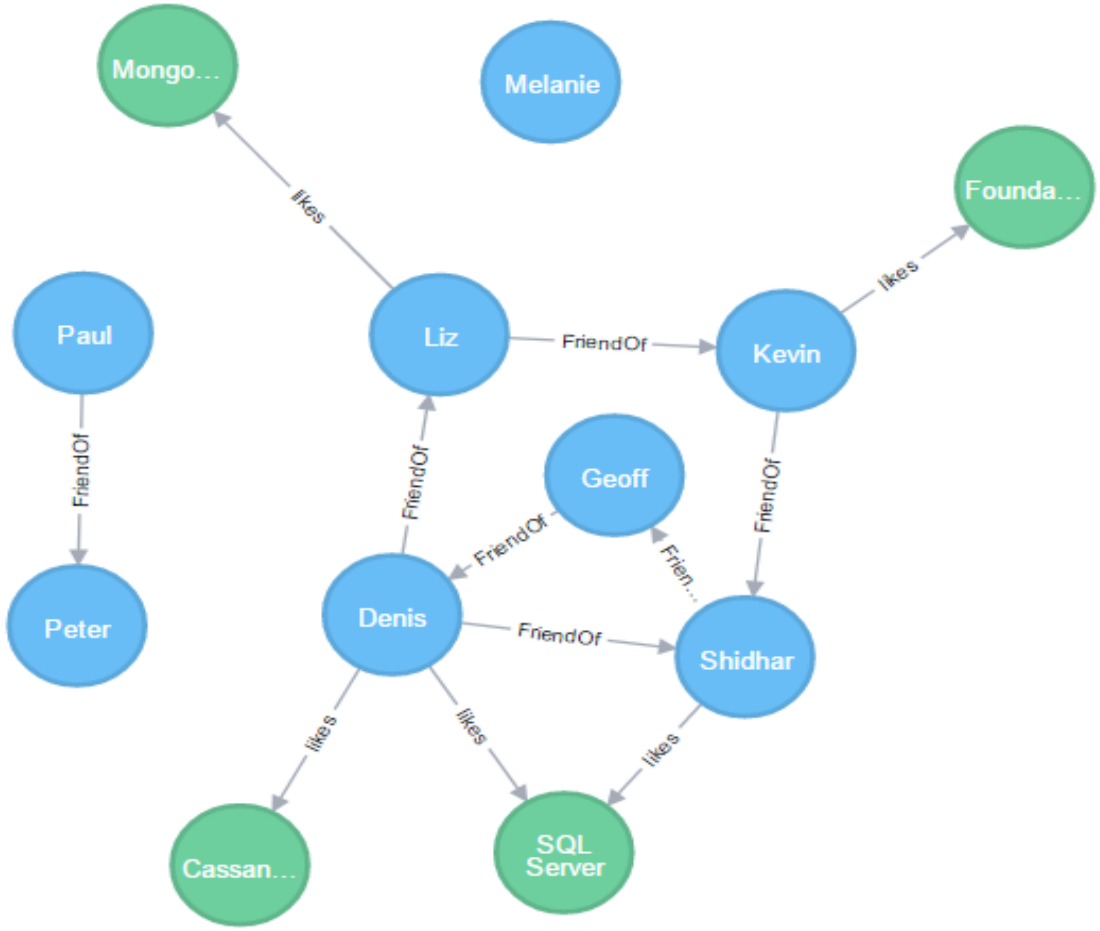

CRM 2
Fin 2
Fin 6
CRM 5
Fin 7
CRM 4
Fin 4
BPG 3
Fin 3
BP 4
BP 2
BP 6
Div 3
SubDiv 2
Fin 1
BP 3
BP 7
SubDiv 1
Div 2
Fin 5
Fin 2
CRM 2
BPG 1
CRM 1

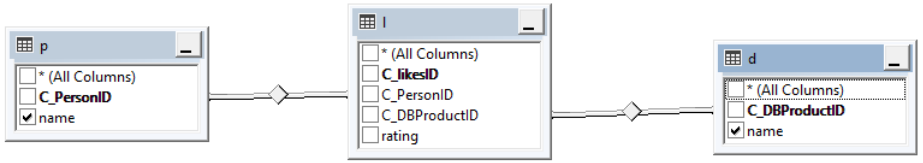
OK



1. Intro
2. **Graphs**
3. Knowledge Graphs
4. Implementation: RDF & SPARQL
5. Structure with RDFS & OWL
6. The closed world of SHACL
7. Wrap Up

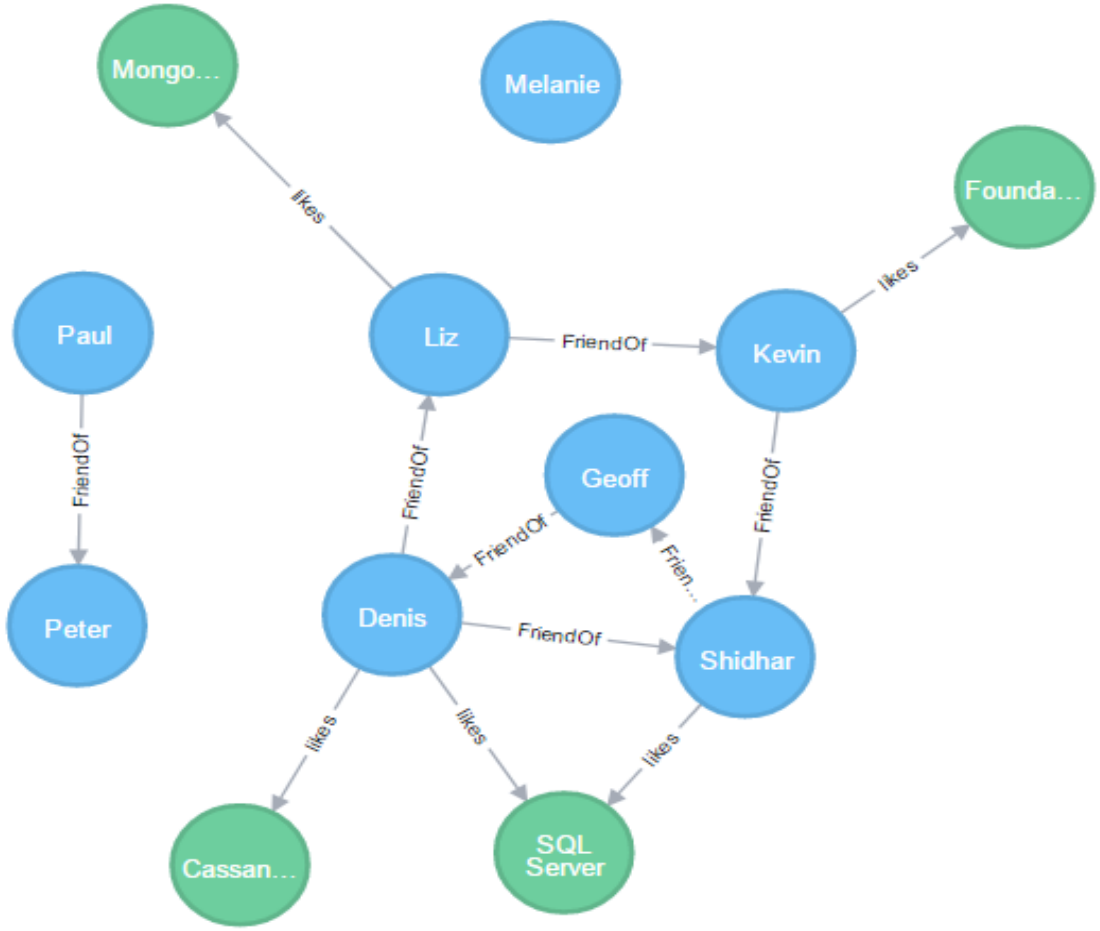
- Nodes
- Edges

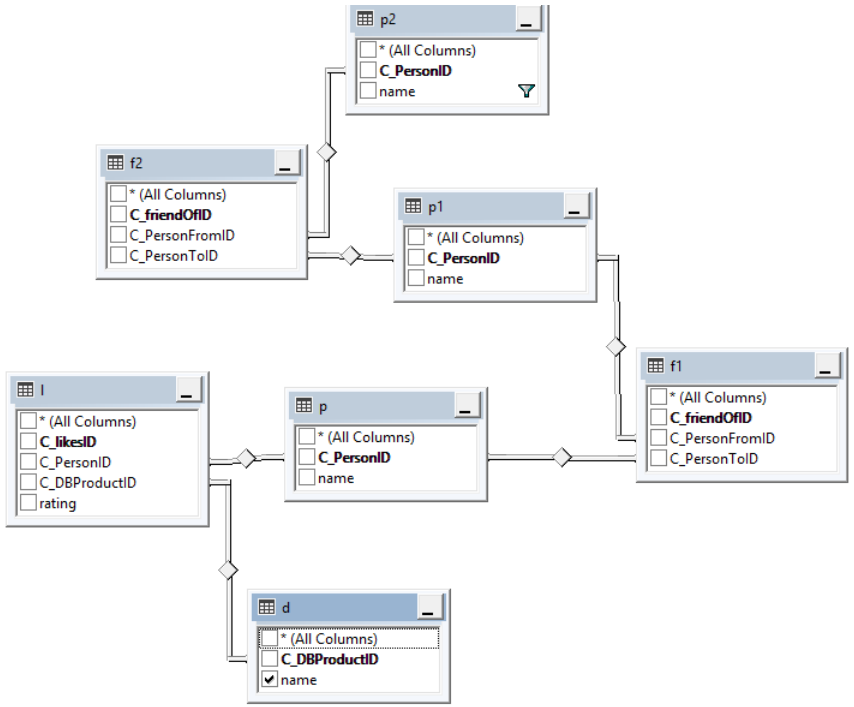

Mongo...
Melanie
Founda...
likes
likes
Paul
Liz
FriendOf
Kevin
FriendOf
FriendOf
FriendOf
Geoff
FriendOf
Frien...
Peter
Denis
FriendOf
Shidhar
likes
likes
likes
Cassan...
SQL Server



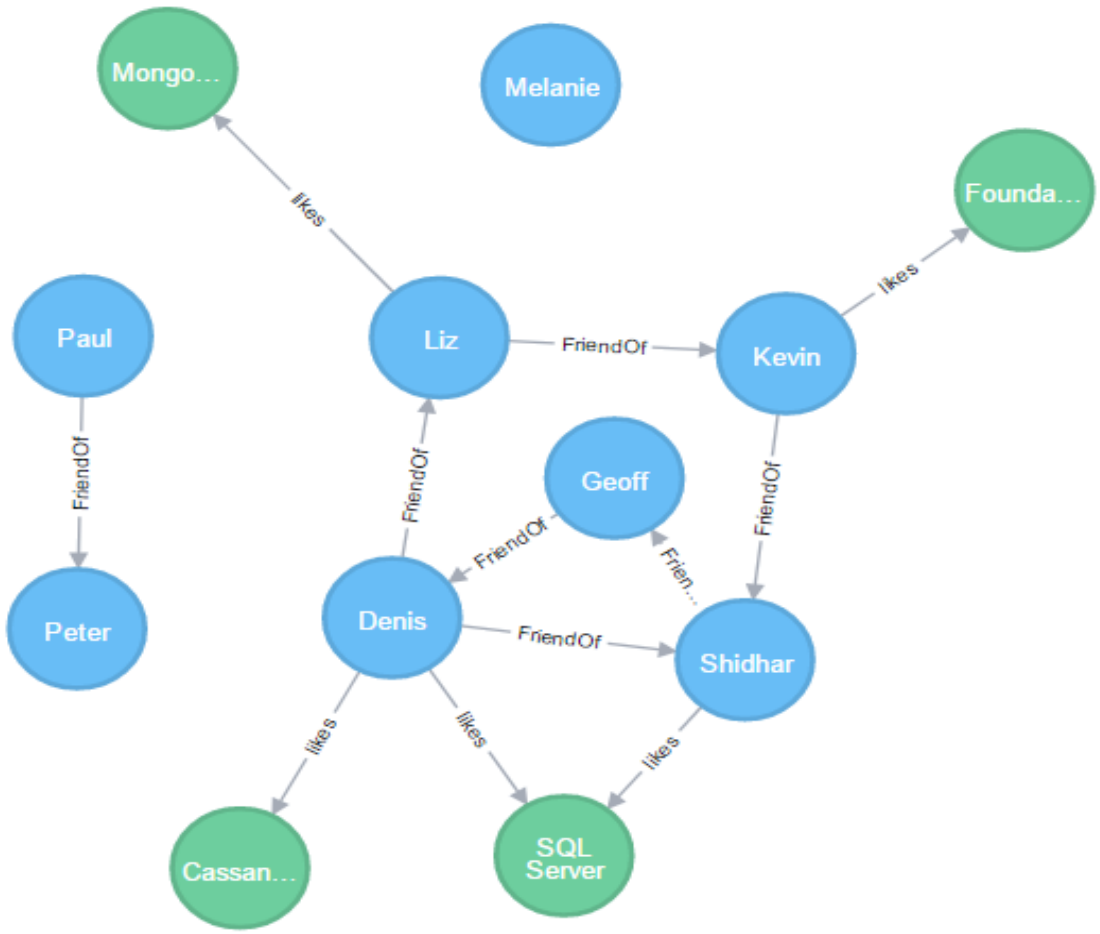
SQL:

A Developer likes a procuct

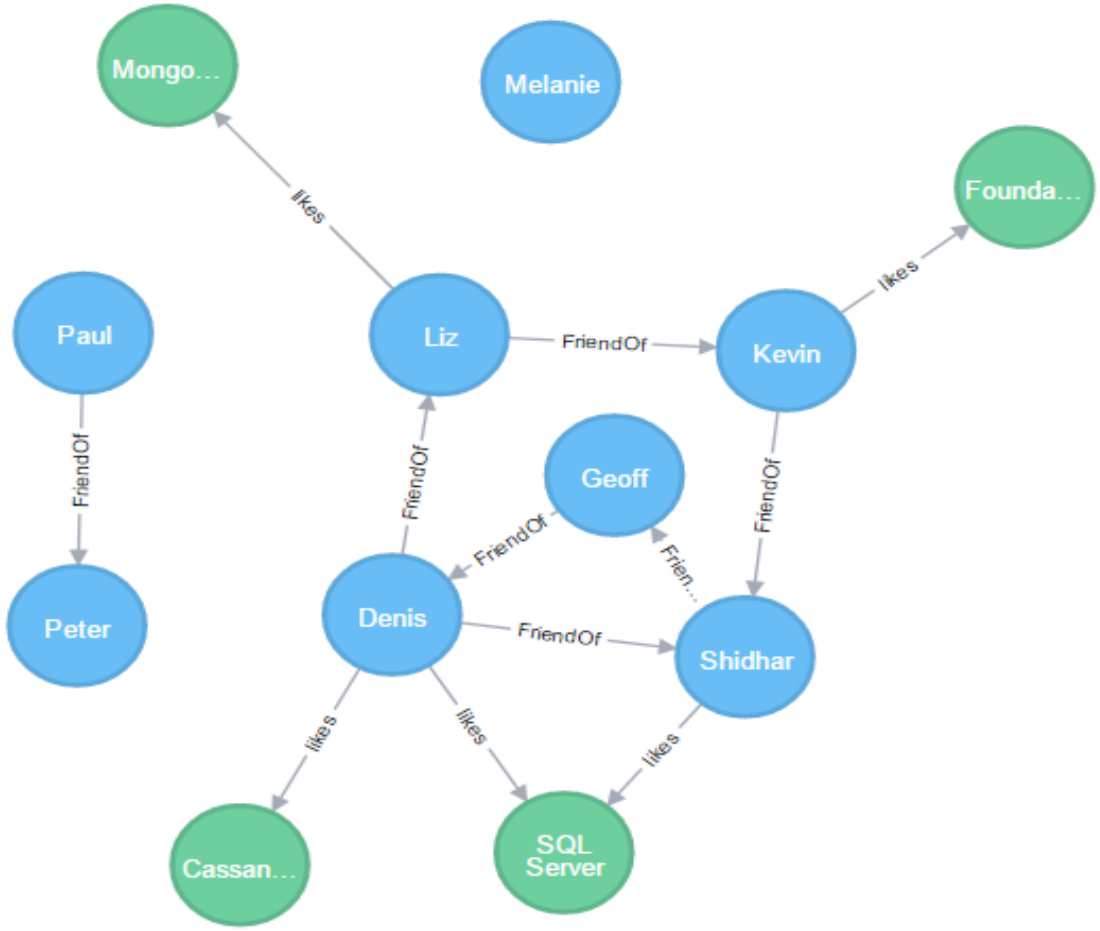


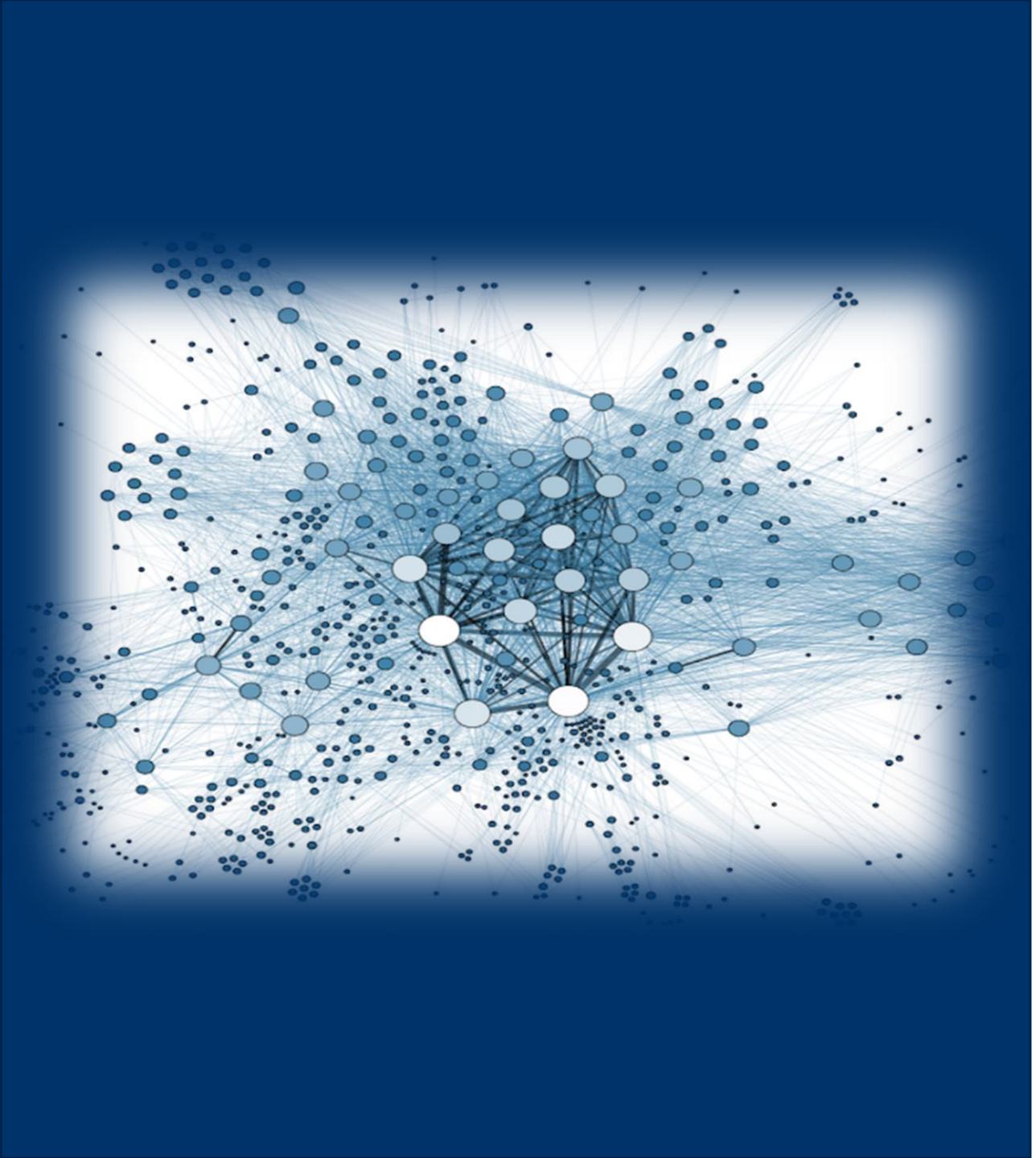


SQL:
Friend of a friend likes a procuct



Demo with neo4j & cypher

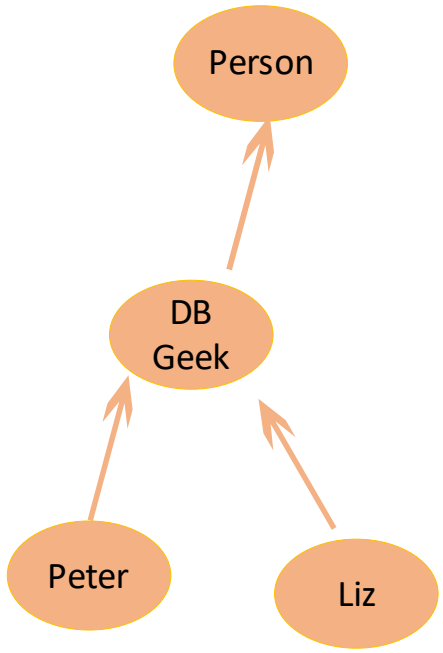


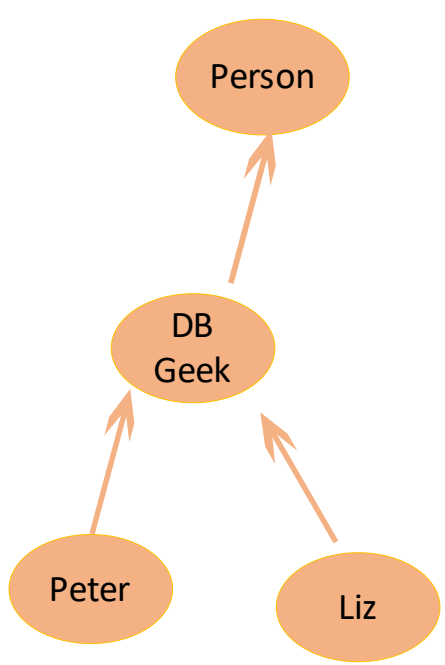


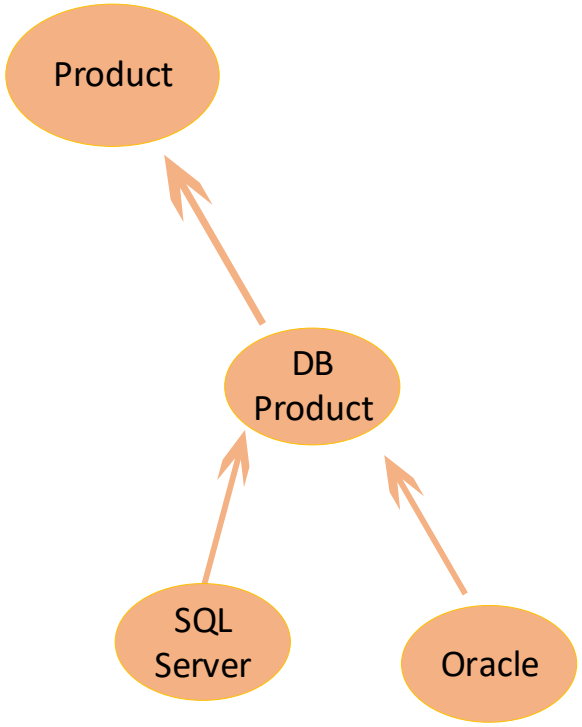
1. Intro
2. Graphs
3. **Knowledge Graphs**
4. Implementation: RDF & SPARQL
5. Structure with RDFS & OWL
6. The closed world of SHACL
7. Wrap Up

Wanted: Some structure/rules

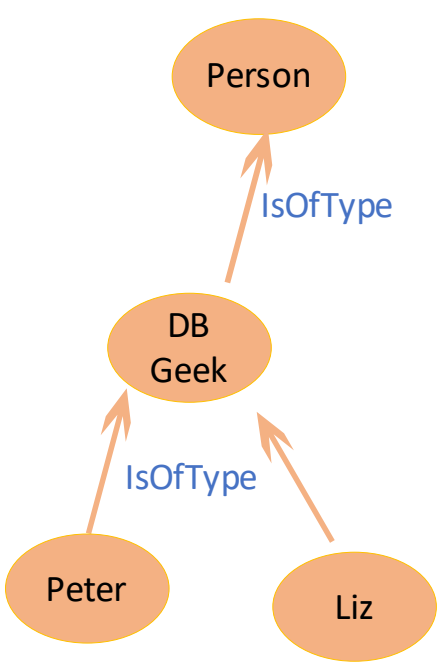
Wanted: Some structure/rules

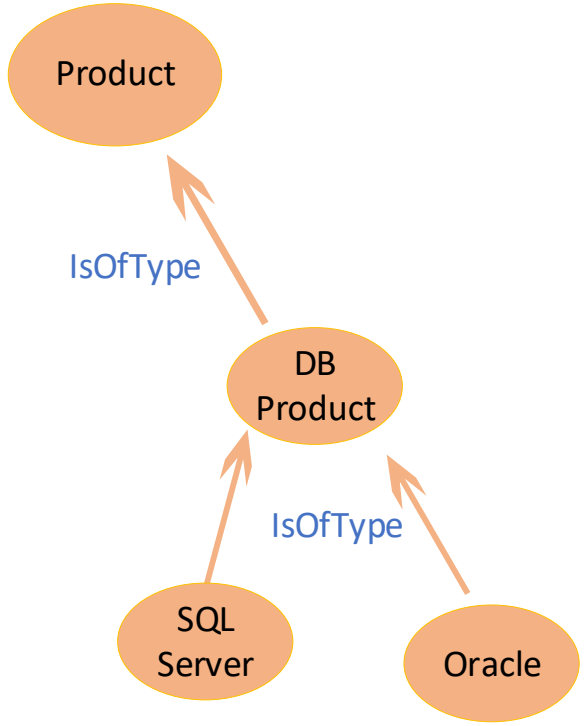




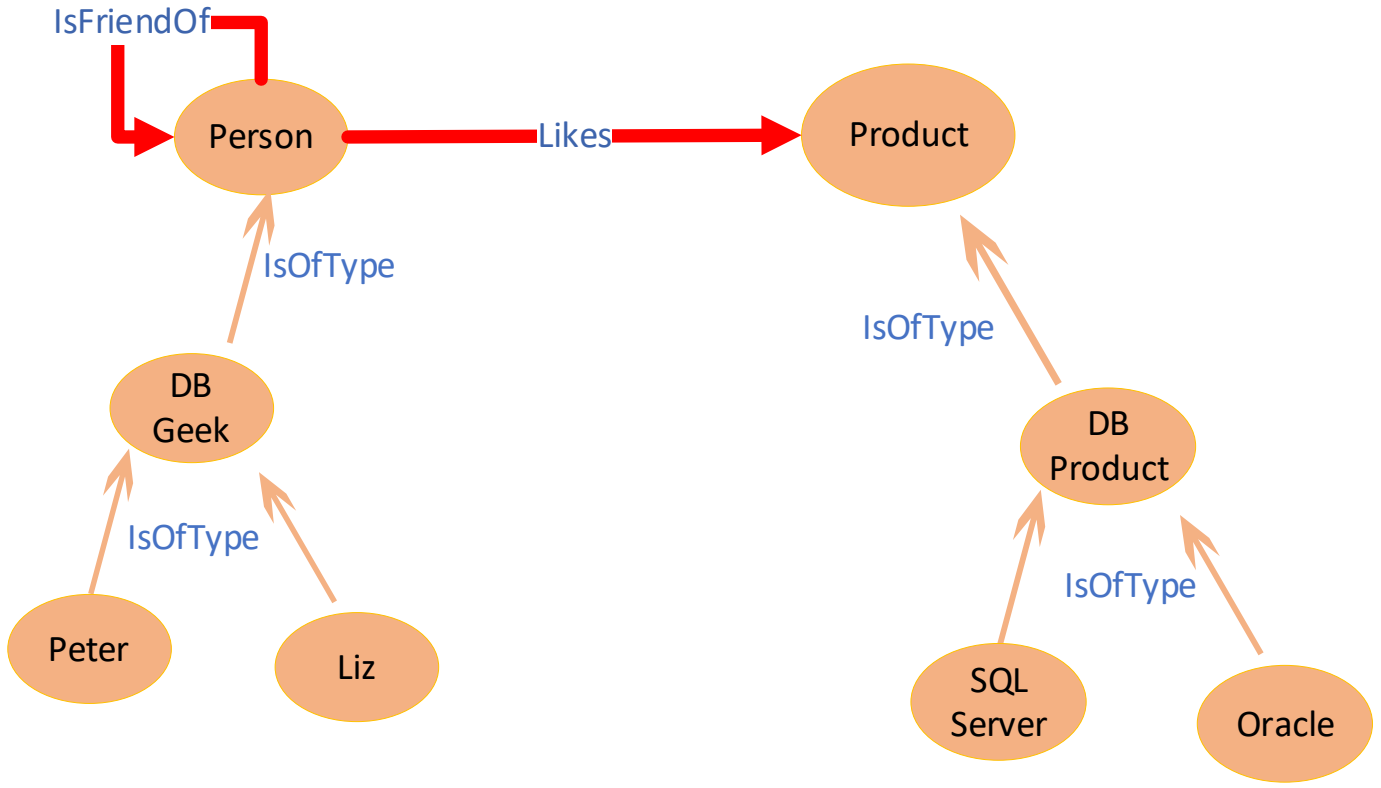


Wanted: Some structure/rules



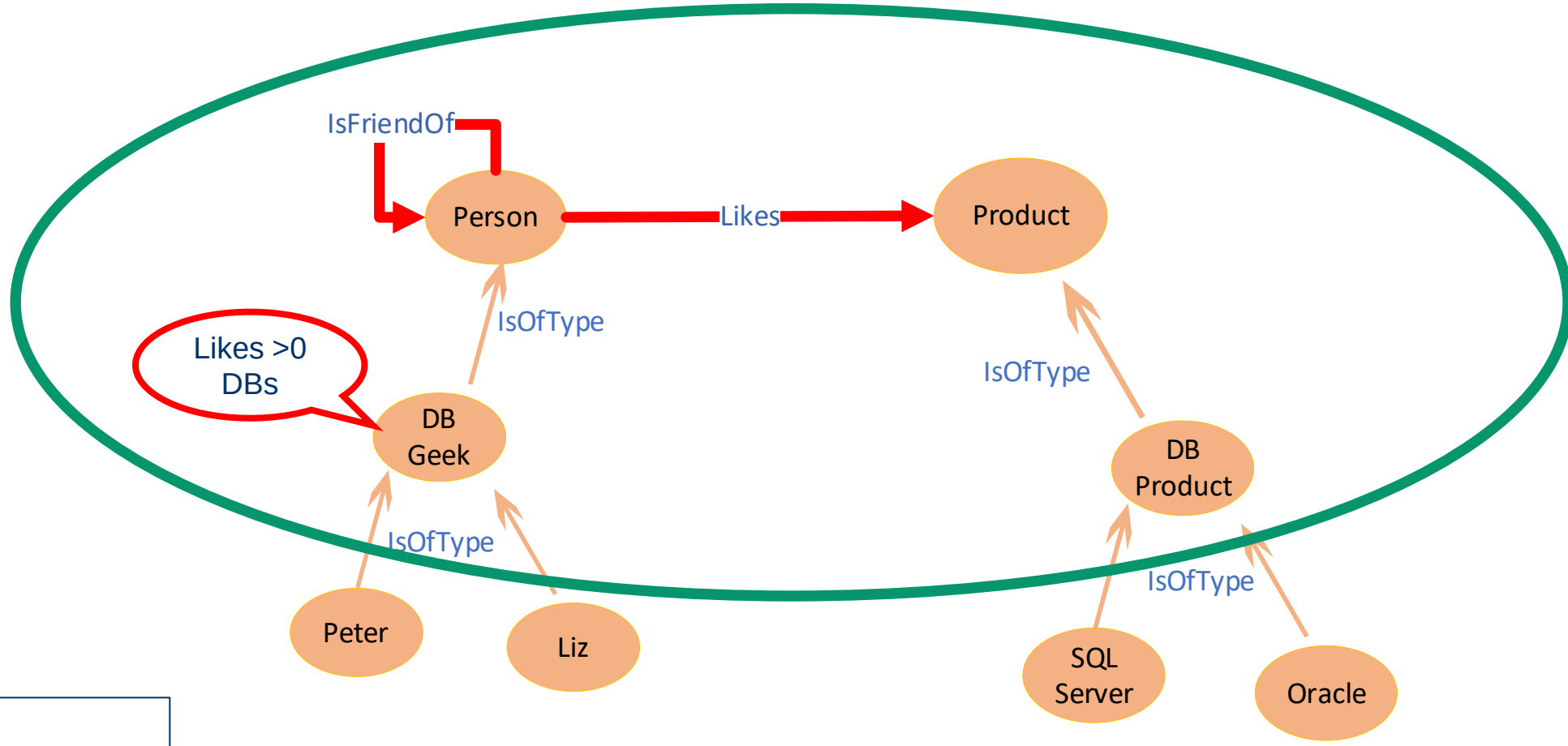


Wanted: Some structure/rules



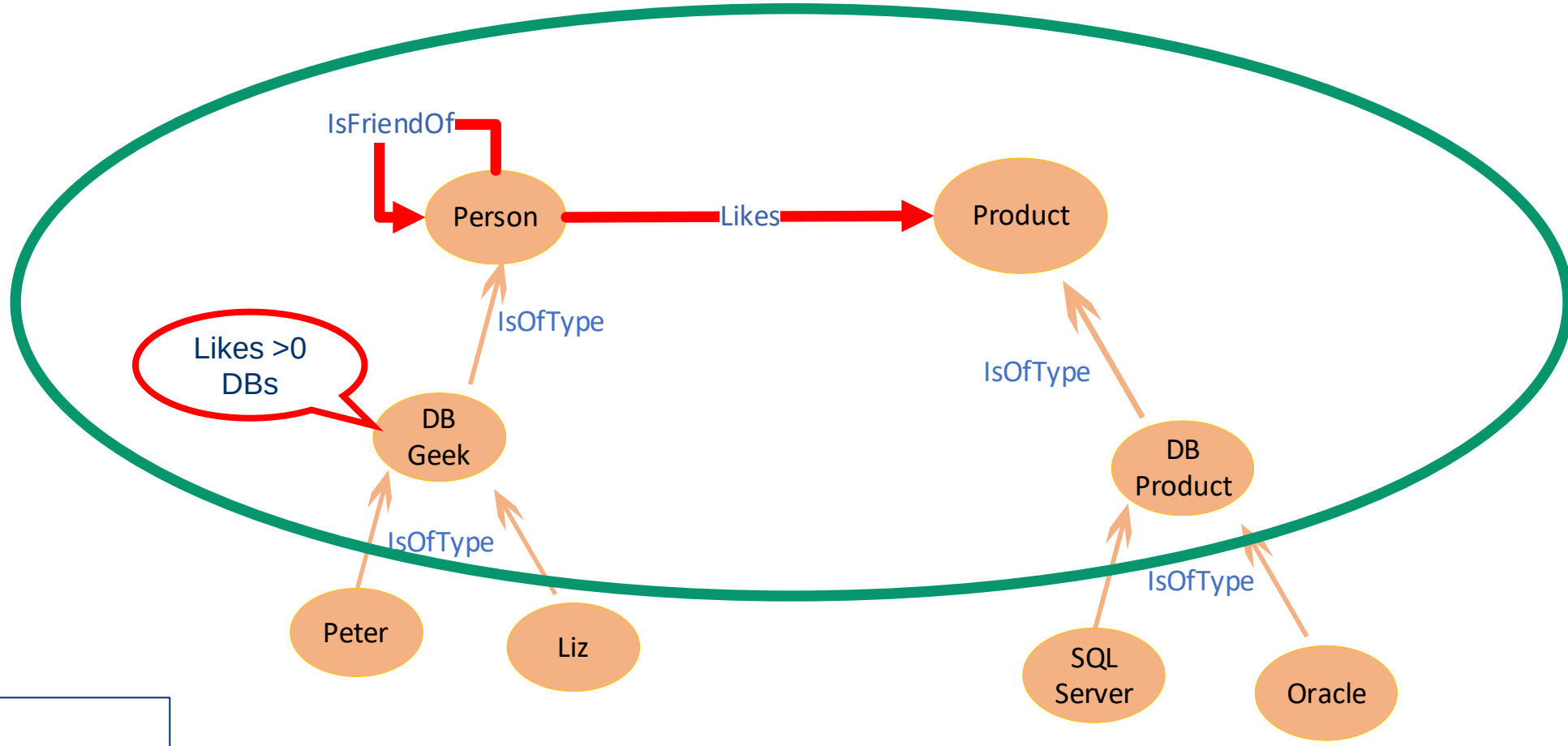
Wanted: Some structure/rules

Meta Daten



Wanted: Some structure/rules

Ontology



Wanted: Some structure/rules

Classes & individuals

Properties

Relationships (Handled as Properties too)

Rules & Constrains

Ontologies: A Graph to rule them all …

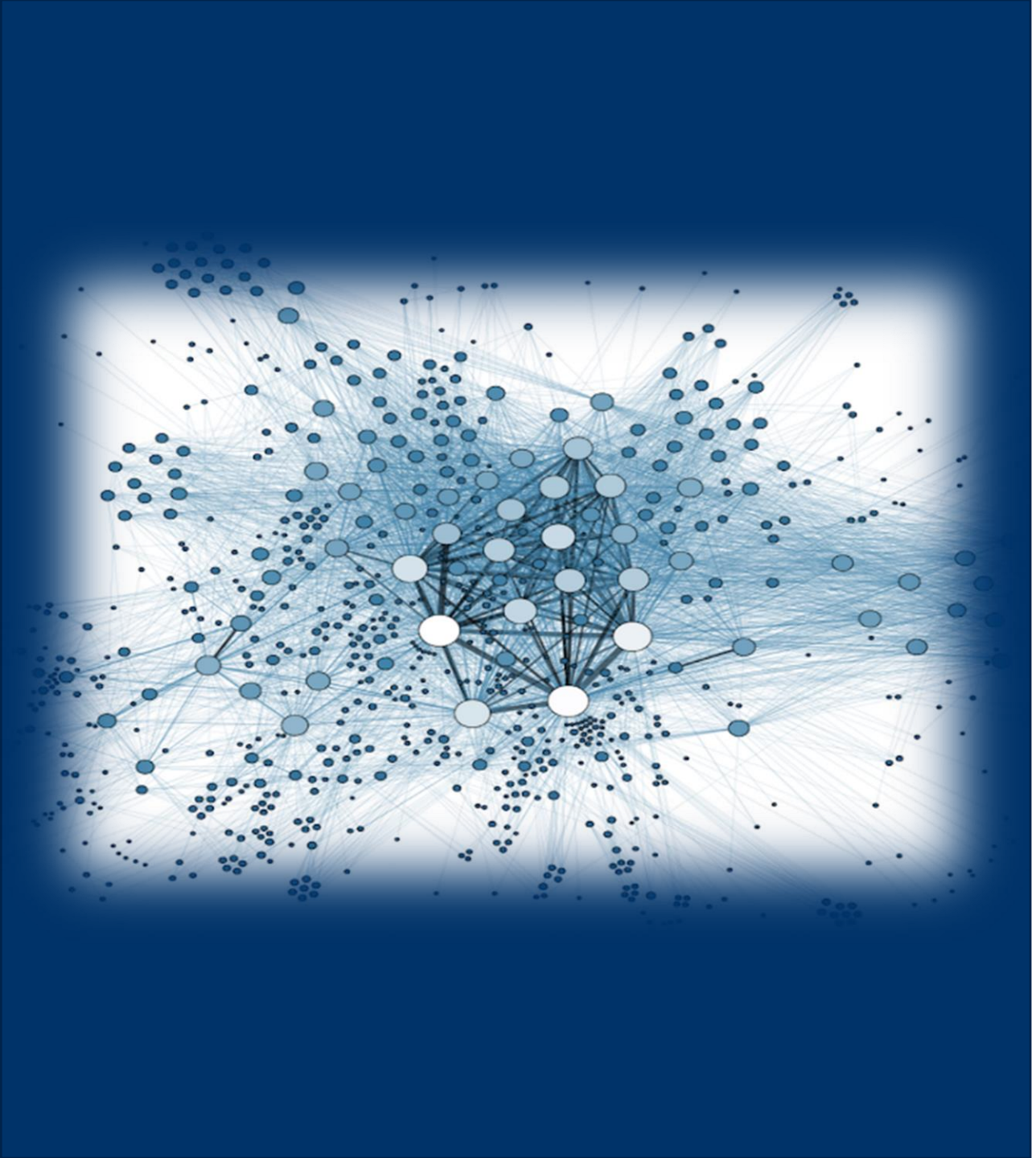
Knowledge Graph: Data Graph + Metadata Graph

Knowledge Graph: Data Graph + Ontology

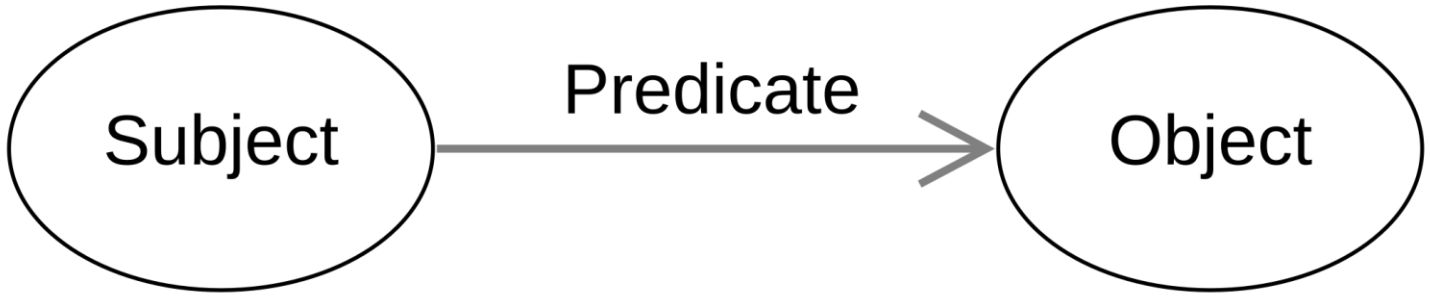


The Open World assumption

And its dark sister …



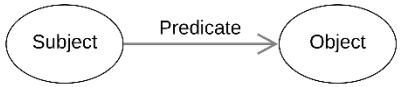
1. Intro
2. Graphs
3. Knowledge Graphs
4. **Implementation: RDF & SPARQL**
5. Structure with RDFS & OWL
6. The closed world of SHACL
7. Wrap Up



Tripple

RDF

Resource Description Framework



Turtle

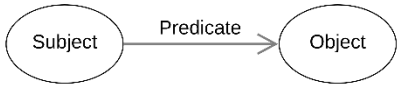
Terse RDF Triple Language

```
@prefix ex: <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .


ex:Developer7 rdf:type ex:Developer .
ex:Developer7 ex:hasName "Denis" .
ex:Developer7 ex:likes ex:Product1 .
ex:Developer7 ex:likes ex:Product2 .
ex:Developer7 ex:FriendOf ex:Developer8 .

# ...

ex:Product1 rdf:type ex:DatabaseProduct .
ex:Product1 ex:hasName "SQL Server" .
```



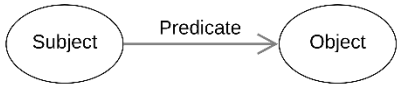
Turtle

Terse RDF Triple Language

```
<http://example.org/>:Developer7 <http://www.w3.org/1999/02/22-rdf-syntax-ns#>:type <http://example.org/>:Developer .
<http://example.org/>:Developer7 <http://example.org/>:hasName "Denis" .
<http://example.org/>:Developer7 <http://example.org/>:likes <http://example.org/>:Product1 .
<http://example.org/>:Developer7 <http://example.org/>:likes <http://example.org/>:Product2 .
<http://example.org/>:Developer7 <http://example.org/>:FriendOf <http://example.org/>:Developer8 .

# ...

<http://example.org/>:Product1 <http://www.w3.org/1999/02/22-rdf-syntax-ns#>:type <http://example.org/>:DatabaseProduct .
<http://example.org/>:Product1 <http://example.org/>:hasName "SQL Server" .
```



Turtle

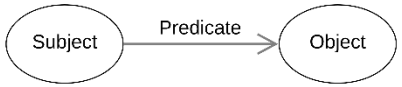
Terse RDF Triple Language

```
@prefix ex: <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .


ex:Developer7 rdf:type ex:Developer .
ex:Developer7 ex:hasName "Denis" .
ex:Developer7 ex:likes ex:Product1 .
ex:Developer7 ex:likes ex:Product2 .
ex:Developer7 ex:FriendOf ex:Developer8 .

# ...

ex:Product1 rdf:type ex:DatabaseProduct .
ex:Product1 ex:hasName "SQL Server" .
```



Turtle

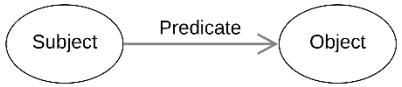
Terse RDF Triple Language

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix ex: <http://example.org/> .

ex:Developer7 rdf:type ex:Developer ;
    ex:hasName "Denis" ;
    ex:likes ex:Product1 ,
        ex:Product2 ;
    ex:FriendOf ex:Developer8 .

# ...

ex:Product1 rdf:type ex:DatabaseProduct .
ex:Product1 ex:hasName "SQL Server" .
```



Turtle

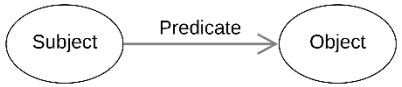
Terse RDF Triple Language

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix ex: <http://example.org/> .

ex:Developer7 a ex:Developer ;
    ex:hasName "Denis" ;
    ex:likes ex:Product1 ,
        ex:Product2 ;
    ex:FriendOf ex:Developer8 .

# ...

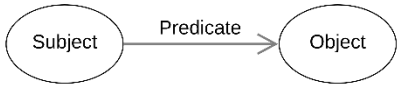
ex:Product1 a ex:DatabaseProduct .
ex:Product1 ex:hasName "SQL Server" .
```



Let’s Demo

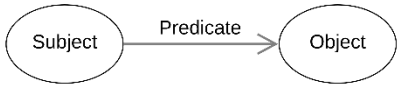
GraphDB

protégé



Let’s Demo

GraphDB



Turtle

Explicit classes & Properties

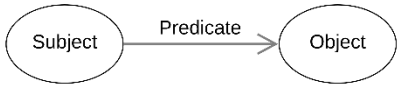
```
@prefix ex: <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

# Classes
ex:Developer a rdf:Class .
ex:DatabaseProduct a rdf:Class .

# Properties
ex:hasName a rdf:Property .
ex:likes a rdf:Property .
ex:FriendOf a rdf:Property .

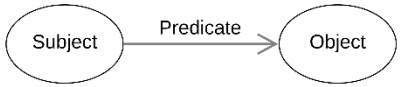
# Instances
ex:Developer7 a ex:Developer ;
    ex:hasName "Denis" ;
    ex:likes ex:Product1 ;
    ex:FriendOf ex:Developer8 .
…

ex:Product7 a ex:DatabaseProduct ;
    ex:hasName "Volt" .
```



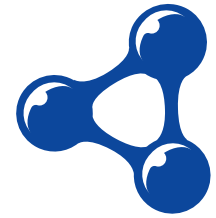
Demo Protege





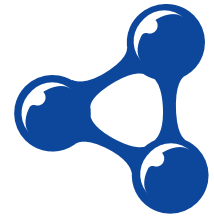
An ***RDF graph*** is a set of RDF triples.

https://www.w3.org/TR/rdf11-concepts/#dfn-concrete-rdf-syntax



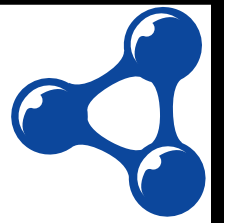
SPARQL

SPARQL **P**rotocol **A**nd **R**DF **Q**uery **L**anguage



Friends Demo





```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX d: <http://dbpedia.org/ontology/>
PREFIX : <http://dbpedia.org/resource/>

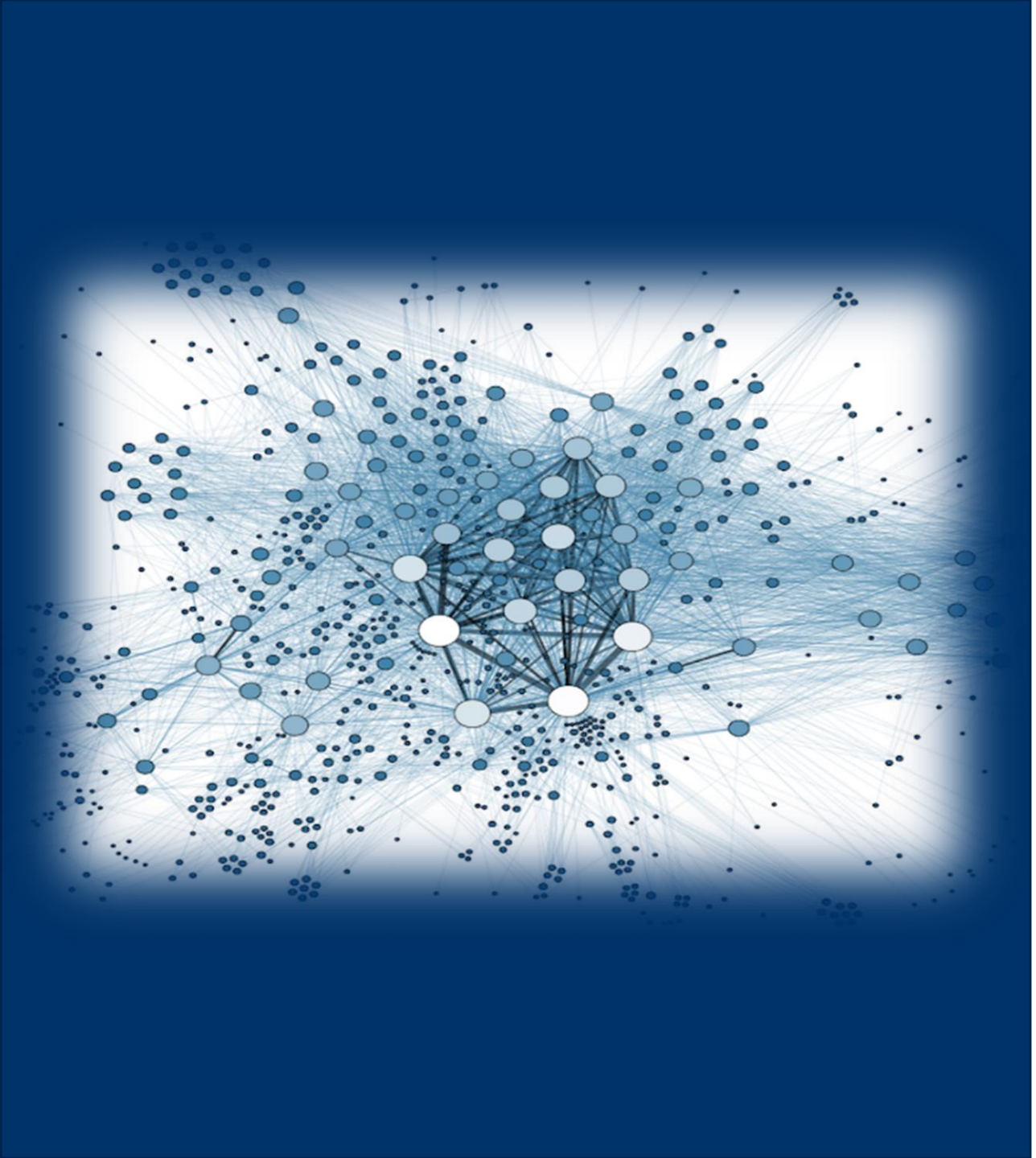
SELECT DISTINCT ?albumName ?ProducerName
WHERE {
  ?album d:producer ?Producer.
  ?Producer rdfs:label ?ProducerName.
  ?album rdfs:label ?albumName.
  FILTER (regex(?ProducerName, "Dieter Bohlen", "i"))
  FILTER (lang(?ProducerName) = "en")
  FILTER (lang(?albumName) = "en")
}
```

Excurse: Open Data Sets
Demo: SPARQL against DBPedia

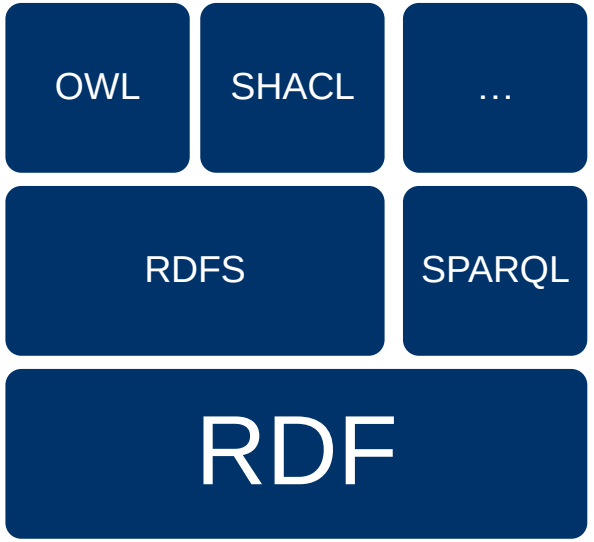
Public sources

Open Data Sets

- DBPedia
- https://www.govdata.de
- http://www.geonames.org/
- …



1. Intro
2. Graphs
3. Knowledge Graphs
4. Implementation: RDF & SPARQL
5. **Structure with RDFS & OWL**
6. The closed world of SHACL
7. Wrap Up

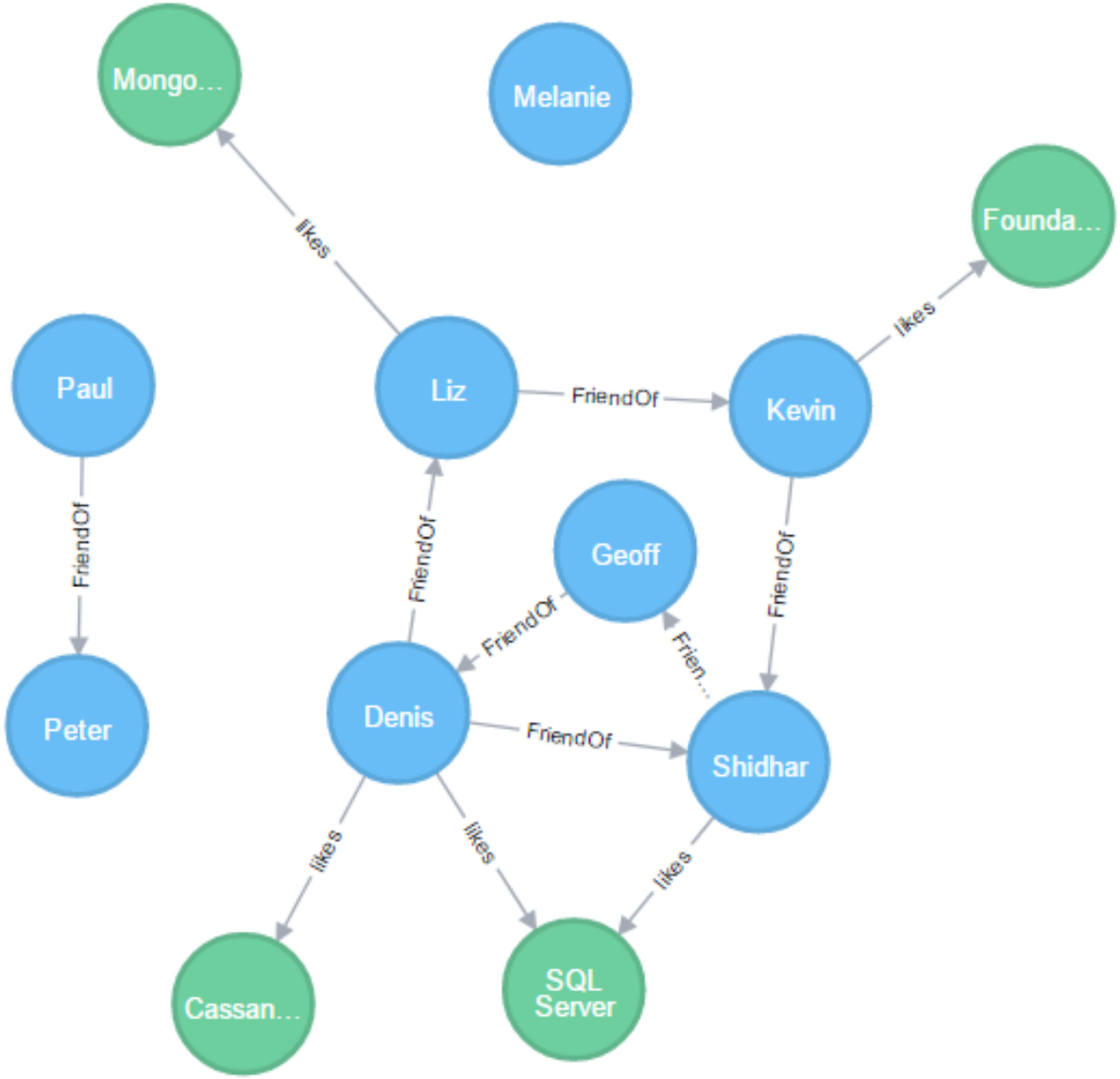


An ***RDF graph*** is a set of RDF triples.

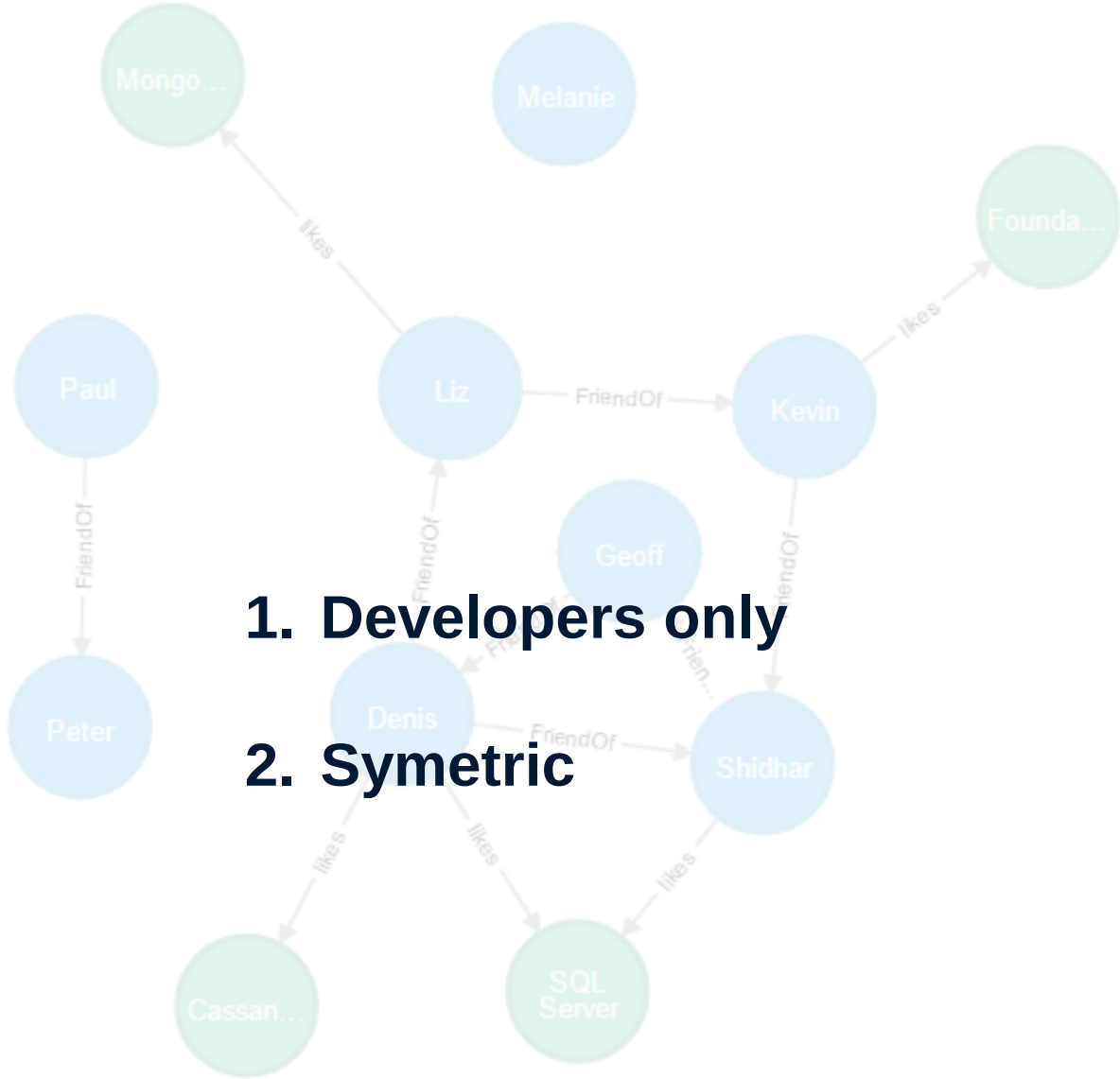
Structure comes with RDFS, OWL etc.

https://www.w3.org/TR/rdf11-concepts/#dfn-concrete-rdf-syntax

Back to (true) friendship

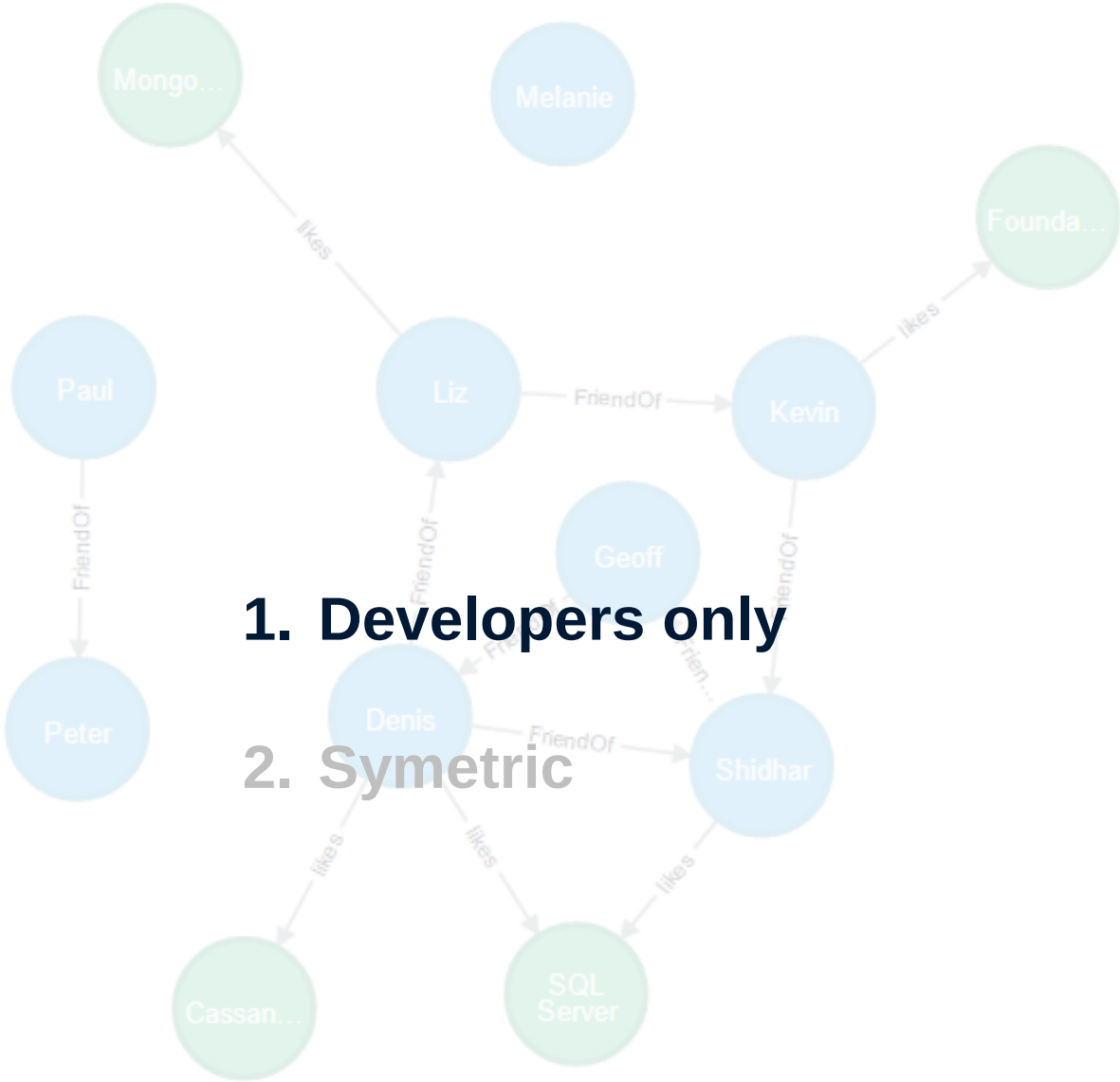


Back to (true) friendship



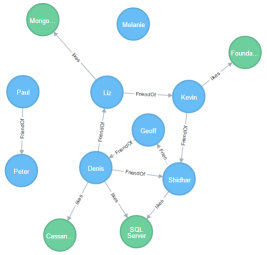
1. Developers only
2. Symetric

Back to (true) friendship



1. Developers only
2. Symetric

RDFS




```
@prefix ex: <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

# Classes
ex:Developer a rdfs:Class .
ex:DatabaseProduct a rdfs:Class .

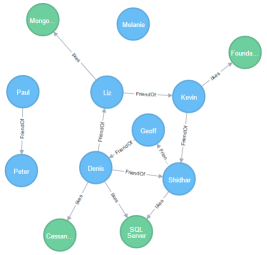
# Properties
ex:hasName a rdf:Property ;
    rdfs:domain ex:Developer ;
    rdfs:range rdfs:Literal .

ex:likes a rdf:Property ;
    rdfs:domain ex:Developer ;
    rdfs:range ex:DatabaseProduct .

ex:FriendOf a rdf:Property ;
    rdfs:domain ex:Developer ;
    rdfs:range ex:Developer .

# Instances
ex:Developer7 a ex:Developer ;
    ex:hasName "Denis" ;
    ex:likes ex:Product1 ;
    ex:FriendOf ex:Developer8 .
```



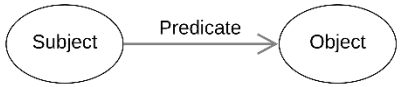


Demo Protégé

Friends in RDFS

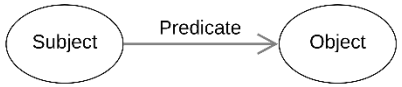
RDFS_Friends.ttl

RDF/RDFS classes



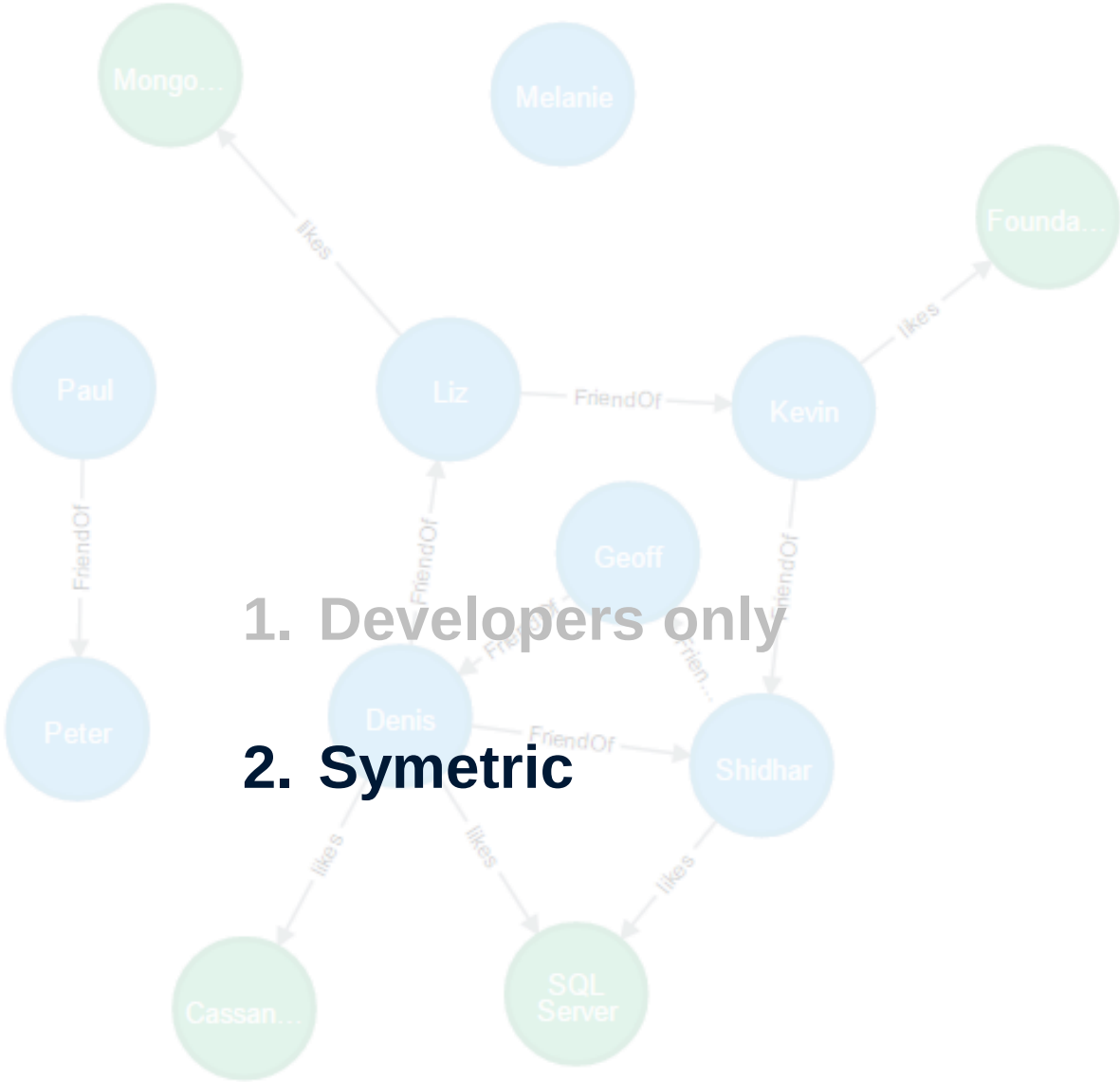
Class name	comment
rdfs:Resource	The class resource, everything.
rdfs:Literal	The class of literal values, e.g. textual strings and integers.
rdf:langString	The class of language-tagged string literal values.
rdf:HTML	The class of HTML literal values.
rdf:XMLLiteral	The class of XML literal values.
rdfs:Class	The class of classes.
rdf:Property	The class of RDF properties.
rdfs:Datatype	The class of RDF datatypes.
rdf:Statement	The class of RDF statements.
rdf:Bag	The class of unordered containers.
rdf:Seq	The class of ordered containers.
rdf:Alt	The class of containers of alternatives.
rdfs:Container	The class of RDF containers.
rdfs:ContainerMembershipProperty	The class of container membership properties, rdf:_1, rdf:_2, ..., all of which are sub-properties of 'member'.
rdf:List	The class of RDF Lists.

RDF/RDFS properties



Property name	comment	domain	range
rdf:type	The subject is an instance of a class.	rdfs:Resource	rdfs:Class
rdfs:subClassOf	The subject is a subclass of a class.	rdfs:Class	rdfs:Class
rdfs:subPropertyOf	The subject is a subproperty of a property.	rdf:Property	rdf:Property
rdfs:domain	A domain of the subject property.	rdf:Property	rdfs:Class
rdfs:range	A range of the subject property.	rdf:Property	rdfs:Class
rdfs:label	A human-readable name for the subject.	rdfs:Resource	rdfs:Literal
rdfs:comment	A description of the subject resource.	rdfs:Resource	rdfs:Literal
rdfs:member	A member of the subject resource.	rdfs:Resource	rdfs:Resource
rdf:first	The first item in the subject RDF list.	rdf:List	rdfs:Resource
rdf:rest	The rest of the subject RDF list after the first item.	rdf:List	rdf:List
rdfs:seeAlso	Further information about the subject resource.	rdfs:Resource	rdfs:Resource
rdfs:isDefinedBy	The definition of the subject resource.	rdfs:Resource	rdfs:Resource
rdf:value	Idiomatic property used for structured values.	rdfs:Resource	rdfs:Resource
rdf:subject	The subject of the subject RDF statement.	rdf:Statement	rdfs:Resource
rdf:predicate	The predicate of the subject RDF statement.	rdf:Statement	rdfs:Resource
rdf:object	The object of the subject RDF statement.	rdf:Statement	rdfs:Resource

Back to (true) friendship

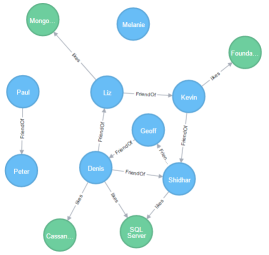


1. Developers only
2. **Symetric**



OWL

Web Ontology Language




```
@prefix ex: <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .

# Classes
ex:Developer a rdfs:Class .
ex:DatabaseProduct a rdfs:Class .

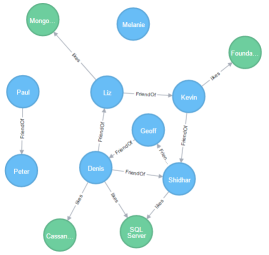
# Properties
ex:hasName a rdf:Property ;
    rdfs:domain ex:Developer ;
    rdfs:range rdfs:Literal .

ex:likes a rdf:Property ;
    rdfs:domain ex:Developer ;
    rdfs:range ex:DatabaseProduct .

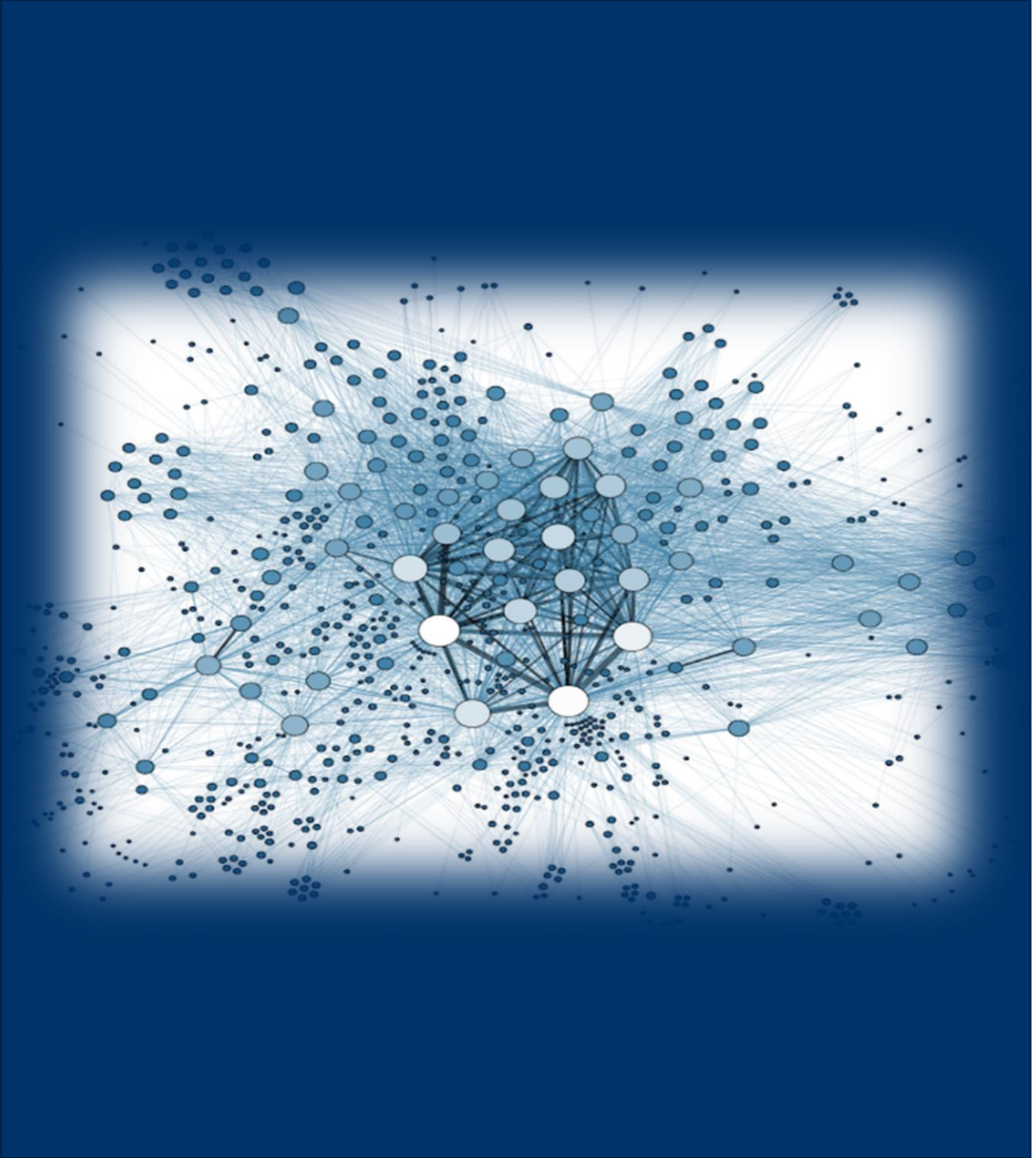
ex:FriendOf a owl:SymmetricProperty, rdf:Property ;
    rdfs:domain ex:Developer ;
    rdfs:range ex:Developer .

# Instances
ex:Developer7 a ex:Developer ;
    ex:hasName "Denis" ;
    ex:likes ex:Product1 ;
    ex:FriendOf ex:Developer8 .
```

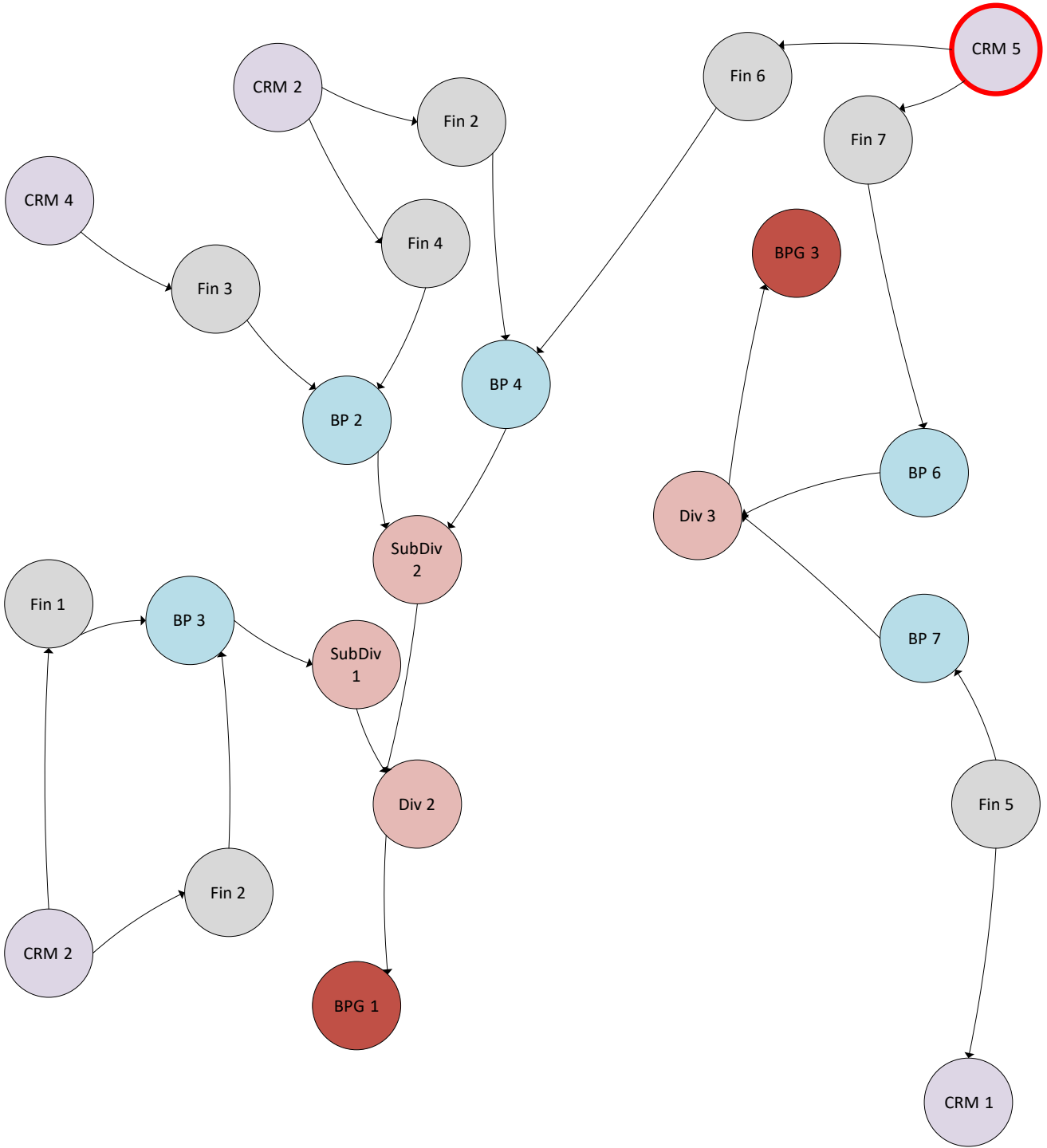




Demo Protege



1. Intro
2. Graphs
3. Knowledge Graphs
4. Implementation: RDF & SPARQL
5. Structure with RDFS & OWL
6. **The closed world of SHACL**
7. Wrap Up



Remember:

Not allowed …

The closed world of SHACL

Shape **C**onstrain **L**anguage



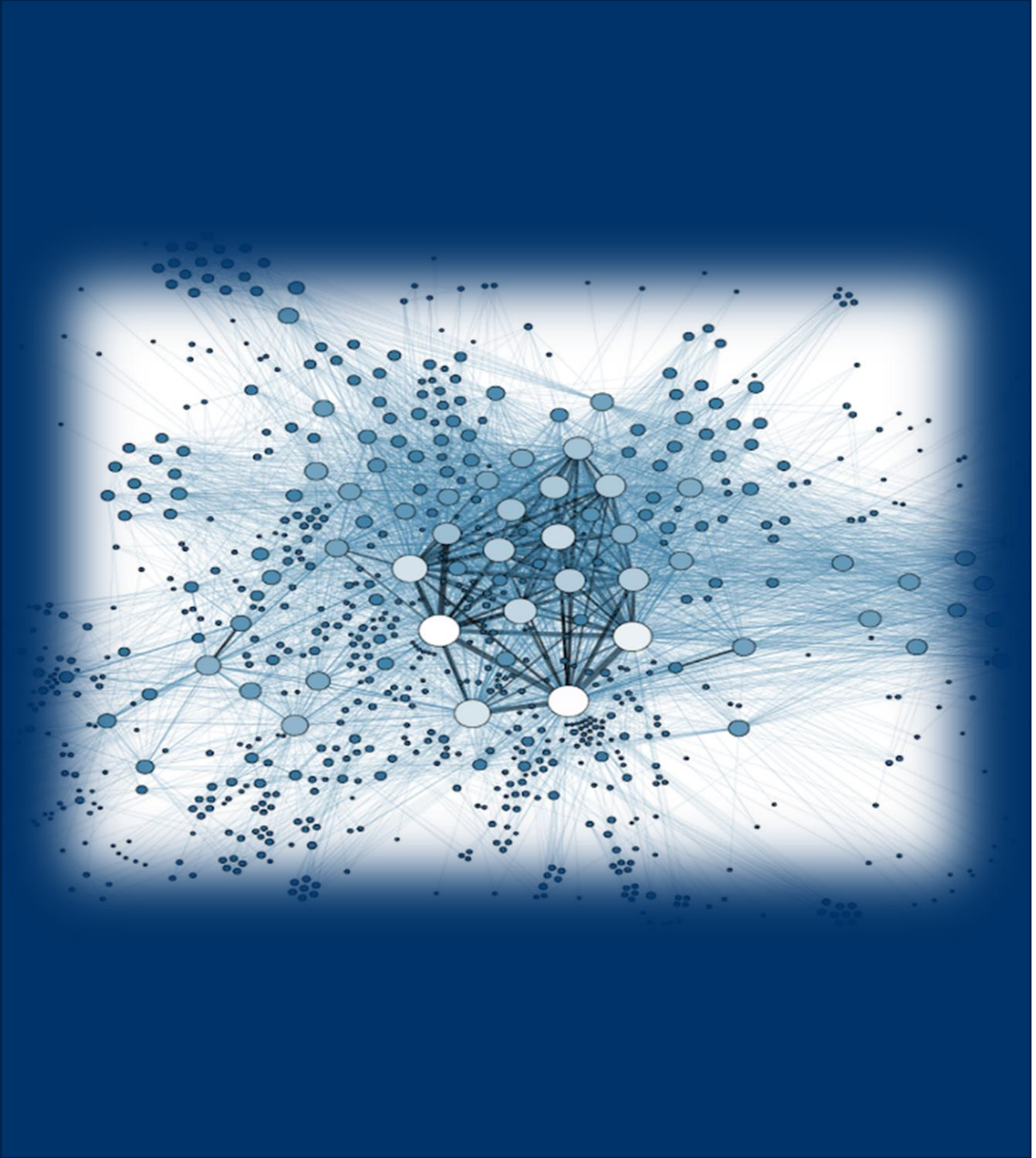


```
@prefix ex: <http://example.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sh: <http://www.w3.org/ns/shacl#> .
@prefix owl:   <http://www.w3.org/2002/07/owl#> .
@prefix xsd:   <http://www.w3.org/2001/XMLSchema#> .

ex:DeveloperShape a sh:NodeShape ;
    sh:targetClass ex:Developer ;
    sh:property [
        sh:path ex:hasName ;
        sh:datatype xsd:string ;
        sh:pattern "[A-Z]" ;
        sh:minCount 1;
        sh:maxCount 1
    ] ;
    sh:property [
        sh:path ex:likes ;
        sh:class ex:DatabaseProduct ;
        sh:maxCount 1
    ] ;
    sh:property [
        sh:path ex:FriendOf ;
        sh:class ex:Developer ;
        sh:targetClass ex:Developer;
#         sh:maxCount 1
    ] .
```

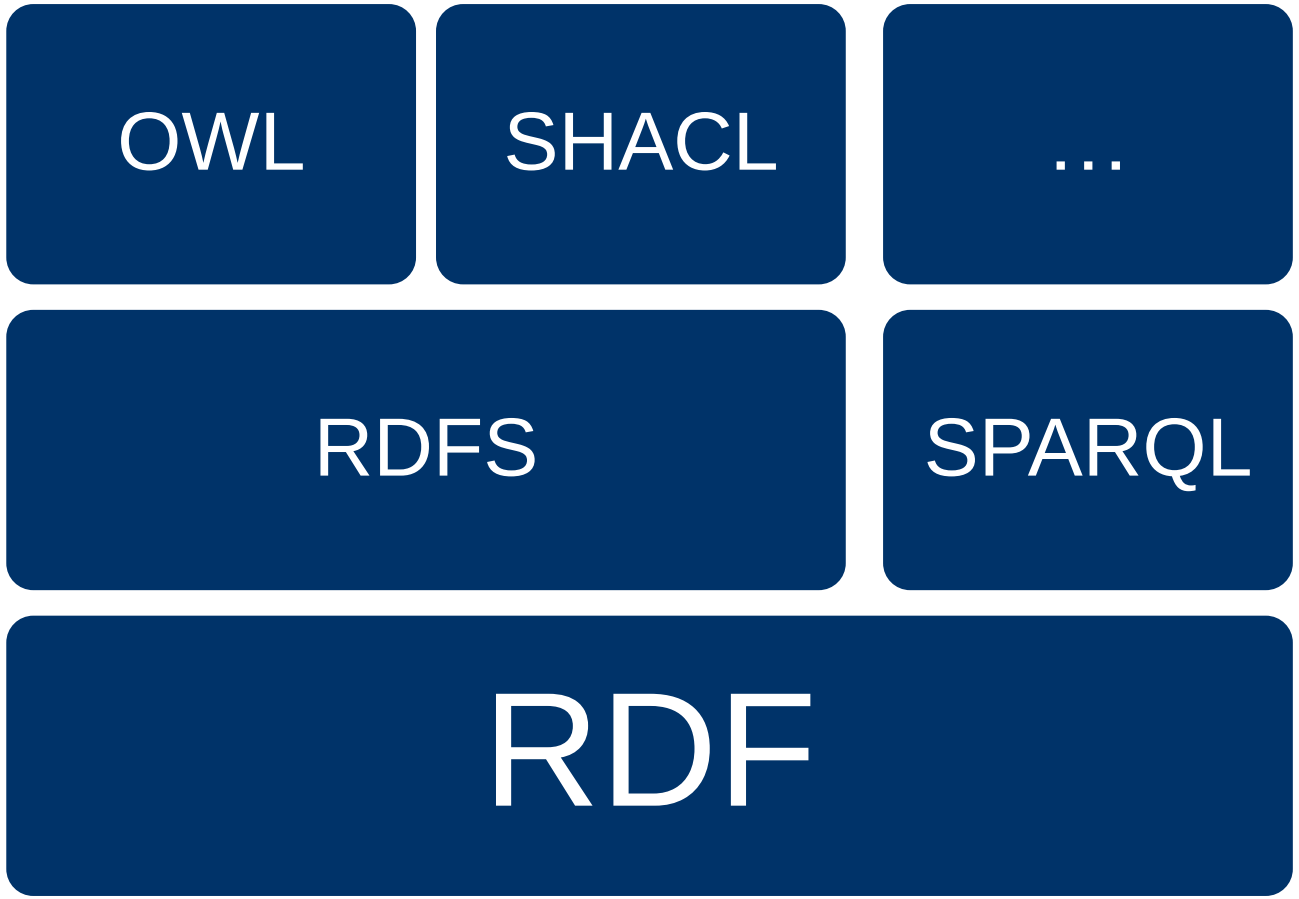


Demo



1. Intro
2. Graphs
3. Knowledge Graphs
4. Implementation: RDF & SPARQL
5. Structure with RDFS & OWL
6. The closed world of SHACL
7. **Wrap Up**

The Semantic Web Stack



https://www.w3.org/TR/rdf11-concepts/#dfn-concrete-rdf-syntax

Public sources

Open Data Sets
- DBPedia
- https://www.govdata.de
- http://www.geonames.org/
- …

Industry Vocabularies
- FIBO for financial services
- Friend of a Friend (FOAF) for social networking
- Schema.org for general business.
- …

Technical Vocabularies
- Dublin Core for metadata
- Prov-O for provenance data
- Simple Knowledge Organization System (SKOS) for knowledge organization
- Schema.org for general business.
- …

Open



Versus



Closed

Back to the AI Hype Cycle

1. **Complement Statistical Reasoning**
2. **Contextual information as base for LLMs**
3. **Graph RAG**

Further ressources

https://w3c.github.io/rdf-primer/spec/

https://kgbook.org/

https://dbpedia.org/sparql

https://www.michaeldebellis.com/post/new-protege-pizza-tutorial

https://www.michaeldebellis.com/post/owlvspropgraphs

https://spinrdf.org/shacl-and-owl.html

https://learn.microsoft.com/en-us/azure/cosmos-db/gen-ai/cosmos-ai-graph

5. Further ressources

SHACL:

https://rdfshape.weso.es/shaclValidate

https://book.validatingrdf.com/bookHtml011.html

https://archive.topquadrant.com/technology/shacl/tutorial/

https://archive.topquadrant.com/technology/shacl/tutorial-sparql-constraints/

Feedback please !

Knowledge Graphs



SQL
KONFERENZ
2025

