Vadalog:

Introduction, Extensions and Business Applications

Emanuel Sallinger

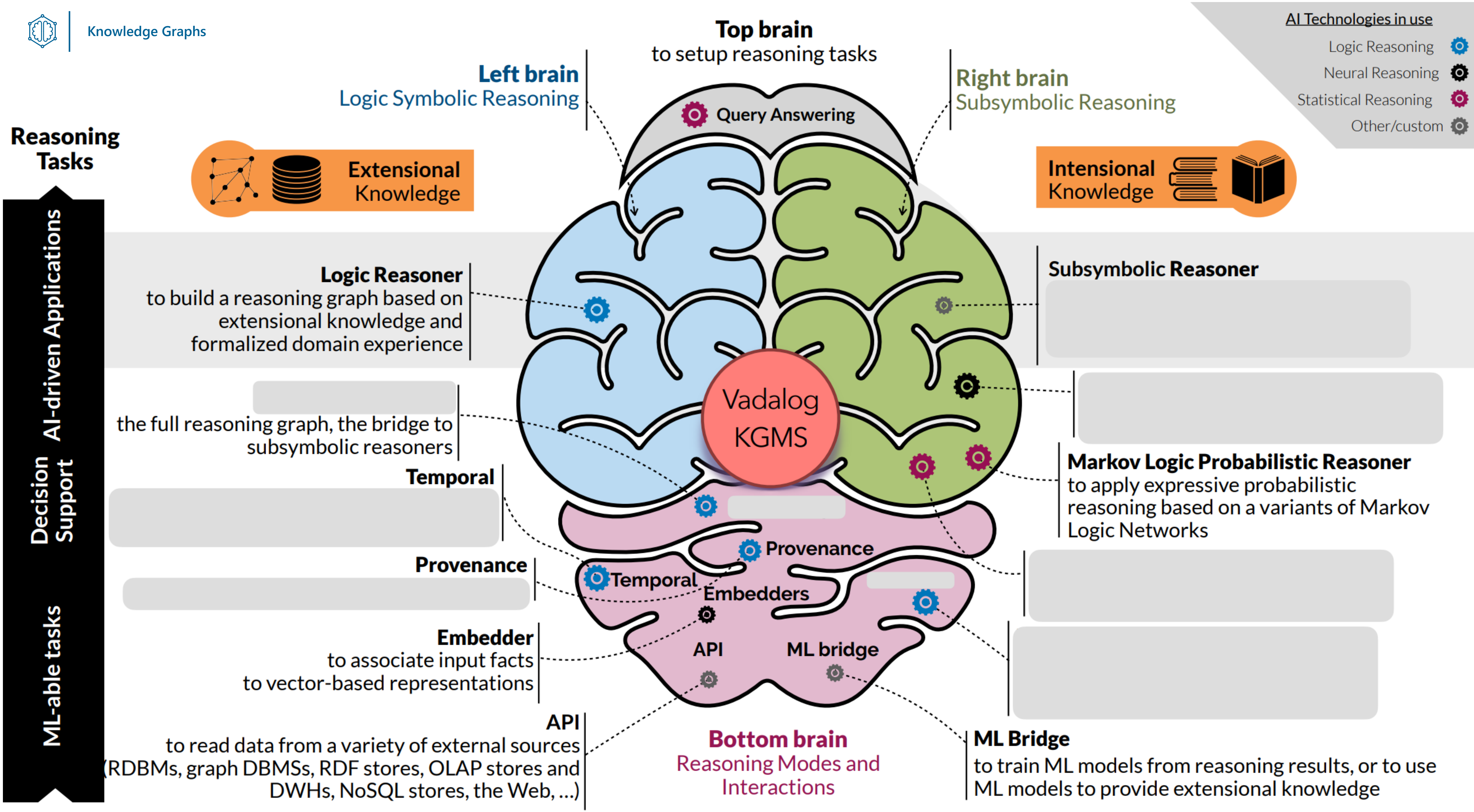
TU WIEN

Knowledge Graph Lab

UNIVERSITY OF OXFORD

The 18th Reasoning Web Summer School
Declarative AI
30 September 2022

WWTF


Top brain
to setup reasoning tasks
Left brain
Logic Symbolic Reasoning
Right brain
Subsymbolic Reasoning
AI Technologies in use
Logic Reasoning
Neural Reasoning
Statistical Reasoning
Other/custom
Reasoning Tasks
AI-driven Applications
Decision Support
ML-able tasks
Extensional Knowledge
Intensional Knowledge
Query Answering
Logic Reasoner
to build a reasoning graph based on extensional knowledge and formalized domain experience
the full reasoning graph, the bridge to subsymbolic reasoners
Subsymbolic Reasoner
Vadalog KGMS
Markov Logic Probabilistic Reasoner
to apply expressive probabilistic reasoning based on a variants of Markov Logic Networks
Temporal
Provenance
Embedders
API
ML bridge
Embedder
to associate input facts to vector-based representations
API
to read data from a variety of external sources (RDBMs, graph DBMSs, RDF stores, OLAP stores and DWHs, NoSQL stores, the Web, ...)
Bottom brain
Reasoning Modes and Interactions
ML Bridge
to train ML models from reasoning results, or to use ML models to provide extensional knowledge

2016–2019 | Selected Highlights

AMW 2016 CIUDAD DE PANAMÁ

LANGUAGE

Data Wrangling for Big Data: Towards a Lingua Franca for Data Wrangling

IJCAI-17 MELBOURNE

LOGIC

Swift Logic for Big Data and Knowledge Graphs

SIGMOD/PODS 2017 Chicago, Illinois, USA, May 14-19

PROJECT

The VADA Architecture for Cost-Effective Data Wrangling

MEDI 2018

DATA SCIENCE

Data Science with Vadalog: Bridging Machine Learning and Reasoning

VLDB2018

SYSTEM

The Vadalog System: Datalog-based Reasoning for Knowledge Graphs

Journal of Big Data 2019

PROJECT ARCHITECTURE

VADA: an architecture for end user informed data preparation

SIGMOD/PODS AMSTERDAM 2019

SPACE EFFICIENCY

The Space-Efficient Core of Vadalog

ICDE MACAU 2019

ENTERPRISE AI

Knowledge Graphs and Enterprise AI: The Promise of an Enabling Technology

Datalog 2.0 2019

RECOMMENDER SYSTEMS

Feature Engineering and Explainability with Vadalog: A Recommender Systems Application

2020 Selected Highlights

EMERGENCY RESPONSE
(not peer-reviewed)

COVID-19: LOCKDOWN

COVID-19 and Company Knowledge Graphs: Assessing Golden Powers and Economic Impact of Selective Lockdown via AI Reasoning

EDBT ICDT COPENHAGEN 2020

ENTERPRISE AI IN PRACTICE

Weaving Enterprise Knowledge Graphs: The Case of Company Ownership Graphs

Declarative AI 2020

COVID-19: TAKEOVERS

Reasoning on Company Takeovers during the COVID-19 Crisis with Knowledge Graphs

Declarative AI 2020

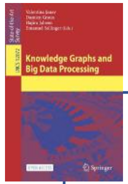
PROBABILISTIC

Reasoning Under Uncertainty in Knowledge Graphs

Declarative AI 2020

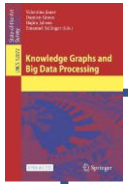
MONEY LAUNDERING

Rule-based Anti-Money Laundering in Financial Intelligence Units: Experience and Vision



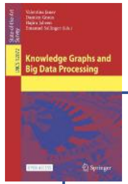
KNOWLEDGE GRAPHS

Knowledge Graphs: The Layered Perspective



BOOK

Knowledge Graphs and Big Data Processing



KG EMBEDDINGS

Reasoning in Knowledge Graphs: An Embeddings Spotlight

2021 Selected Highlights

EDBT ICDT NICOSIA 2021

CONFIDENTIALITY

Financial Data Exchange with Statistical Confidentiality: A Reasoning-based Approach

ICDE 2021 VIRTUAL CONFERENCE GREECE

COMPANY CONTROL

Distributed Company Control in Company Shareholding Graphs

Declarative AI 2021

TEMPORAL

Monotonic Aggregation for Temporal Datalog

Declarative AI 2021

HARMFUL JOINS

Eliminating Harmful Joins in Warded Datalog+/-.

Declarative AI 2021

INDUSTRIAL BLOCKCHAIN

Rule-based Blockchain Knowledge Graphs: Declarative AI for Solving Industrial Blockchain Challenges

ECAI 2020 Santiago de Compostela

JOINS

Traversing Knowledge Graphs with Good Old (and New) Joins

ECAI 2020 Santiago de Compostela

HYBRID AI

Augmenting Logic-based Knowledge Graphs: The Case of Company Graphs

ECAI 2020 Santiago de Compostela

BLOCKCHAIN VISION

Blockchains as Knowledge Graphs - Blockchains for Knowledge Graphs

Knowledge Graphs

People and Groups

Emanuel Sallinger

Prof. Dr.

Emanuel Sallinger

Knowledge Graph Lab

University of **Oxford**

Knowledge Graph Lab

Formally:
Vienna Science and Technology Fund (WWTF)
Vienna Research Group (VRG) on
Scalable Reasoning in Knowledge Graphs (VRG18-013)

TU WIEN Informatics

TU WIEN Institute of Logic and Computation

dbai Database and Artificial Intelligence Group

WWTF VIENNA SCIENCE AND TECHNOLOGY FUND

Center for AI and ML
SIG Knowledge Graphs

Joint Knowledge Graph Labs

ROMA TRE UNIVERSITÀ DEGLI STUDI

UNIVERSITY OF OXFORD

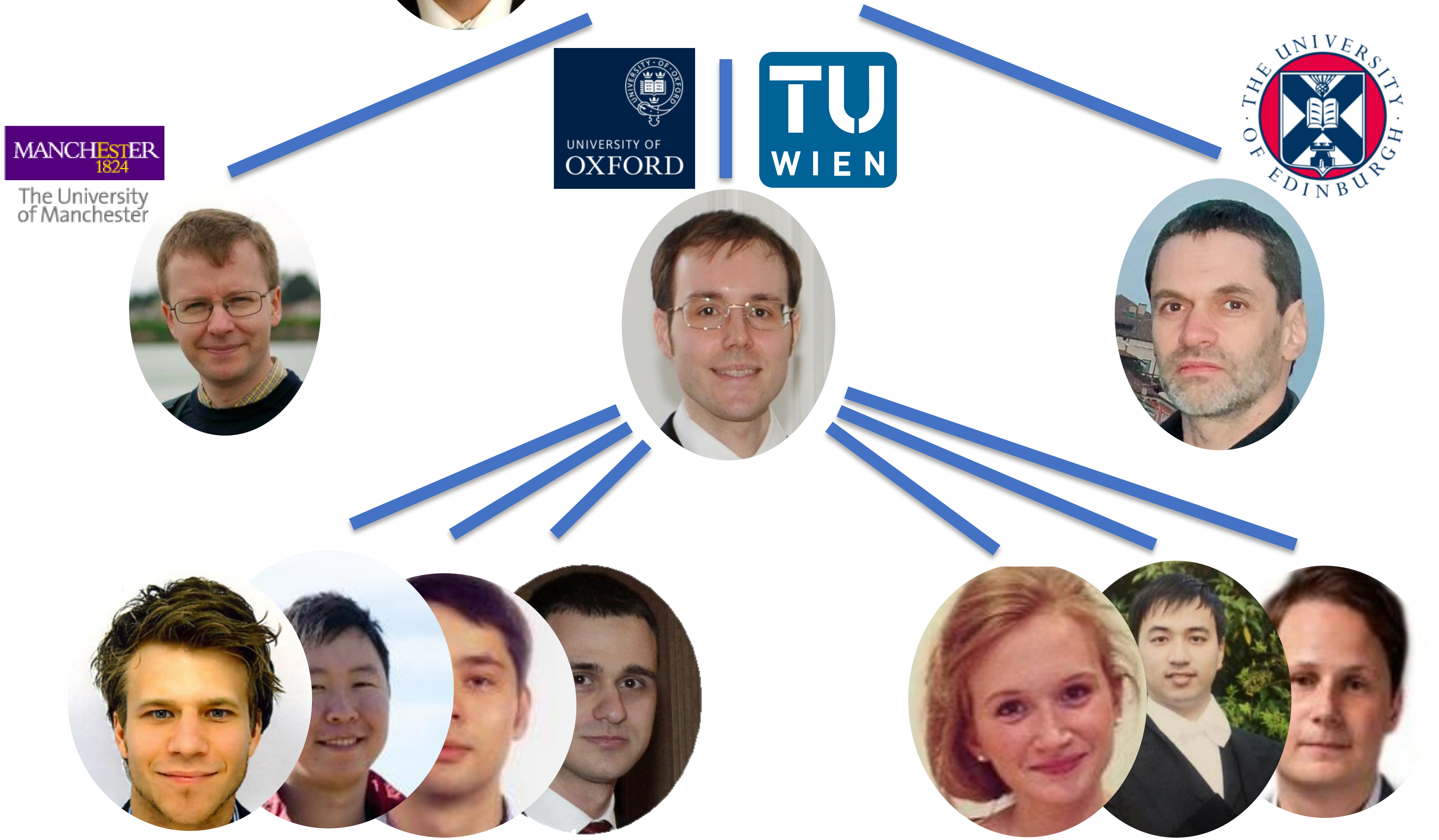
TU WIEN

BANCA D'ITALIA EUROSISTEMA

ART Applied Research Team

POLITECNICO DI MILANO

VADA
Value-Added Data



Banking and Finance

BANCA D'ITALIA
EUROSISTEMA

THE WORLD BANK

Manufacturing

BOEING

Miele

SIEMENS

Logistics

BMW GROUP

HOFER

… and more

bp

U.S. AIR FORCE

Meltwater

Knowledge Graphs

People and Group

Emanuel Sallinger

Knowledge Graphs

Motivation

Emanuel Sallinger

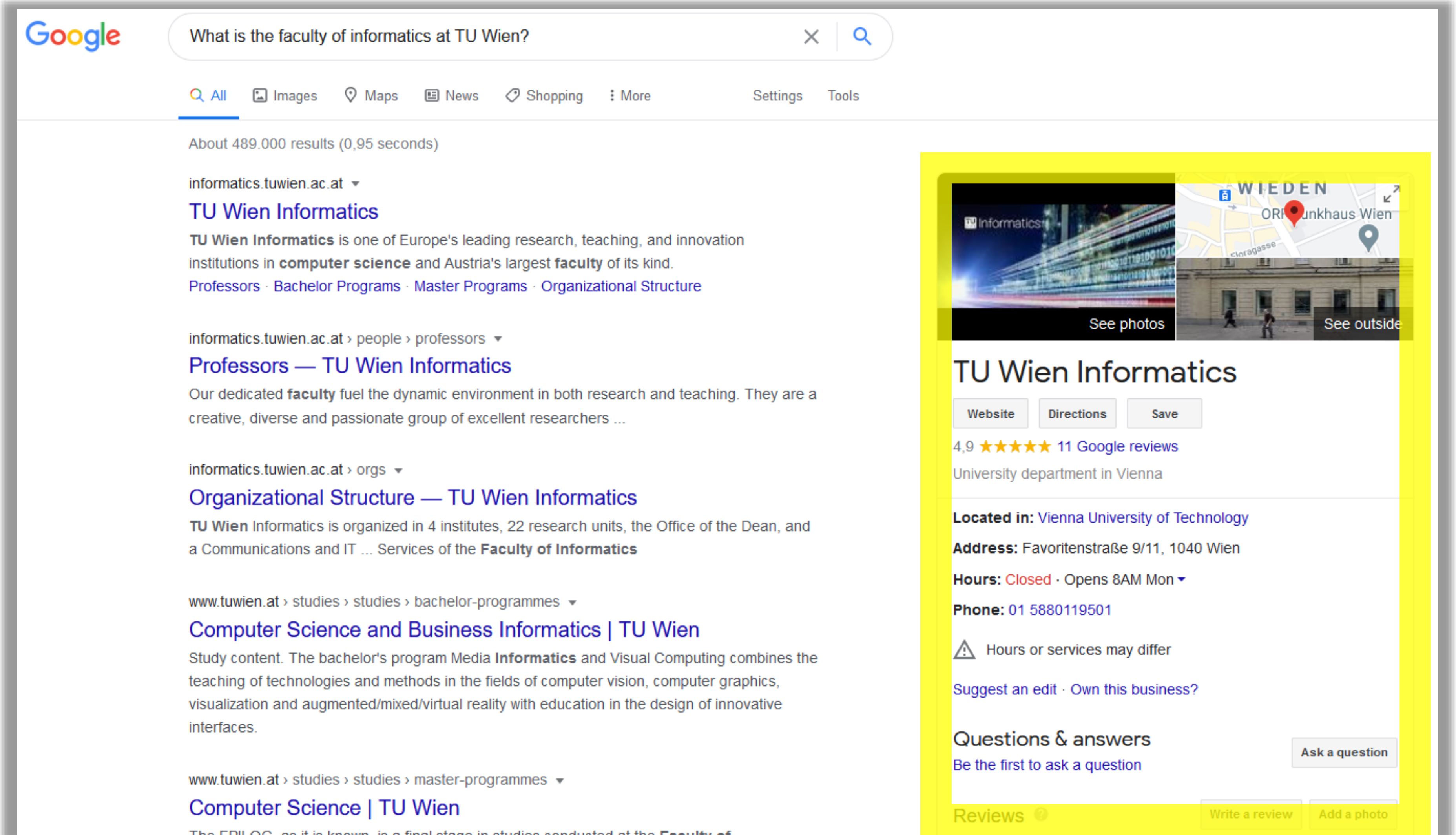
Motivation

1. The **technology** used by Google and others

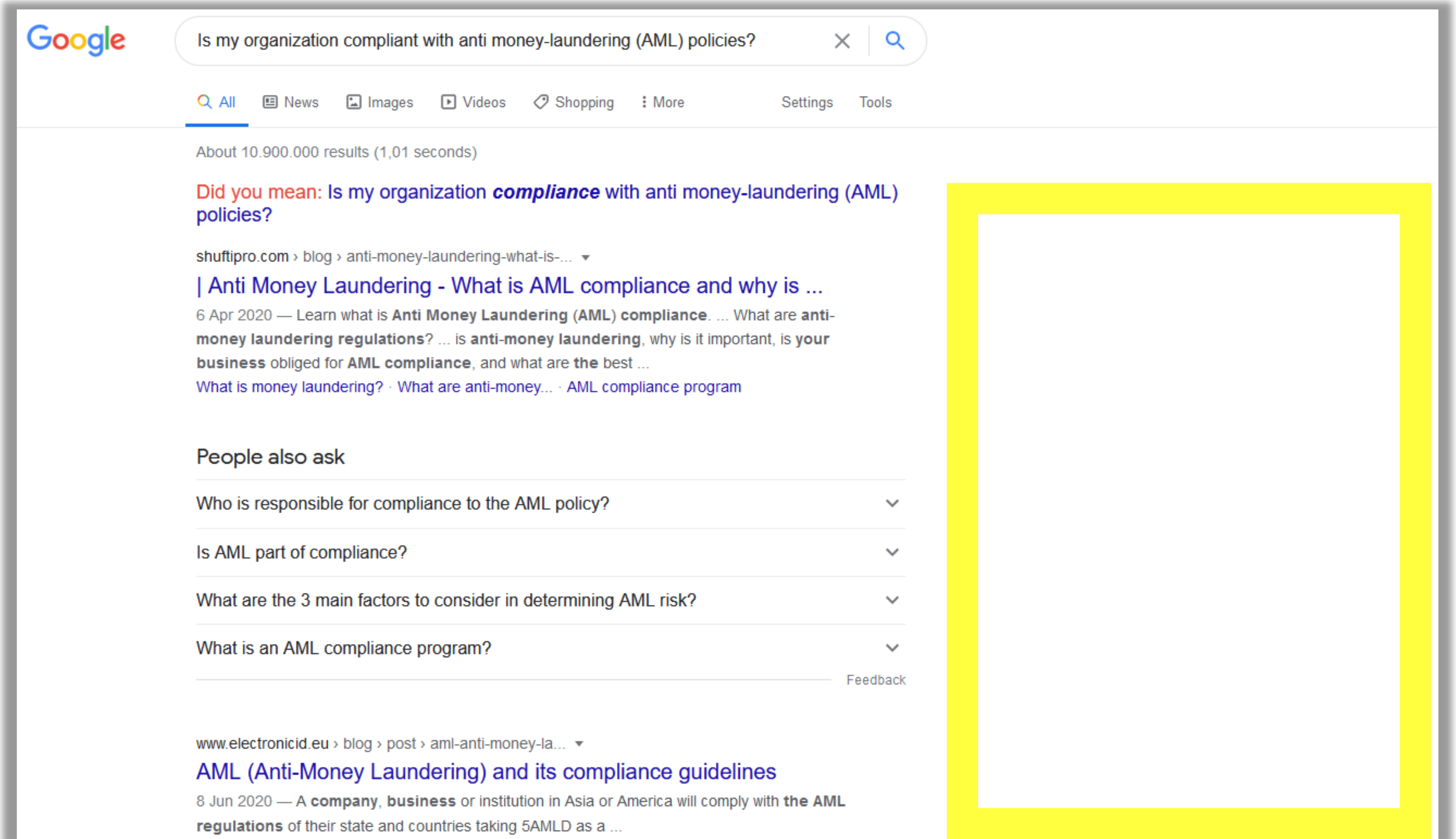
2. A meeting place of **databases**, **data science** and **Artificial Intelligence** research

3. A **skillset** to solve fascinating problems

"What is the faculty of informatics at TU Wien"?



"Is my organization compliant with anti money-laundering (AML) policies?"

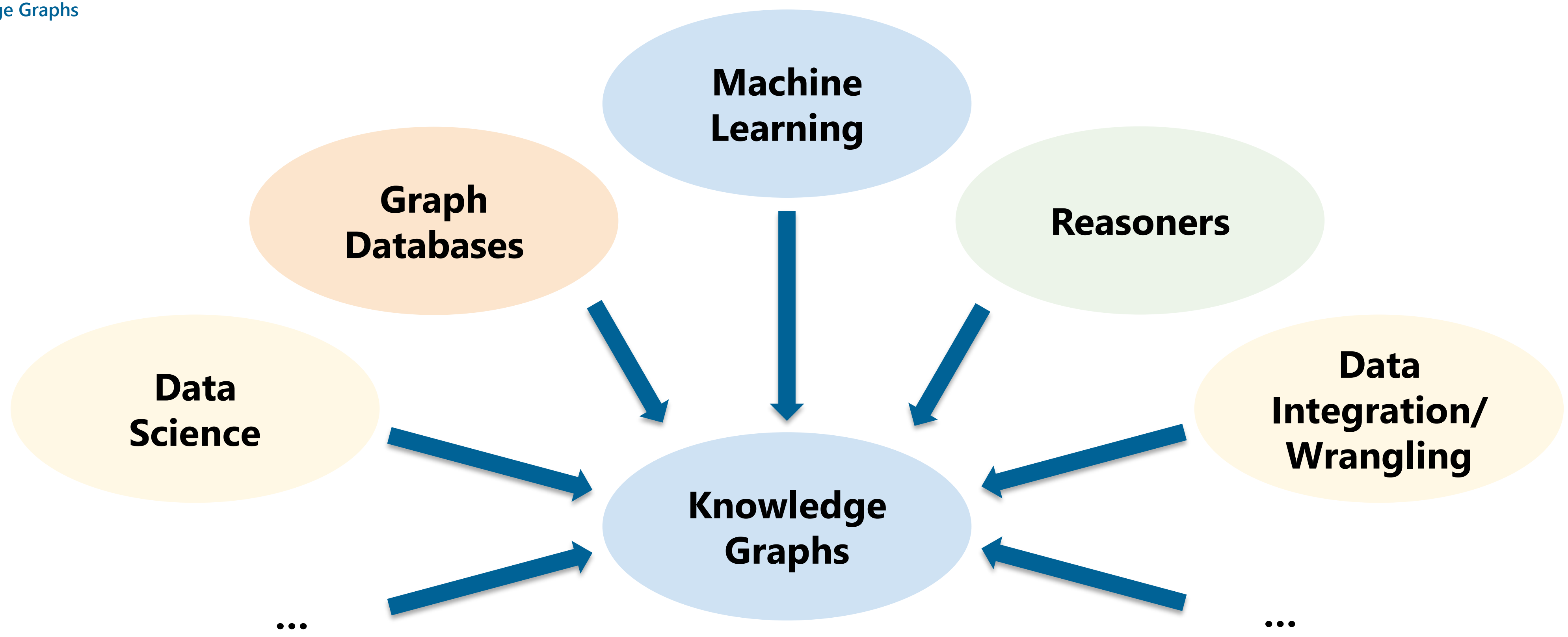


Motivation

1. The **technology** used by Google and others

2. A meeting place of **databases**, **data science** and **artificial intelligence** research

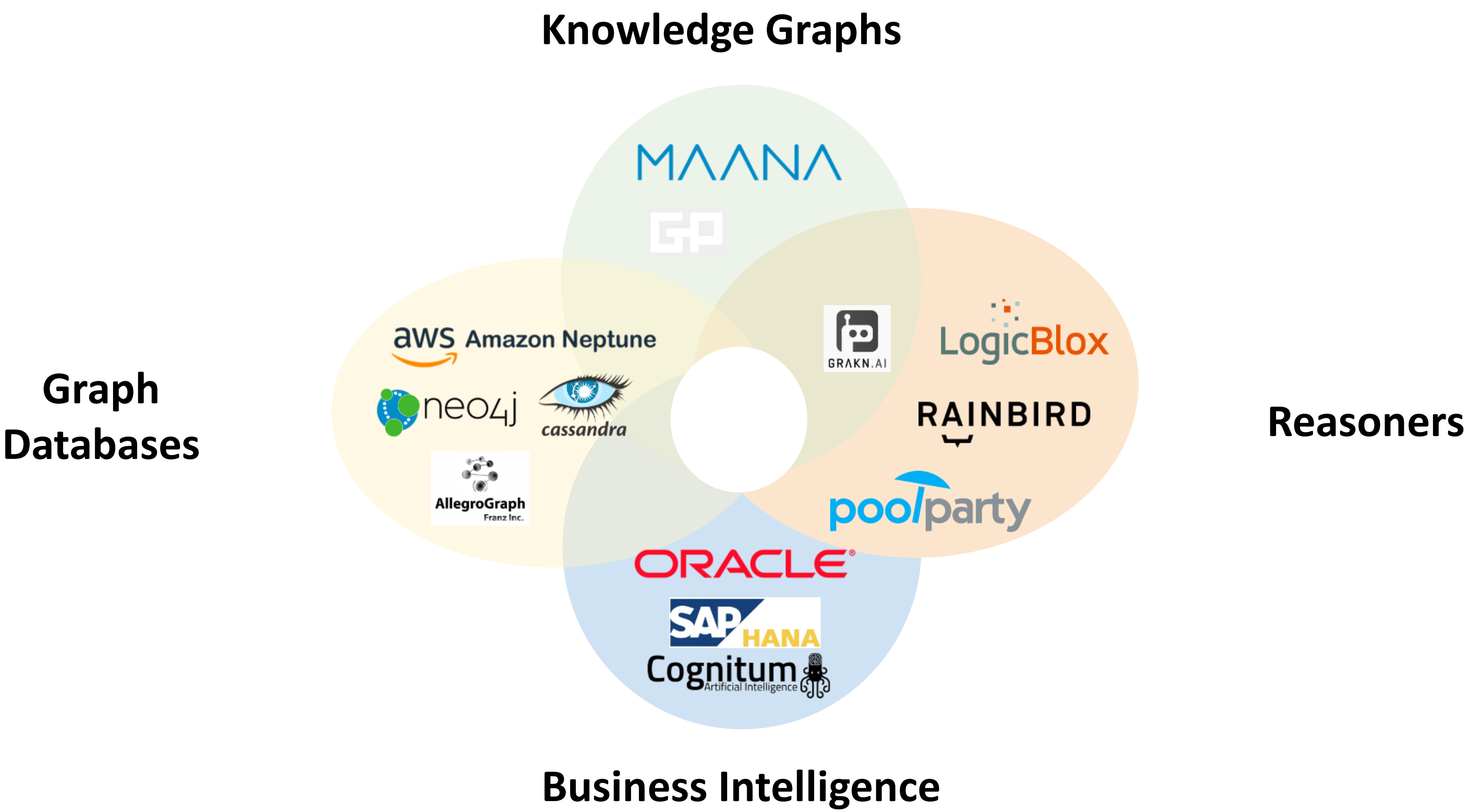
3. A **skillset** to solve fascinating problems



Dgraph ontotext cassandra DeepReason.ai Google Azure Cosmos DB RAINBIRD ...

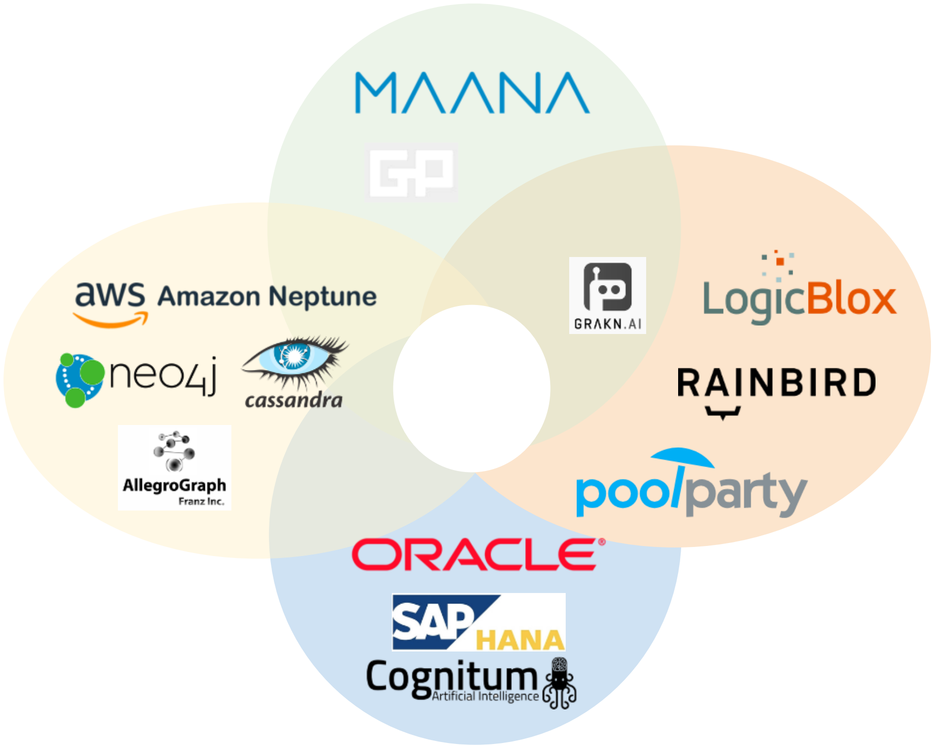
Cognitum Artificial Intelligence neo4j Gephi SAP HANA ORACLE ontop JanusGraph ...

aws Amazon Neptune AllegroGraph Franz Inc. GRAKN.AI MAANA poolparty LogicBlox ...


Knowledge Graphs
Graph Databases
Reasoners
Business Intelligence
MAANA
aws Amazon Neptune
neo4j
cassandra
AllegroGraph
Franz Inc.
GRAKN.AI
LogicBlox
RAINBIRD
poolparty
ORACLE
SAP HANA
Cognitum
Artificial Intelligence

Knowledge Graphs

Graph Databases



Reasoners

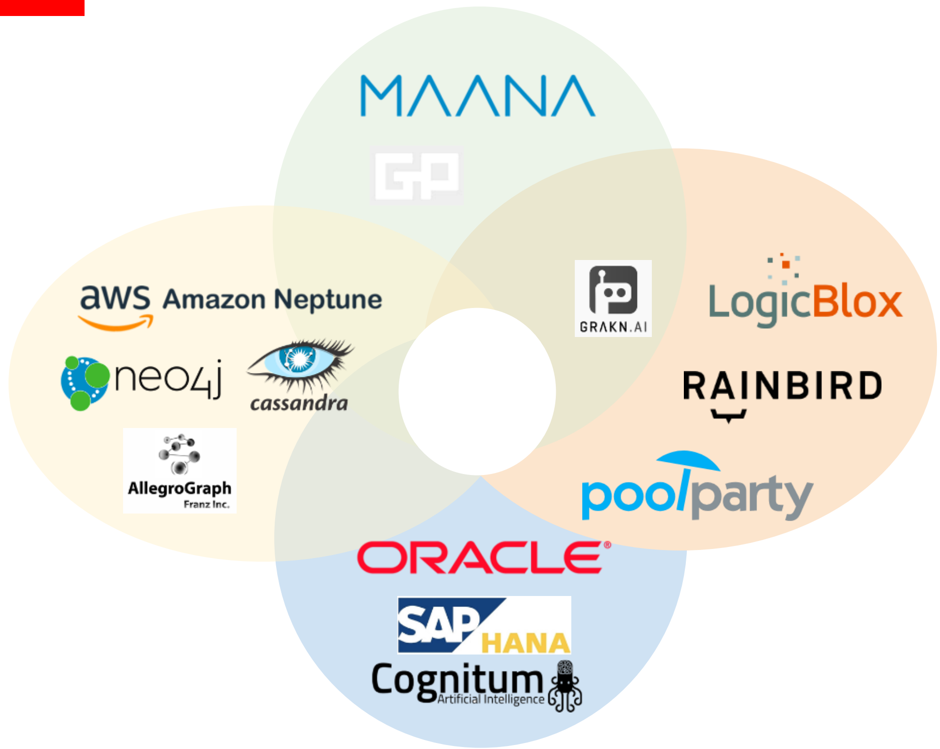
Business Intelligence

ad-hoc

expressive power

Knowledge Graphs

Graph Databases



Reasoners

Business Intelligence

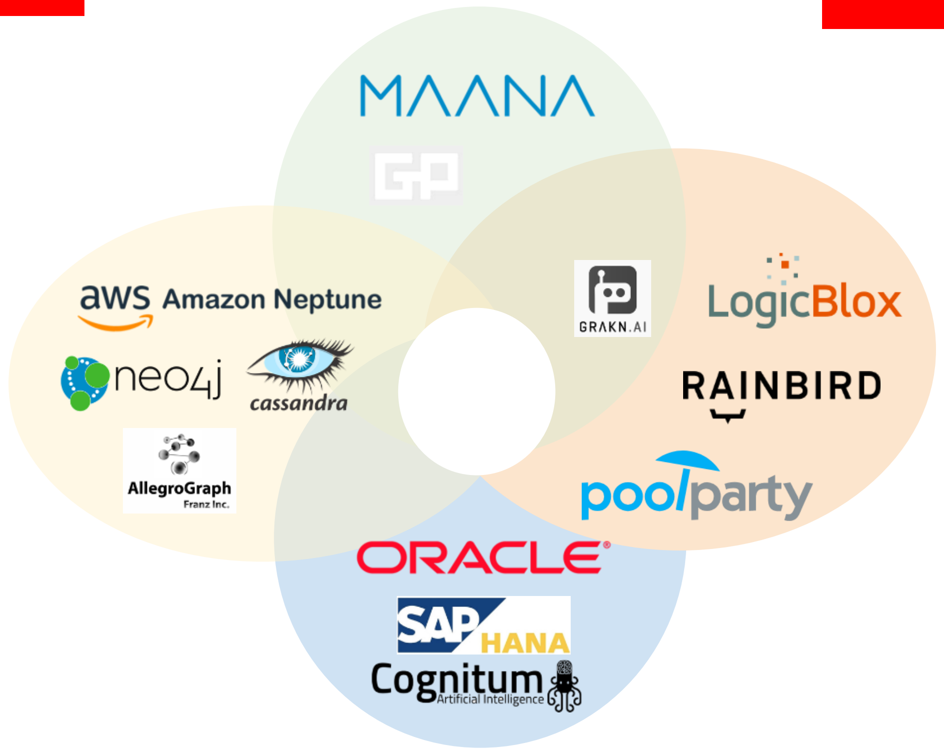
ad-hoc

expressive power

scalability

Knowledge Graphs

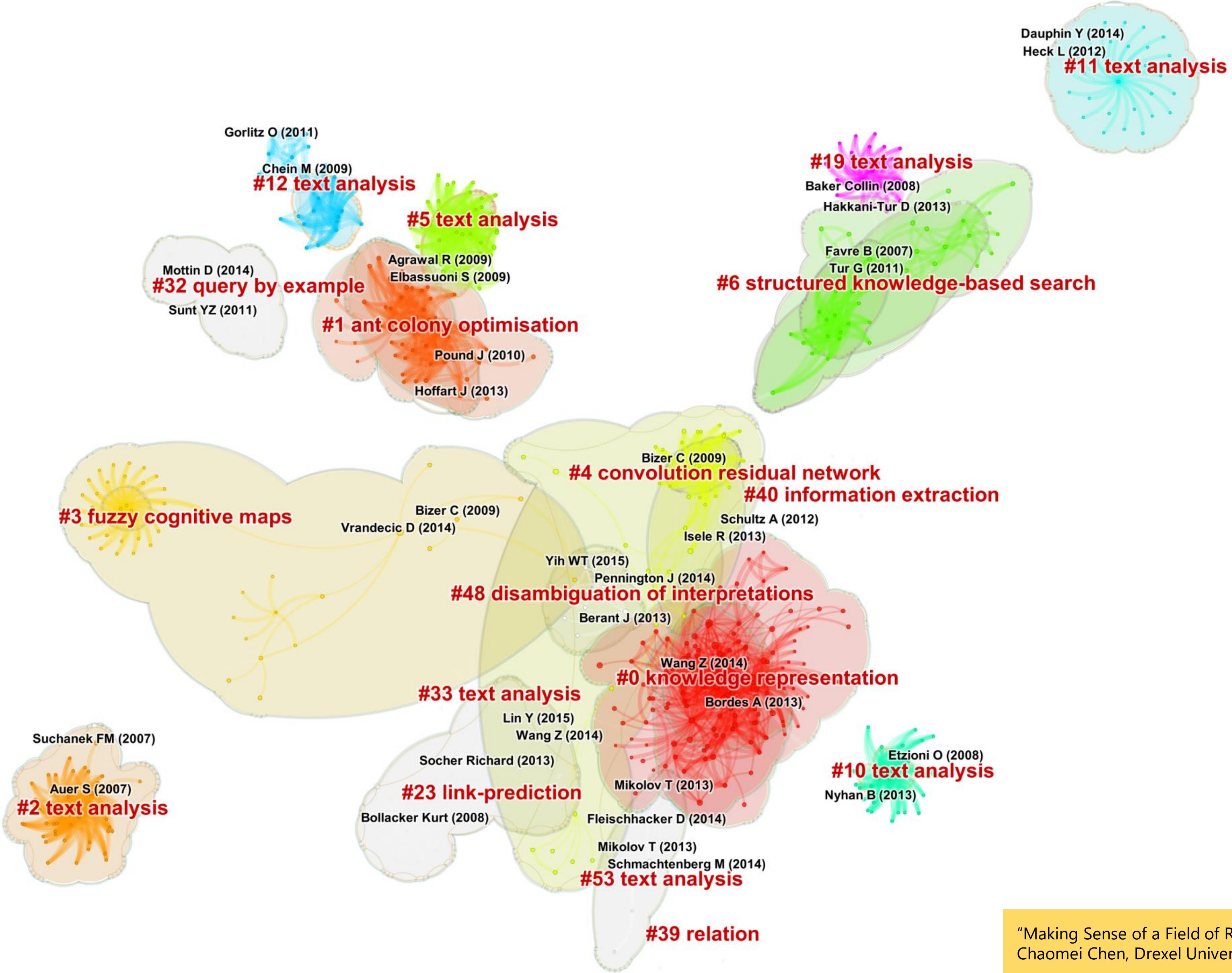
Graph Databases



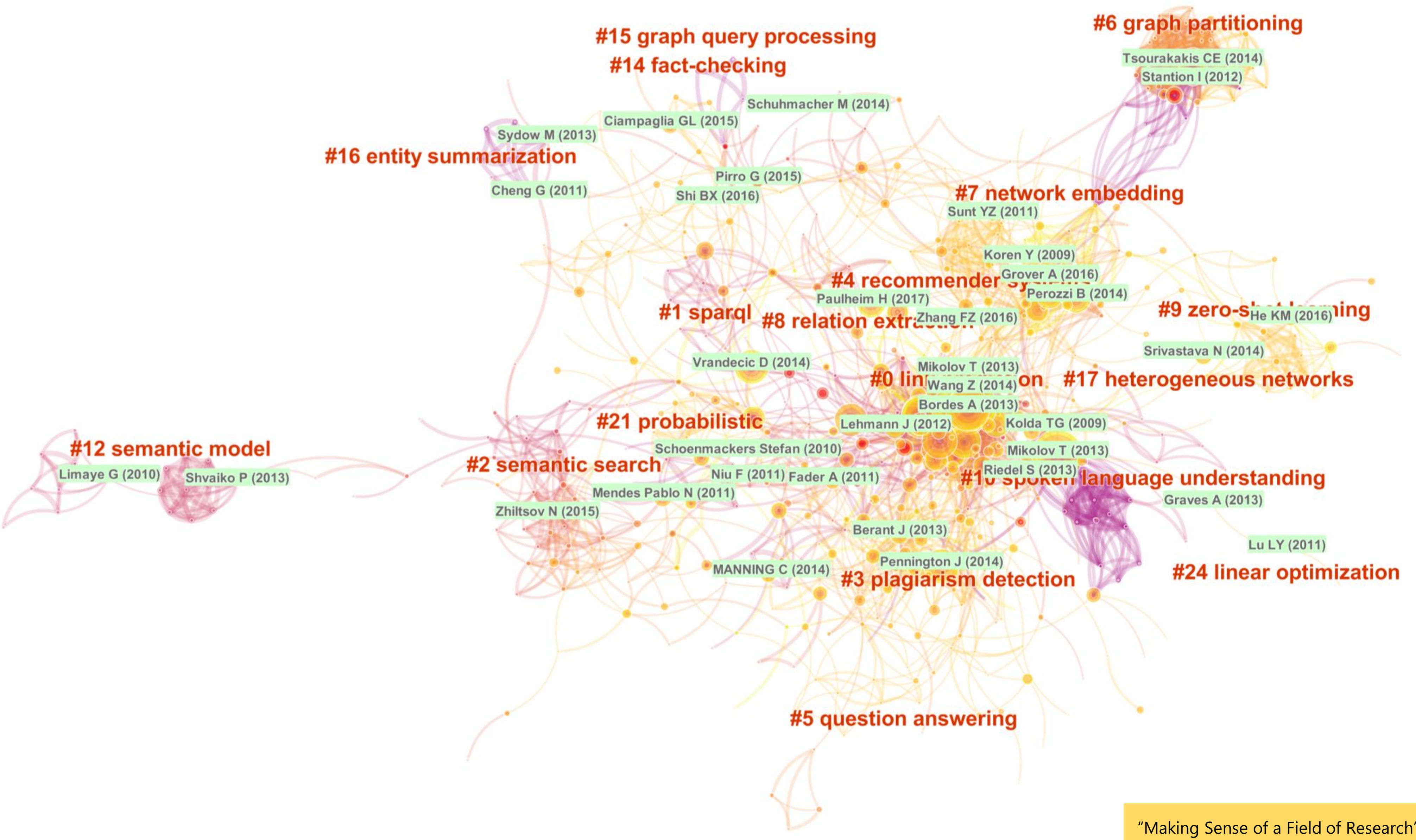
Reasoners

Business Intelligence

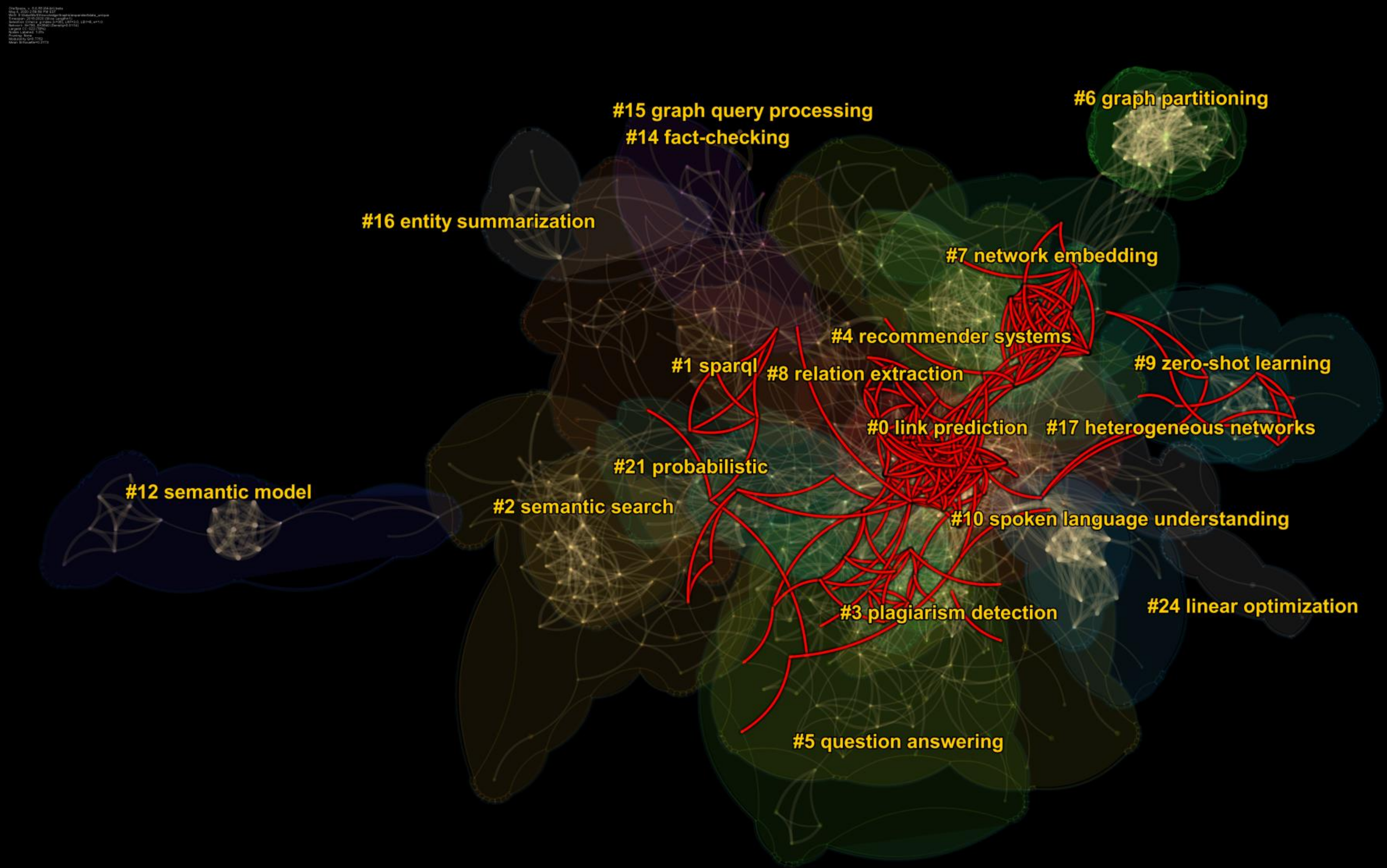
ad-hoc



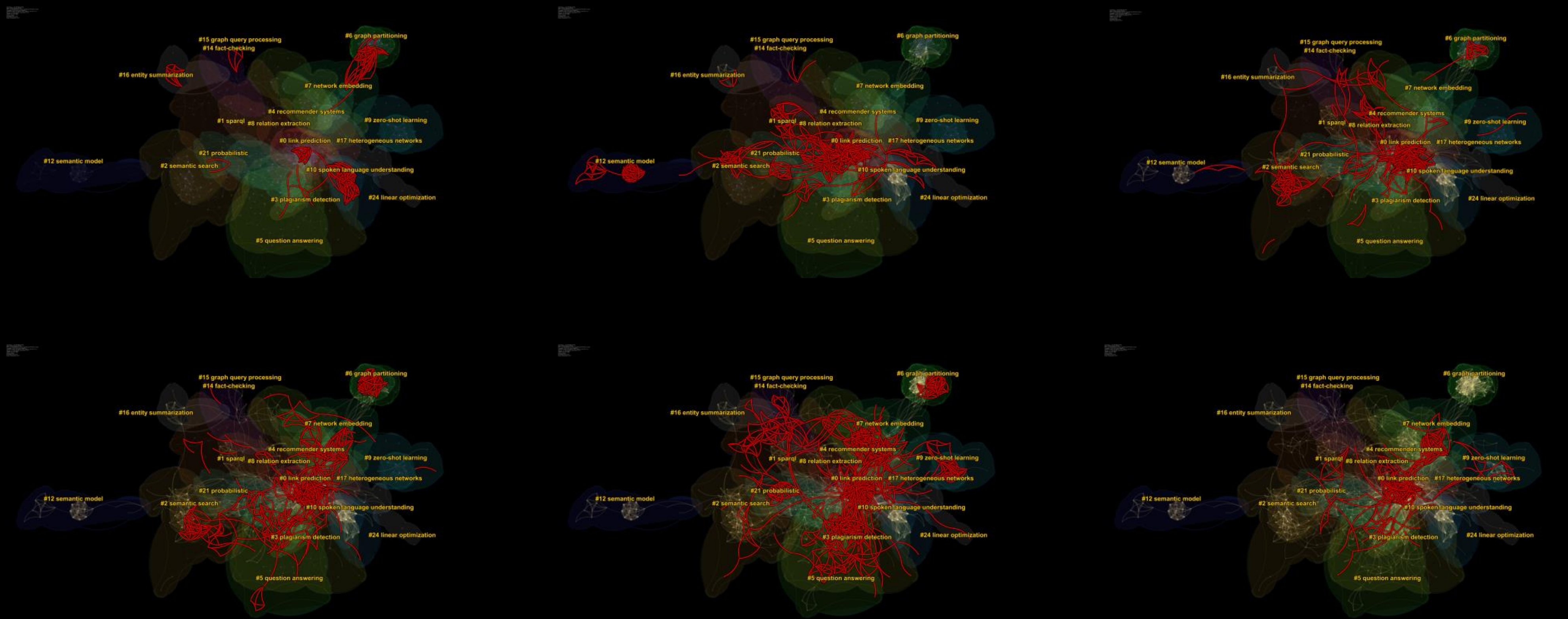
“Making Sense of a Field of Research”
Chaomei Chen, Drexel University, 2020



“Making Sense of a Field of Research”
Chaomei Chen, Drexel University, 2020



“Making Sense of a Field of Research”
Chaomei Chen, Drexel University, 2020



"Making Sense of a Field of Research"
Chaomei Chen, Drexel University, 2020

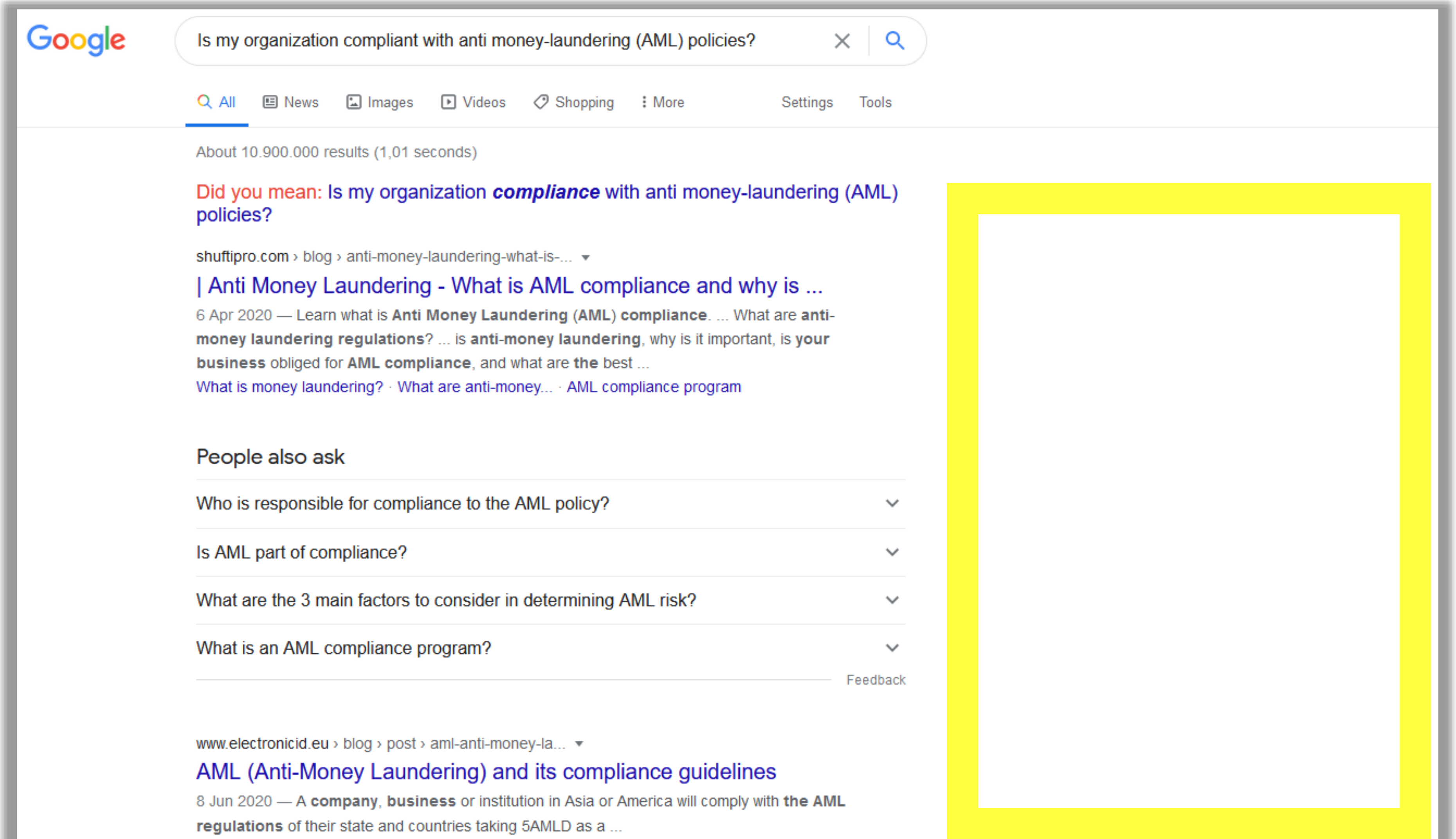
Motivation

1. The **technology** used by Google and others

2. A meeting place of **databases**, **data science** and **artificial intelligence** research

3. A **skillset** to solve fascinating problems

"Is my organization compliant with anti money-laundering (AML) policies?"



FINANCIAL TIMES

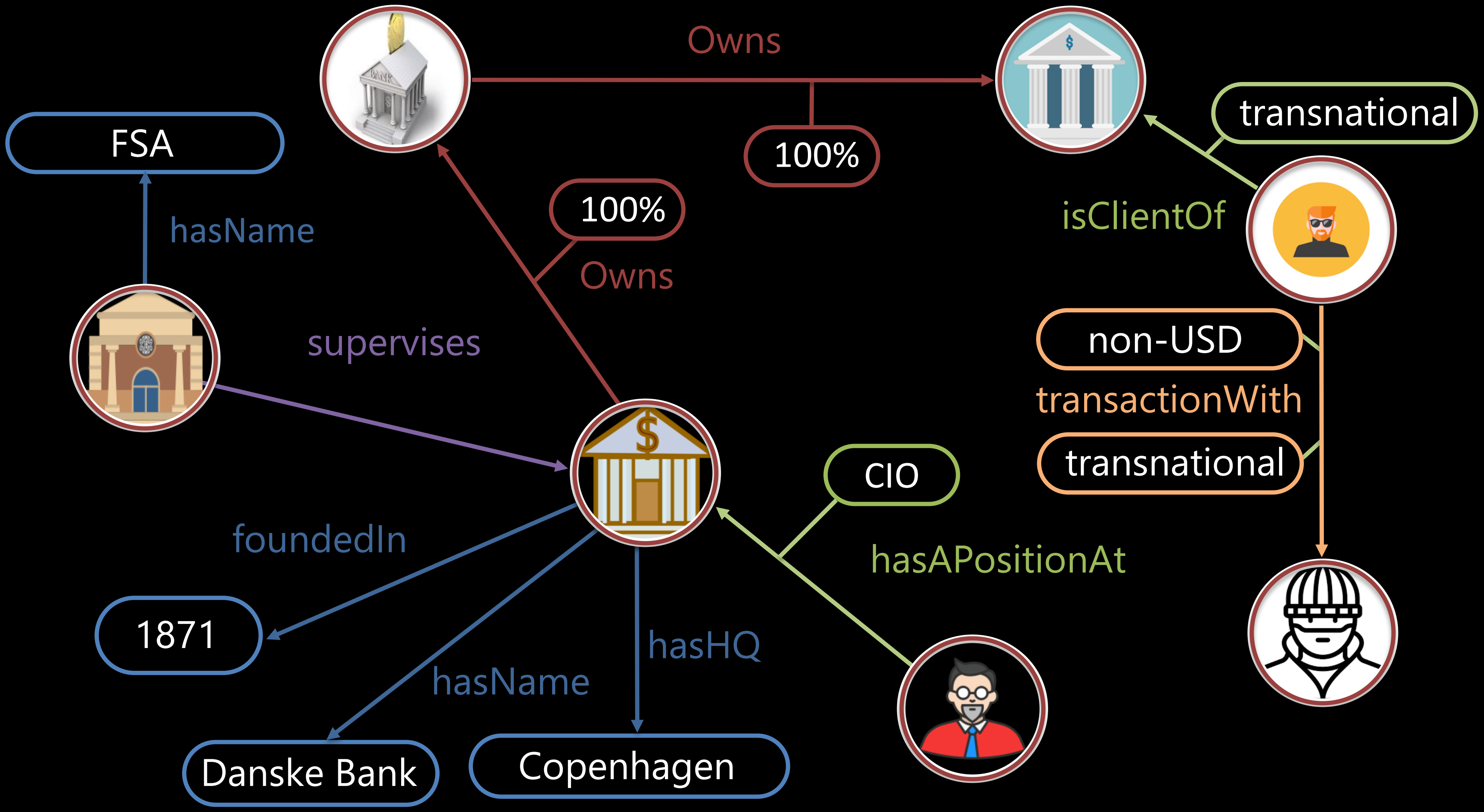
Danske Bank AS + Add to myFT

Danske Bank chairman ousted by main shareholder after scandal

Maersk family brings in new blood to stabilise lender in wake of €200bn money laundering

€200bn money laundering

Ole Andersen will step down as chairman of Danske Bank at an extraordinary general meeting in the next few weeks © Bloomberg


Owns
100%
transnational
FSA
hasName
100%
Owns
isClientOf
supervises
non-USD
transactionWith
transnational
CIO
foundedIn
hasAPositionAt
1871
hasHQ
hasName
Danske Bank
Copenhagen

REUTERS

Business Markets World Politics TV More

BUSINESS NEWS MARCH 26, 2020 / 11:36 AM / 2 MONTHS AGO

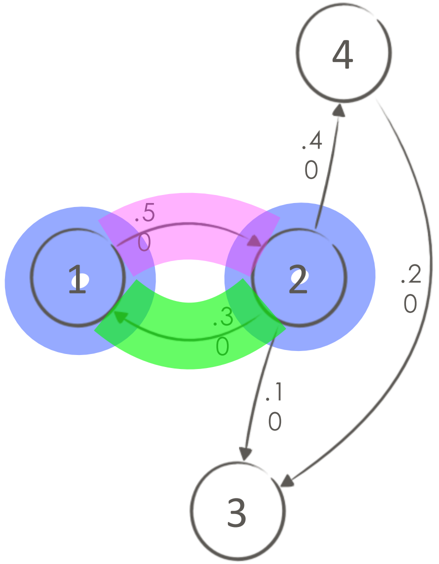
EU leaders to shield strategic firms from hostile interest amid crisis

Francesco Guarascio, Gabriela Baczynska

4 MIN READ

BRUSSELS (Reuters) - European Union leaders will on Thursday back plans to defend healthcare, infrastructure and other firms seen as having strategic role from hostile foreign takeovers, draft EU summit conclusions show.

"Integrated Ownership"



A^2

	1	2	3	4
1	.15		.05	.20
2		.15	.08	
3				
4				

A

	1	2	3	4
1		.50		
2	.30		.10	.40
3				
4			.20	

A^3

	1	2	3	4
1		.075	.04	
2	.045		.015	.06
3				
4				

Facebook KG Social graph with people, places, things

Amazon PG knowledge graph of all products

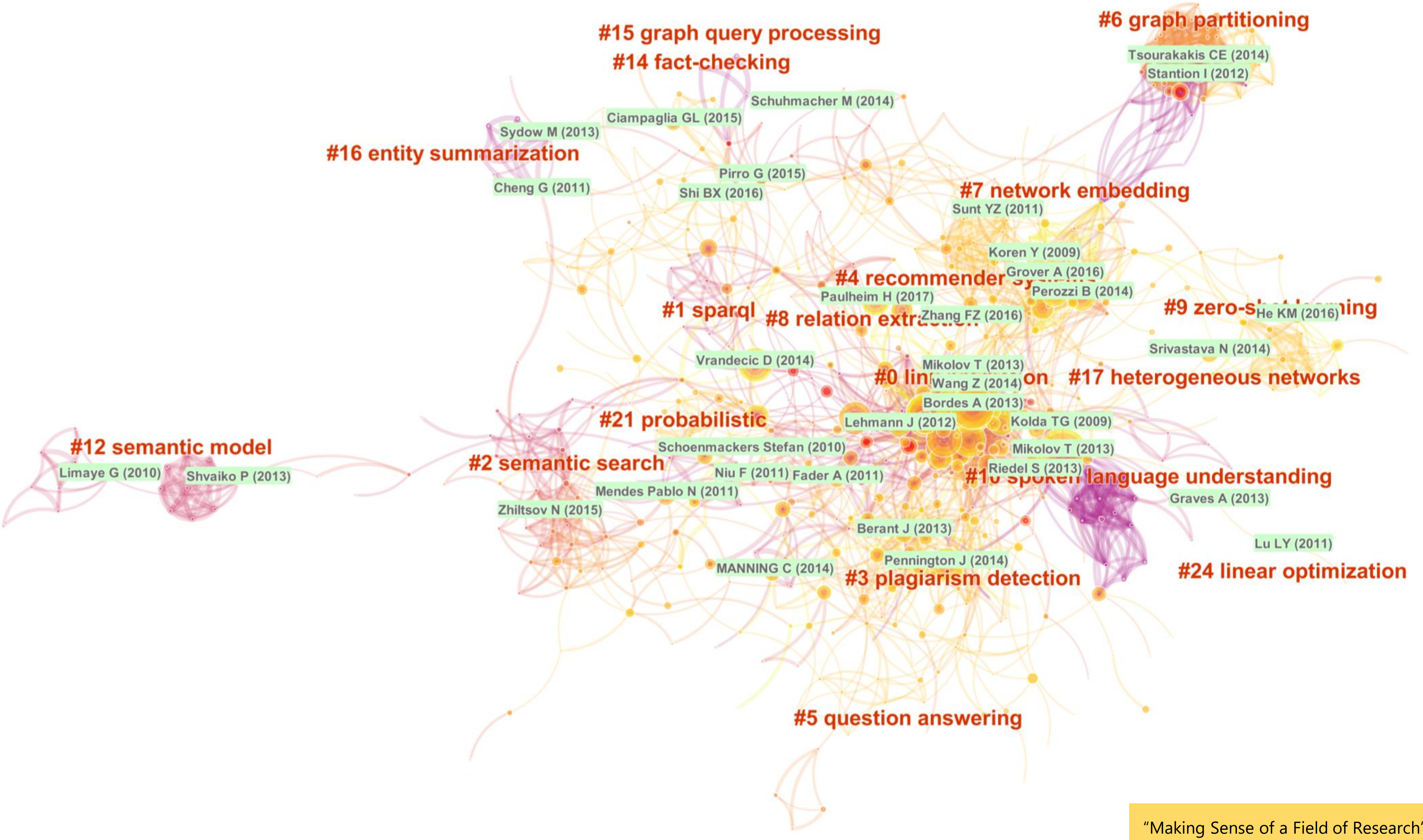
Factual Businesses & places

Wolfram KB World facts + mathematics

RIT People, skills, recruiting

Commercial Banks Customers, Companies, Risks, ...

Rating Agencies Companies, Evaluations, Risks, ...



"Making Sense of a Field of Research"
Chaomei Chen, Drexel University, 2020

Motivation

1. The **technology** used by Google and others

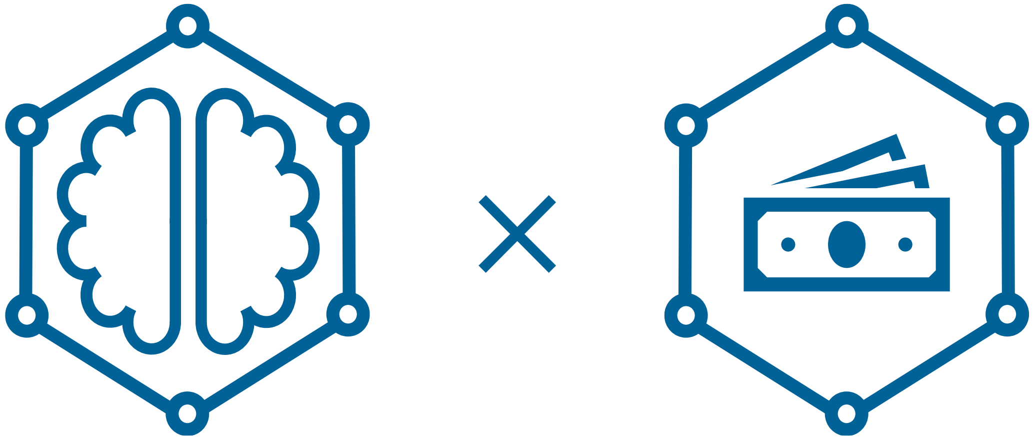
2. A meeting place of **databases**, **data science** and **artificial intelligence** research

3. A **skillset** to solve fascinating problems

Knowledge Graphs

Motivation

Emanuel Sallinger



Overview

Representations

With Financial KGs as a Running Example

Emanuel Sallinger

Representations

KG Embeddings
Widely-applied, large
family of ML models

Logical Knowledge in KGs
Highly expressive, diverse
family of logical models.

Graph Neural Networks
Using the KG structure
as a neural network.

Data Models
Overview of data models
in different communities

FINANCIAL TIMES

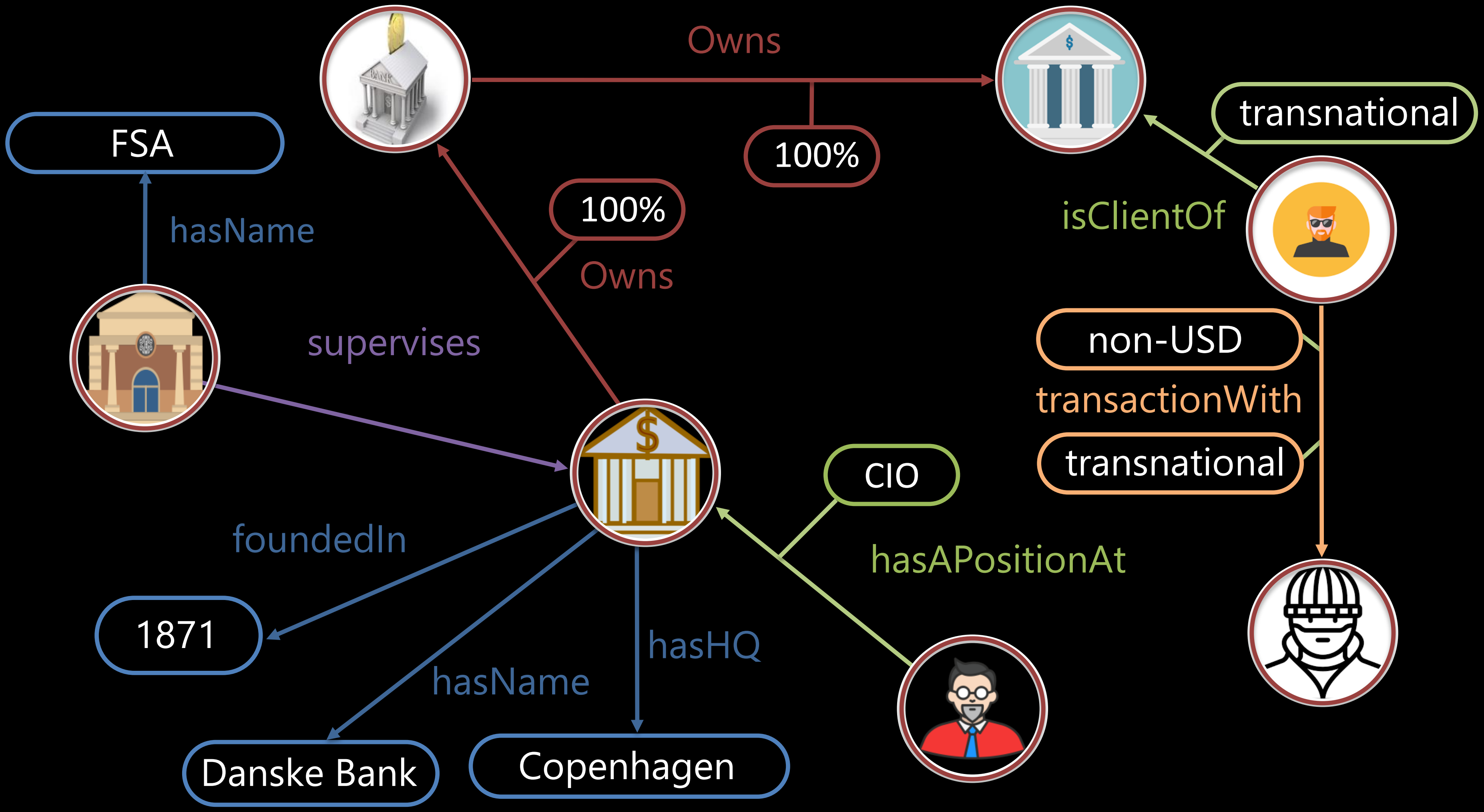
Danske Bank AS + Add to myFT

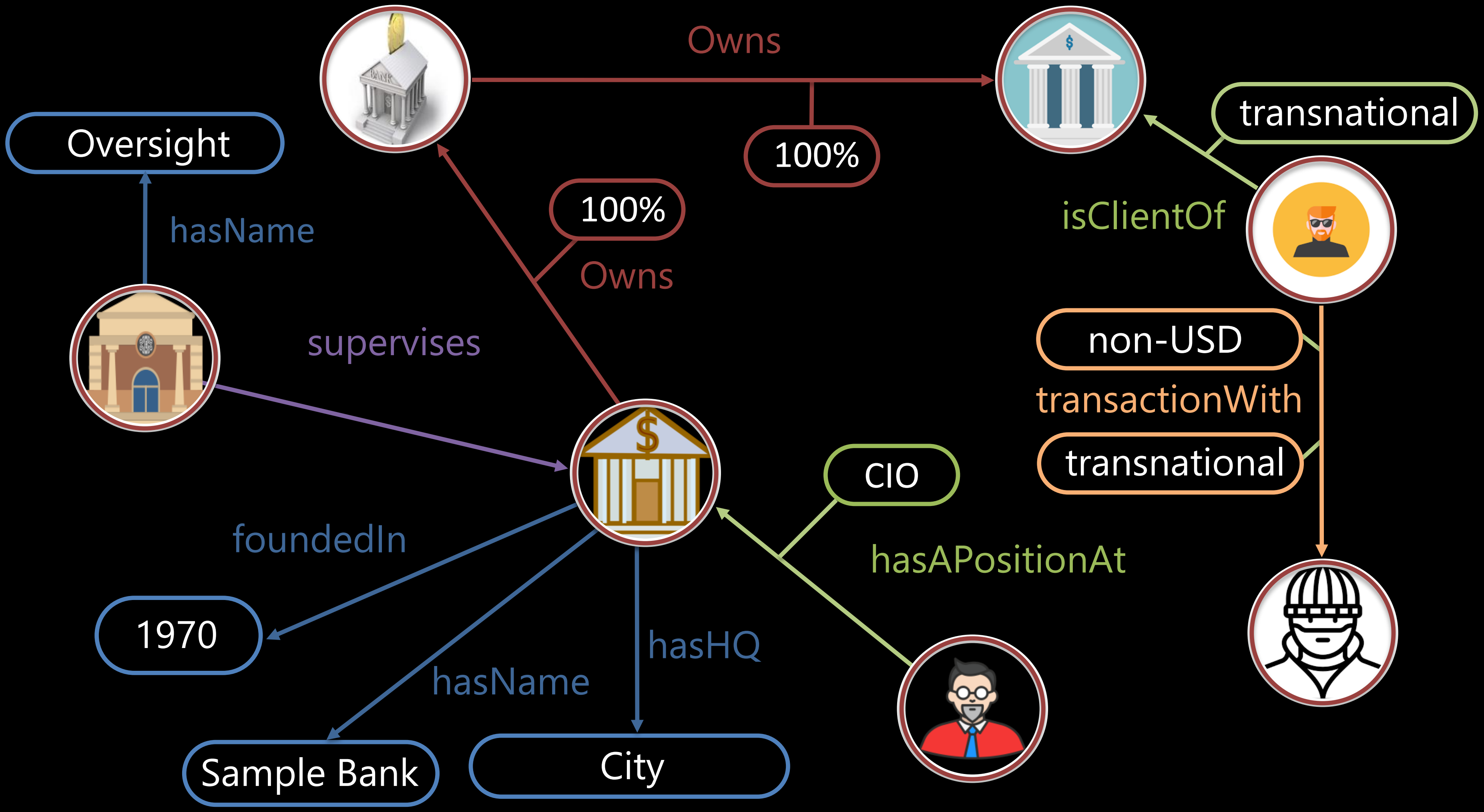
Danske Bank chairman ousted by main shareholder after scandal

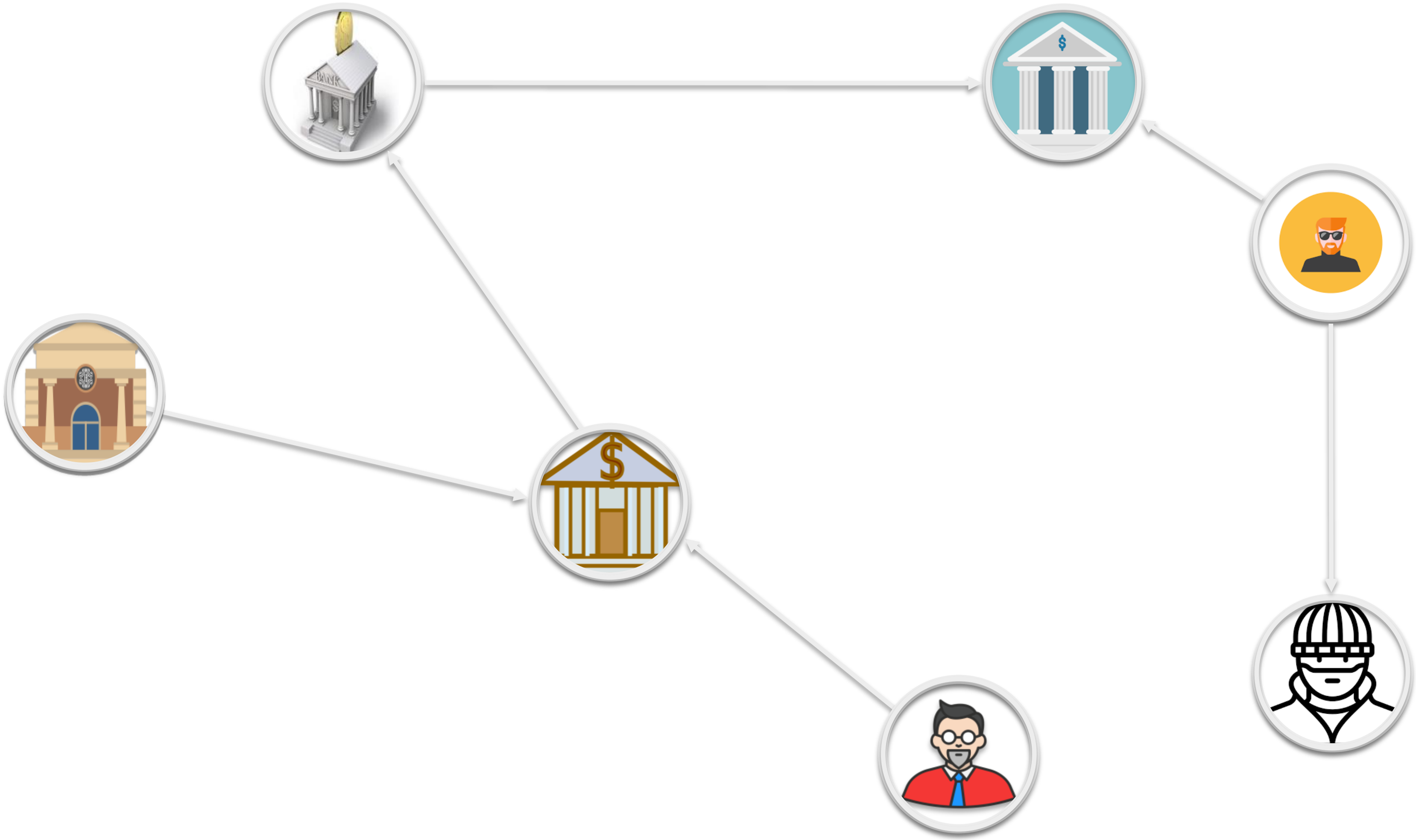
Maersk family brings in new blood to stabilise lender in wake of €200bn money laundering

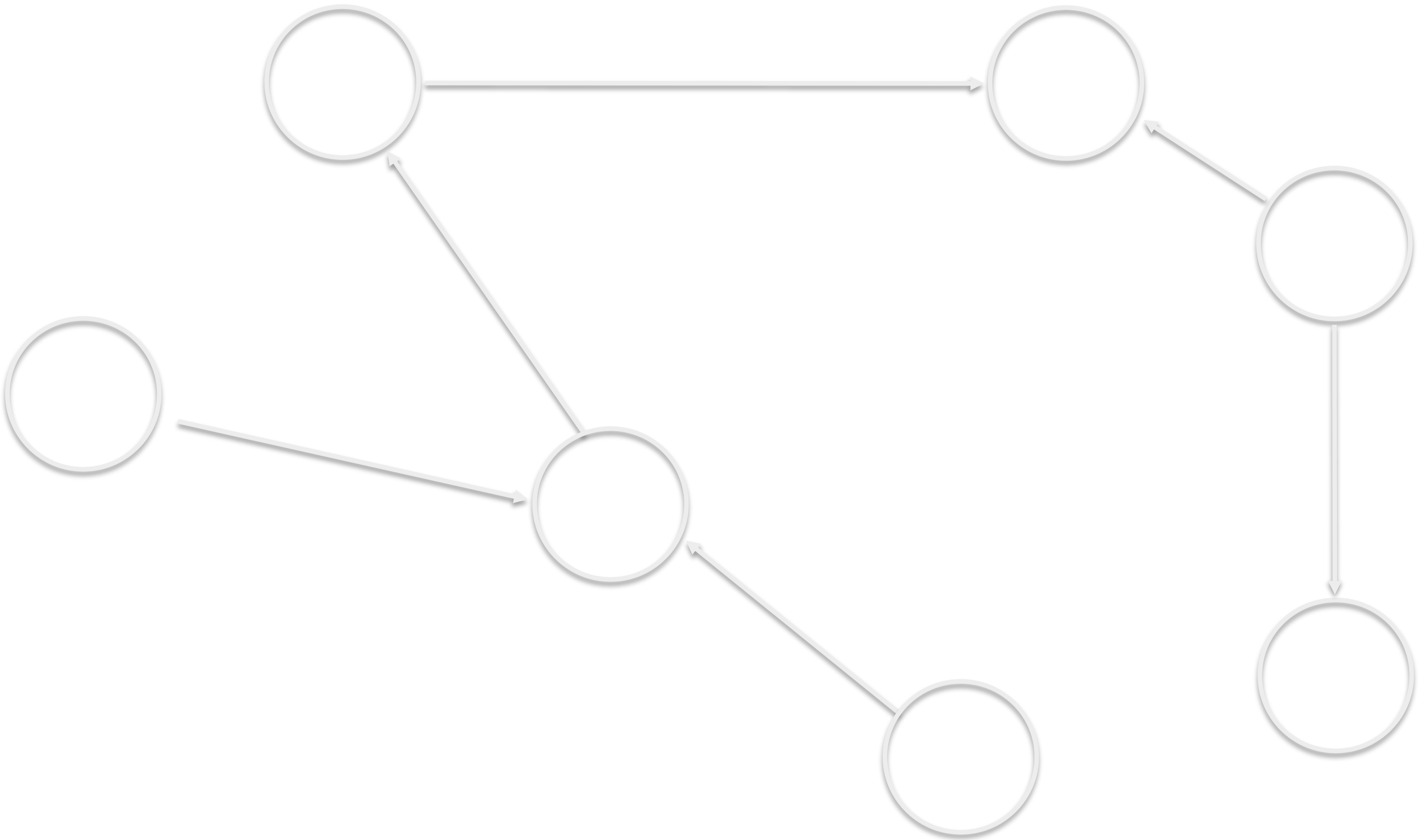
Ole Andersen will step down as chairman of Danske Bank at an extraordinary general meeting in the next few weeks © Bloomberg

€200bn money laundering


Owns
100%
transnational
FSA
hasName
100%
Owns
isClientOf
supervises
non-USD
transactionWith
transnational
CIO
foundedIn
hasAPositionAt
1871
hasHQ
hasName
Danske Bank
Copenhagen

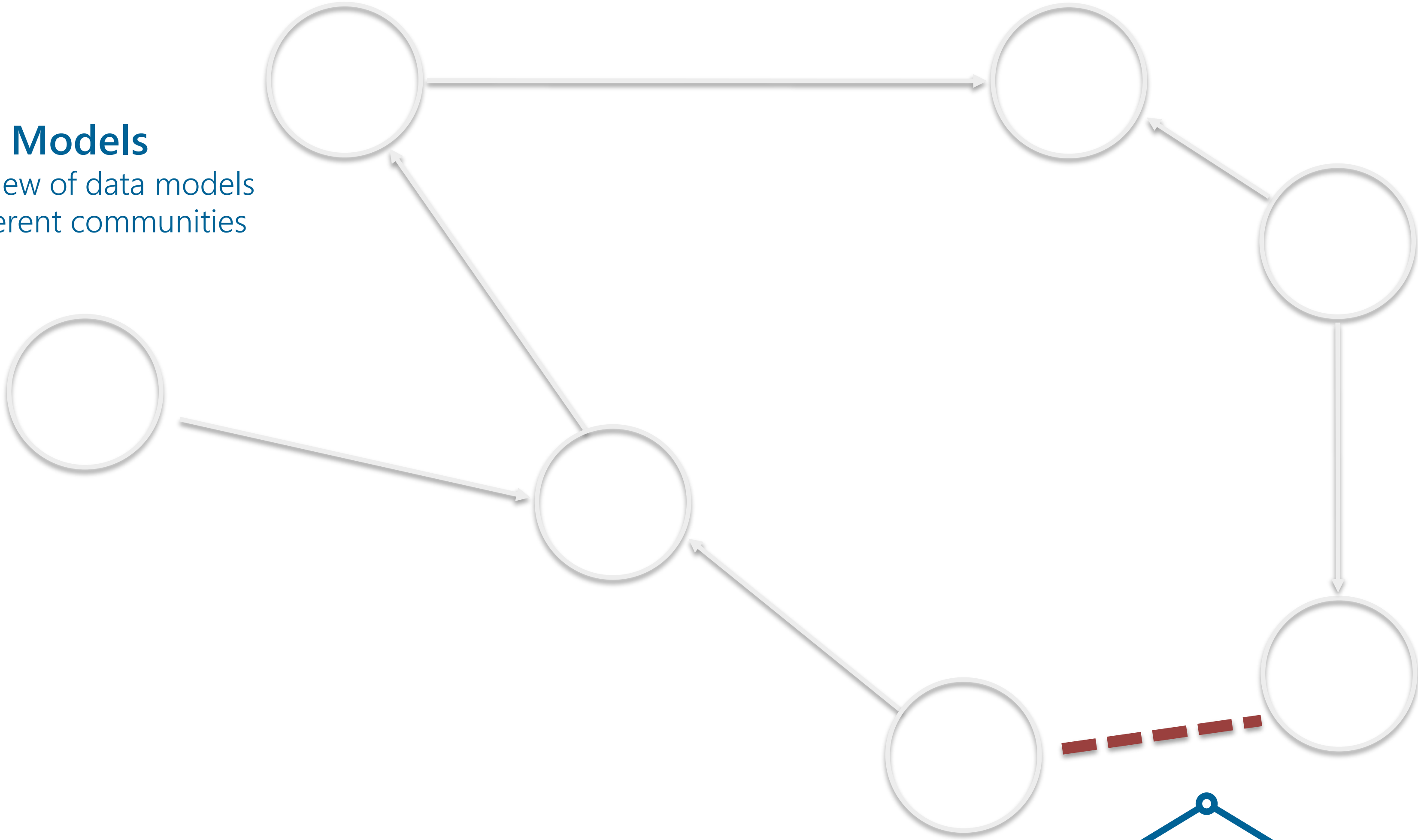

Owns
100%
transnational
Oversight
100%
hasName
isClientOf
Owns
supervises
non-USD
transactionWith
transnational
CIO
foundedIn
hasAPositionAt
1970
hasHQ
hasName
Sample Bank
City





Data Models

Overview of data models
in different communities

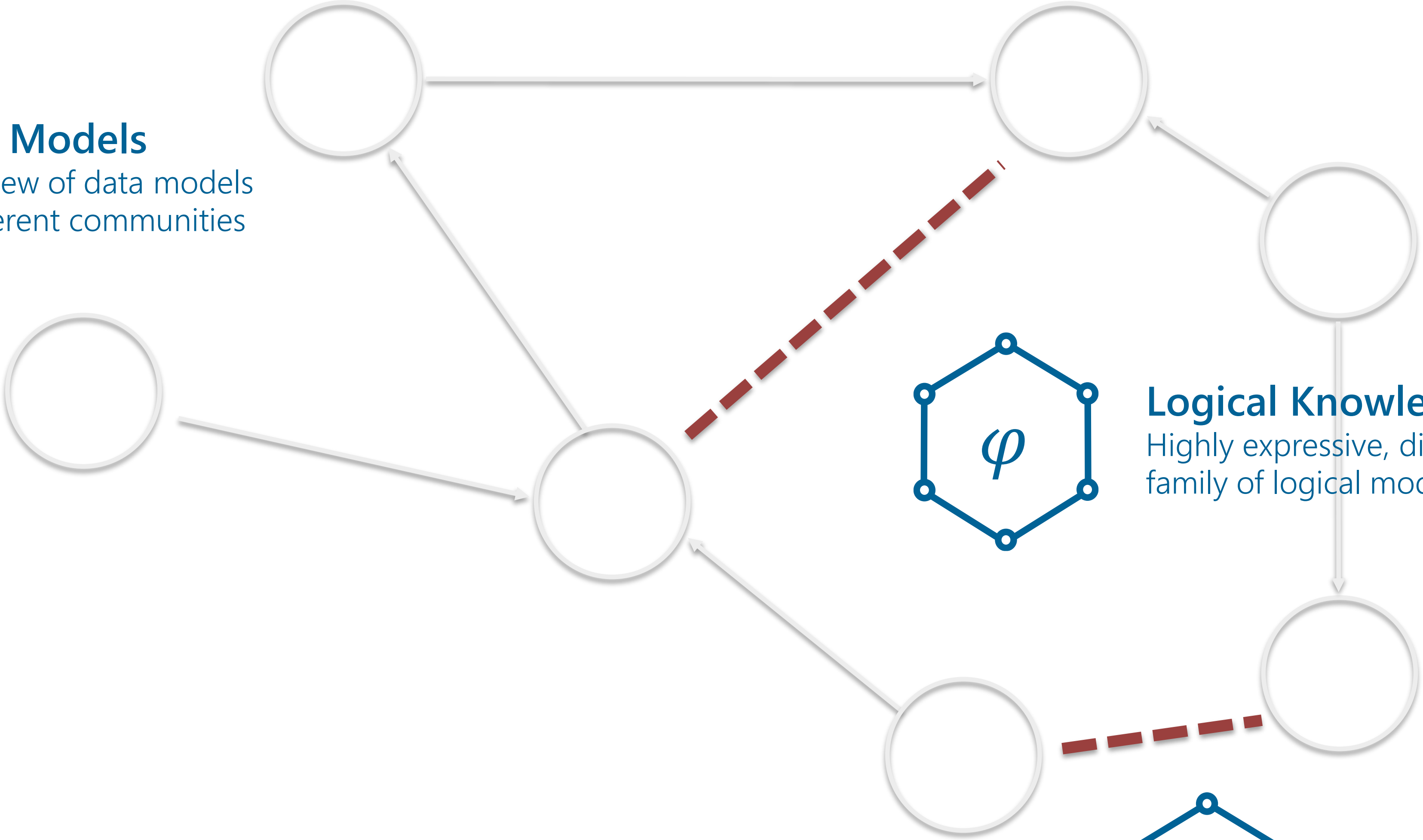


Data Models

Overview of data models in different communities

KG Embeddings

Widely-applied, large family of ML models



Data Models

Overview of data models
in different communities

Logical Knowledge in KGs

Highly expressive, diverse
family of logical models.

KG Embeddings

Widely-applied, large
family of ML models

Data Models
Overview of data models in different communities

Logical Knowledge in KGs
Highly expressive, diverse family of logical models.

Graph Neural Networks
Using the KG structure *as* a neural network.

KG Embeddings
Widely-applied, large family of ML models

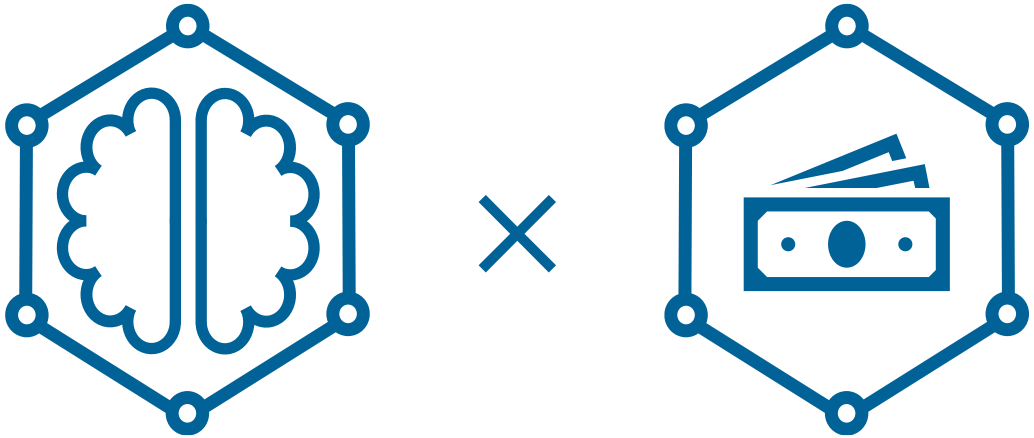
Representations

KG Embeddings
Widely-applied, large
family of ML models

Logical Knowledge in KGs
Highly expressive, diverse
family of logical models.

Graph Neural Networks
Using the KG structure
as a neural network.

Data Models
Overview of data models
in different communities

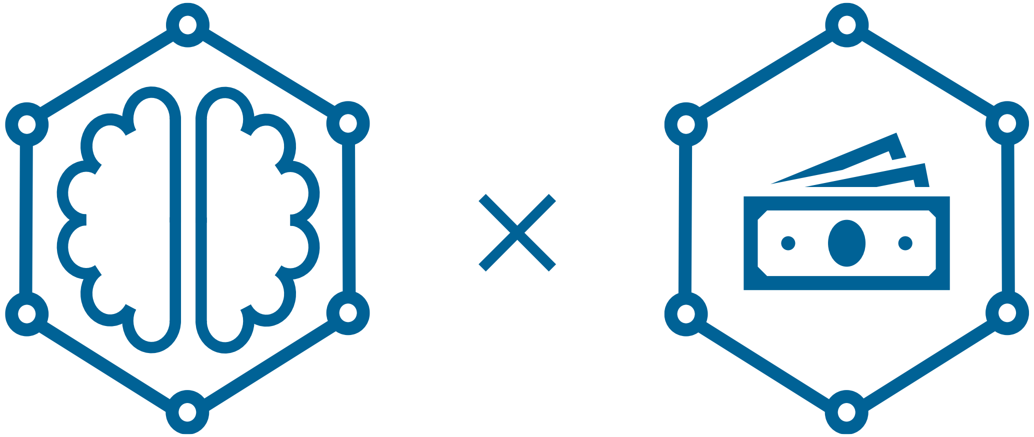


Overview

Representations

With Financial KGs as a Running Example

Emanuel Sallinger



Overview

Systems

With Financial KGs as a Running Example

Emanuel Sallinger

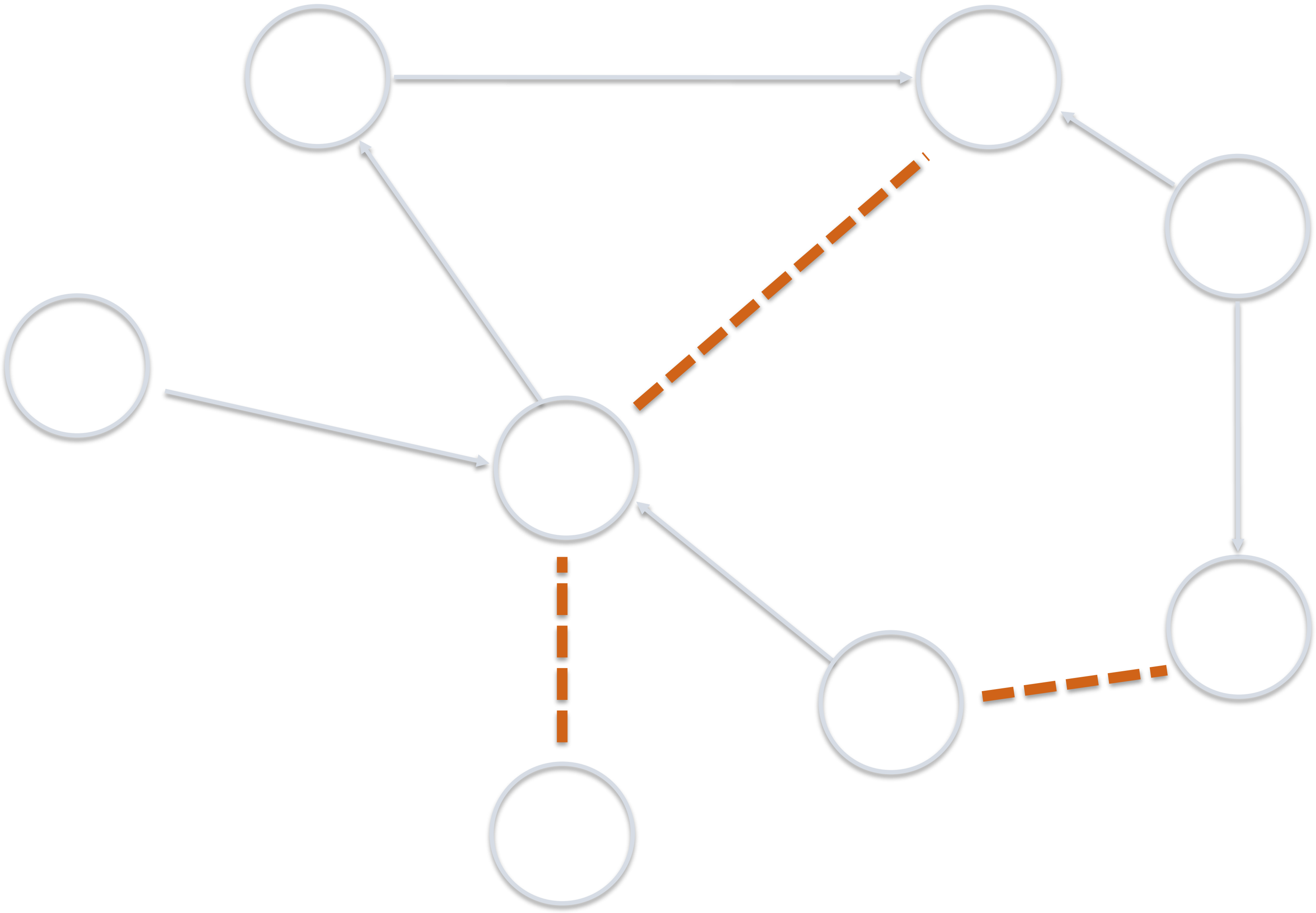
Systems

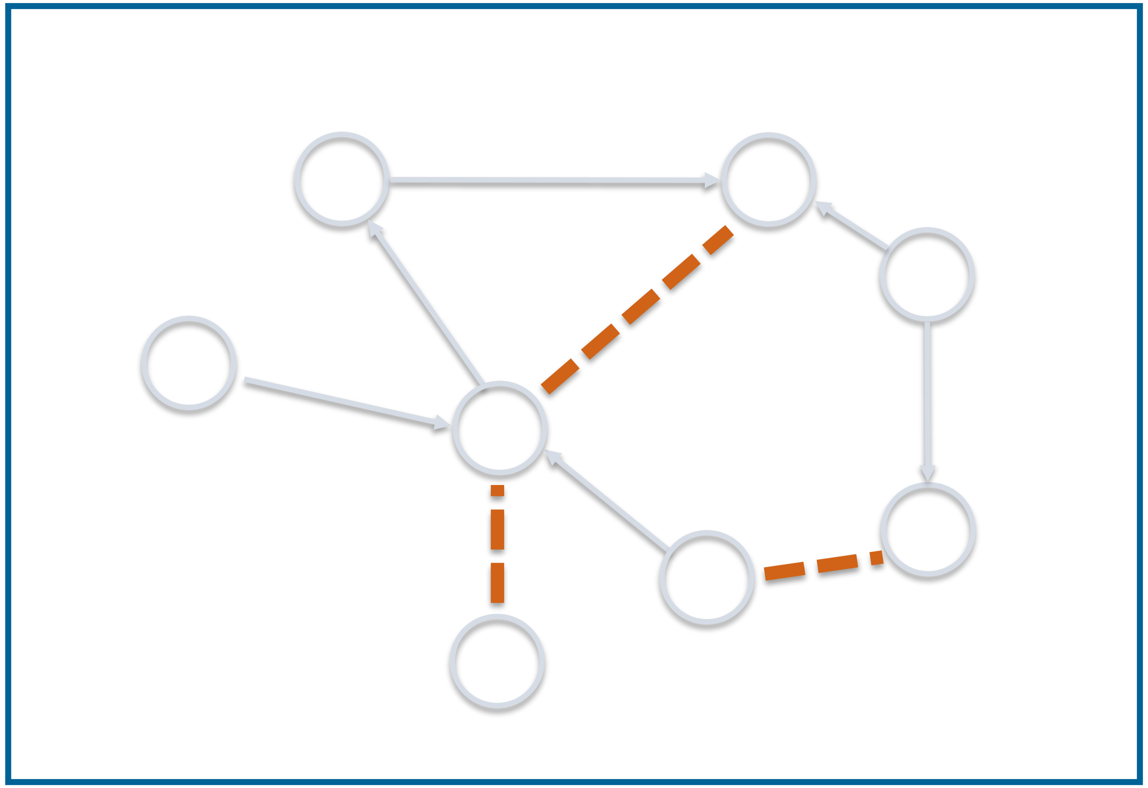
Architectures
The big picture of building IT architectures for KGs

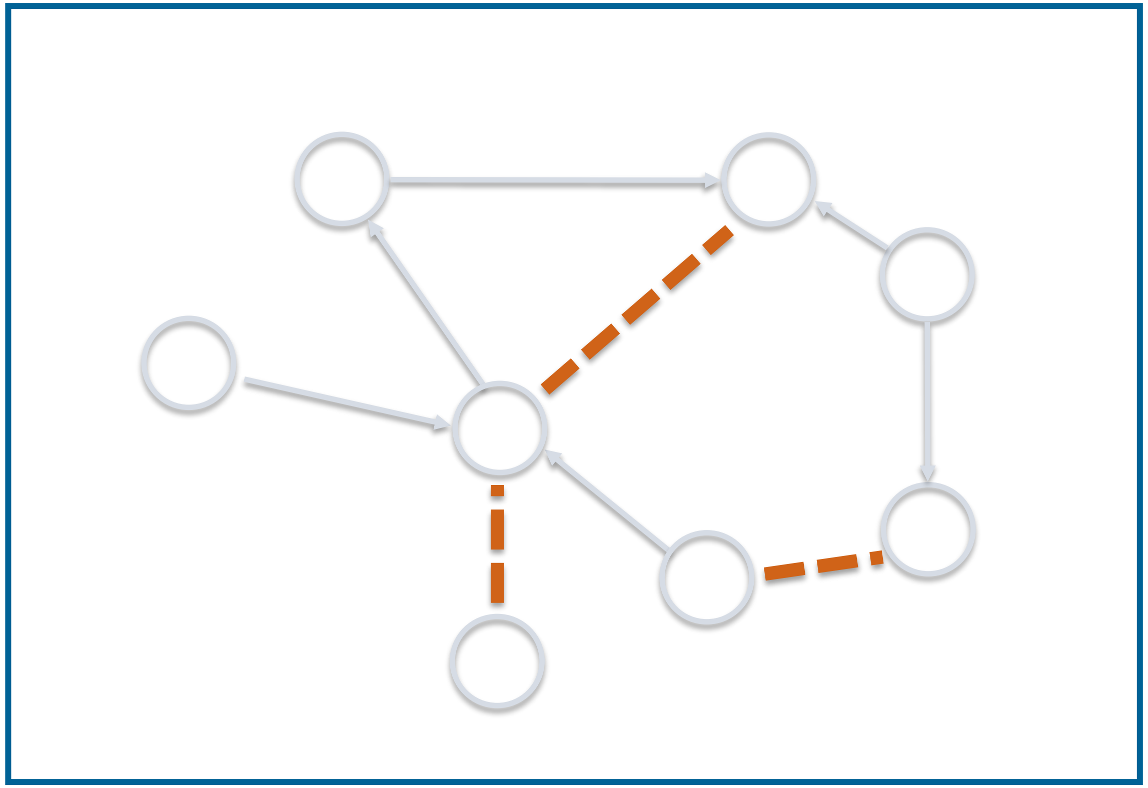
Scalable Reasoning
Making use of the knowledge in the KG

KG Creation
How to create a KG from heterogeneous data?

KG Evolution
How to update, correct and complete a KG?

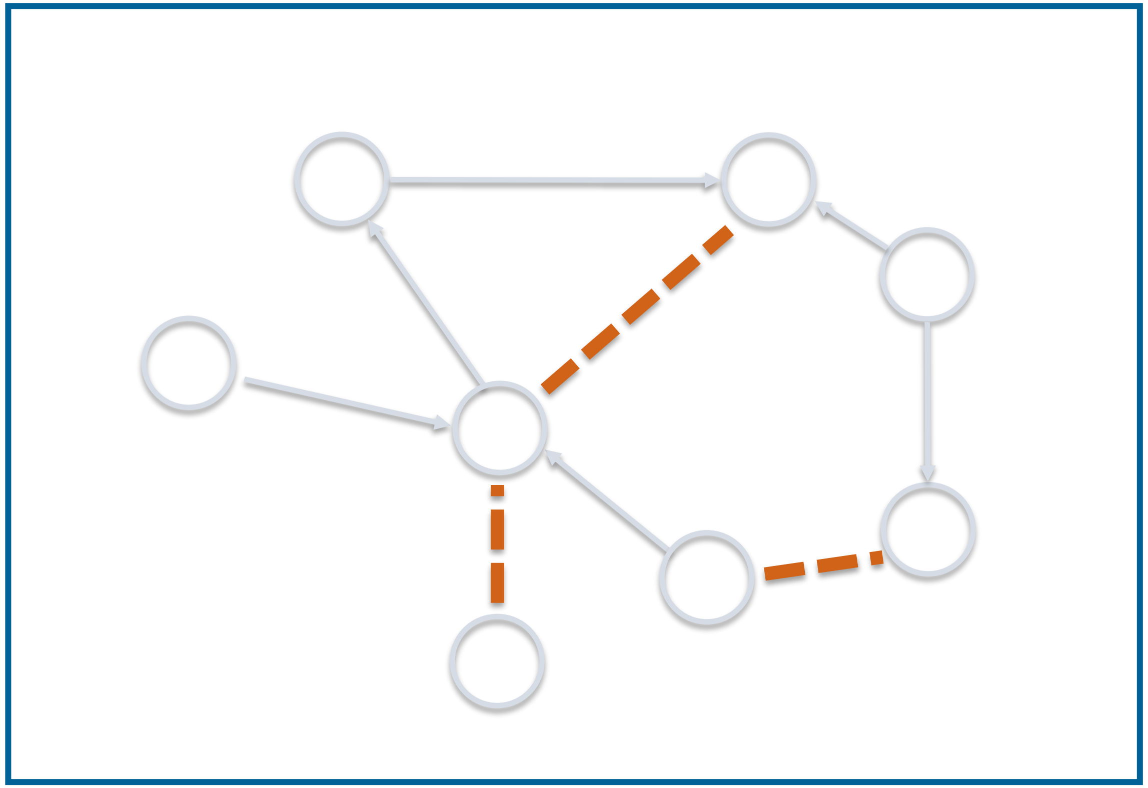






Architectures

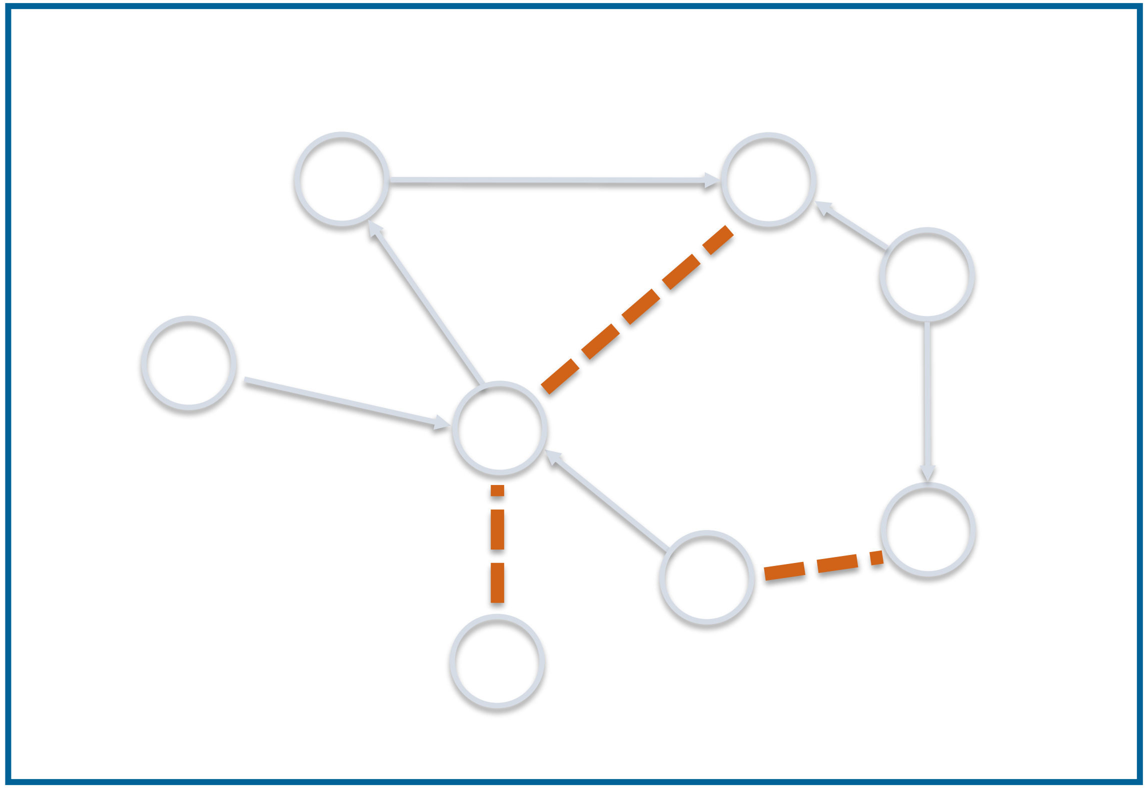
The big picture of building IT architectures for KGs



Architectures

The big picture of building IT architectures for KGs

The example graph

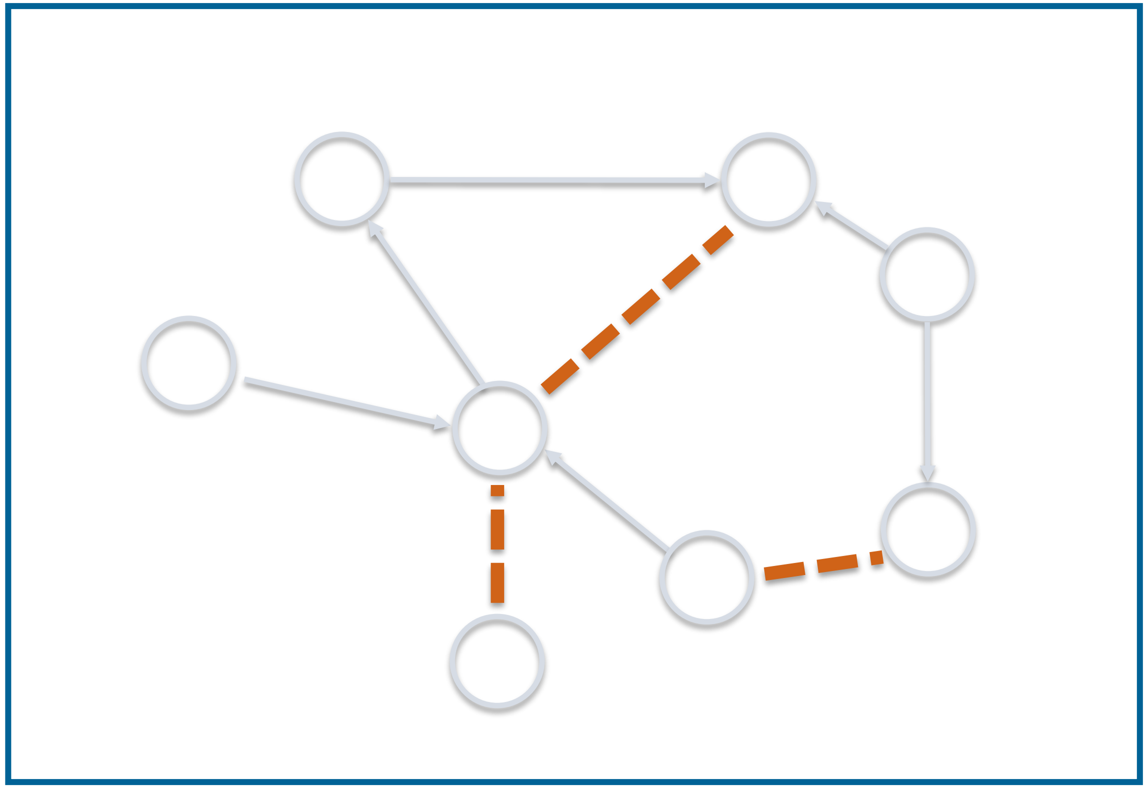


Architectures

The big picture of building IT architectures for KGs

The example graph

Millions of companies



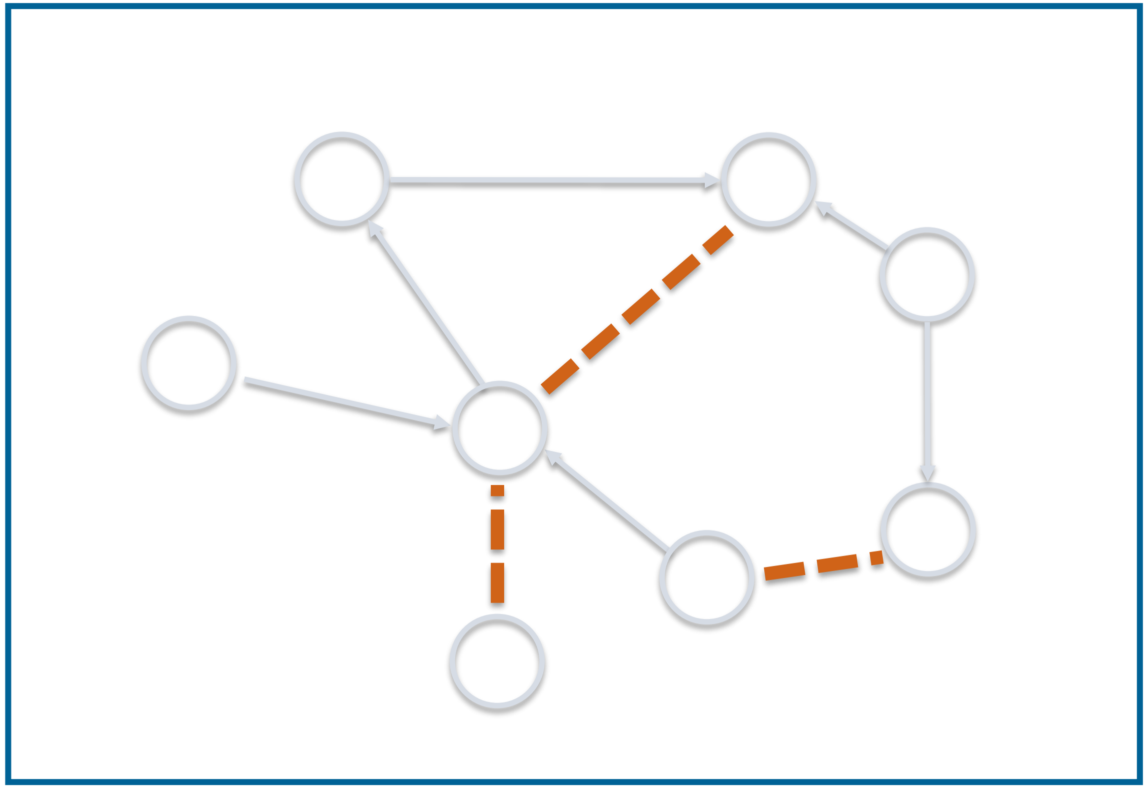
Architectures

The big picture of building IT architectures for KGs

The example graph

Millions of companies

10.000+ TPS
(transactions per second)



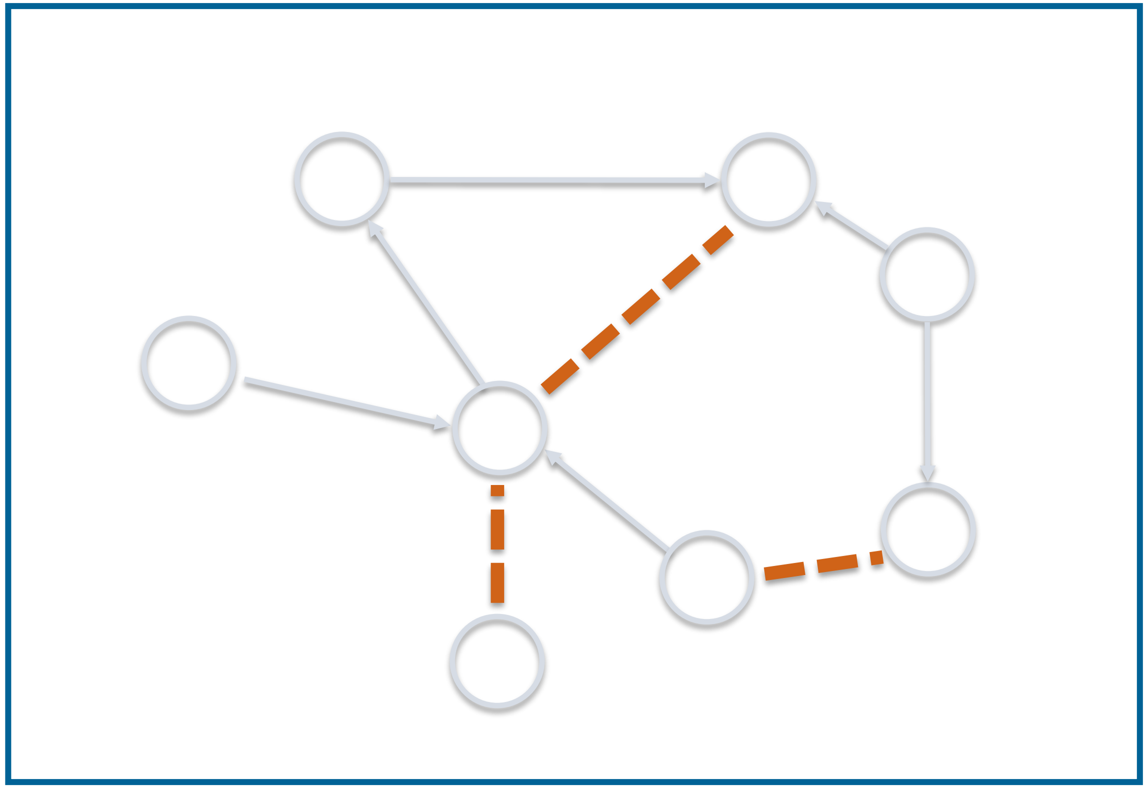
Architectures

The big picture of building IT architectures for KGs

The example graph

Millions of companies

10.000+ TPS
(transactions per second)



Architectures
The big picture of building
IT architectures for KGs

The example graph

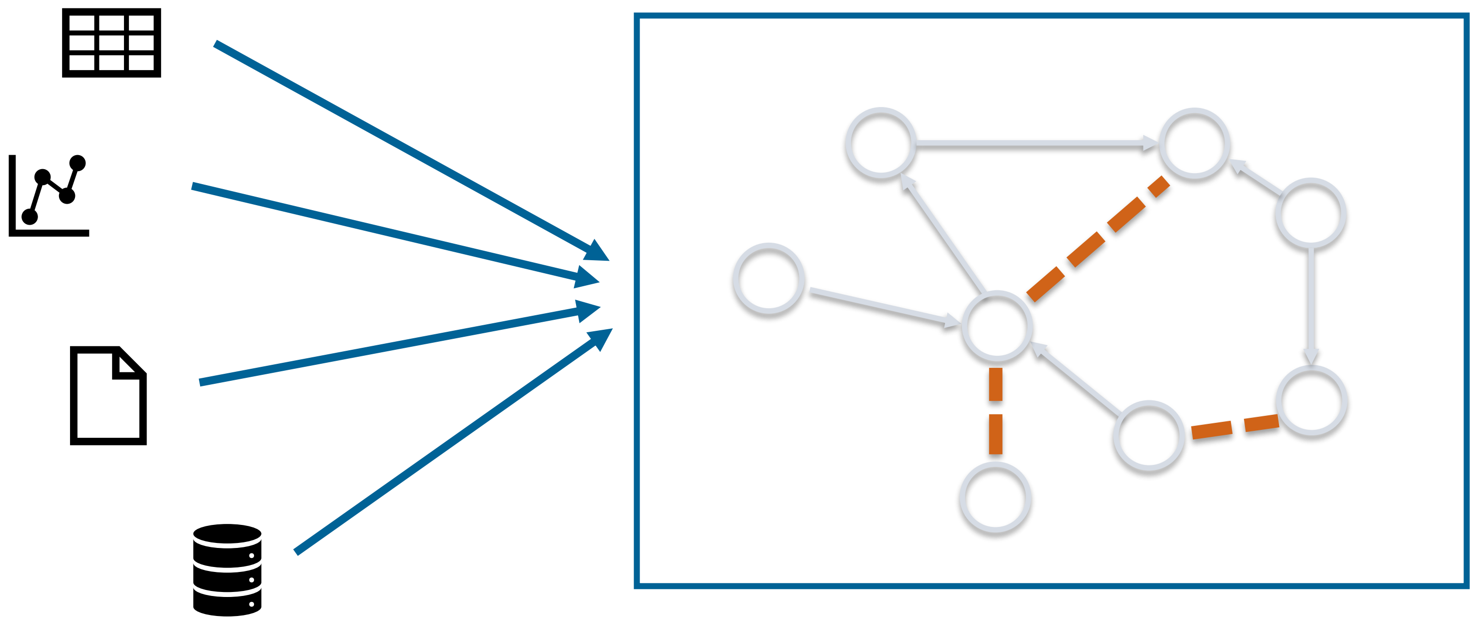
Millions of companies

10.000+ TPS
(transactions per second)

Scalable Reasoning
Making use of the
knowledge in the KG

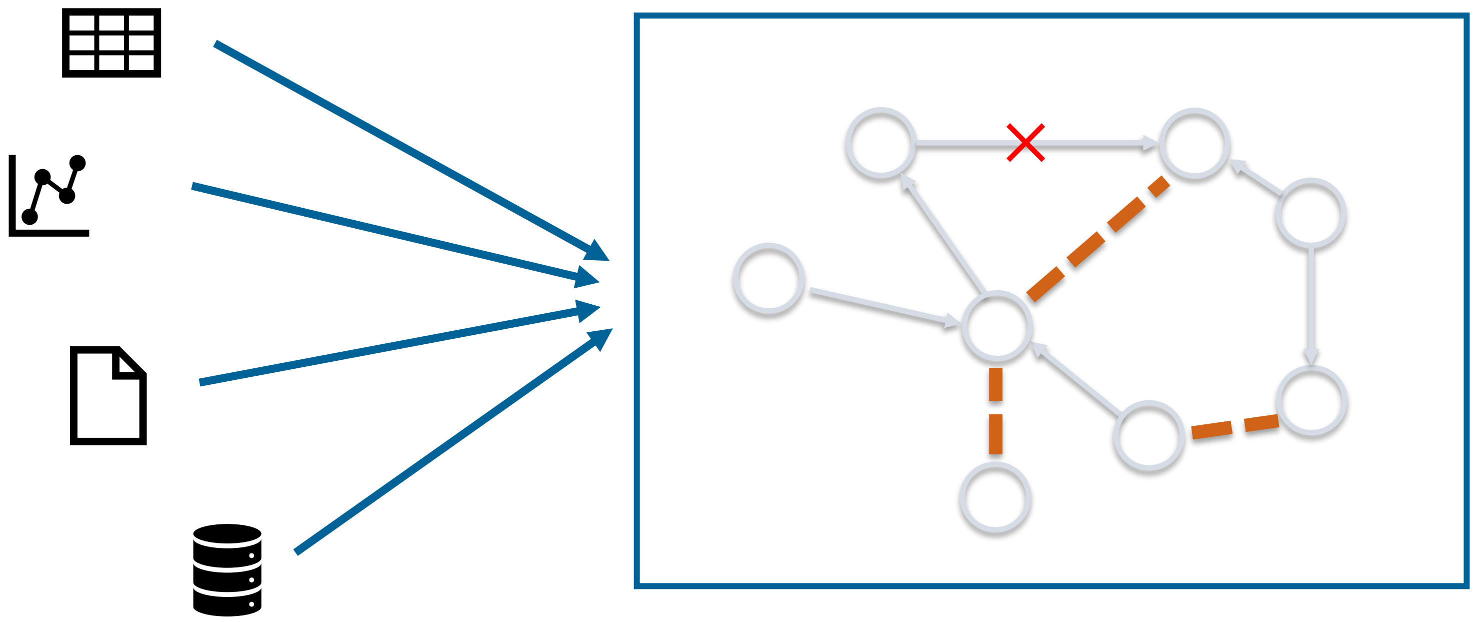
KG Creation

How to create a KG from heterogeneous data?



KG Creation
How to create a KG from heterogeneous data?

KG Evolution
How to update, correct and complete a KG?



Systems

Architectures
The big picture of building IT architectures for KGs

Scalable Reasoning
Making use of the knowledge in the KG

KG Creation
How to create a KG from heterogeneous data?

KG Evolution
How to update, correct and complete a KG?

Overview

Systems

With Financial KGs as a Running Example

Emanuel Sallinger

Overview

Applications

With Financial KGs as a Running Example

Emanuel Sallinger

Applications

Real-World Applications
Overview of diverse applications

Financial KGs
Concrete applications in finance and economics

Services
Which service to provide based on KGs?

Connections
.. between KGs, AI, ML and Data Science.

Applications

Real-World Applications
Overview of diverse applications

Financial KGs
Concrete applications in finance and economics

Services
Which service to provide based on KGs?

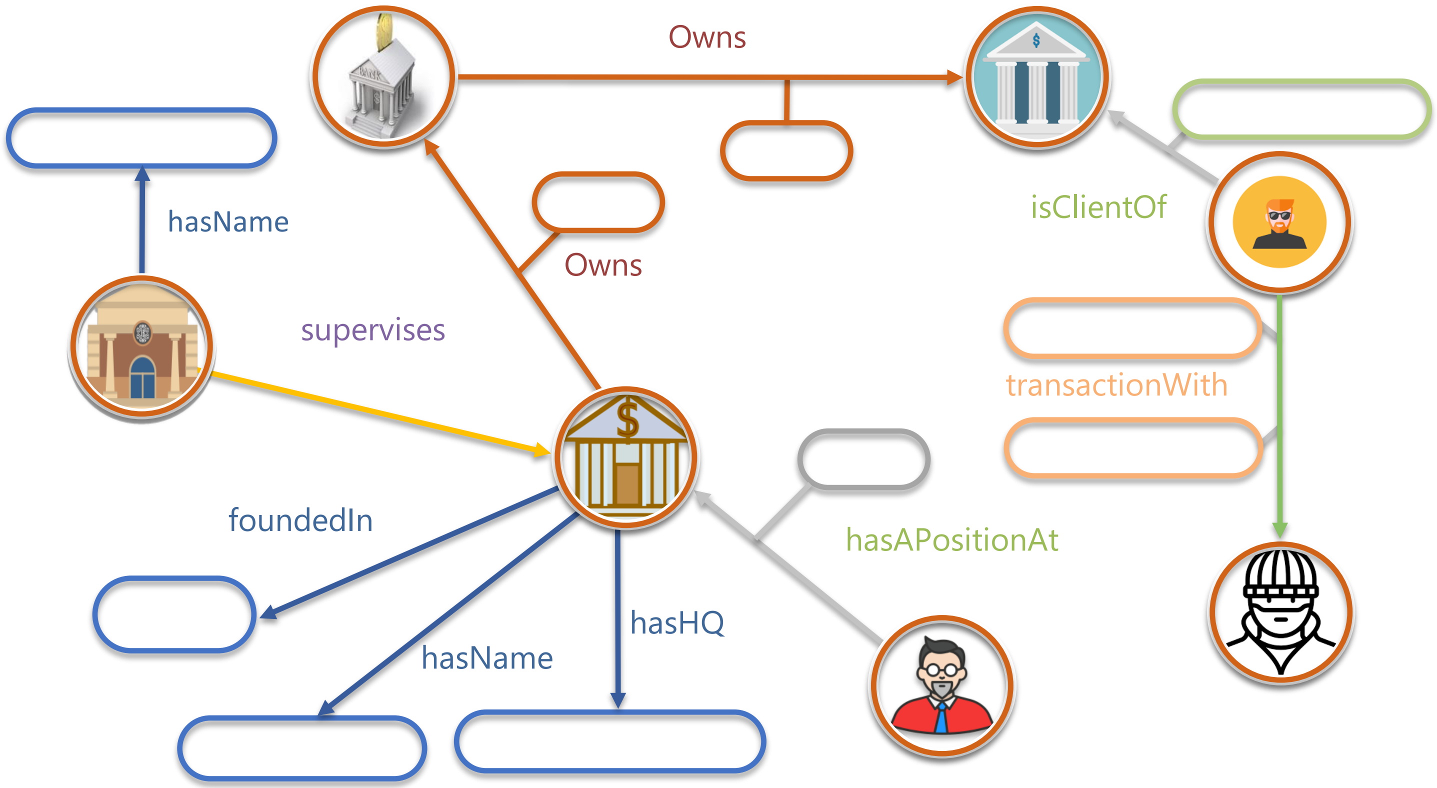
Connections
.. between KGs, AI, ML and Data Science.

Real-World Applications

Overview of diverse applications

Financial KGs

Concrete applications in finance and economics



Applications

Real-World Applications
Overview of diverse applications

Financial KGs
Concrete applications in finance and economics

Services
Which service to provide based on KGs?

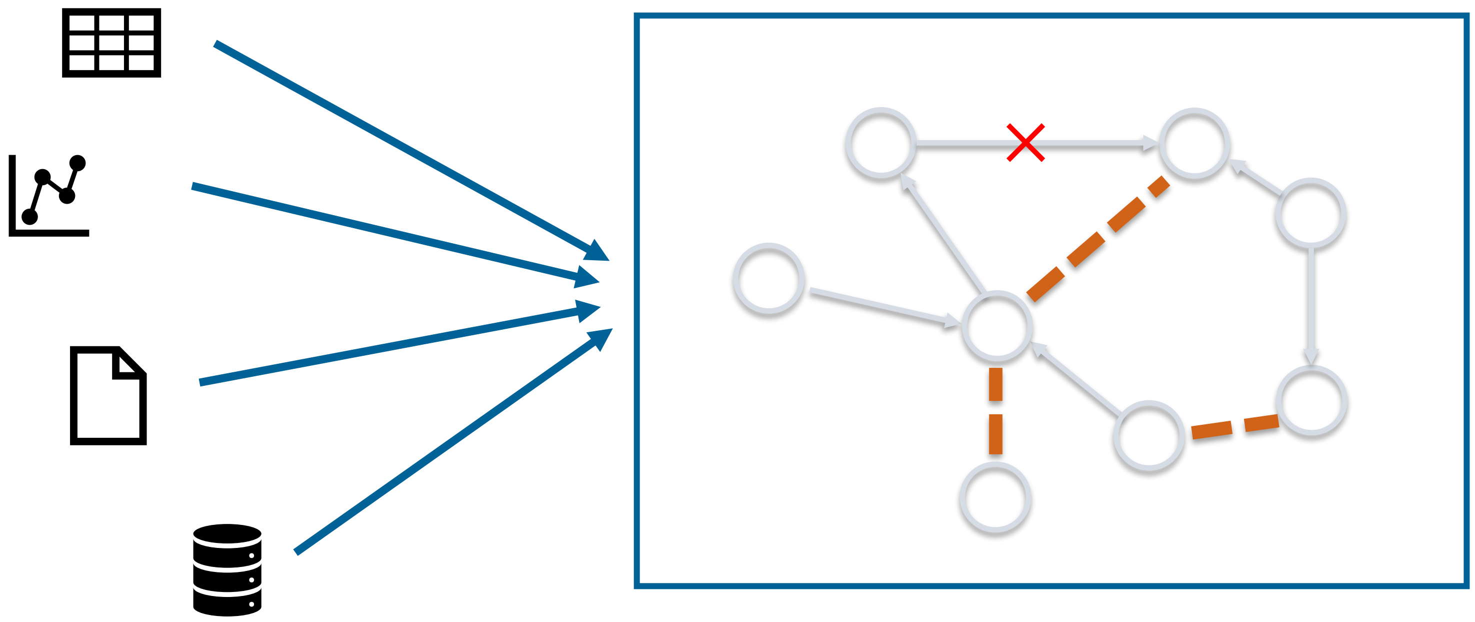
Connections
.. between KGs, AI, ML and Data Science.

KG Creation

How to create a KG from heterogeneous data?

KG Evolution

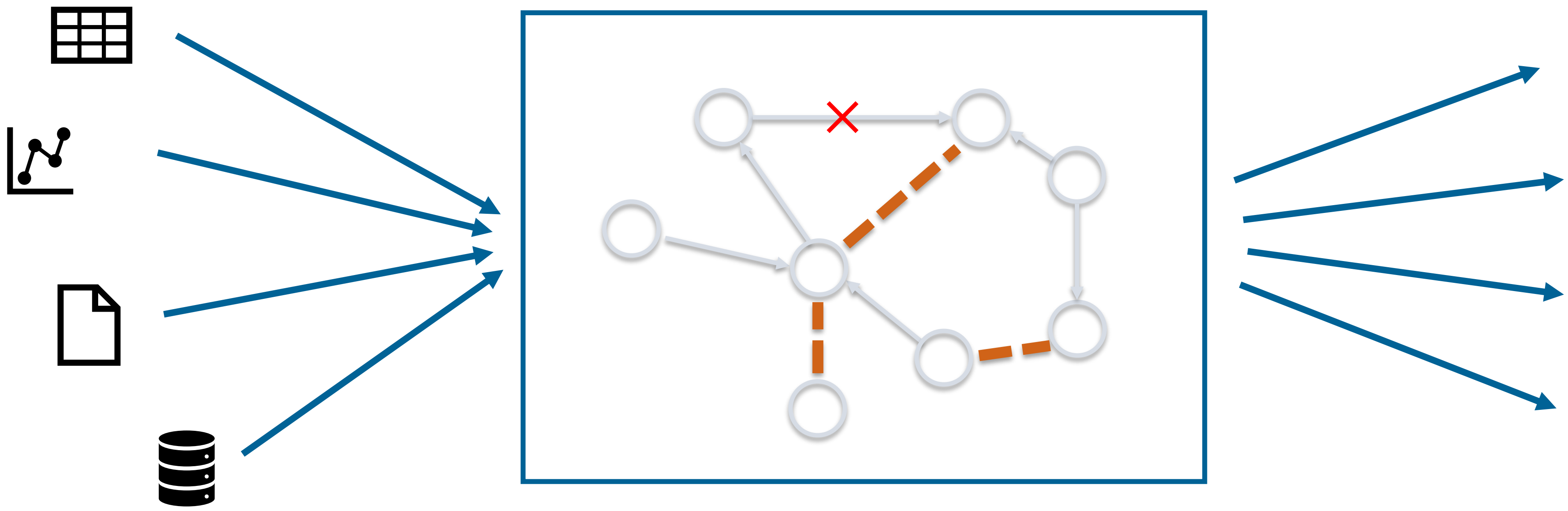
How to update, correct and complete a KG?



KG Creation
How to create a KG from heterogeneous data?

KG Evolution
How to update, correct and complete a KG?

Services
Which service to provide based on KGs?



Applications

Real-World Applications
Overview of diverse applications

Financial KGs
Concrete applications in finance and economics

Services
Which service to provide based on KGs?

Connections
.. between KGs, AI, ML and Data Science.

Data Science

Artificial Intelligence

Knowledge Graphs

Machine Learning

Neural Networks

Deep Learning

Applications

Real-World Applications
Overview of diverse applications

Financial KGs
Concrete applications in finance and economics

Services
Which service to provide based on KGs?

Connections
.. between KGs, AI, ML and Data Science.

Representations

Graph

Knowledge Graph
Data Models

Knowledge

Knowledge Graph
Embeddings

Logical
Knowledge

Graph Neural
Networks

Systems

Architectures
The big picture of building IT architectures for KGs

Scalable Reasoning
Making use of the knowledge in the KG

KG Creation
How to create a KG from heterogeneous data?

KG Evolution
How to update, correct and complete a KG?

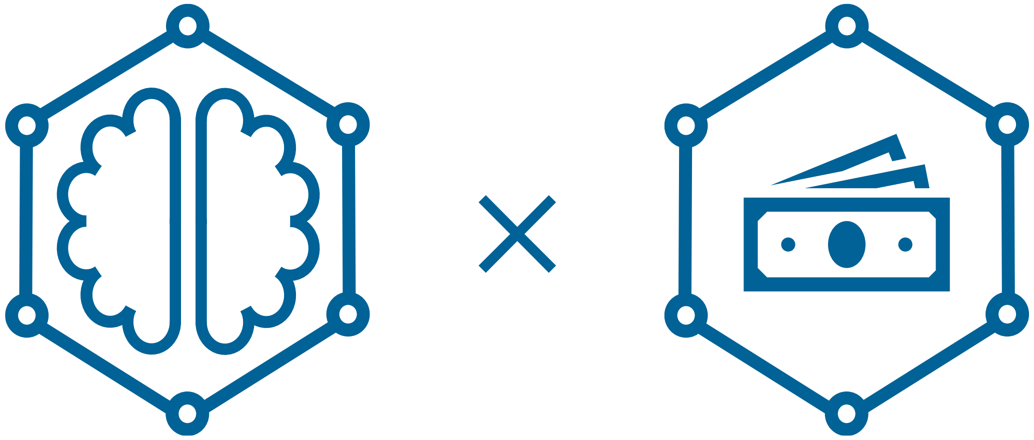
Applications

Real-World Applications
Overview of diverse applications

Financial KGs
Concrete applications in finance and economics

Services
Which service to provide based on KGs?

Connections
.. between KGs, AI, ML and Data Science.



Overview

Applications

With Financial KGs as a Running Example

Emanuel Sallinger

Logical Knowledge in KGs

Emanuel Sallinger

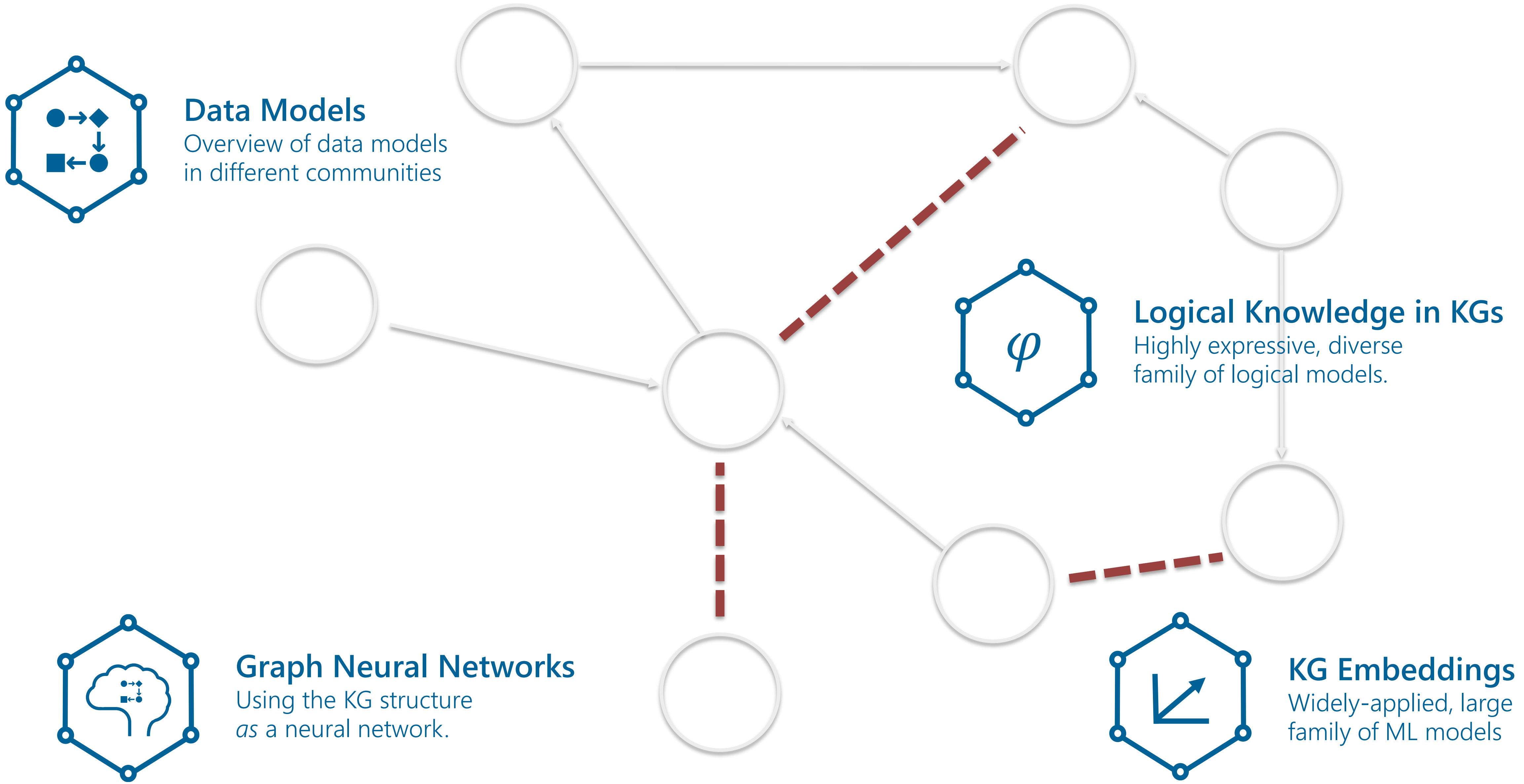
Representations

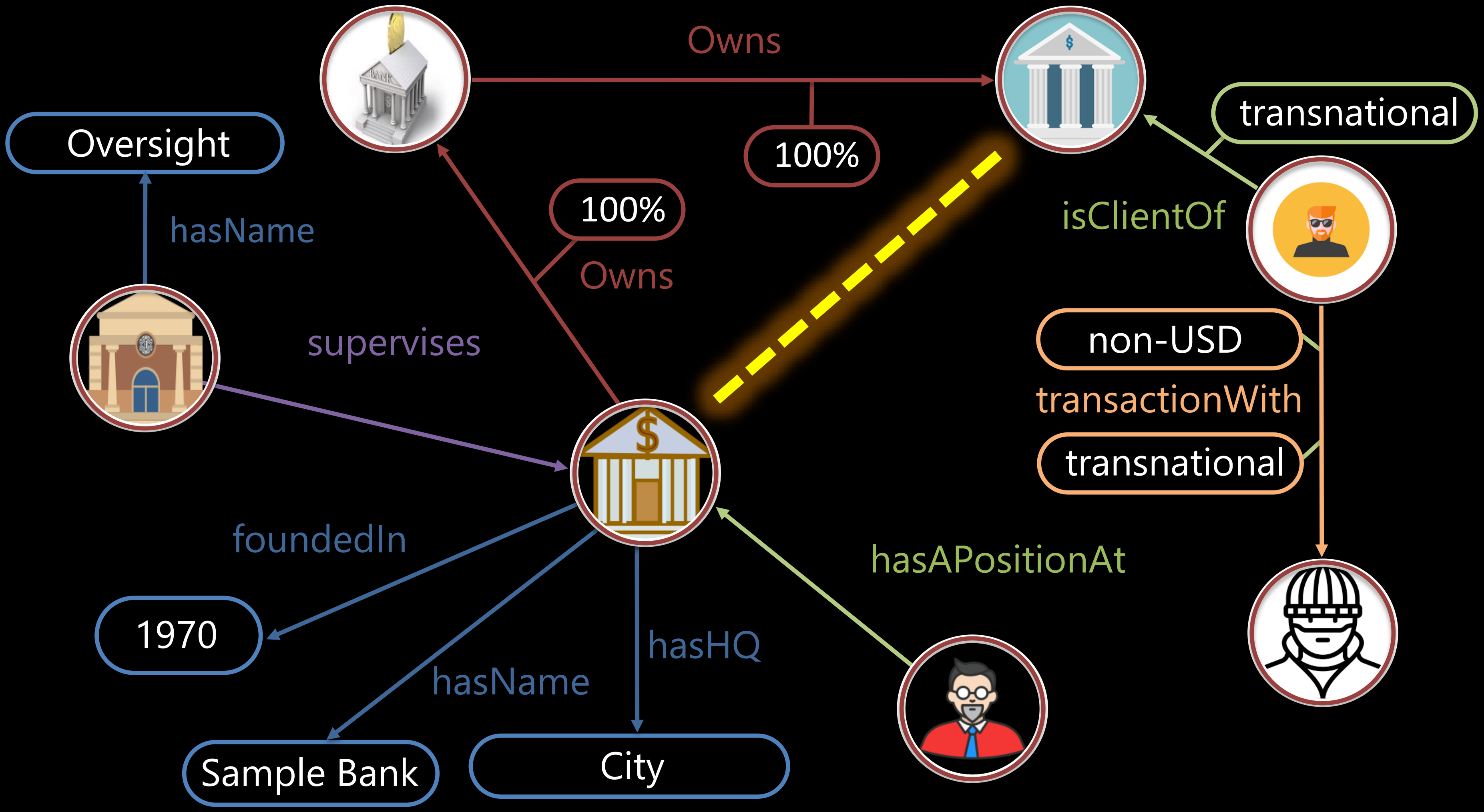
KG Embeddings
Widely-applied, large
family of ML models

Logical Knowledge in KGs
Highly expressive, diverse
family of logical models.

Graph Neural Networks
Using the KG structure
as a neural network.

Data Models
Overview of data models
in different communities


Data Models
Overview of data models
in different communities
Logical Knowledge in KGs
Highly expressive, diverse
family of logical models.
Graph Neural Networks
Using the KG structure
as a neural network.
KG Embeddings
Widely-applied, large
family of ML models


Owns
100%
transnational
Oversight
hasName
100%
Owns
isClientOf
supervises
non-USD
transactionWith
transnational
foundedIn
hasAPositionAt
1970
hasHQ
hasName
Sample Bank
City

Representations

Graph

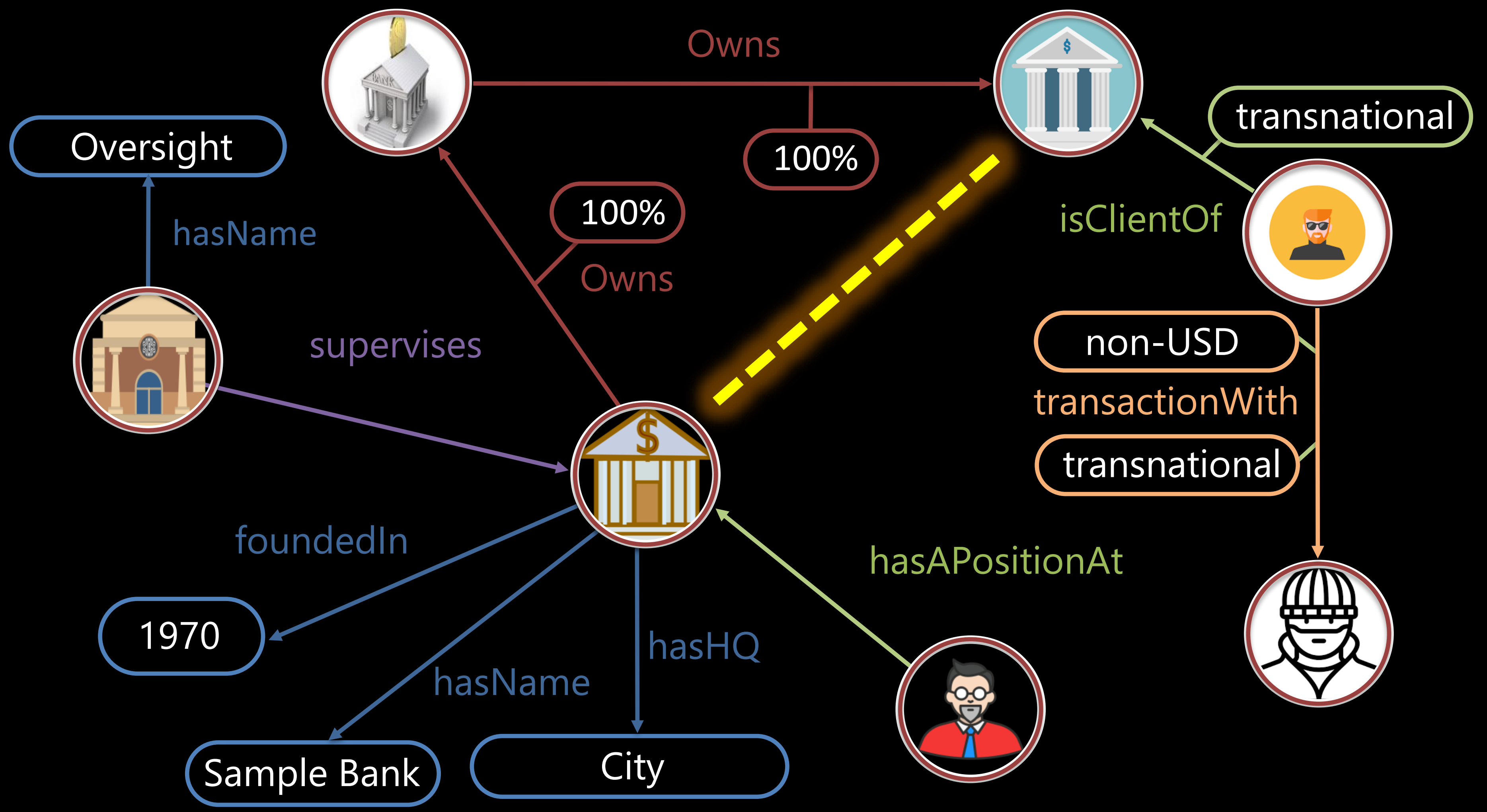
Knowledge Graph
Data Models

Knowledge

Knowledge Graph
Embeddings

Logical
Knowledge

Graph Neural
Networks


Owns
100%
Oversight
hasName
100%
Owns
transnational
isClientOf
supervises
non-USD
transactionWith
transnational
foundedIn
hasAPositionAt
1970
hasHQ
hasName
Sample Bank
City

x **controls** y if

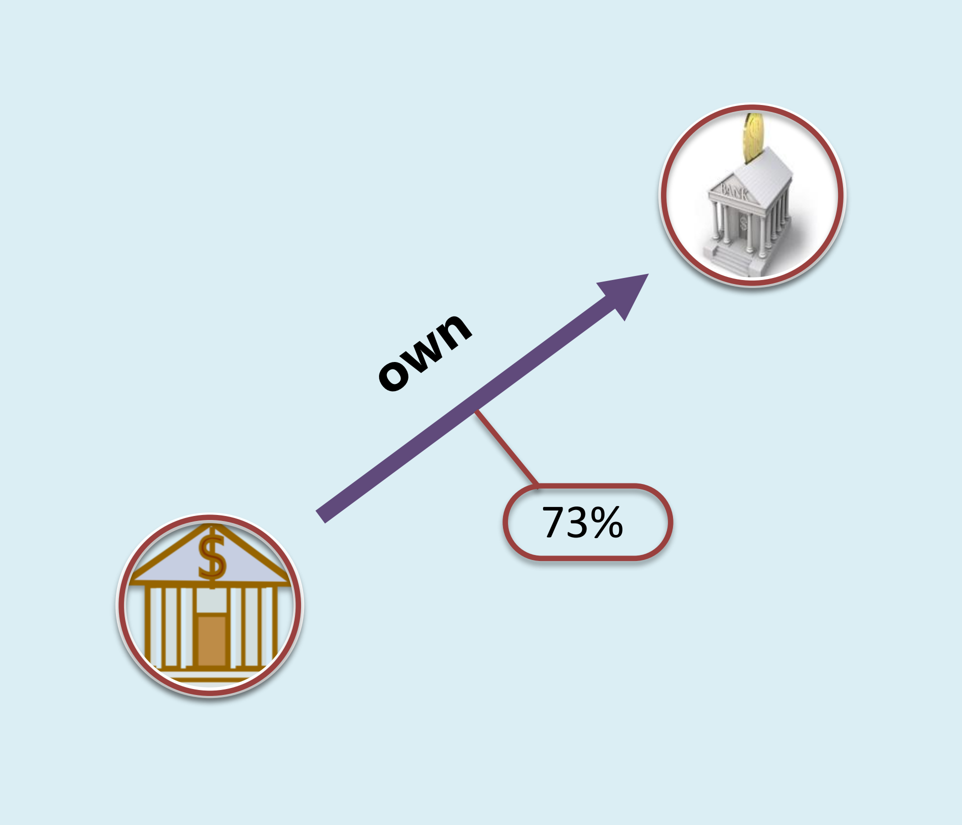
x **directly holds** over 50% of y

First-order Logic

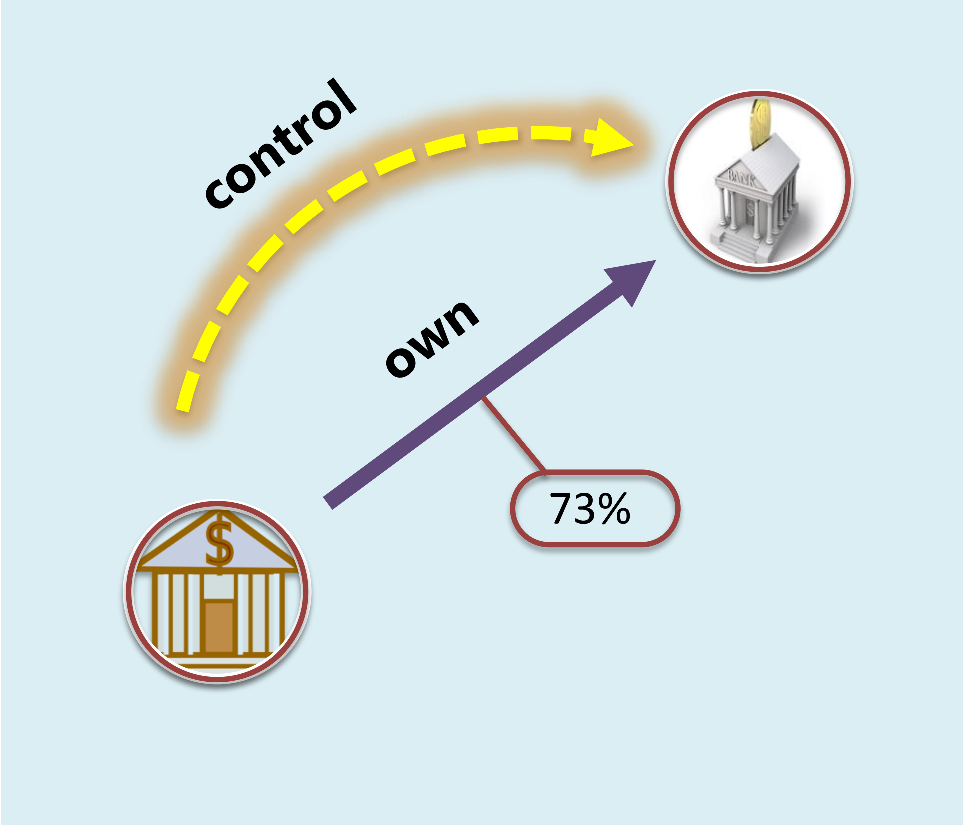
$$\forall x, y\ (own(x, y, w), w > 0.5 \rightarrow control(x, y))$$

x **controls** y if

x **directly holds** over 50% of y



$\rightarrow$



x **controls** y if
  x **directly holds** over 50% of y

Datalog

```
control(X,Y) :- own(X,Y,W), W > 0.5.
```

x **controls** y if
x **directly holds** over 50% of y

SQL

```
SELECT x,y INTO control
FROM company
WHERE w > 0.5
```

x **controls** y if

x **directly holds** over 50% of y

Relational Calculus

$$control = \{(x, y) \mid own(x, y, w), w > 0.5\}$$

x **controls** y if
x **directly holds** over 50% of y

Relational Algebra

$$control = \sigma_{w>0.5}\ own$$

x **controls** y if
x **directly holds** over 50% of y

Cypher (Graph DBs)

```
MATCH  (x:Company) –[o:OWN]-> (y:Company)]
WHERE o.w > 0.5
CREATE (x) –[:CONTROL]-> (y)
```

...

Cypher

SPARQL

Datalog

SQL

First-order Logic

Relational Algebra

Logical Knowledge in KGs

Emanuel Sallinger

Logical Knowledge in KGs

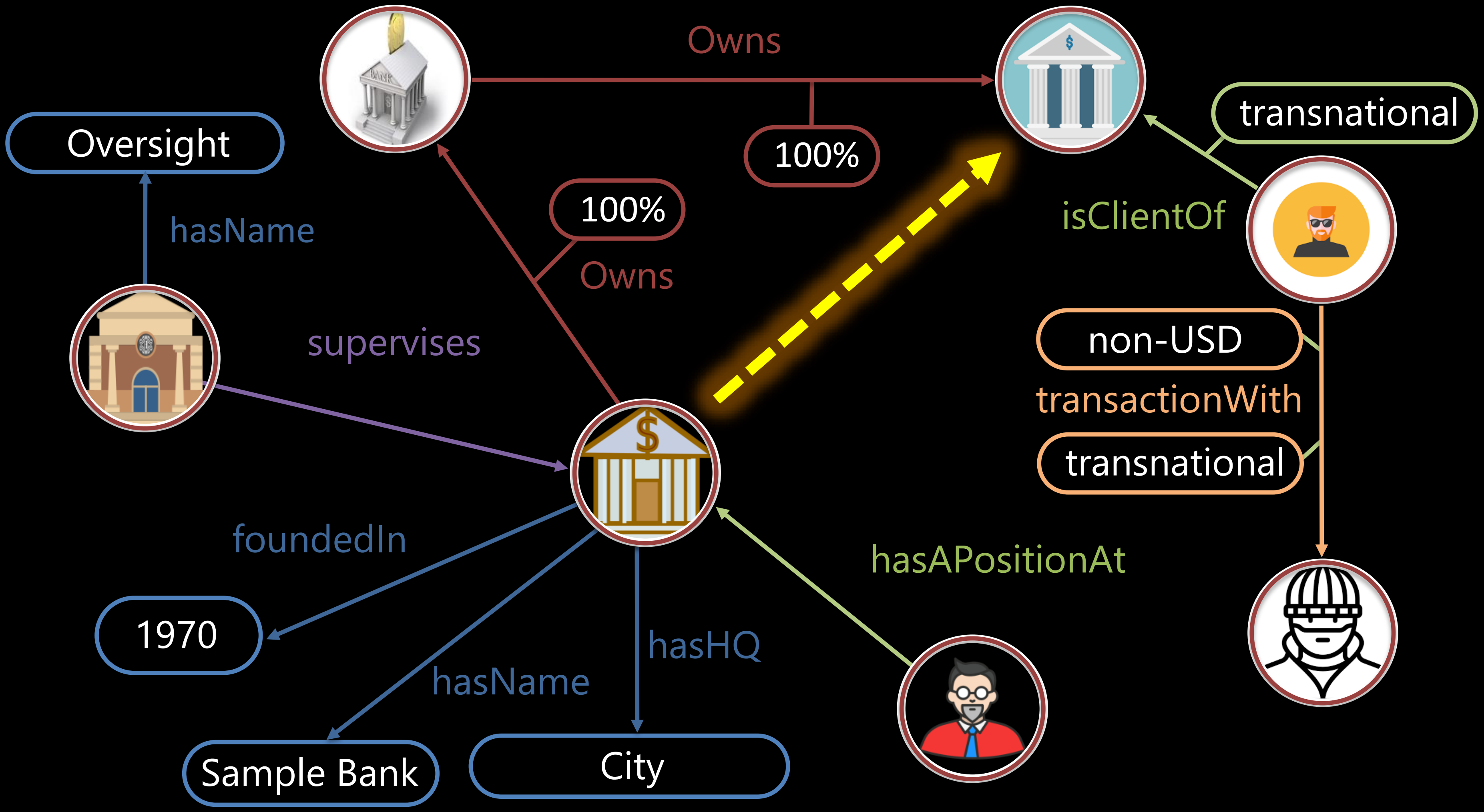
Challenges: Recursion and Creation

Emanuel Sallinger

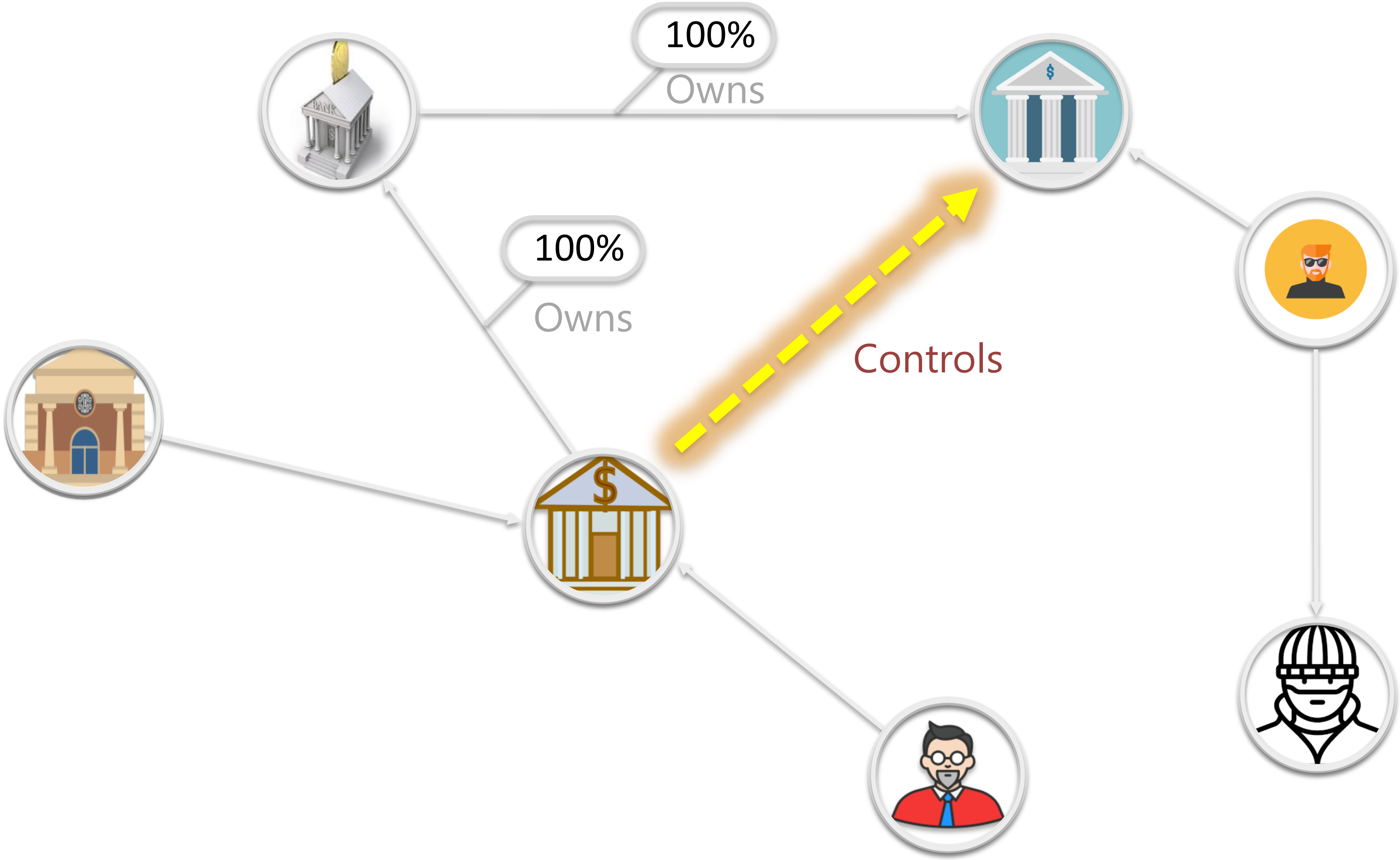
Main Challenges

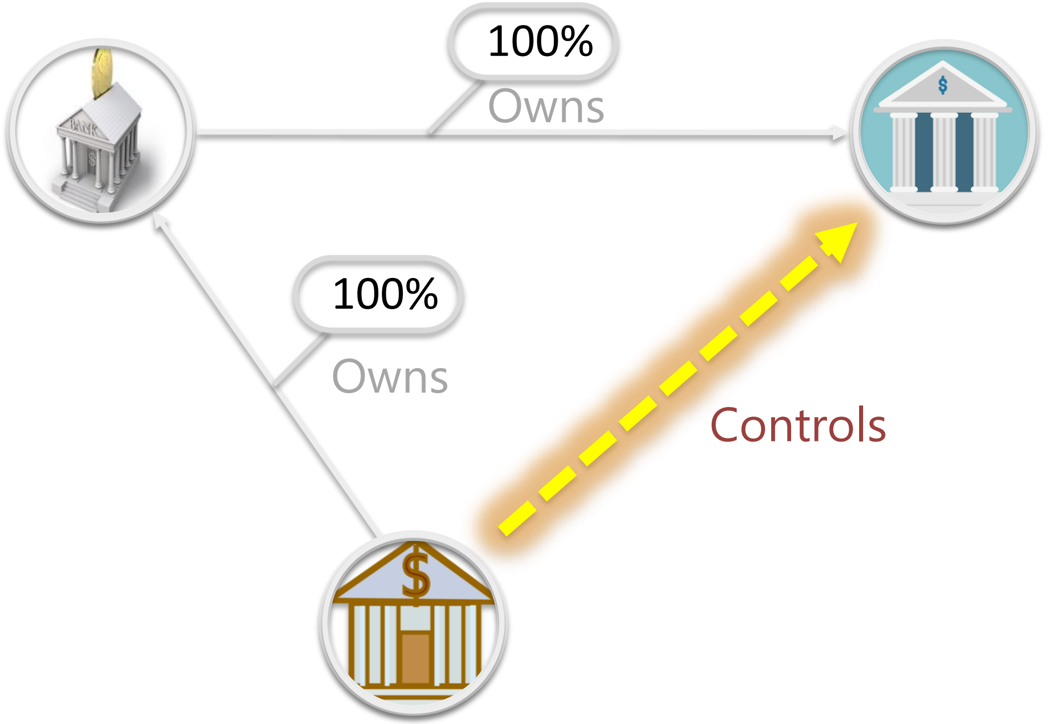
1. **Recursion**
   unlimited graph exploration

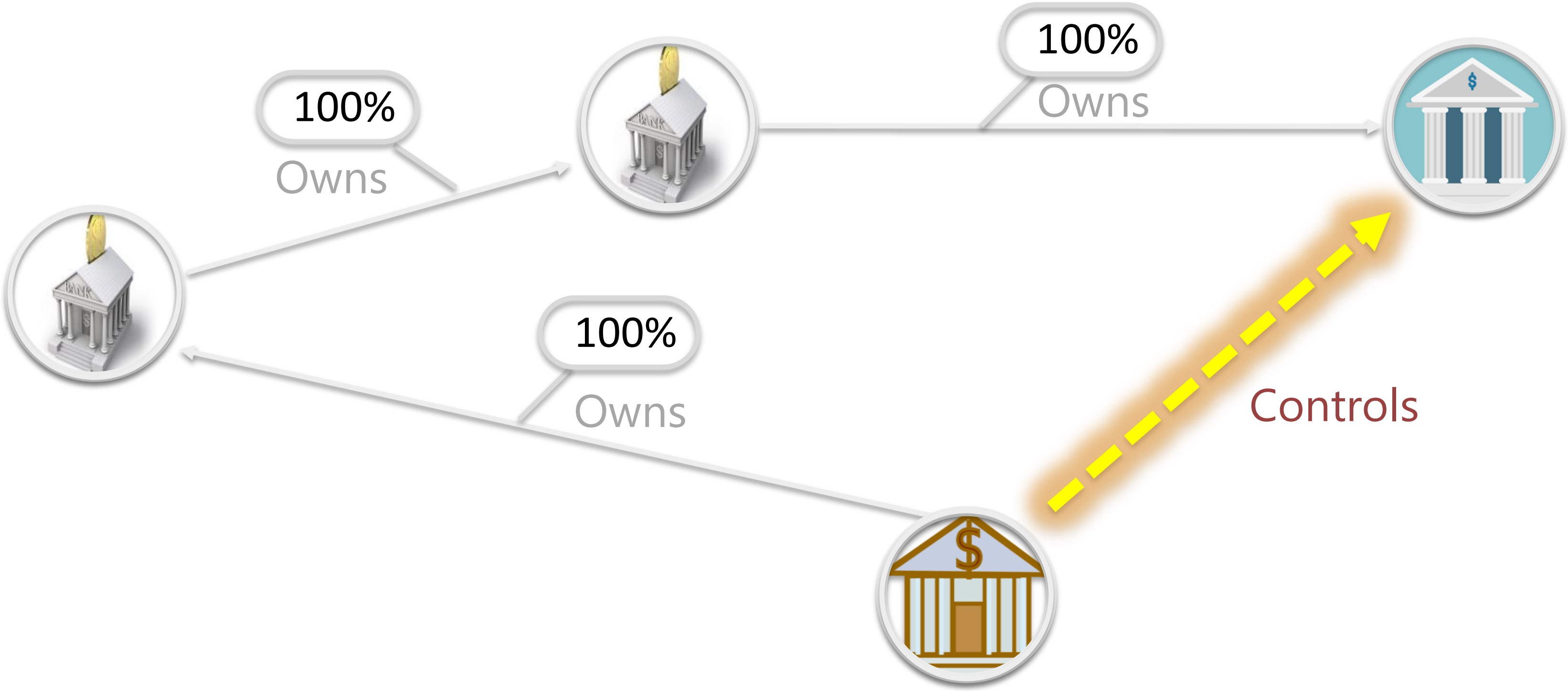
2. **Object Creation**
   exploring unknown parts of the KG

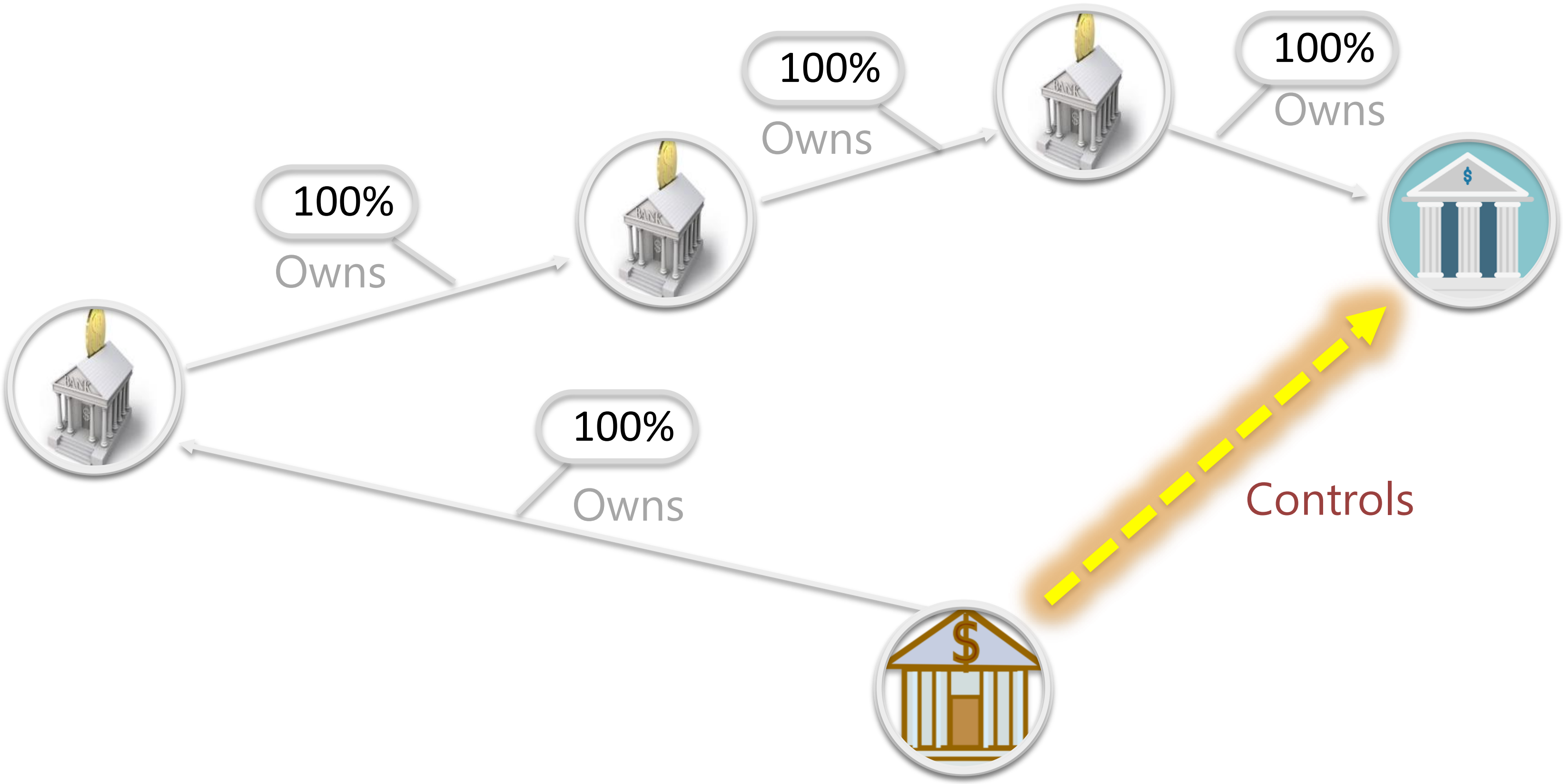


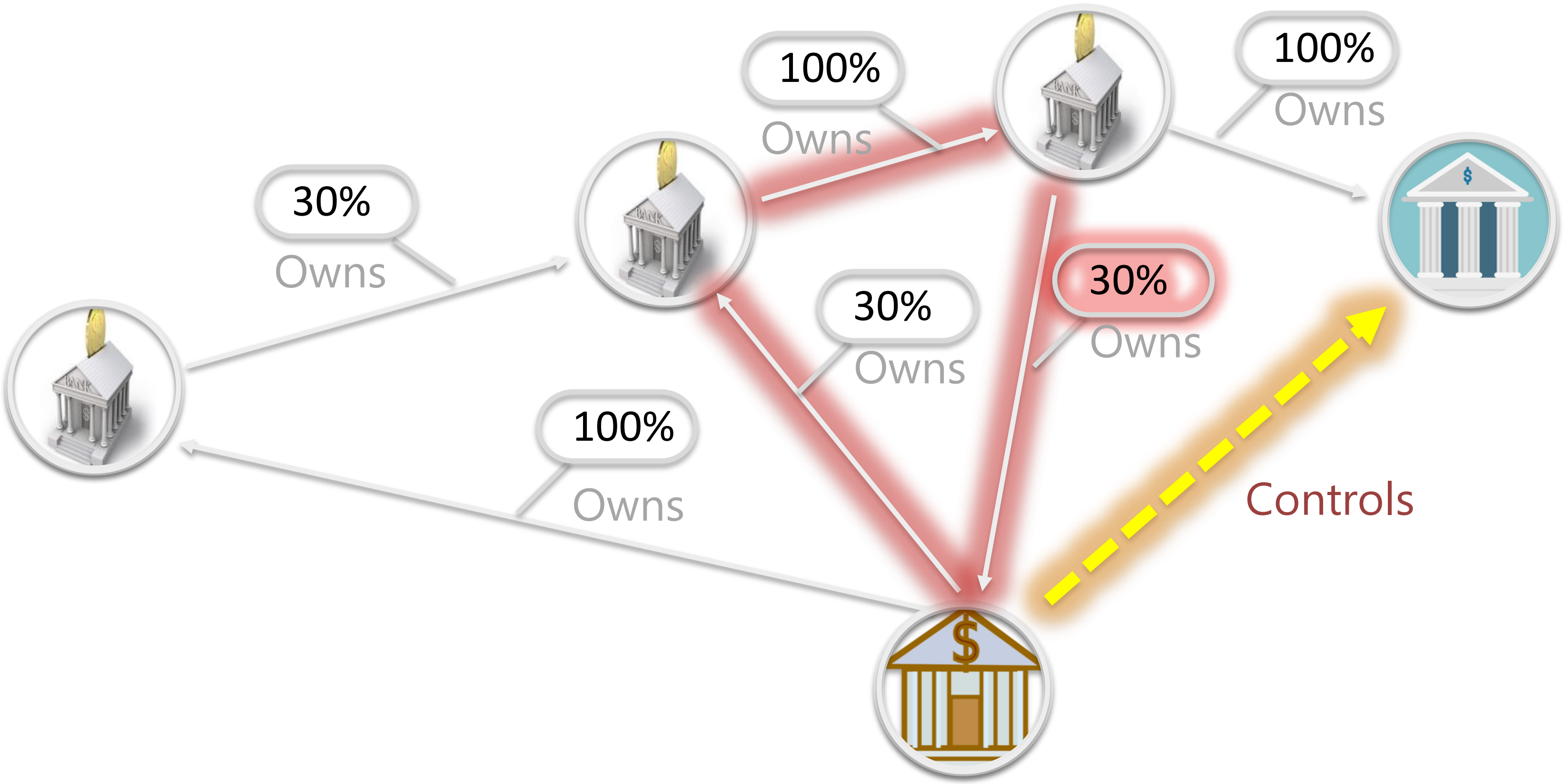
Owns
100%
transnational
Oversight
100%
hasName
Owns
isClientOf
supervises
non-USD
transactionWith
transnational
foundedIn
hasAPositionAt
1970
hasHQ
hasName
Sample Bank
City


100%
Owns
100%
Owns
Controls


100%
Owns
100%
Owns
Controls


100%
Owns
100%
Owns
100%
Owns
Controls


100%
Owns
100%
Owns
100%
Owns
100%
Owns
Controls

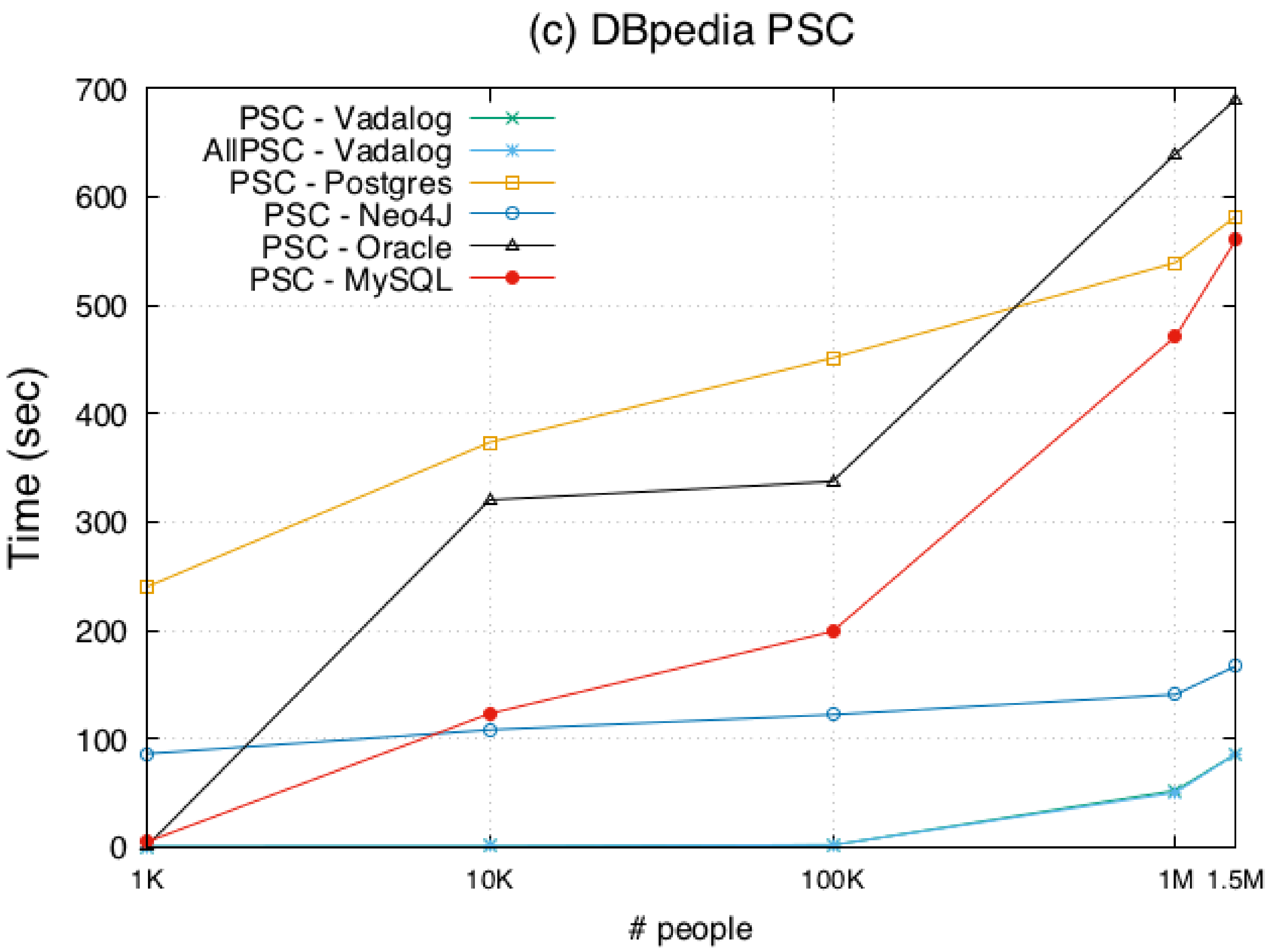

100%
Owns
100%
Owns
30%
Owns
30%
Owns
30%
Owns
100%
Owns
Controls

x **controls** y if

x **directly holds** over 50% of y, or

x **controls a set** S of companies that **jointly control** y

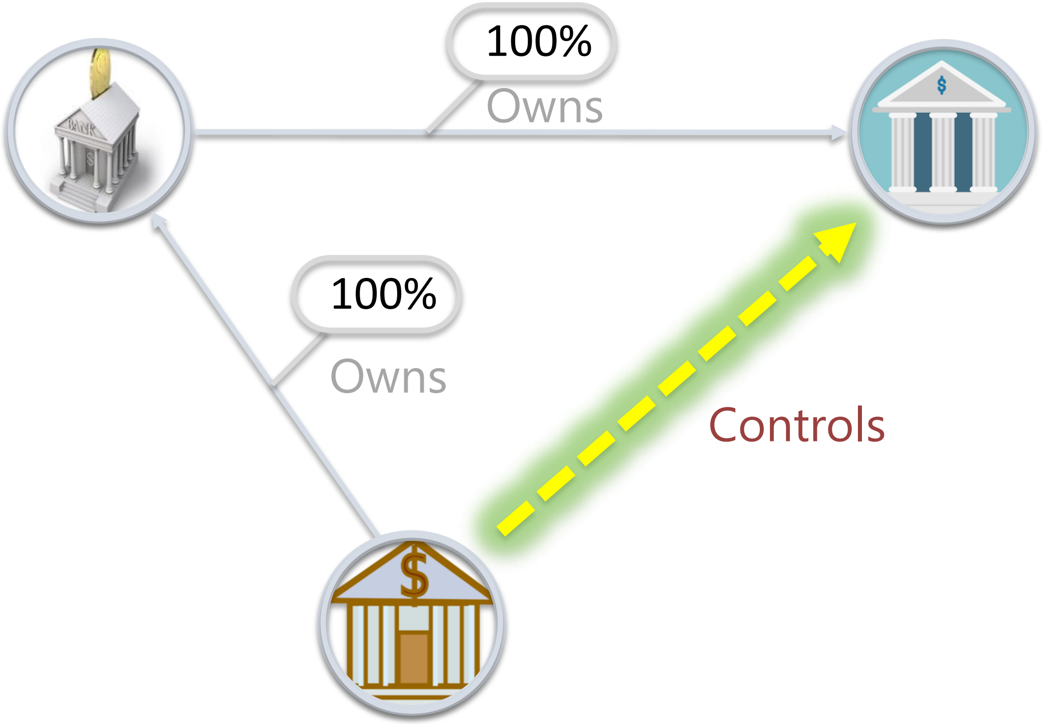
```
control(X,Y) :- own(X,Y,W), W > 0.5.
control(X,Z) :- control(X,Y), own(Y,Z,W),
                V=msum(W,<Y>), V > 0.5.
```

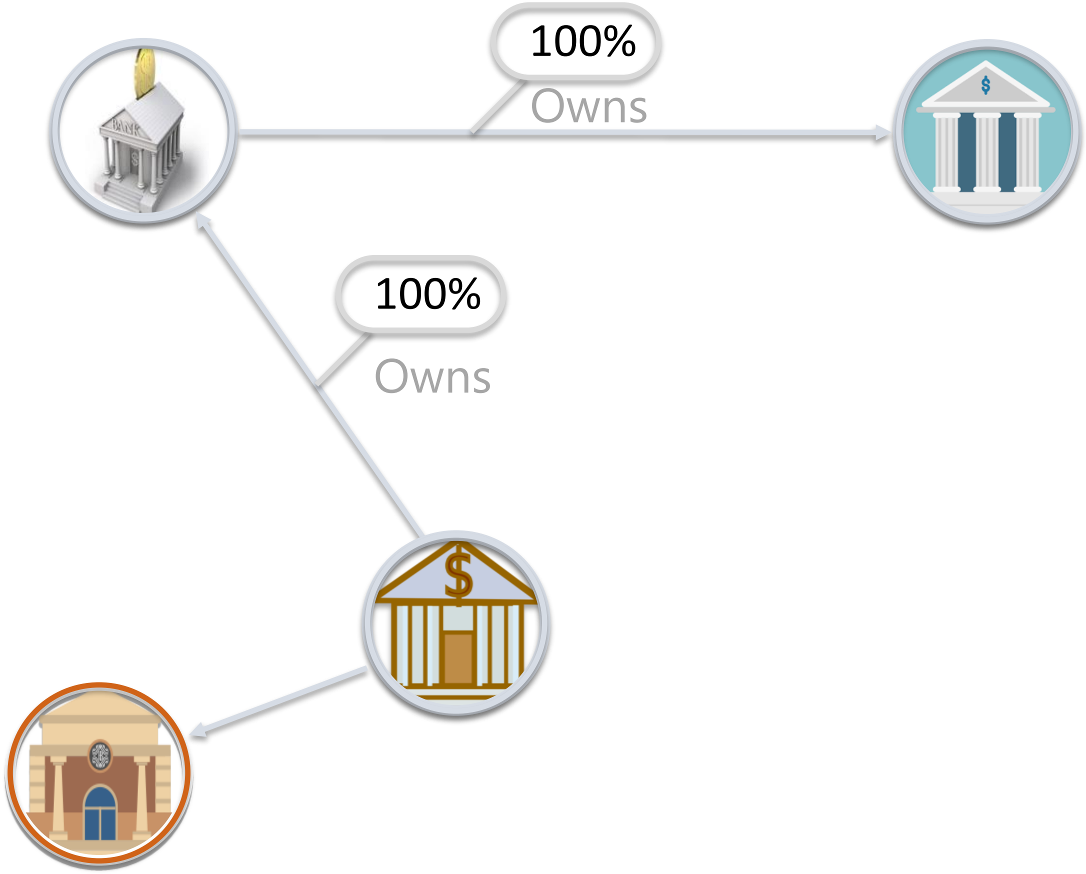

(c) DBpedia PSC
PSC - Vadalog
AllPSC - Vadalog
PSC - Postgres
PSC - Neo4J
PSC - Oracle
PSC - MySQL
Time (sec)
0
100
200
300
400
500
600
700
1K
10K
100K
1M
1.5M
people

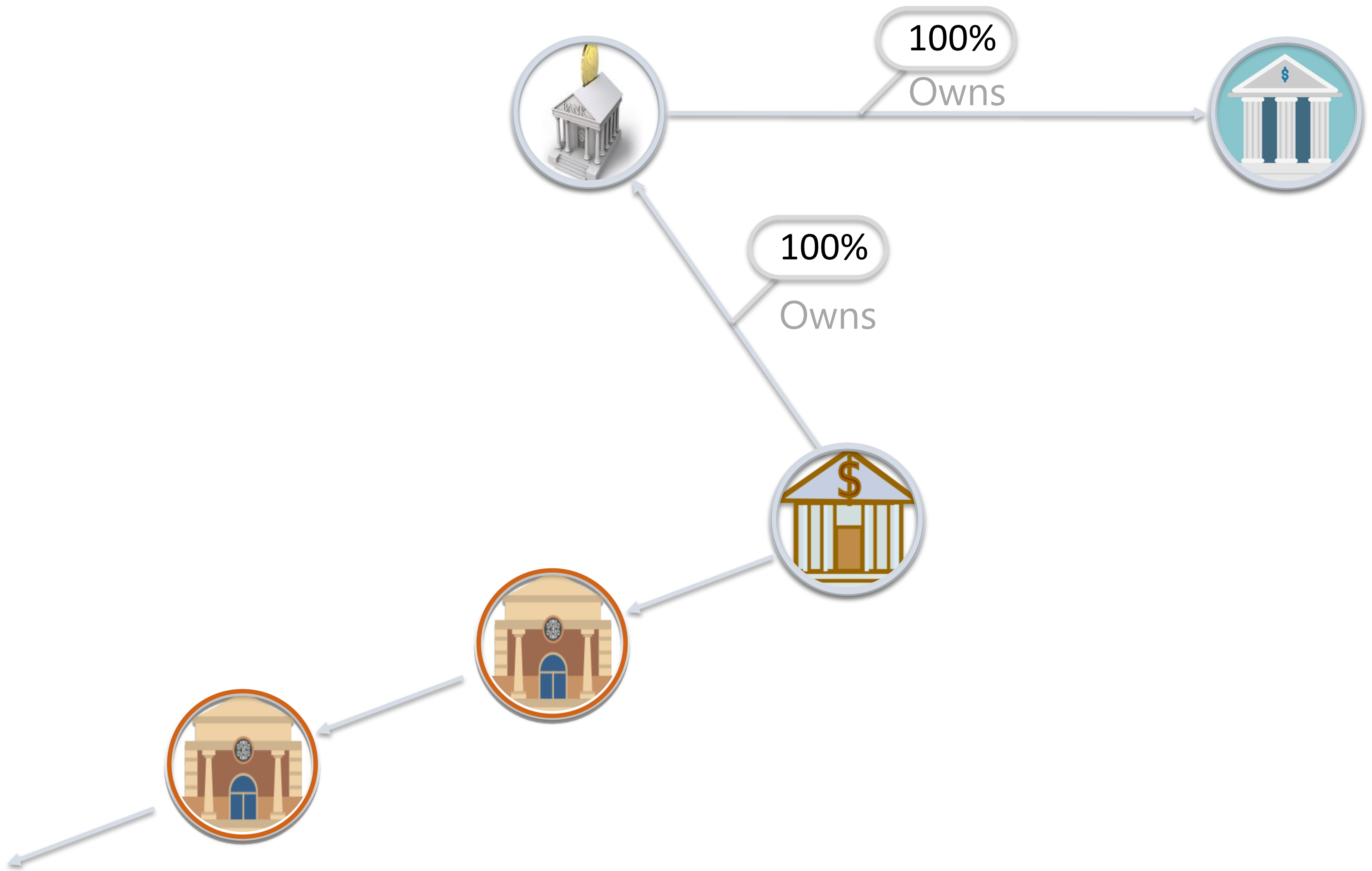
Main Challenges

1. **Recursion**
   unlimited graph exploration

2. **Object Creation**
   exploring unknown parts of the KG


100%
Owns
100%
Owns
Controls


100%
Owns
100%
Owns


100%
Owns
100%
Owns

x **controls** y if

x **directly holds** over 50% of y, or

x **controls a set** S of companies that **jointly control** y

```
control(X,Y) :- own(X,Y,W), W > 0.5.
control(X,Z) :- control(X,Y), own(Y,Z,W),
                V=msum(W,<Y>), V > 0.5.
```

Main Challenges

1. **Recursion**
   unlimited graph exploration

2. **Object Creation**
   exploring unknown parts of the KG

Query answering under
recursive Datalog with object creation
(existential quantification) is **undecidable**.

Logical Knowledge in KGs

Challenges: Recursion and Creation

Emanuel Sallinger

Logical Knowledge in KGs

Recursion in Datalog

Primer

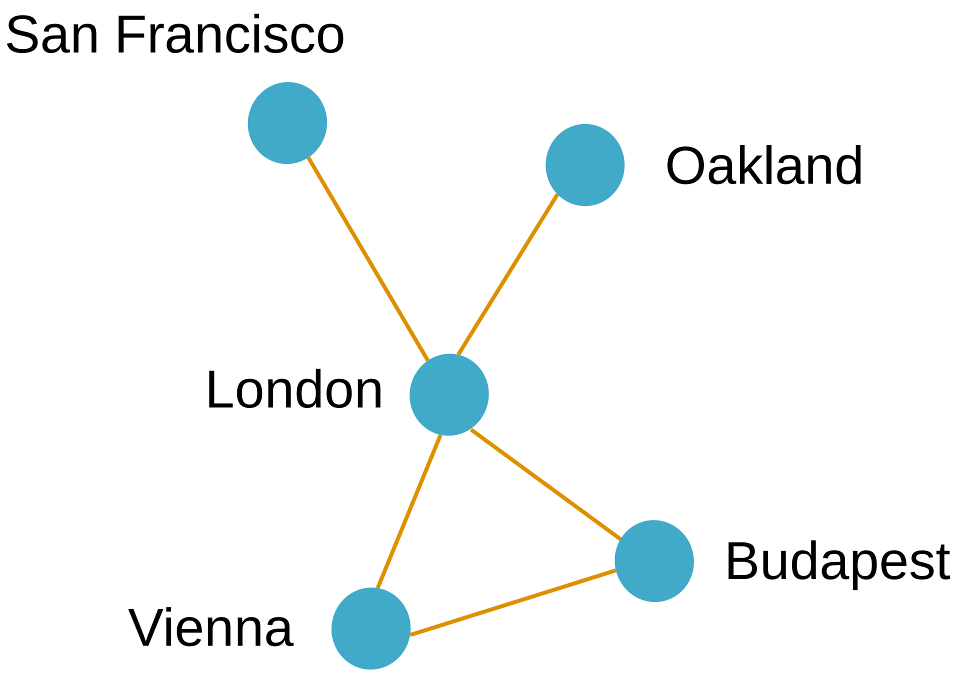
Emanuel Sallinger

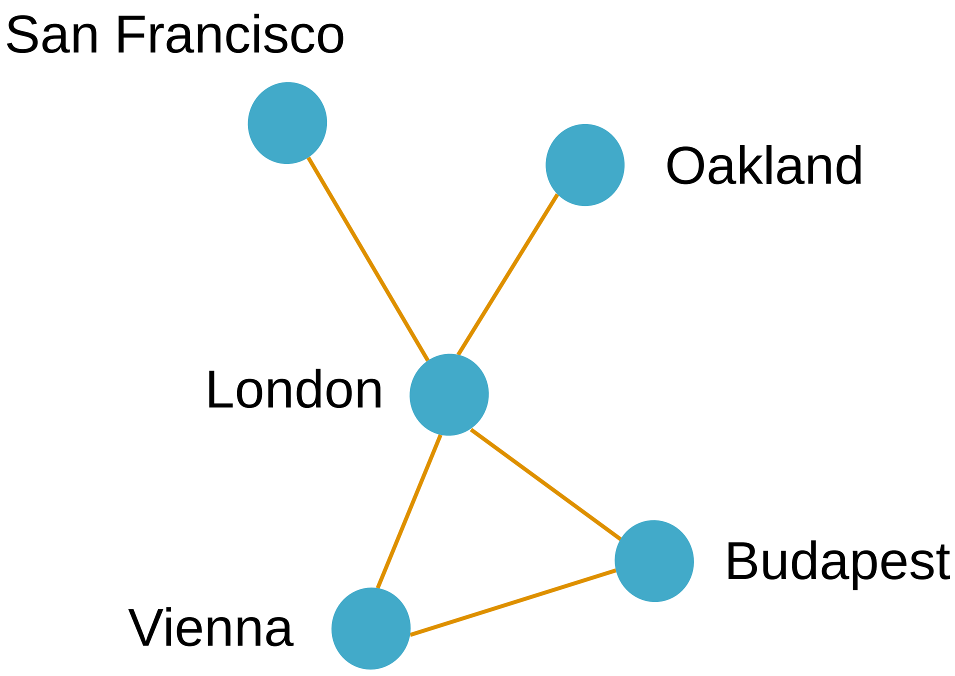
SQL

```
WITH RECURSIVE temp AS (
   SELECT * FROM flight
UNION ALL
   SELECT t.Origin, f.Destination
   FROM temp t, flight f
   WHERE t.Destination = f.Origin
)
SELECT * INTO connection FROM temp
```

Datalog

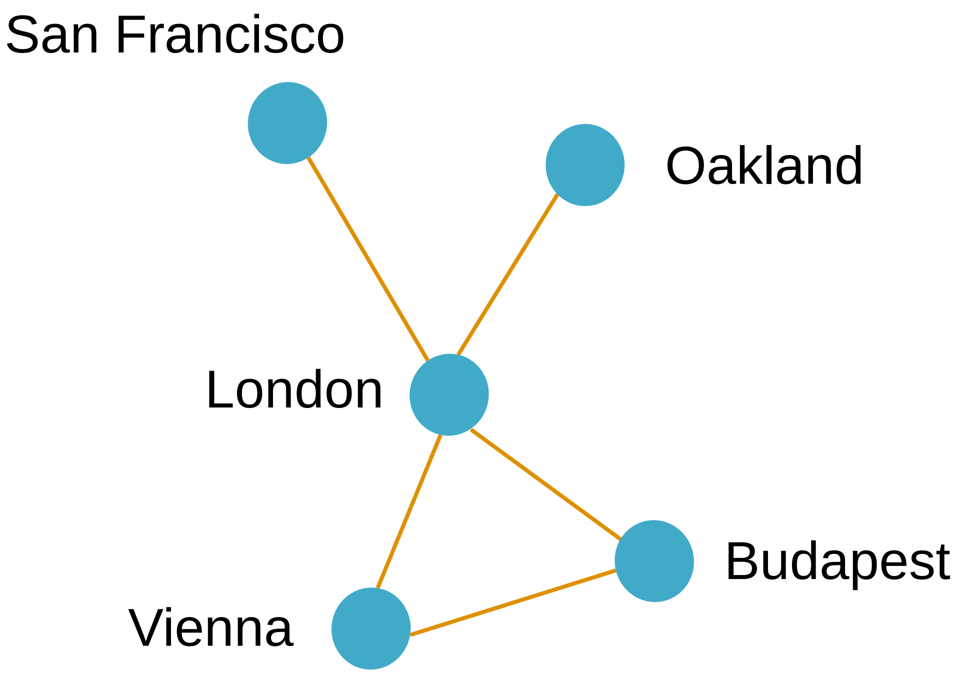
```
connection(X,Y) :- flight(X,Y,_).
connection(X,Z) :- connection(Y,Z), flight(X,Y,_).
```


San Francisco
Oakland
London
Budapest
Vienna



airport

IATA	City
VIE	Vienna
LHR	London
LGW	London
BUD	Budapest
OAK	Oakland
SFO	San Francisco



flight

Origin	Destination	Airline
VIE	LHR	BA
LHR	SFO	BA
LGW	OAK	DI
BUD	VIE	OS

airport

IATA	City
VIE	Vienna
LHR	London
LGW	London
BUD	Budapest
OAK	Oakland
SFO	San Francisco

airport

IATA	City
VIE	Vienna
LHR	London
SFO	San Francisco

→

city

City
Vienna
London
San Francisco

airport

IATA	City
VIE	Vienna
LHR	London
SFO	San Francisco

city

City
Vienna
London
San Francisco

SQL

```
SELECT City INTO city FROM airport
```

airport

IATA	City
VIE	Vienna
LHR	London
SFO	San Francisco

city

City
Vienna
London
San Francisco

SQL

```
SELECT City INTO city FROM airport
```

Datalog

```
city(City) :- airport(IATA,City).
```

```
city(City) :- airport(IATA,City).
```

$$\underbrace{\texttt{city(City)}}_{\text{head}}\ \texttt{:-}\ \underbrace{\texttt{airport(IATA,City).}}_{\text{body}}$$

head body

```
city(City) :- airport(IATA,City).
```

then if

$$\overbrace{\texttt{city(City)}}^{\text{head}}\ \texttt{:-}\ \overbrace{\texttt{airport(IATA,City).}}^{\text{body}}$$

SELECT

FROM

```
city(City) :- airport(IATA,City).
```

```
city(City) :- airport(IATA,City).
```

table (or relation, edge, node, ML, etc.)

```
city(City) :- airport(IATA,City).
```

table (or relation, edge, node, ML, etc.)

```
SELECT City INTO city FROM airport
```

variable (not necessarily attribute name)

```
city(City) :- airport(IATA,City).
```

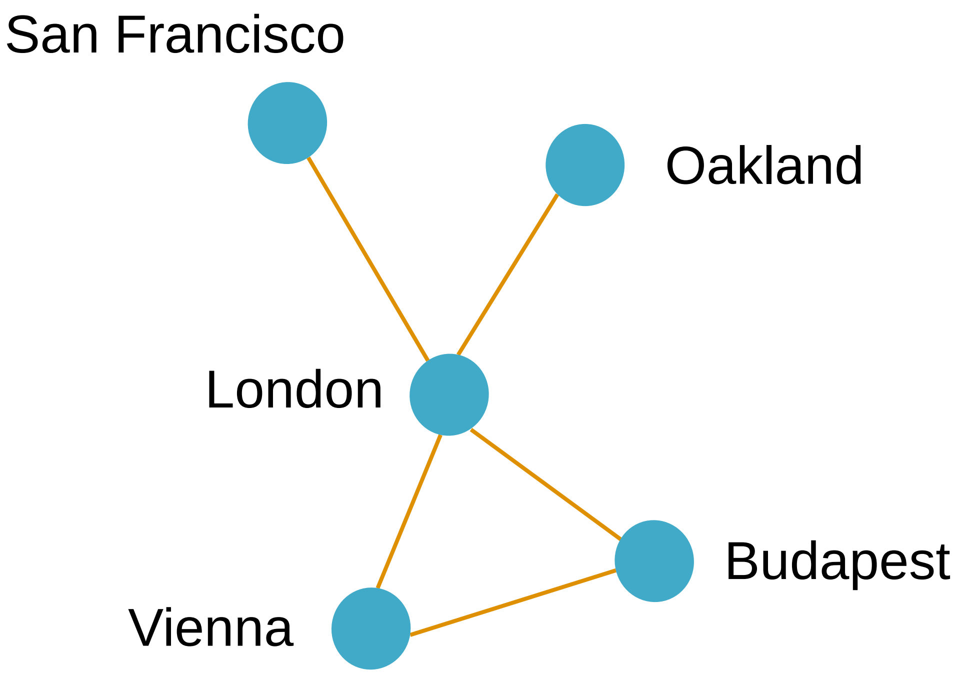
table (or relation, edge, node, ML, etc.)

```
SELECT City INTO city FROM airport
```

variable (not necessarily attribute name)

```
city( X  ) :- airport( X  , Y  ).
```

table (or relation, edge, node, ML, etc.)

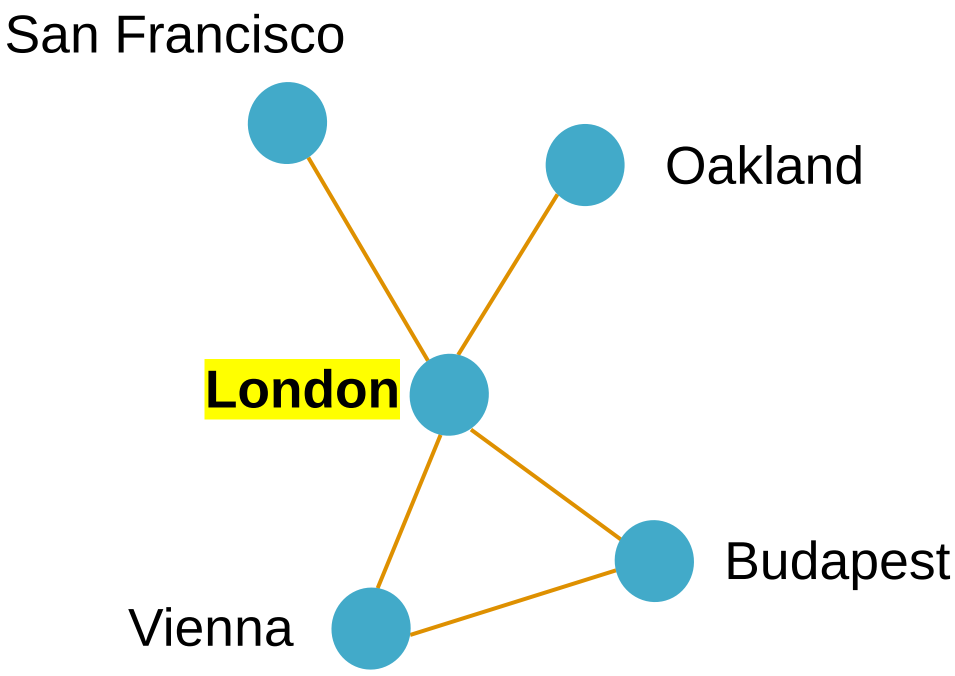


flight

Origin	Destination	Airline
VIE	LHR	BA
LHR	SFO	BA
LGW	OAK	DI
BUD	VIE	OS

airport

IATA	City
VIE	Vienna
LHR	London
LGW	London
BUD	Budapest
OAK	Oakland
SFO	San Francisco

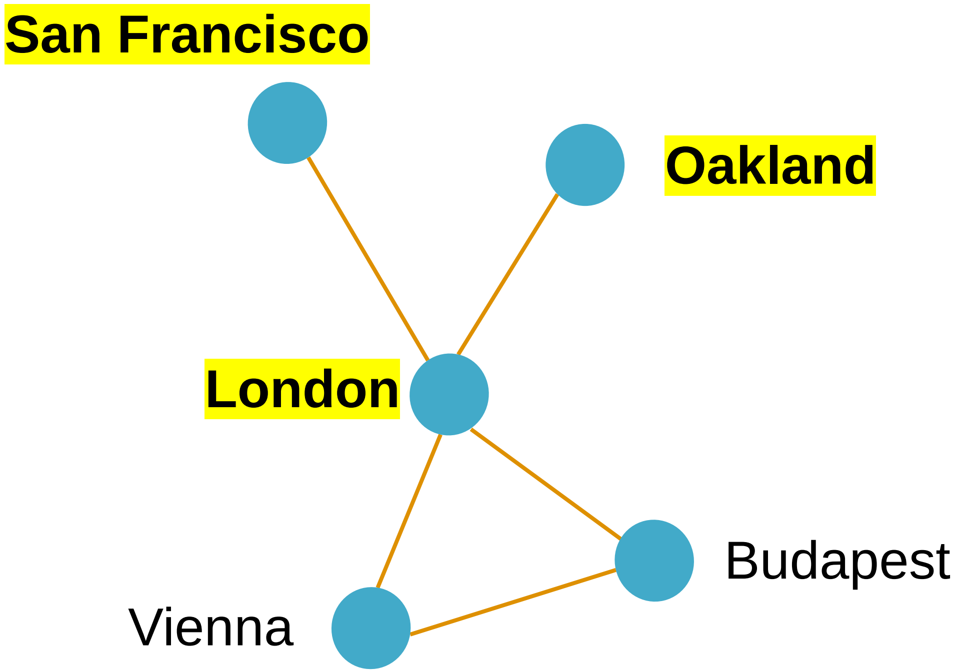


flight

Origin	Destination	Airline
VIE	LHR	BA
LHR	SFO	BA
LGW	OAK	DI
BUD	VIE	OS

airport

IATA	City
VIE	Vienna
LHR	**London**
LGW	**London**
BUD	Budapest
OAK	Oakland
SFO	San Francisco



flight

Origin	Destination	Airline
VIE	LHR	BA
LHR	SFO	BA
LGW	OAK	DI
BUD	VIE	OS

airport

IATA	City
VIE	Vienna
LHR	London
LGW	London
BUD	Budapest
OAK	Oakland
SFO	San Francisco

join

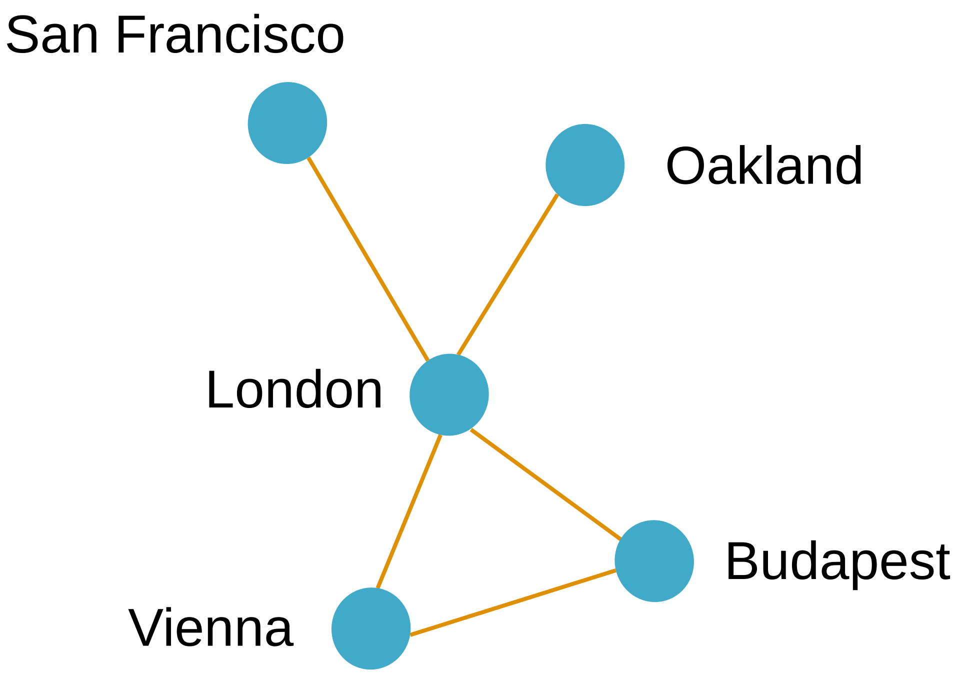
```
dest(D) :- flight(L,D,A), airport(L,"London").
```

```
SELECT f.Destination INTO dest
FROM flight f, airport a
WHERE f.Origin = a.IATA AND a.City = "London"
```

join

```
dest(D) :- flight(L,D,A), airport(L,City),
           City = "London".
```

```
SELECT f.Destination INTO dest
FROM flight f, airport a
WHERE f.Origin = a.IATA AND a.City = "London"
```

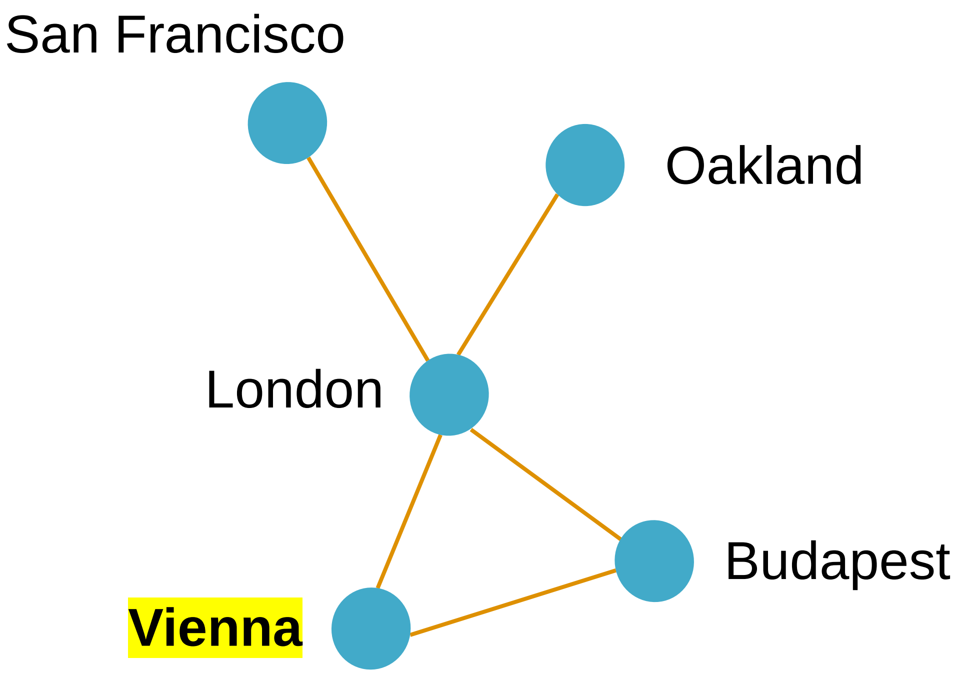


flight

Origin	Destination	Airline
VIE	LHR	BA
LHR	SFO	BA
LGW	OAK	DI
BUD	VIE	OS

airport

IATA	City
VIE	Vienna
LHR	London
LGW	London
BUD	Budapest
OAK	Oakland
SFO	San Francisco

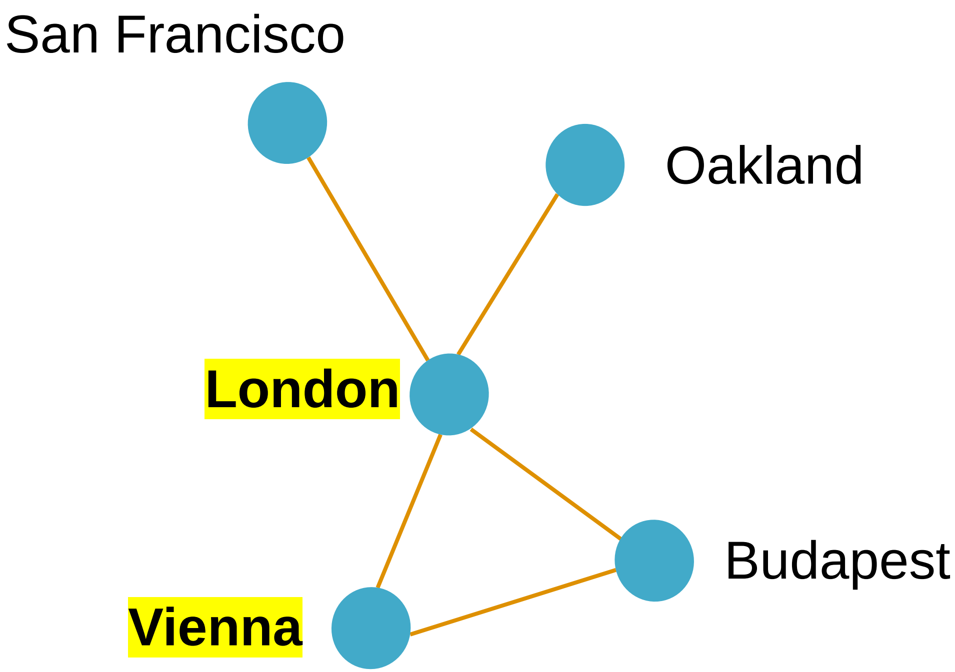


flight

Origin	Destination	Airline
VIE	LHR	BA
LHR	SFO	BA
LGW	OAK	DI
BUD	VIE	OS

airport

IATA	City
VIE	Vienna
LHR	London
LGW	London
BUD	Budapest
OAK	Oakland
SFO	San Francisco

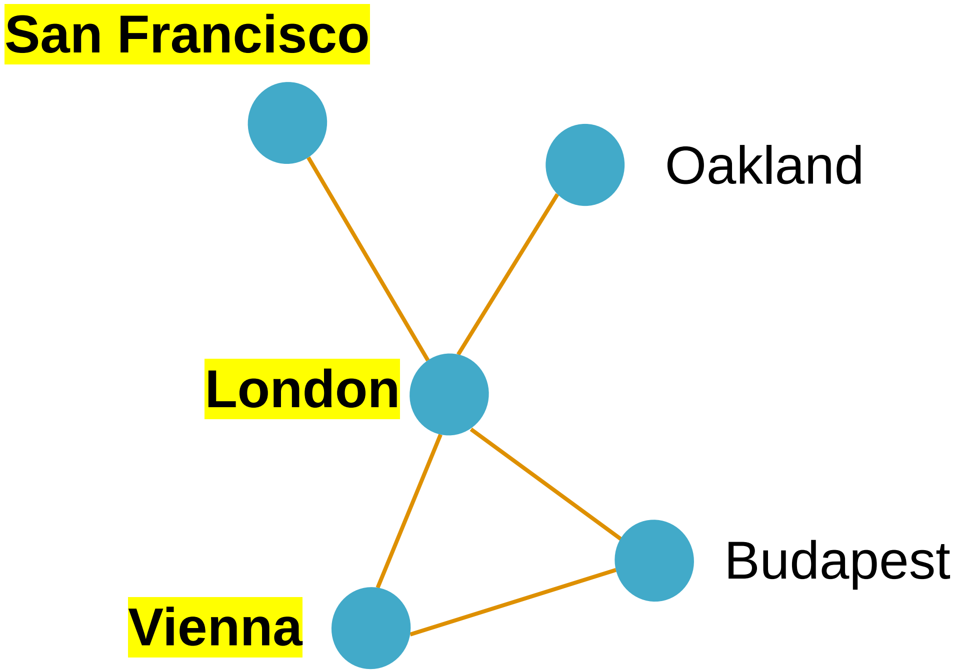


flight

Origin	Destination	Airline
VIE	LHR	BA
LHR	SFO	BA
LGW	OAK	DI
BUD	VIE	OS

airport

IATA	City
VIE	Vienna
LHR	London
LGW	London
BUD	Budapest
OAK	Oakland
SFO	San Francisco



flight

Origin	Destination	Airline
VIE	LHR	BA
LHR	SFO	BA
LGW	OAK	DI
BUD	VIE	OS

airport

IATA	City
VIE	Vienna
LHR	London
LGW	London
BUD	Budapest
OAK	Oakland
SFO	San Francisco

SQL

```
WITH RECURSIVE temp AS (
   SELECT * FROM flight
UNION ALL
   SELECT t.Origin, f.Destination
   FROM temp t, flight f
   WHERE t.Destination = f.Origin
)
SELECT * INTO connection FROM temp
```

Datalog

```
connection(X,Y) :- flight(X,Y,_).
connection(X,Z) :- connection(Y,Z), flight(X,Y,_).
```

Syntax of Datalog

A **datalog rule** is an expression of the form

$$R_1(x_1) :- R_2(x_2), \ldots, R_n(x_n).$$

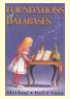
where R_i are relation names and x_i are lists of variable names.

- Each variable in x_1 must occur in at least one of $x_2, \ldots, x_n$.
- A **datalog programme** is a finite set of datalog rules.

See Abitebeoul, Hull and Vianu's
"Foundations of Databases" for more details

Semantics of Datalog

- Various semantics are possible:
  - Model-theoretic semantics
  - Fixpoint semantics
  - Proof-theoretic semantics
- Important tool: The “chase” procedure
  - Informally, starting with a database and applying rules to generate new tuples.
- More than we can cover here. For the interested:



See Abitebeoul, Hull and Vianu’s
“Foundations of Databases” for more details

Datalog

```
connection(X,Y) :- flight(X,Y,_).
connection(X,Z) :- connection(Y,Z), flight(X,Y,_).
```

First-order Logic

$$\forall X,Y,U\ (connection(X,Y) \leftarrow flight(X,Y,U))$$
$$\forall X,Y,Z,U\ connection(X,Z) \leftarrow connection(Y,Z) \wedge flight(X,Y,U)$$

Logical Knowledge in KGs
Recursion in Datalog

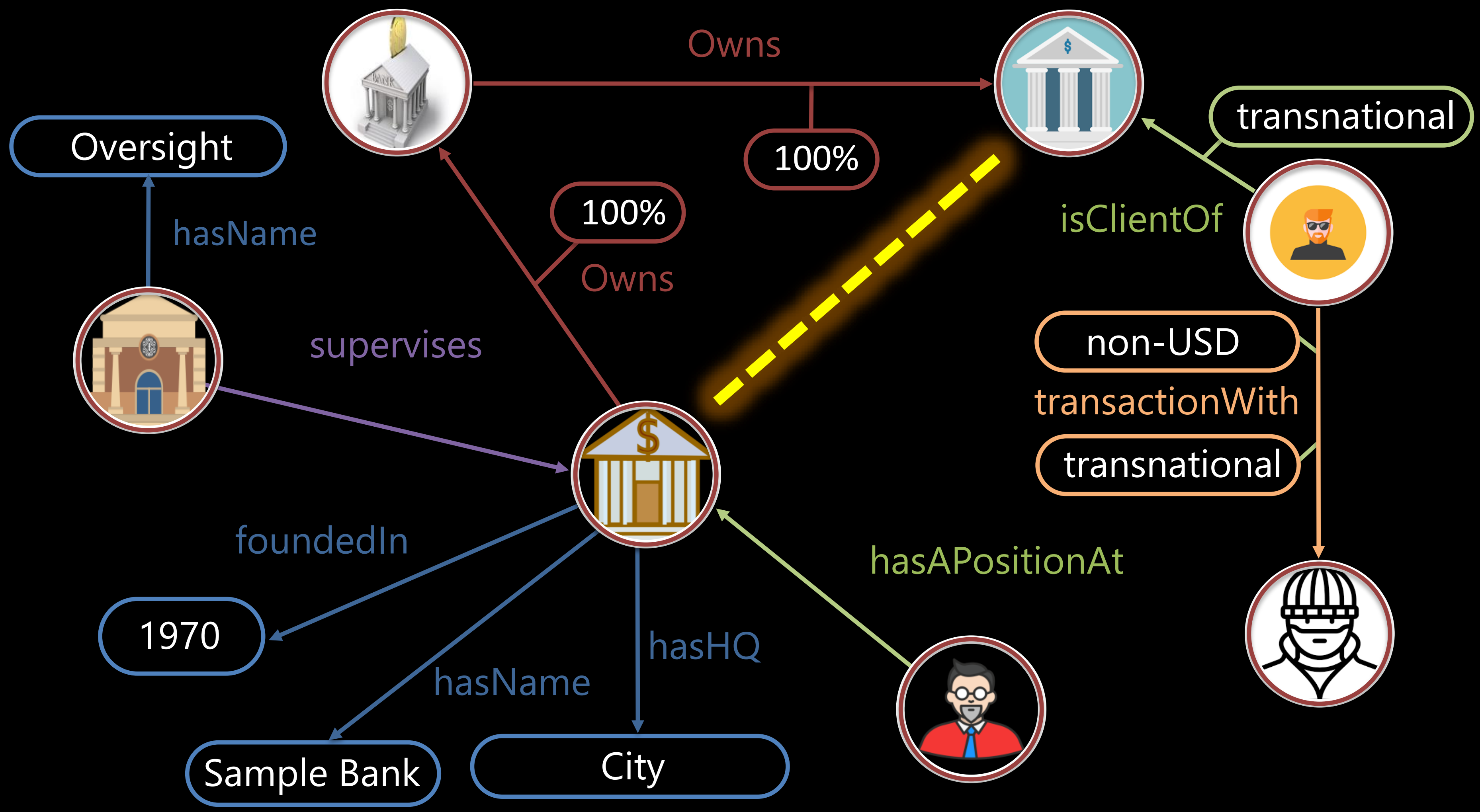
Primer

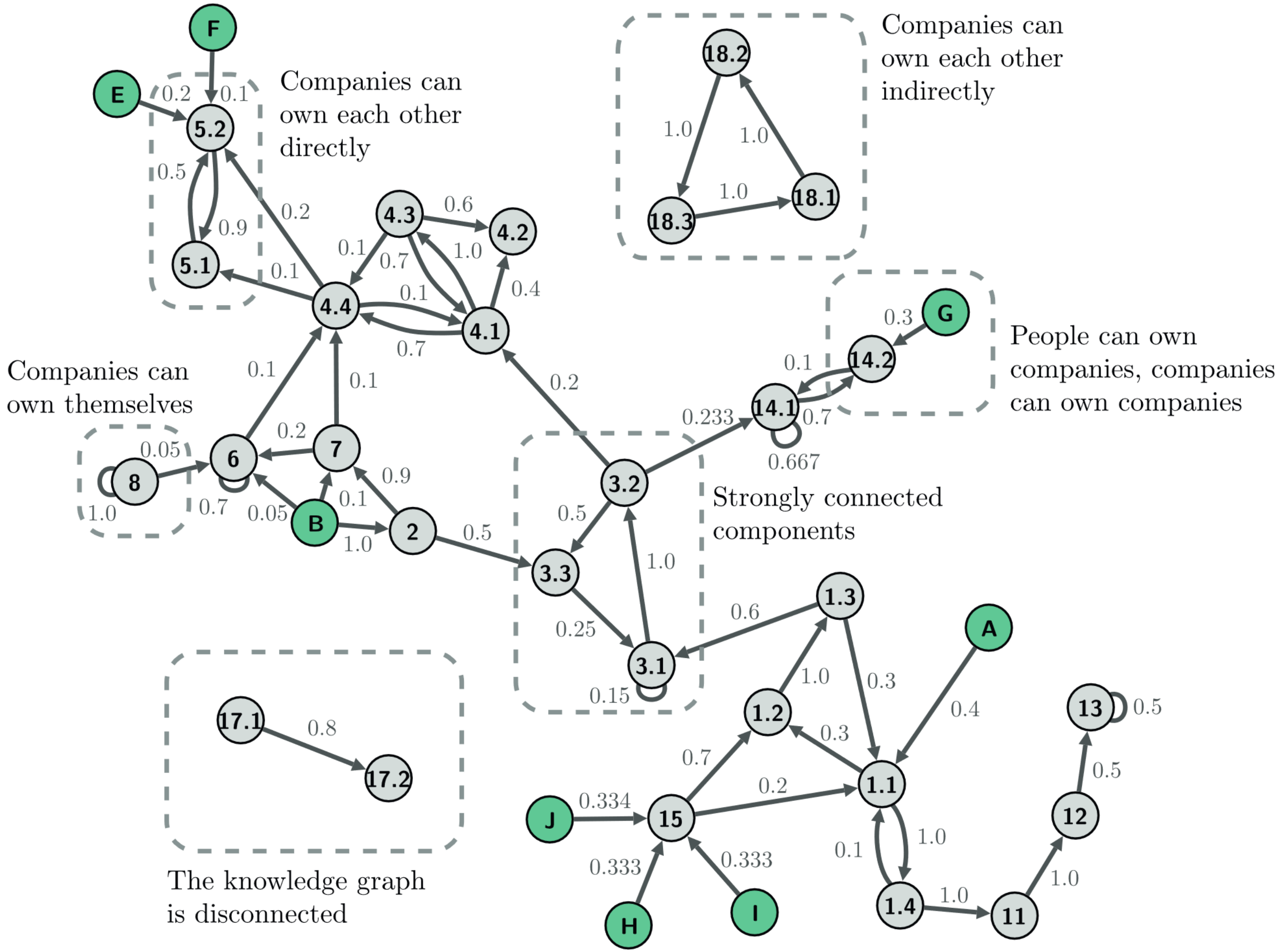
Emanuel Sallinger

Logical Knowledge in KGs

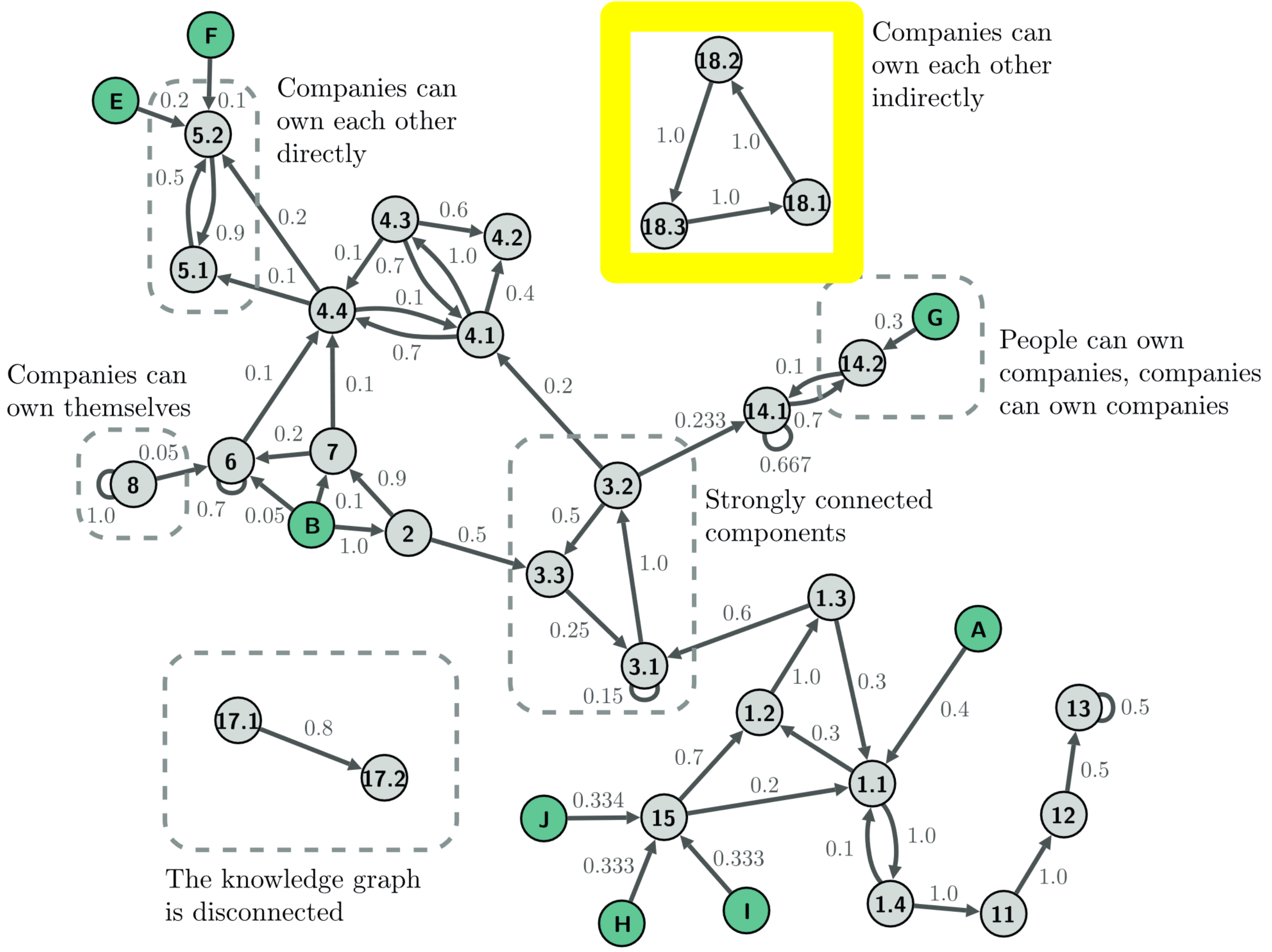
Recursion in the Real World

Emanuel Sallinger

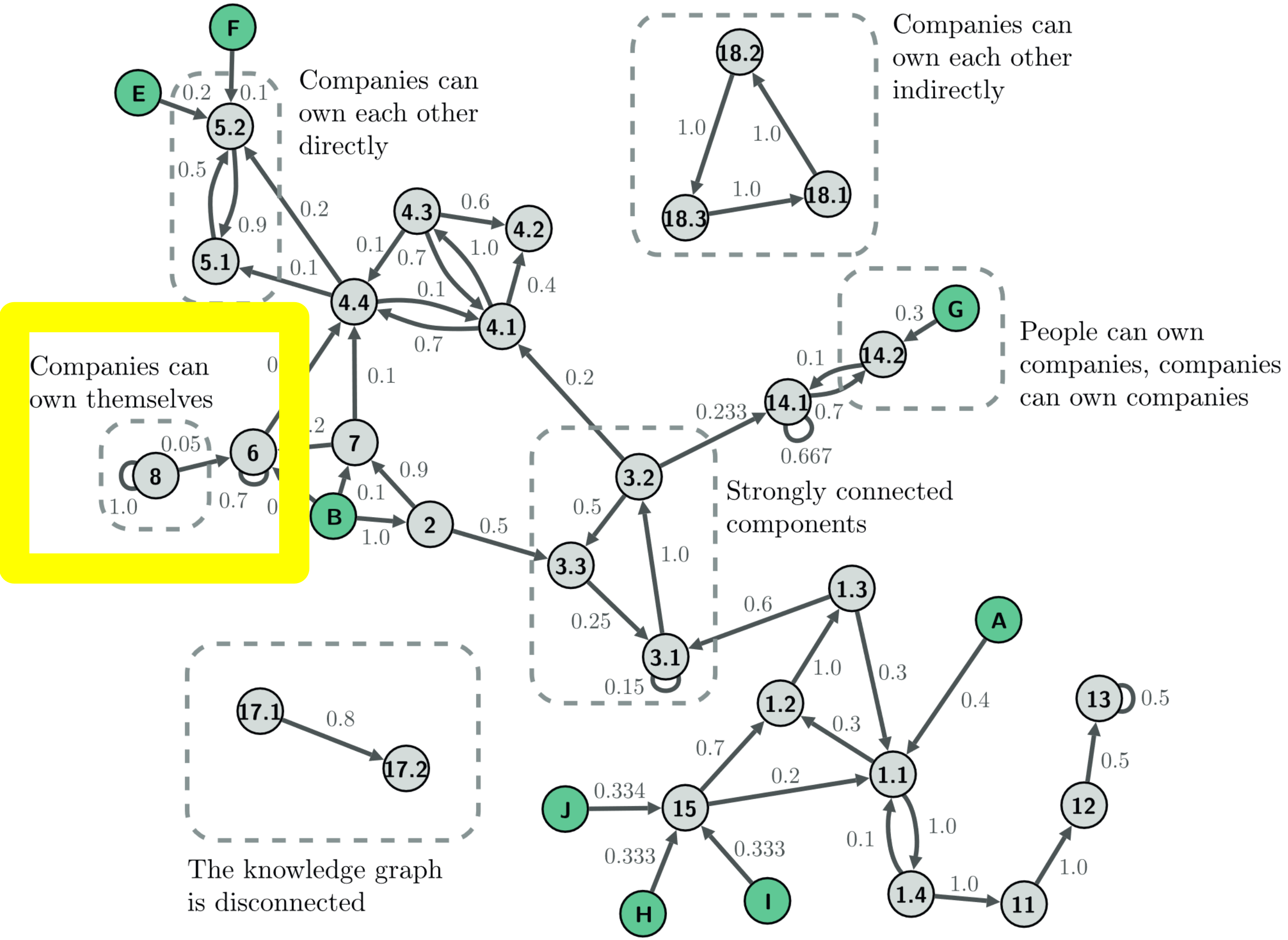

Owns
100%
transnational
Oversight
100%
hasName
Owns
isClientOf
supervises
non-USD
transactionWith
transnational
foundedIn
hasAPositionAt
1970
hasHQ
hasName
Sample Bank
City



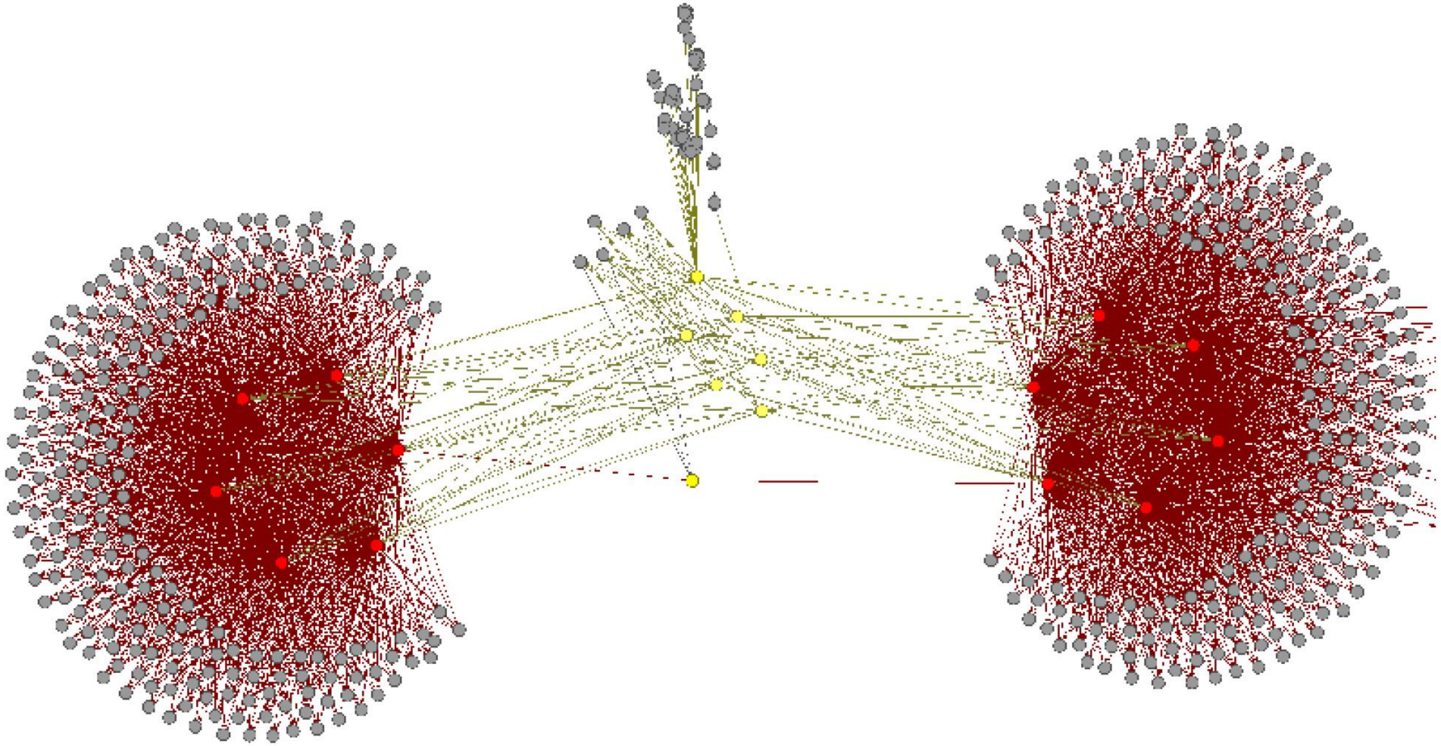
Source: https://arxiv.org/abs/2004.10119

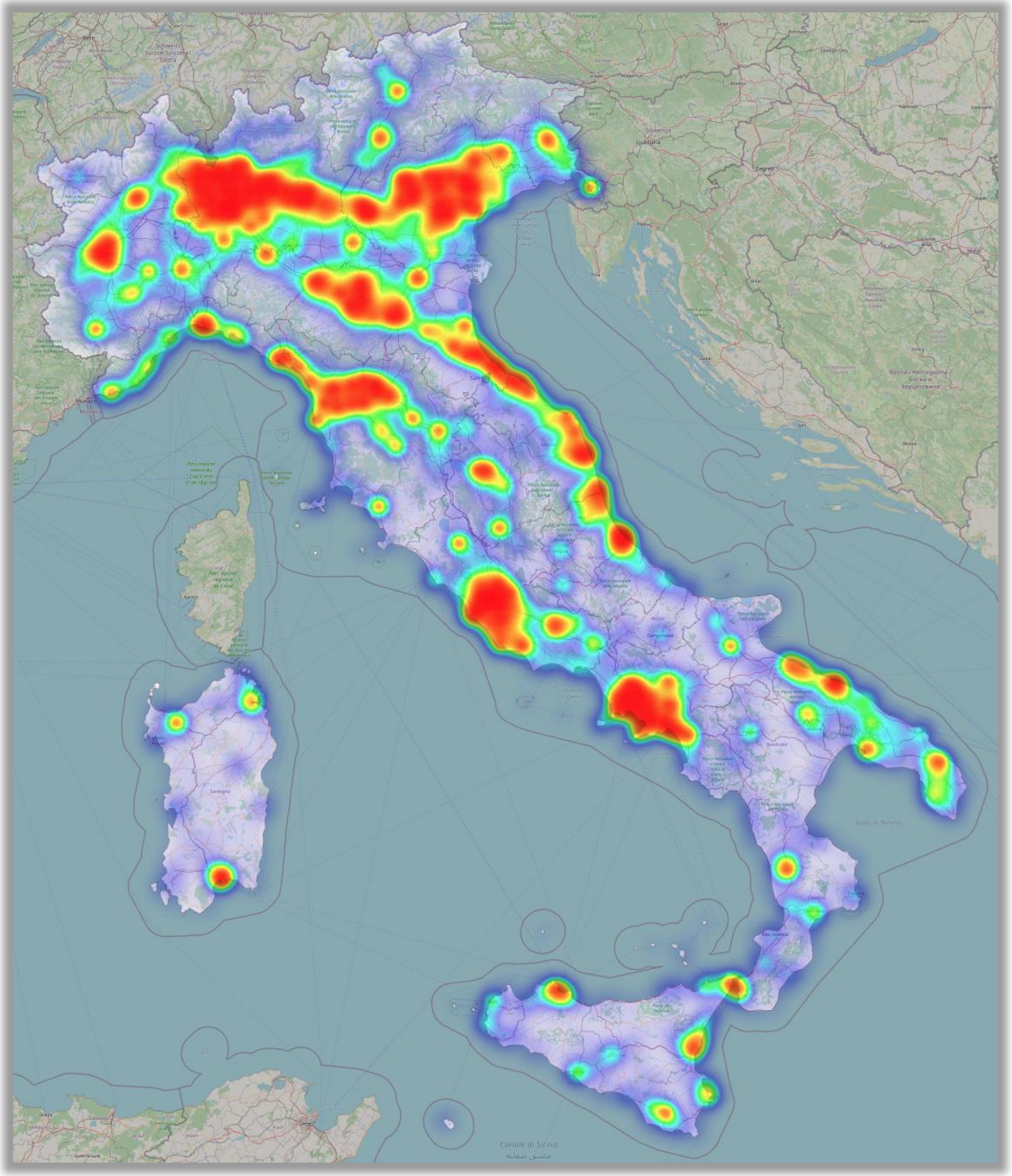


Source: https://arxiv.org/abs/2004.10119



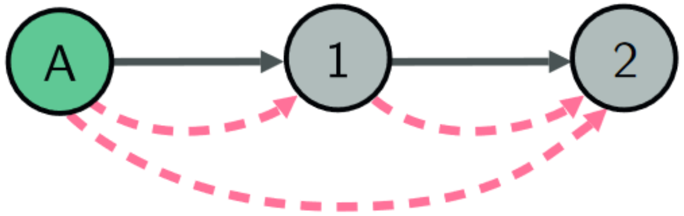
Source: https://arxiv.org/abs/2004.10119



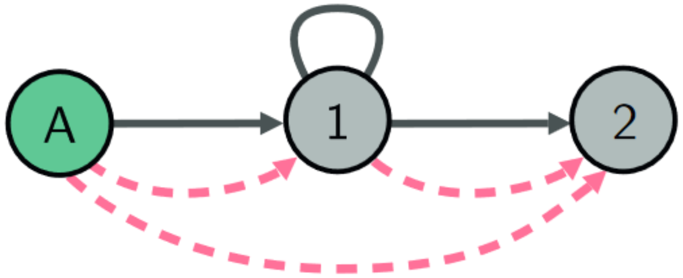


Ownership and Control

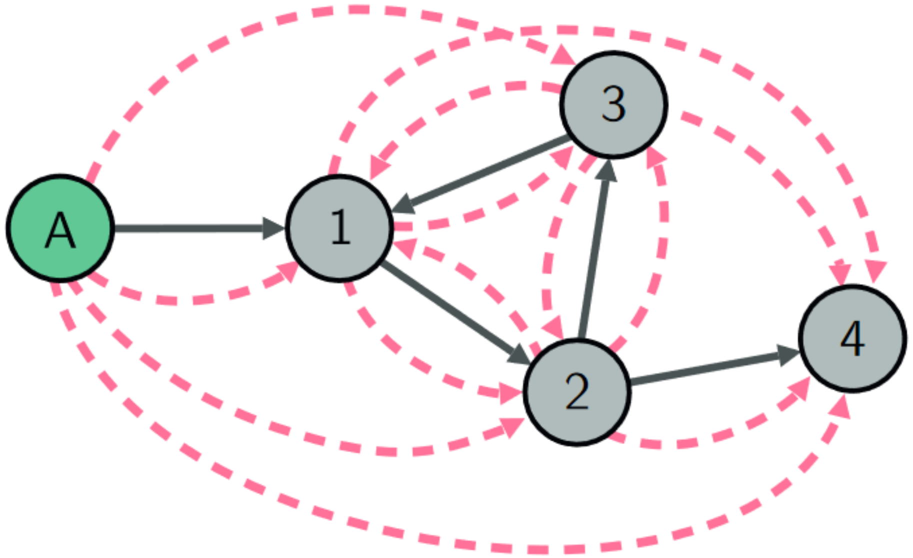
Ownership



(a) A simple indirect ownership.



(b) Indirect ownership with a self-loop.



(c) Indirect ownership with a strongly connected component.

Ownership

Definition 1. *An ϵ–Baldone path P from s to t is a path $[s, p_1, \ldots, p_n, t]$ such that $s \neq p_i$ for $i = 1, \ldots, n$ and $w(P) > \epsilon$, with $\epsilon \in \mathbb{R}^+$ and $0 < \epsilon \leq 1$. We denote the weight of an ϵ–Baldone path as $w_\epsilon(P)$.*

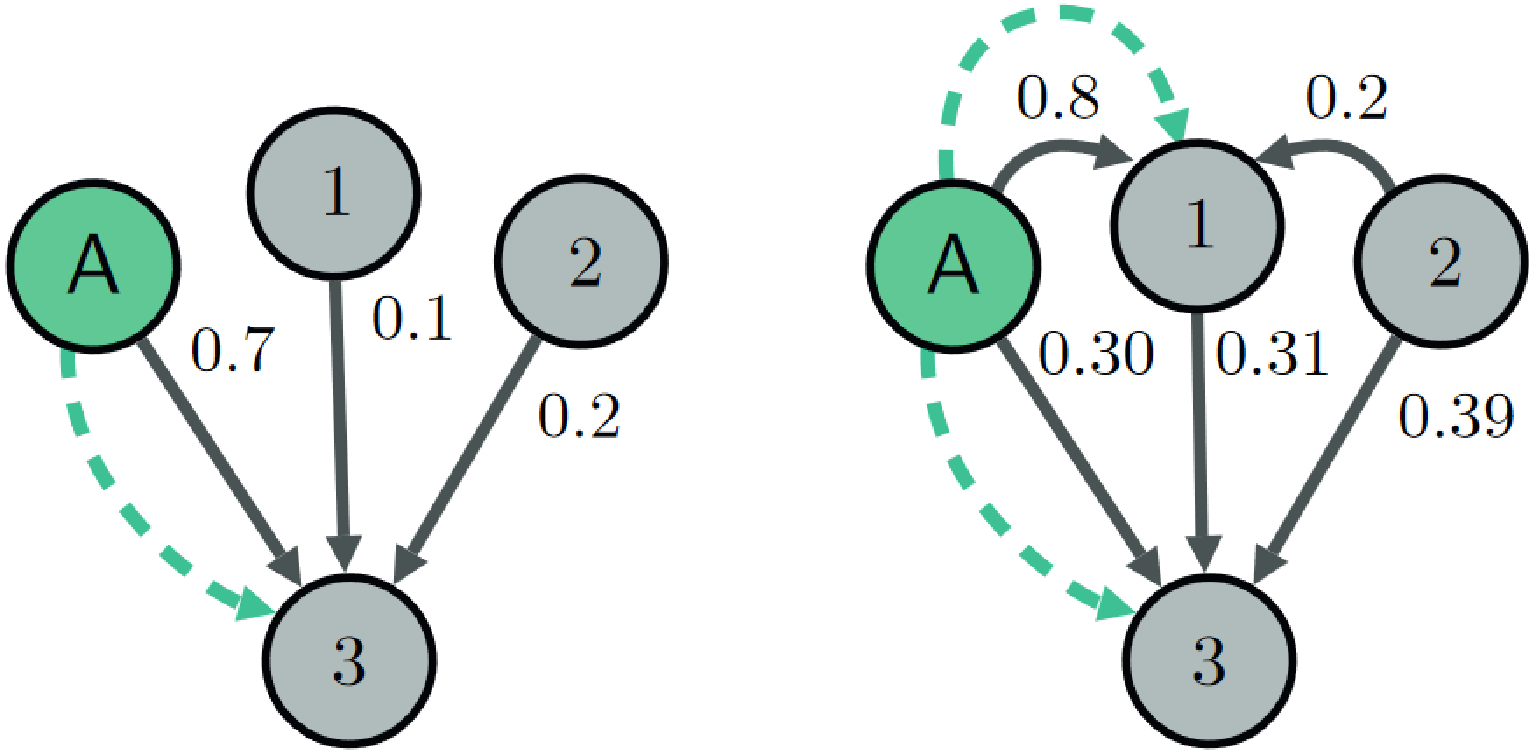
Definition 2. *The* ϵ-Baldone ownership *of a company s on a company t in a graph G is a function $O^G_\epsilon(s,t) : (N \times N) \rightarrow \mathbb{R}^+ \cup \{\infty\}$ defined as $(s,t) \rightarrow \sum_{P_i \in B_\epsilon} w_\epsilon(P_i)$, where B_ϵ is the set of all possible ϵ-Baldone paths from s to t.*

Definition 3. *The* Baldone ownership *(which we will also refer to as* integrated ownership*) of a company s on a company t in a graph G is a function $O^G(s,t)$: $(N \times N) \rightarrow \mathbb{R} \cup \{\infty\}$ defined as $(s,t) \rightarrow lim_{\epsilon \rightarrow 0} O_\epsilon(s,t)$.*

$$Own(x, y, w), w > \epsilon, v = sum(w), p = [x,y] \longrightarrow IOwn(x, y, v, p). \qquad (1)$$

$$IOwn(x, z, w_1, p_1), IOwn(z, y, w_2, p_2), p = p_1|p_2, BaldonePath(p, v, \epsilon),$$
$$v = sum(w_1 \times w_2), \longrightarrow IOwn(x, y, v, p). \qquad (2)$$

Control



(a) Direct control. (b) Indirect control.

Control

Definition 4. *A company (or a person) x* controls *a company y, if: (i) x directly owns more than 50% of y; or, (ii) x controls a set of companies that jointly (i.e., summing the shares), and possibly together with x, own more than 50% of y.*

$$Control(x) \rightarrow Control(x, x) \qquad (1)$$

$$Control(x, y), Own(y, z, w), v = msum(w, \langle y \rangle), v > 0.5 \rightarrow Control(x, z) \qquad (2)$$

Golden Power

Golden Power Check

Golden Power Check

Goal: The general goal is checking whether an acquisition (of shares, stocks, etc.) causes any strategic Italian company to become controlled by a foreign company.

Setting: Let S be a set of strategic companies and F be a set of foreign companies. Let t be a transaction (e.g., an offer issued by a company x to buy an amount s of shares of a company y), with $x, y \in S \cup F$.

Question: Decide whether t causes any company in F to control a company in S.

Insight: Consider exerting "golden power" to block t.

Golden Power Check

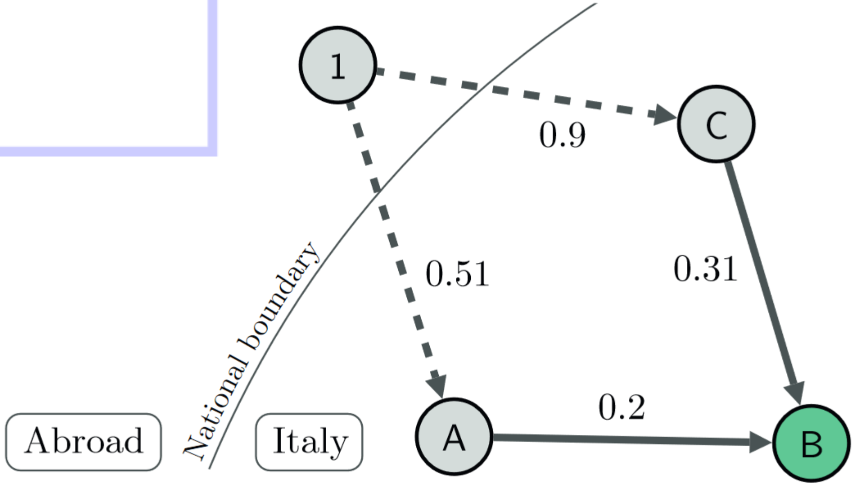
Golden Power Check

Goal: The general goal is checking whether an acquisition (of shares, stocks, etc.) causes any strategic Italian company to become controlled by a foreign company.

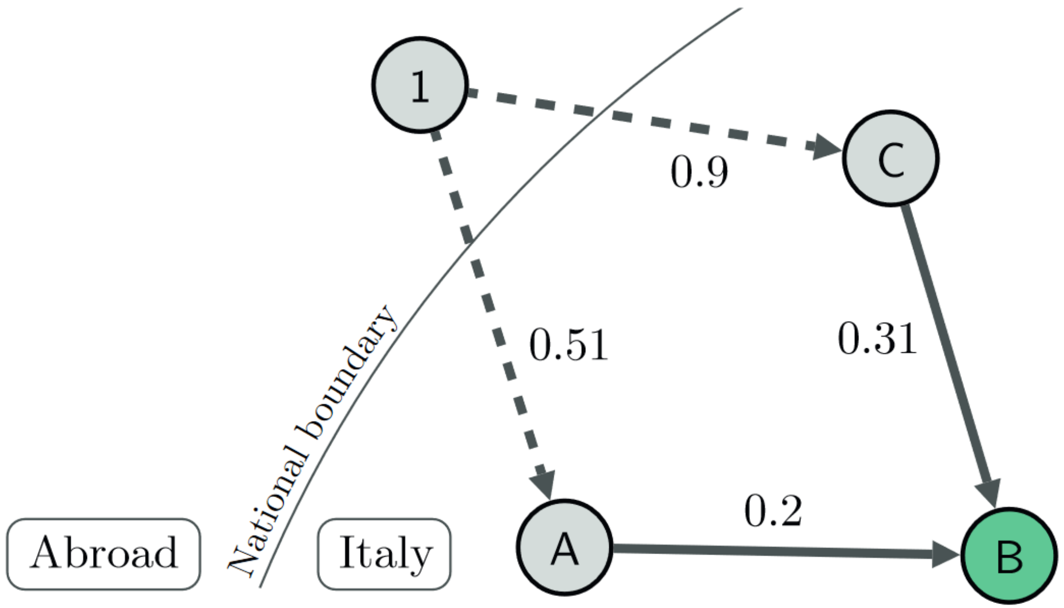
Setting: Let S be a set of strategic companies and F be a set of foreign companies. Let t be a transaction (e.g., an offer issued by a company x to buy an amount s of shares of a company y), with $x, y \in S \cup F$.

Question: Decide whether t causes any company in F to control a company in S.

Insight: Consider exerting "golden power" to block t.



Golden Power Check

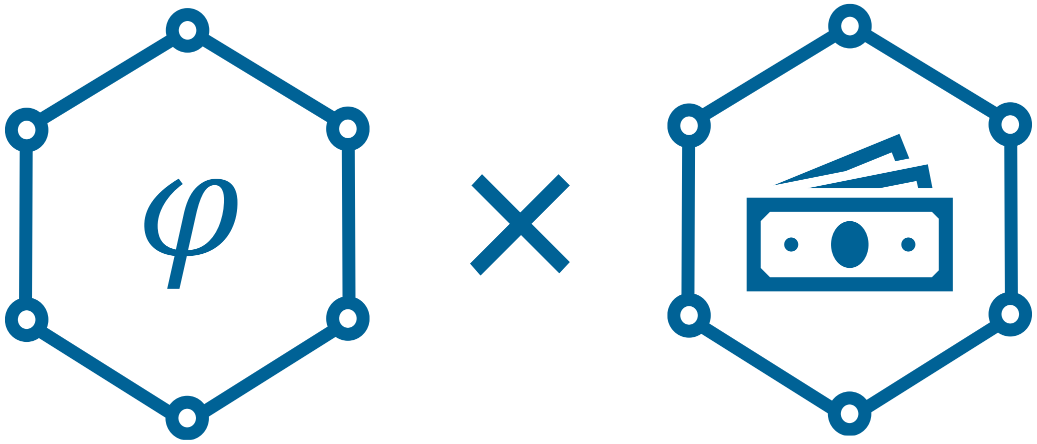


$$T(x, y, w) \rightarrow Own(x, y, w) \tag{1}$$

$$Control(x, y) \rightarrow Control(x, x) \tag{2}$$

$$Control(x, y), Own(y, z, w), v = msum(w, \langle y \rangle), v > 0.5 \rightarrow Control(x, z) \tag{3}$$

$$F(x), S(y), Control(x, y) \rightarrow GPCheck(x, y) \tag{4}$$



Logical Knowledge in KGs

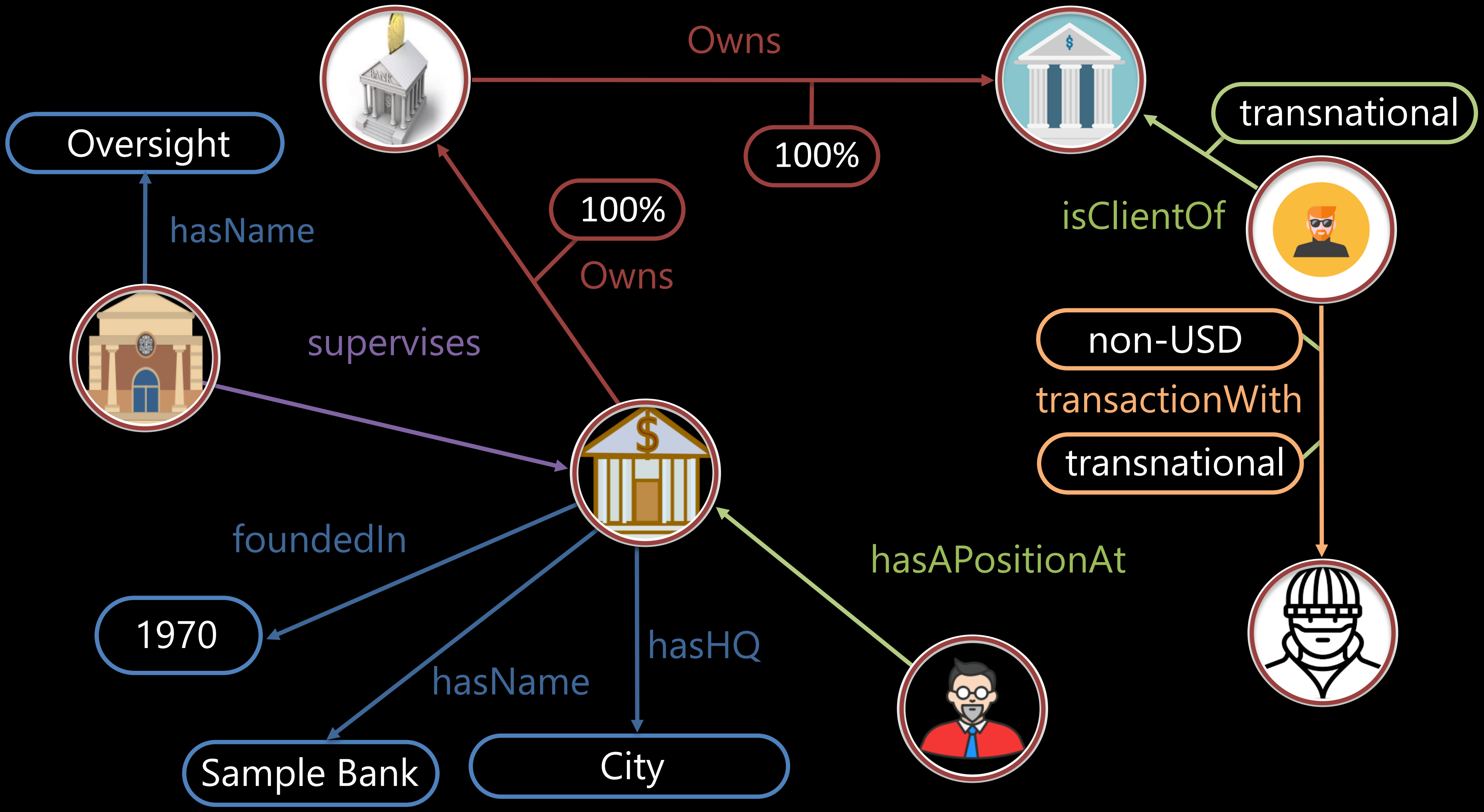
Recursion in the Real World

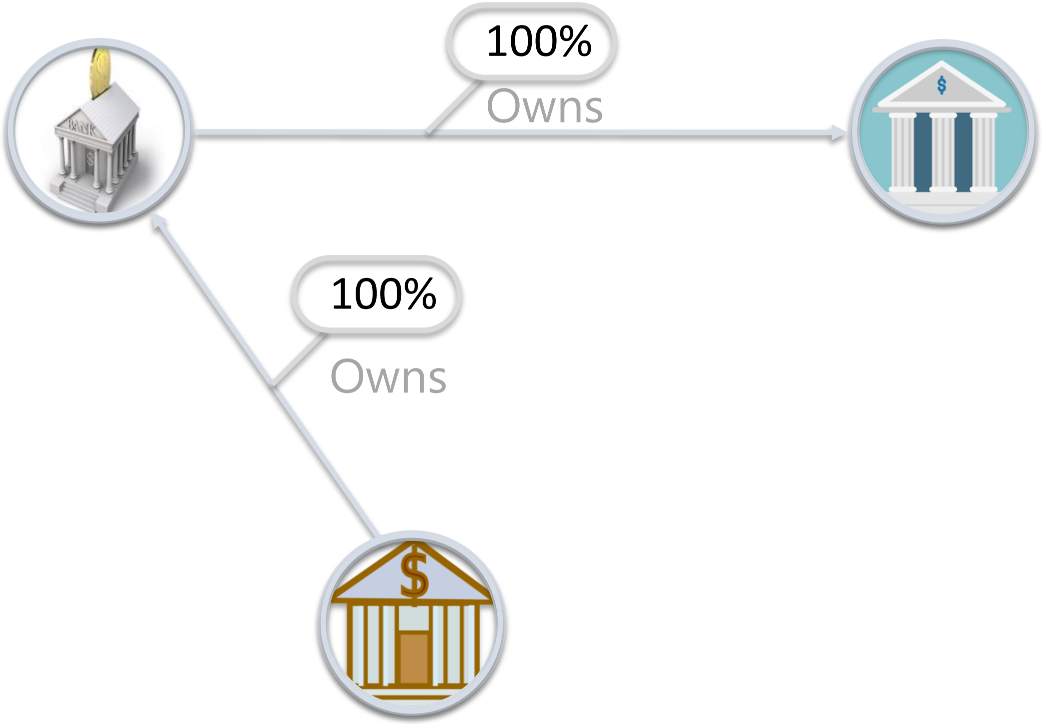
Emanuel Sallinger

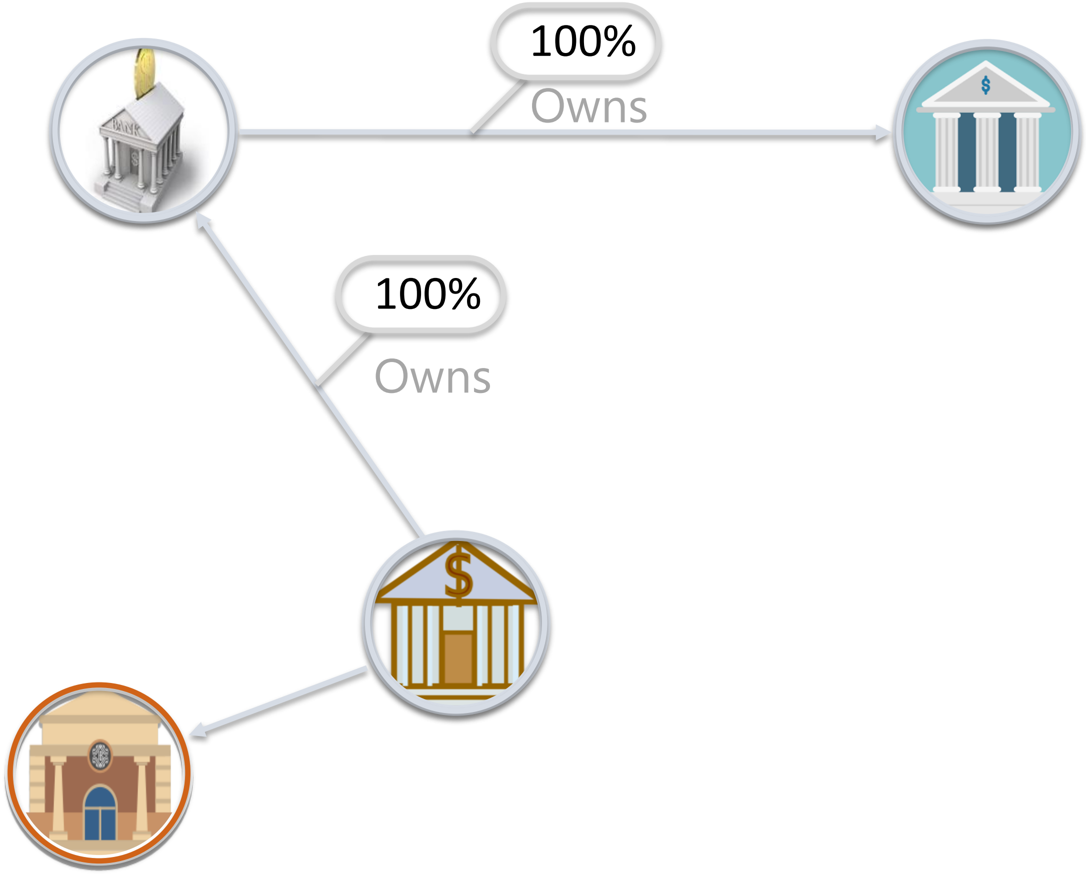
Logical Knowledge in KGs

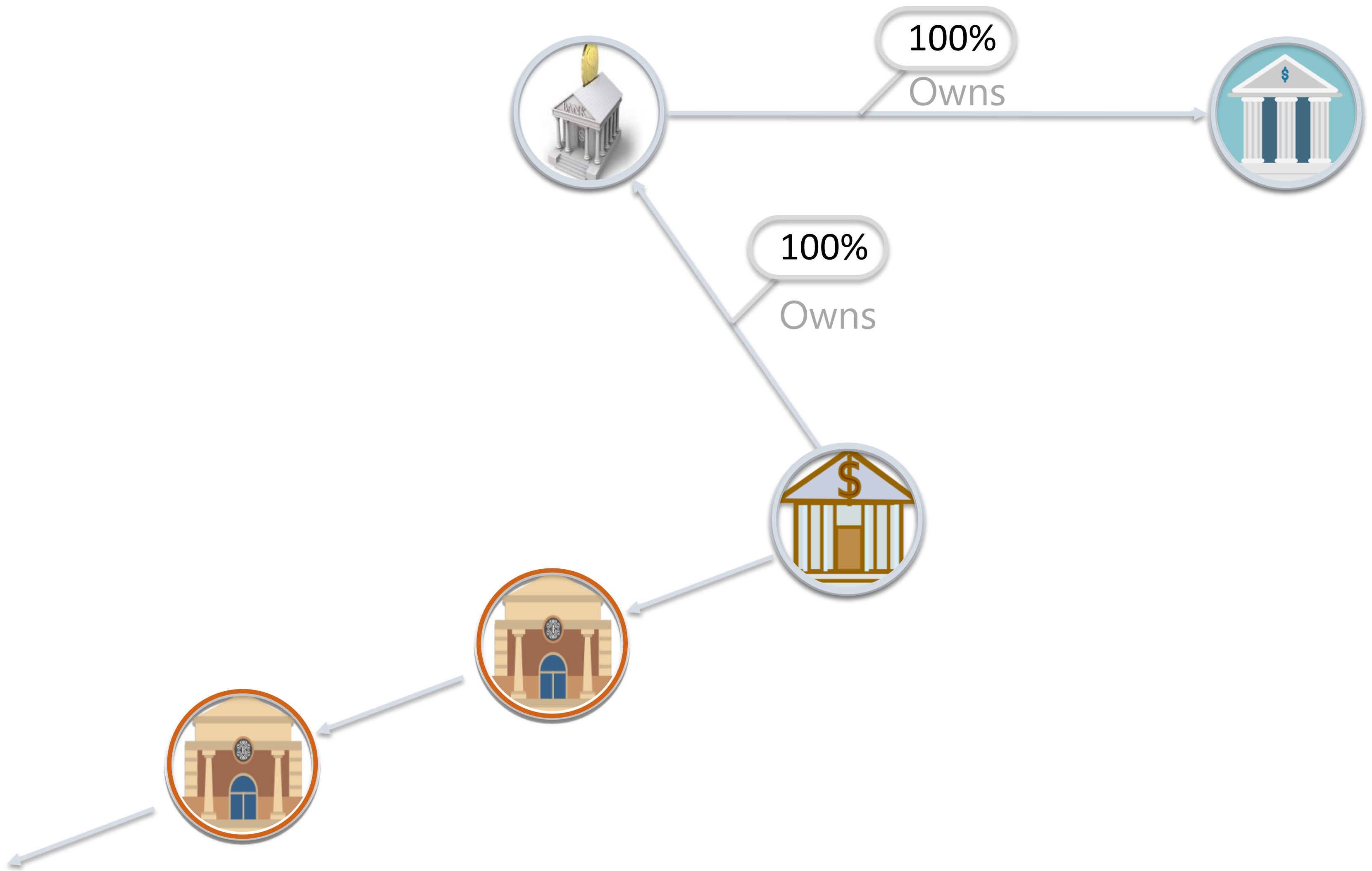
Creation

Emanuel Sallinger


Owns
100%
transnational
Oversight
100%
isClientOf
hasName
Owns
supervises
non-USD
transactionWith
transnational
foundedIn
hasAPositionAt
1970
hasHQ
hasName
Sample Bank
City


100%
Owns
100%
Owns


100%
Owns
100%
Owns


100%
Owns
100%
Owns

$$\mathrm{PLC}(\mathrm{HSBC}, \mathrm{UK}, \mathrm{London}) \rightarrow \mathrm{Company}(\mathrm{HSBC})$$

$$\forall x, z \; \mathrm{PLC}(x, \mathrm{UK}, z) \rightarrow \mathrm{Company}(x)$$

$$\forall x, y, z\ \text{PLC}(x, y, z) \rightarrow \text{Company}(x)$$

$$\forall x, y, z\ \mathrm{PLC}(x, y, z) \rightarrow \exists l\ \mathrm{Company}(x, l)$$

$$\overbrace{\forall x, y, z\ \exists l}^{Q}\ \overbrace{\mathrm{PLC}(x, y, z)}^{\varphi} \rightarrow \overbrace{\mathrm{Company}(x, l)}^{\psi}$$

$$\underbrace{\forall x, y, z\ \exists l}_{Q}\ \underbrace{\mathrm{PLC}(x, y, z)}_{\varphi} \to \underbrace{\mathrm{Company}(x, l)}_{\psi}$$

Knowledge

"**Dependency**"

"**Rule**"

"Logical **Formula**"

"Ontology **Axiom**"

"**Constraint**"

"Named **Query**"

SQL (PostgreSQL)

```
CREATE SEQUENCE locationSequence;

SELECT id, nextval(‘locationSequence’)
INTO Company FROM PLC
```

```
Company(C,L) :- PLC(X,Y,C).
```

Recall

SQL: **NULL**

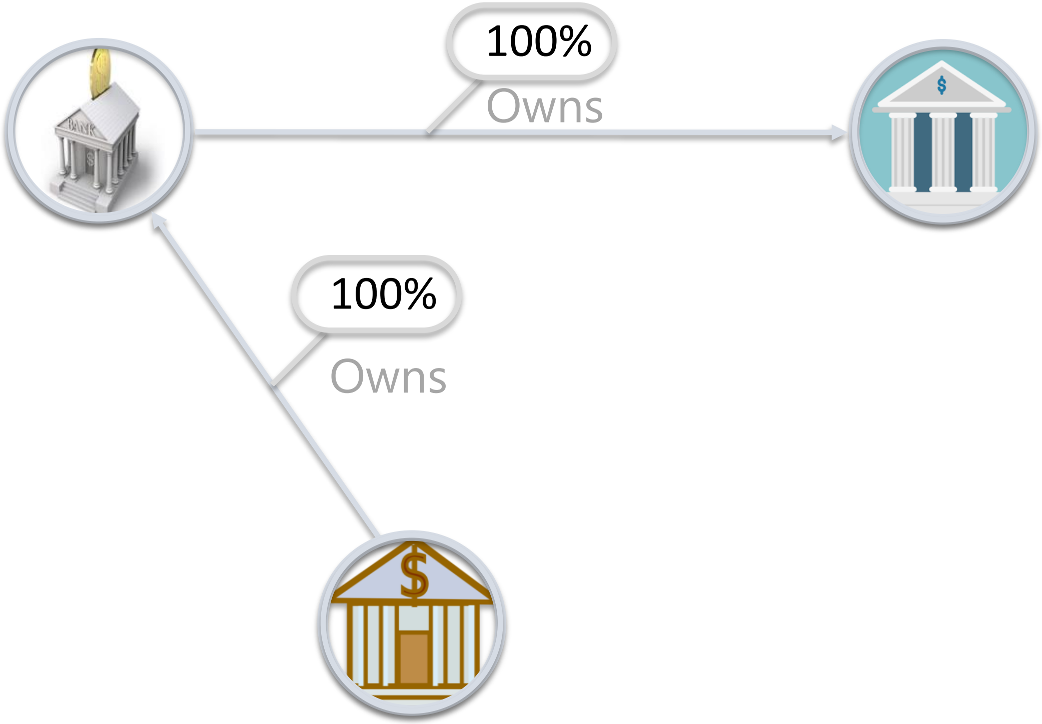
v **IS [NOT] NULL**

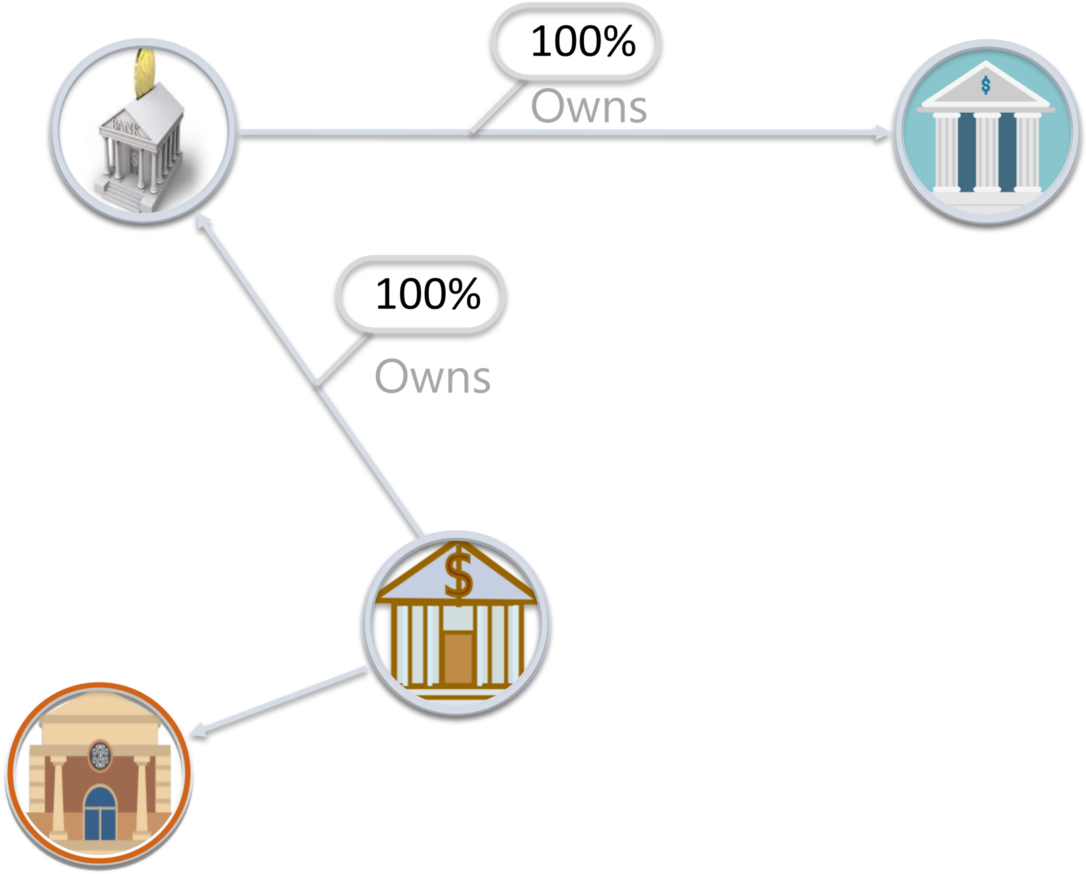
- **NULL** indicates an “unknown” value
  - many applications (not applicable, not existent, …)
  - often misused
- **NULL** is different from any other value, including itself
  - **NULL** = **NULL** is not true!
  - to test for null, use **IS** [**NOT**] **NULL**

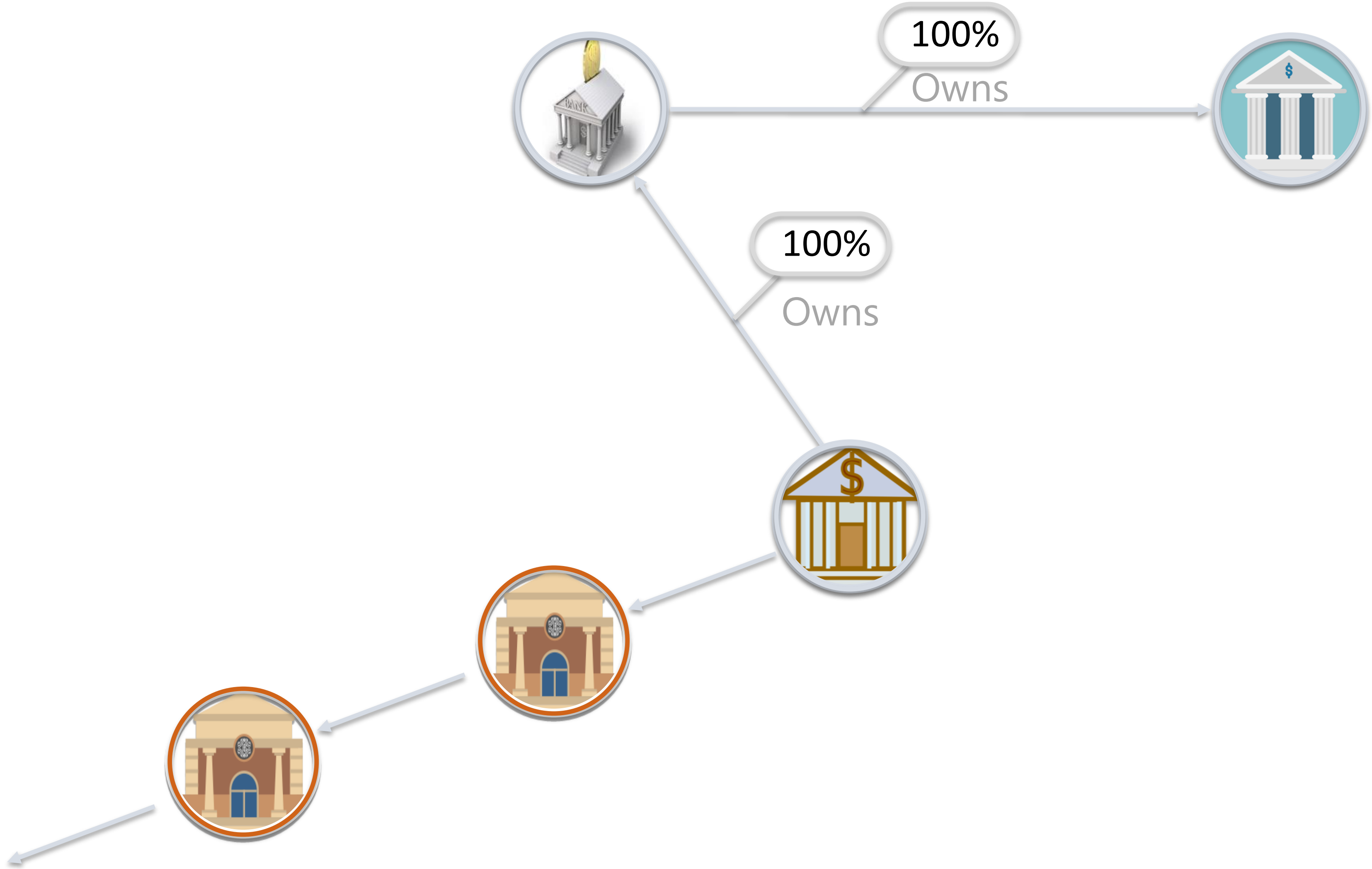
Recall

SQL: **NULL**

φ	ψ	φ AND ψ	φ OR ψ	NOT φ
true	true	true	true	false
true	false	false	true	false
true	null	null	true	false
false	true	false	true	true
false	false	false	false	true
false	null	false	null	true
null	true	null	true	null
null	false	false	null	null
null	null	null	null	null


100%
Owns
100%
Owns


100%
Owns
100%
Owns


100%
Owns
100%
Owns

Logical Knowledge in KGs

Warded and Vadalog

Emanuel Sallinger

$$\overbrace{\forall x, y, z\ \exists l}^{Q}\ \overbrace{\mathrm{PLC}(x, y, z)}^{\varphi} \to \overbrace{\mathrm{Company}(x, l)}^{\psi}$$

Knowledge

"**Dependency**"

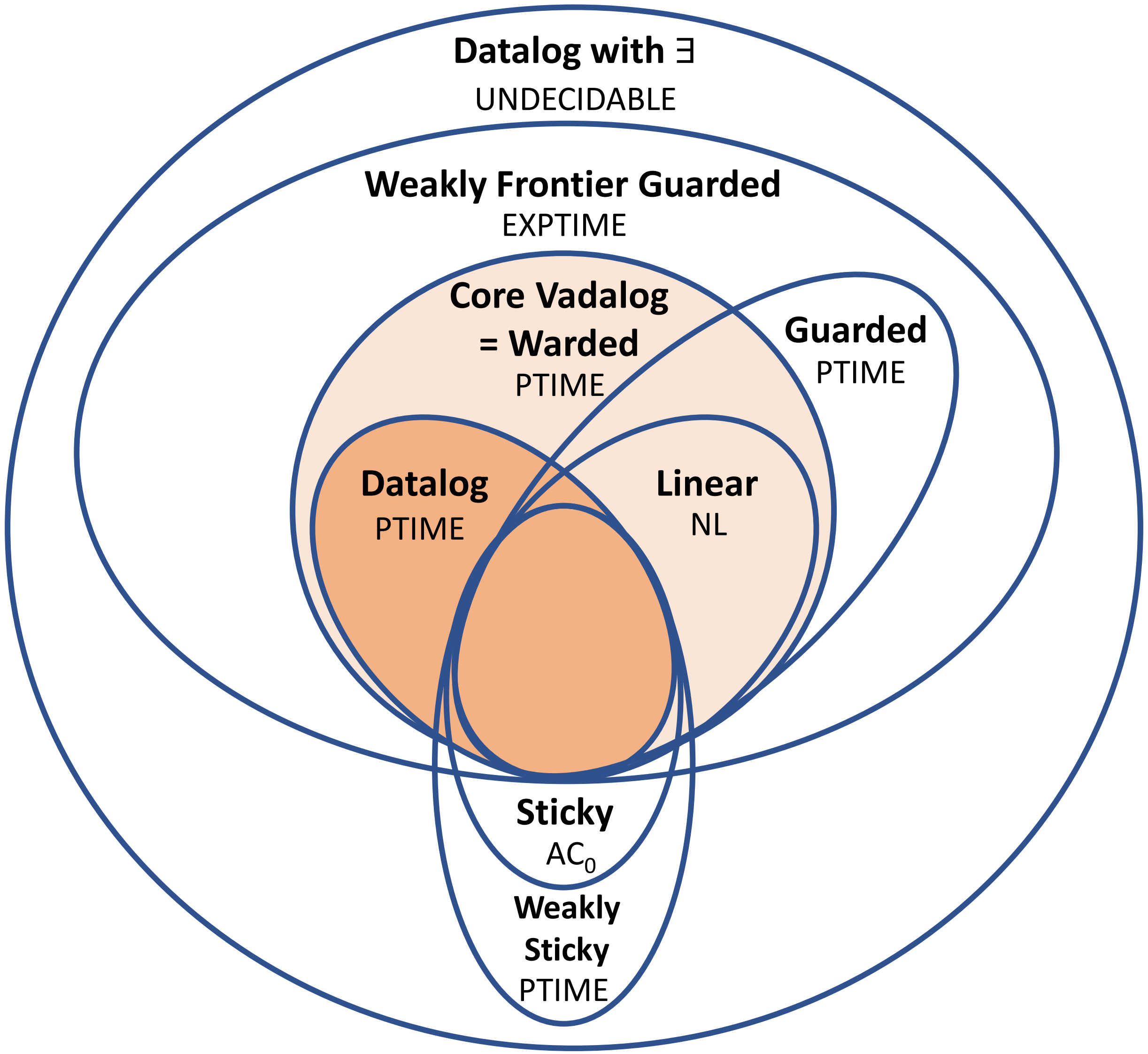
"**Rule**"

"Logical **Formula**"

"Ontology **Axiom**"

"**Constraint**"

"Named **Query**"



Not shown:
Shy

Language Requirements

1. **Recursive** Reasoning:
   Full support of recursive **Datalog**

2. **Ontological** Reasoning:
   Expressive power of **SPARQL** and **OWL 2QL**

3. **Tractable** Reasoning: **polynomial time**,
   sub-fragments that are **fully parallelizable**

$$\mathrm{Emp}(x) \rightarrow \exists z\, \mathrm{Mgr}(x, z) \qquad \mathrm{Mgr}(x, y) \wedge \mathrm{Pers}(x) \rightarrow \mathrm{Emp}(y)$$

$$\mathrm{Emp}(x) \rightarrow \exists z\, \mathrm{Mgr}(x, z) \qquad \mathrm{Mgr}(x, y) \wedge \mathrm{Pers}(x) \rightarrow \mathrm{Emp}(y)$$

$$\text{Emp}(x) \rightarrow \exists \boldsymbol{z}\, \text{Mgr}(x, \boldsymbol{z}) \qquad \text{Mgr}(x, \boldsymbol{y}) \wedge \text{Pers}(x) \rightarrow \text{Emp}(\boldsymbol{y})$$

$$\mathrm{Emp}(x) \rightarrow \exists \boldsymbol{z}\, \mathrm{Mgr}(x, \boldsymbol{z}) \qquad \mathrm{Mgr}(x, \overbrace{\boldsymbol{y}) \wedge \mathrm{Pers}(x) \rightarrow \mathrm{Emp}(\boldsymbol{y}}^{\textbf{Dangerous}})$$

Dangerous

$\mathrm{Emp}(x) \rightarrow \exists z\, \mathrm{Mgr}(x, z)$ $\quad$ $\mathrm{Mgr}(x, y) \wedge \mathrm{Pers}(x) \rightarrow \mathrm{Emp}(y)$

Ward

1. all the "**dangerous**" variables should coexist in a single body-atom α, called the **ward**

Harmless

$$\mathrm{Emp}(x) \rightarrow \exists z\, \mathrm{Mgr}(x, z) \qquad \underbrace{\mathrm{Mgr}(x, y)}_{\textbf{Ward}} \land \mathrm{Pers}(x) \rightarrow \mathrm{Emp}(y)$$

1. all the "**dangerous**" variables should coexist in a single body-atom α, called the **ward**
2. the ward can share only "**harmless**" variables with the rest of the body

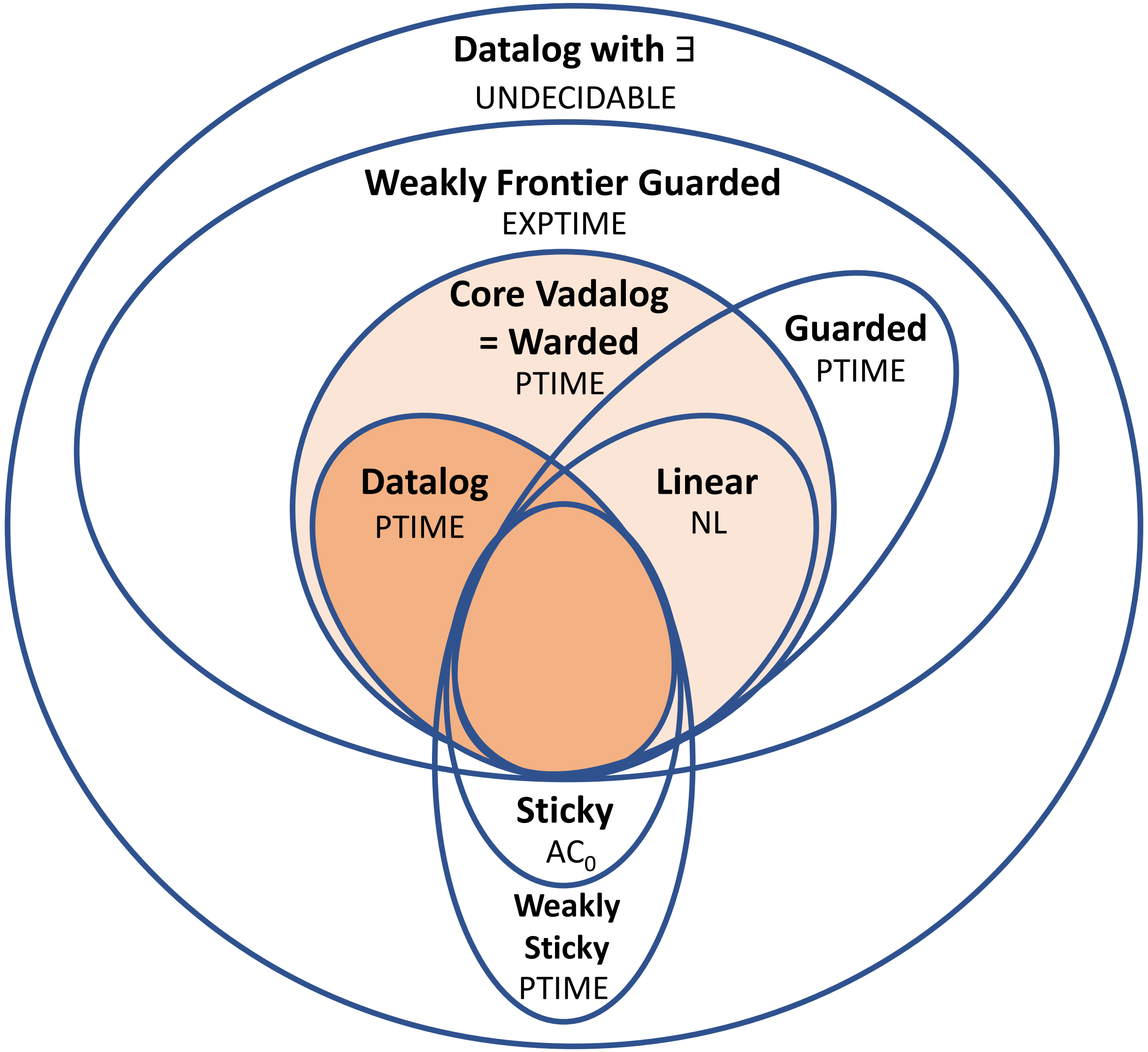
$$\text{Emp}(x) \rightarrow \exists \boldsymbol{z}\ \text{Mgr}(x, \boldsymbol{z}) \qquad \text{Mgr}(x, \overset{\text{Harmless}}{\boldsymbol{y}})\ \wedge\ \overset{\text{Harmless}}{\text{Pers}(x)} \rightarrow \text{Emp}(\boldsymbol{y})$$

Harmless

Ward

1. all the "**dangerous**" variables should coexist in a single body-atom α, called the **ward**
2. the ward can share only "**harmless**" variables with the rest of the body

Warded Dependencies



Not shown:
Shy

Main Challenges

1. **Recursion**
   unlimited graph exploration

2. **Object Creation**
   exploring unknown parts of the KG

Logical Knowledge in KGs

Warded and Vadalog

Emanuel Sallinger

Logical Knowledge in KGs

Creation

Emanuel Sallinger

Logical Knowledge in KGs

Vadalog – Interacting with the World

Emanuel Sallinger

```
connection(X,Y) :- flight(X,Y).
connection(X,Z) :- connection(Y,Z), flight(X,Y).
```

```
flight("VIE","LHR","BA").
flight("LHR","SFO","BA").
flight("LGW","OAK","DI").
flight("BUD","VIE","OS").

connection(X,Y) :- flight(X,Y).
connection(X,Z) :- connection(Y,Z), flight(X,Y).
```

```
flight("VIE","LHR","BA").
flight("LHR","SFO","BA").
flight("LGW","OAK","DI").
flight("BUD","VIE","OS").

connection(X,Y) :- flight(X,Y).
connection(X,Z) :- connection(Y,Z), flight(X,Y).

@output("connection").
```

Binding

```
@input("flight ").
@bind("flight","postgres","mw_kg4","eflight").
```

Binding

```
@input("flight ").
@bind("flight","postgres","mw_kg4","eflight").
```

```
@bind("atomName","data source",
   "outer container","inner container").
```

Query Binding

```
@qbind("flight","postgres","mw_kg_db6",
  "select O,D,F from TestTable
   where k_id between 394823 and 4458773").
```

Query Binding

```
@qbind("flight","postgres","mw_kg_db6",
  "select O,D,F from TestTable
   where k_id between 394823 and 4458773").
```

```
@qbind("atomName","data source",
   "outer container","query").
```

Query Binding

```
@qbind("flight", "mw_graph_db7",
"MATCH (a)-[o:Carrier]->(b)
 RETURN a,b,o.Carrier").
```

In- and Output Binding

```
flightOnly(X,Y) :- flight(X,Y,Z).

@input("flight ").
@bind("flight","csv","/d2","s1.csv").

@output("flightOnly").
@bind("flightOnly","csv","/d3","result.csv").
```

Logical Knowledge in KGs

Vadalog – Interacting with the World

Emanuel Sallinger

Logical Knowledge in KGs

Vadalog –Aggregation and Operators

Emanuel Sallinger

Operators (Example)

```
a("meltwater").
b("san francisco").

q(Y,J) :- a(X), b(Y), J = substring(X,5,9).

@output("q").
```

```
q("water","san francisco").
```

Operators (Example)

```
balanceItem("loans",23.0).
balanceItem("deposits",20.0).

operations(Q,Z,A) :-
   balanceItem(I1,X), balanceItem(I2,Y),
   I1="loans", I2="deposits", Z=X+Y, A=(X+Y)/2.

@output("operations").
```

```
operations(z1,43,21.5).
```

Aggregation

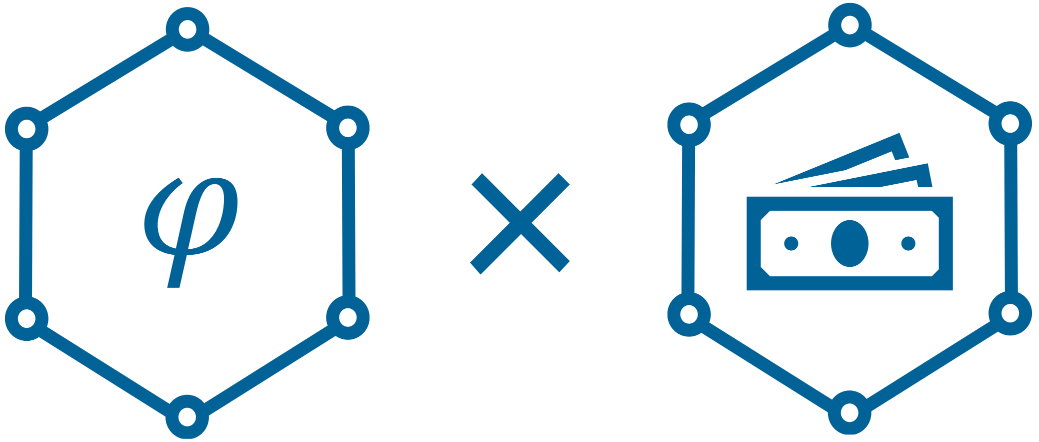
```
s(1.0,"a"). s(2.0,"a"). s(3.0,"a").
s(4.0,"b"). s(3.0,"b").

f(J,Y) :- s(X,Y), J = sum(X).

@output("f").
```

```
f(6.0,"a").
f(7.0,"b").
```

Logical Knowledge in KGs

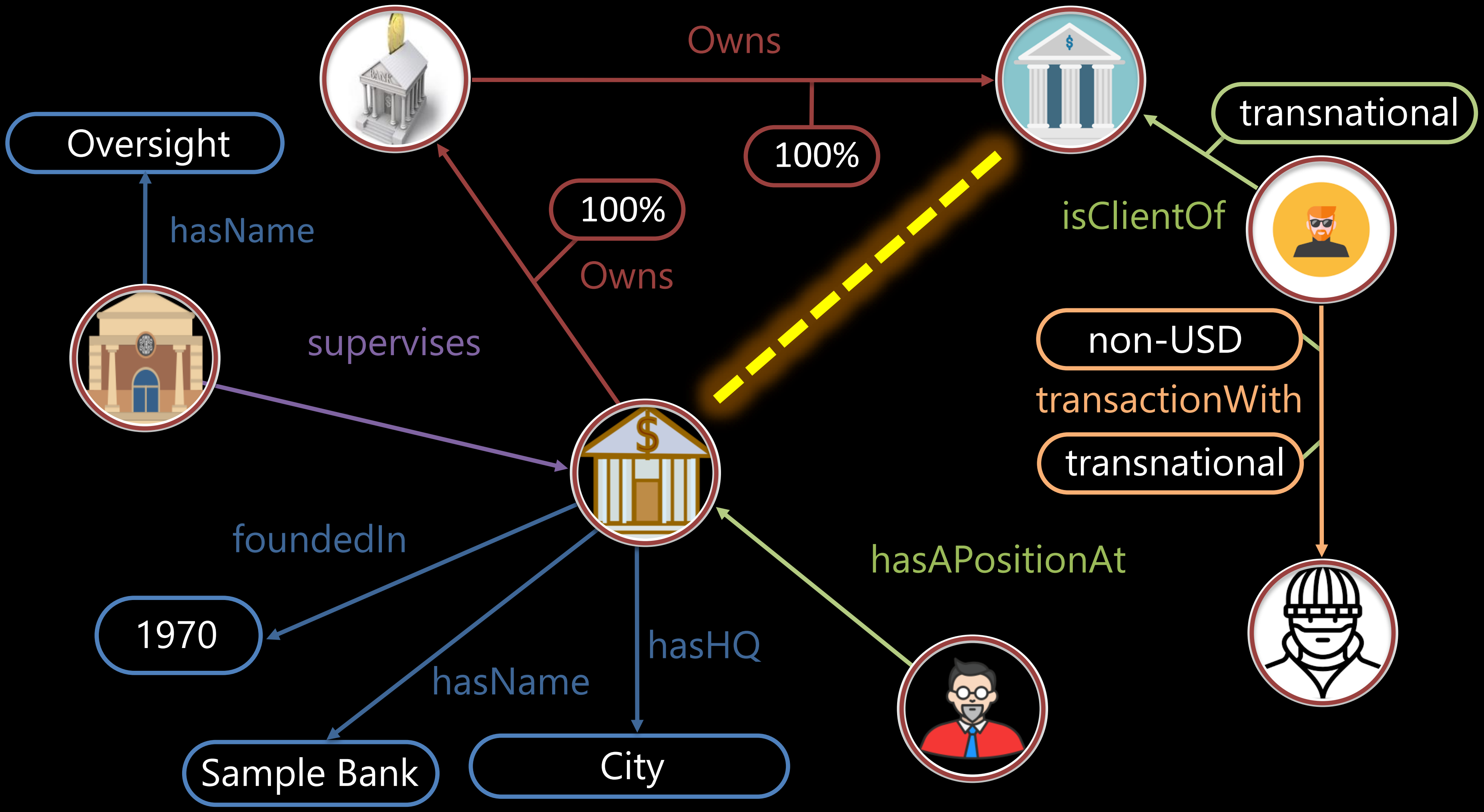
Vadalog –Aggregation and Operators

Emanuel Sallinger

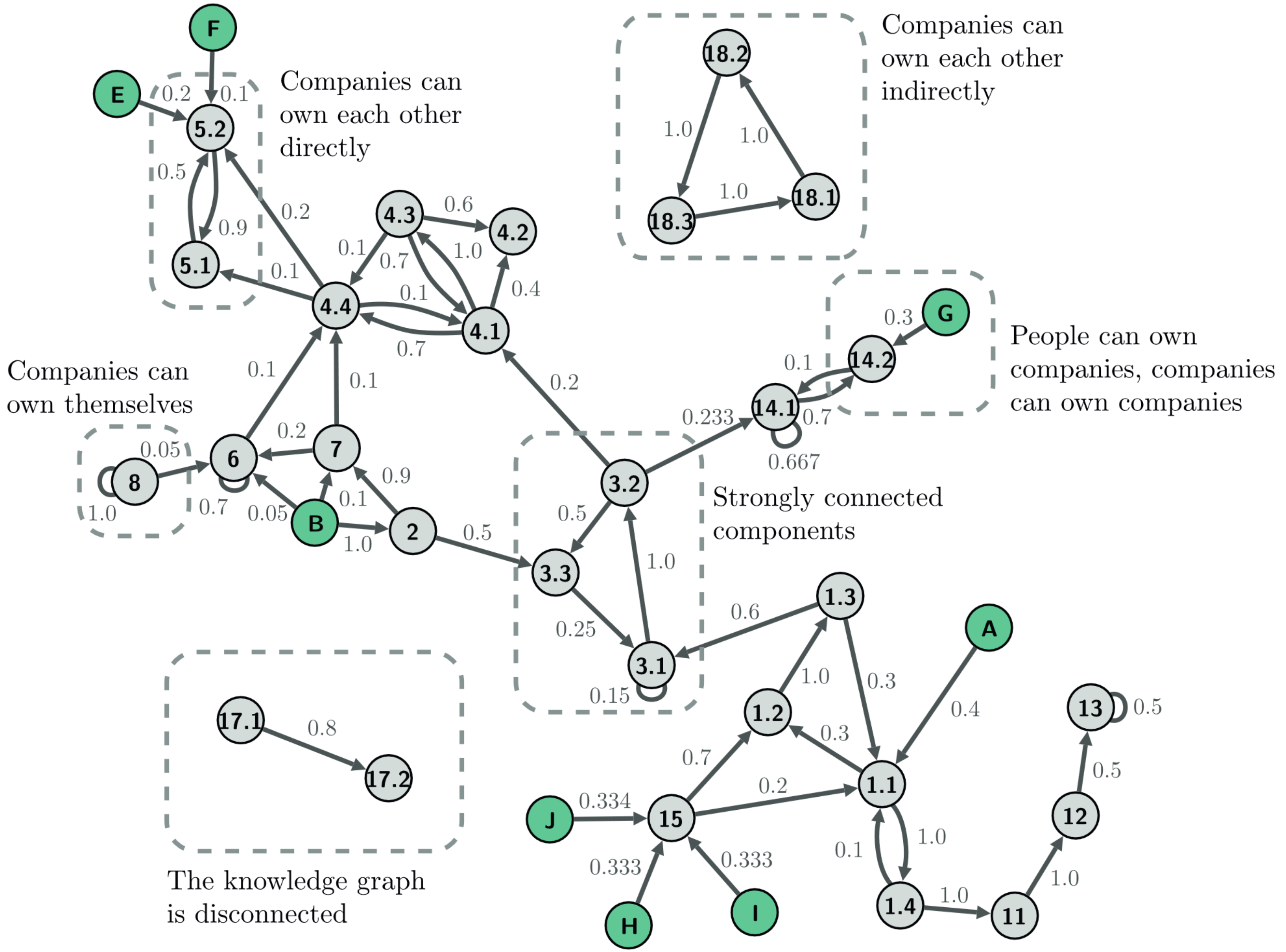


Logical Knowledge in KGs
The Real World

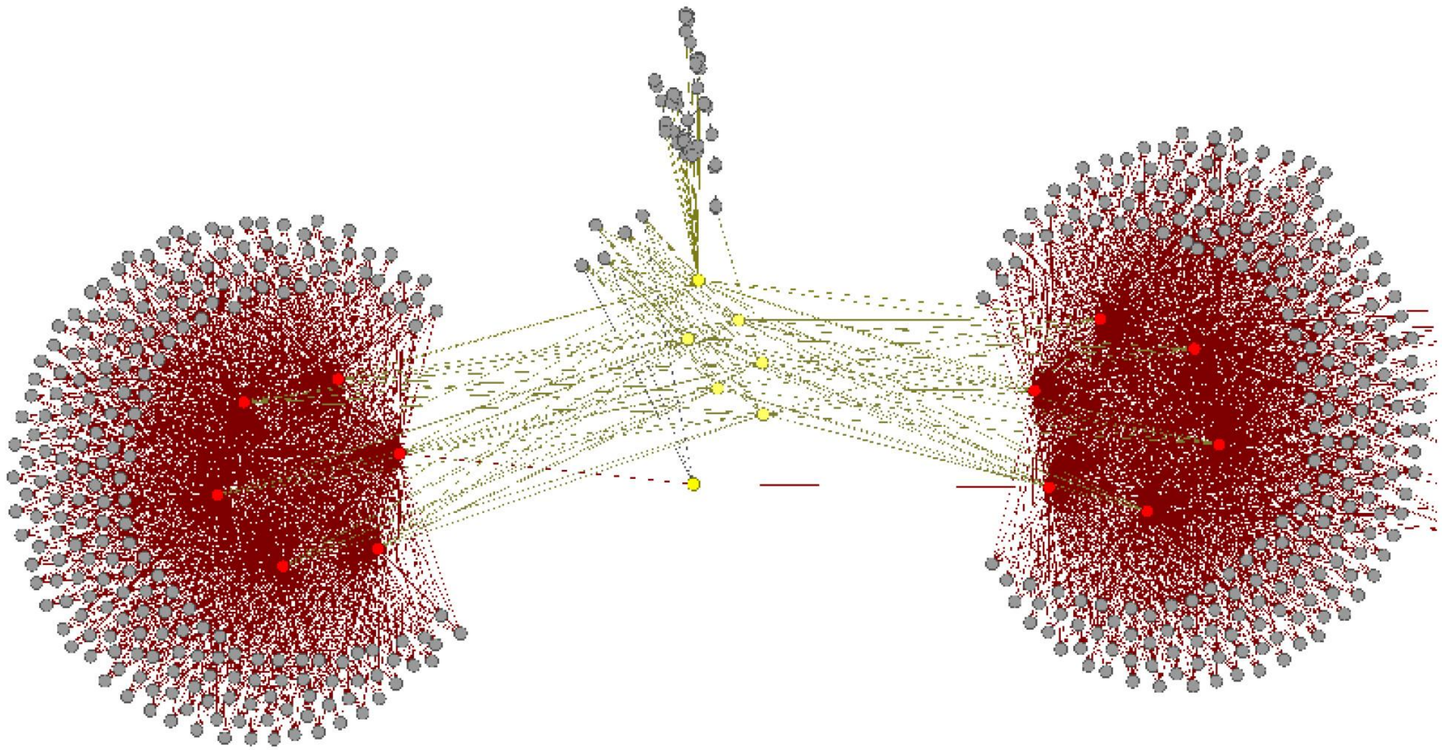
Emanuel Sallinger

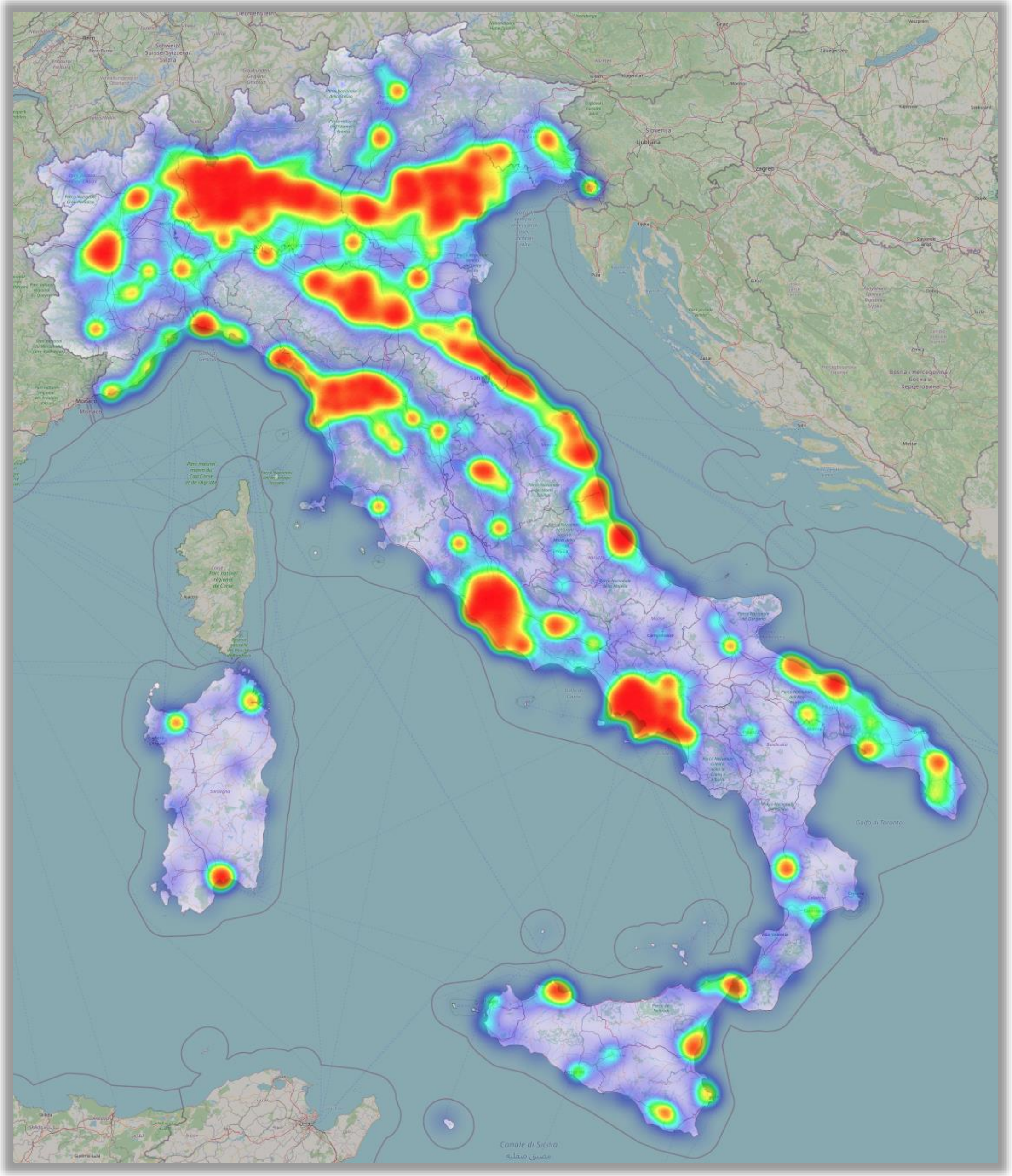


Owns
100%
transnational
Oversight
hasName
100%
Owns
isClientOf
supervises
non-USD
transactionWith
transnational
foundedIn
hasAPositionAt
1970
hasHQ
hasName
Sample Bank
City



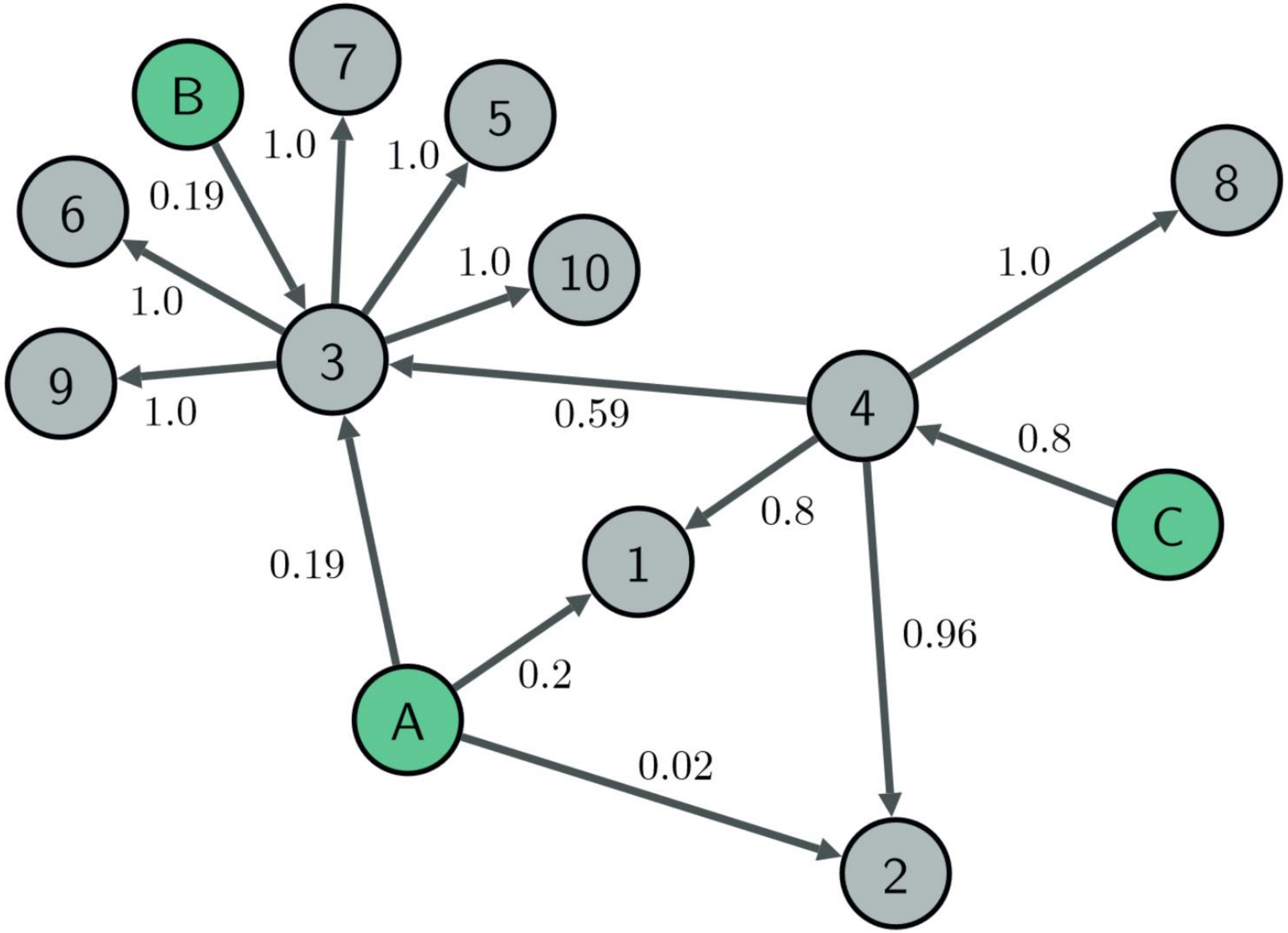
Source: https://arxiv.org/abs/2004.10119





Conglomerates

Conglomerates



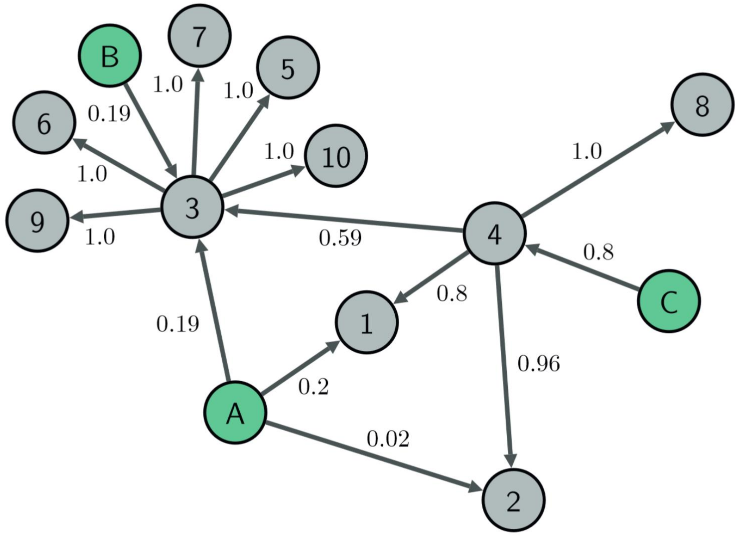
Conglomerates

Definition 5. *The* undirected ϵ-Baldone ownership *of a company s on a company t in a graph G is a function* $\mathcal{U}_\epsilon^G(s,t) : max\{O_\epsilon^G(s,t), O_\epsilon^G(t,s)\}$.

Definition 6. *The binary relation* vicinity $\mathcal{V}_\epsilon^G$ *holds between two companies x and y of a graph G (and we say — they are close —), if: (i)* $\mathcal{U}_\epsilon^G(x,y) > 0$*; or, (ii) there exists a third party z of G such that* $\mathcal{U}_\epsilon^G(z,x) > 0$ *and* $\mathcal{U}_\epsilon^G(z,y) > 0$.

Definition 7. *An ϵ-conglomerate is an equivalence class of $\mathcal{V}_\epsilon^{+G}$, where + denotes the transitive closure of the vicinity relation defined on a graph G.*

Conglomerates



$$O(a,b,x), x > \epsilon \rightarrow U(a,b,x), U(b,a,x) \tag{1}$$

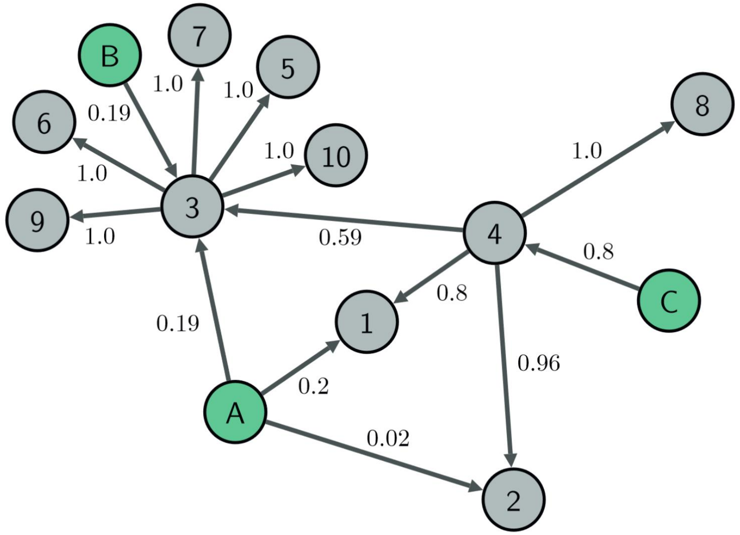
$$U(a,b,x), a > b \rightarrow \exists z \; C(z,a), C(z,b) \tag{2}$$

$$U(w,a,x), U(w,b,x) \rightarrow \exists z \; C(z,a), C(z,b) \tag{3}$$

$$Company(a) \rightarrow \exists z \; C(z,a) \tag{4}$$

$$C(y,a), C(x,a) \rightarrow x = y \tag{5}$$

Conglomerates



$$O(a,b,x), x > \epsilon \rightarrow U(a,b,x), U(b,a,x) \quad (1)$$

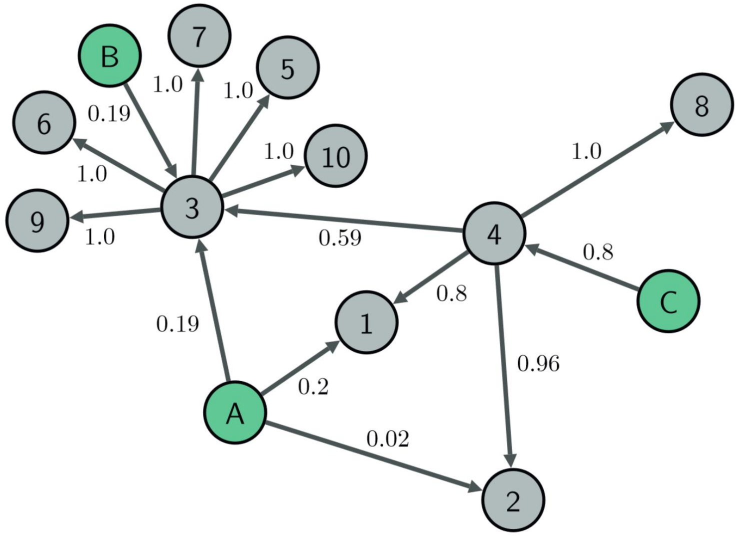
$$U(a,b,x), a > b \rightarrow \exists z\ C(z,a), C(z,b) \quad (2)$$

$$U(w,a,x), U(w,b,x) \rightarrow \exists z\ C(z,a), C(z,b) \quad (3)$$

$$Company(a) \rightarrow \exists z\ C(z,a) \quad (4)$$

$$C(y,a), C(x,a) \rightarrow x = y \quad (5)$$

Conglomerates



$$O(a,b,x), x > \epsilon \longrightarrow U(a,b,x), U(b,a,x) \quad (1)$$

$$U(a,b,x), a > b \longrightarrow \exists z \; C(z,a), C(z,b) \quad (2)$$

$$U(w,a,x), U(w,b,x) \longrightarrow \exists z \; C(z,a), C(z,b) \quad (3)$$

$$Company(a) \longrightarrow \exists z \; C(z,a) \quad (4)$$

$$C(y,a), C(x,a) \longrightarrow x = y \quad (5)$$

Collusion

Considering Collusion

Collusion Golden Power Check

Goal: The goal is checking whether an acquisition (of shares, stocks, etc.) causes any strategic Italian company to be possibly controlled by a set of foreign companies acting in collusion.

Setting: Let S be a set of strategic companies and F be a set of foreign companies. Let t be a transaction (e.g., an offer issued by a company x to buy an amount s of shares of a company y), with $x, y \in S \cup F$.

Question: Decide whether t causes F to jointly control a company in S.

Insight: Consider the possibility to exert Golden Powers to block t.

Considering Collusion

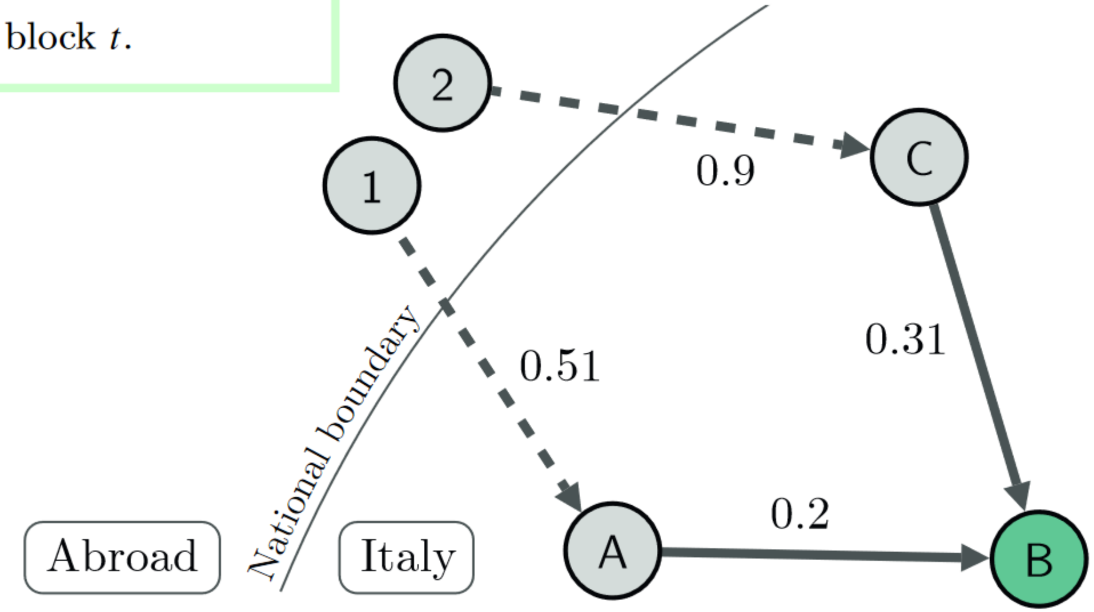
Collusion Golden Power Check

Goal: The goal is checking whether an acquisition (of shares, stocks, etc.) causes any strategic Italian company to be possibly controlled by a set of foreign companies acting in collusion.

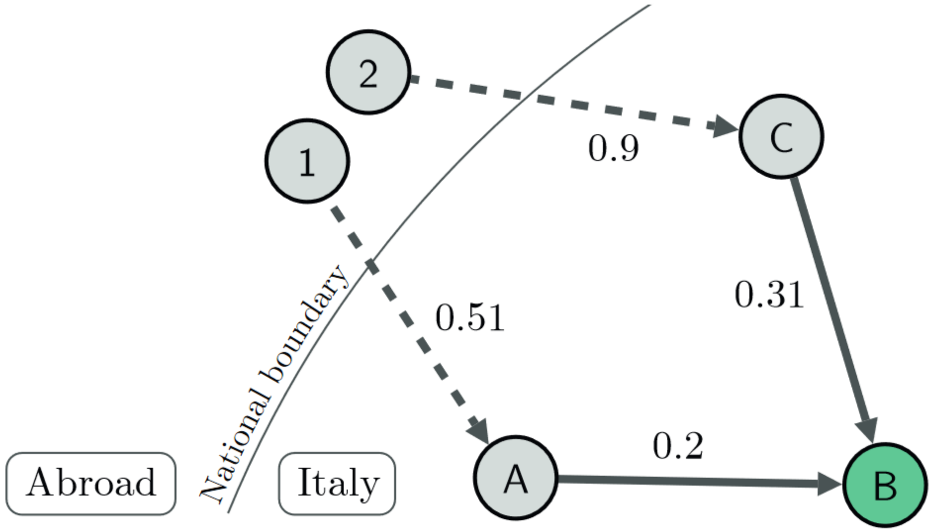
Setting: Let S be a set of strategic companies and F be a set of foreign companies. Let t be a transaction (e.g., an offer issued by a company x to buy an amount s of shares of a company y), with $x, y \in S \cup F$.

Question: Decide whether t causes F to jointly control a company in S.

Insight: Consider the possibility to exert Golden Powers to block t.



Considering Collusion



$$F(x), F(y) \rightarrow Control(x, y) \quad (1)$$

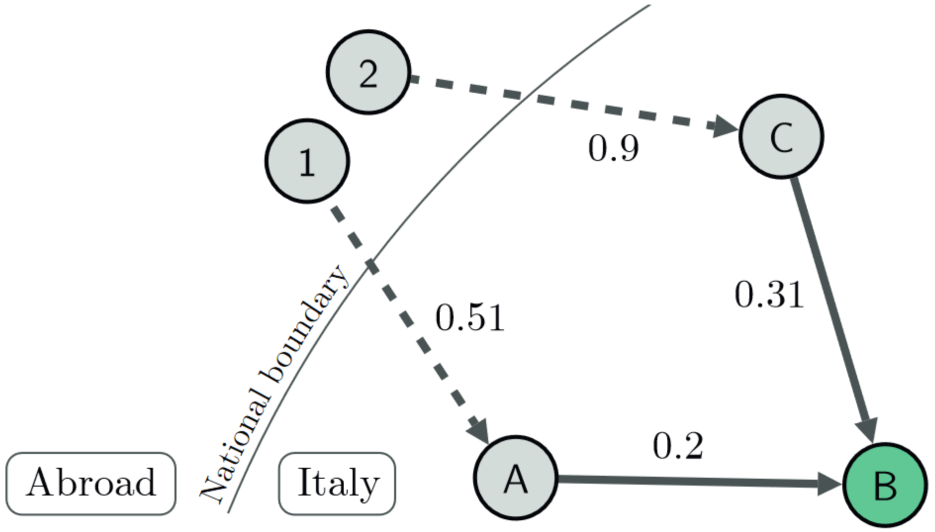
$$T(x, y, w) \rightarrow Own(x, y, w) \quad (2)$$

$$Control(x, y) \rightarrow Control(x, x) \quad (3)$$

$$Control(x, y), Own(y, z, w), v = msum(w, \langle y \rangle), v > 0.5 \rightarrow Control(x, z) \quad (4)$$

$$F(x), S(y), Control(x, y) \rightarrow CGPCheck(x, y) \quad (5)$$

Considering Collusion



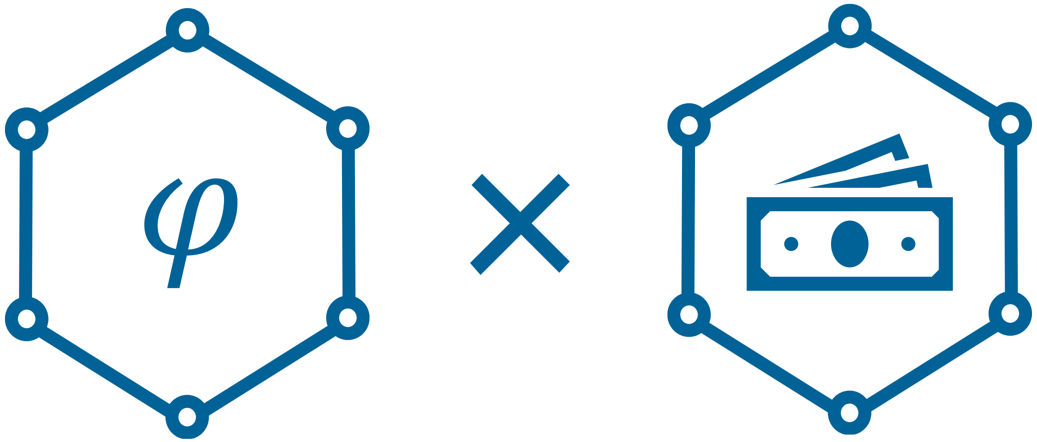
$$F(x), F(y) \rightarrow Control(x, y) \quad (1)$$

$$T(x, y, w) \rightarrow Own(x, y, w) \quad (2)$$

$$Control(x, y) \rightarrow Control(x, x) \quad (3)$$

$$Control(x, y), Own(y, z, w), v = msum(w, \langle y \rangle), v > 0.5 \rightarrow Control(x, z) \quad (4)$$

$$F(x), S(y), Control(x, y) \rightarrow CGPCheck(x, y) \quad (5)$$

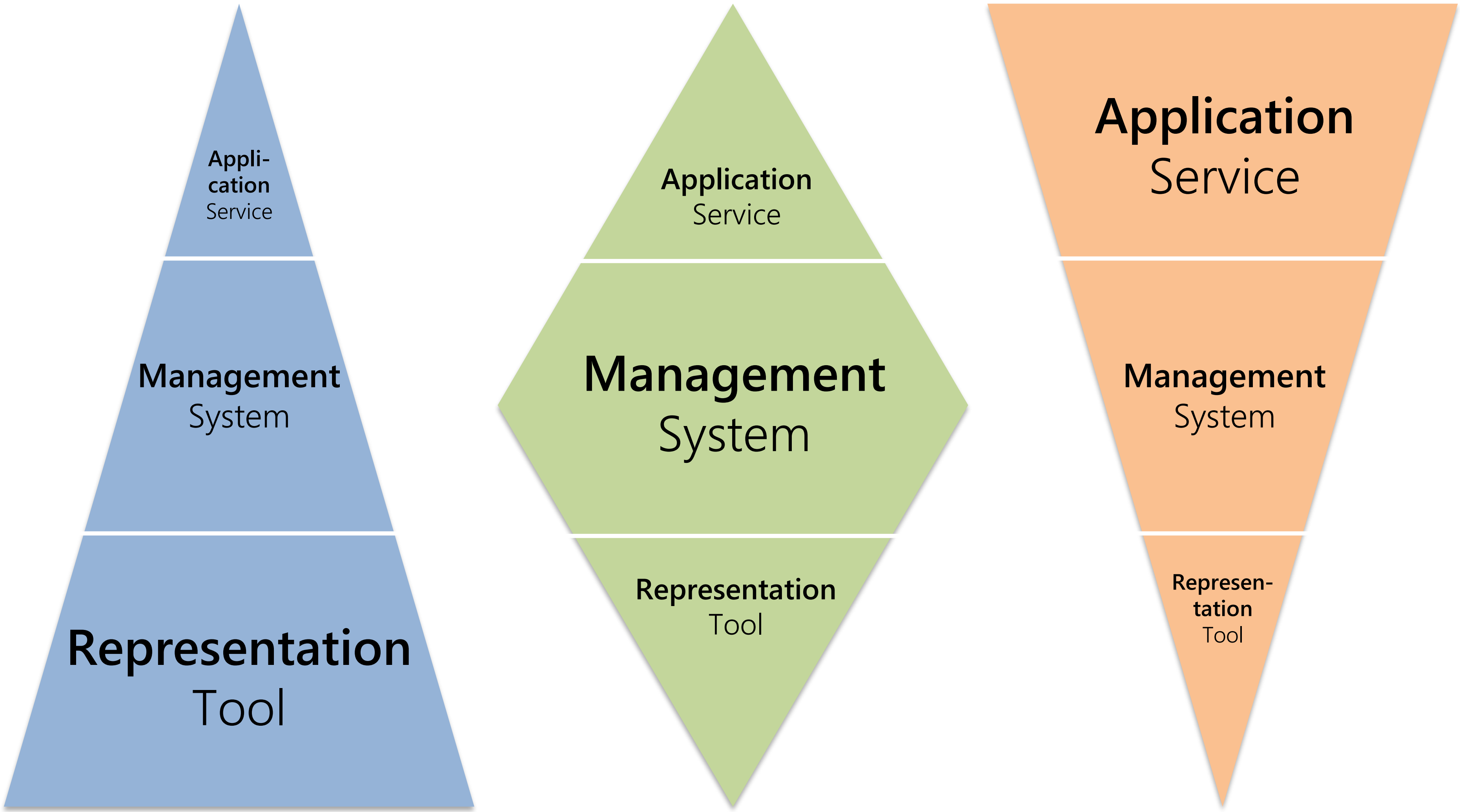


Logical Knowledge in KGs
The Real World

Emanuel Sallinger

Knowledge Graph Management Systems

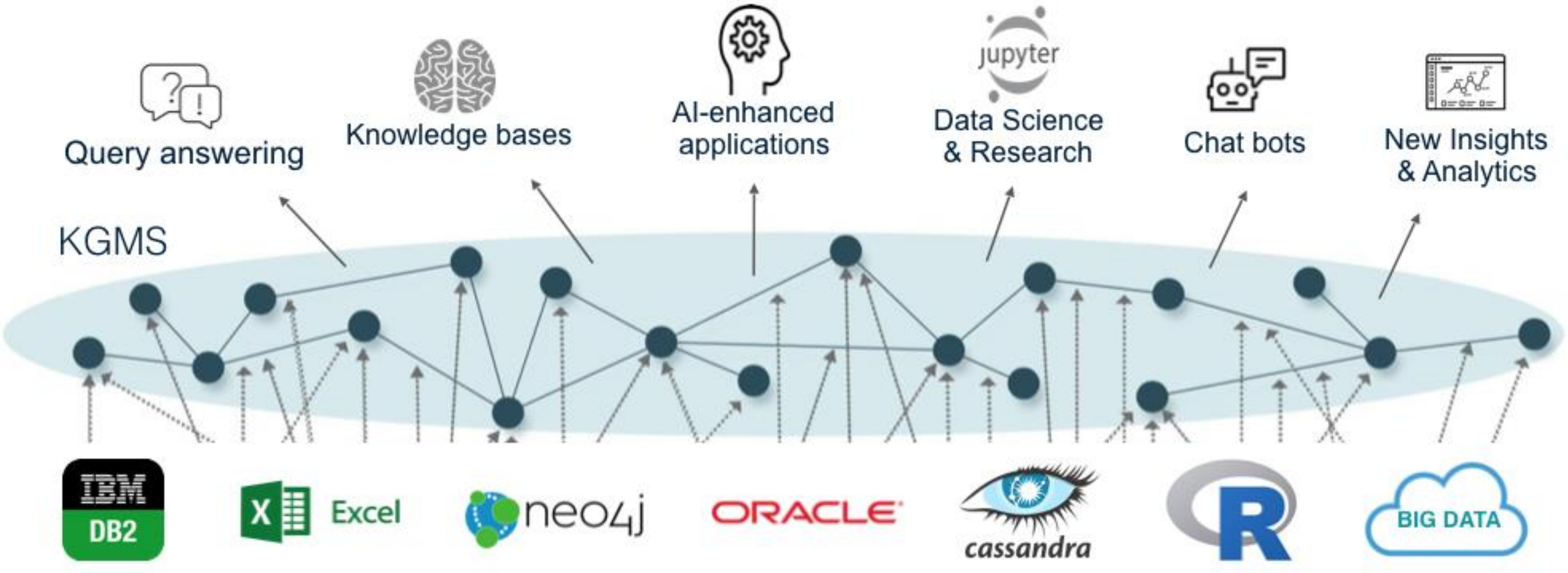
Emanuel Sallinger


Appli-
cation
Service
Management
System
Representation
Tool
Application
Service
Management
System
Representation
Tool
Application
Service
Management
System
Represen-
tation
Tool

"Enterprise Knowledge Graphs must be deployed into a **Knowledge Graph Management System**, a middleware that:

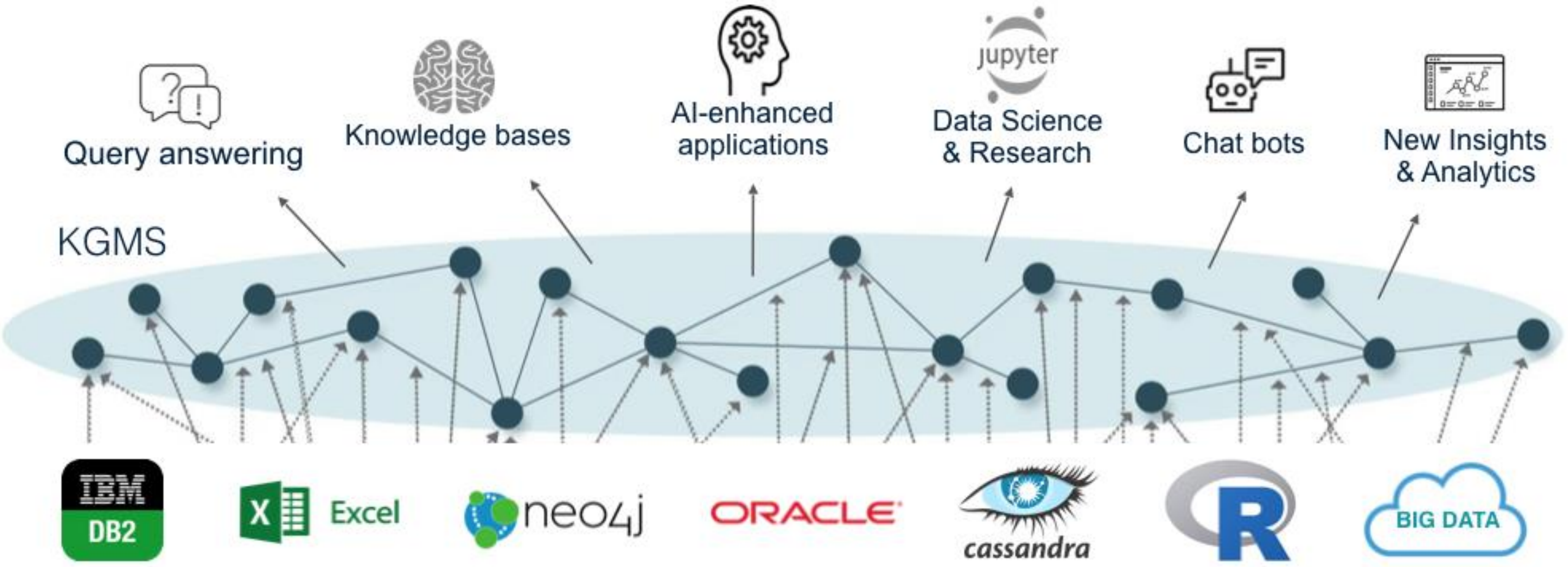
1. provides a language and a formalism for representation of and reasoning on KGs
2. can access a rich set of data sources (including big data);
3. can embed procedural and third-party code
4. provides reasoning services via a rich set of APIs."

Bellomarini, Fakhoury, Gottlob, Sallinger.
Knowledge Graphs and Enterprise AI: The Promise of an Enabling Technology.
ICDE 2019.



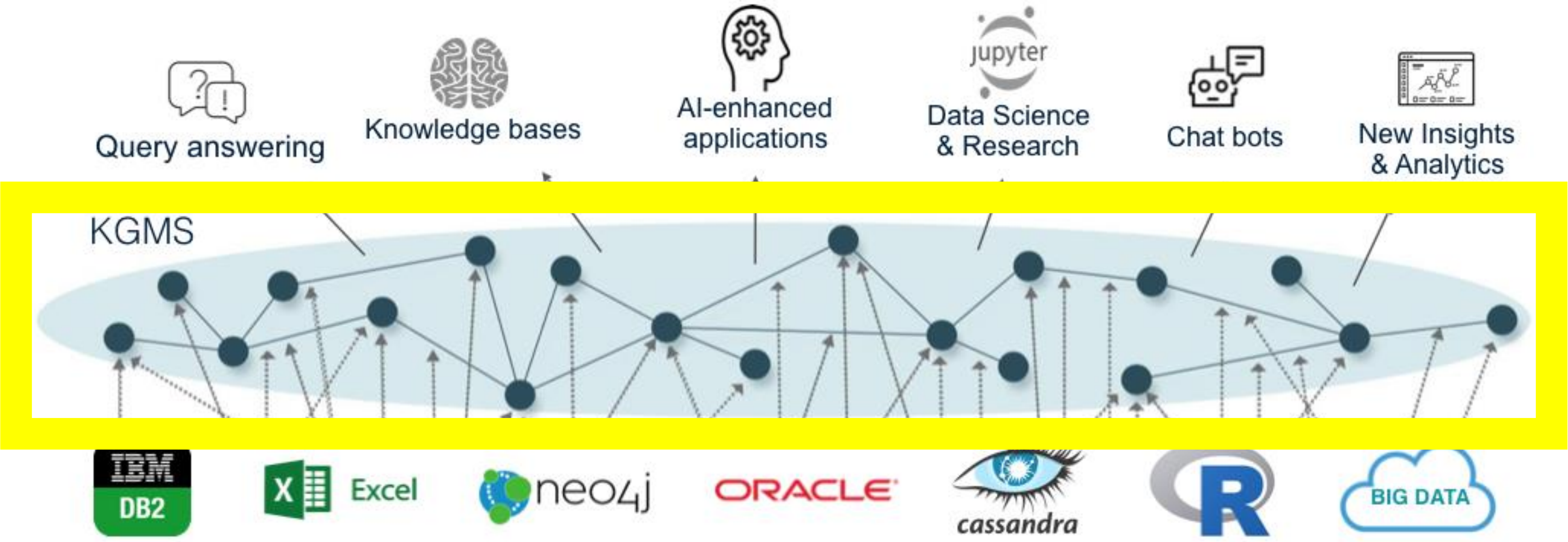
Source: Bellomarini, Fakhoury, Gottlob, Sallinger. Knowledge Graphs and Enterprise AI: The Promise of an Enabling Technology. ICDE 2019.

1. provides a language and a formalism
   for representation of and reasoning on KGs
2. can access a rich set of data sources (including big data);
3. can embed procedural and third-party code
4. provides reasoning services via a rich set of APIs.



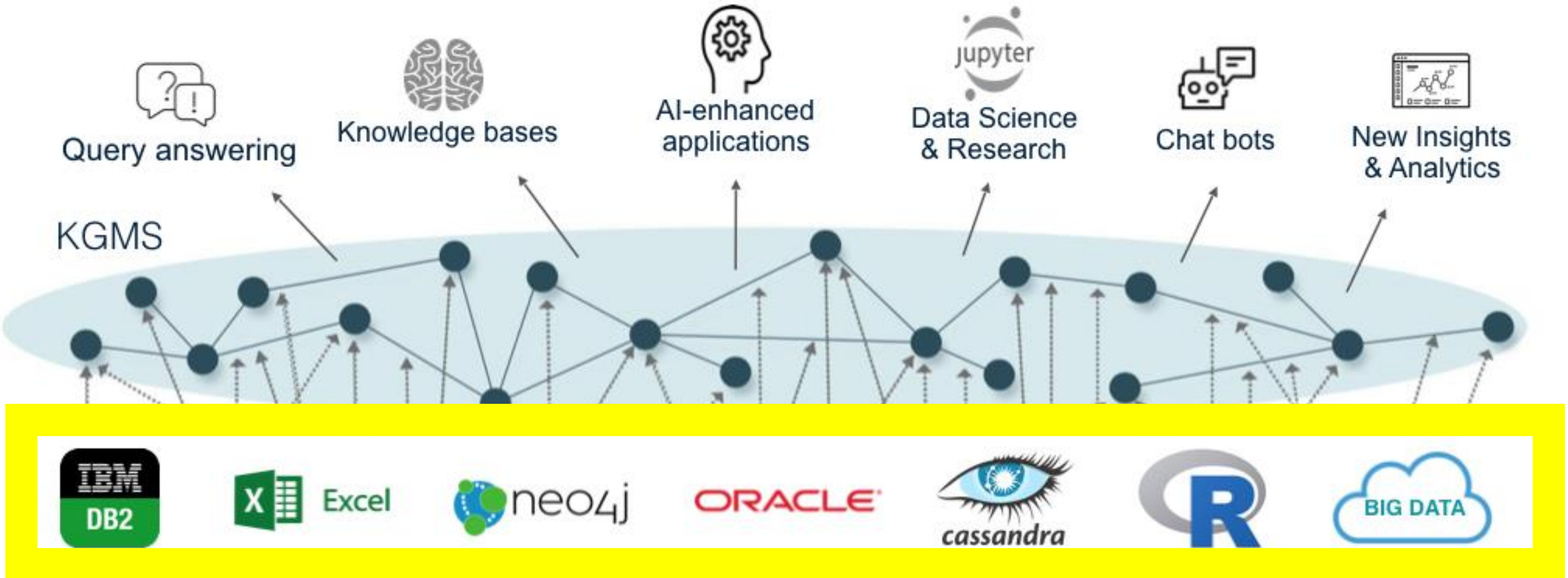
Source: Bellomarini, Fakhoury, Gottlob, Sallinger. Knowledge Graphs and Enterprise AI: The Promise of an Enabling Technology. ICDE 2019.

1. **provides a language and a formalism for representation of and reasoning on KGs**
2. can access a rich set of data sources (including big data);
3. can embed procedural and third-party code
4. provides reasoning services via a rich set of APIs.



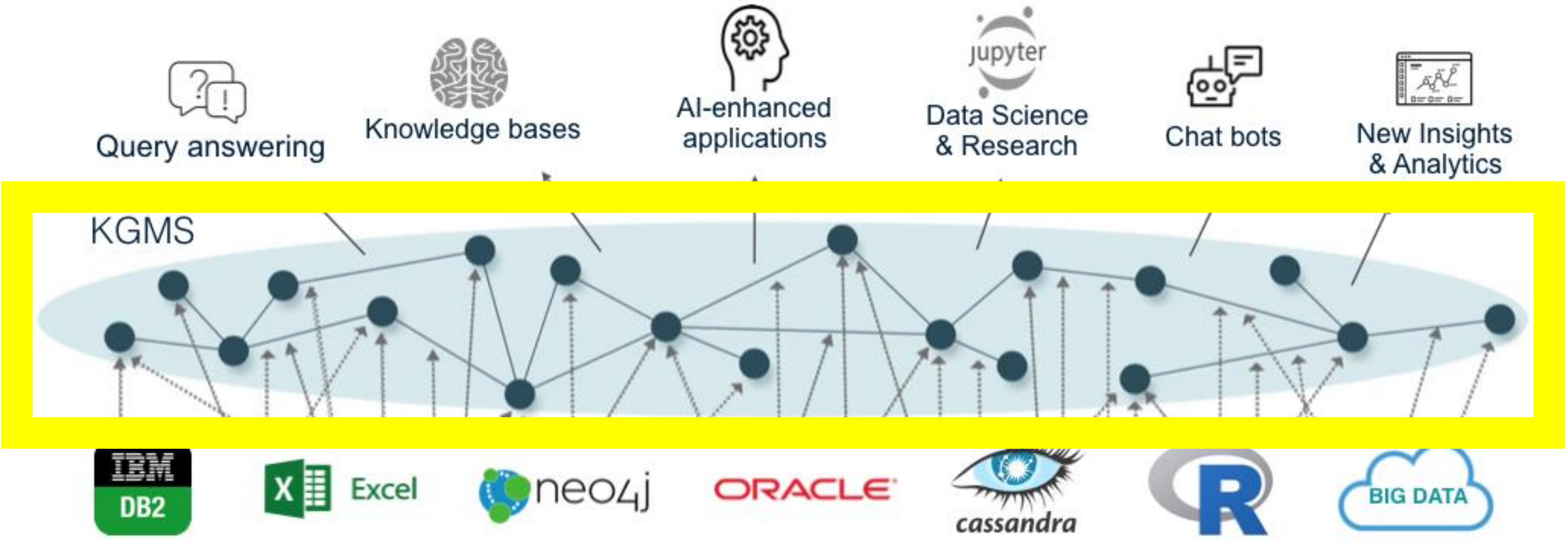
Source: Bellomarini, Fakhoury, Gottlob, Sallinger. Knowledge Graphs and Enterprise AI: The Promise of an Enabling Technology. ICDE 2019.

1. provides a language and a formalism
   for representation of and reasoning on KGs
2. **can access a rich set of data sources (including big data);**
3. can embed procedural and third-party code
4. provides reasoning services via a rich set of APIs.



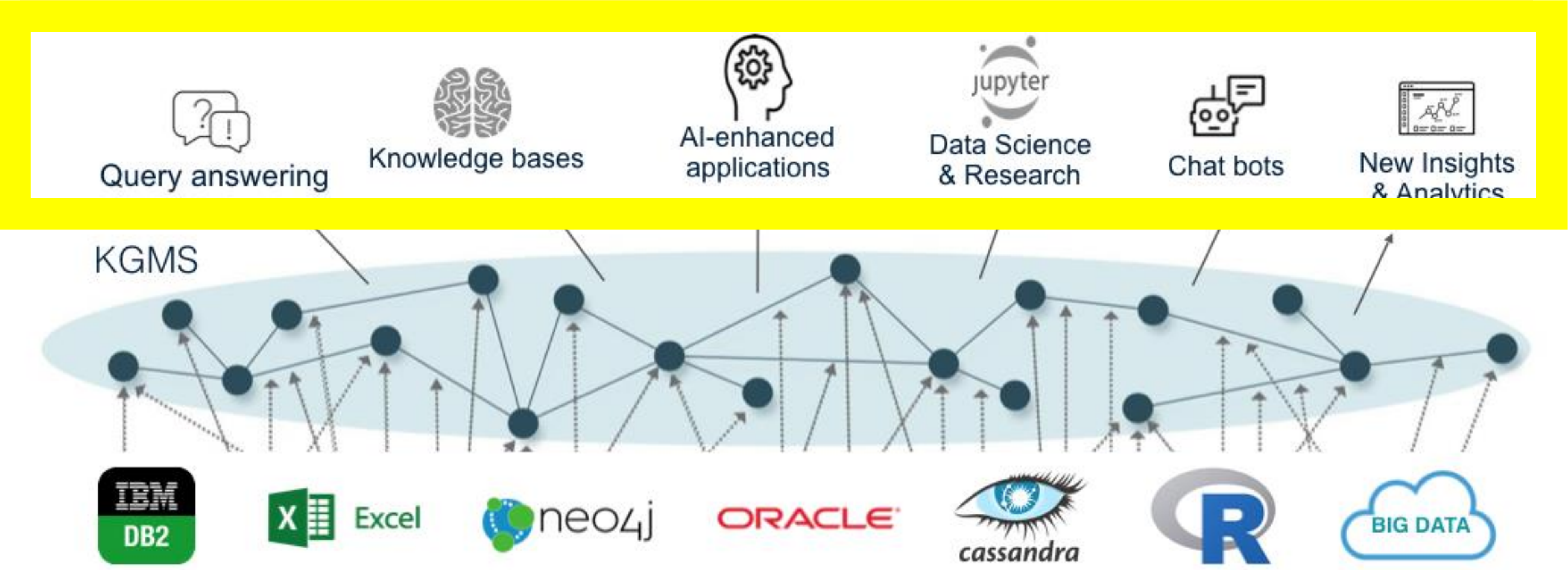
Source: Bellomarini, Fakhoury, Gottlob, Sallinger. Knowledge Graphs and Enterprise AI: The Promise of an Enabling Technology. ICDE 2019.

1. provides a language and a formalism
   for representation of and reasoning on KGs
2. can access a rich set of data sources (including big data);
3. **can embed procedural and third-party code**
4. provides reasoning services via a rich set of APIs.



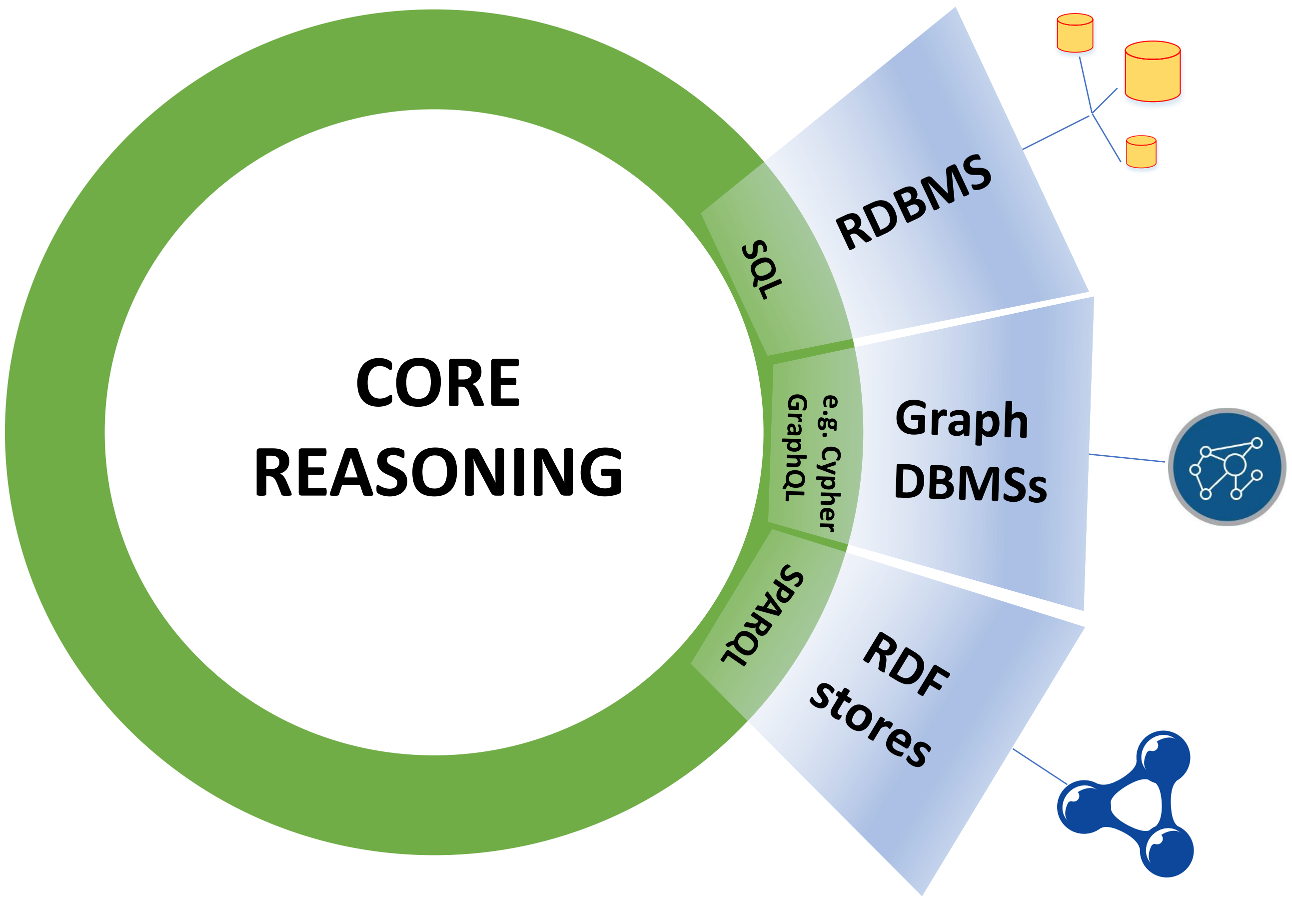
Source: Bellomarini, Fakhoury, Gottlob, Sallinger. Knowledge Graphs and Enterprise AI: The Promise of an Enabling Technology. ICDE 2019.

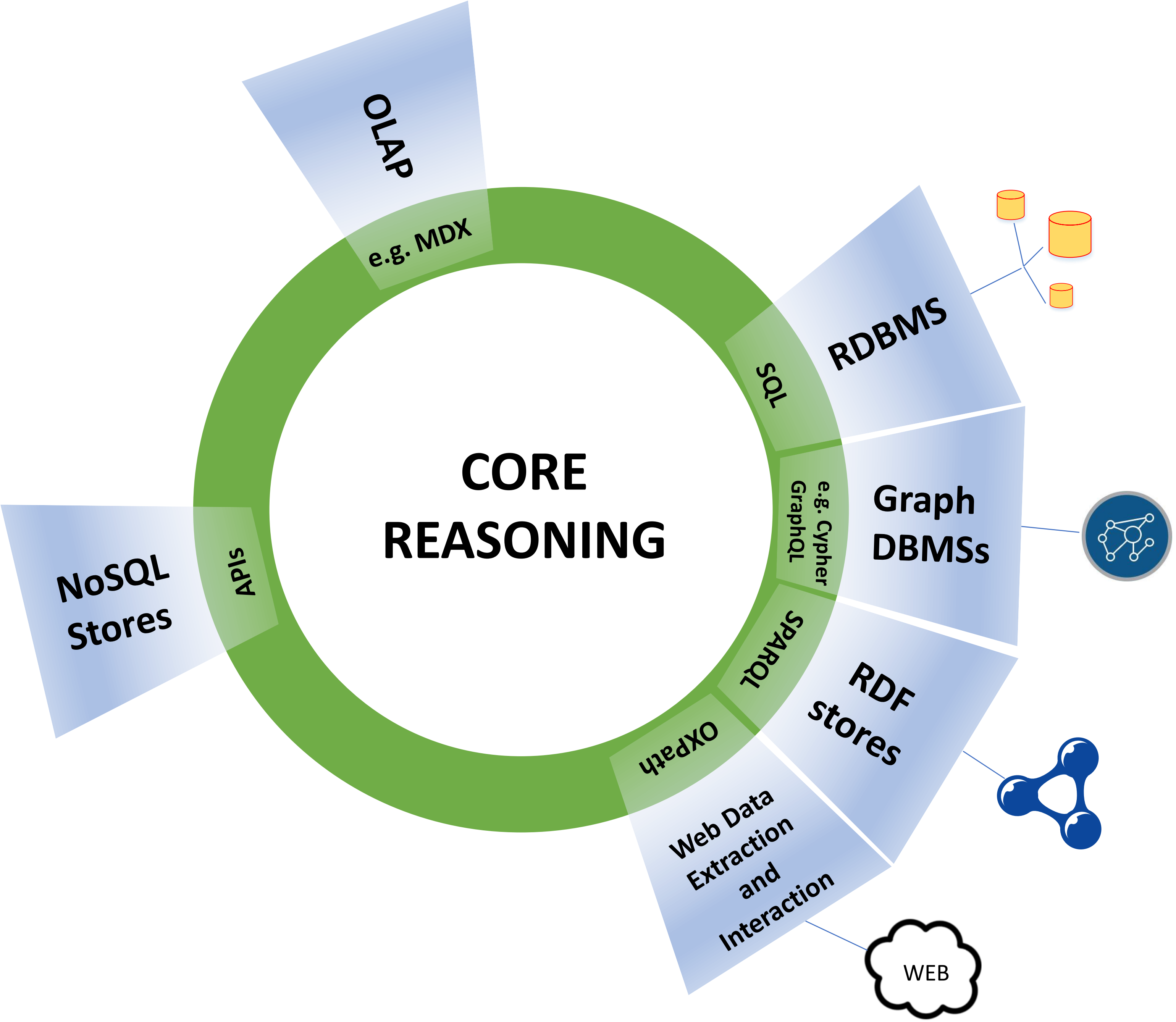
1. provides a language and a formalism
   for representation of and reasoning on KGs
2. can access a rich set of data sources (including big data);
3. can embed procedural and third-party code
4. **provides reasoning services via a rich set of APIs.**

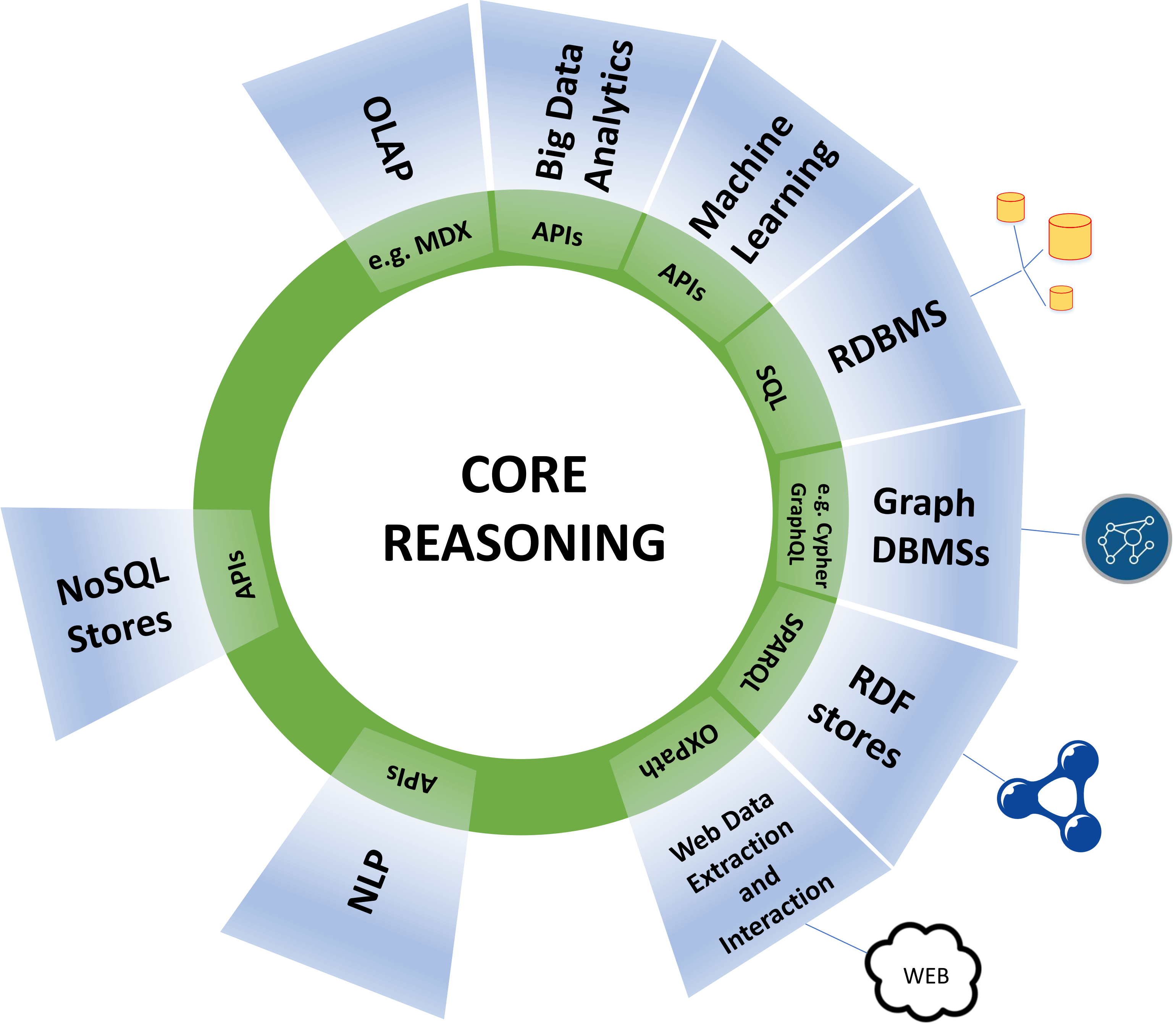


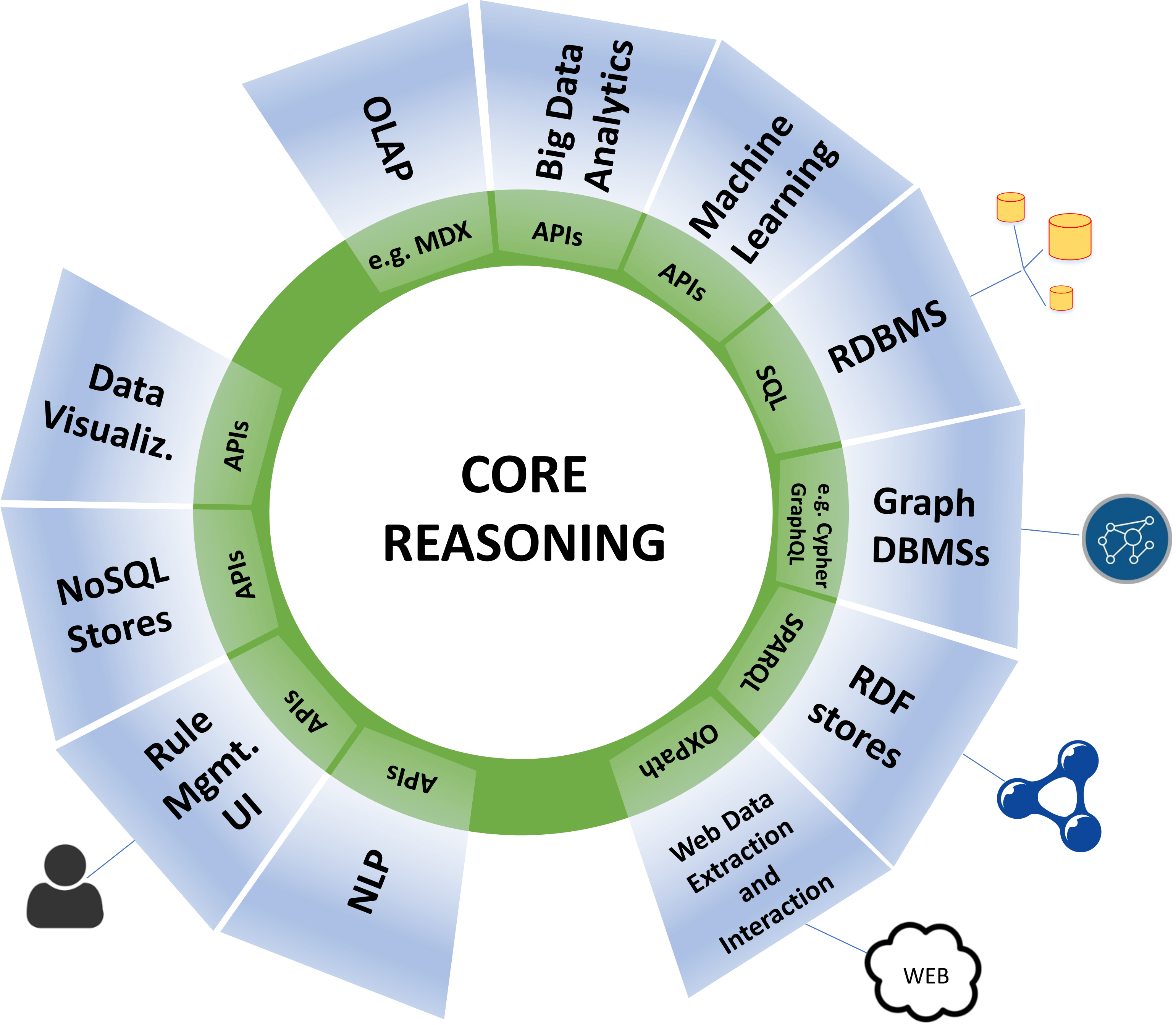
Source: Bellomarini, Fakhoury, Gottlob, Sallinger. Knowledge Graphs and Enterprise AI: The Promise of an Enabling Technology. ICDE 2019.

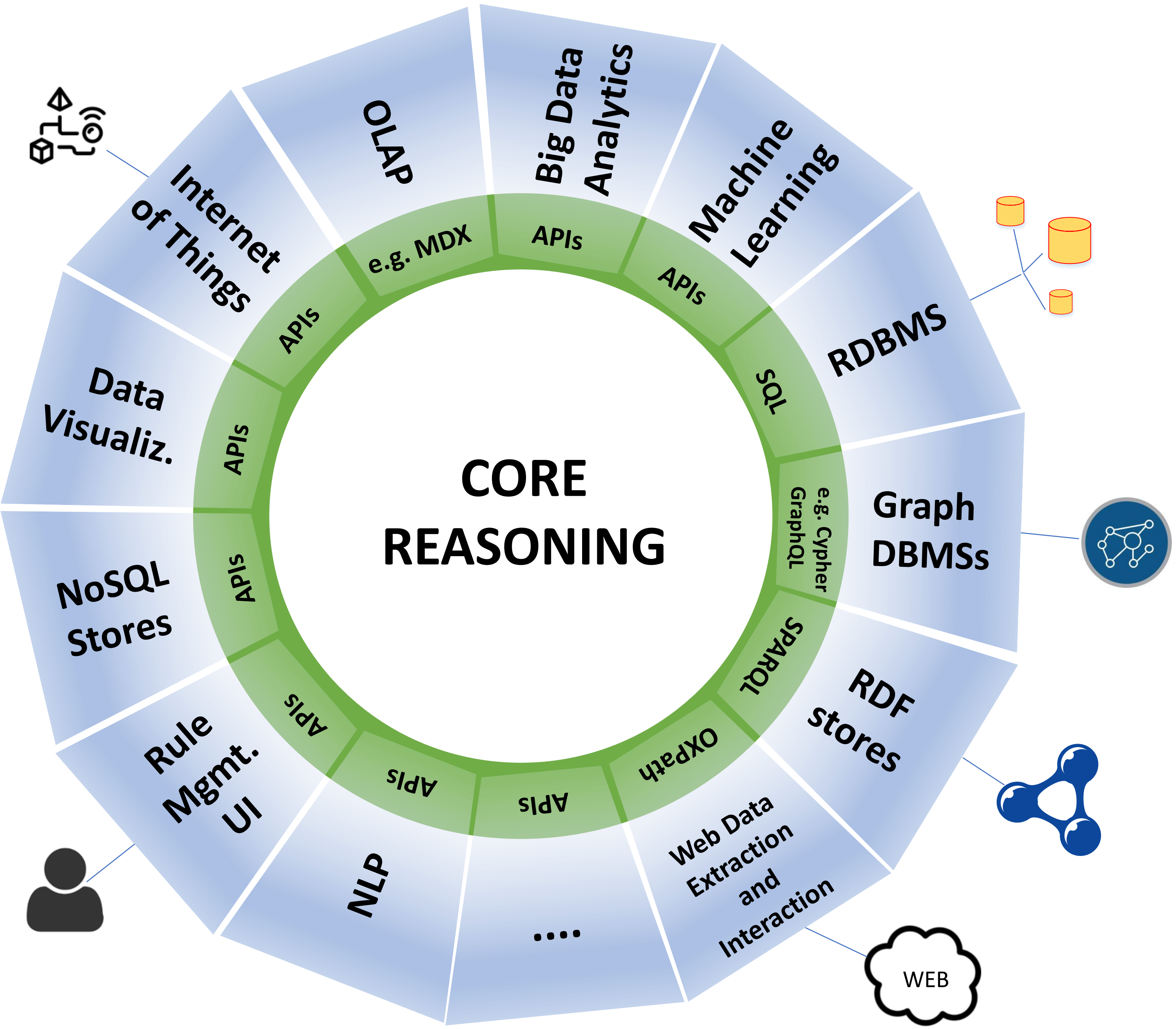
CORE REASONING

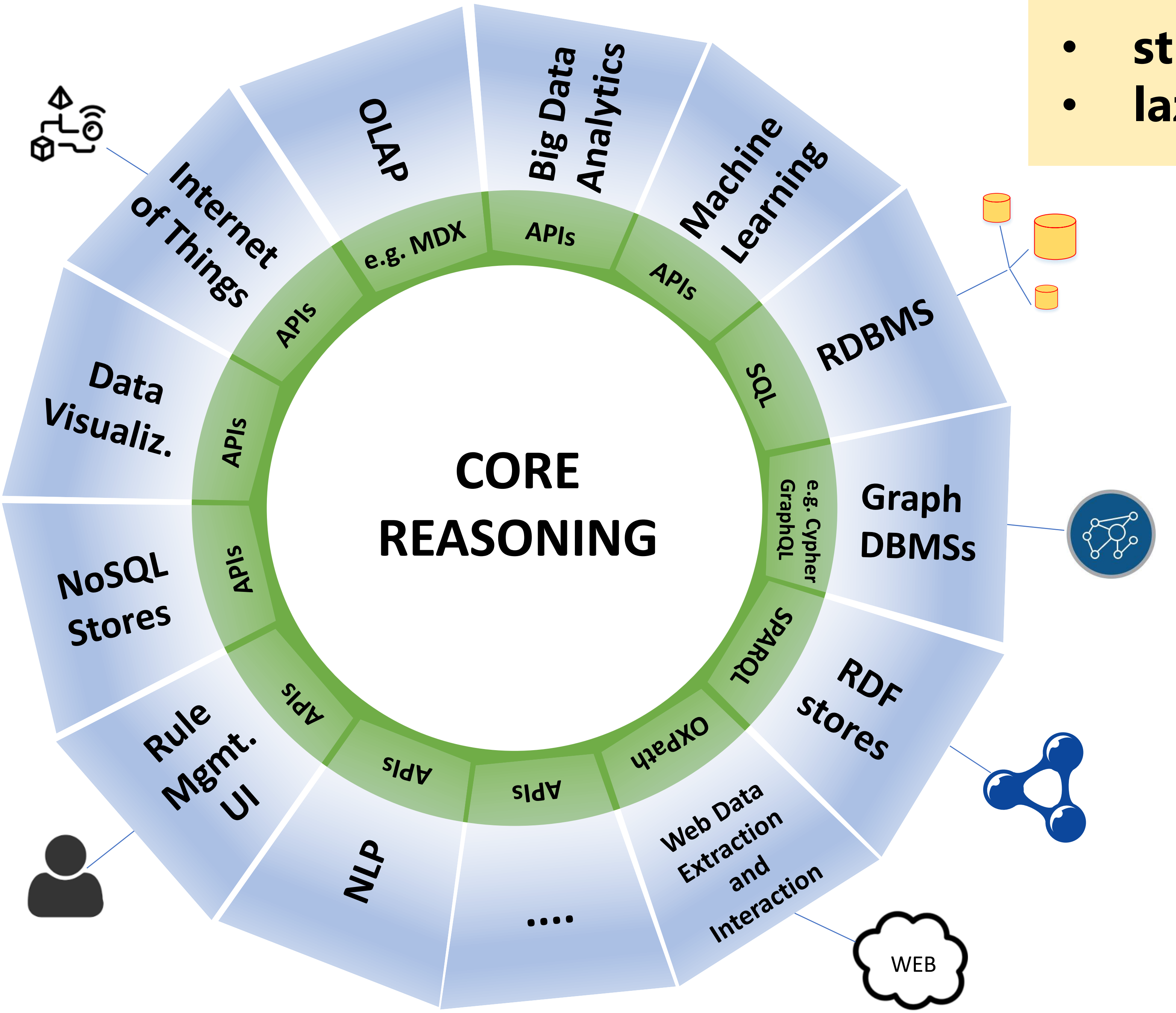

CORE
REASONING
SQL
e.g. Cypher
GraphQL
SPARQL
RDBMS
Graph
DBMSs
RDF
stores


CORE REASONING
OLAP
e.g. MDX
RDBMS
SQL
Graph DBMSs
e.g. Cypher GraphQL
RDF stores
SPARQL
Web Data Extraction and Interaction
OXPath
WEB
NoSQL Stores
APIs


CORE
REASONING
OLAP
e.g. MDX
Big Data
Analytics
APIs
Machine
Learning
APIs
RDBMS
SQL
Graph
DBMSs
e.g. Cypher
GraphQL
RDF
stores
SPARQL
Web Data
Extraction
and
Interaction
OXPath
WEB
NLP
APIs
NoSQL
Stores
APIs

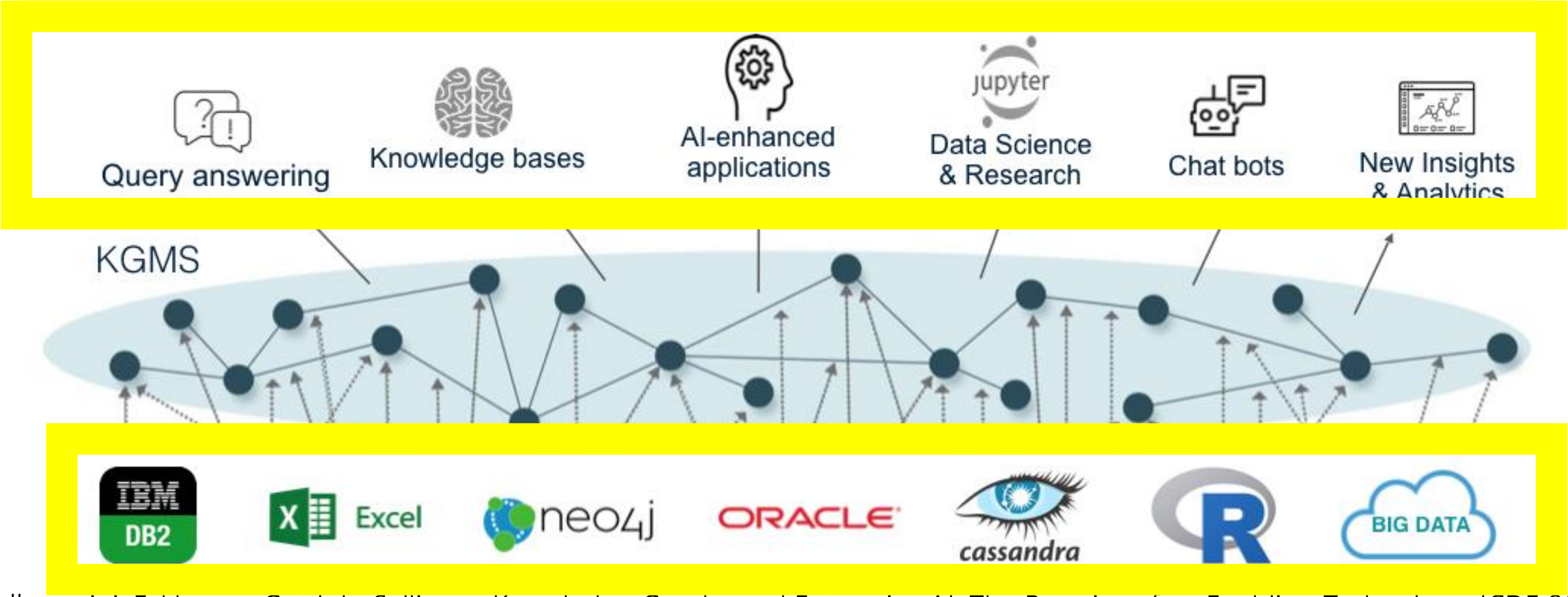

CORE REASONING
OLAP
e.g. MDX
Big Data Analytics
APIs
Machine Learning
APIs
RDBMS
SQL
Graph DBMSs
e.g. Cypher GraphQL
RDF stores
SPARQL
Web Data Extraction and Interaction
OXPath
WEB
NLP
APIs
Rule Mgmt. UI
APIs
NoSQL Stores
APIs
Data Visualiz.
APIs


CORE
REASONING
OLAP
Big Data Analytics
Machine Learning
RDBMS
Graph DBMSs
RDF stores
Web Data Extraction and Interaction
....
NLP
Rule Mgmt. UI
NoSQL Stores
Data Visualiz.
Internet of Things
e.g. MDX
APIs
APIs
SQL
e.g. Cypher GraphQL
SPARQL
OXPath
APIs
APIs
APIs
APIs
APIs
APIs
WEB



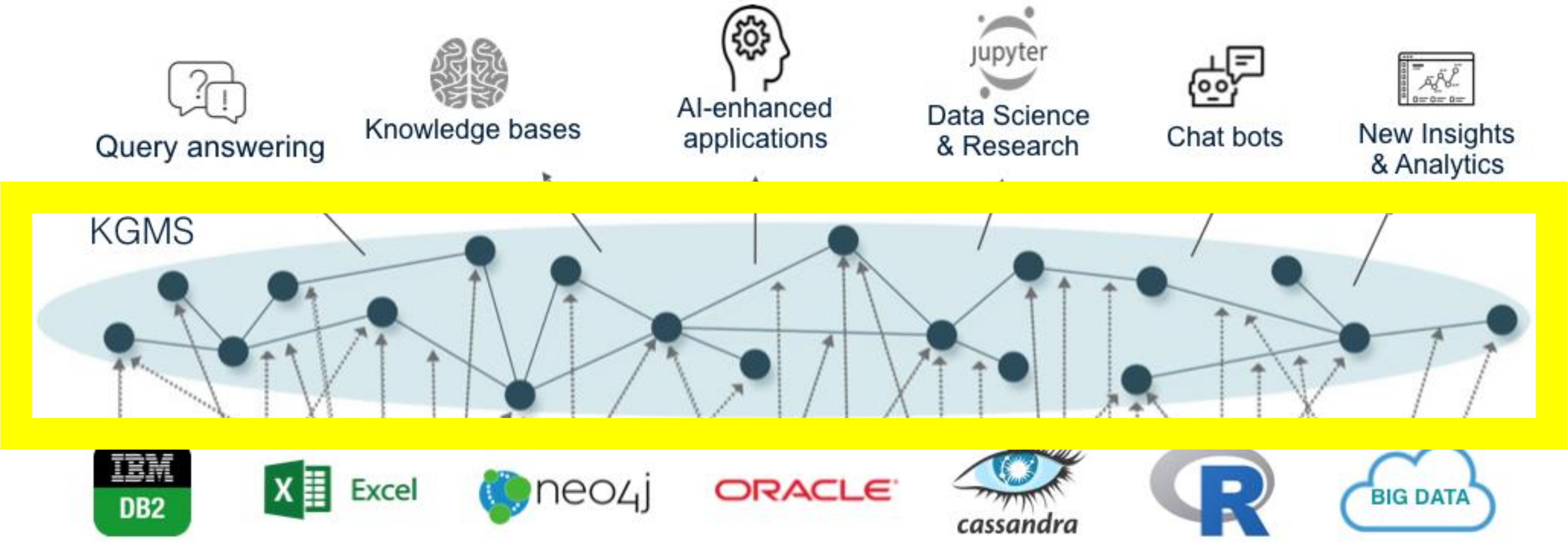
- streamable
- lazy

1. provides a language and a formalism
   for representation of and reasoning on KGs
2. **can access a rich set of data sources (including big data);**
3. **can embed procedural and third-party code**
4. provides reasoning services via a rich set of APIs.

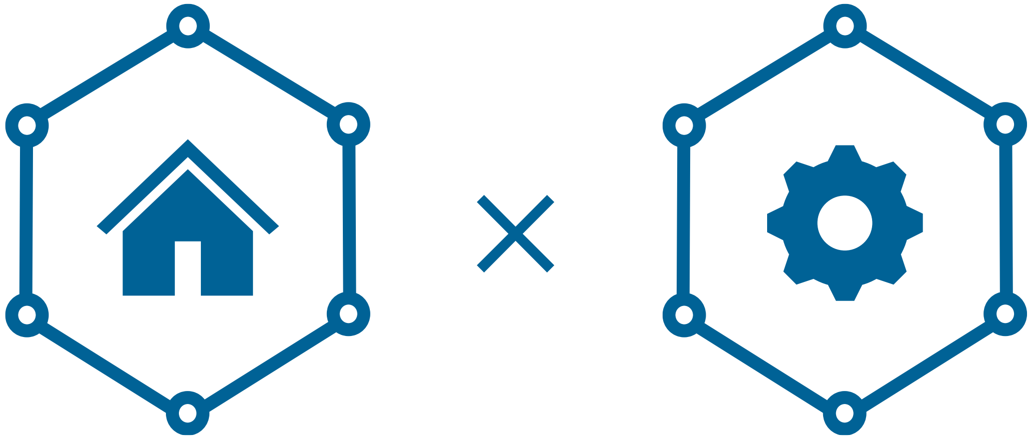


Source: Bellomarini, Fakhoury, Gottlob, Sallinger. Knowledge Graphs and Enterprise AI: The Promise of an Enabling Technology. ICDE 2019.

1. **provides a language and a formalism for representation of and reasoning on KGs**
2. can access a rich set of data sources (including big data);
3. can embed procedural and third-party code
4. provides reasoning services via a rich set of APIs.



Source: Bellomarini, Fakhoury, Gottlob, Sallinger. Knowledge Graphs and Enterprise AI: The Promise of an Enabling Technology. ICDE 2019.



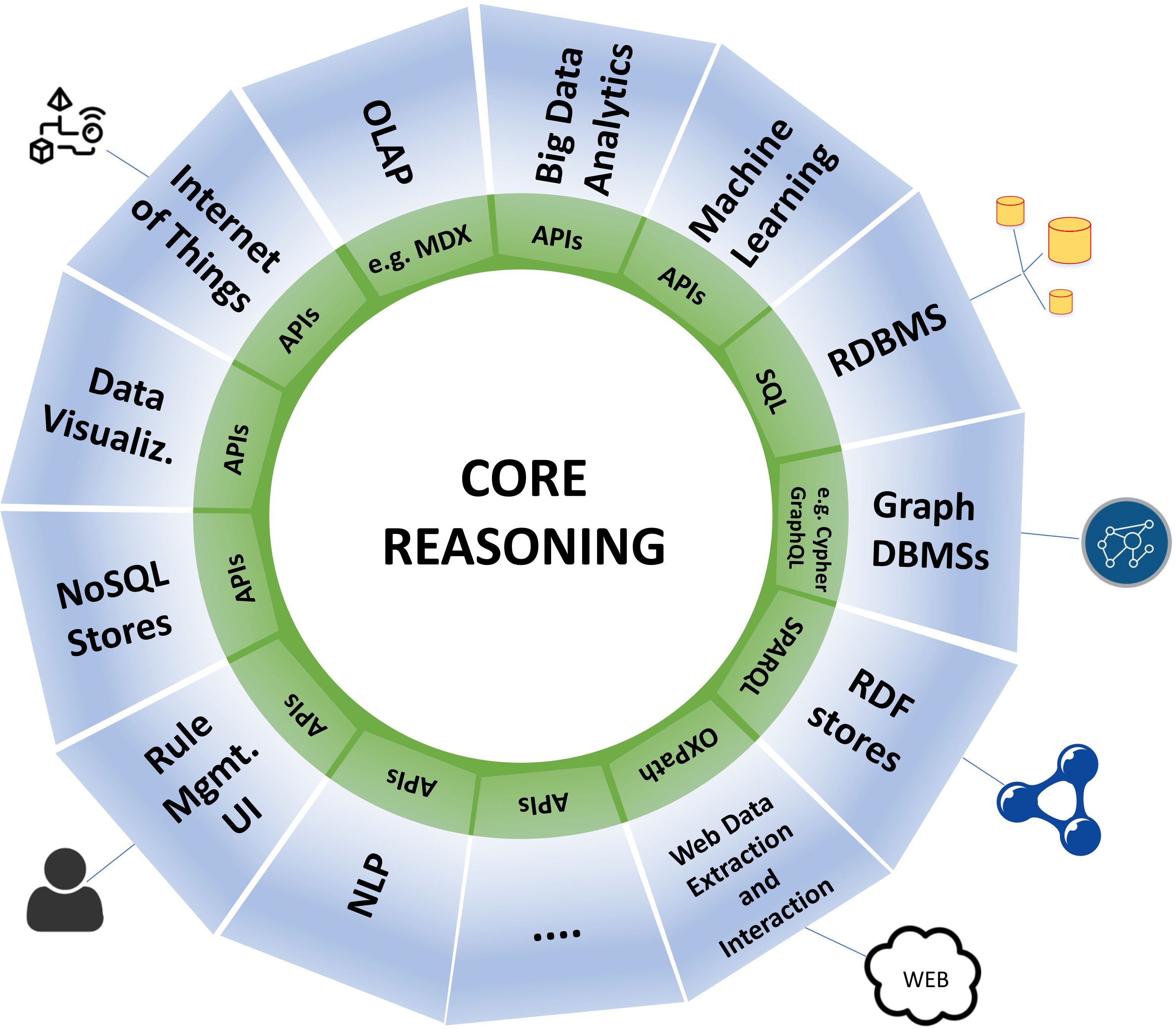
Knowledge Graph Management Systems

Emanuel Sallinger

Knowledge Graph Management Systems

Reasoning

Emanuel Sallinger


CORE
REASONING
OLAP
e.g. MDX
Big Data
Analytics
APIs
Machine
Learning
APIs
RDBMS
SQL
Graph
DBMSs
e.g. Cypher
GraphQL
RDF
stores
SPARQL
Web Data
Extraction
and
Interaction
OXPath
WEB
....
APIs
NLP
APIs
Rule
Mgmt.
UI
APIs
NoSQL
Stores
APIs
Data
Visualiz.
APIs
Internet
of Things
APIs

CORE REASONING

Reasoning Requirements

1. **Recursive** Reasoning: *full recursion over graphs*
2. **Ontological** Reasoning: *object creation, …*

Reasoning Requirements

1. **Recursive** Reasoning: *full recursion over graphs*
2. **Ontological** Reasoning: *object creation, …*
3. **Numerical** Reasoning: *numeric computation and aggregation*
4. **Probabilistic** Reasoning: *uncertain information*

Reasoning Requirements

1. **Recursive** Reasoning: *full recursion over graphs*
2. **Ontological** Reasoning: *object creation, …*
3. **Numerical** Reasoning: *numeric computation and aggregation*
4. **Probabilistic** Reasoning: *uncertain information*
5. **Subsymbolic** Reasoning: *low-dimensional spaces*

Reasoning Requirements

1. **Recursive** Reasoning: *full recursion over graphs*
2. **Ontological** Reasoning: *object creation, …*
3. **Numerical** Reasoning: *numeric computation and aggregation*
4. **Probabilistic** Reasoning: *uncertain information*
5. **Subsymbolic** Reasoning: *low-dimensional spaces*
6. **Temporal** Reasoning: *reasoning over time*

Reasoning Requirements

1. **Recursive** Reasoning: *full recursion over graphs*
2. **Ontological** Reasoning: *object creation, …*
3. **Numerical** Reasoning: *numeric computation and aggregation*
4. **Probabilistic** Reasoning: *uncertain information*
5. **Subsymbolic** Reasoning: *low-dimensional spaces*
6. **Temporal** Reasoning: *reasoning over time*
7. **Scalable** Reasoning: *coping with large datasets*

CORE REASONING

Vadalog

1. **Recursive** Reasoning:
   Full support of recursive **Datalog**
2. **Ontological** Reasoning:
   Expressive power of **SPARQL** and **OWL 2 QL**
3. **Scalable** Reasoning: **polynomial time**,
   sub-fragments that are **fully parallelizable**

Reasoning Requirements

1. **Recursive** Reasoning: *full recursion over graphs*
2. **Ontological** Reasoning: *object creation, …*
3. **Numerical** Reasoning: *numeric computation and aggregation*
4. **Probabilistic** Reasoning: *uncertain information*
5. **Subsymbolic** Reasoning: *low-dimensional spaces*
6. **Temporal** Reasoning: *reasoning over time*
7. **Scalable** Reasoning: *coping with large datasets*

Reasoning Requirements

1. **Recursive** Reasoning: *full recursion over graphs*
2. **Ontological** Reasoning: *object creation, …*
3. **Numerical** Reasoning: *numeric computation and aggregation*
4. **Probabilistic** Reasoning: *uncertain information*
5. **Subsymbolic** Reasoning: *low-dimensional spaces*
6. **Temporal** Reasoning: *reasoning over time*
7. **Scalable** Reasoning: *coping with large datasets*

Reasoning Requirements

1. **Recursive** Reasoning: *full recursion over graphs*
2. **Ontological** Reasoning: *object creation, …*
3. **Numerical** Reasoning: *numeric computation and aggregation*
4. **Probabilistic** Reasoning: *uncertain information*
5. **Subsymbolic** Reasoning: *low-dimensional spaces*
6. **Temporal** Reasoning: *reasoning over time*
7. **Scalable** Reasoning: *coping with large datasets*

Knowledge Graph Management Systems

Reasoning

Emanuel Sallinger

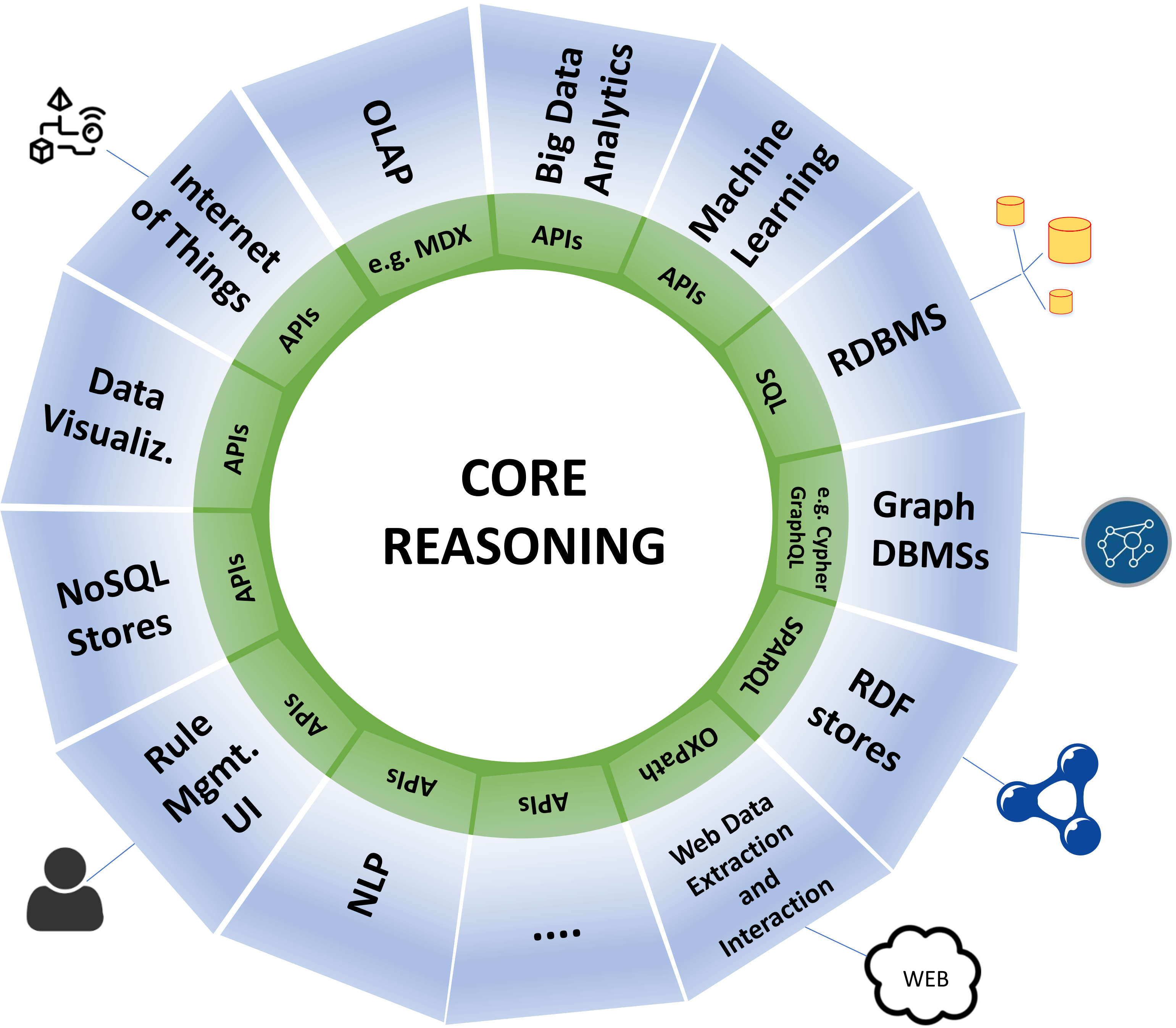
Knowledge Graph Management Systems

Vadalog

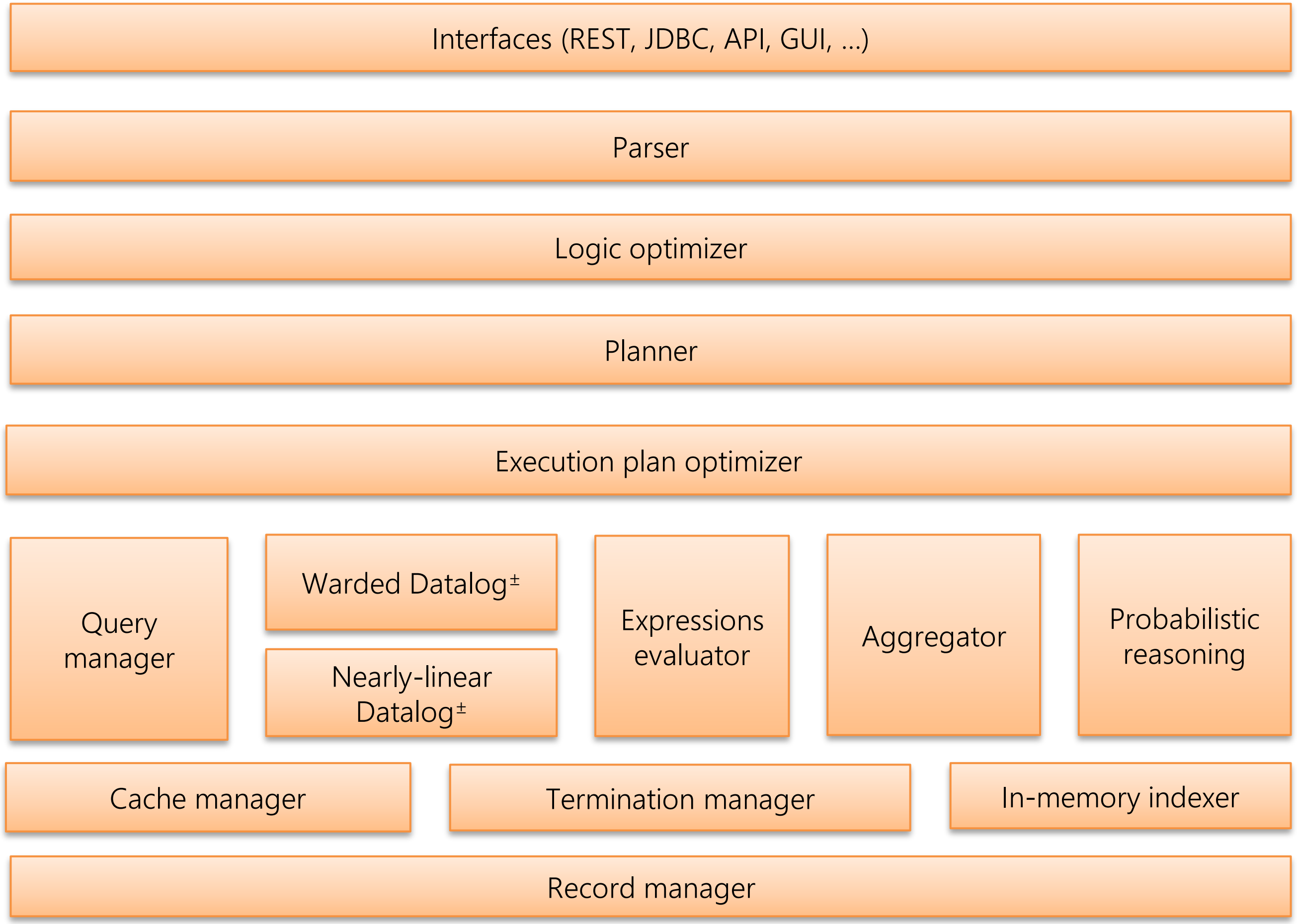
Emanuel Sallinger

Reasoning Requirements

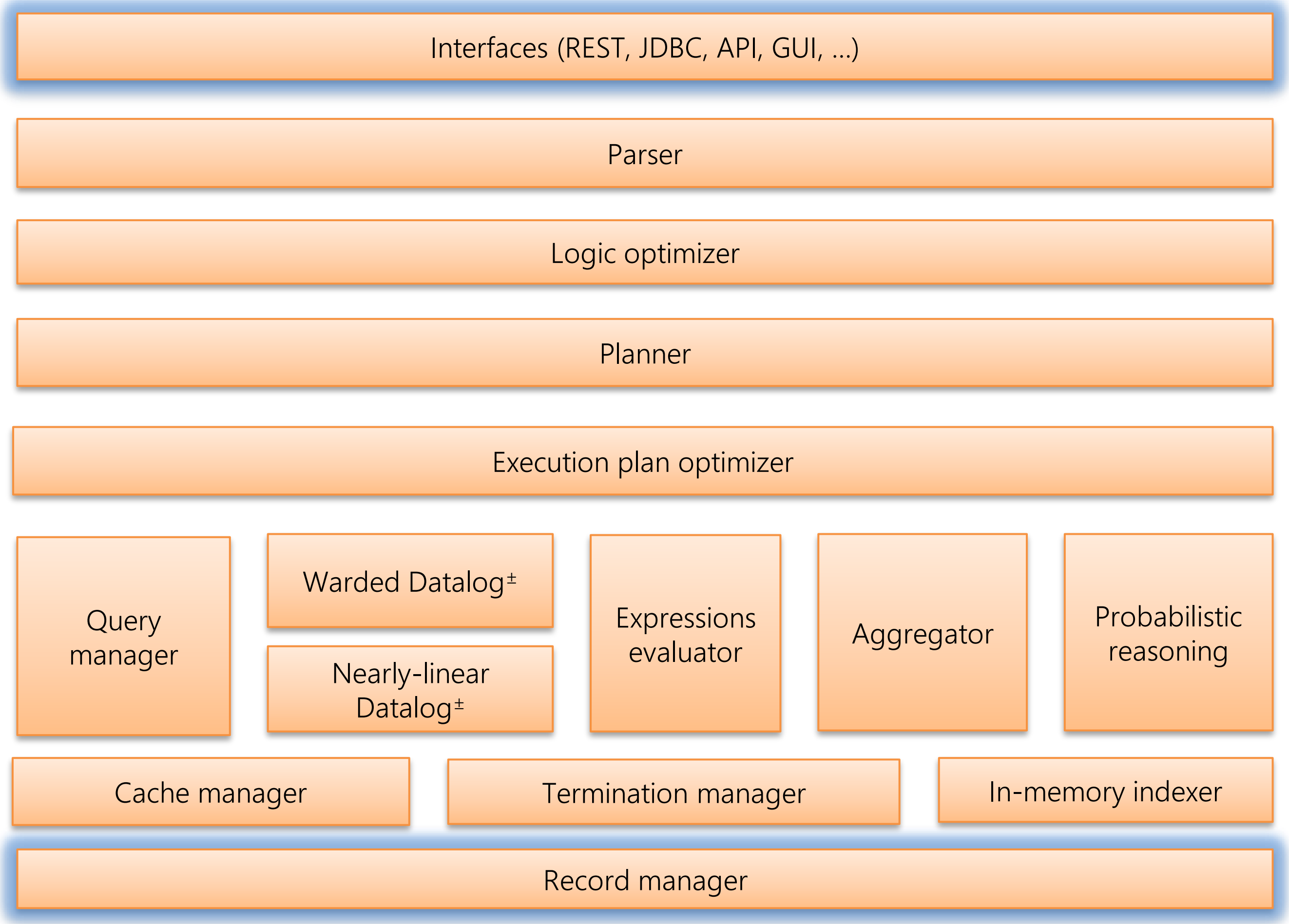
1. **Recursive** Reasoning: *full recursion over graphs*
2. **Ontological** Reasoning: *object creation, …*
3. **Numerical** Reasoning: *numeric computation and aggregation*
4. **Probabilistic** Reasoning: *uncertain information*
5. **Subsymbolic** Reasoning: *low-dimensional spaces*
6. **Temporal** Reasoning: *reasoning over time*
7. **Scalable** Reasoning: *coping with large datasets*

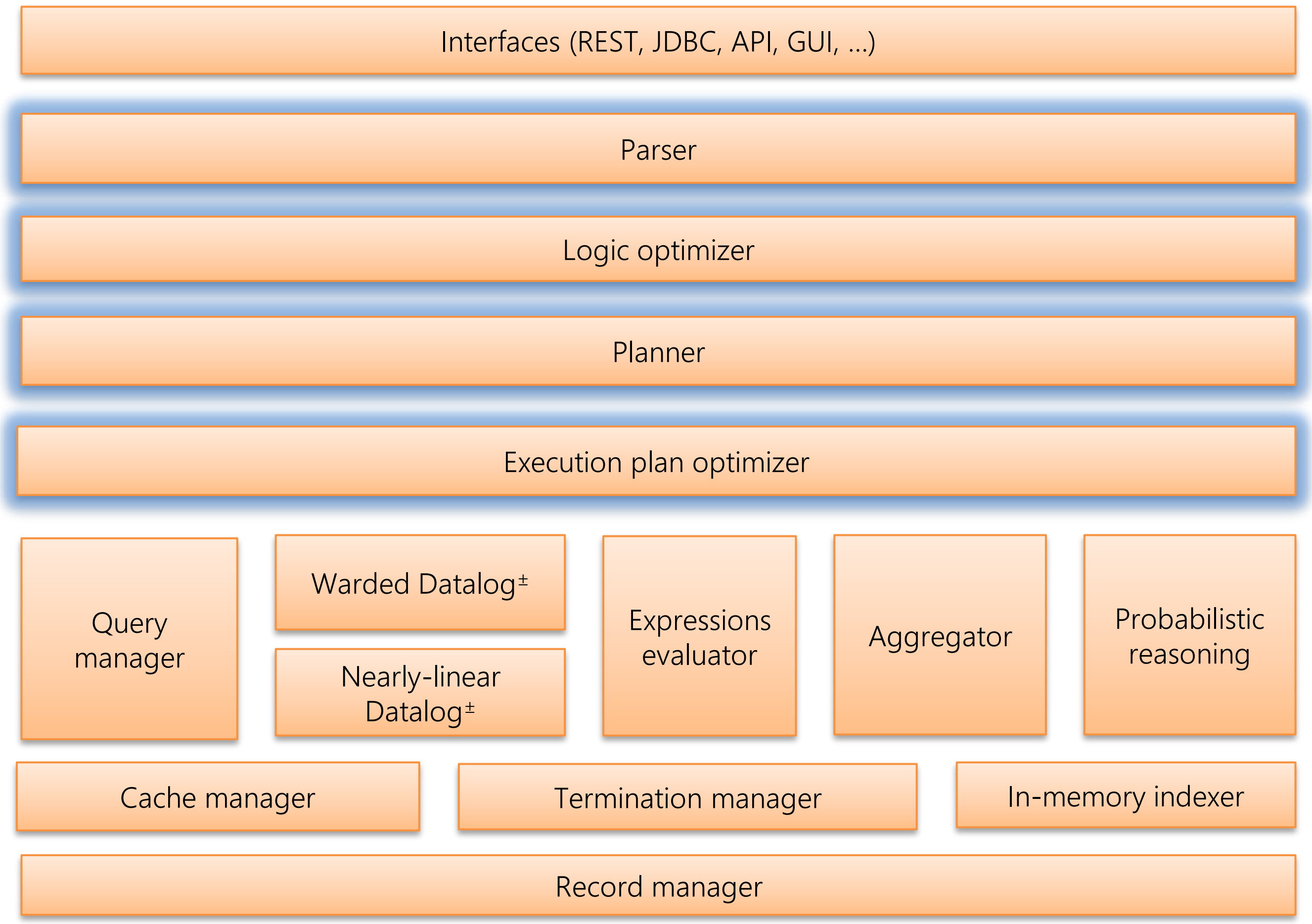

CORE
REASONING
OLAP
e.g. MDX
Big Data Analytics
APIs
Machine Learning
APIs
RDBMS
SQL
Graph DBMSs
e.g. Cypher GraphQL
RDF stores
SPARQL
Web Data Extraction and Interaction
OXPath
WEB
....
APIs
NLP
APIs
Rule Mgmt. UI
APIs
NoSQL Stores
APIs
Data Visualiz.
APIs
Internet of Things
APIs

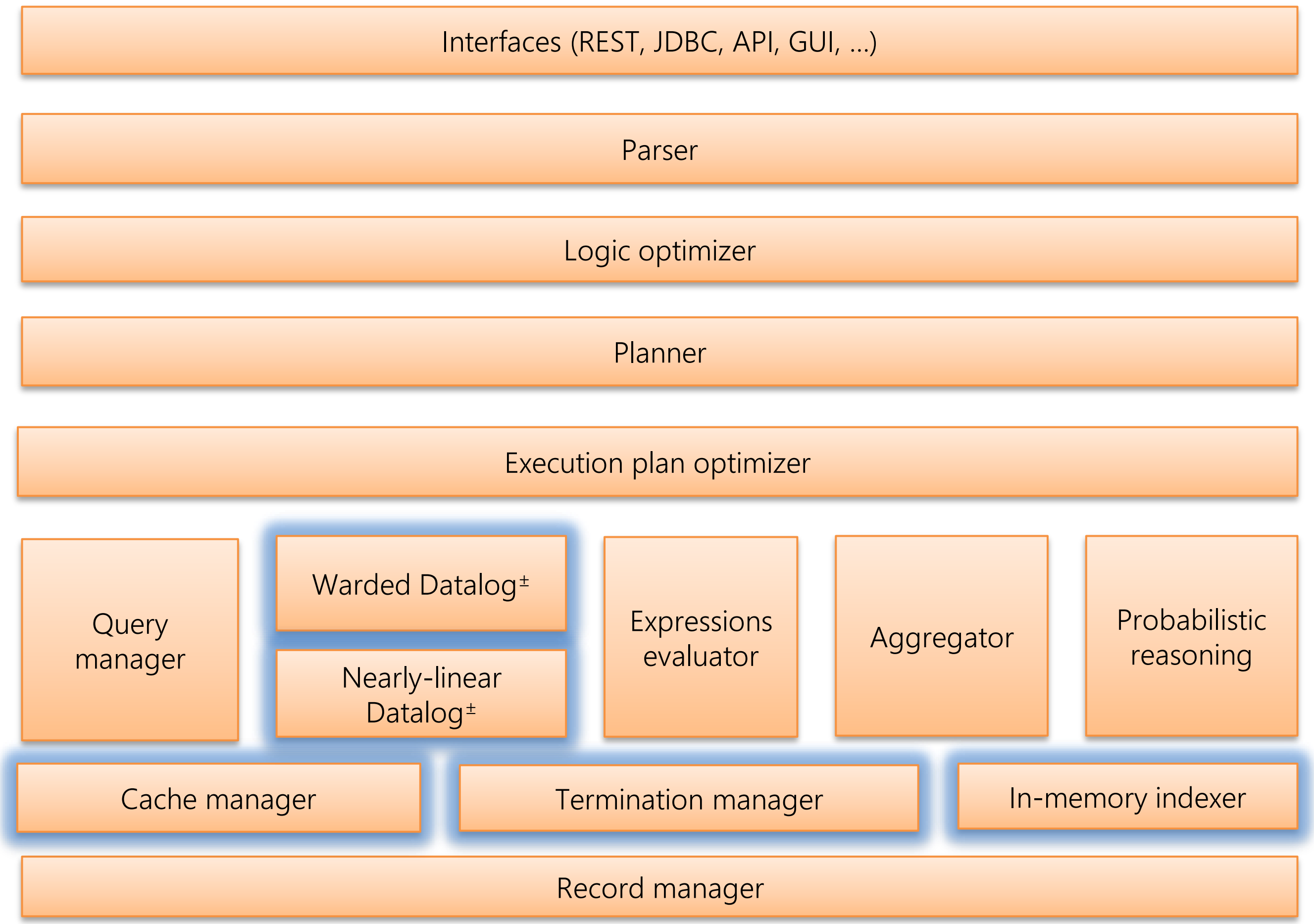
CORE REASONING



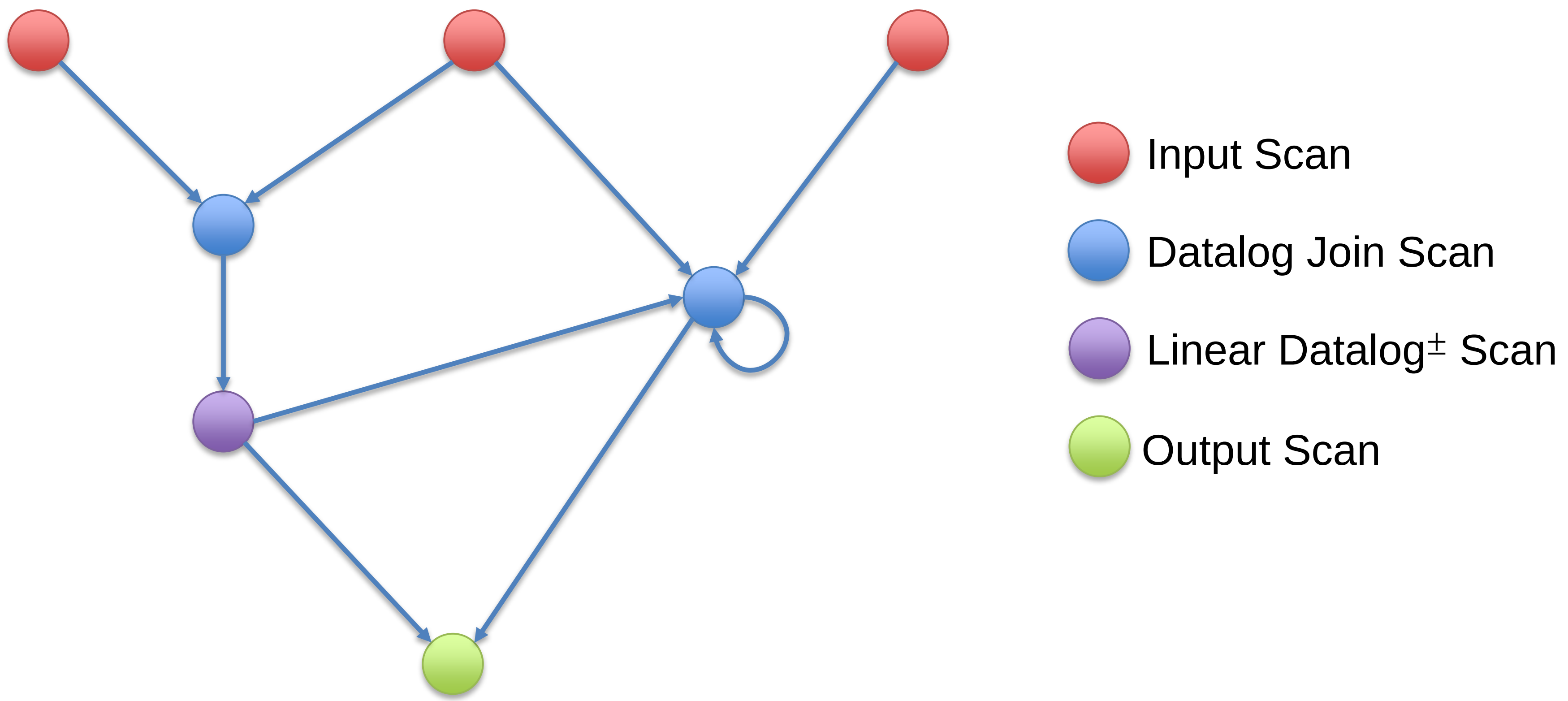
Interfaces (REST, JDBC, API, GUI, ...)
Parser
Logic optimizer
Planner
Execution plan optimizer
Query manager
Warded Datalog±
Nearly-linear Datalog±
Expressions evaluator
Aggregator
Probabilistic reasoning
Cache manager
Termination manager
In-memory indexer
Record manager


Interfaces (REST, JDBC, API, GUI, ...)
Parser
Logic optimizer
Planner
Execution plan optimizer
Query manager
Warded Datalog±
Nearly-linear Datalog±
Expressions evaluator
Aggregator
Probabilistic reasoning
Cache manager
Termination manager
In-memory indexer
Record manager

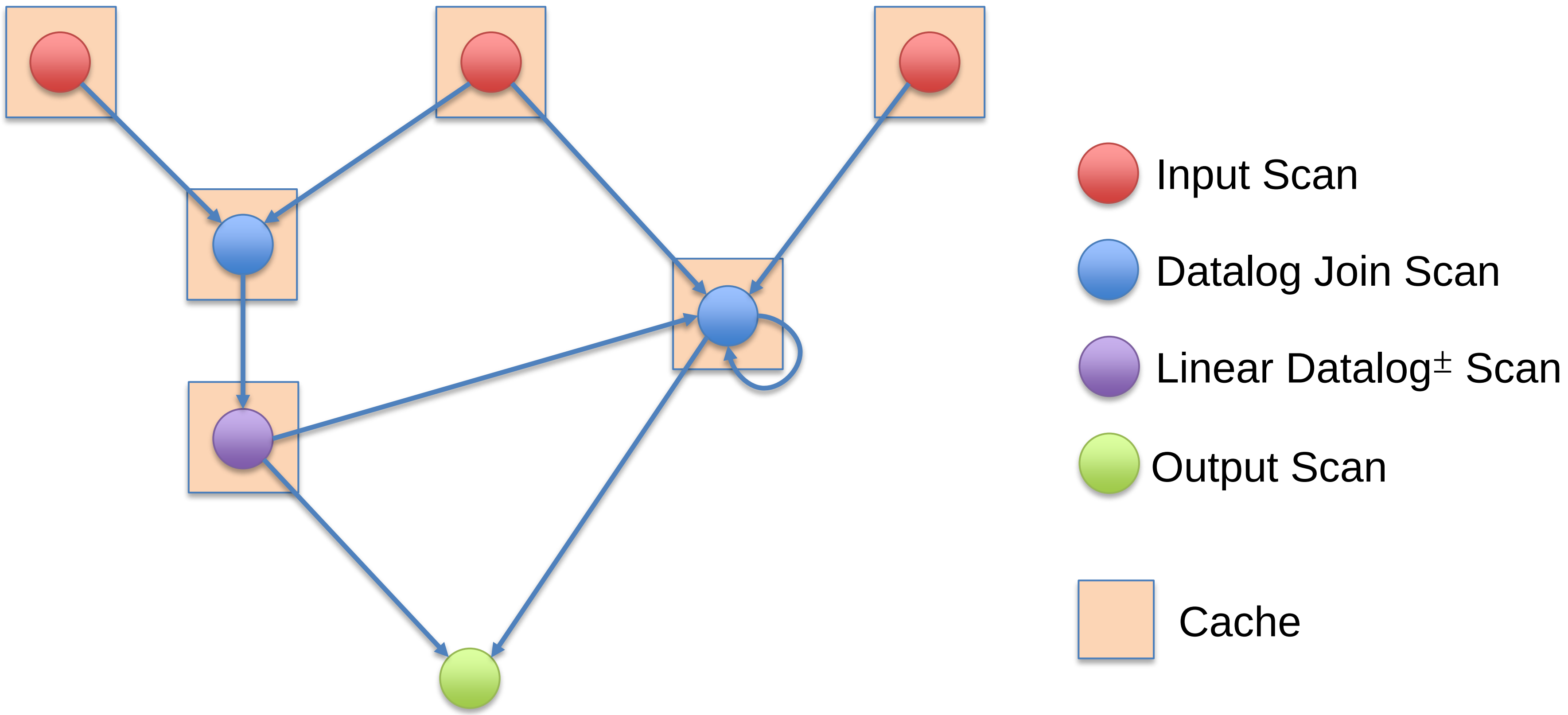

Interfaces (REST, JDBC, API, GUI, ...)
Parser
Logic optimizer
Planner
Execution plan optimizer
Query manager
Warded Datalog±
Nearly-linear Datalog±
Expressions evaluator
Aggregator
Probabilistic reasoning
Cache manager
Termination manager
In-memory indexer
Record manager


Interfaces (REST, JDBC, API, GUI, ...)
Parser
Logic optimizer
Planner
Execution plan optimizer
Query manager
Warded Datalog±
Nearly-linear Datalog±
Expressions evaluator
Aggregator
Probabilistic reasoning
Cache manager
Termination manager
In-memory indexer
Record manager

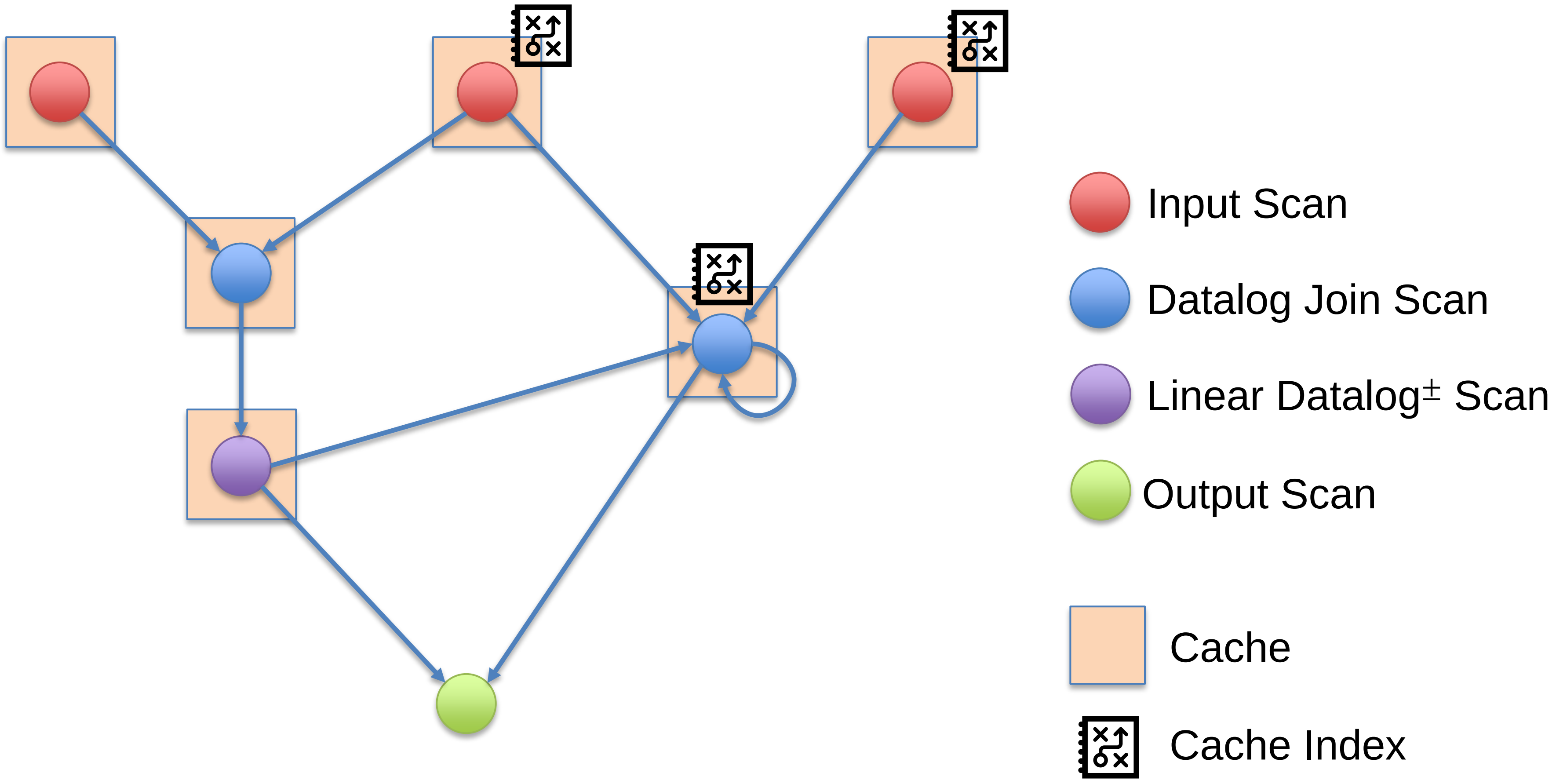
In-Memory Stream Processing



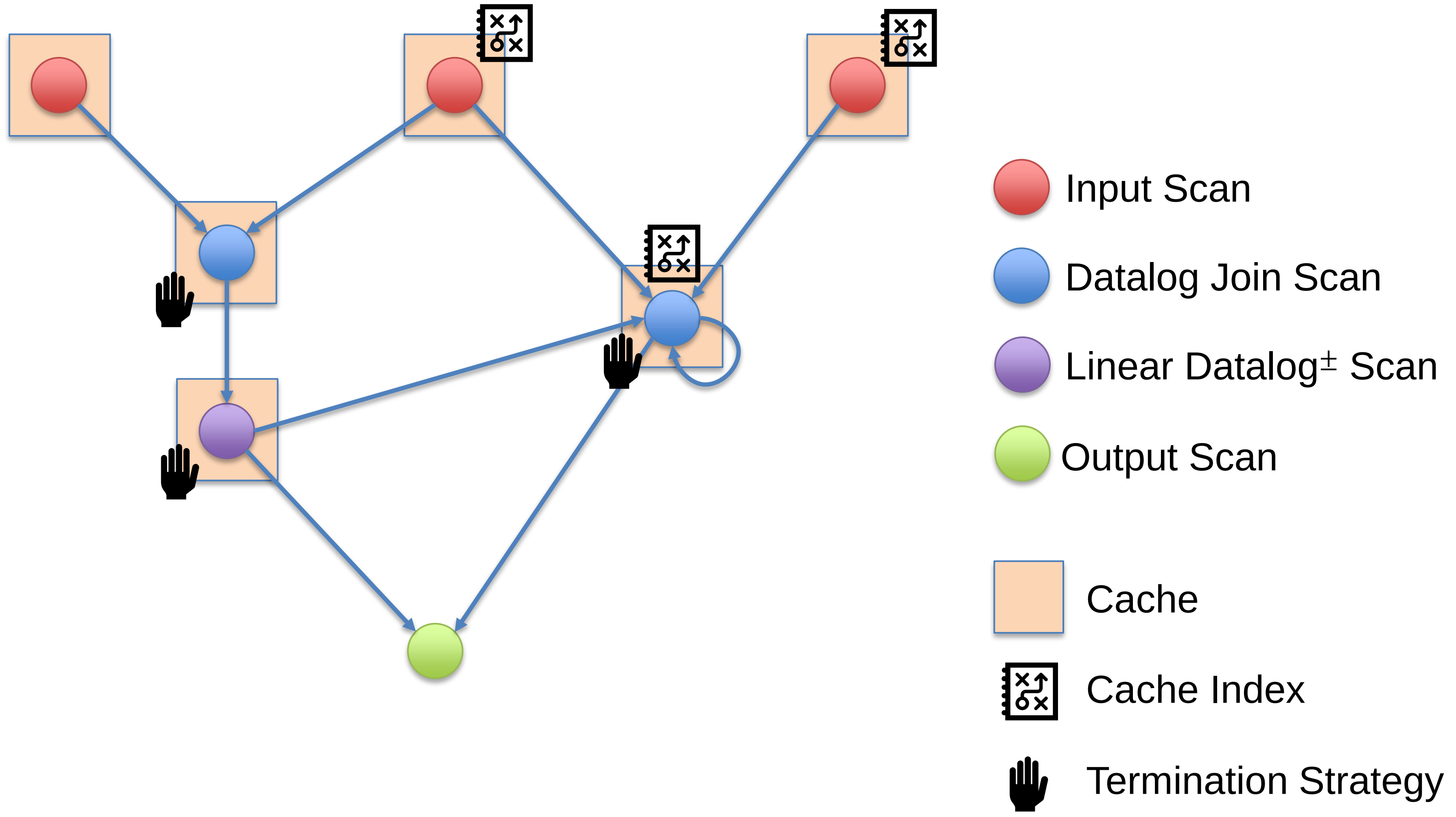
In-Memory Stream Processing

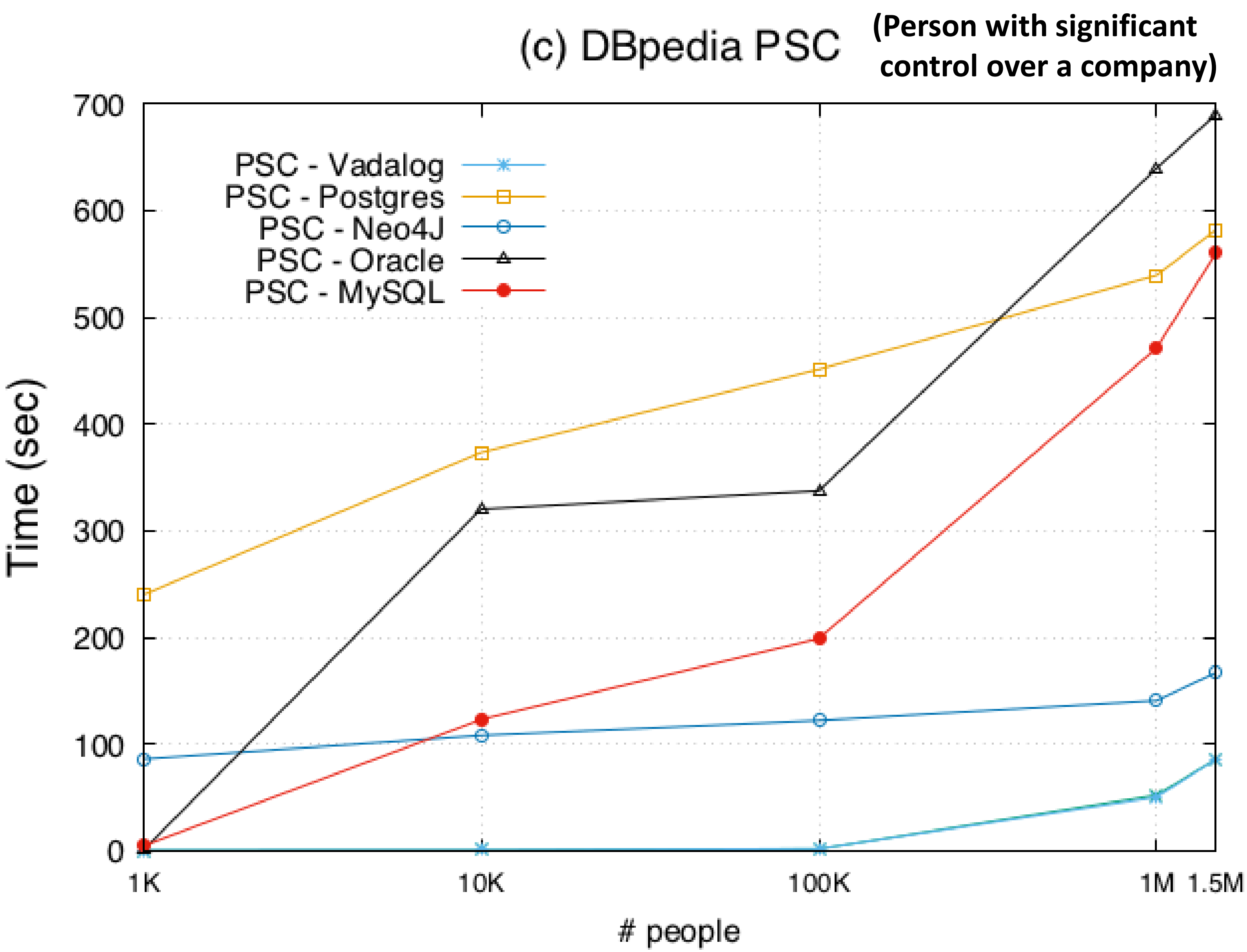


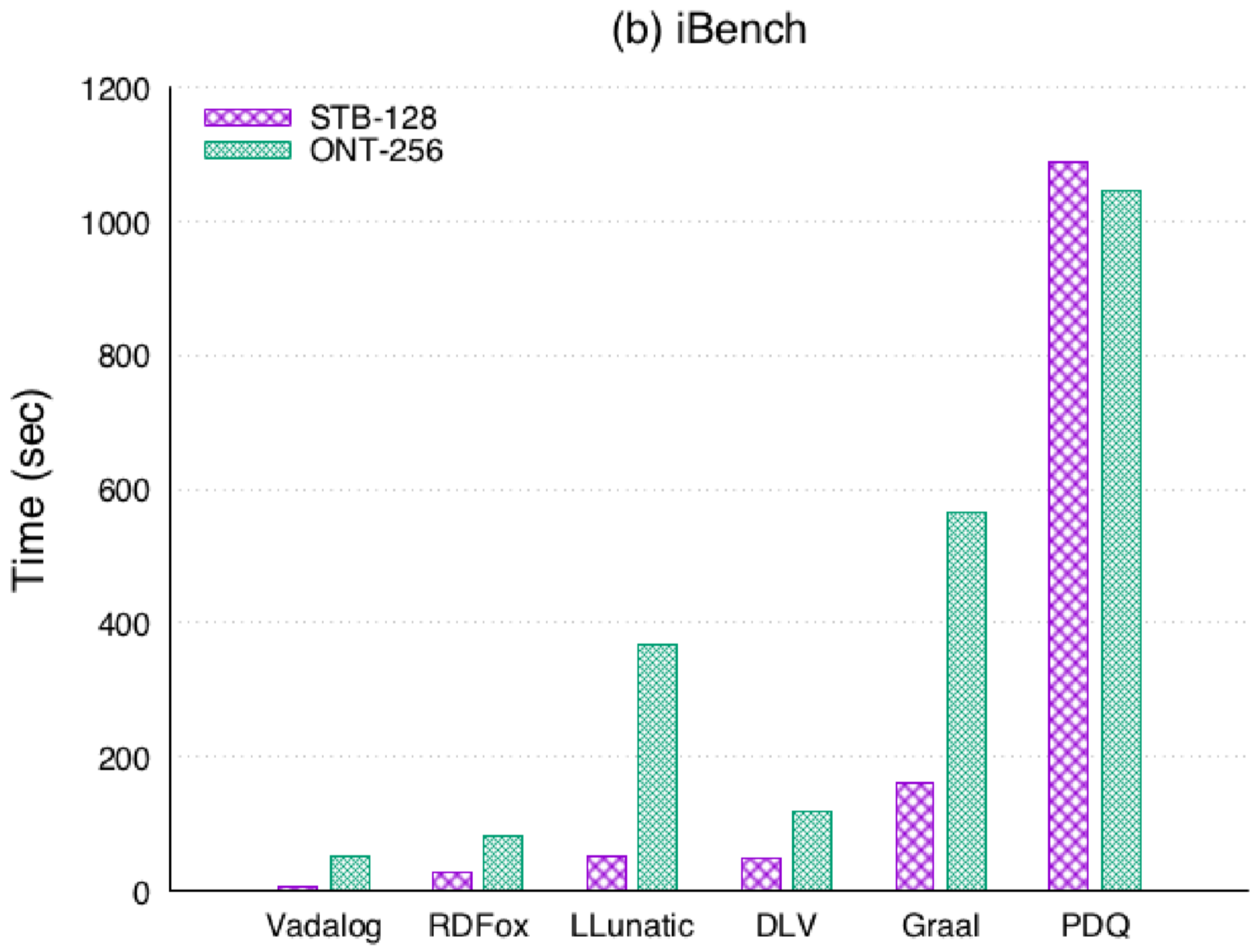
In-Memory Stream Processing

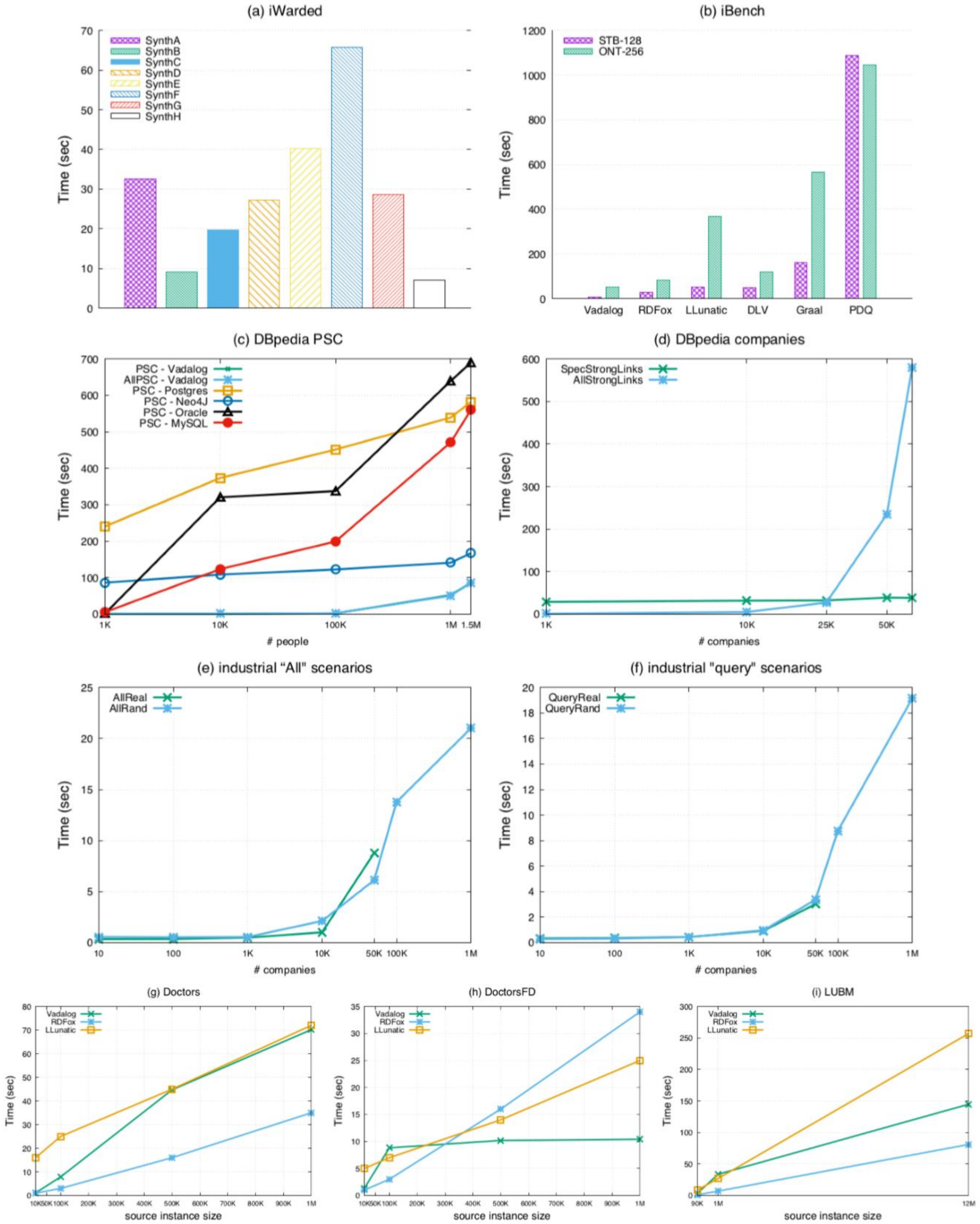


In-Memory Stream Processing




(c) DBpedia PSC
(Person with significant control over a company)
PSC - Vadalog
PSC - Postgres
PSC - Neo4J
PSC - Oracle
PSC - MySQL
Time (sec)
0
100
200
300
400
500
600
700
1K
10K
100K
1M
1.5M
people


(b) iBench
STB-128
ONT-256
Time (sec)
0
200
400
600
800
1000
1200
Vadalog
RDFox
LLunatic
DLV
Graal
PDQ


(a) iWarded
SynthA
SynthB
SynthC
SynthD
SynthE
SynthF
SynthG
SynthH
Time (sec)
(b) iBench
STB-128
ONT-256
Time (sec)
Vadalog
RDFox
LLunatic
DLV
Graal
PDQ
(c) DBpedia PSC
PSC - Vadalog
AllPSC - Vadalog
PSC - Postgres
PSC - Neo4J
PSC - Oracle
PSC - MySQL
Time (sec)
people
(d) DBpedia companies
SpecStrongLinks
AllStrongLinks
Time (sec)
companies
(e) industrial "All" scenarios
AllReal
AllRand
Time (sec)
companies
(f) industrial "query" scenarios
QueryReal
QueryRand
Time (sec)
companies
(g) Doctors
Vadalog
RDFox
LLunatic
Time (sec)
source instance size
(h) DoctorsFD
Vadalog
RDFox
LLunatic
Time (sec)
source instance size
(i) LUBM
Vadalog
RDFox
LLunatic
Time (sec)
source instance size

Knowledge Graph Management Systems

Vadalog

Emanuel Sallinger

Knowledge Graph Management Systems

Vadalog

Advanced

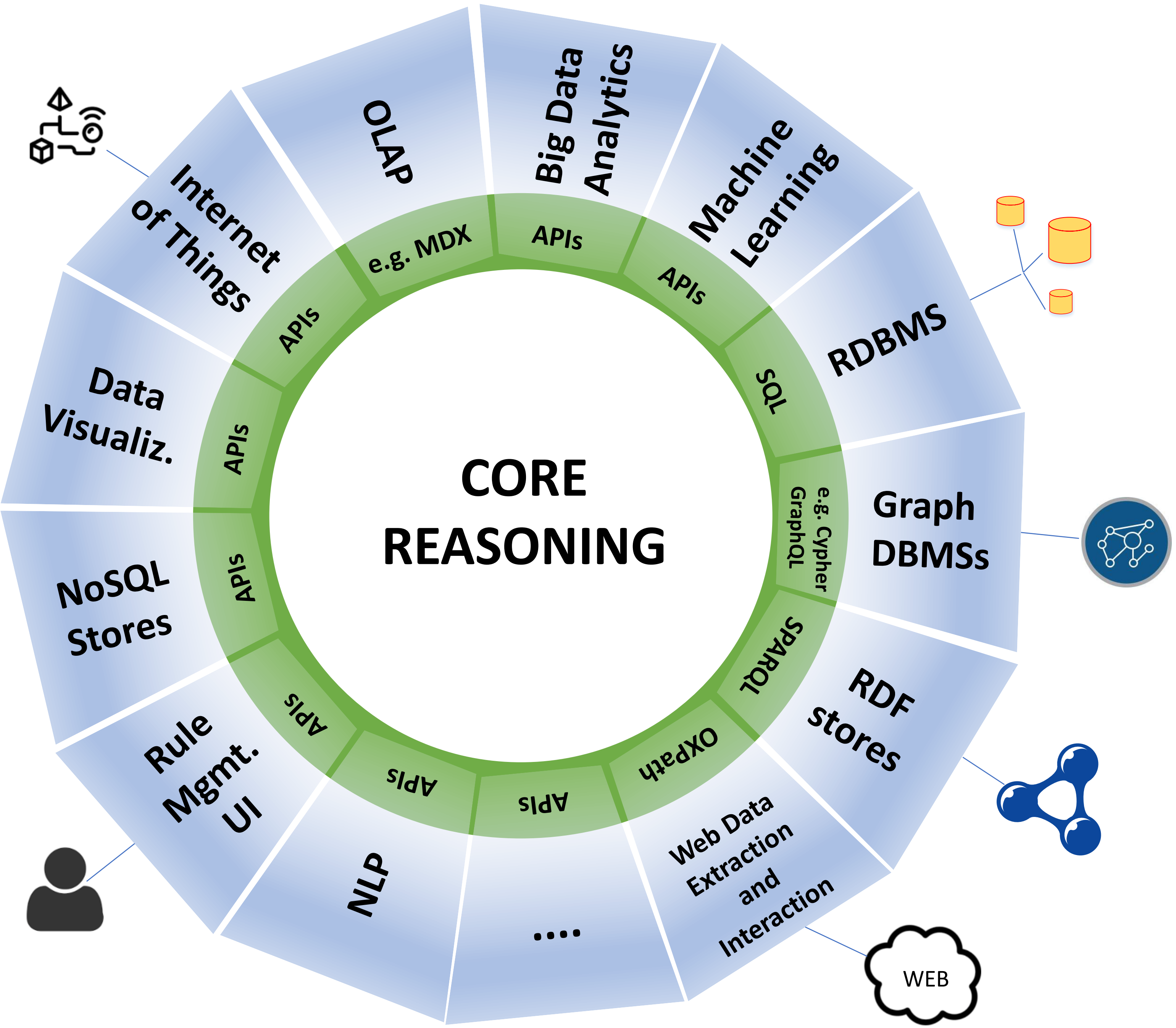
Emanuel Sallinger

WARNING: Advanced Unit

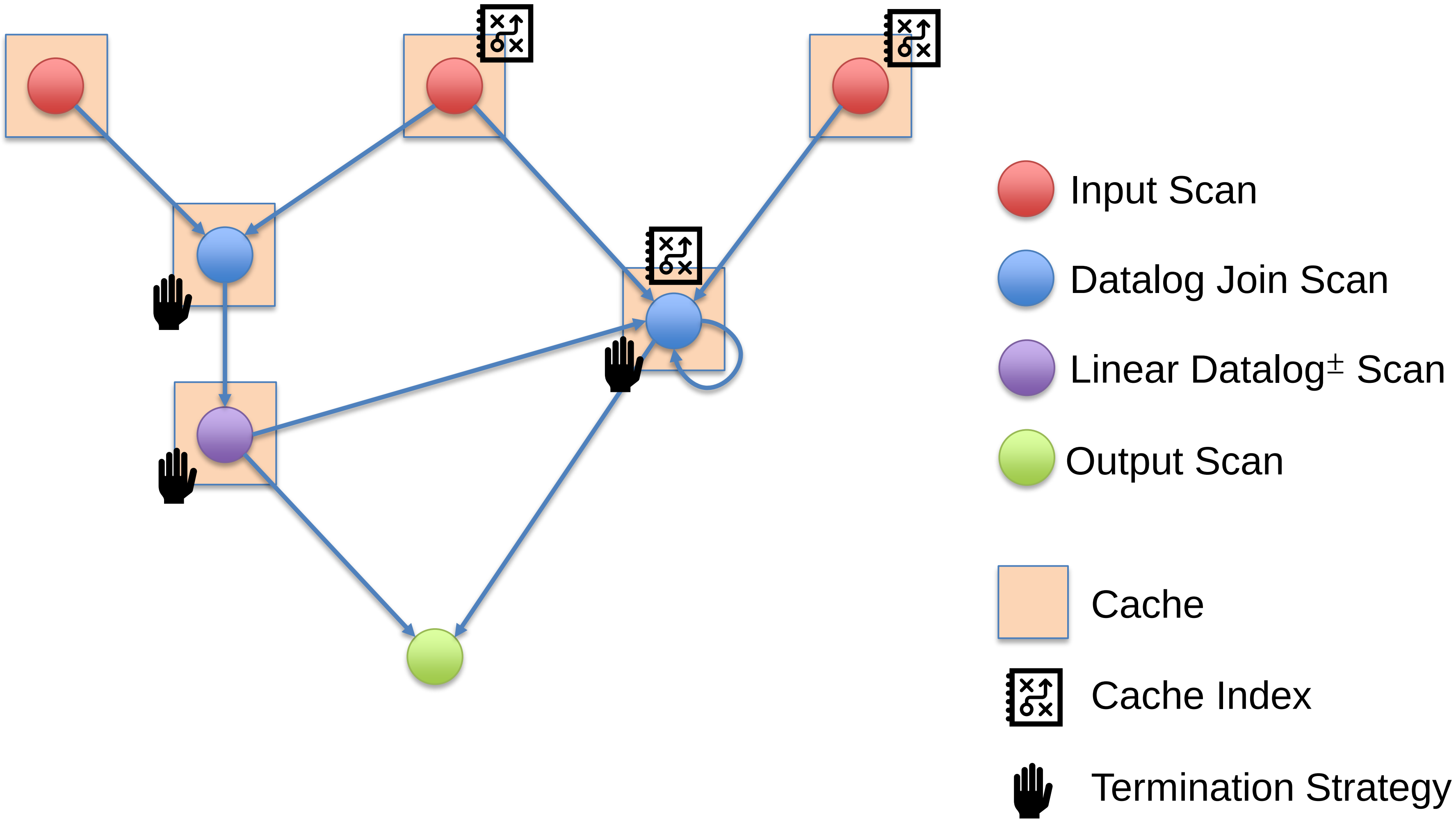
Only for the interested

Needs further reading to understand all concepts

Bellomarini, Sallinger, Gottlob.
The Vadalog System: Datalog-based Reasoning for Knowledge Graphs.
VLDB, 2018.

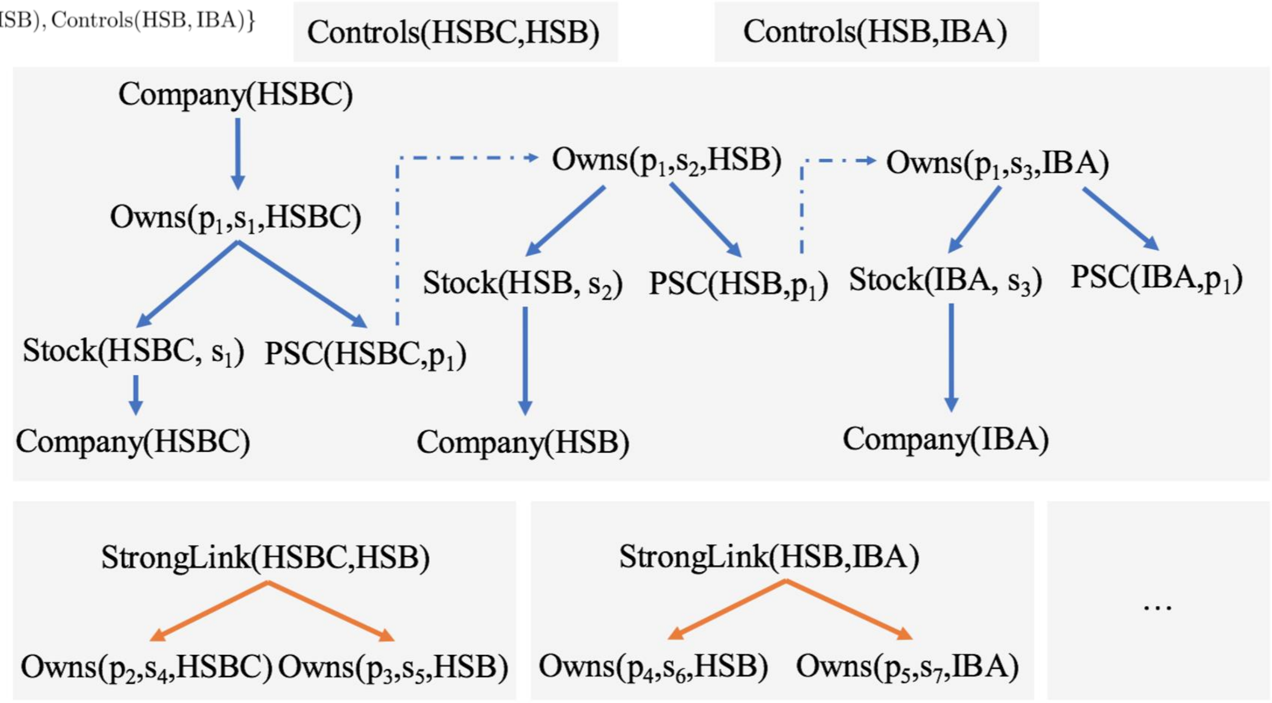

CORE REASONING
OLAP
e.g. MDX
Big Data Analytics
APIs
Machine Learning
APIs
RDBMS
SQL
Graph DBMSs
e.g. Cypher GraphQL
RDF stores
SPARQL
Web Data Extraction and Interaction
OXPath
WEB
....
APIs
NLP
APIs
Rule Mgmt. UI
APIs
NoSQL Stores
APIs
Data Visualiz.
APIs
Internet of Things
APIs

In-Memory Stream Processing



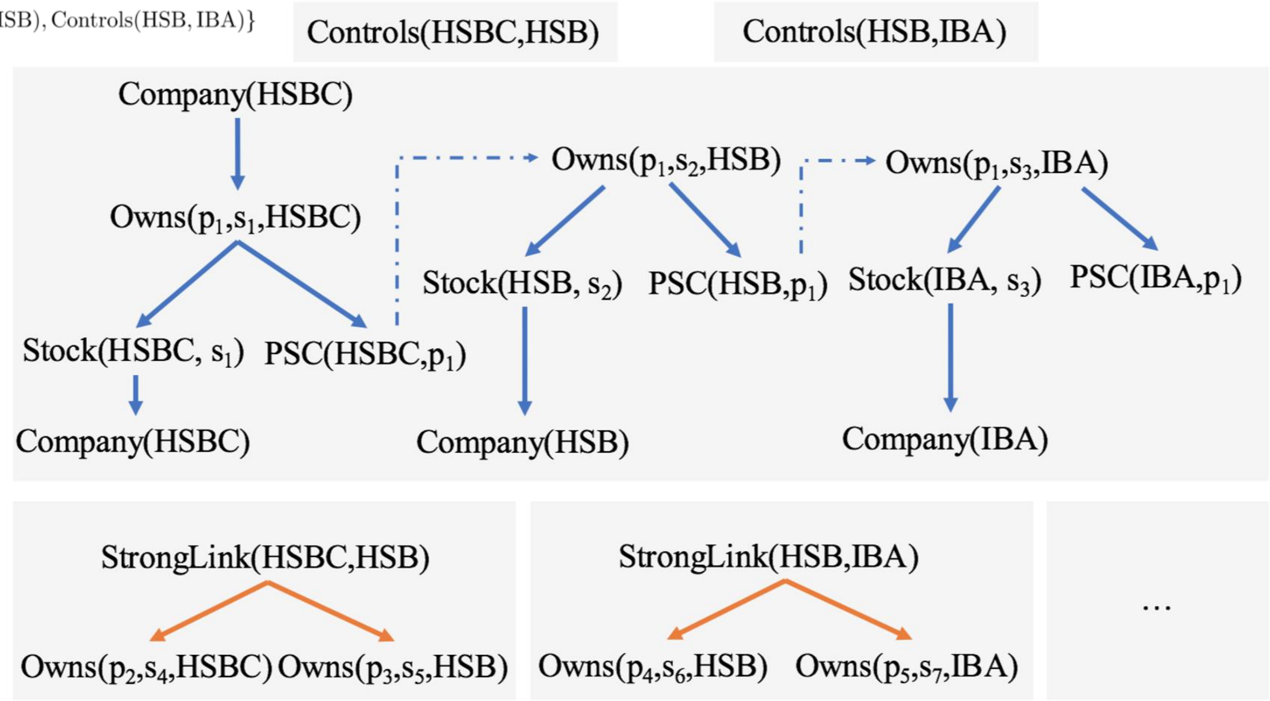
$$
\begin{aligned}
1 &: \mathrm{Company}(x) \to \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, x)\\
2 &: \mathrm{Owns}(\hat{p}, \hat{s}, x) \to \mathrm{Stock}(x, \hat{s})\\
3 &: \mathrm{Owns}(\hat{p}, \hat{s}, x) \to \mathrm{PSC}(x, \hat{p})\\
4 &: \mathrm{PSC}(x, \hat{p}), \mathrm{Controls}(x, y) \to \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, y)\\
5 &: \mathrm{PSC}(x, \hat{p}), \mathrm{PSC}(y, \hat{p}) \to \underline{\mathrm{StrongLink}}(x, y)\\
6 &: \mathrm{StrongLink}(x, y) \to \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, x)\\
7 &: \mathrm{StrongLink}(x, y) \to \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, y)\\
8 &: \mathrm{Stock}(x, \hat{s}) \to \mathrm{Company}(x).
\end{aligned}
$$

$$
\begin{aligned}
D = \{&\mathrm{Company(HSBC)}, \mathrm{Company(HSB)}, \mathrm{Company(IBA)},\\
&\mathrm{Controls(HSBC, HSB)}, \mathrm{Controls(HSB, IBA)}\}
\end{aligned}
$$



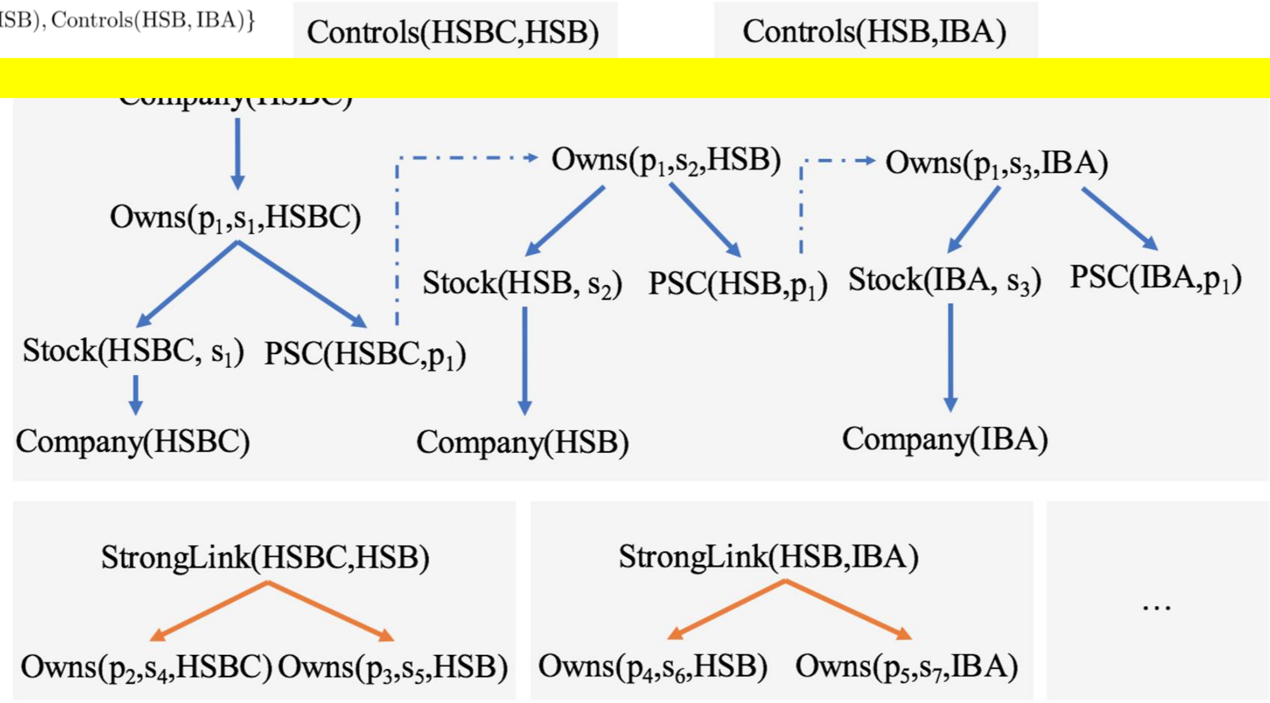
$$
\begin{aligned}
1 &: \mathrm{Company}(x) \rightarrow \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, x)\\
2 &: \mathrm{Owns}(\hat{p}, \hat{s}, x) \rightarrow \mathrm{Stock}(x, \hat{s})\\
3 &: \mathrm{Owns}(\hat{p}, \hat{s}, x) \rightarrow \mathrm{PSC}(x, \hat{p})\\
4 &: \mathrm{PSC}(x, \hat{p}), \mathrm{Controls}(x, y) \rightarrow \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, y)\\
5 &: \mathrm{PSC}(x, \hat{p}), \mathrm{PSC}(y, \hat{p}) \rightarrow \underline{\mathrm{StrongLink}}(x, y)\\
6 &: \mathrm{StrongLink}(x, y) \rightarrow \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, x)\\
7 &: \mathrm{StrongLink}(x, y) \rightarrow \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, y)\\
8 &: \mathrm{Stock}(x, \hat{s}) \rightarrow \mathrm{Company}(x).
\end{aligned}
$$

$$
\begin{aligned}
D = \{&\mathrm{Company}(\mathrm{HSBC}), \mathrm{Company}(\mathrm{HSB}), \mathrm{Company}(\mathrm{IBA}),\\
&\mathrm{Controls}(\mathrm{HSBC}, \mathrm{HSB}), \mathrm{Controls}(\mathrm{HSB}, \mathrm{IBA})\}
\end{aligned}
$$



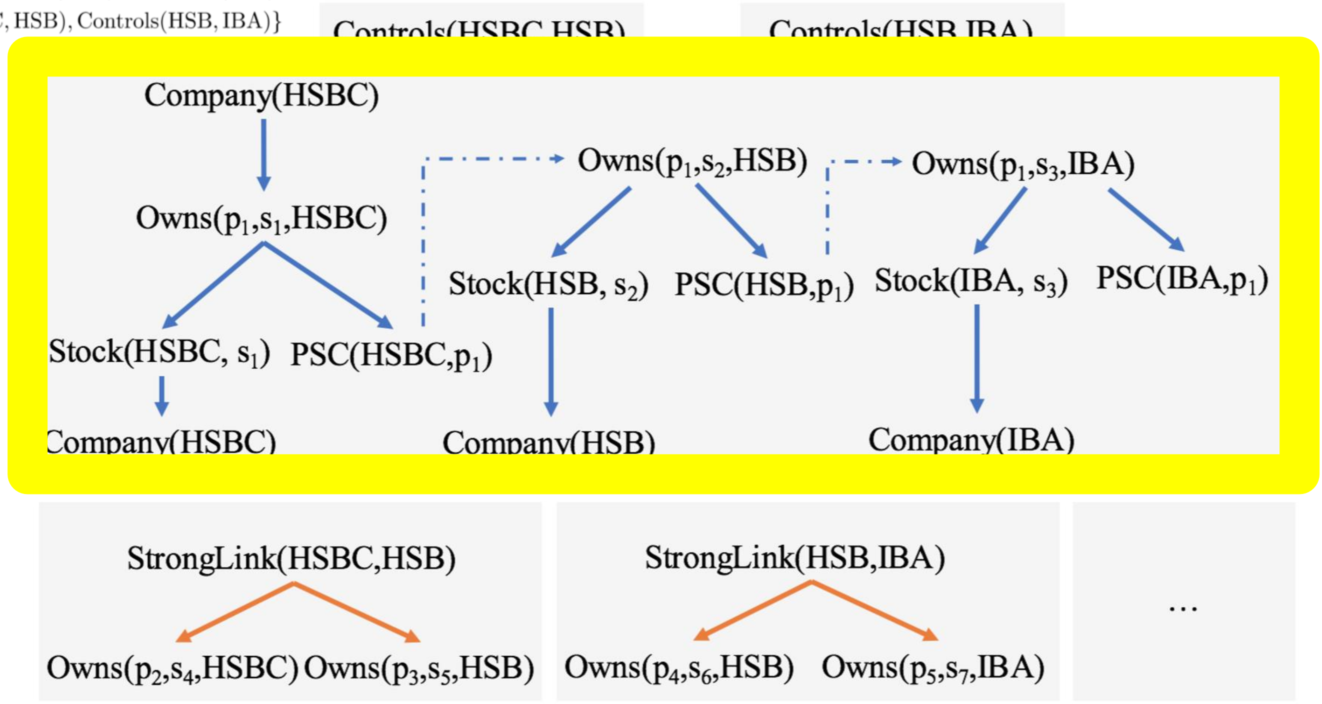
$$
\begin{aligned}
1 &: \mathrm{Company}(x) \rightarrow \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, x) \\
2 &: \mathrm{Owns}(\hat{p}, \hat{s}, x) \rightarrow \mathrm{Stock}(x, \hat{s}) \\
3 &: \mathrm{Owns}(\hat{p}, \hat{s}, x) \rightarrow \mathrm{PSC}(x, \hat{p}) \\
4 &: \mathrm{PSC}(x, \hat{p}), \mathrm{Controls}(x, y) \rightarrow \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, y) \\
5 &: \mathrm{PSC}(x, \hat{p}), \mathrm{PSC}(y, \hat{p}) \rightarrow \underline{\mathrm{StrongLink}}(x, y) \\
6 &: \mathrm{StrongLink}(x, y) \rightarrow \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, x) \\
7 &: \mathrm{StrongLink}(x, y) \rightarrow \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, y) \\
8 &: \mathrm{Stock}(x, \hat{s}) \rightarrow \mathrm{Company}(x).
\end{aligned}
$$

$$
= \{\mathrm{Company}(\mathrm{HSBC}), \mathrm{Company}(\mathrm{HSB}), \mathrm{Company}(\mathrm{IBA}), \mathrm{Controls}(\mathrm{HSBC}, \mathrm{HSB}), \mathrm{Controls}(\mathrm{HSB}, \mathrm{IBA})\}
$$



$$
\begin{aligned}
1 &: \text{Company}(x) \to \exists p \exists s \; \text{Owns}(\hat{p}, \hat{s}, x) \\
2 &: \text{Owns}(\hat{p}, \hat{s}, x) \to \text{Stock}(x, \hat{s}) \\
3 &: \text{Owns}(\hat{p}, \hat{s}, x) \to \text{PSC}(x, \hat{p}) \\
4 &: \text{PSC}(x, \hat{p}), \text{Controls}(x, y) \to \exists s \; \text{Owns}(\hat{p}, \hat{s}, y) \\
5 &: \text{PSC}(x, \hat{p}), \text{PSC}(y, \hat{p}) \to \underline{\text{StrongLink}}(x, y) \\
6 &: \text{StrongLink}(x, y) \to \exists p \exists s \; \text{Owns}(\hat{p}, \hat{s}, x) \\
7 &: \text{StrongLink}(x, y) \to \exists p \exists s \; \text{Owns}(\hat{p}, \hat{s}, y) \\
8 &: \text{Stock}(x, \hat{s}) \to \text{Company}(x).
\end{aligned}
$$

$$
\begin{aligned}
D = \{&\text{Company}(\text{HSBC}), \text{Company}(\text{HSB}), \text{Company}(\text{IBA}), \\
&\text{Controls}(\text{HSBC}, \text{HSB}), \text{Controls}(\text{HSB}, \text{IBA})\}
\end{aligned}
$$



Dangerous

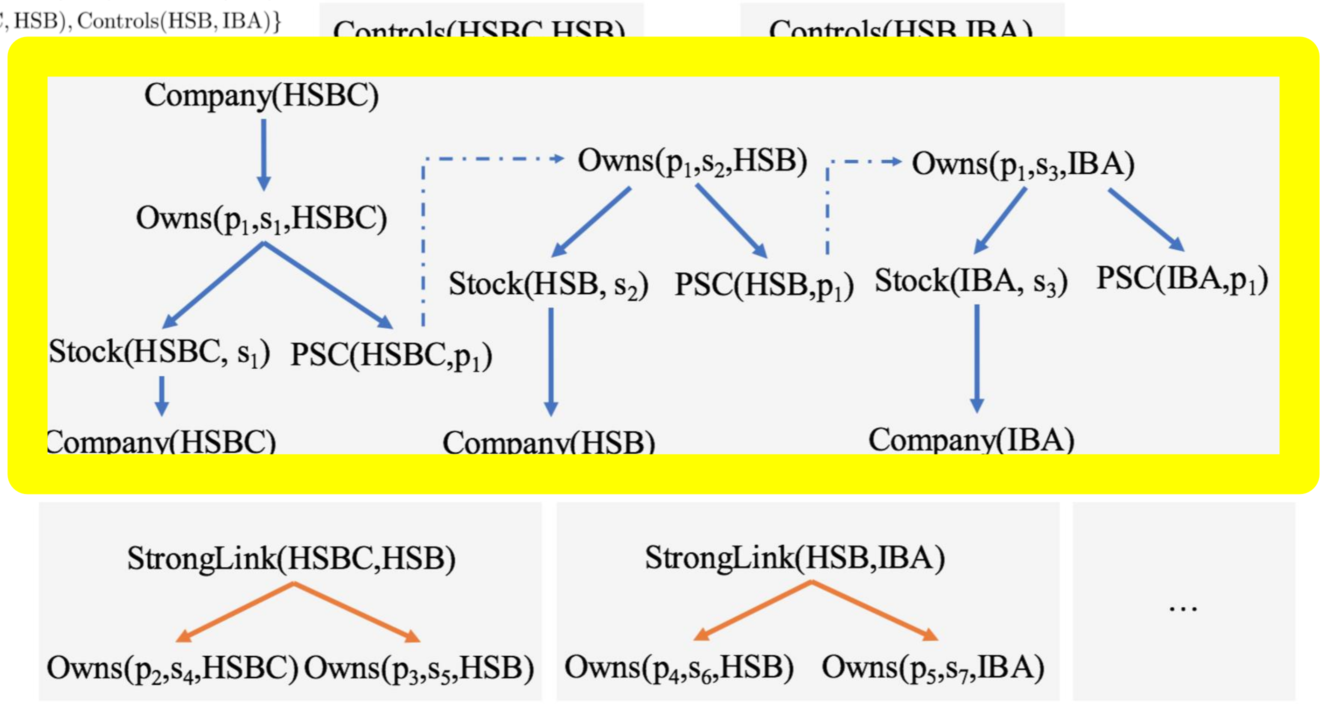
$$\mathrm{Emp}(x) \to \exists \mathbf{z}\, \mathrm{Mgr}(x, \mathbf{z}) \qquad \mathrm{Mgr}(x, \mathbf{y}) \land \mathrm{Pers}(x) \to \mathrm{Emp}(\mathbf{y})$$

Ward

1. all the "**dangerous**" variables should coexist in a single body-atom α, called the **ward**

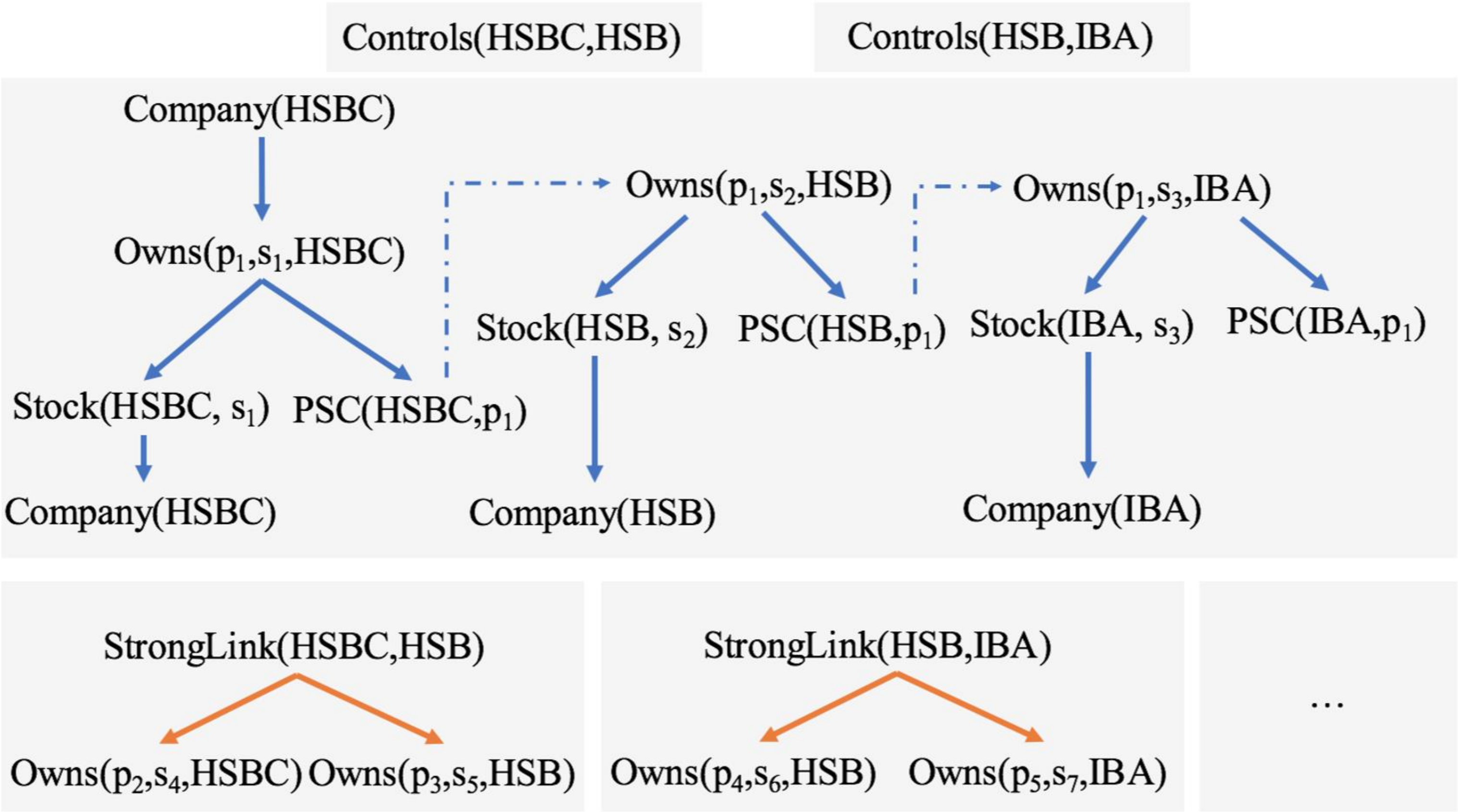
$$
\begin{aligned}
1 &: \mathrm{Company}(x) \rightarrow \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, x)\\
2 &: \mathrm{Owns}(\hat{p}, \hat{s}, x) \rightarrow \mathrm{Stock}(x, \hat{s})\\
3 &: \mathrm{Owns}(\hat{p}, \hat{s}, x) \rightarrow \mathrm{PSC}(x, \hat{p})\\
4 &: \mathrm{PSC}(x, \hat{p}), \mathrm{Controls}(x, y) \rightarrow \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, y)\\
5 &: \mathrm{PSC}(x, \hat{p}), \mathrm{PSC}(y, \hat{p}) \rightarrow \underline{\mathrm{StrongLink}}(x, y)\\
6 &: \mathrm{StrongLink}(x, y) \rightarrow \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, x)\\
7 &: \mathrm{StrongLink}(x, y) \rightarrow \exists p \exists s\ \mathrm{Owns}(\hat{p}, \hat{s}, y)\\
8 &: \mathrm{Stock}(x, \hat{s}) \rightarrow \mathrm{Company}(x).
\end{aligned}
$$

$$
\begin{aligned}
D = \{&\mathrm{Company}(\mathrm{HSBC}), \mathrm{Company}(\mathrm{HSB}), \mathrm{Company}(\mathrm{IBA}),\\
&\mathrm{Controls}(\mathrm{HSBC}, \mathrm{HSB}), \mathrm{Controls}(\mathrm{HSB}, \mathrm{IBA})\}
\end{aligned}
$$



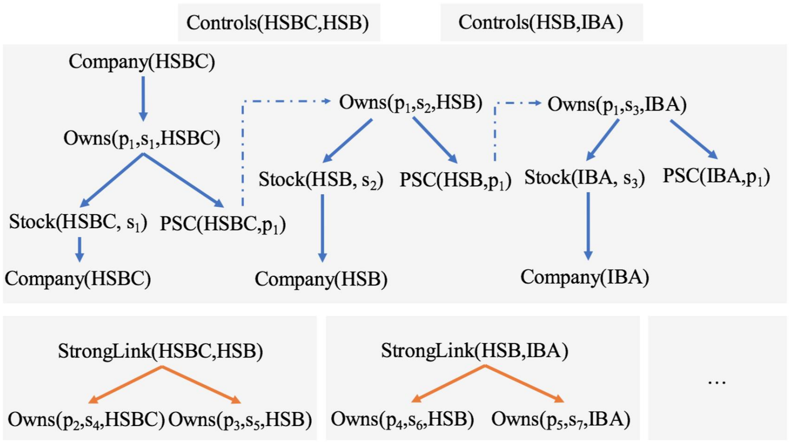
Theorem

Let a and b be two facts in a warded forest. If they are isomorphic, then $subtree(a)$ is isomorphic to $subtree(b)$.



Theorem

Let a and b be two facts in a warded forest. If they are isomorphic, then $subtree(a)$ is isomorphic to $subtree(b)$.

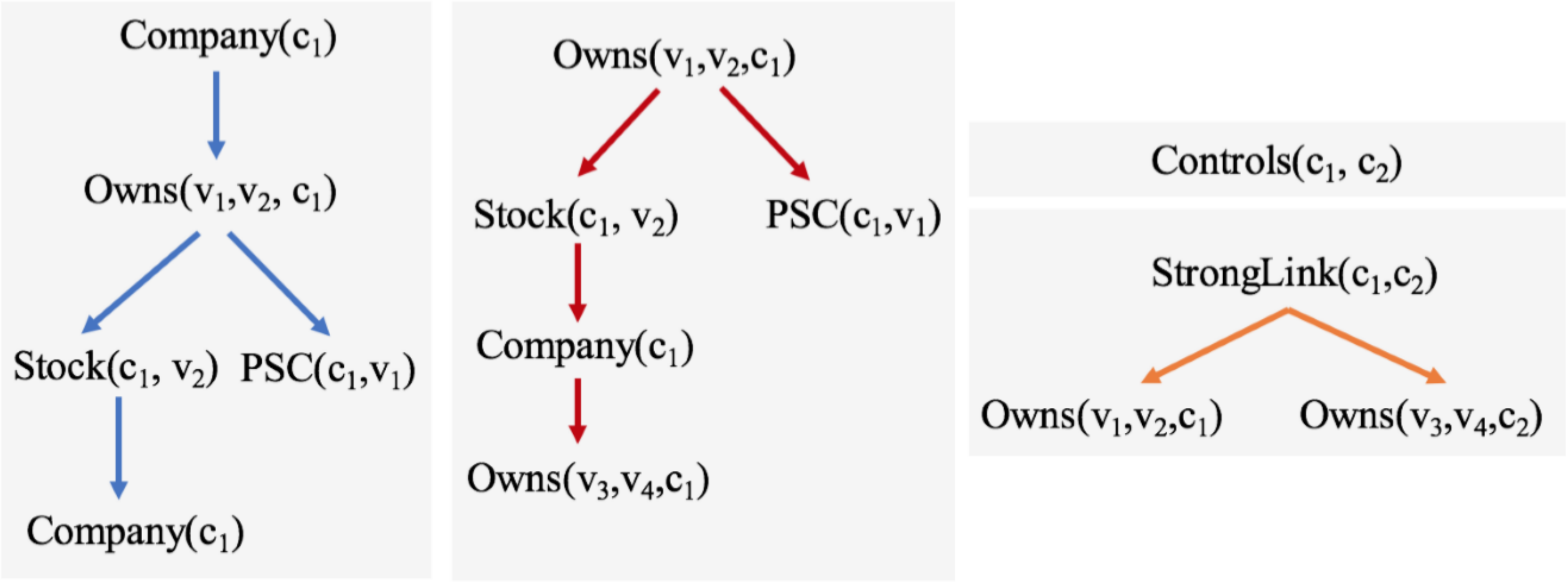


Theorem

Let a and b be two facts in the chase graph of a set of harmless warded rules. If a and b are isomorphic, then $subgraph(a)$ is isomorphic to $subgraph(b)$.

Proposition

Let a and b be two facts in a linear forest. If they are pattern-isomorphic, then $subtree(a)$ is pattern-isomorphic $to\ subtree(b)$.



Algorithm 1 Termination strategy for the chase step.

```
function CHECK_TERMINATION(a)
    if a.generating_rule == {LINEAR or WARDED} then
        if ∃λ ∈ S[π(a.l_root)] s.t. λ ⊆ a.provenance then
            return false                                  ▷ beyond a stop provenance
        else if ∃λ ∈ S[π(a.l_root)] s.t. a.provenance ⊂ λ then
            return true                                   ▷ within a stop provenance
        else                                              ▷ continue exploration
            if ∃g in G[a.w_root] s.t. a isomorphic to g then
                S[π(a.l_root)]=a.provenance
                return false                              ▷ isomorphism found
            else
                G[a.w_root].append(a)
                return true                               ▷ isomorphism not found
    else if a ∉ G then                                    ▷ other non-linear generating rules
        G[a.w_root].append(a)                             ▷ and reset provenance
        return true
    else                                                  ▷ the new tree is redundant
        return false
```

Algorithm 2 A generic chase using the termination strategy.

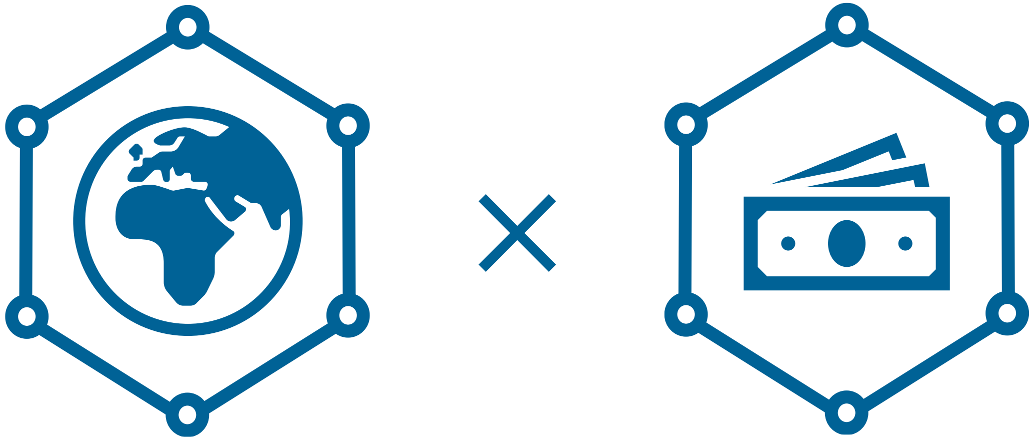
```
function CHASE(D, Σ)
    for all σ ∈ Σ and x to which σ applies do
        if CHECK_TERMINATION(σ(x)) then
            D = D ∪ {σ(x)}
```

Knowledge Graph Management Systems

Vadalog

Advanced

Emanuel Sallinger



Financial Knowledge Graphs

Company Knowledge Graphs

Emanuel Sallinger

BANCA D'ITALIA

Ongoing work on the reference historical KG for all Italian companies, including:

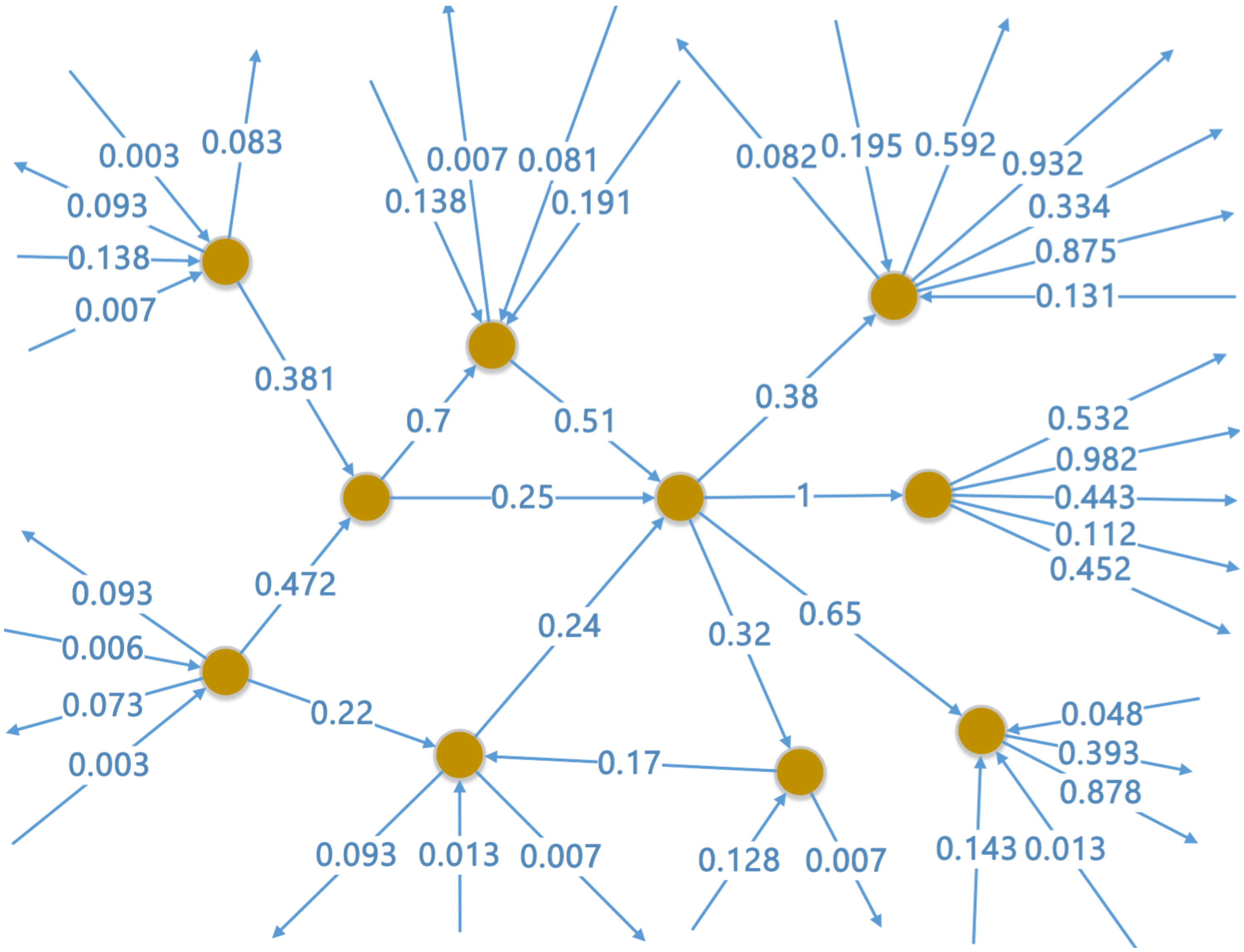
- 6 million **companies**
- 10 million **people**
- 30 million **ownerships**
- 20 million **roles** (e.g. managers)
- 200k company **events** (like M&A)

- 50+ million **nodes**
- 100+ million **edges**
- 1+ billion **properties**

plus, adding **derived** extensional knowledge:

- company **control**
- **family links** between people

Ownership Graph



Creditworthiness

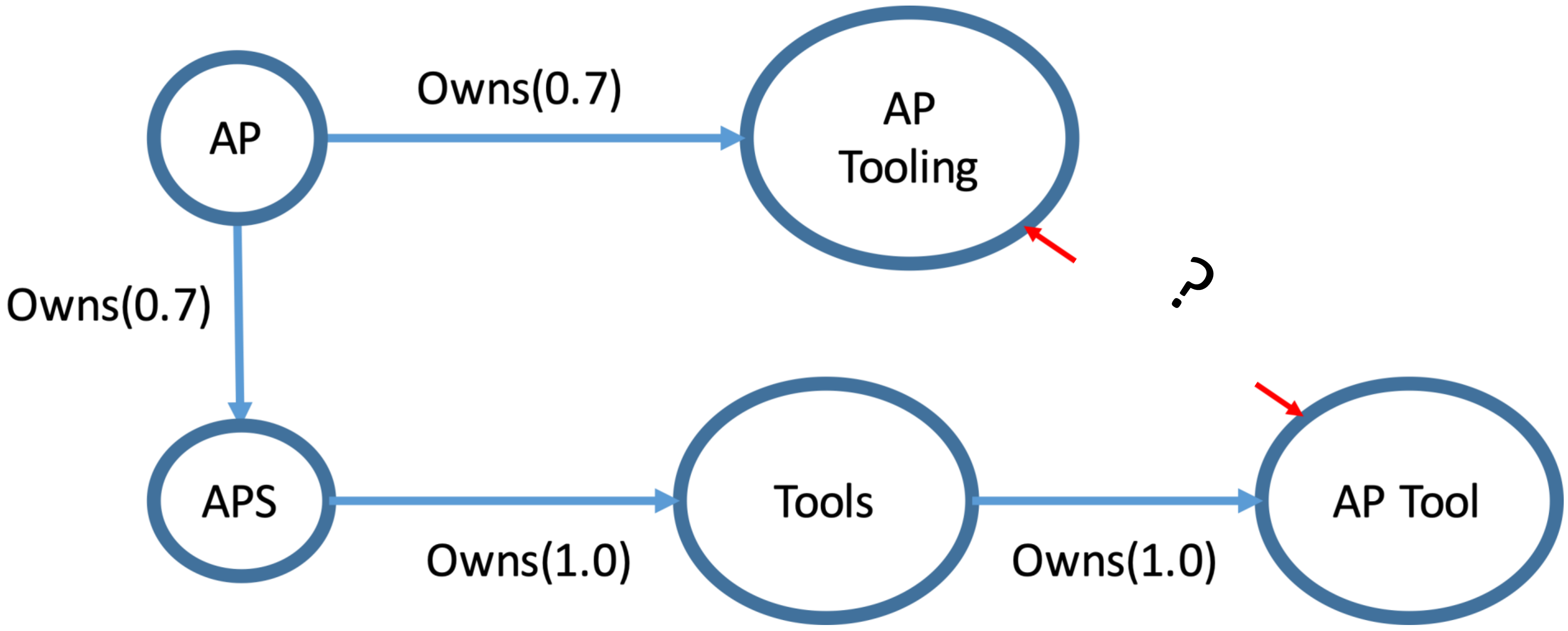
- Probability that a borrower company will not default on its debt obligations

Credit DB				
Company name	Total Debts	…	Avg Debt to Income Ratio	Rating
AP Tool	0	…	0.28	A
ACME	50k	…	0.46	B
Big S	20k	…	0.89	D
AP Tooling	0	…	0.10	A

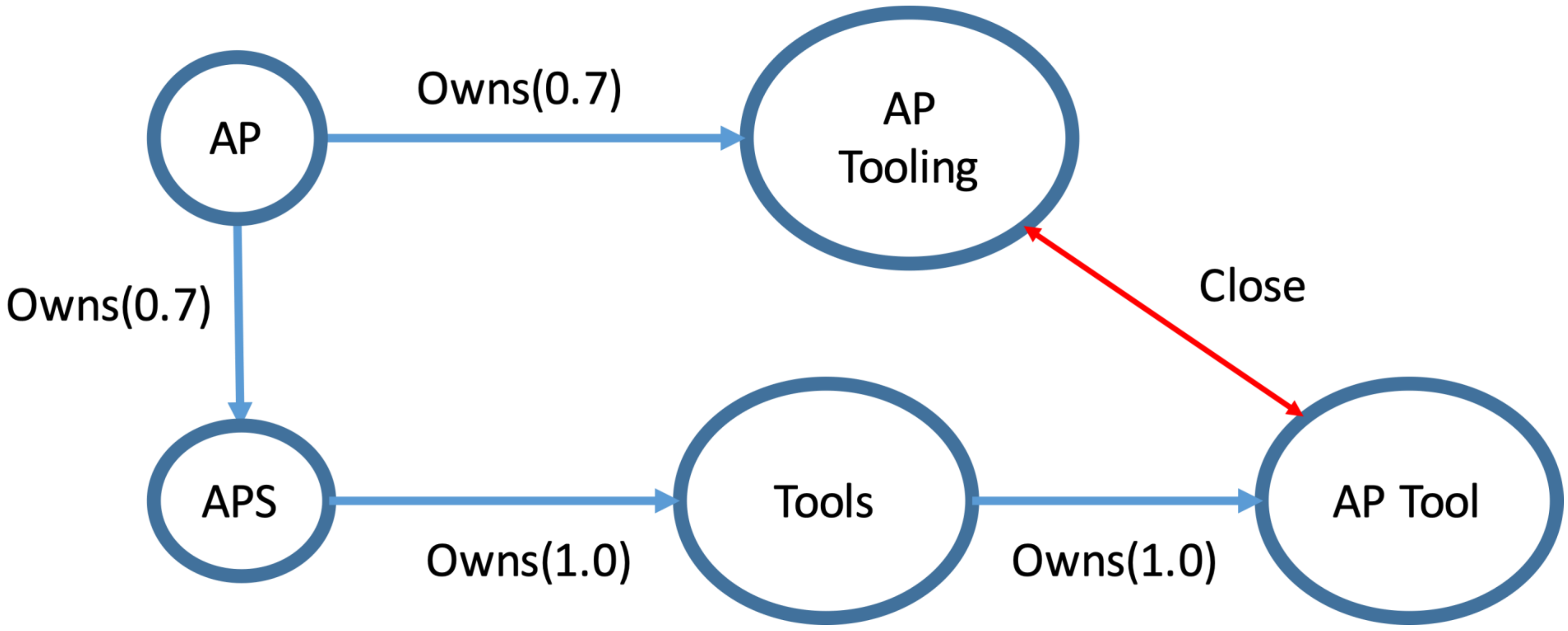
LOAN DENIED

AP Tool asks for 1M loan, with AP Tooling as guarantor

Company Graphs



Company Graphs



Company Graphs

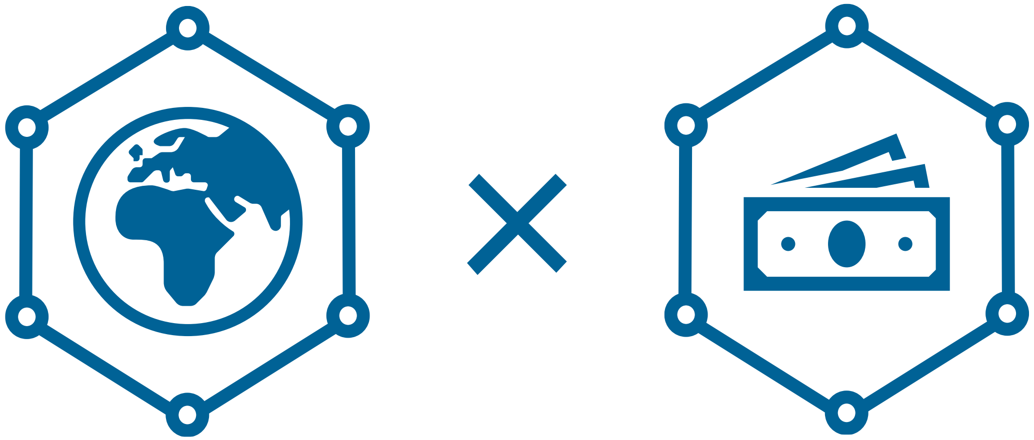
We build graphs of company networks, to:

1. reveal **power**
   1. finding **controllers**
   2. studying the **structure** of market
   3. studying **dispersion** of control
   4. global **shareholding** analysis
   5. intercept and prevent **hostile takeovers**

2. detect **collusion** and do **forensics**
   1. support **AML**
   2. detecting **ultimate beneficial owners**

3. evaluate and guarantee **anonymity** of **microdata**

4. evaluate **risks**
5. model **propagations** (e.g., of shocks)
6. guarantee **compliance**
7. perform enhanced **due diligence**
8. understand **foreign shareholder structures**
9. know real **cash flows**



Financial Knowledge Graphs

Company Knowledge Graphs

Emanuel Sallinger

Overview

Temporal Knowledge Graphs

Bonus

Emanuel Sallinger

DatalogMTL

Temporal Operators

flags(x,z) :- monitors(x,z), $\boxminus_{[0,5]} \diamondminus_{[0,3]}$ signal(z).

Intervals

Hidden Time Information

monitors(“Server1“,“SSH“)@[0,∞).
signal(“SSH“)@[2,2].
signal(“SSH“)@4.

DatalogMTL

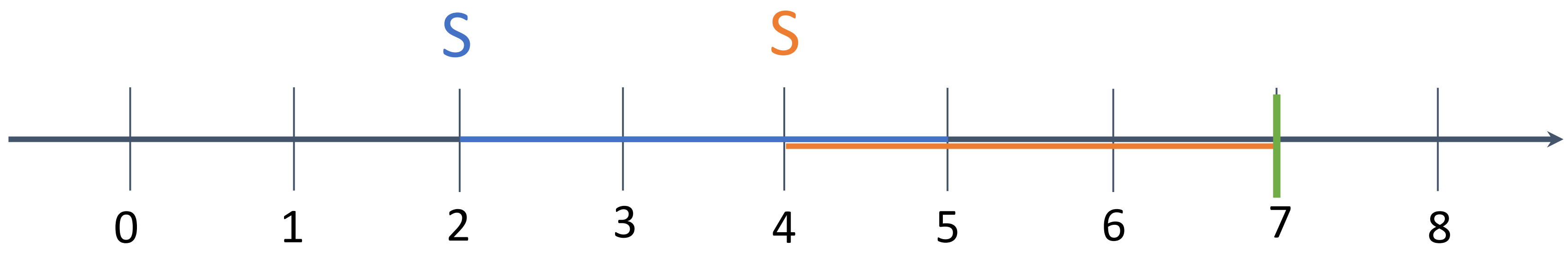
Temporal Operators

$$\text{flags}(x,z) \text{ :- } \text{monitors}(x,z),\ \boxminus_{[0,5]} \diamondminus_{[0,3]}\ \text{signal}(z).$$

Intervals

Hidden Time Information

monitors(“Server1“,“SSH“)@[0,∞).
signal(“SSH“)@[2,2].
signal(“SSH“)@4.



DatalogMTL – Intervals ϱ

ϱ of form $<t1, t2>$, where t1 in $\mathbb{Q}_2^{\geq 0}$, t2 in$\mathbb{Q}_2^{\geq 0} \cup \{\infty\}$

- [t1,t2], only if $t2 \neq \infty$
- [t1,t2)
- (t1,t2], only if $t2 \neq \infty$
- (t2,t2)

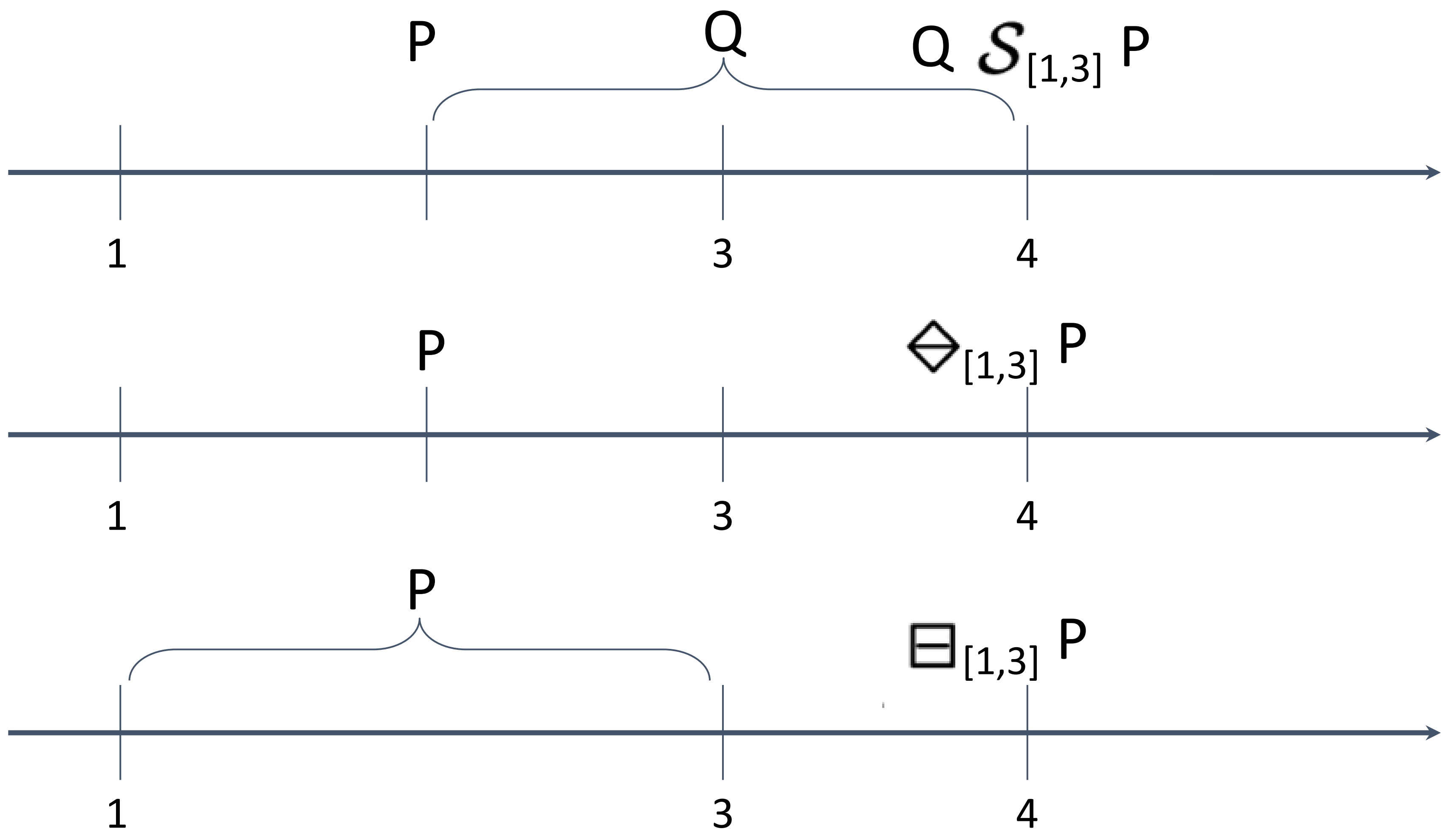
DatalogMTL – Rules

$$B \leftarrow A_1, \ldots, A_n$$

$$A := \top \mid \bot \mid P(\tau) \mid \boxplus_\varrho A \mid \boxminus_\varrho A \mid \diamondplus_\varrho A \mid \diamondminus_\varrho A \mid A\mathcal{U}_\varrho A \mid A\mathcal{S}_\varrho A$$

$$B := \top \mid \bot \mid P(\tau) \mid \boxplus_\varrho A \mid \boxminus_\varrho A$$

DatalogMTL – Temporal Operators



Consistency Checking - Data Complexity

<table>
<tr><th rowspan="2"></th><th colspan="2">Rational Timeline</th><th colspan="2">Integer Timeline</th></tr>
<tr><th>First-order</th><th>Prop.</th><th>First-order</th><th>Prop.</th></tr>
<tr><td>DatalogMTL</td><td colspan="2" rowspan="3">PSpace-c</td><td rowspan="2">PSpace-c</td><td rowspan="6">NC^1-c</td></tr>
<tr><td>$\mathrm{DatalogMTL}_{\mathrm{lin}}$</td></tr>
<tr><td>$\mathrm{DatalogMTL}_{\mathrm{core}}$</td><td>NL-h</td></tr>
<tr><td>$\mathrm{DatalogMTL}^{\boxminus}_{\mathrm{lin}}$</td><td></td><td rowspan="2">P-hard</td><td>PSpace-c</td></tr>
<tr><td>$\mathrm{DatalogMTL}^{\boxminus}_{\mathrm{core}}$</td><td></td><td>$\mathrm{NC}^1$-h</td></tr>
<tr><td>$\mathrm{DatalogMTL}^{\diamondminus}_{\mathrm{lin}}$</td><td colspan="2">NL-c</td><td>NL-c</td></tr>
<tr><td>$\mathrm{DatalogMTL}^{\diamondminus}_{\mathrm{core}}$</td><td colspan="2">$\mathrm{TC}^0$-c</td><td colspan="2">in $\mathrm{AC}^0[k]$ for $k \in \mathbb{N}$, not in AC^0</td></tr>
</table>

Core: B :- A or $\bot$:- $A_1 \land A_2$

Linear: At most one A_i is IDB or $\bot$:- $A_1 \land A_2$

Logical Reasoning

Temporal Knowledge Graphs

Bonus

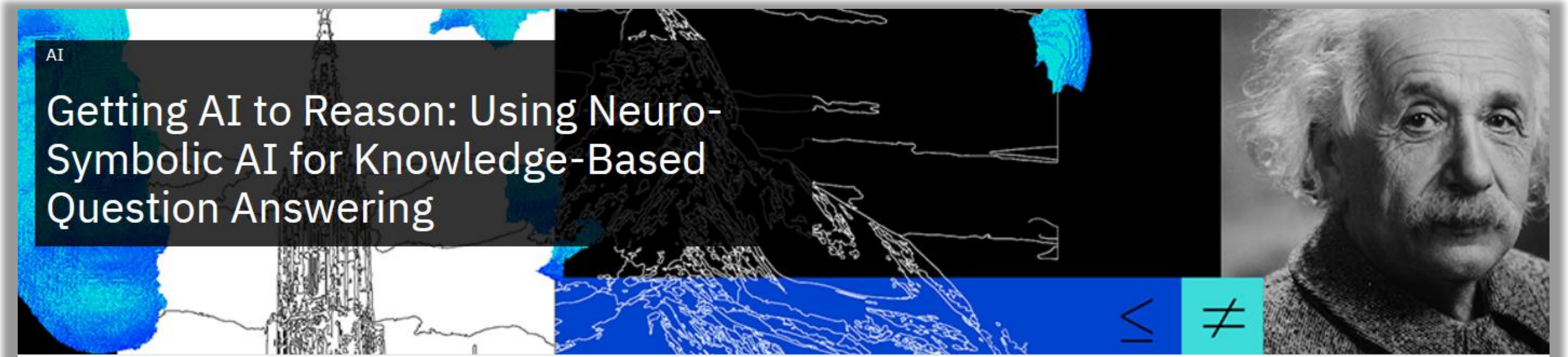
Emanuel Sallinger

Grand Challenges

Neuro-symbolic KGs

Teaser

Emanuel Sallinger



December 4, 2020 | Written by: Salim Roukos, Alex Gray, and Pavan Kapanipathi

Categorized: AI | Neuro-symbolic AI | Neuro-symbolic AI

Share this post:

Language is what makes us human. Asking questions is how we learn.

Building on the foundations of deep learning and symbolic AI, we have developed technology that can answer complex questions with minimal domain-specific training. Initial results are very encouraging – the system outperforms current state-of-the-art techniques on two prominent datasets with no need for specialized end-to-end training.

*Read more on IBM Research's efforts in neuro-symbolic 'common sense' AI **here***

Source: https://www.ibm.com/blogs/research/2020/12/ai-neurosymbolic-common-sense/

SAMSUNG

Samsung Research

ABOUT US | RESEARCH INSIGHTS | GLOBAL R&D NETWORKS | NEWS&EVENTS | CAREERS | CULTURE

Samsung Research

"When deep learning meets logic": a three days virtual workshop on neural-symbolic integration sponsored by Samsung Research

on January 15, 2021

The effort to integrate logic with deep learning has intensified in recent years and has the potential to give rise to a new computational paradigm in which symbolic knowledge is used to assist deep learning systems or extend their capabilities, while offering, at the same time, a path towards the grounding of symbols and the induction of knowledge from low-level sensory data.

The synergy of these two different worlds is the topic of the workshop *"When deep learning meets logic",* which will take place online from February 15 to 17th, 2021. This workshop aims at the following: (i) present the applications that are enabled by this computational paradigm; (ii) explore the state-of-the-art and understand its level of maturity and adoption by the industrial sector; in particular, techniques and theory developed from both the deep learning and the logic communities will be presented; and (iii) identify some of the big questions that are open in this area and single out problems that require further investigation.

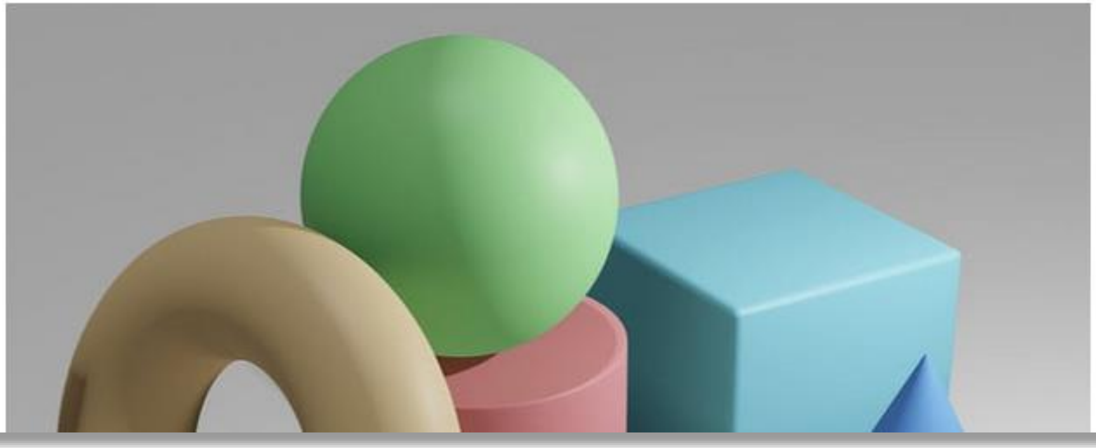
Source: https://research.samsung.com/news/-When-deep-learning-meets-logic-a-three-days-virtual-workshop-on-neural-symbolic-integration-sponsored-by-Samsung-Research

EMERGING TECH

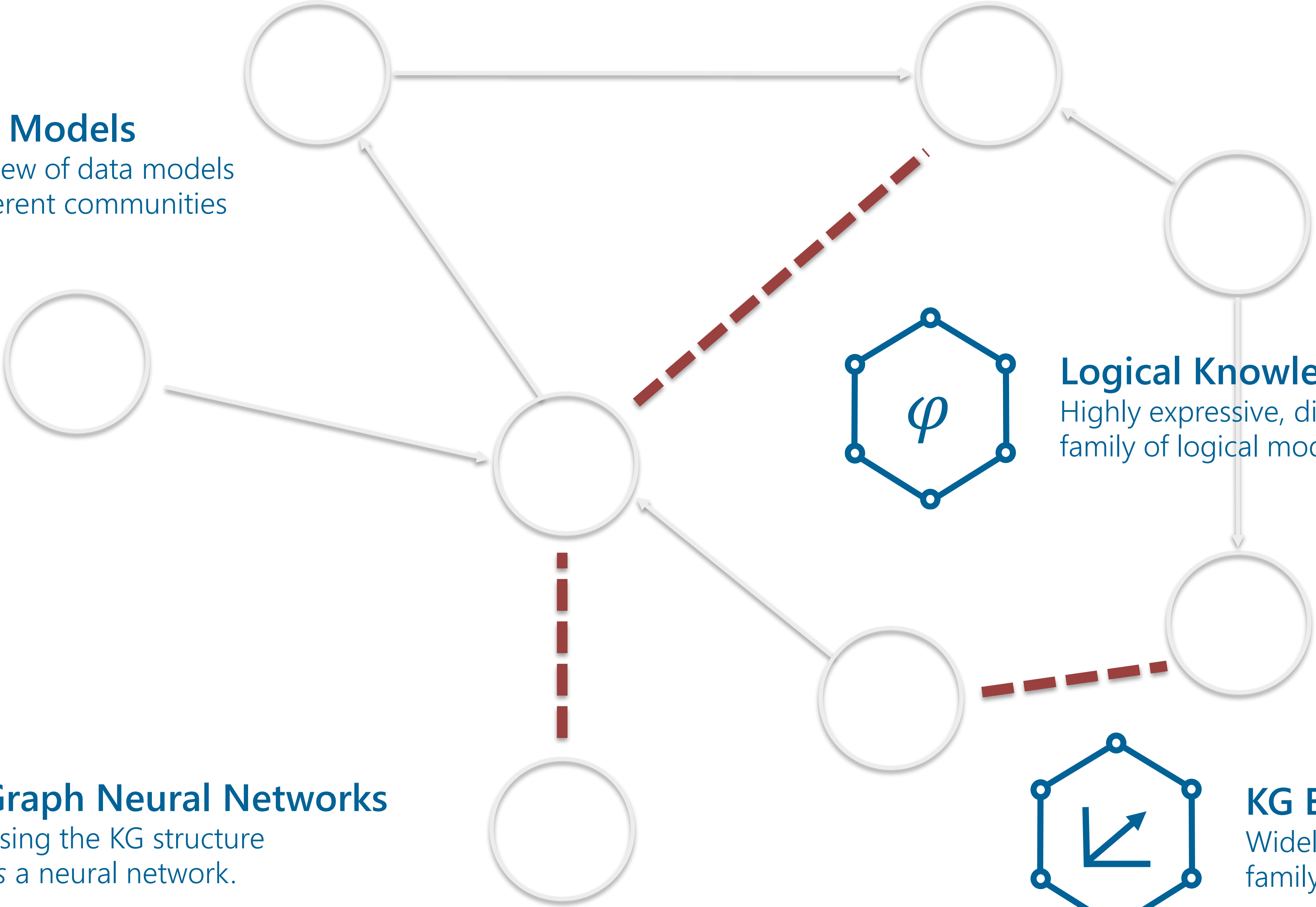
Neuro-symbolic A.I. is the future of artificial intelligence. Here's how it works

By Luke Dormehl
January 5, 2020

Picture a tray. On the tray is an assortment of shapes: Some cubes, others spheres. The shapes are made from a variety of different materials and represent an assortment of sizes. In total there are, perhaps, eight objects. My question: "Looking at the objects, are there an equal number of large things and metal spheres?"



Source: https://www.digitaltrends.com/cool-tech/neuro-symbolic-ai-the-future/



Data Models
Overview of data models
in different communities

Logical Knowledge in KGs
Highly expressive, diverse
family of logical models.

Graph Neural Networks
Using the KG structure
as a neural network.

KG Embeddings
Widely-applied, large
family of ML models

Representations

Graph

Knowledge Graph Data Models

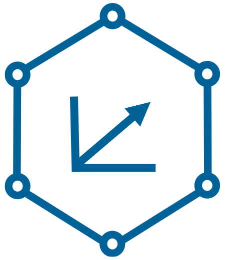
Knowledge

Knowledge Graph Embeddings

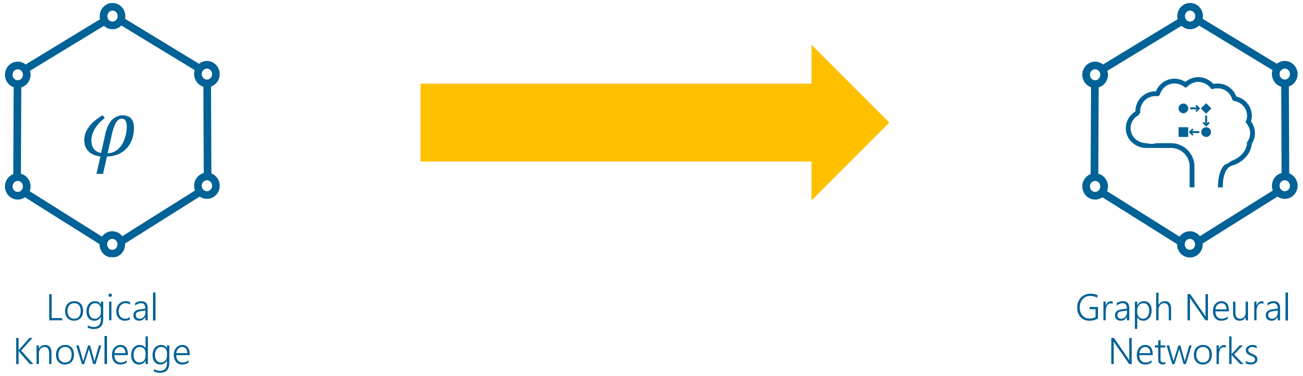
Logical Knowledge

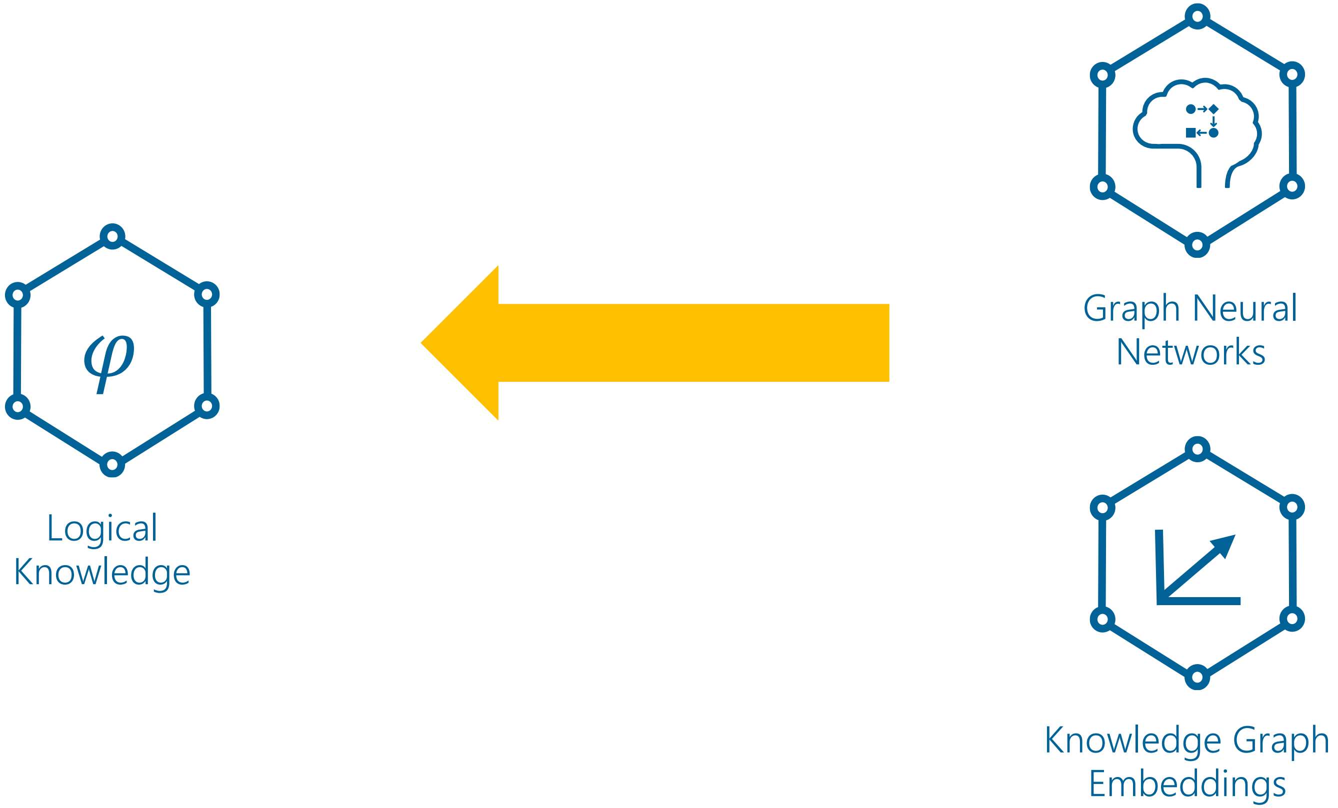
Graph Neural Networks

Logical Knowledge



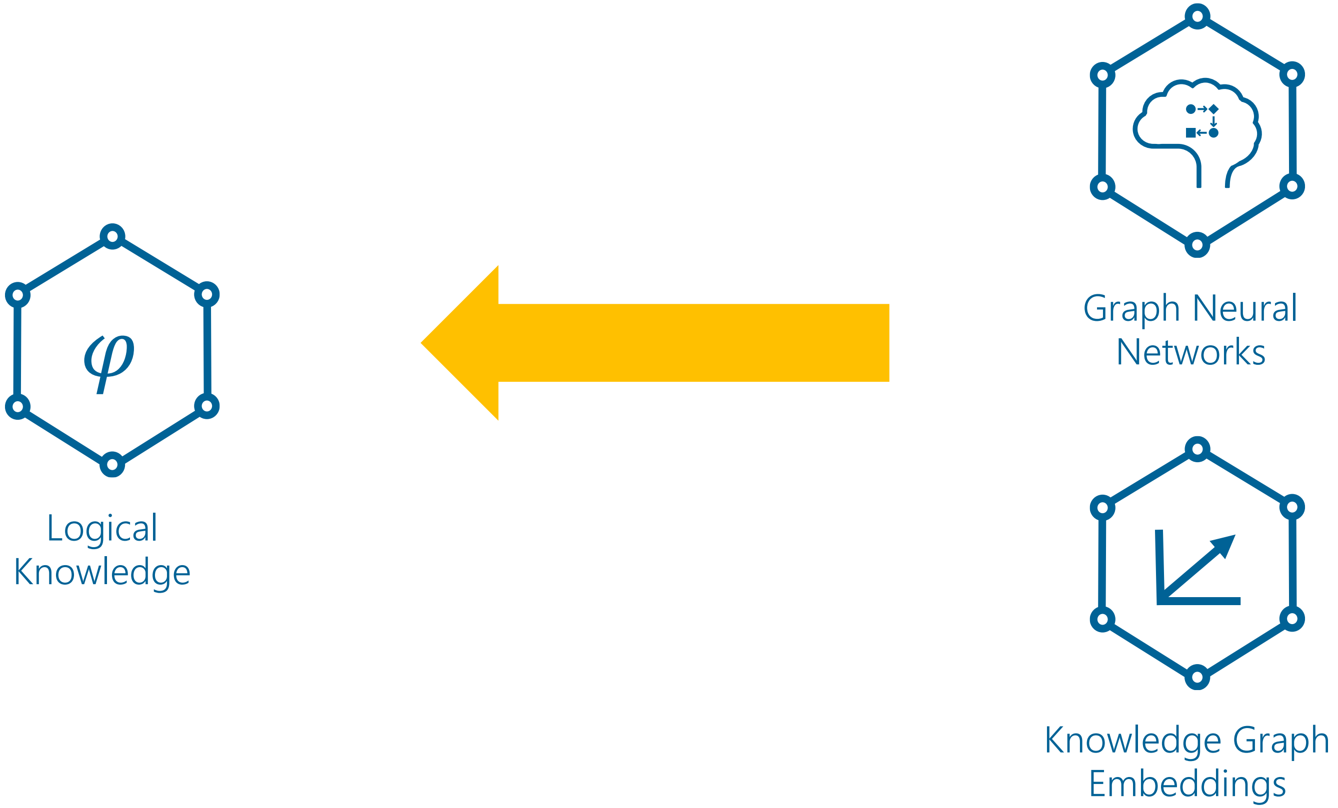
Knowledge Graph Embeddings


Logical Knowledge
Graph Neural Networks



Logical Knowledge
Graph Neural Networks
Knowledge Graph Embeddings

"Explainable AI"



Grand Challenges

Neuro-symbolic KGs

Teaser

Emanuel Sallinger