



# The LDBC Benchmark Suite

Gábor Szárnyas  
(CWI Amsterdam, LDBC)

FOSDEM 2023 HPC room

# LDBC: Linked Data Benchmark Council

Non-profit company founded in 2012

Designs graph benchmarks & governs their use

Research on graph schemas and query languages

Open-source under Apache v2



[github.com/ldbc](https://github.com/ldbc)

## Sponsor Members



## Regular Members



# **LDDBC Social Network Benchmark**

**Database systems**

Data set

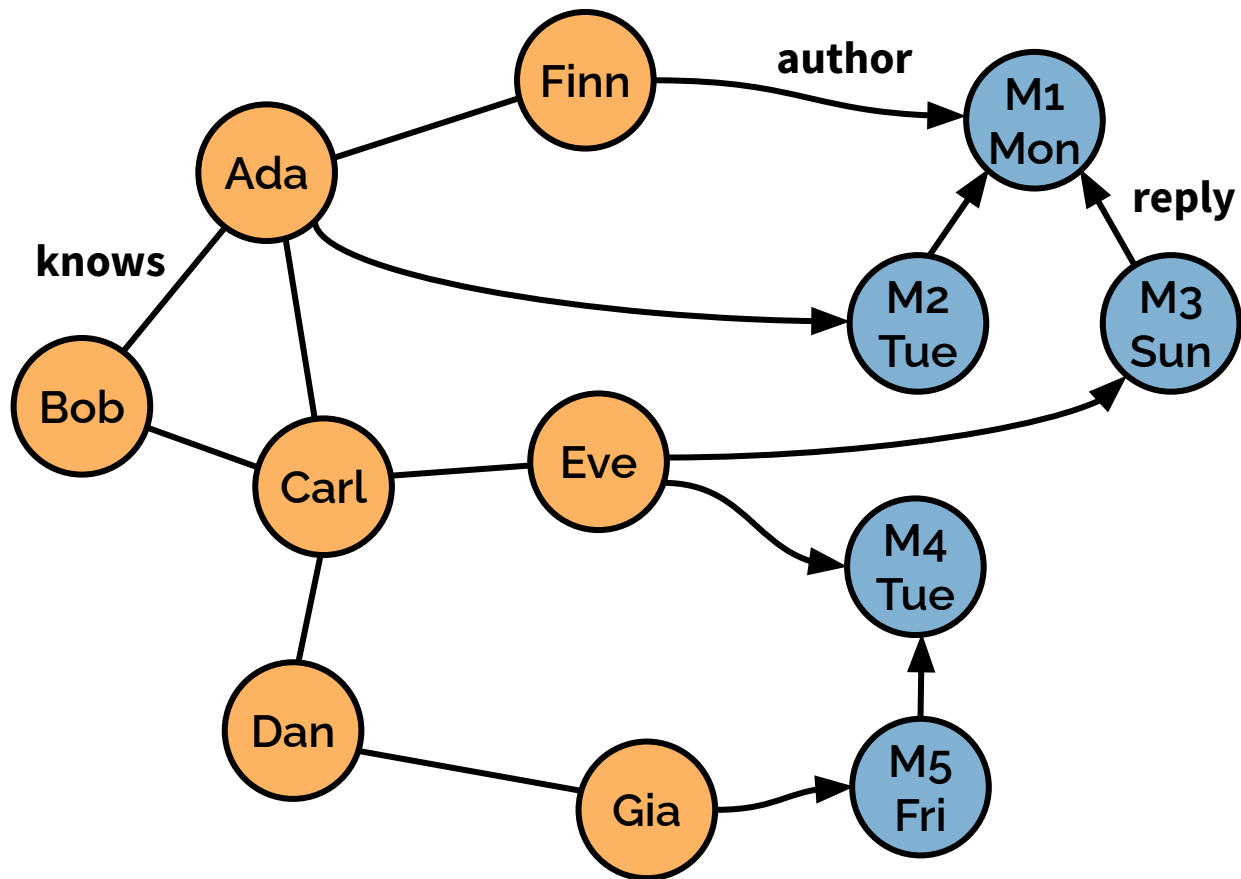
Queries

Updates

Data set

Queries

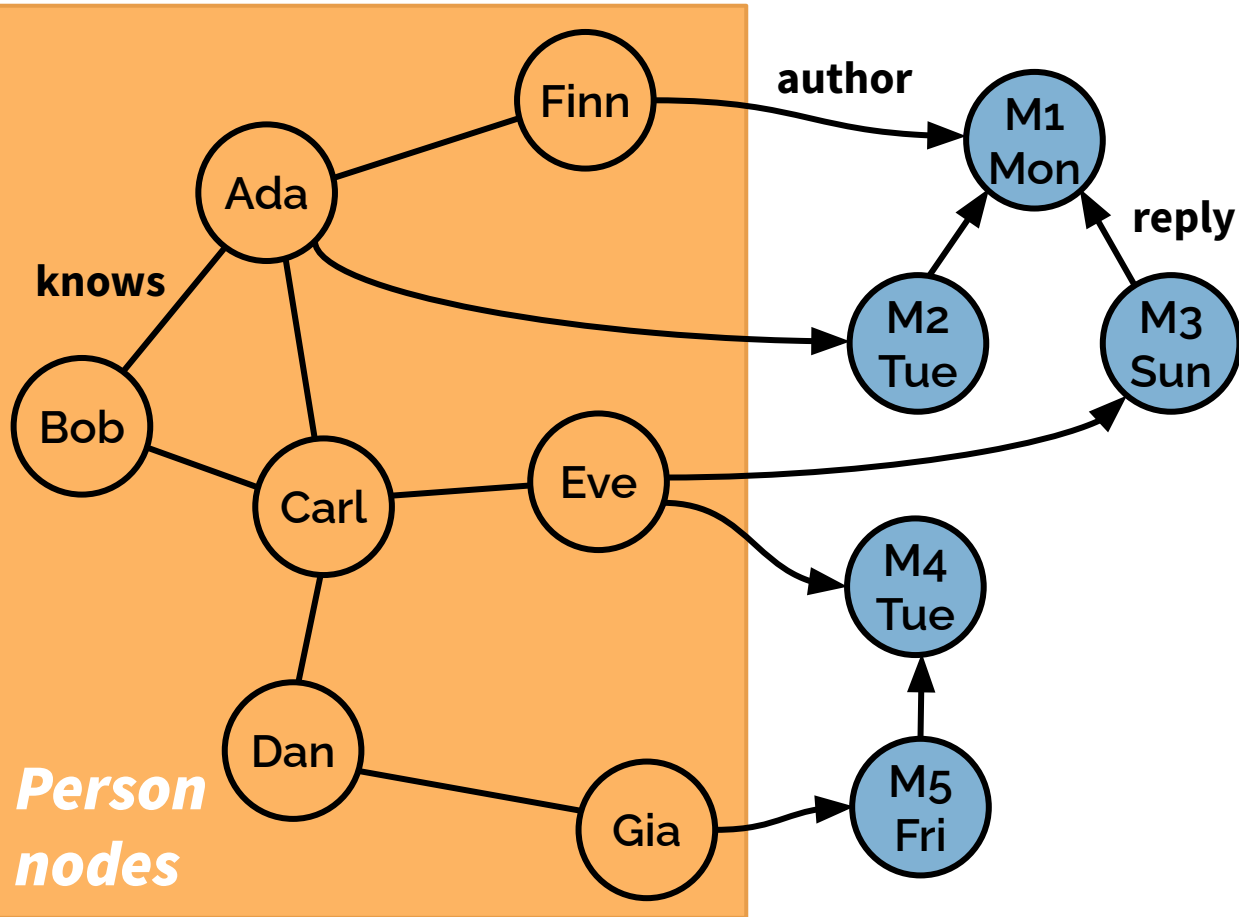
Updates



Data set

Queries

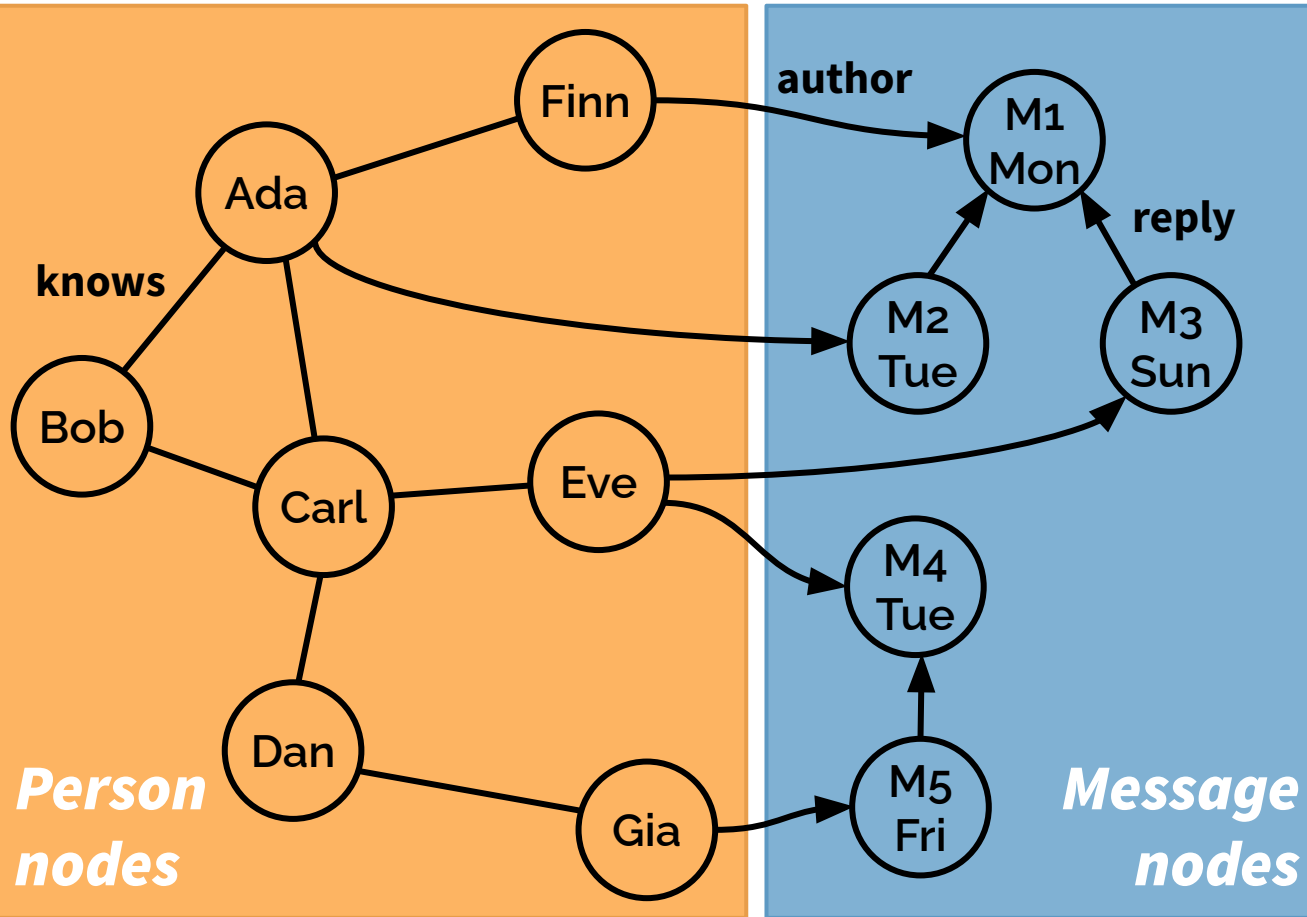
Updates



Data set

Queries

Updates



*Person nodes*

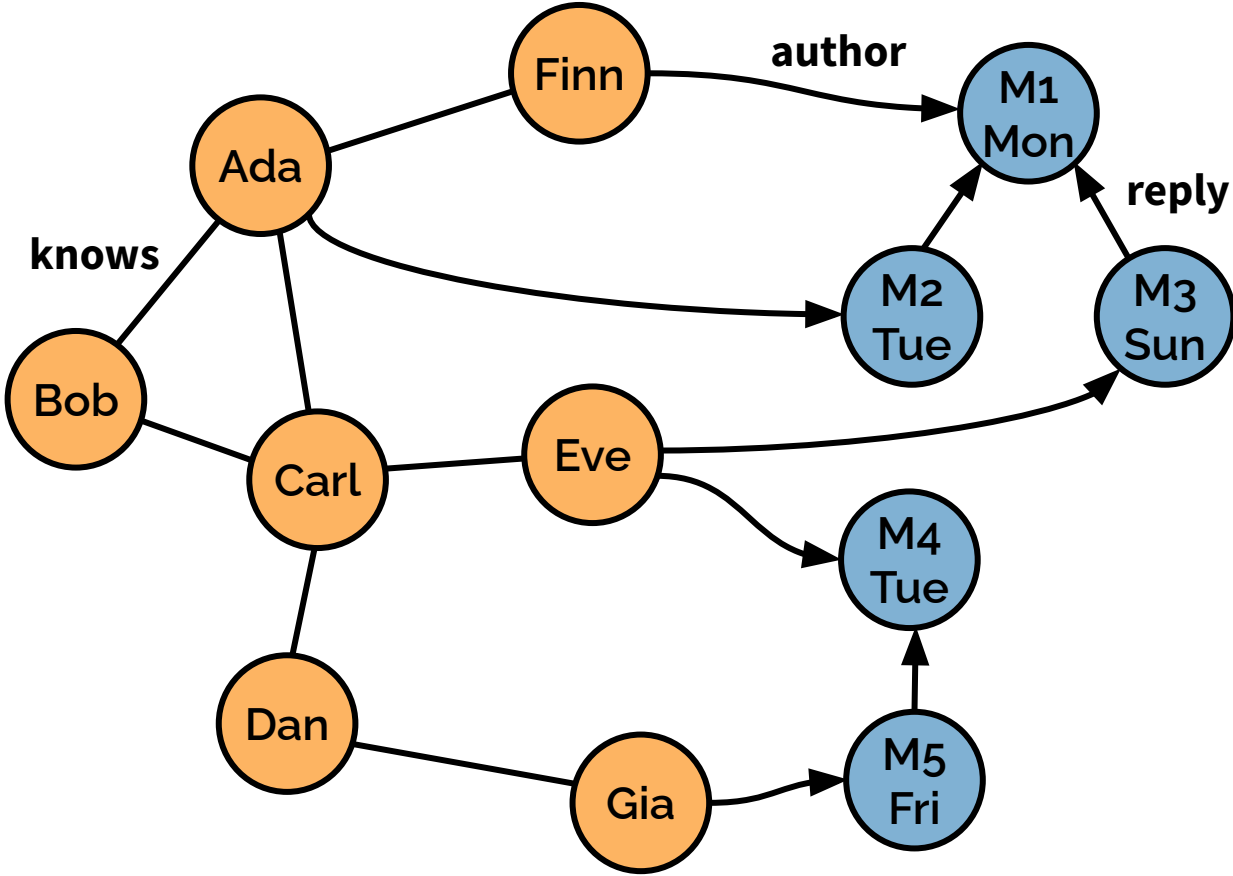
*Message nodes*



Data set

Queries

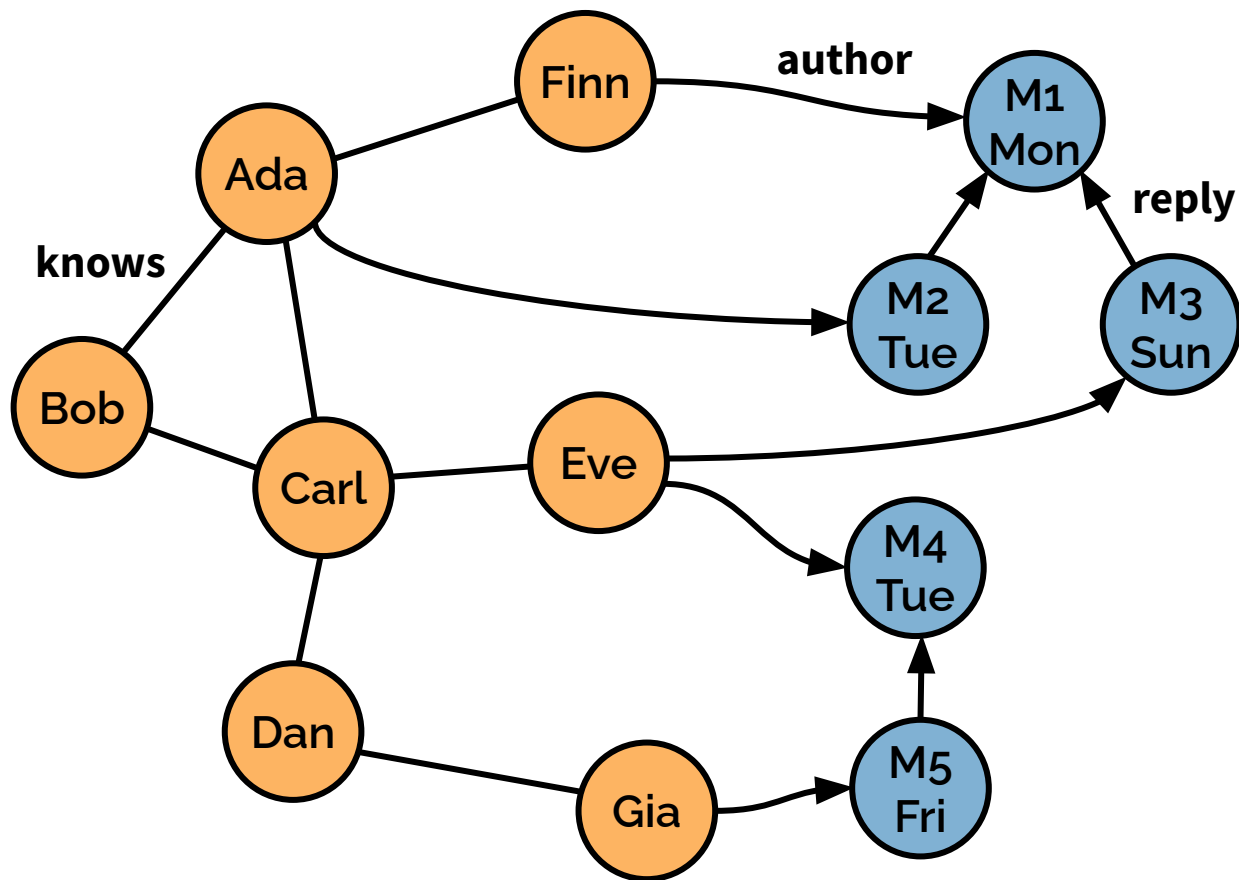
Updates



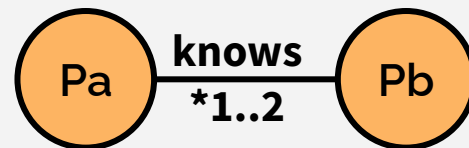
Data set

Queries

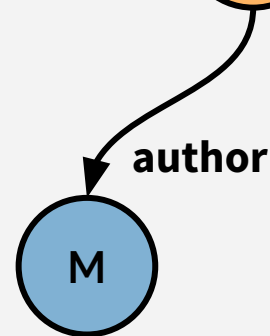
Updates



**Q9(\$name, \$day)**



*name = \$name*

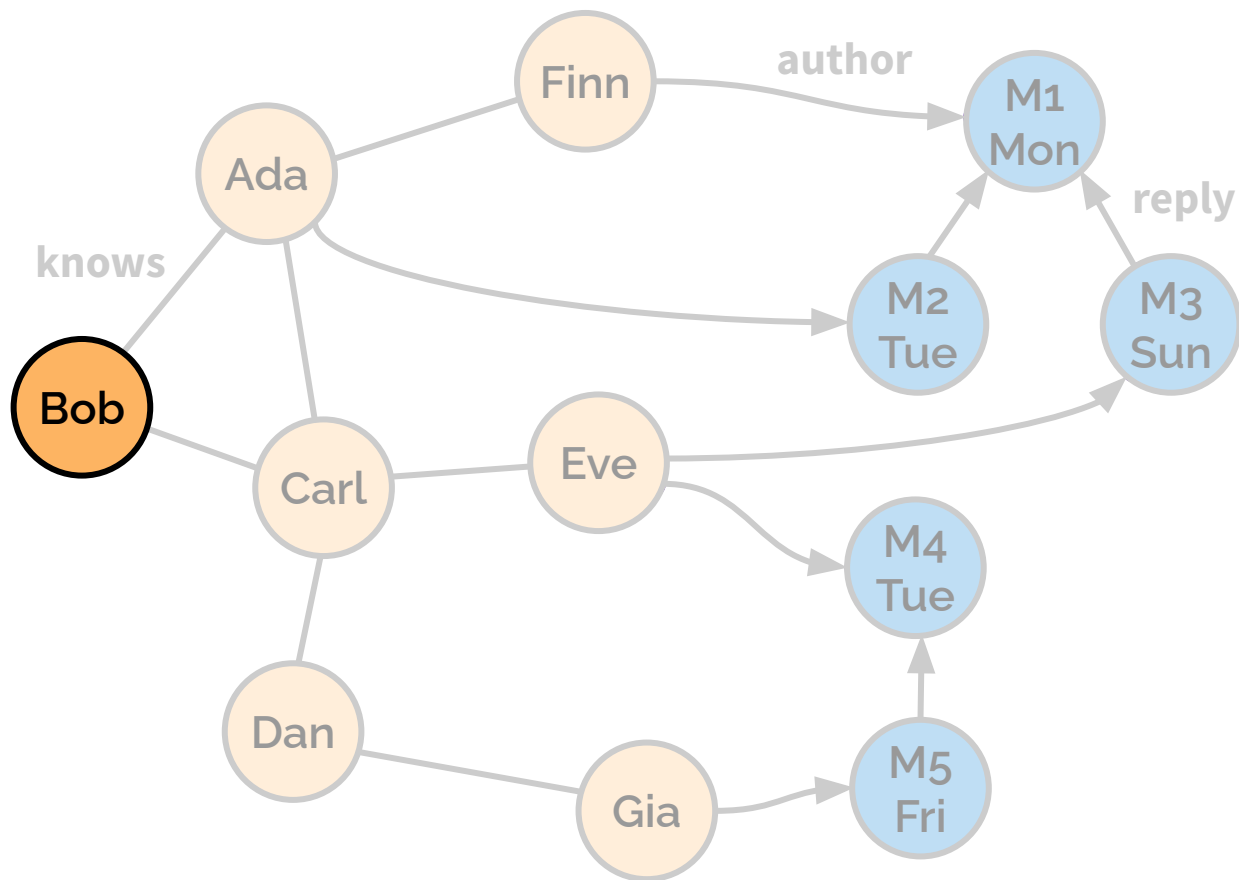


*creation date < \$day*

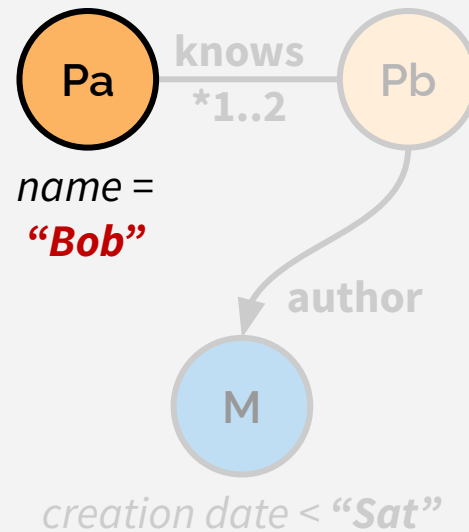
Data set

Queries

Updates



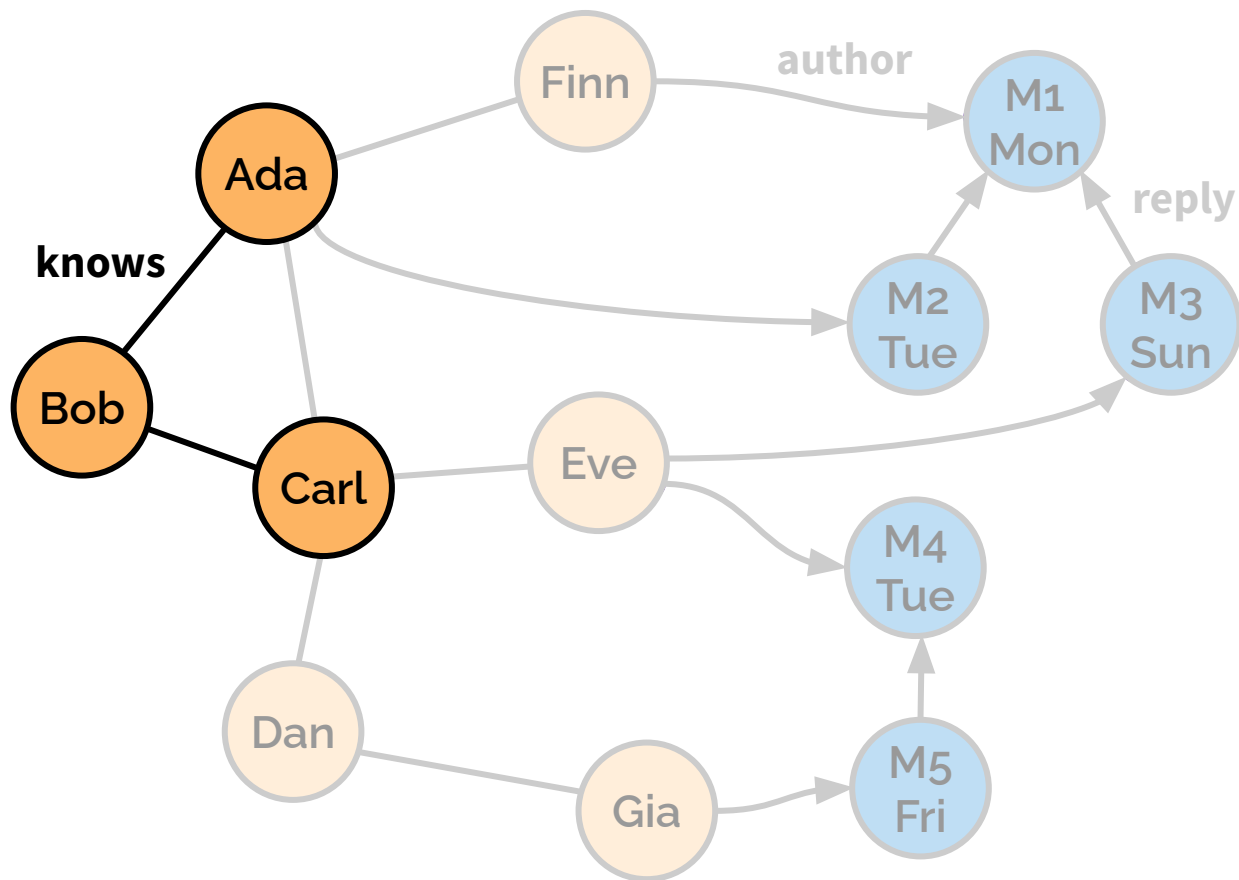
Q9(**"Bob"**, **"Sat"**)



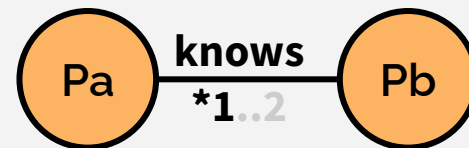
Data set

Queries

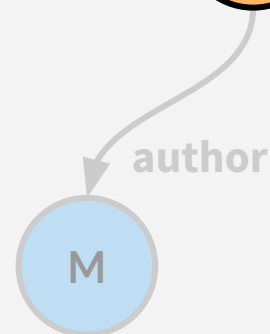
Updates



Q9(**"Bob"**, **"Sat"**)



name =  
**"Bob"**

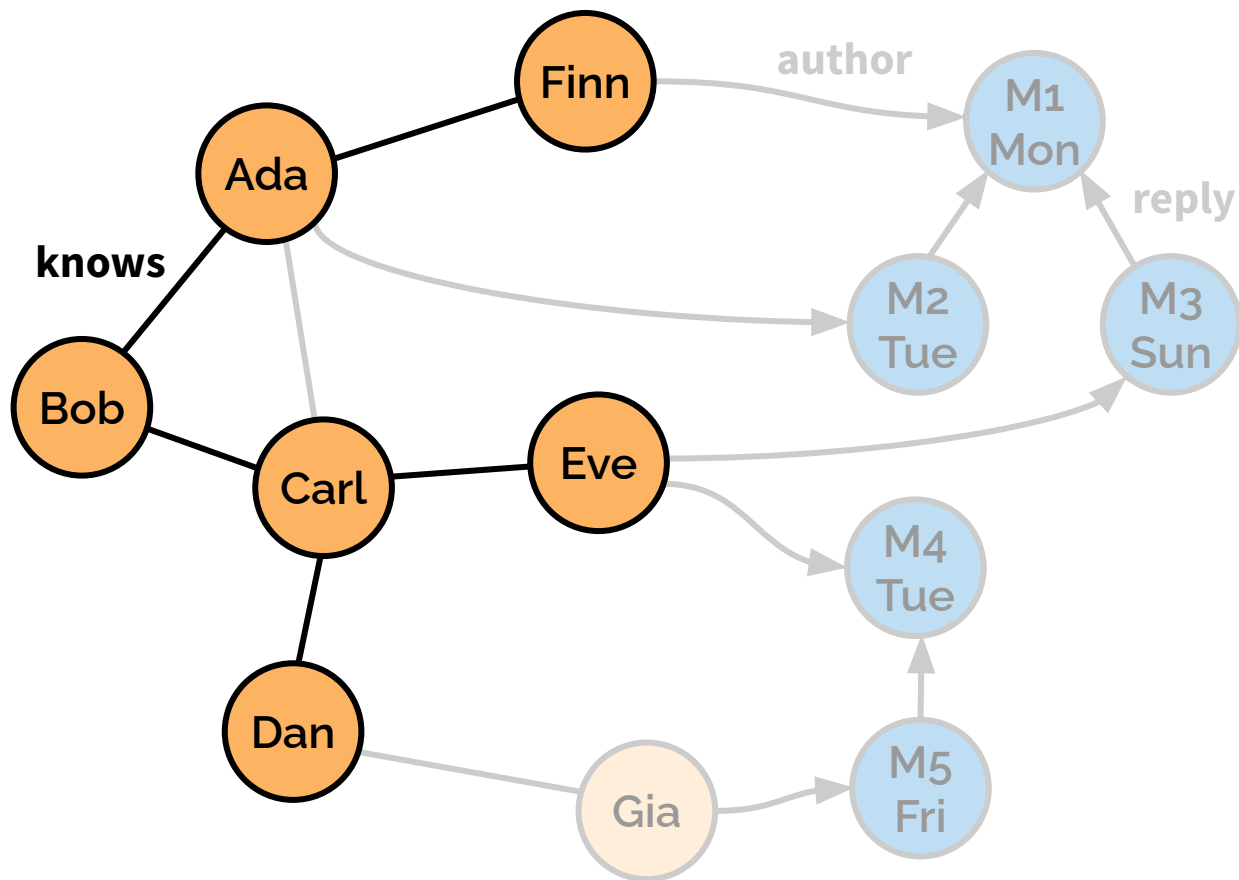


creation date < **"Sat"**

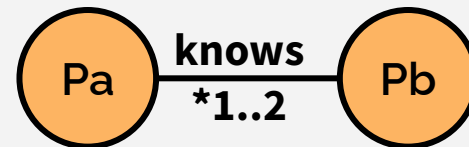
Data set

Queries

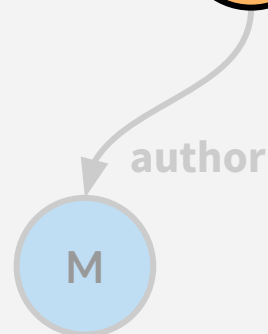
Updates



Q9(**"Bob"**, **"Sat"**)



name =  
**"Bob"**

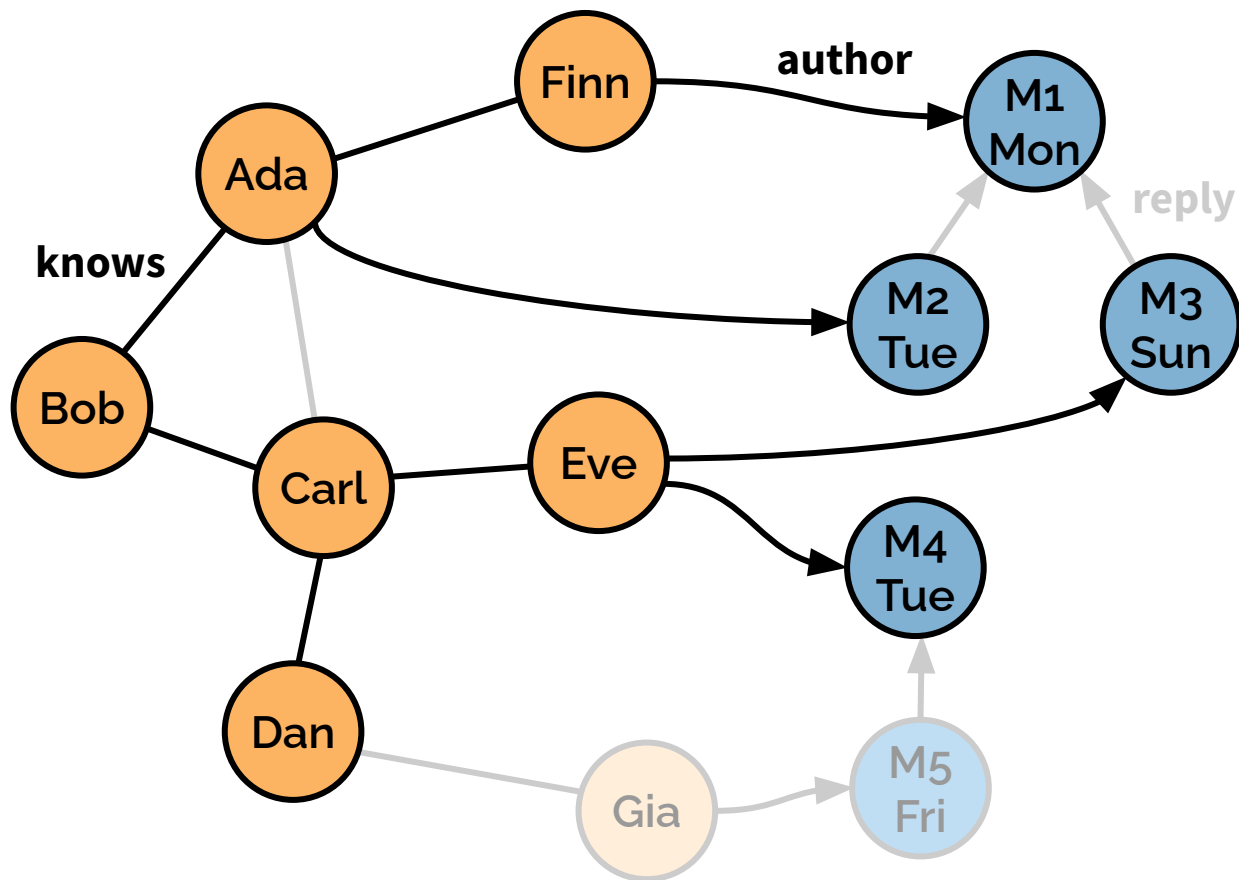


creation date < **"Sat"**

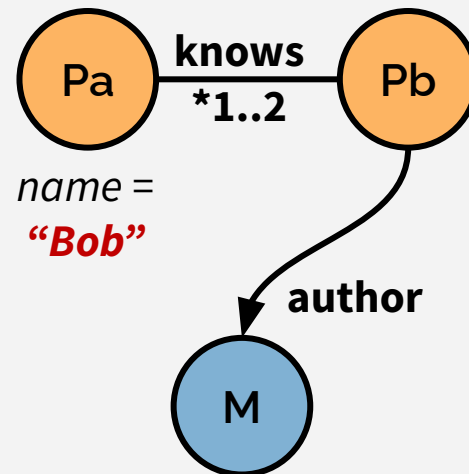
Data set

Queries

Updates



Q9(**"Bob"**, **"Sat"**)



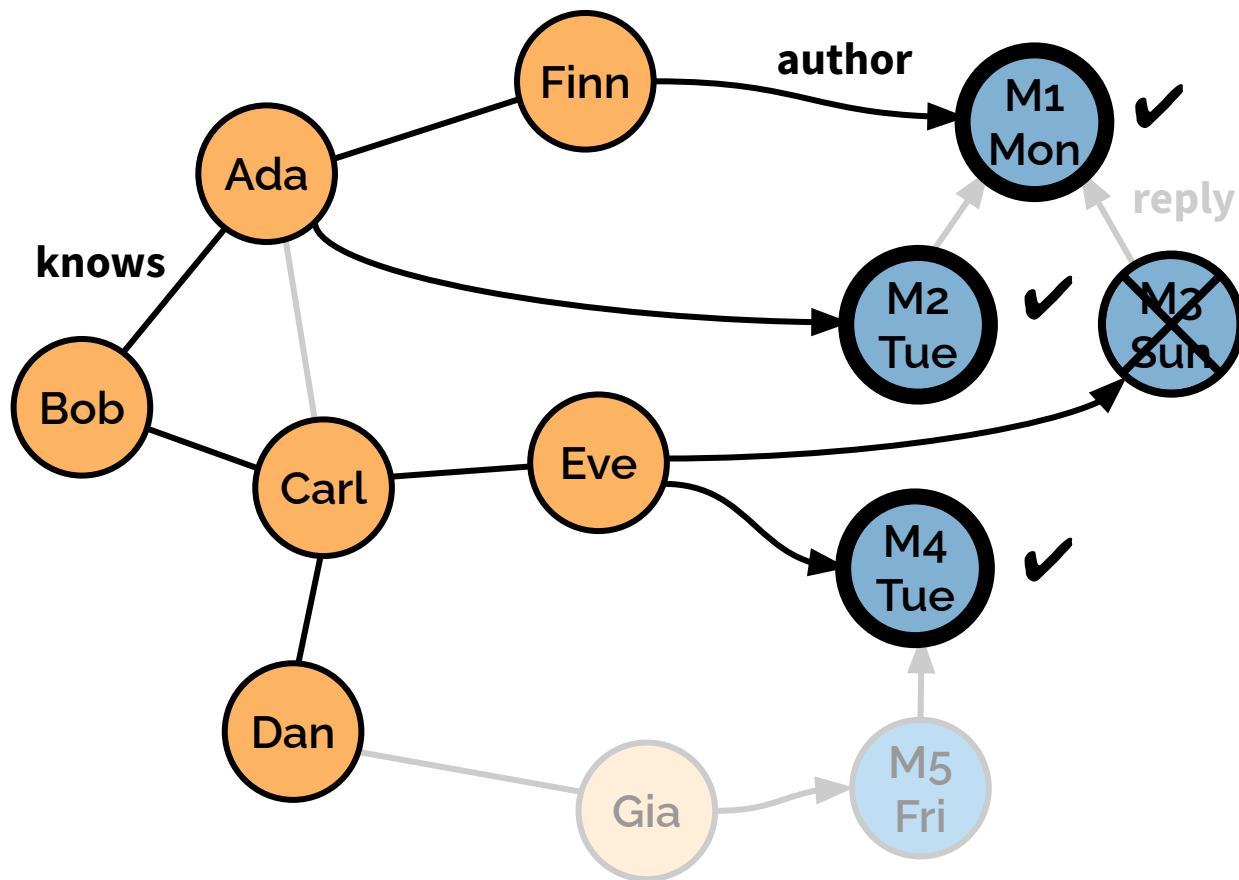
*name =*  
**"Bob"**

*creation date <* **"Sat"**

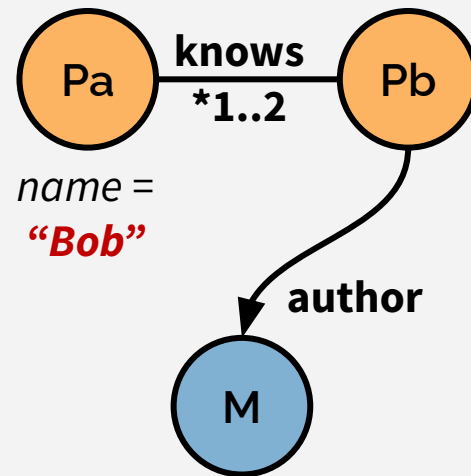
Data set

Queries

Updates



Q9("Bob", "Sat")

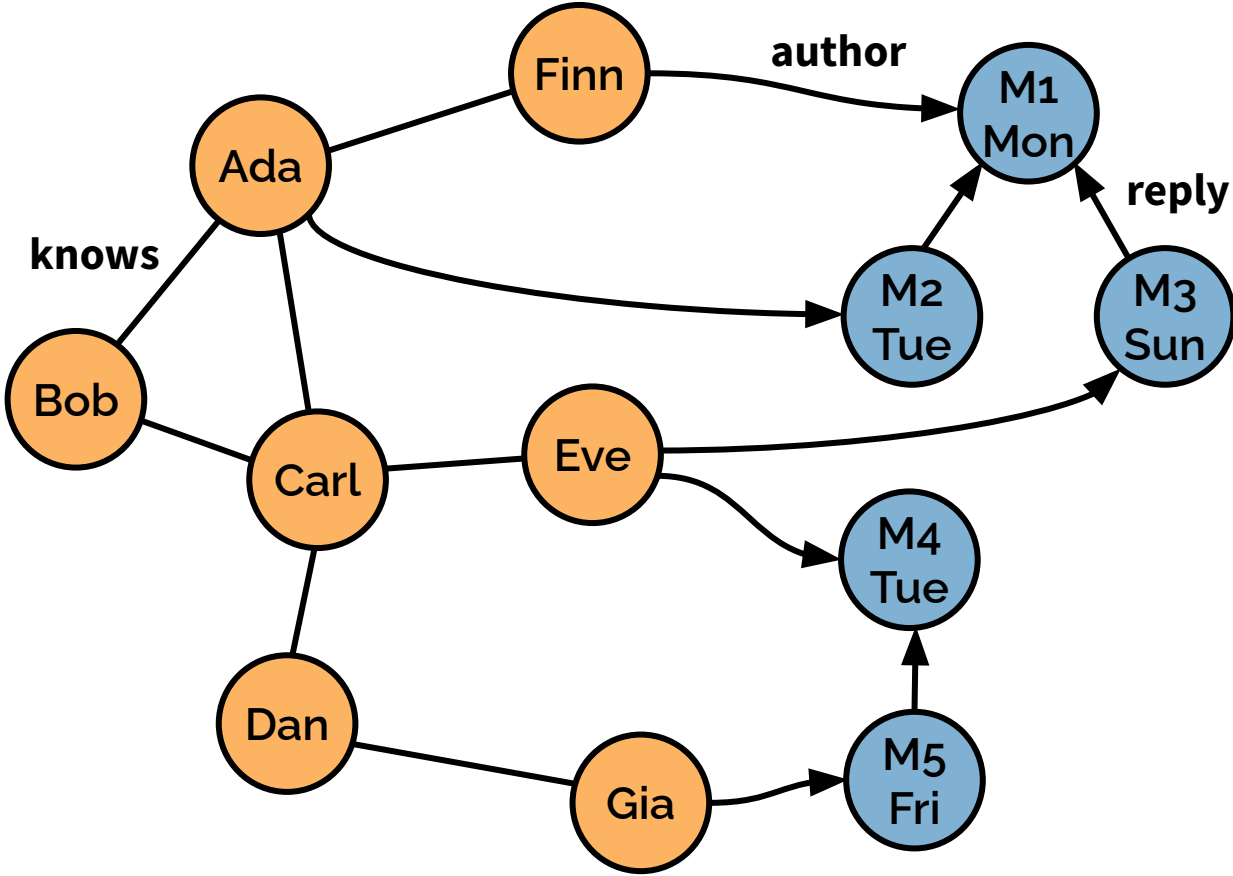


*creation date < "Sat"*

Data set

Queries

Updates

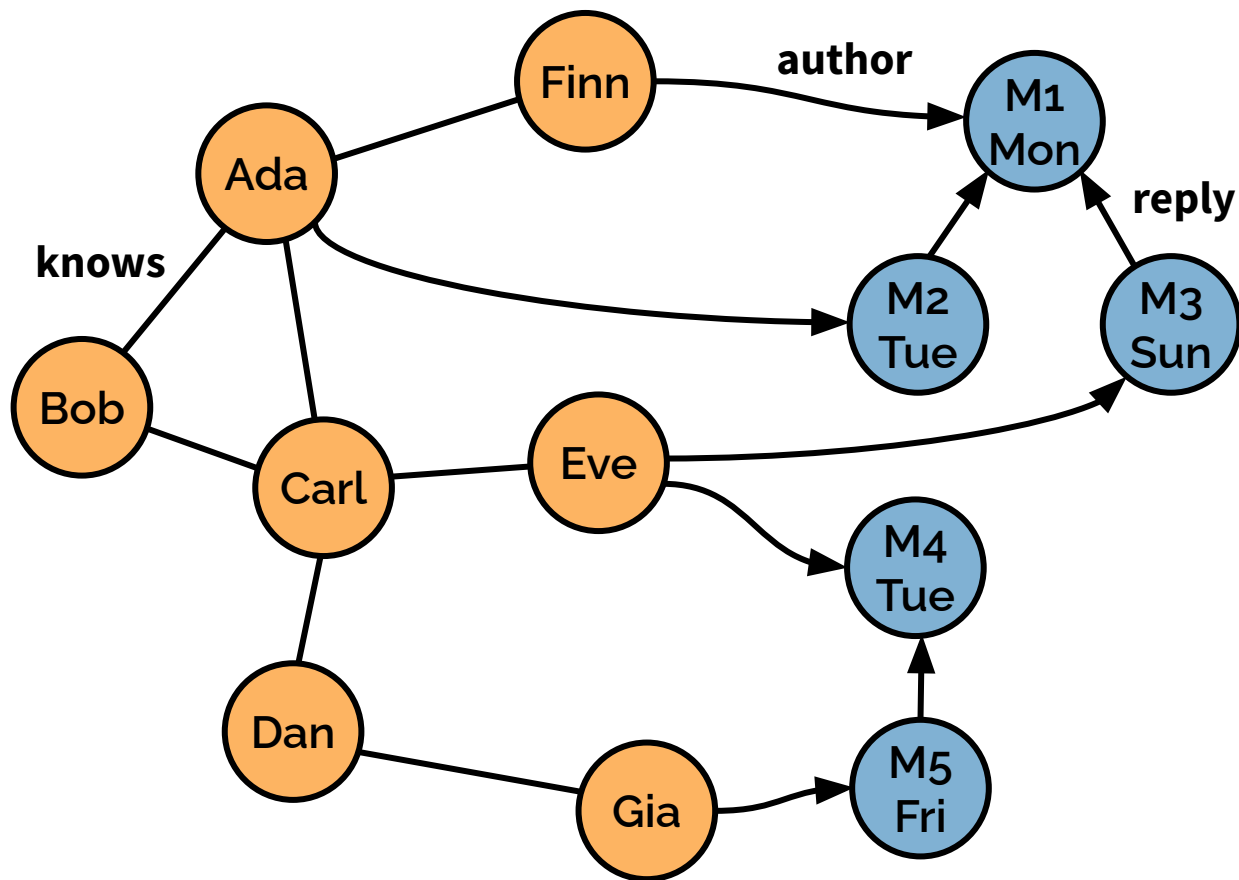




Data set

Queries

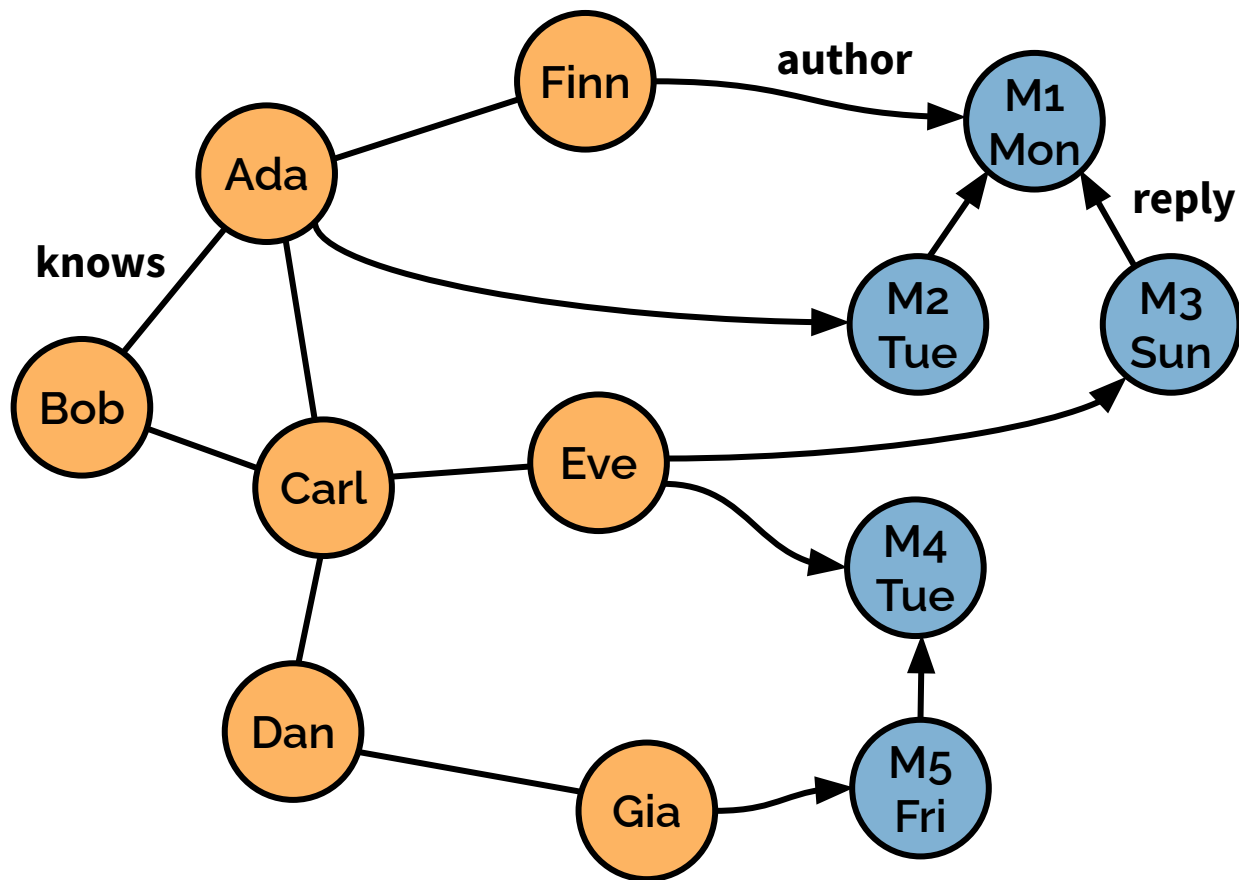
Updates



Data set

Queries

Updates



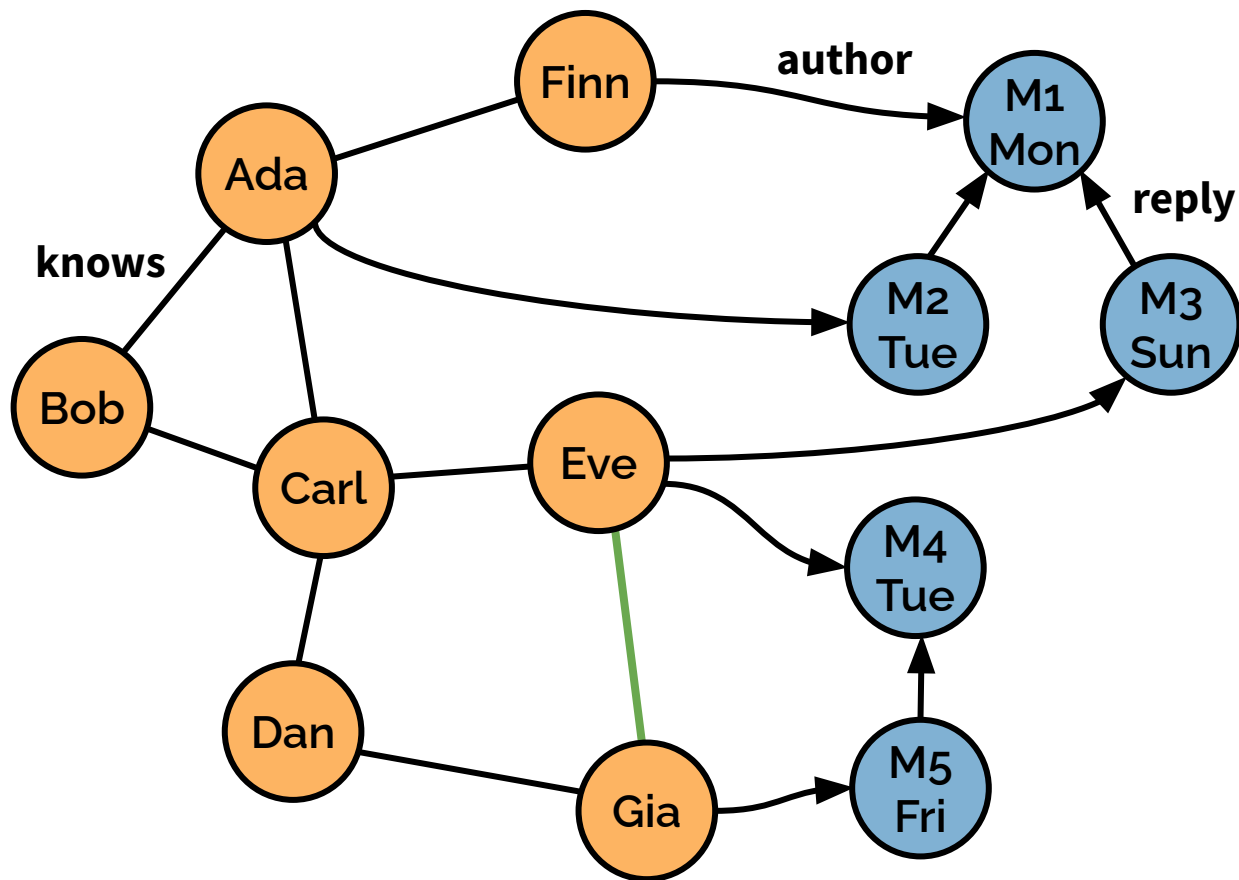
Updates

+ knows("Eve", "Gia")

Data set

Queries

Updates



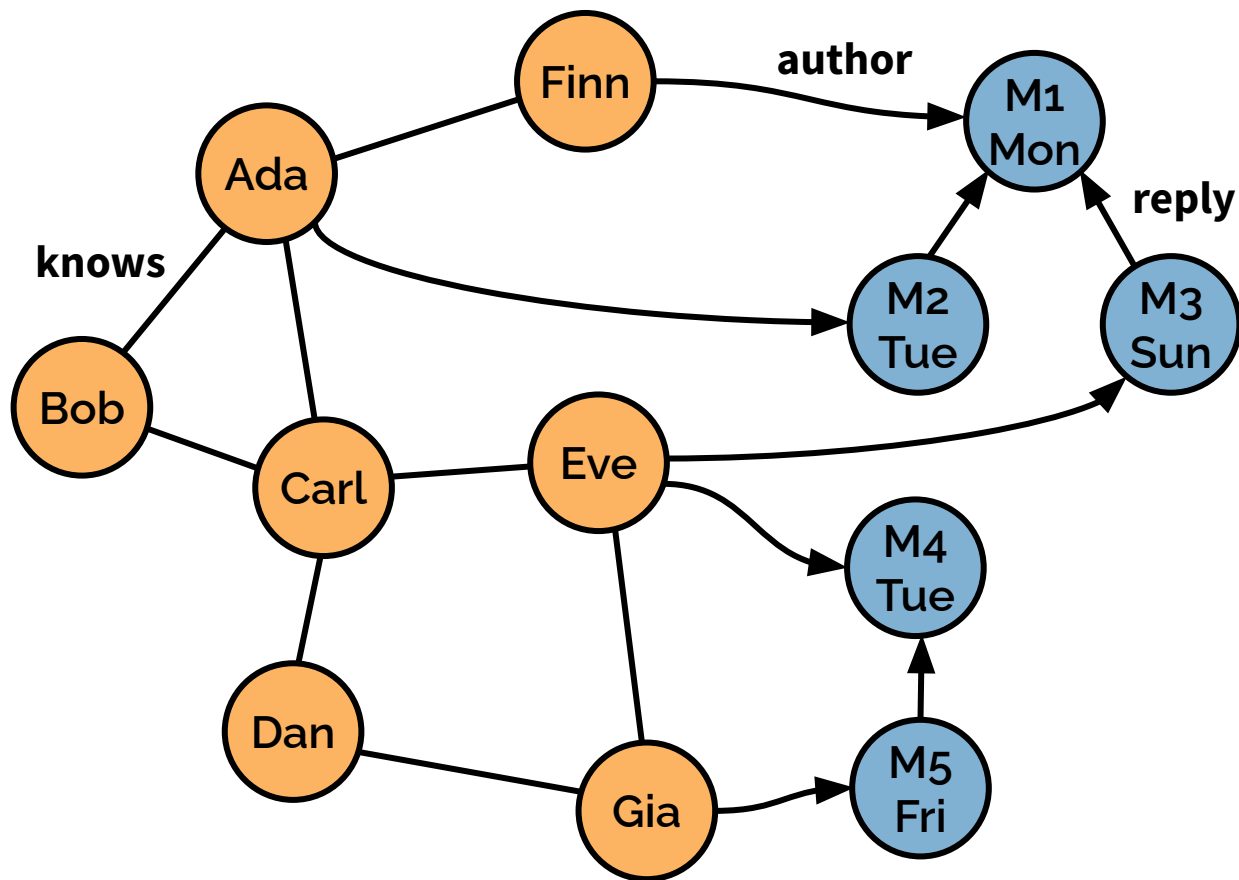
Updates

+ knows("Eve", "Gia")

Data set

Queries

Updates



Updates

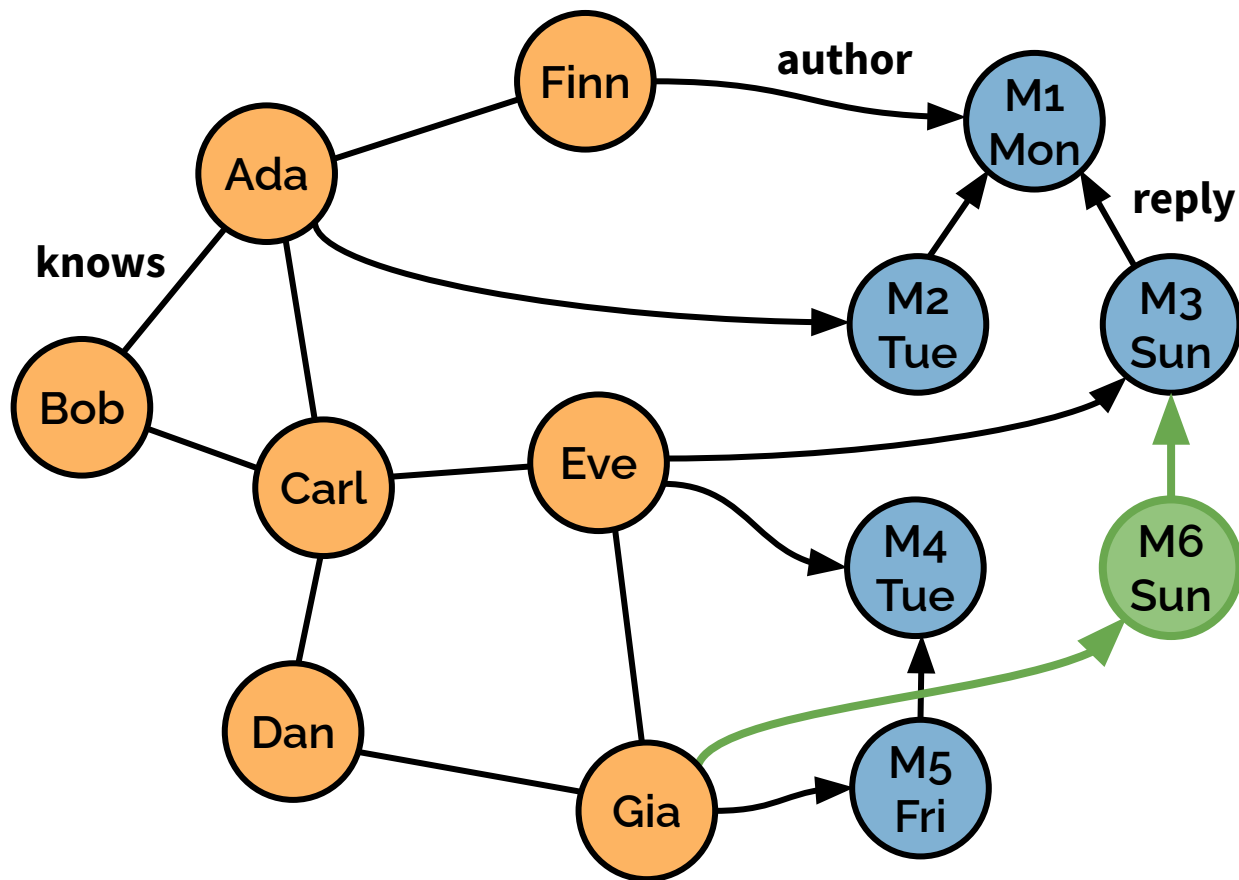
+ knows("Eve", "Gia")

+ Comment("Gia", "M3")

Data set

Queries

Updates



Updates

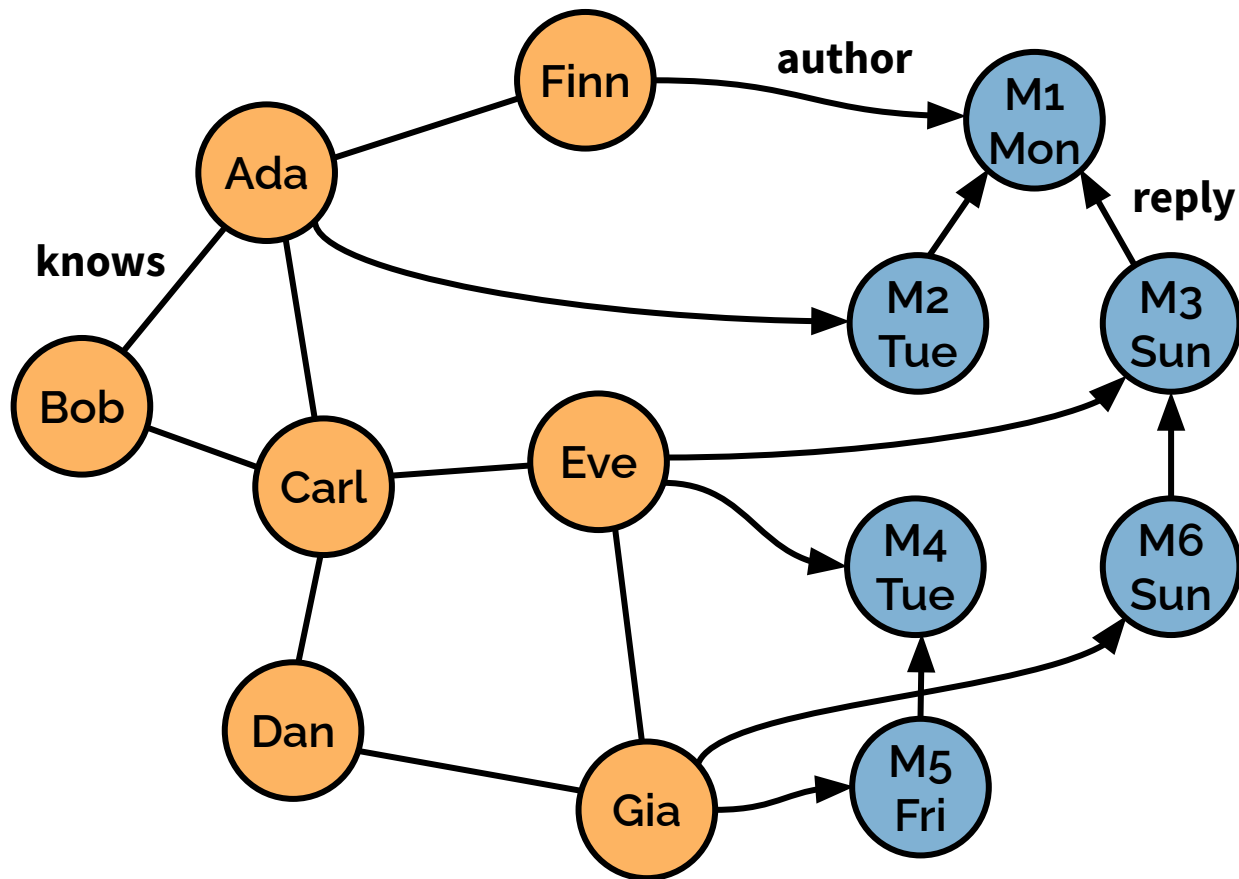
+ knows("Eve", "Gia")

+ Comment("Gia", "M3")

Data set

Queries

Updates



Updates

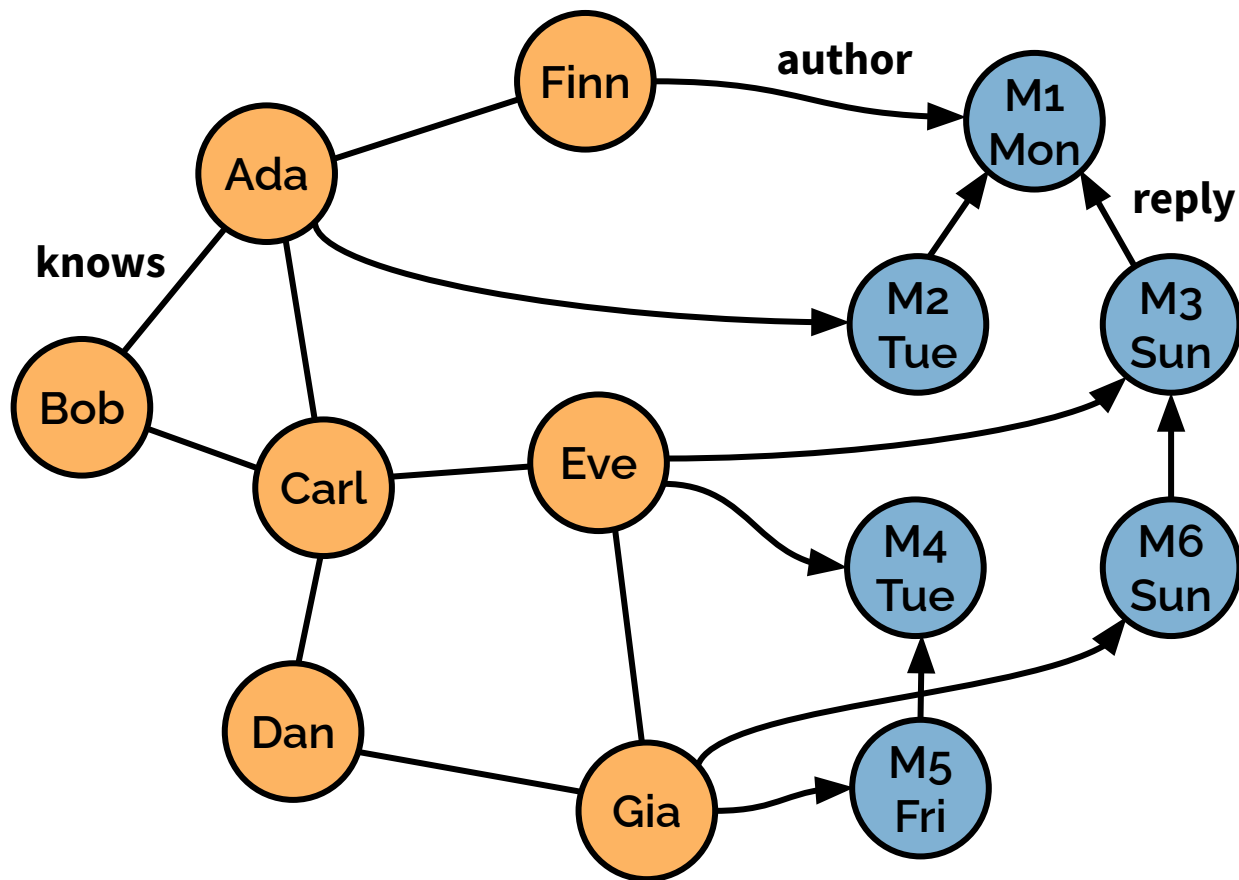
+ knows("Eve", "Gia")

+ Comment("Gia", "M3")

Data set

Queries

Updates



Updates

+ knows("Eve", "Gia")

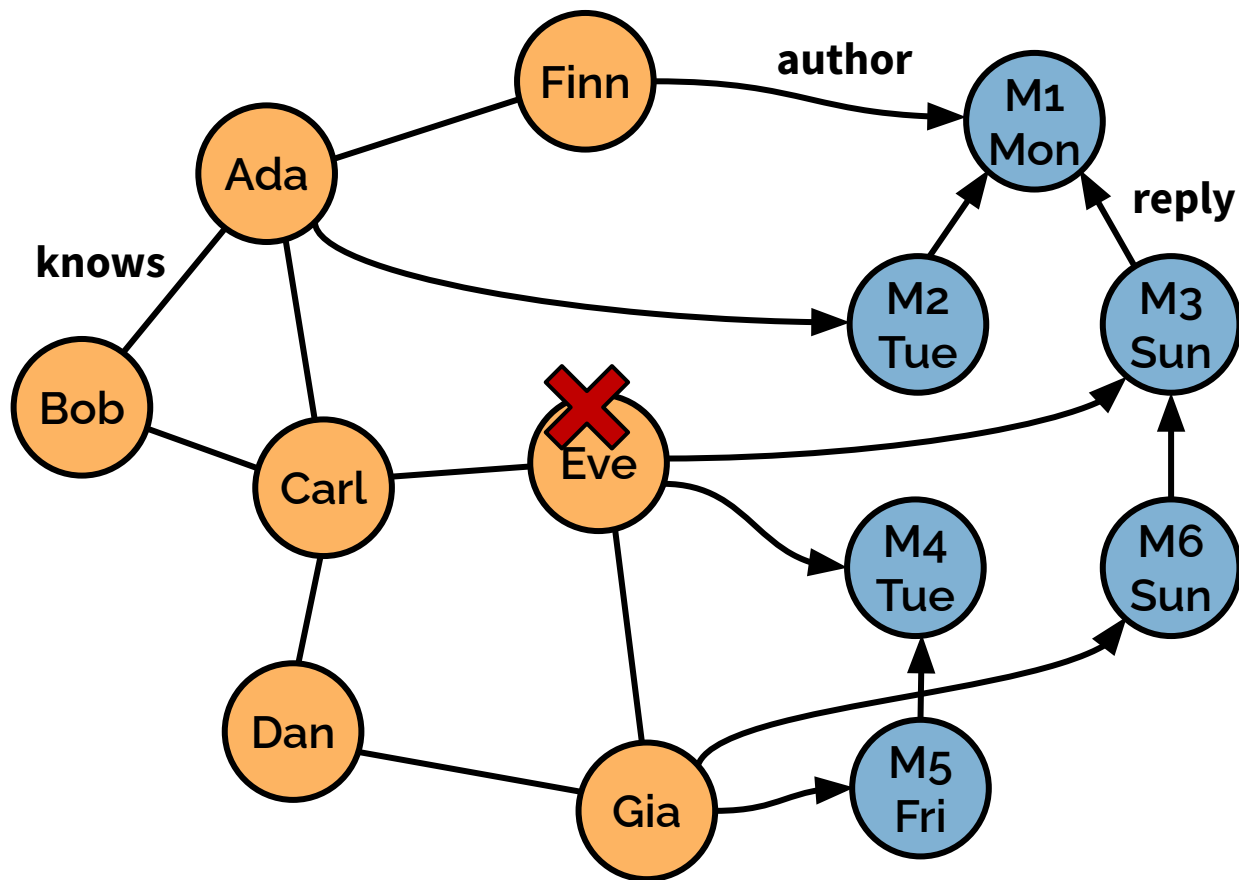
+ Comment("Gia", "M3")

- Person("Eve")

Data set

Queries

Updates



Updates

+ knows("Eve", "Gia")

+ Comment("Gia", "M3")

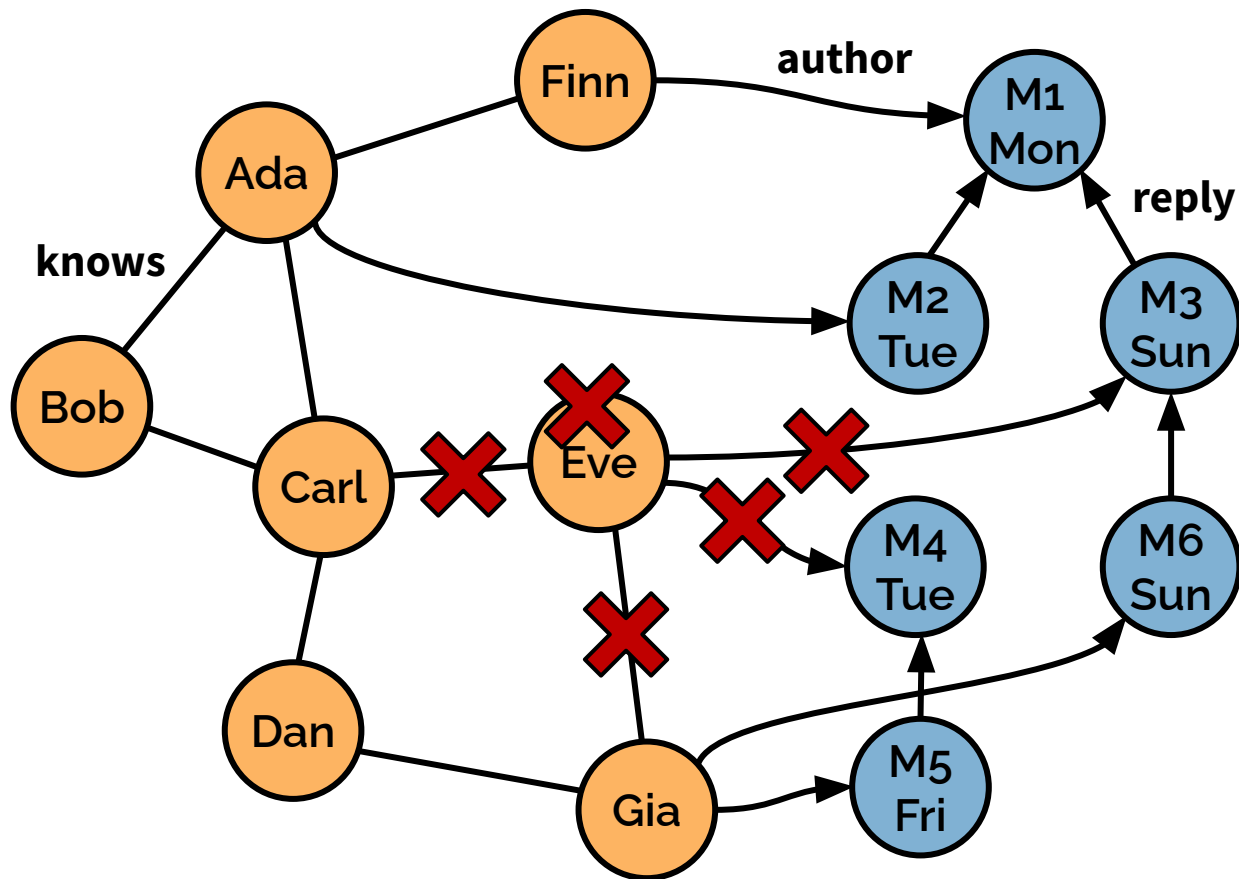
- Person("Eve")



Data set

Queries

Updates



Updates

+ knows("Eve", "Gia")

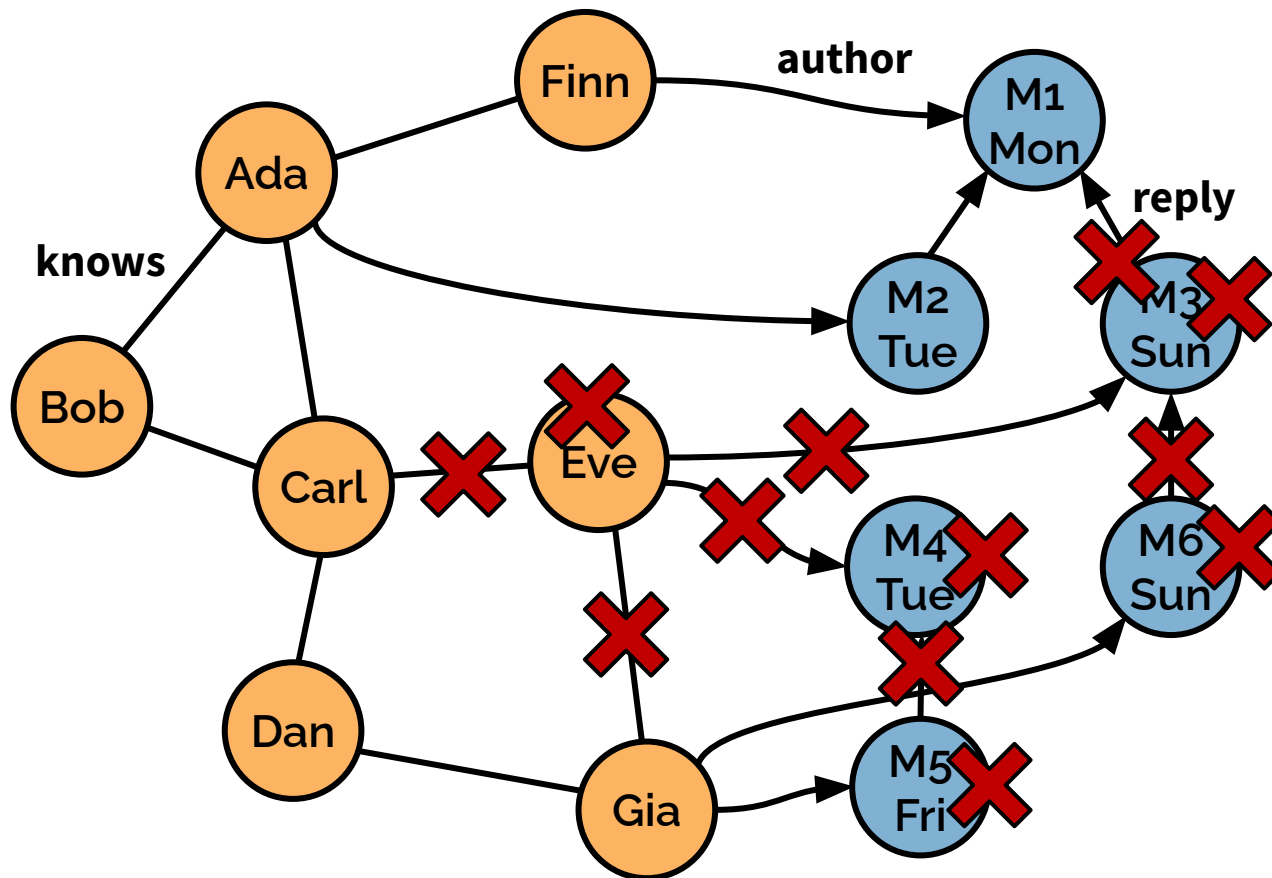
+ Comment("Gia", "M3")

- Person("Eve")

Data set

Queries

Updates



Updates

+ knows("Eve", "Gia")

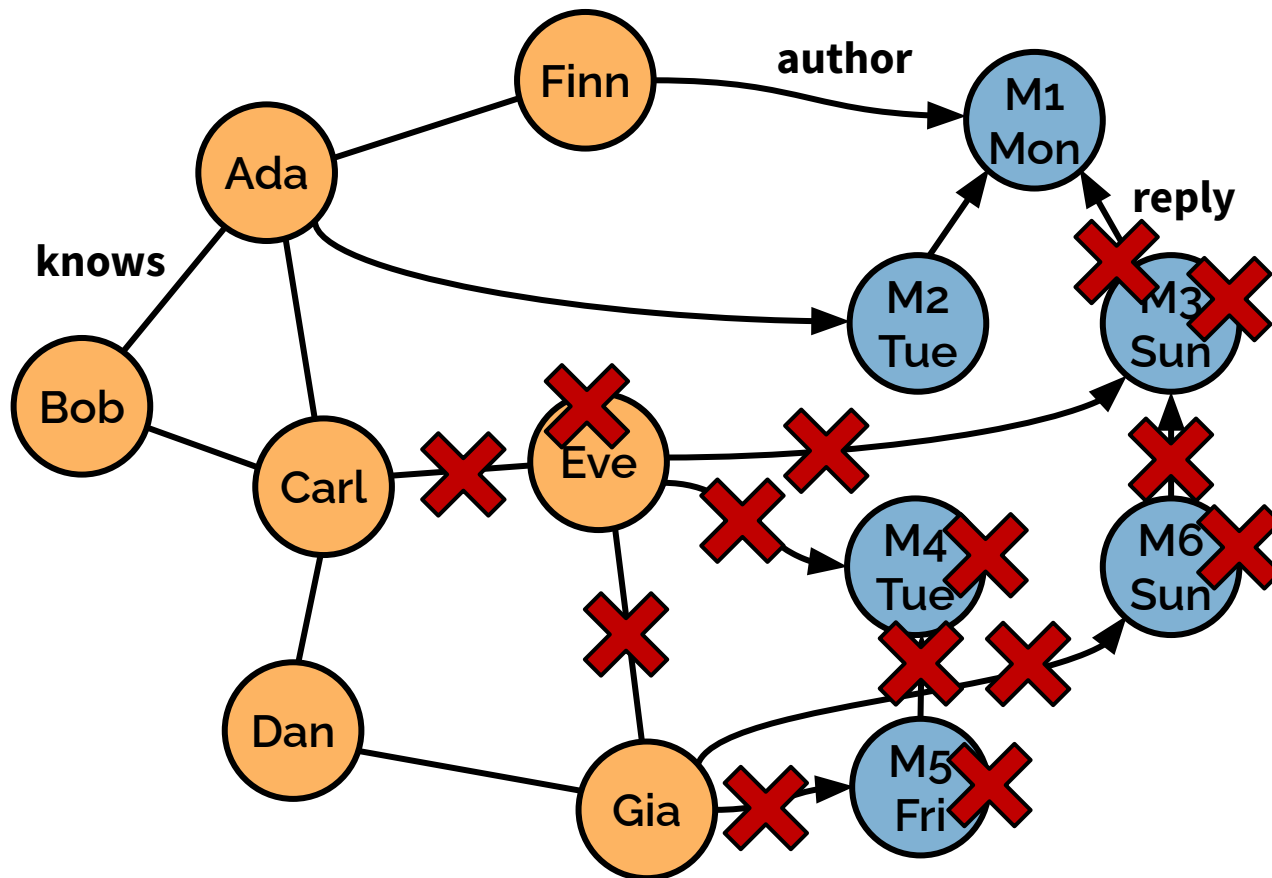
+ Comment("Gia", "M3")

- Person("Eve")

Data set

Queries

Updates



Updates

+ knows("Eve", "Gia")

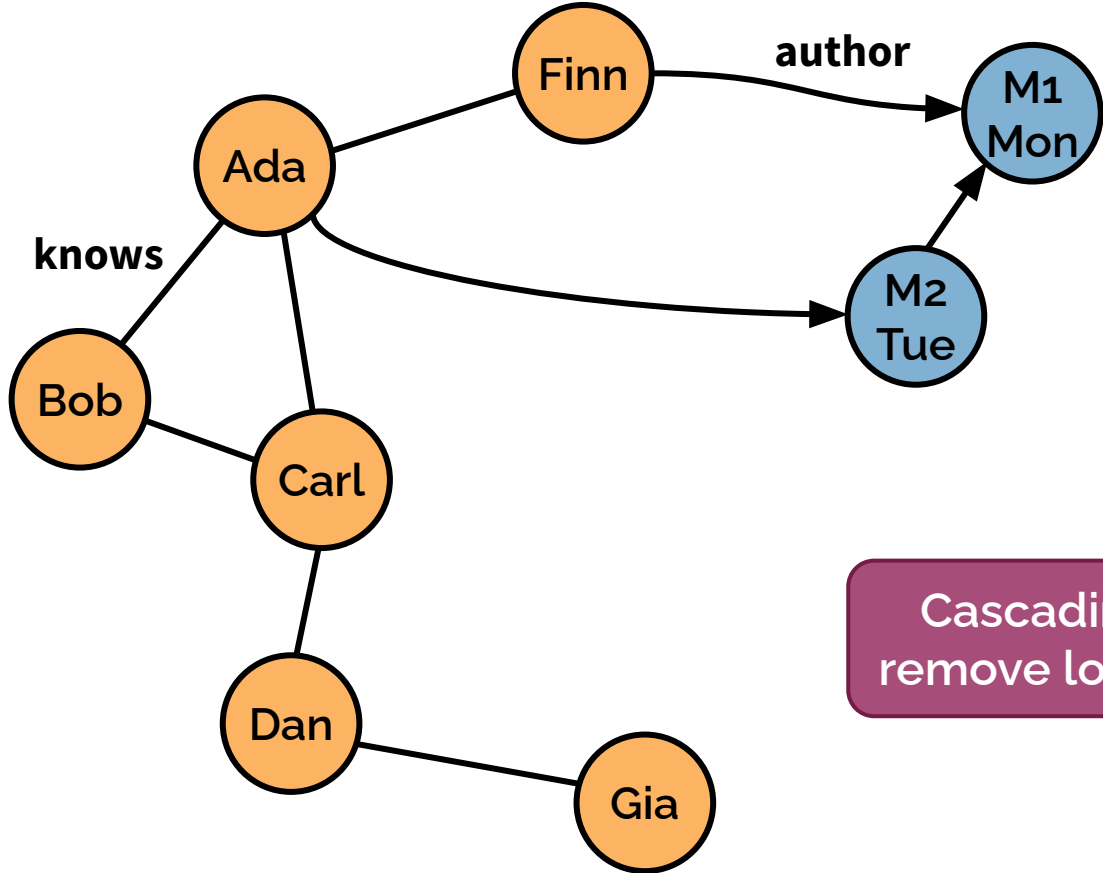
+ Comment("Gia", "M3")

- Person("Eve")

Data set

Queries

Updates



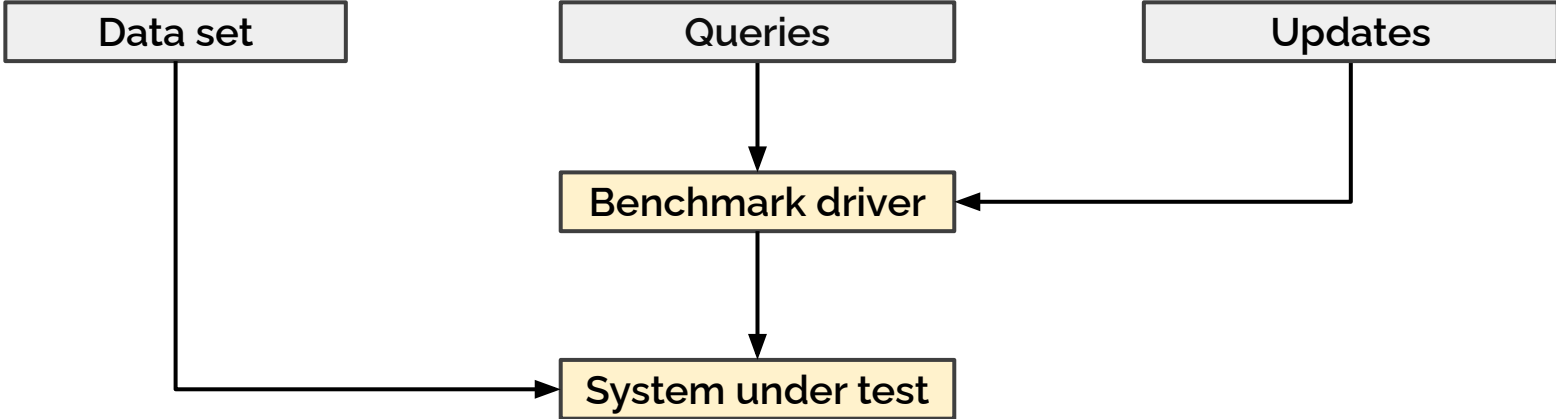
Updates

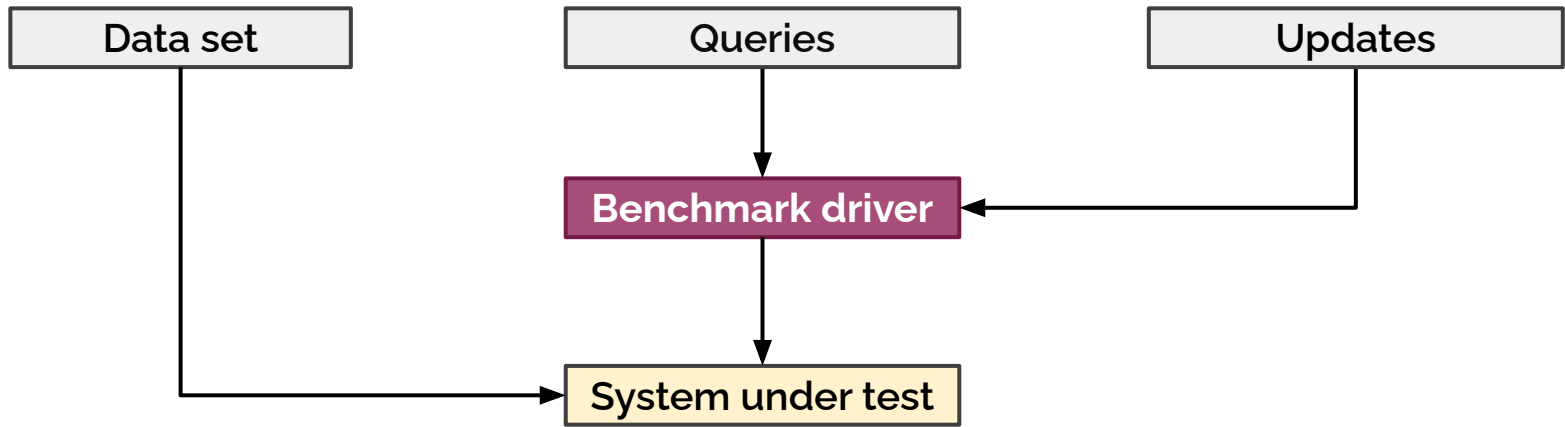
+ knows("Eve", "Gia")

+ Comment("Gia", "M3")

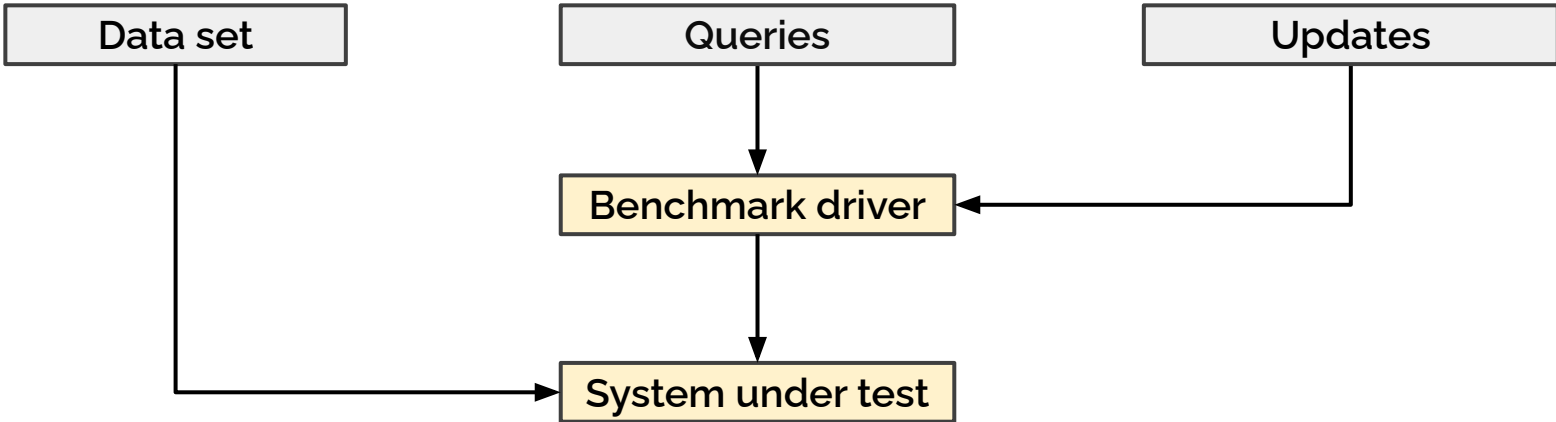
- Person("Eve")

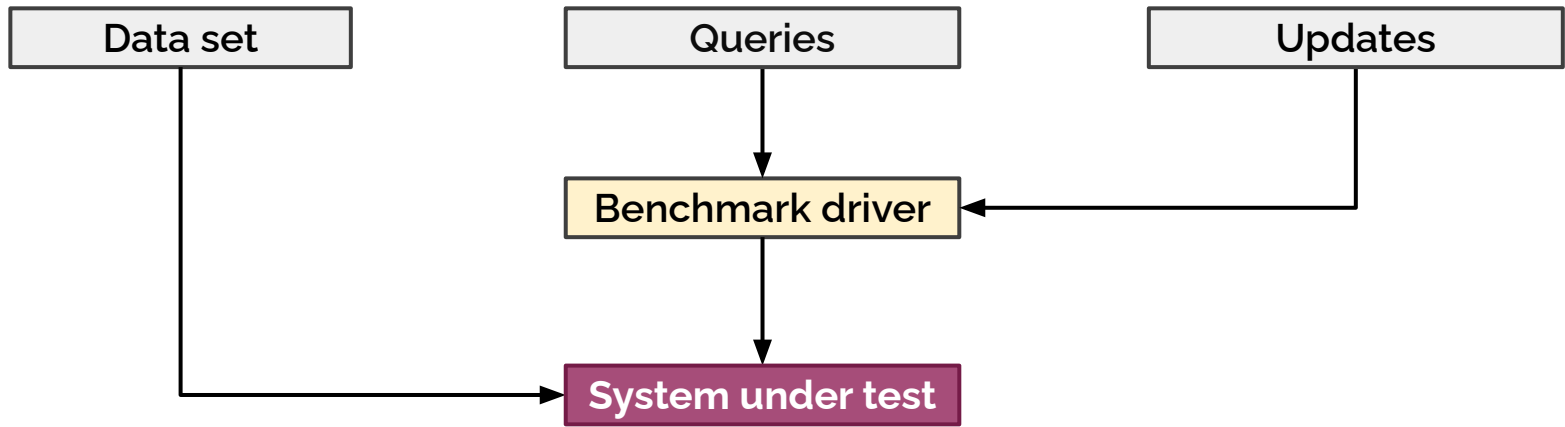
Cascading deletes  
remove lots of entities





- Runs queries and updates concurrently
- Schedules operations to be executable
- Collects results





Candidate systems:

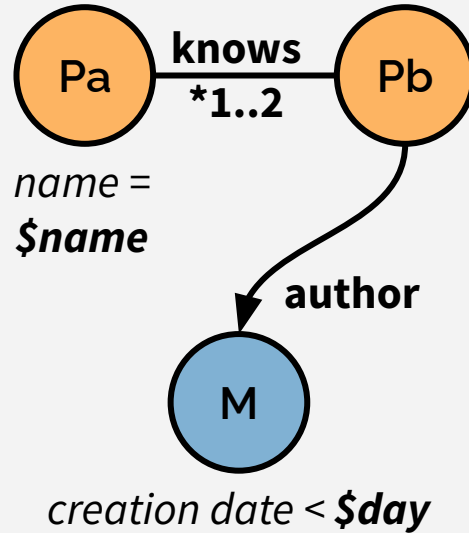
- Graph databases
- Triplestores
- Relational databases



## SQL

```
SELECT DISTINCT m.id
FROM (
  SELECT k.p2id AS id
  FROM person Pa,
        knows k
  WHERE Pa.name = $name
        AND Pa.id = k.p1id
  UNION
  SELECT k2.p2id AS id
  FROM person Pa,
        knows k1,
        knows k2
  WHERE Pa.name = $name
        AND Pa.id = k1.p1id
        AND k1.p2id = k2.p1id
        AND k1.p1id <> k2.p2id
) Pb,
Message m
WHERE Pb.id = m.authorId
      AND m.creationDate < $day
```

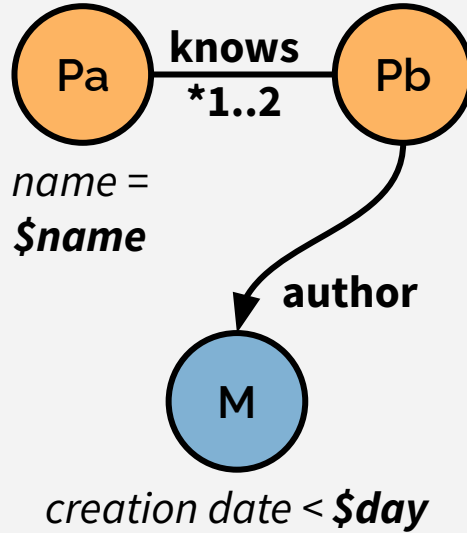
## Q9(\$name, \$day)



## SQL

```
SELECT DISTINCT m.id
FROM (
  SELECT k.p2id AS id
  FROM person Pa,
       knows k
  WHERE Pa.name = $name
       AND Pa.id = k.p1id
  UNION
  SELECT k2.p2id AS id
  FROM person Pa,
       knows k1,
       knows k2
  WHERE Pa.name = $name
       AND Pa.id = k1.p1id
       AND k1.p2id = k2.p1id
       AND k1.p1id <> k2.p2id
) Pb,
Message m
WHERE Pb.id = m.authorId
   AND m.creationDate < $day
```

## Q9(\$name, \$day)

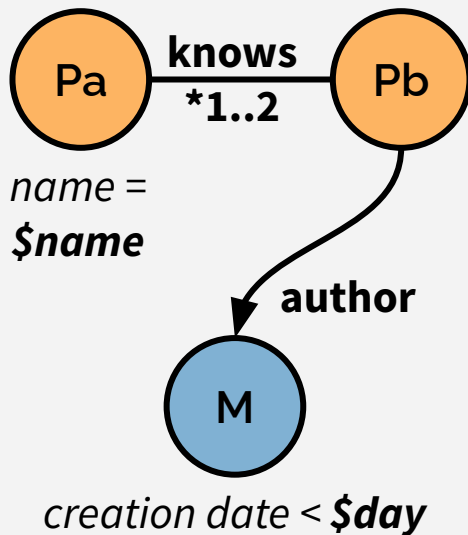


Graph pattern matching language with visual graph syntax inspired by Cypher

## SQL

```
SELECT DISTINCT m.id
FROM (
  SELECT k.p2id AS id
  FROM person Pa,
       knows k
  WHERE Pa.name = $name
       AND Pa.id = k.p1id
  UNION
  SELECT k2.p2id AS id
  FROM person Pa,
       knows k1,
       knows k2
  WHERE Pa.name = $name
       AND Pa.id = k1.p1id
       AND k1.p2id = k2.p1id
       AND k1.p1id <> k2.p2id
) Pb,
Message m
WHERE Pb.id = m.authorId
     AND m.creationDate < $day
```

## Q9(\$name, \$day)



Graph pattern matching language with visual graph syntax inspired by Cypher

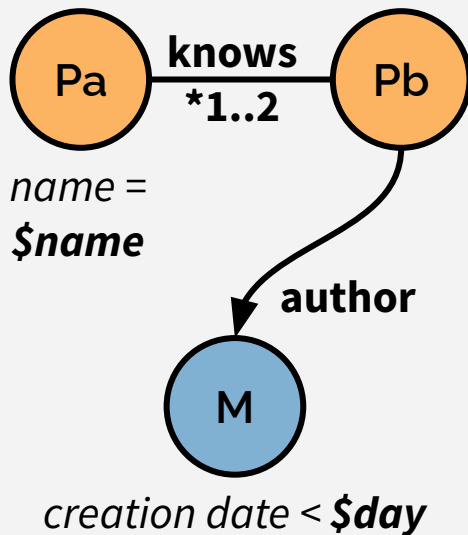
## SQL/PGQ, June 2023

```
SELECT id FROM GRAPH_TABLE ($N
  MATCH ANY ACYCLIC
  (Pa:Person WHERE Pa.name = $name)
  -[:knows]-{1,2} (Pb:Person)
  -[:author]-> (m:Message)
  WHERE m.creationDate < $day
  COLUMNS (m.id))
```

## SQL

```
SELECT DISTINCT m.id
FROM (
  SELECT k.p2id AS id
  FROM person Pa,
       knows k
  WHERE Pa.name = $name
       AND Pa.id = k.p1id
 UNION
  SELECT k2.p2id AS id
  FROM person Pa,
       knows k1,
       knows k2
  WHERE Pa.name = $name
       AND Pa.id = k1.p1id
       AND k1.p2id = k2.p1id
       AND k1.p1id <> k2.p2id
) Pb,
Message m
WHERE Pb.id = m.authorId
     AND m.creationDate < $day
```

## Q9(\$name, \$day)



Graph pattern matching language with visual graph syntax inspired by Cypher

## SQL/PGQ, June 2023

```
SELECT id FROM GRAPH_TABLE (sN
  MATCH ANY ACYCLIC
  (Pa:Person WHERE Pa.name = $name)
  -[:knows]-{1,2} (Pb:Person)
  -[:author]-> (m:Message)
  WHERE m.creationDate < $day
  COLUMNS (m.id))
```

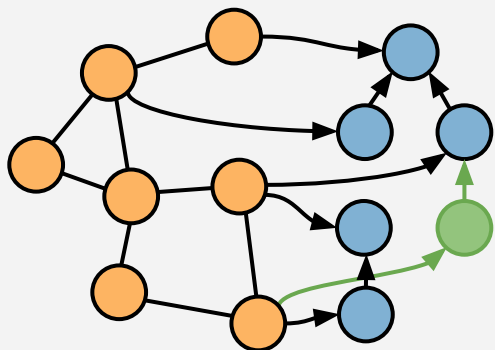
## GQL, 2024

```
MATCH ANY ACYCLIC
(Pa:Person WHERE Pa.name = $name)
-[:knows]-{1,2} (Pb:Person)
-[:author]-> (m:Message)
WHERE m.creationDate < $day
RETURN DISTINCT m.id
```

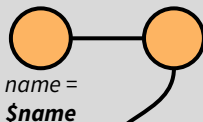
# **SNB Workloads**



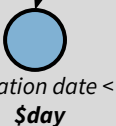
## SNB Interactive



Q9(\$name, \$day)



name =  
\$name



creation date <  
\$day

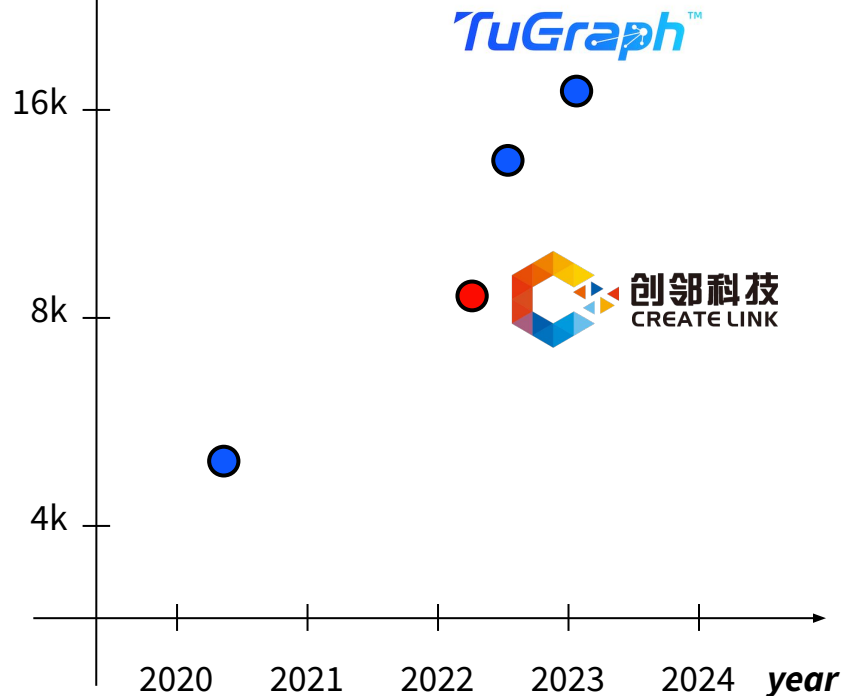
Queries start in 1-2 person nodes

Queries and updates run concurrently

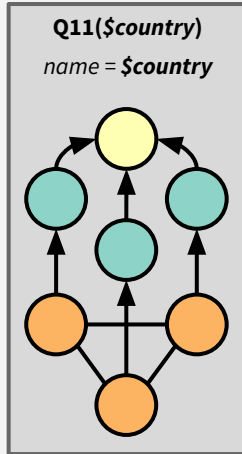
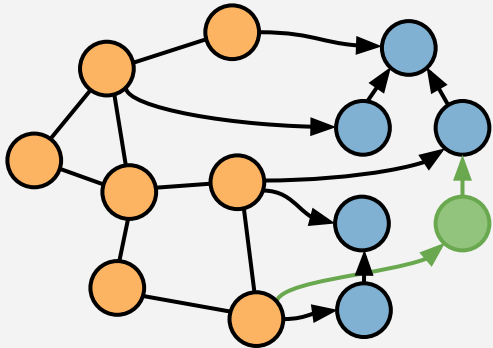
Goal: high throughput (ops/s)

## Results on the 100GB data set

throughput  
log scale



# SNB Business Intelligence



Queries touch on large portions of the data

Both bulk and concurrent updates allowed

Goal: high throughput & low query runtimes

# Results on the 1TB data set



- Power@SF: 30,990
- Throughput@SF: 12,993

Diverse range of CPUs used for SNB:

- AMD EPYC Genoa
- Intel Xeon Ice Lake
- YiTian 710 (Arm v9)

- More results expected in 2023
- Graph accelerators released soon





# Auditing and trademark

Audited benchmark runs can be conducted by independent third-party auditors

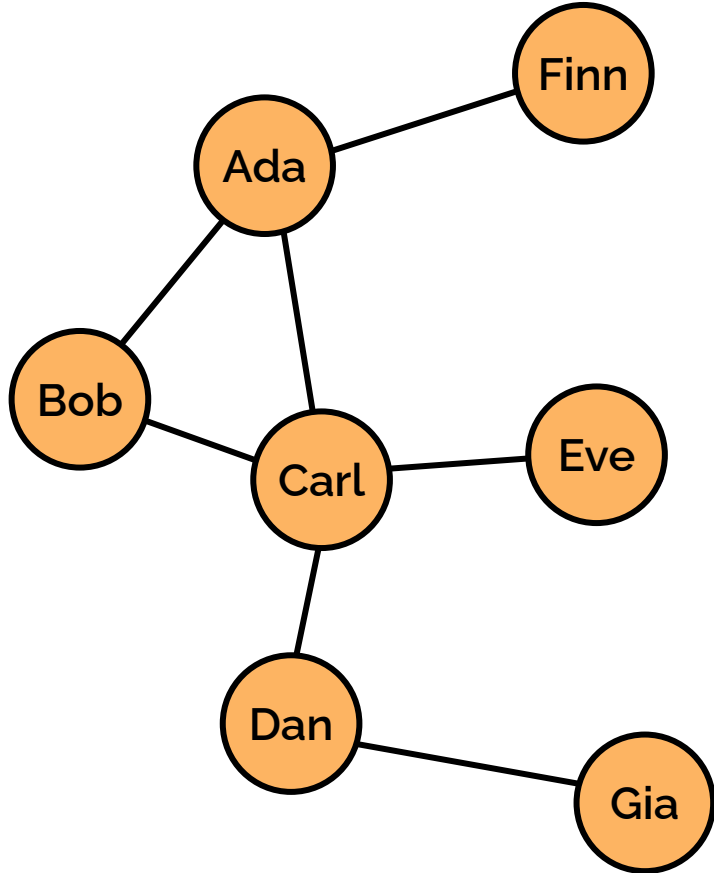
- LDBC is **trademarked** worldwide
- Only a **result produced by a certified auditor is an “LDBC benchmark result”**
- Unofficial benchmark results can be reported with a disclaimer:  
“This is NOT an official LDBC benchmark result”

# **LDDB Graphalytics Benchmark**

**Graph processing frameworks**

*(Apache Giraph, NetworKit, GraphBLAS, etc.)*

## Data set



## Algorithms

untyped,  
unattributed  
graphs

LDBC SNB

Graph500

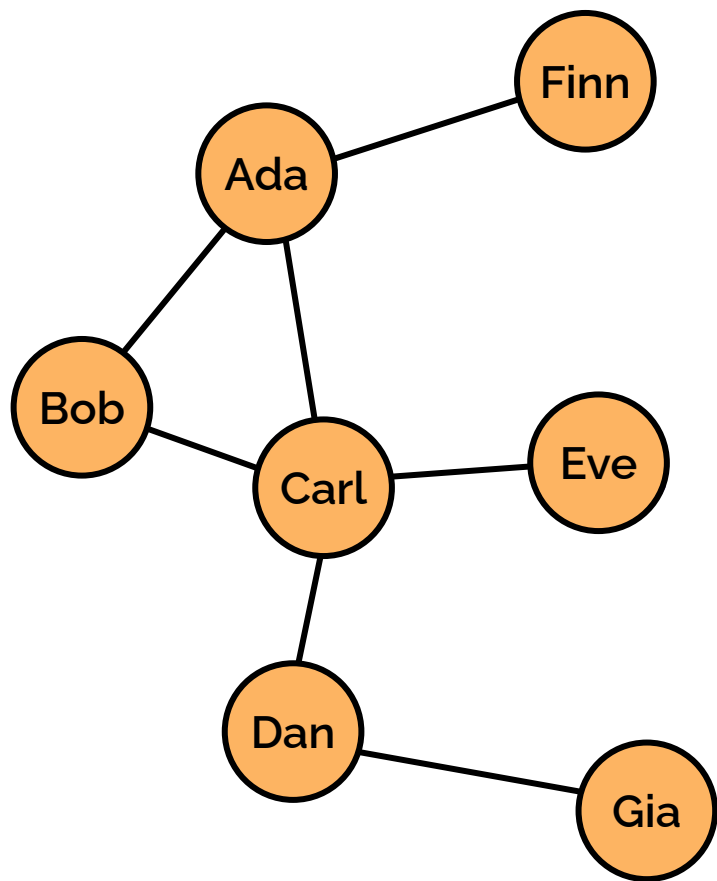
Twitter

Friendster

Patents

wiki-Talk

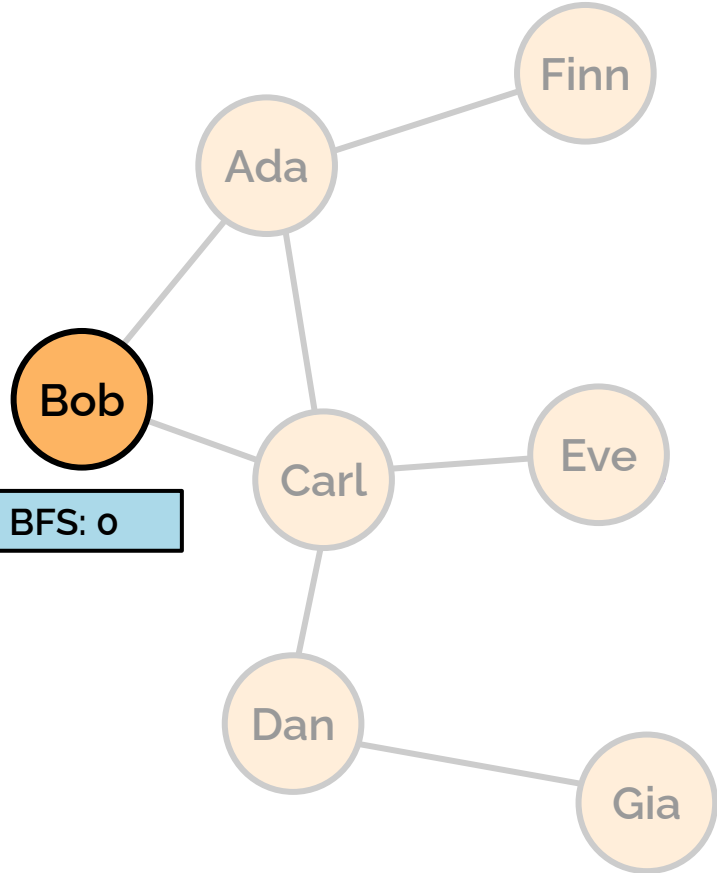
Data set



Algorithms

Graphalytics algorithms

Data set

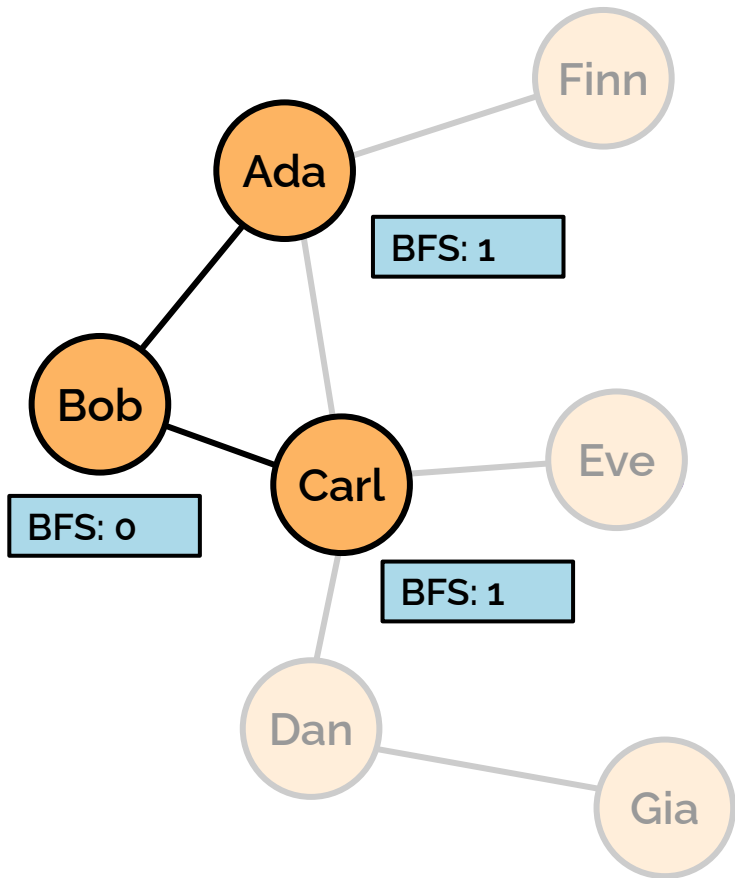


Algorithms

Graphalytics algorithms

Breadth-first search(*source*: "Bob")

Data set

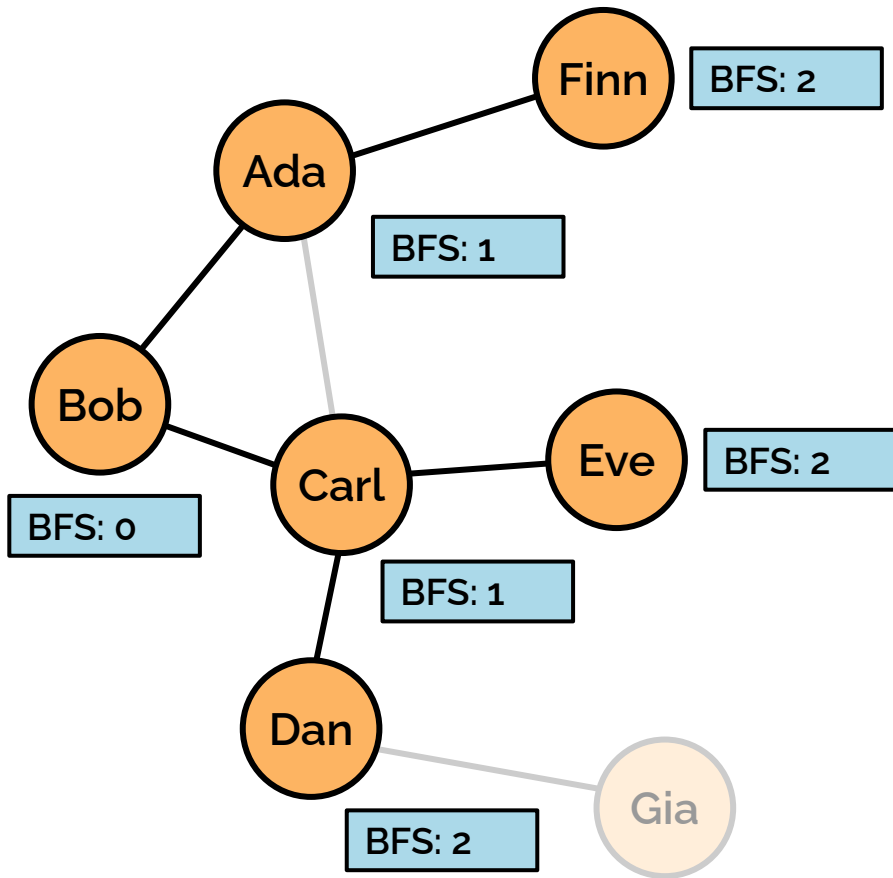


Algorithms

Graphalytics algorithms

Breadth-first search(*source*: "Bob")

Data set

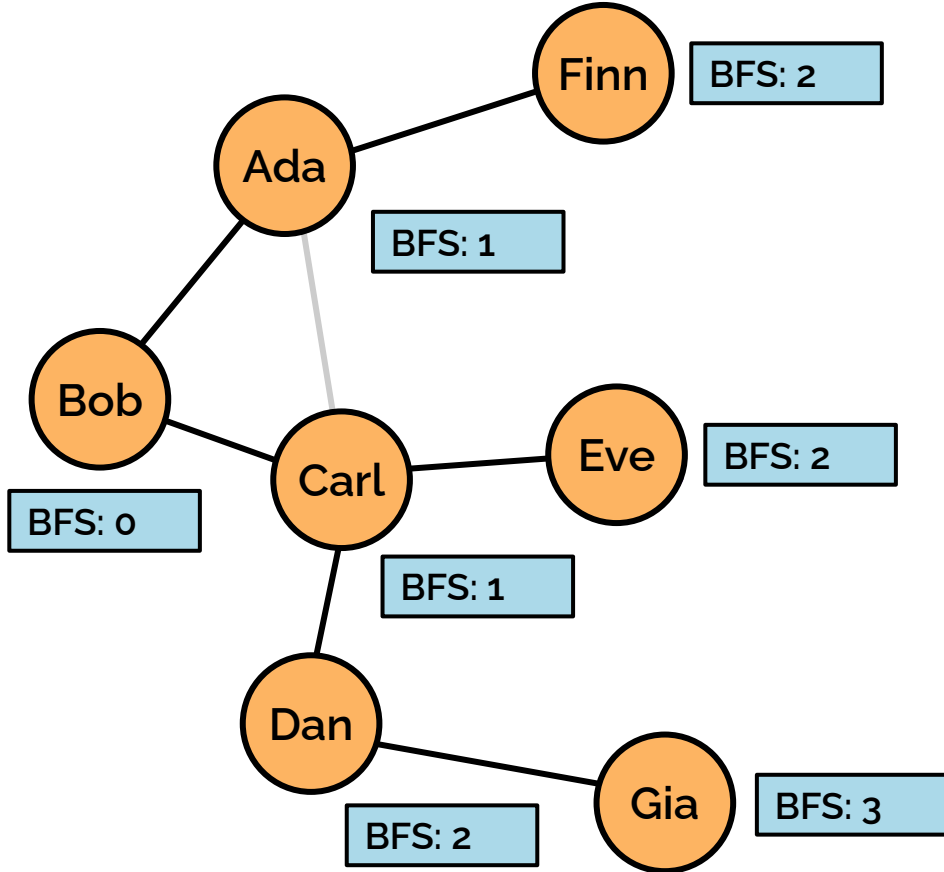


Algorithms

## Graphalytics algorithms

**Breadth-first search**(*source*: "Bob")

Data set



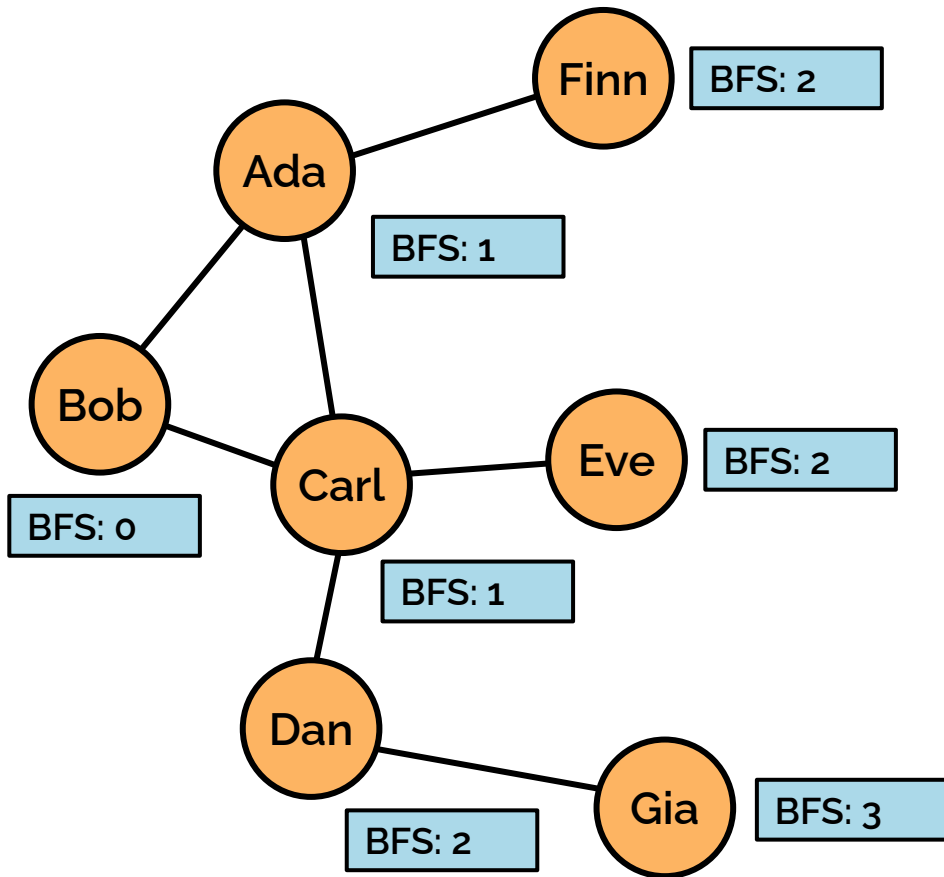
Algorithms

Graphalytics algorithms

Breadth-first search(*source*: "Bob")



Data set



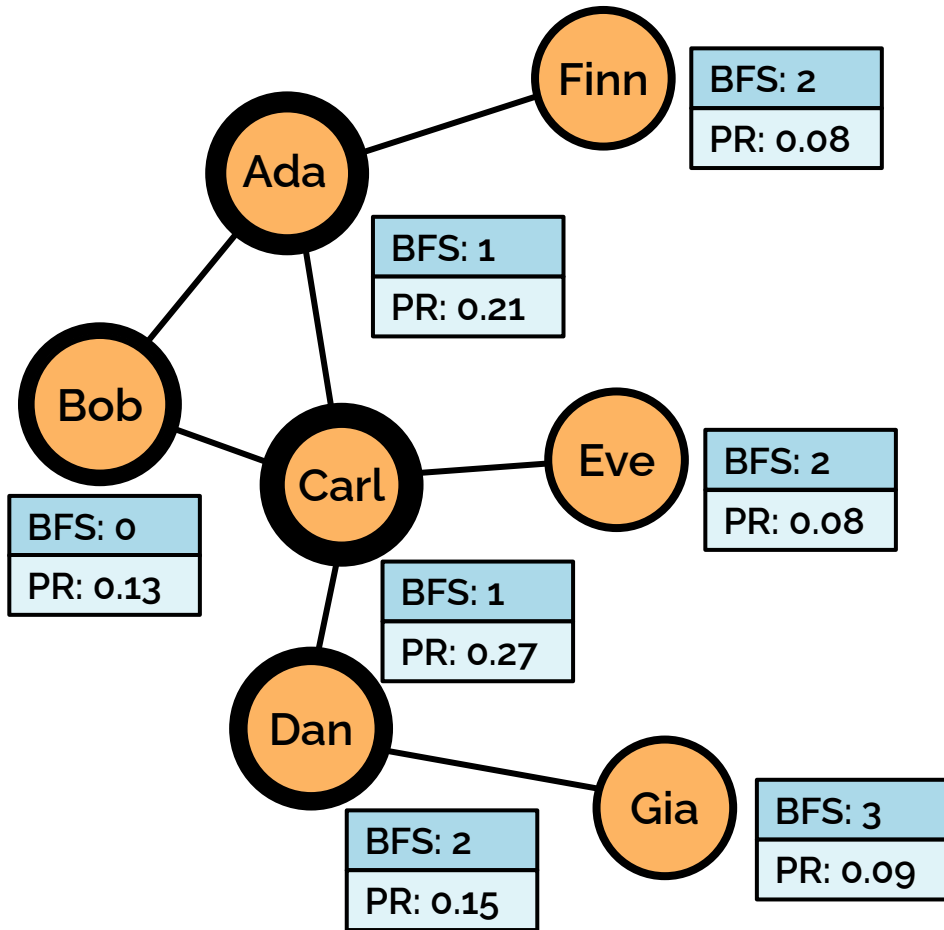
Algorithms

### Graphalytics algorithms

Breadth-first search(*source*: "Bob")

**PageRank(*damping factor*: 0.85, *iterations*: 5)**

Data set



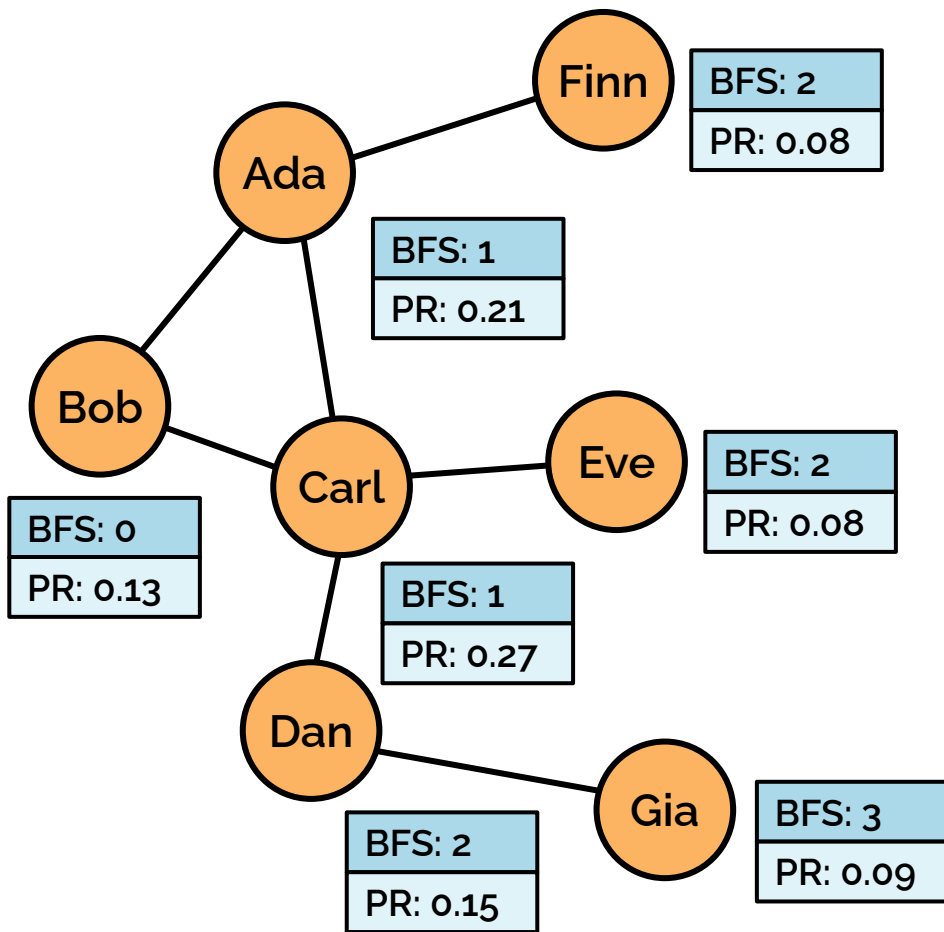
Algorithms

### Graphalytics algorithms

Breadth-first search(*source*: "Bob")

**PageRank(*damping factor*: 0.85, *iterations*: 5)**

Data set



Algorithms

### Graphalytics algorithms

Breadth-first search(*source*: "Bob")

PageRank(*damping factor*: 0.85, *iterations*: 5)

Clustering coefficient

Community detection

Connected components

Shortest paths

Graphalytics spring 2023 competition  
– please reach out if interested

# Wrapping Up...



# Joining LDBC

Members can:

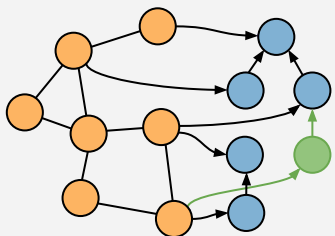
- Participate in benchmark design & research
- Commission audits
- Gain early access to ISO standard drafts, SQL/PGQ and GQL

Pricing:

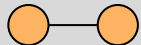
- Free for individuals
- 2,500 EUR/year for companies
- 10,000 EUR/year for sponsor companies

Visit our website at [ldbcouncil.org](https://ldbcouncil.org) and reach out at [info@ldbcouncil.org](mailto:info@ldbcouncil.org)

## SNB Interactive



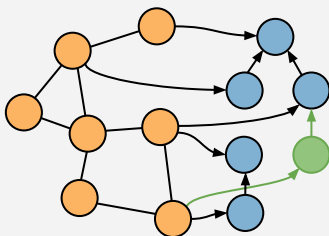
Q9(\$name, \$day)



name =  
\$name

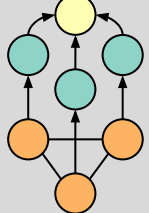
creation date <  
\$day

## SNB Business Intelligence

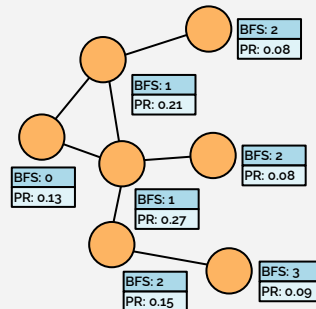


Q11(\$country)

name = \$country



## Graphalytics



Algorithms

BFS	CDLP
PR	SSSP
LCC	WCC

Data sets

LDBC SNB
Graph500
Twitter
Friendster
Patents
wiki-Talk

## Semantic Publishing Benchmark

Target: RDF/SPARQL

Domain: Media/publishing industry

Inferencing & continuous updates

## Financial Benchmark (to be released in 2023)

Target: Distributed systems

Domain: Financial fraud detection

Strict latency bound (20 ms)

## Future benchmark ideas

GNNs

Graph mining

Graph streaming

***LDBC*** 

*The graph & RDF  
benchmark reference*