

SPARQL with Property Paths

Egor V. Kostylev, **Juan L. Reutter**,
Miguel Romero, Domagoj Vrgoc

Oxford - PUC Chile - U. of Chile

Center for **Semantic Web** Research



Property Paths (PP):

Compute paths conforming to a certain regular expression

$\{?x \ a/b^* \ ?y\}$:

$?x$ and $?y$ such that there is a path from $?x$ to $?y$ using a/b^*

Property Paths (PP):

Compute paths conforming to a certain regular expression

Goal: extend foundations for SPARQL with PP's

Property Paths (PP):

Compute paths conforming to a certain regular expression

Goal: extend foundations for SPARQL with PP's

Idea: Look at graph databases!

- In graph databases property paths are called
Regular Path Queries
- Lots and lots of research, coming back from the 90s.

Property Paths (PP):

Compute paths conforming to a certain regular expression

Goal: extend foundations for SPARQL with PP's

Question:

Can we use graph DBs techniques to study Property Paths?

Property Paths (PP):

Compute paths conforming to a certain regular expression

Goal: extend foundations for SPARQL with PP's

Question:

Can we use graph DBs techniques to study Property Paths?

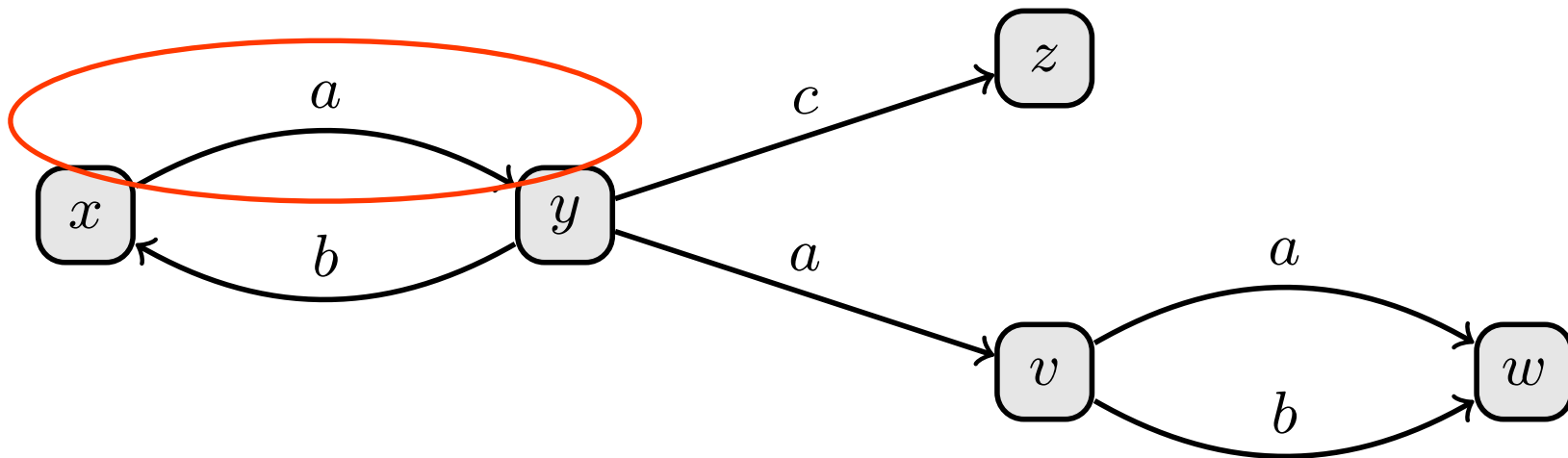
(**Answer:** Yes and no.

Tools were helpful, but more development was needed)

Outline

- Property Paths
- Evaluation
- Containment

Property Paths

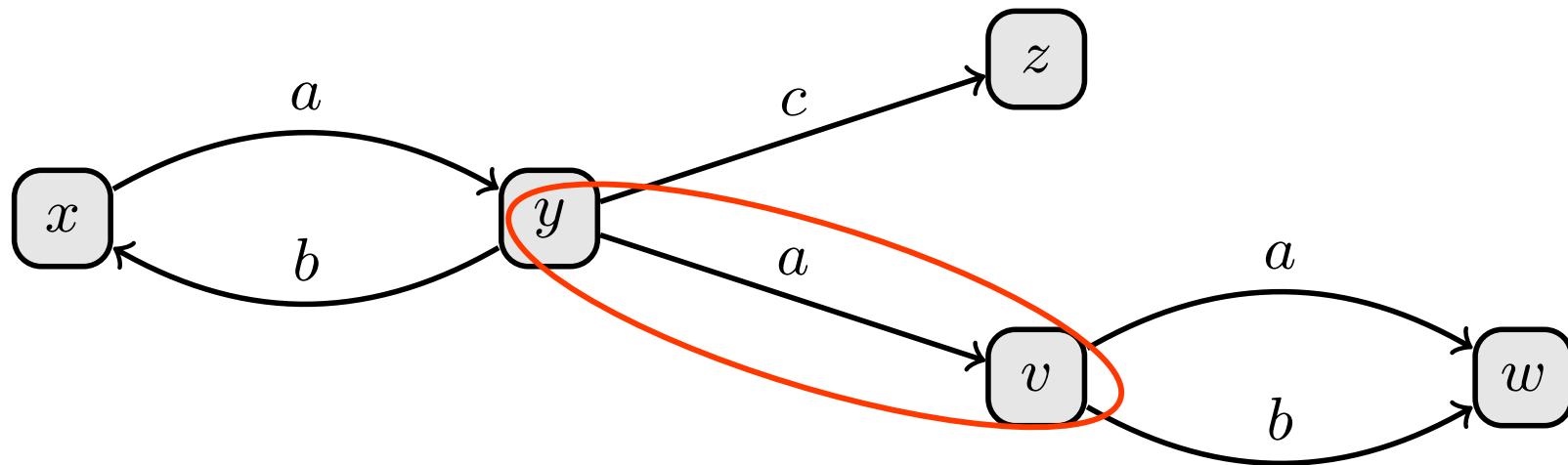


- Just an IRI:

$\{ ?s \ a \ ?o \}$

?s	?o
x	y
y	v
a	w

Property Paths

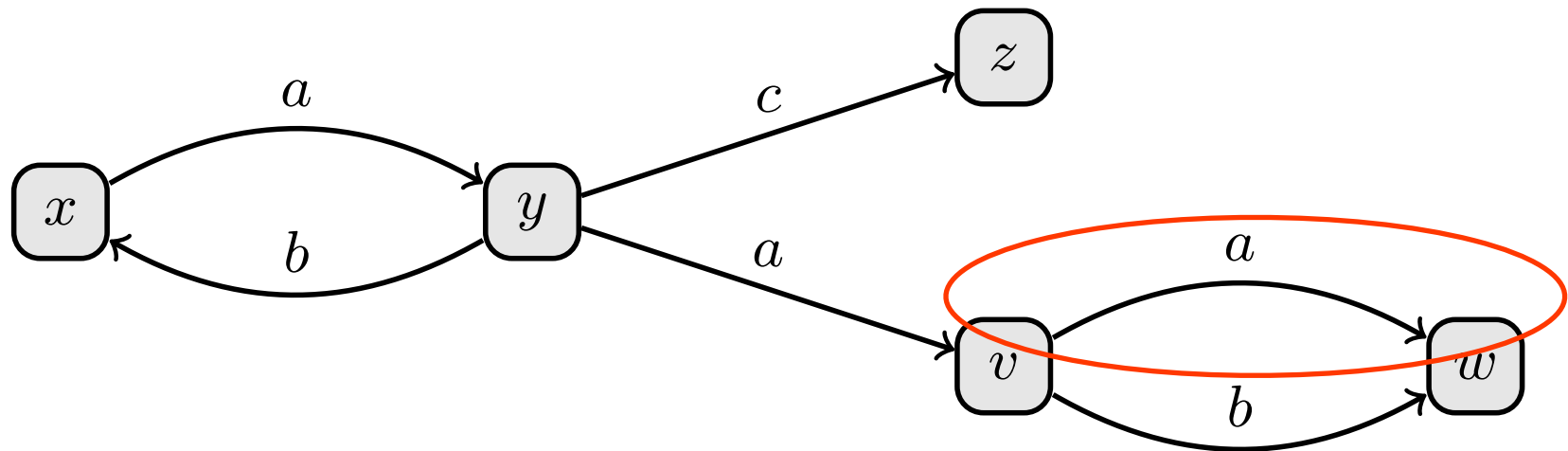


- Just an IRI:

$\{ ?s \ a \ ?o \}$

$?s$	$?o$
x	y
y	v
a	w

Property Paths

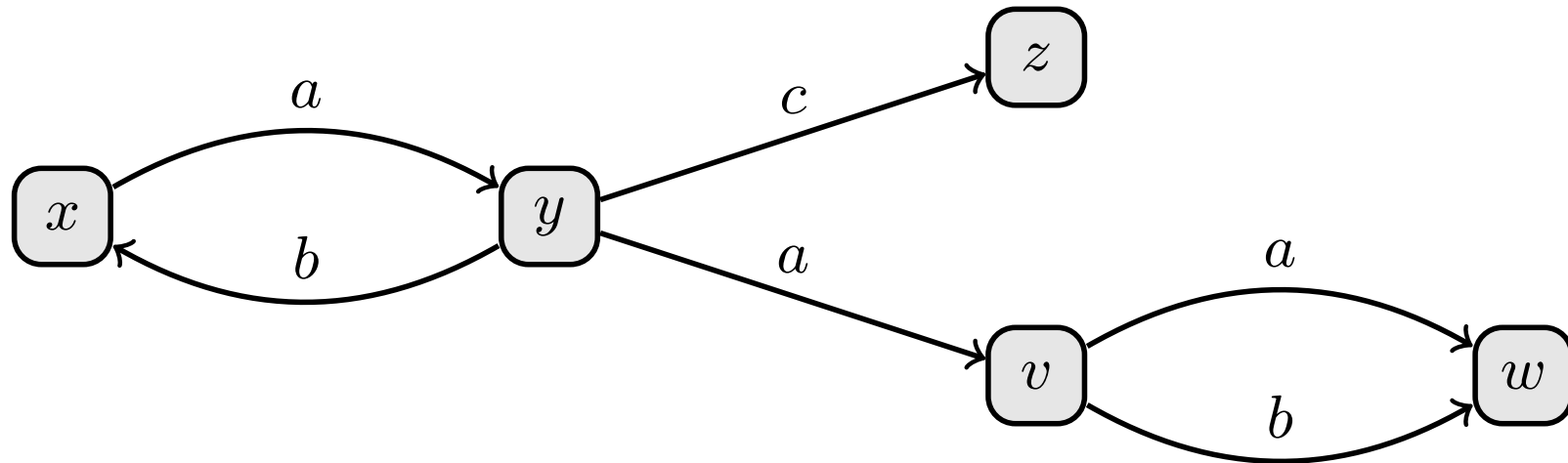


- Just an IRI:

{ ?s a ?o }

?s	?o
x	y
y	v
v	w

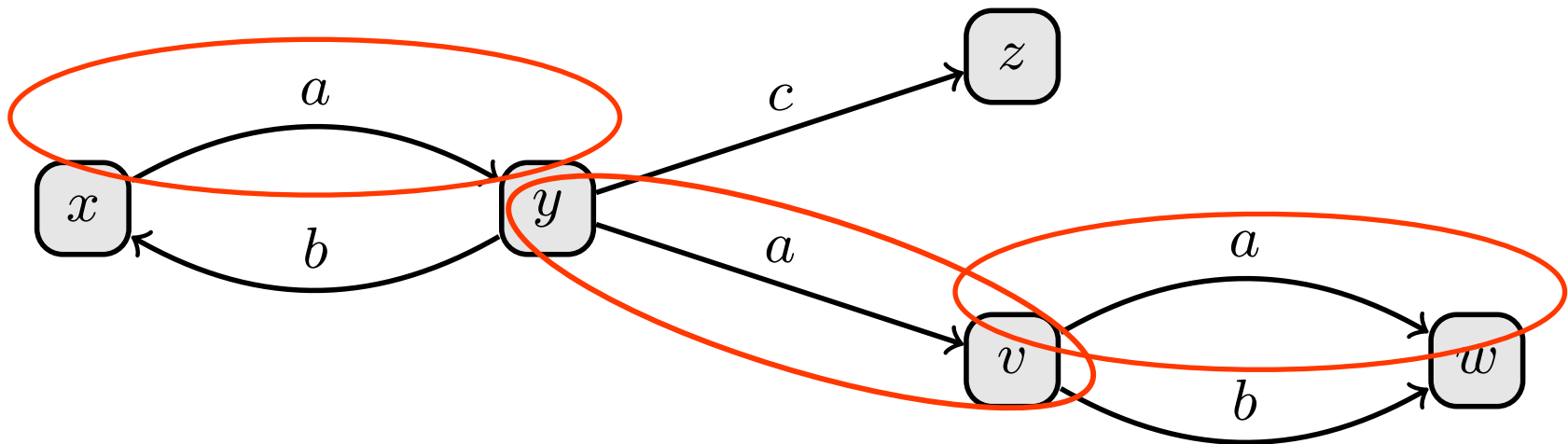
Property Paths



- Reversed IRI:

$\{ ?s \ ^a \ ?o \}$

Property Paths



- Reversed IRI:

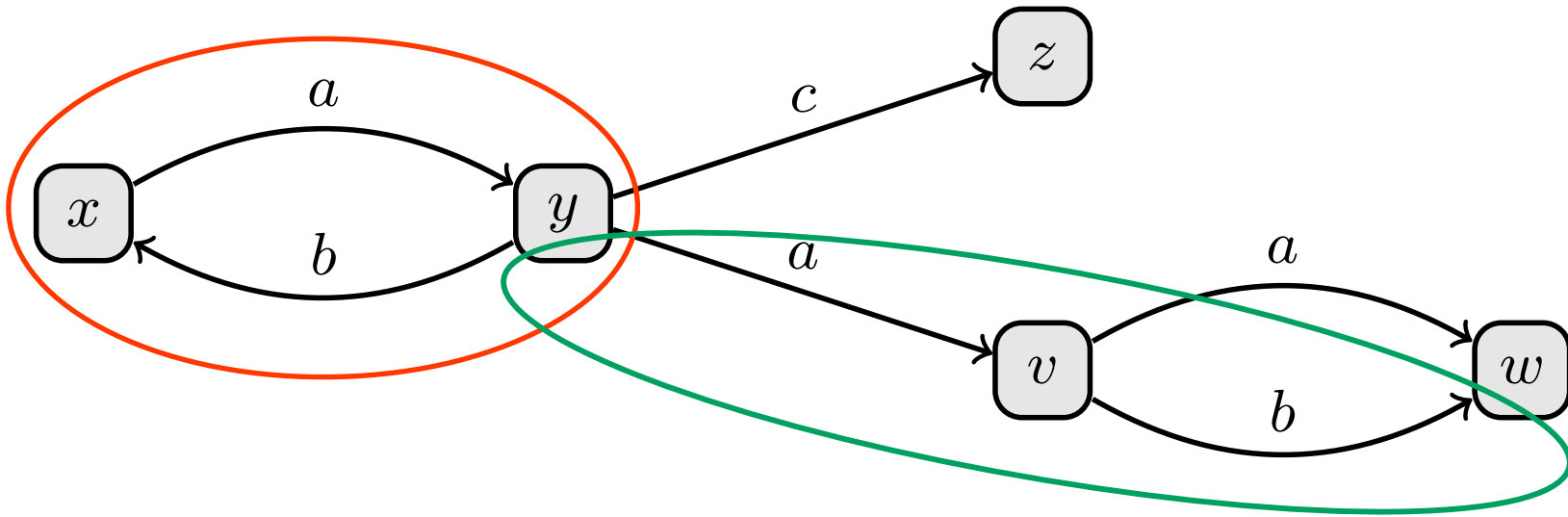
$\{ ?s \ ^a \ ?o \}$

=

$\{ ?o \ a \ ?s \}$

$?s$	$?o$
y	x
v	y
w	v

Property Paths

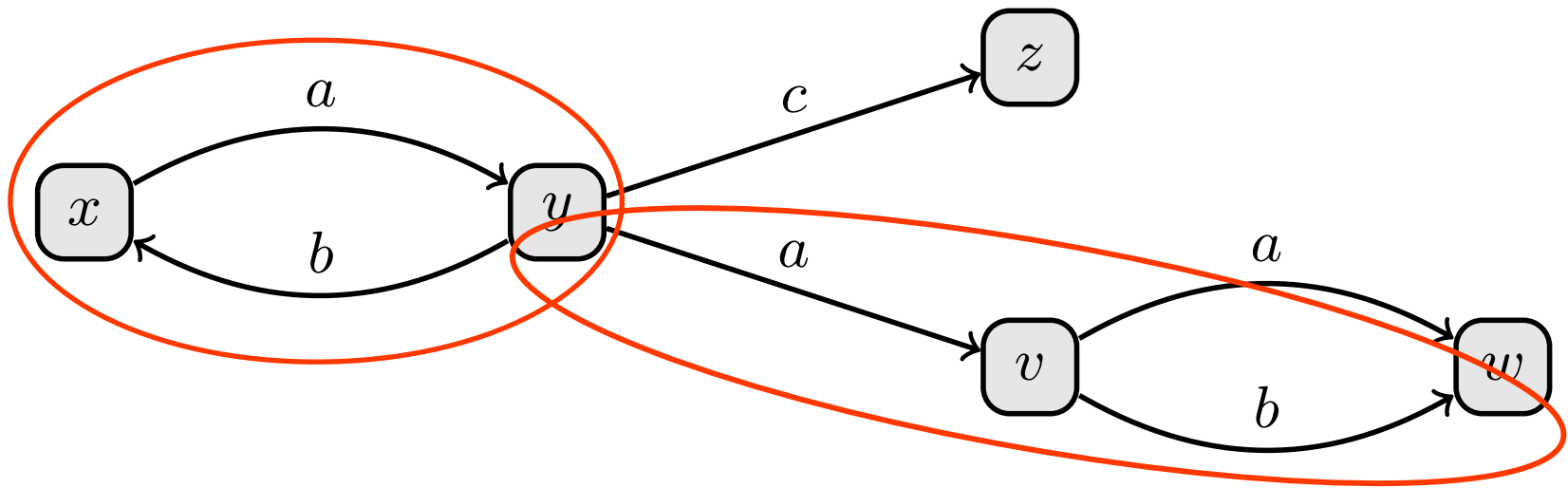


- Concatenation

$\{ ?s \ a/b \ ?o \}$

$?s$	$?o$
x	x
y	w

Property Paths

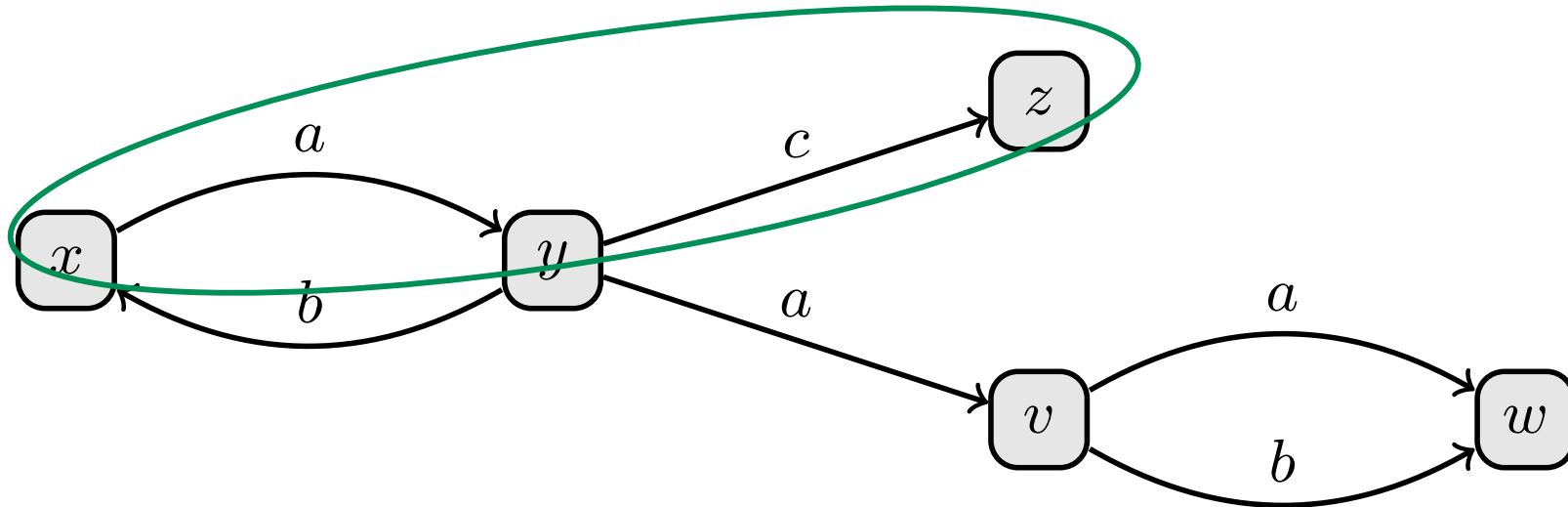


- Disjunction:

$\{ ?s \ (a/b \ | \ a/c) \ ?o \}$

$?s$	$?o$
x	x
x	z
y	w

Property Paths

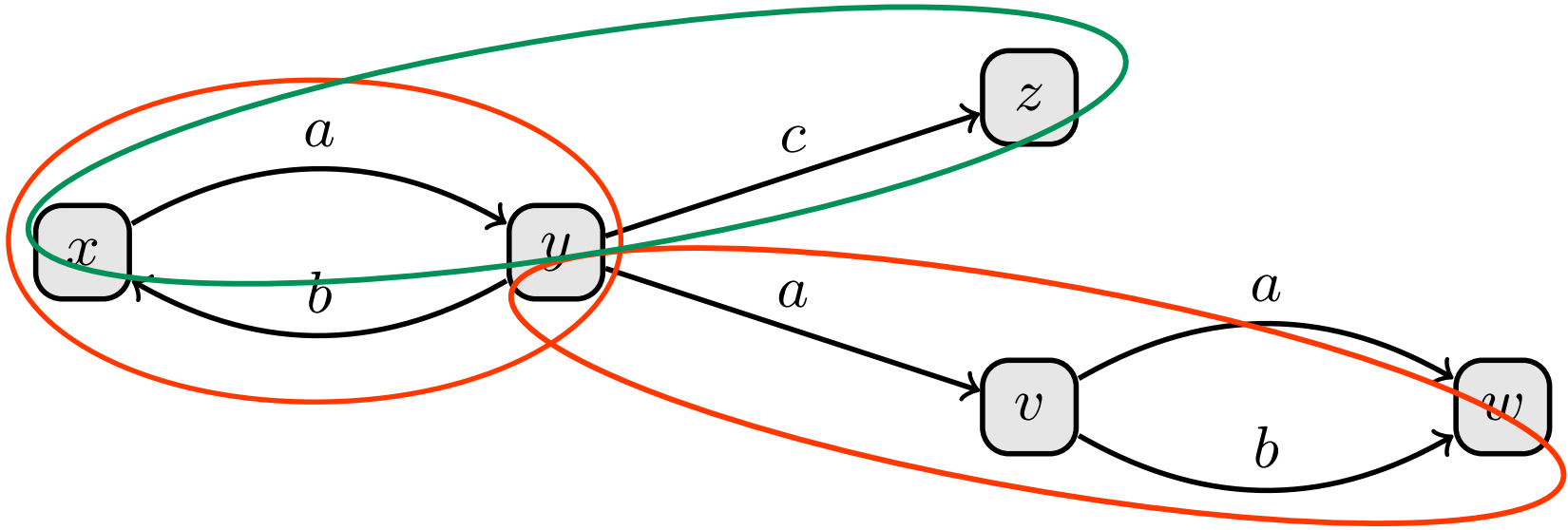


- Disjunction:

$\{ ?s \ (a/b \mid a/c) \ ?o \}$

?s	?o
x	x
x	z
y	w

Property Paths

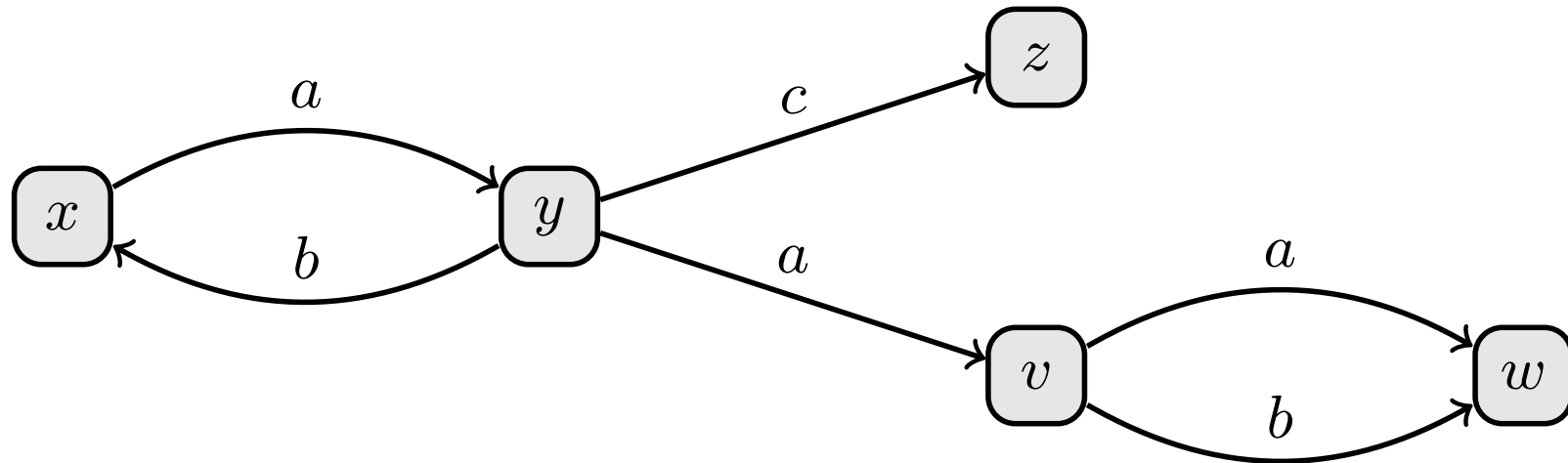


- Disjunction:

$\{ ?s \ (a/b \mid a/c) \ ?o \}$

	?s	?o
	x	x
	x	z
	y	w

Property Paths

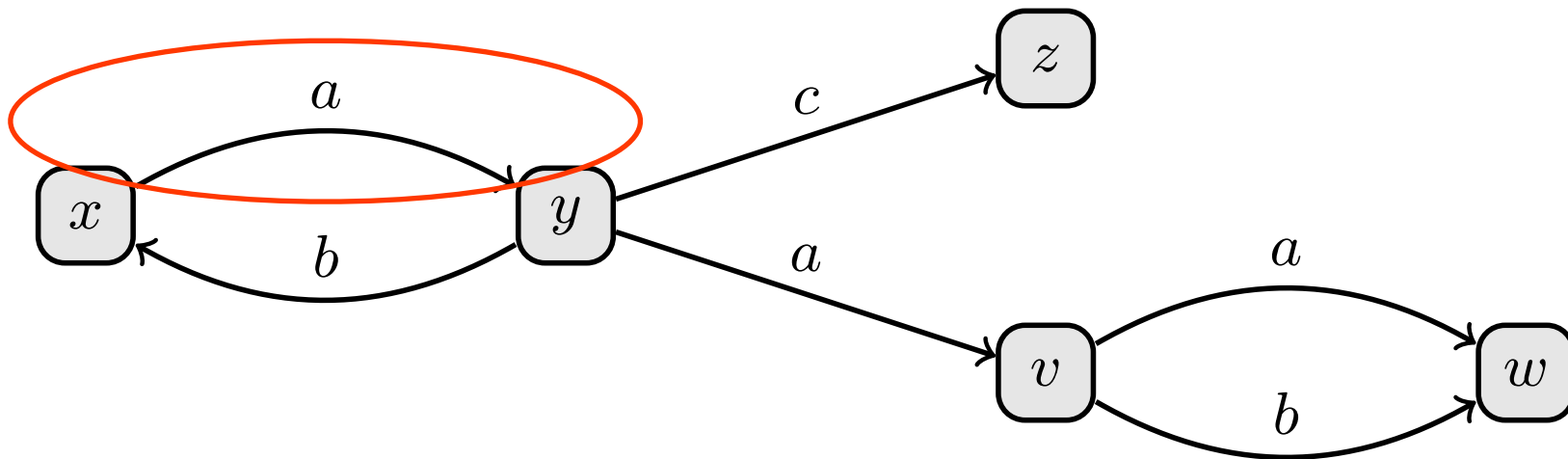


- Star:

$\{ ?s \ a^* \ ?o \}$

$?s$	$?o$
x	x
y	y
z	z
v	v
w	w

Property Paths



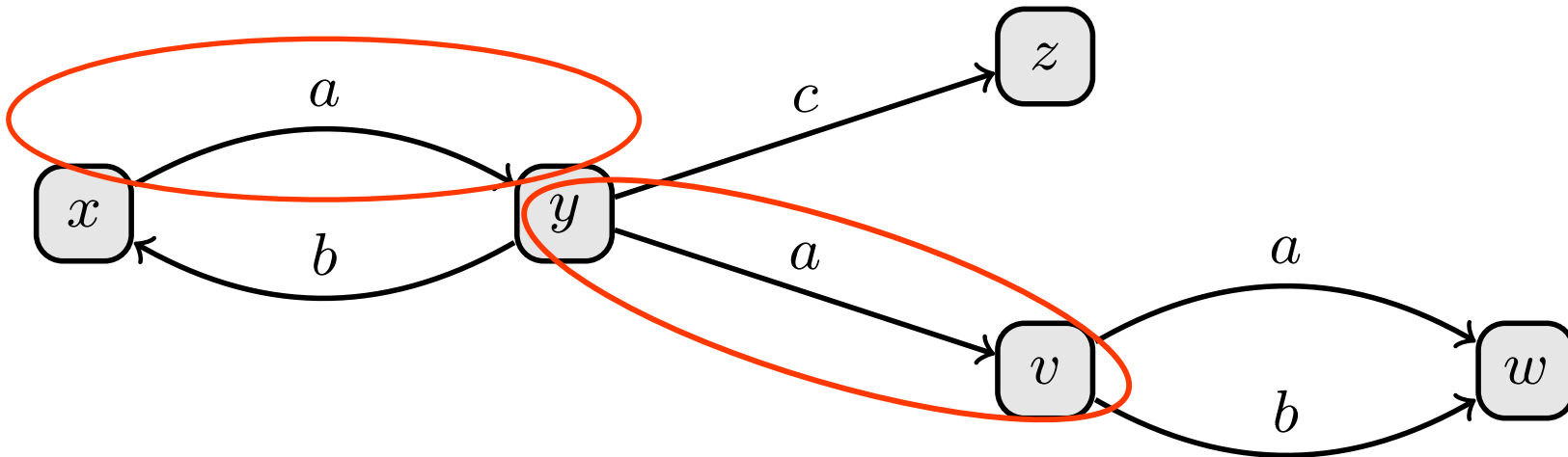
- Star:

$\{ ?s \ a^* \ ?o \}$

?s	?o
x	y
x	v
x	w
y	v
y	w
v	w

?s	?o
x	x
y	y
z	z
v	v
w	w

Property Paths



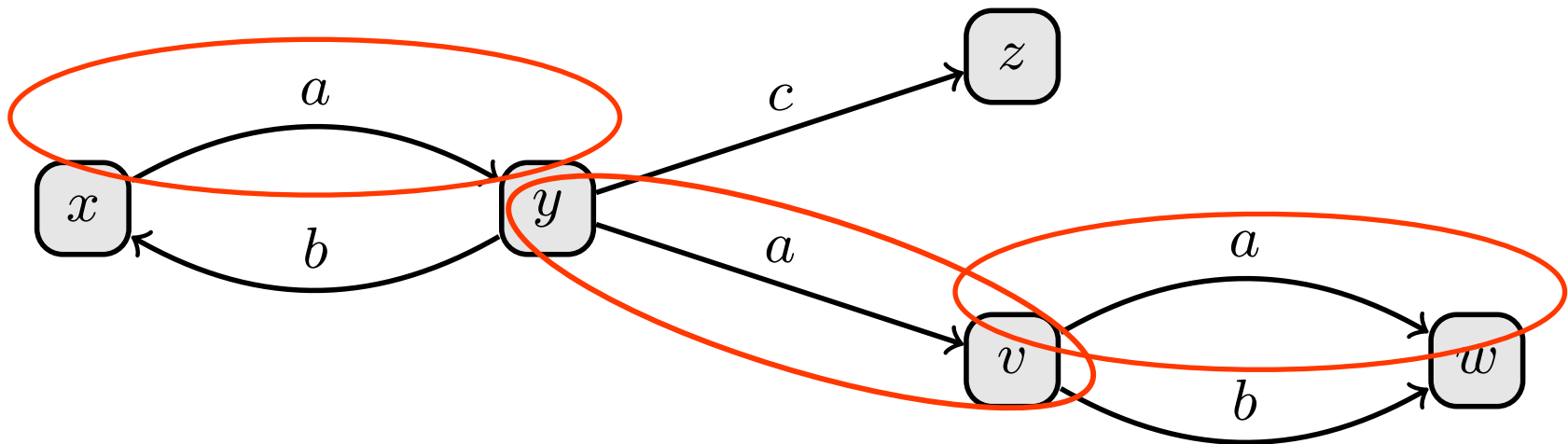
- Star:

$\{ ?s \ a^* \ ?o \}$

?s	?o
x	y
x	v
x	w
y	v
y	w
v	w

?s	?o
x	x
y	y
z	z
v	v
w	w

Property Paths



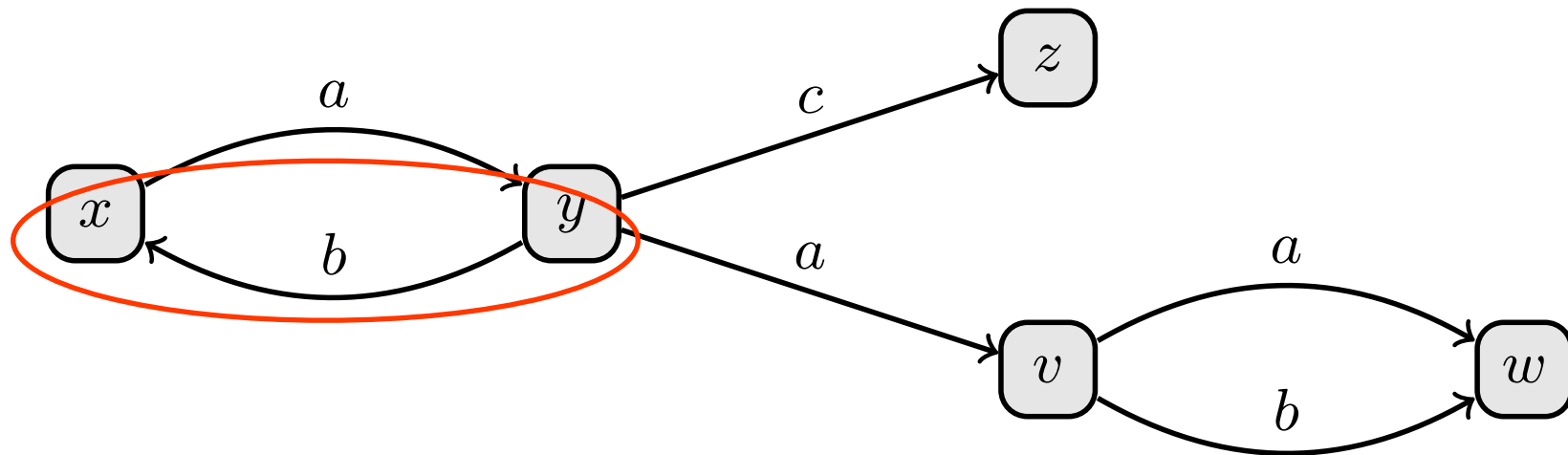
- Star:

$\{ ?s \ a^* \ ?o \}$

?s	?o
x	y
x	v
x	w
y	v
y	w
v	w

?s	?o
x	x
y	y
z	z
v	v
w	w

Property Paths

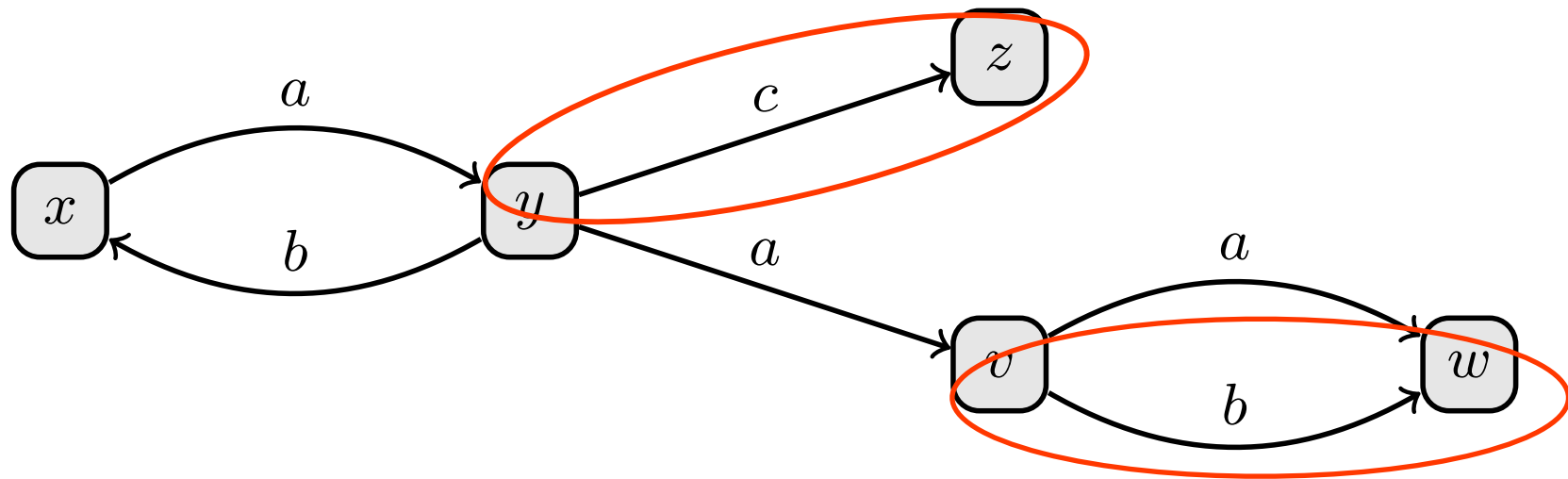


- Negated set:

$\{ ?s \neg(a) ?o \}$

$?s$	$?o$
y	x
y	z
v	w

Property Paths

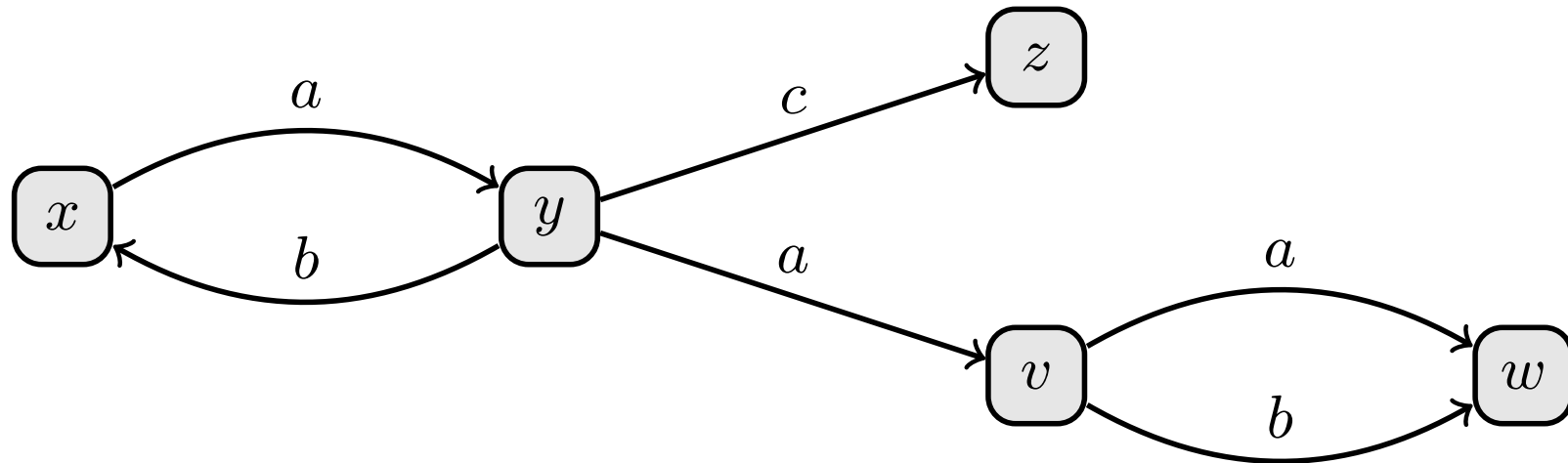


- Negated set:

$\{ ?s \neg(a) ?o \}$

?s	?o
y	x
y	z
v	w

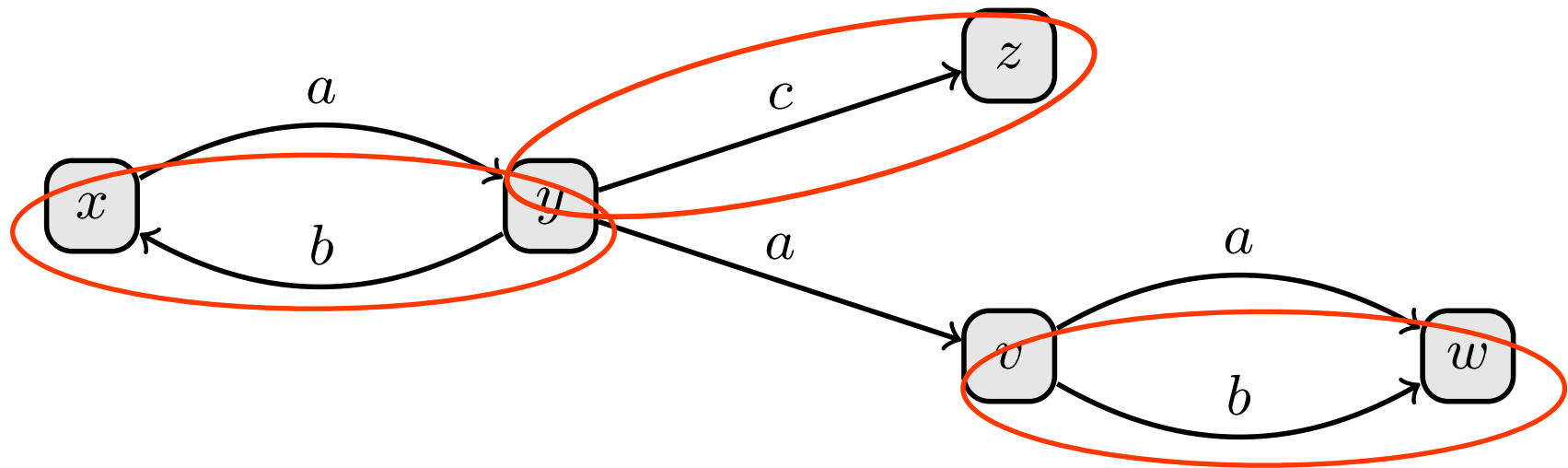
Property Paths



- Negated inverse set:

$\{ ?s \ !(^a) ?o \}$

Property Paths



- Negated inverse set:

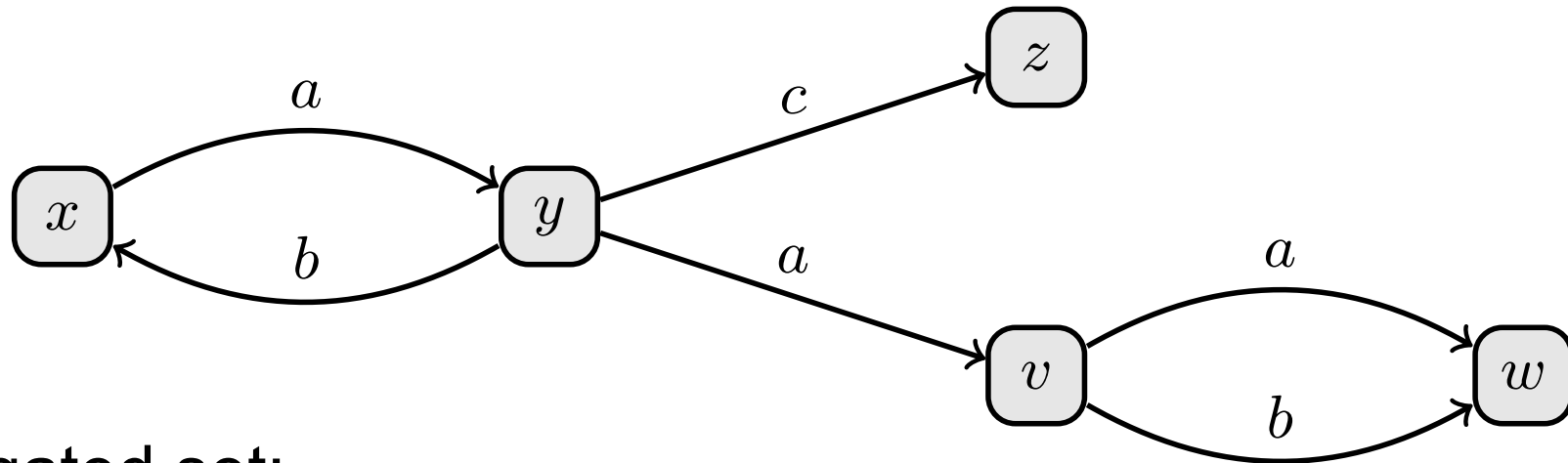
$\{ ?s \neg(^a) ?o \}$

=

$\{ ?o \neg(a) ?s \}$

?s	?o
x	y
z	y
w	v

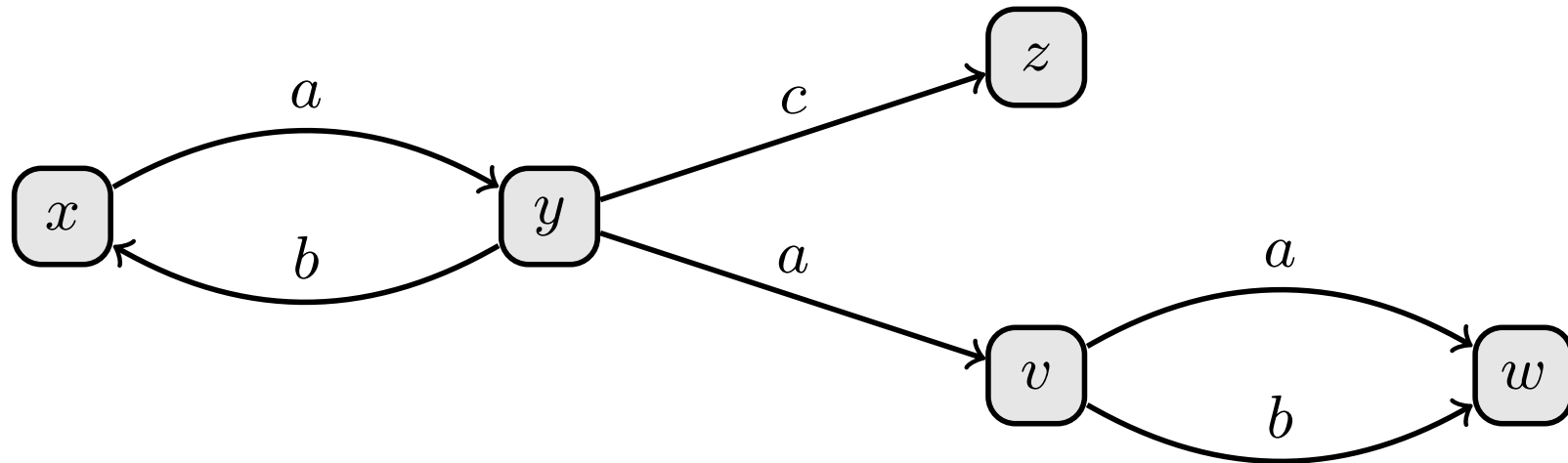
Property Paths



- Negated set:

$\{ ?s \ !\{a,b\} ?o \}$

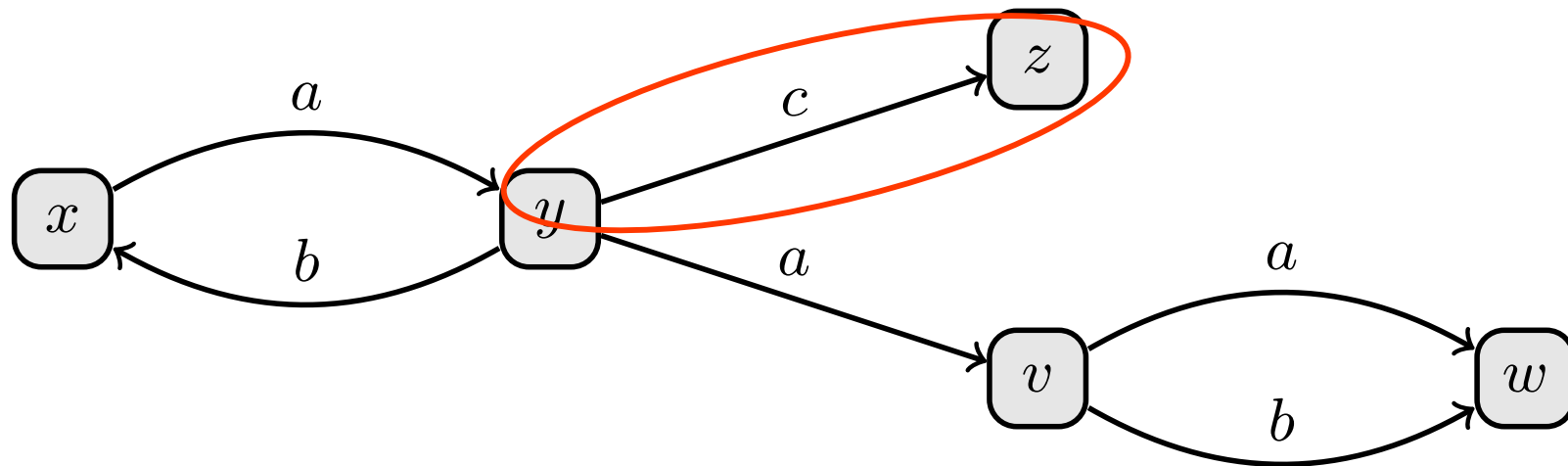
Property Paths



$\{ ?s \{a,b\} ?o \}$ \neq

$\{ ?s (a) | (b) ?o \}$

Property Paths



$\{ ?s \ !\{a,b\} \ ?o \}$

\neq

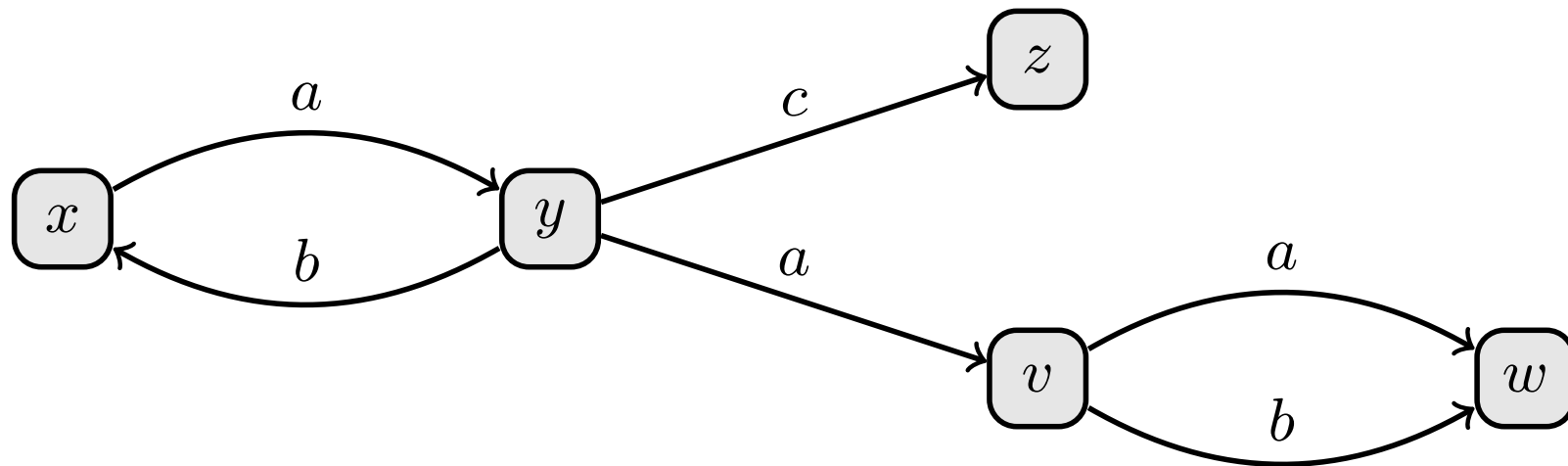
$\{ ?s \ !(a) \ | \ !(b) \ ?o \}$

$?s$	$?o$
------	------

y

z

Property Paths



$\{ ?s \ !\{a,b\} \ ?o \}$

\neq

$\{ ?s \ !(a) \ | \ !(b) \ ?o \}$

?s	?o
----	----

y z

\neq

?s	?o
----	----

y x

y z

v w

+

?s	?o
----	----

x y

y z

v w

Can we use graph DBs techniques
to study Property Paths?

Outline

- Property Paths
- Evaluation
- Containment

Graph DB techniques: query evaluation

Many different techniques, fast BFS algorithms

Several graph DBMS available

Graph DB techniques: query evaluation

Many different techniques, fast BFS algorithms

Several graph DBMS available

Ideally:

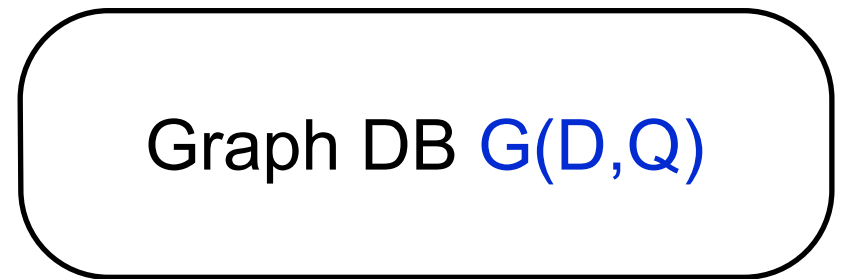
treat RDF documents as graphs and use their algorithms

Theorem:

We can always build $G(D,Q)$
such that the answers of these queries coincide:



$\{?x \text{ expr } ?y\}$
Triple with PP



$(?x, \text{expr}', ?y)$
Regular path query

Theorem:

We can always build $G(D, Q)$

such that the answers of these queries coincide:

Corollary:

can use graph tools for evaluation

(in practice: discarding/renaming what you don't need)

Outline

- Property Paths
- Evaluation
- Containment

Graph DB techniques: equivalence and containment

When are Q and Q' **equivalent** over *any possible database*?

Graph DB techniques: equivalence and containment

When are Q and Q' **equivalent** over *any possible database*?

When is the **answer to Q** **contained** (or **subsumed**)
in the **answer to Q'** over *any possible database*?

Graph DB techniques: equivalence and containment

When are Q and Q' **equivalent** over *any possible database*?

When is the **answer to Q** **contained** (or **subsumed**)
in the **answer to Q'** over *any possible database*?

- Well understood in graph DBs
- Uses in view-based query answering, entailment, query optimization, schema mappings, dependencies....

Graph DB techniques: equivalence and containment

When are Q and Q' **equivalent** over *any possible database*?

When is the **answer to Q** **contained** (or **subsumed**)
in the **answer to Q'** over *any possible database*?

Ideally:

treat RDF documents as graphs and use their algorithms

Containment: graph database?

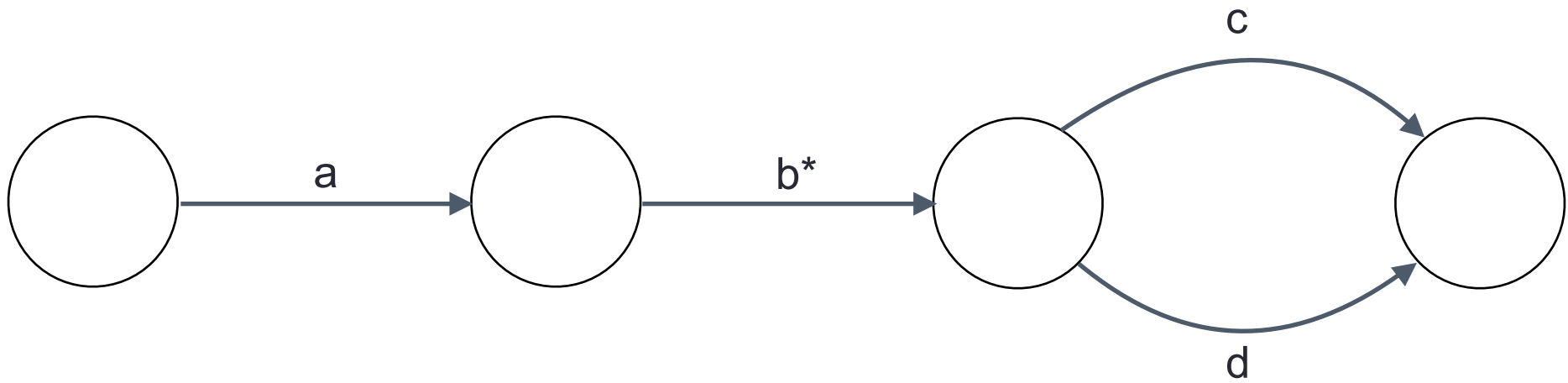
Is Q contained in Q' ?

(rough) algorithm:

- Take Q and construct a **canonical graph database**
- Evaluate Q' in Q

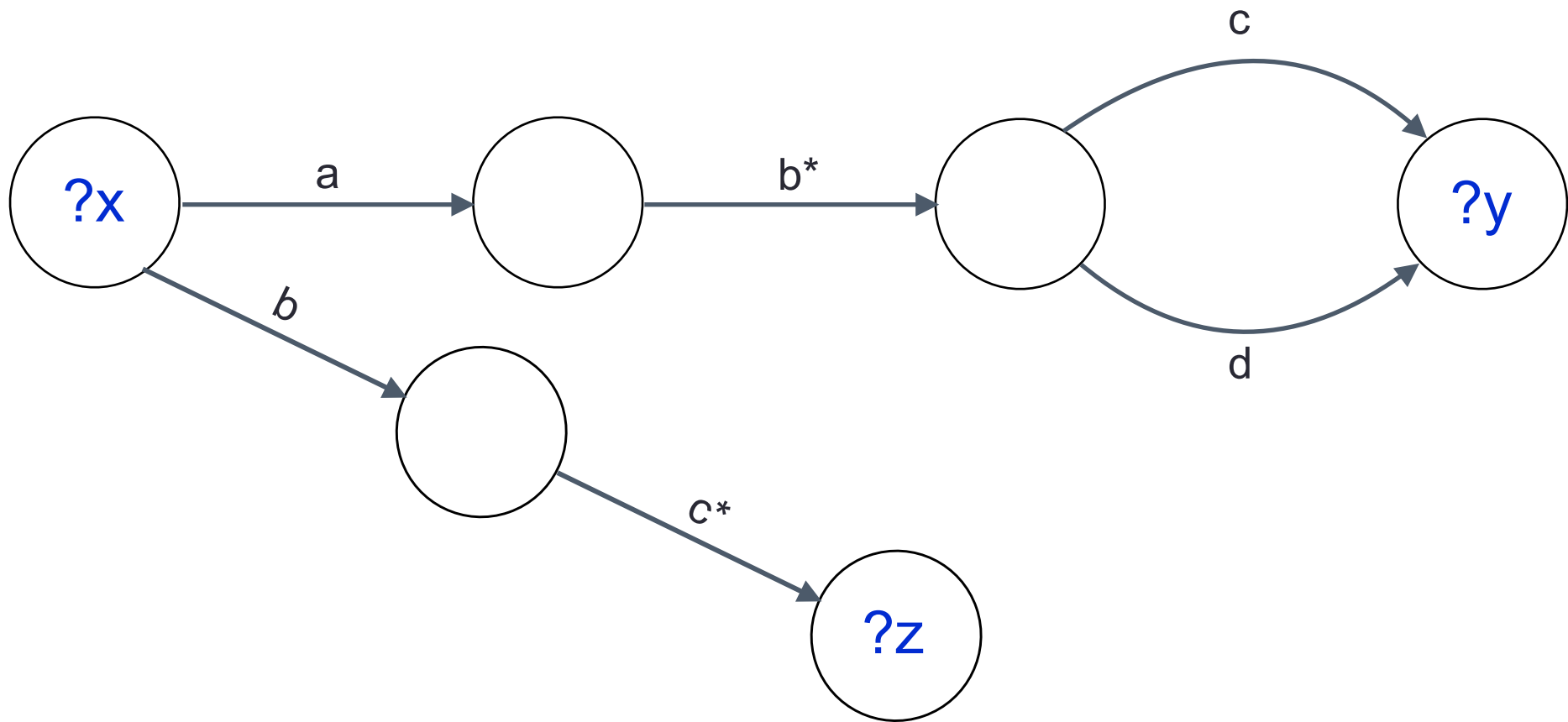
Canonical Graph Database

$a/b^*/(c \mid d)$:



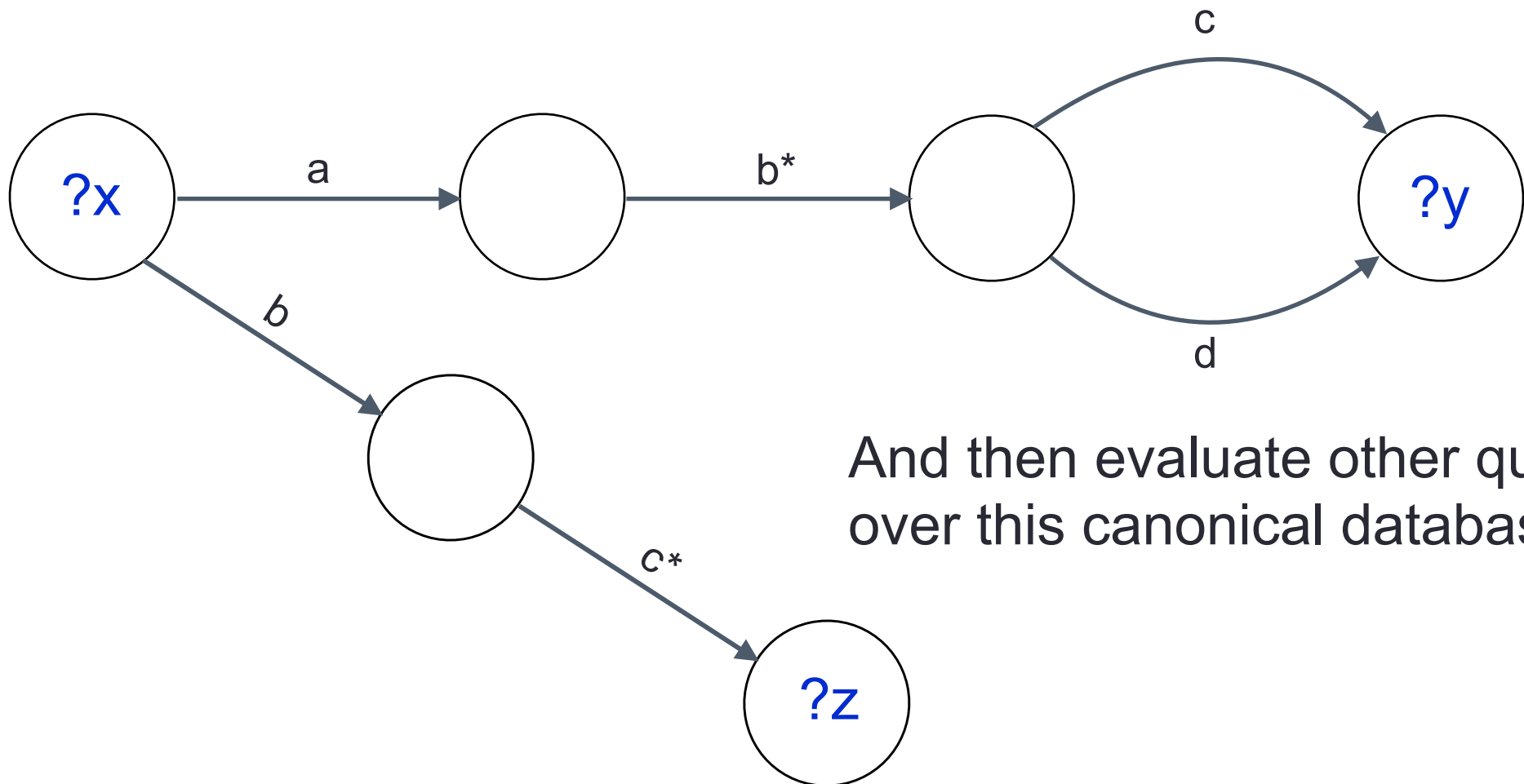
Canonical Graph Database

$(?x, a/b^*/(c \mid d), ?y)$ AND $(?x, b/c^*, ?z)$:



Canonical Graph Database

$(?x, a/b^*/(c \mid d), ?y)$ AND $(?x, b/c^*, ?z)$:



And then evaluate other queries over this canonical database

Why we can't do this for Property Paths

Problem 1

Q = SELECT ?x ?y ?z WHERE {?x !(a) ?y} . {?x !(a) ?z}

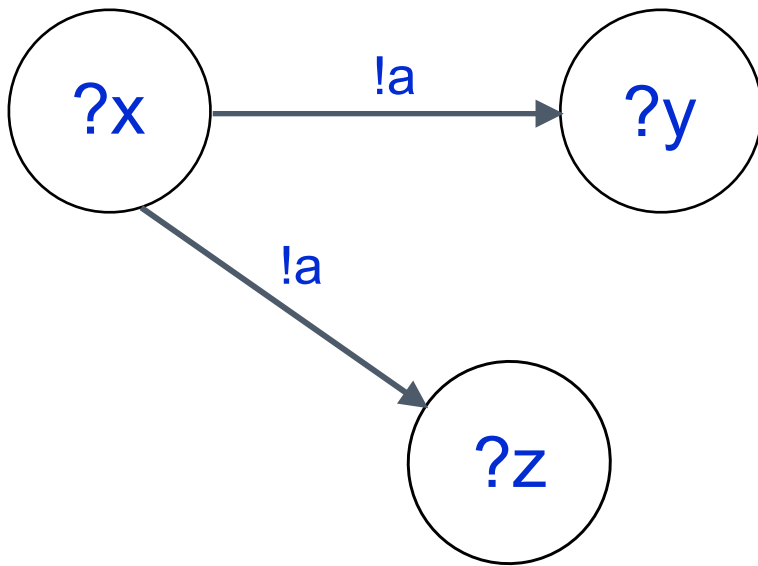
Q' = SELECT ?x ?y ?z WHERE {?x ?v ?y} . {?x ?v ?z}

- Construct canonical graph for Q? evaluate using Q'?

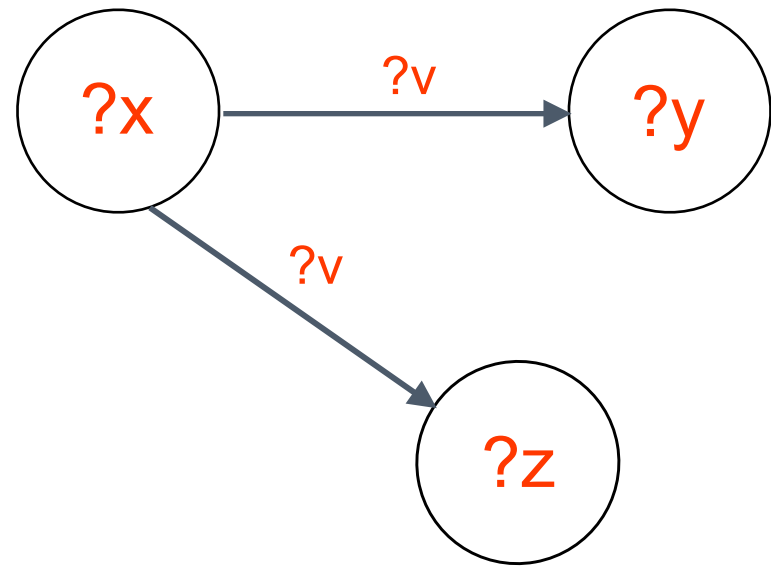
Problem 1

Q = SELECT ?x ?y ?z WHERE {?x !(a) ?y} . {?x !(a) ?z}

Q' = SELECT ?x ?y ?z WHERE {?x ?v ?y} . {?x ?v ?z}



Q

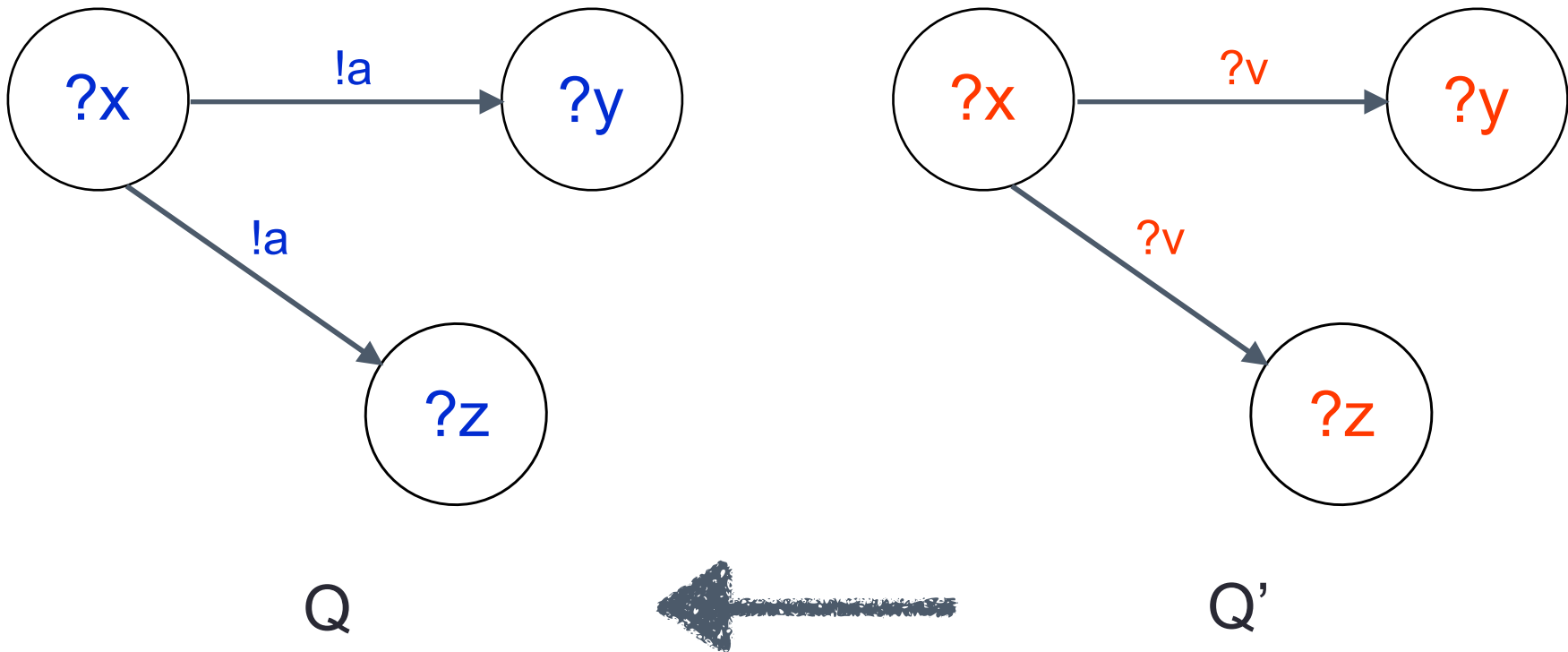


Q'

Problem 1

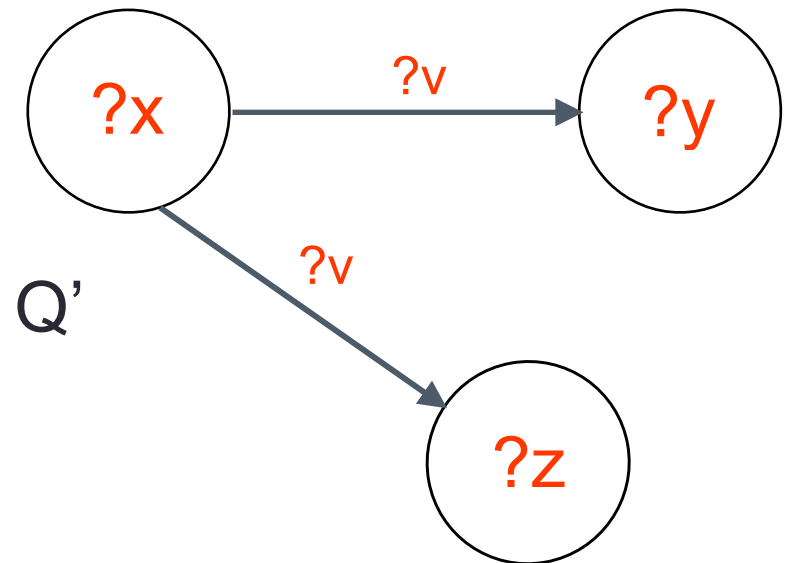
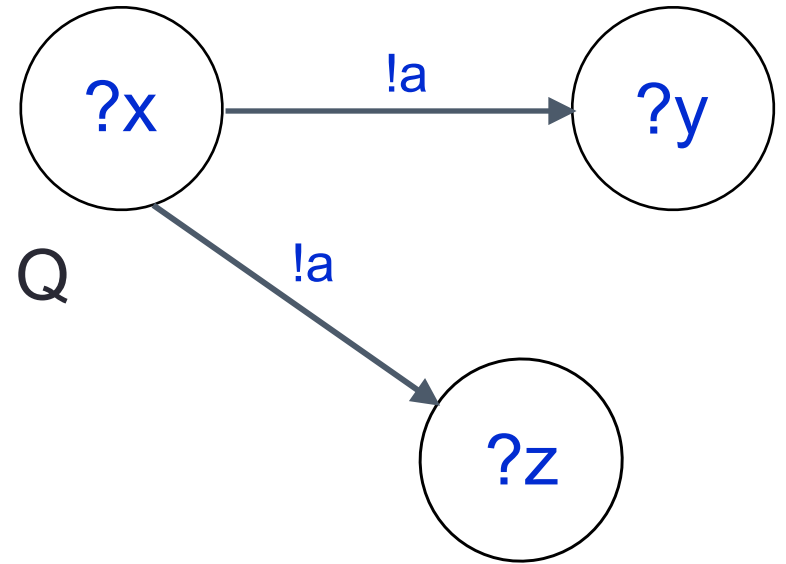
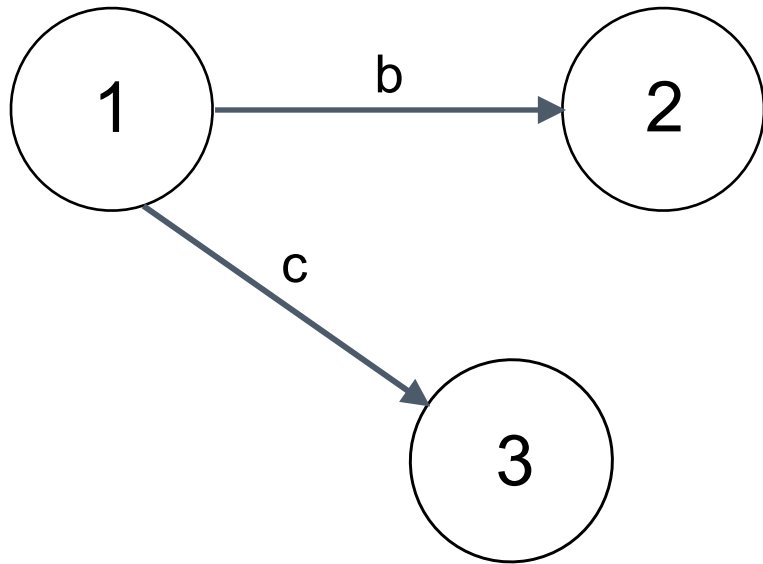
Q = SELECT ?x ?y ?z WHERE {?x !(a) ?y} . {?x !(a) ?z}

Q' = SELECT ?x ?y ?z WHERE {?x ?v ?y} . {?x ?v ?z}

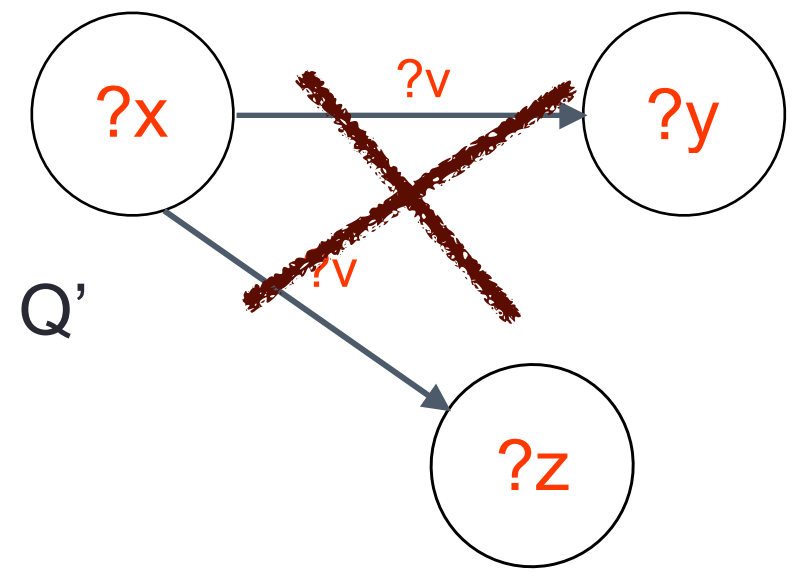
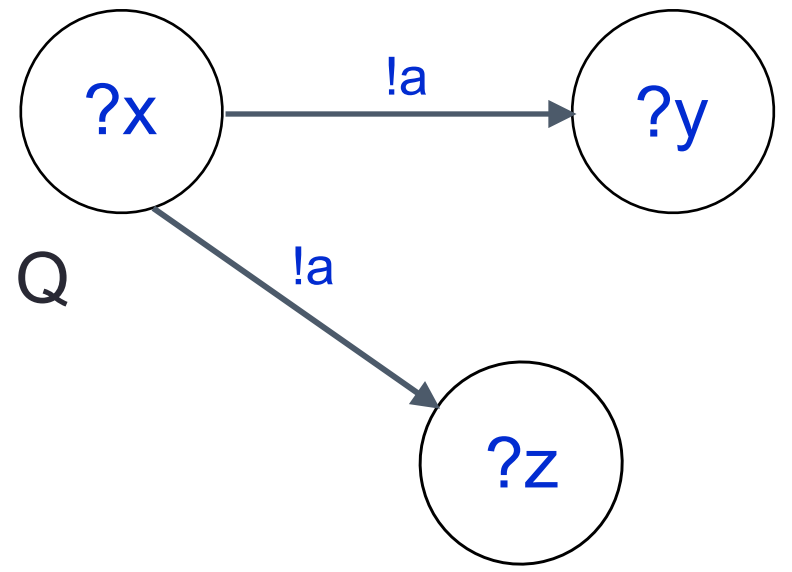
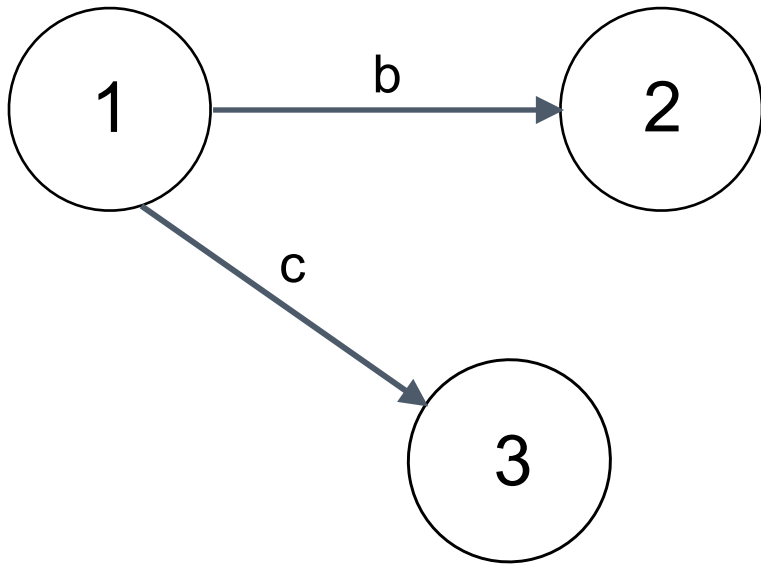


Q is contained in Q'?

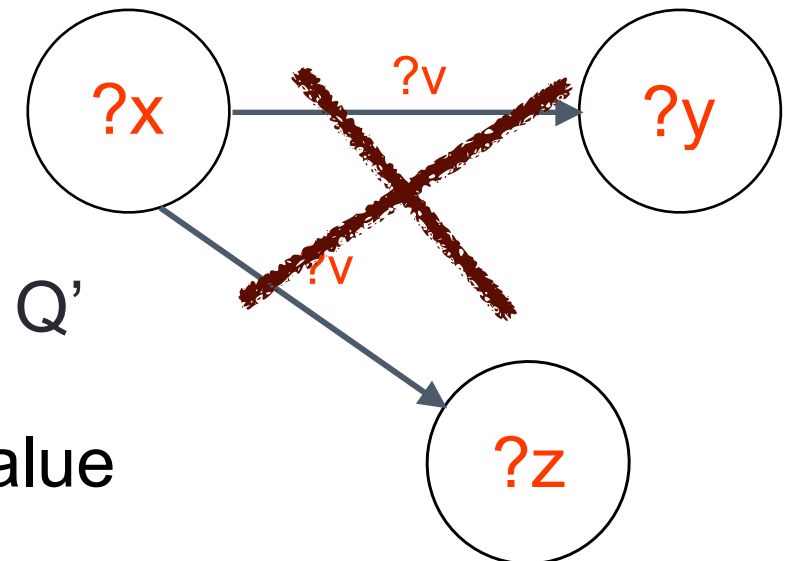
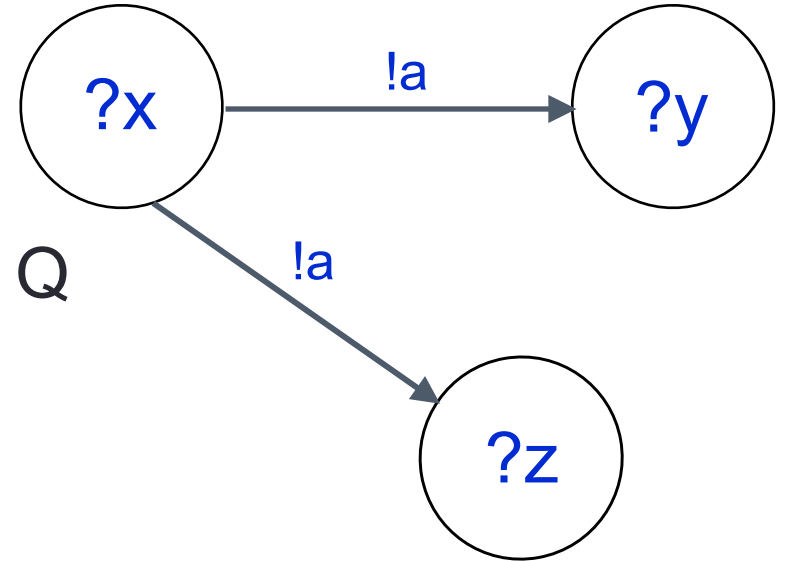
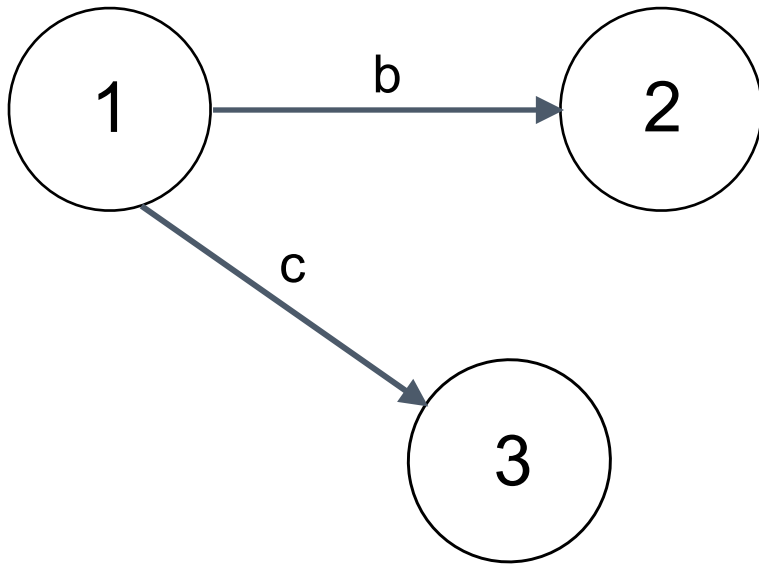
Problem 1



Problem 1



Problem 1

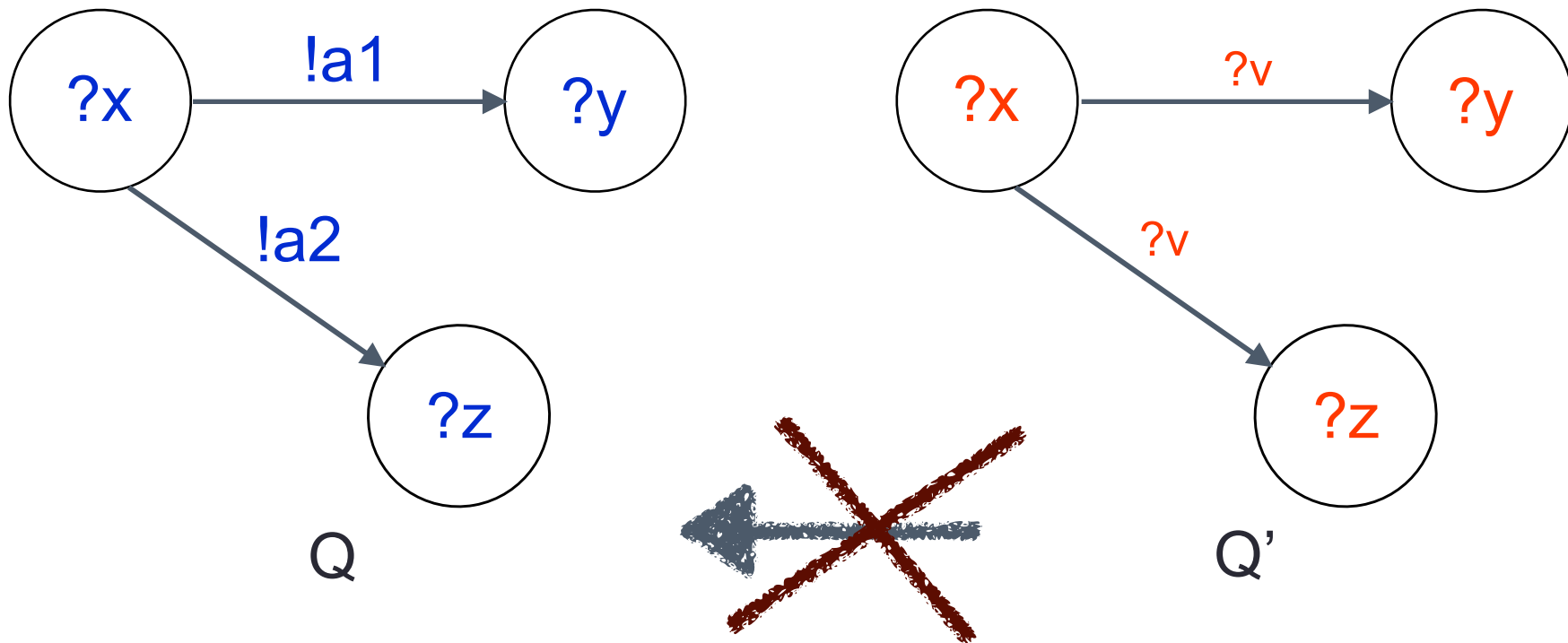


- Each !a must be replaced by fresh value

Problem 1

Q = SELECT ?x ?y ?z WHERE {?x !(a) ?y} . {?x !(a) ?z}

Q' = SELECT ?x ?y ?z WHERE {?x ?v ?y} . {?x ?v ?z}

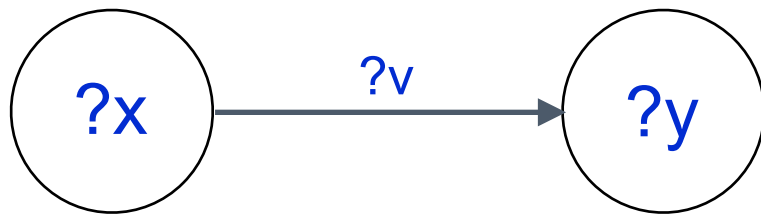


Q is not contained in Q'

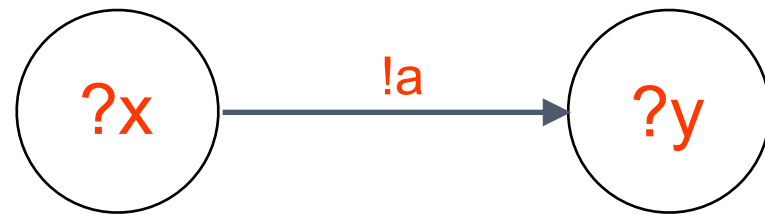
Problem 2

Q = SELECT ?x ?y WHERE {?x ?v ?y}

Q' = SELECT ?x ?y WHERE {?x !(a) ?y}



Q

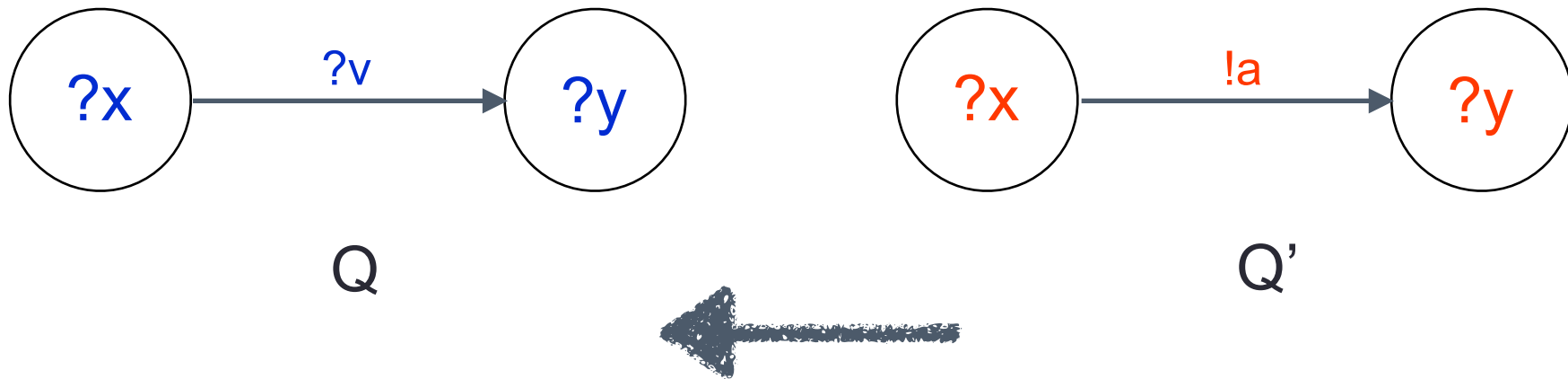


Q'

Problem 2

Q = SELECT ?x ?y WHERE {?x ?v ?y}

Q' = SELECT ?x ?y WHERE {?x !(a) ?y}

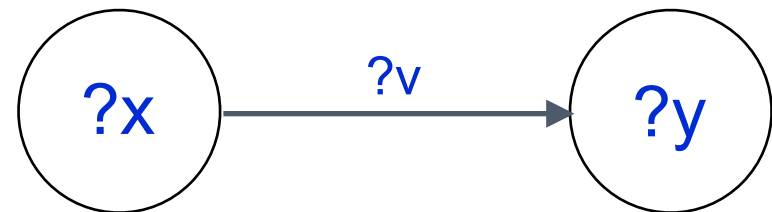
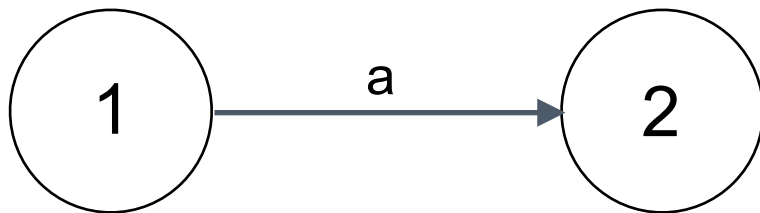


Q is contained in Q'?

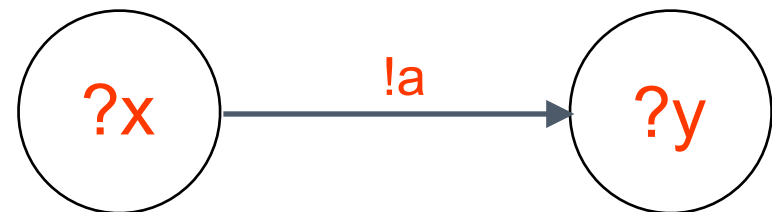
Problem 2

Q = SELECT ?x ?y WHERE {?x ?v ?y}

Q' = SELECT ?x ?y WHERE {?x !(a) ?y}



Q

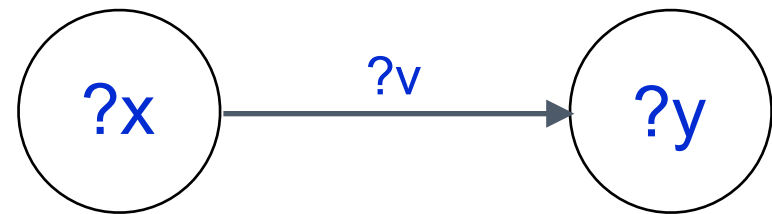
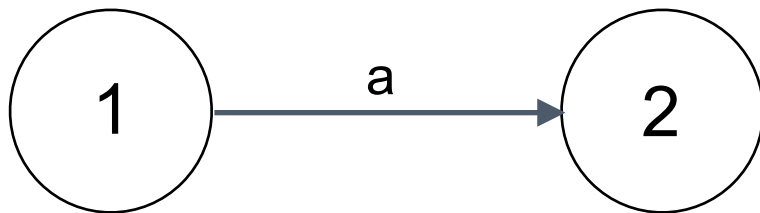


Q'

Problem 2

Q = SELECT ?x ?y WHERE {?x ?v ?y}

Q' = SELECT ?x ?y WHERE {?x !(a) ?y}



Q

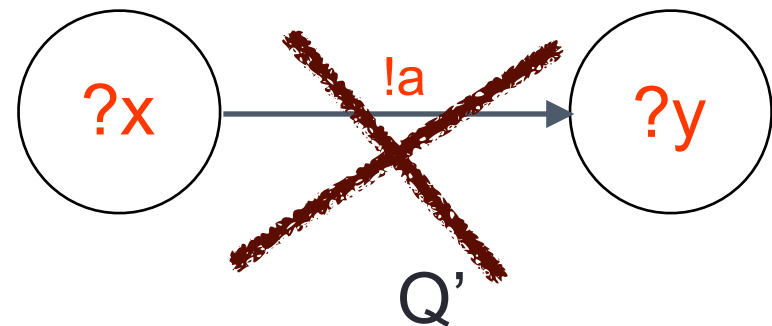
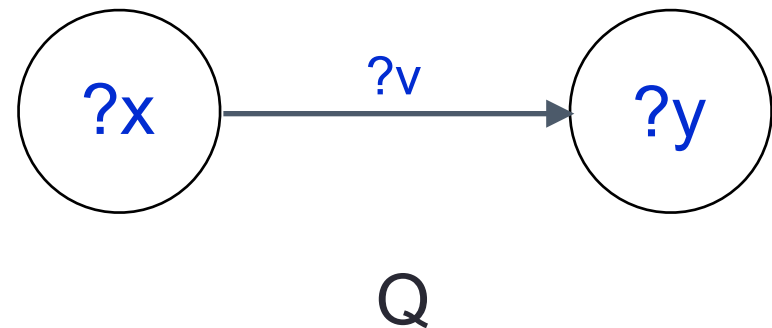
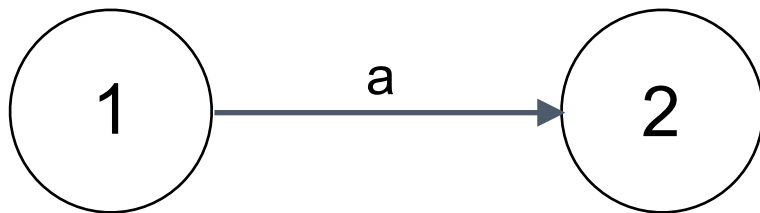


Q'

Problem 2

Q = SELECT ?x ?y WHERE {?x ?v ?y}

Q' = SELECT ?x ?y WHERE {?x !(a) ?y}



- Need to be careful when freezing variables

Canonical SPARQL databases for PPs

We can still build a canonical graph for Q w.r.t. Q' :

- may replace negated sets with a finite number of properties
- may replace variables for IRIs used in Q'

Canonical SPARQL databases for PPs

Containment bounds:

- **basic graph patterns + union**: PSPACE-complete
- **basic graph patterns + union + select**: EXPSPACE-complete

These are the same bounds known for graph DBs

Queries with OPTIONAL

Ideally:

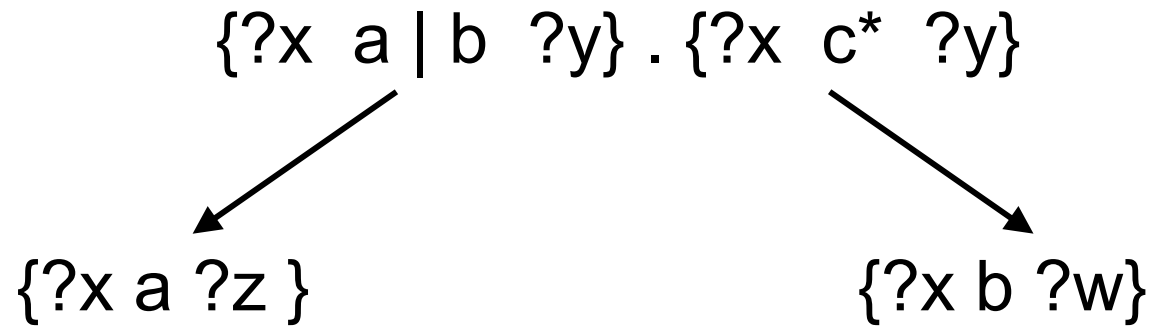
combine our result with algorithms for SPARQL containment

need to work with pattern trees

(Letelier et al 12', Pichler et al. 14',...)

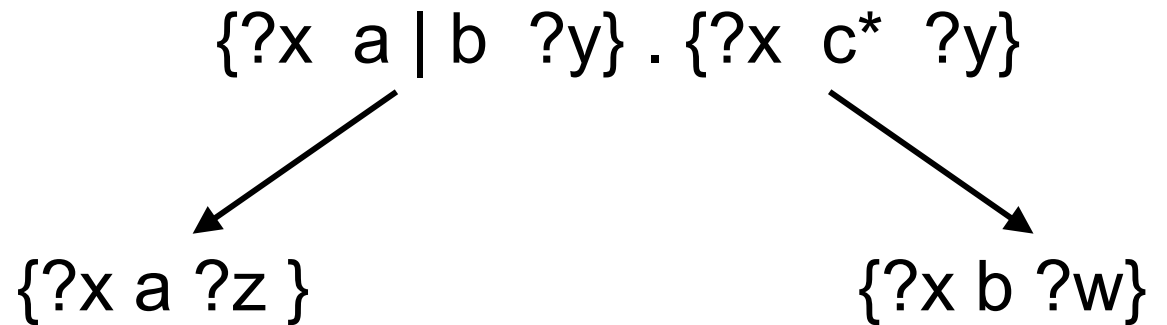
Pattern trees example

$\{?x a \mid b ?y\} . \{?x c^* ?y\} .$
OPTIONAL $\{?x a ?z\} .$
OPTIONAL $\{?x b ?w\}$



Pattern trees example

$\{?x \ a \ | \ b \ ?y\} \cdot \{?x \ c^* \ ?y\} \cdot$
OPTIONAL $\{?x \ a \ ?z \}$.
OPTIONAL $\{?x \ b \ ?w\}$



Previous work establish conditions for containment of SPARQL queries based on pattern trees

Queries with OPTIONAL

Bad news: This characterization won't work under PP's

Queries with OPTIONAL

Bad news: This characterization won't work under PP's

Good news: There are stronger conditions that work
(and extend previous work on pattern trees)

Queries with OPTIONAL

Bad news: This characterization won't work under PP's

Good news: There are stronger conditions that work
(and extend previous work on pattern trees)

Bad news: No time to present them here

Queries with OPTIONAL

Bad news: This characterization won't work under PP's

Good news: There are stronger conditions that work
(and extend previous work on pattern trees)

Bad news: No time to present them here

Good news: Find them in the paper!

Recap (last slide)

We Develop machinery to combine SPARQL and graph DBs

- Easy **evaluation** results
- Algorithms for **containment** and **subsumption**

Still work to do:

- Other operators (FILTER, aggregation...)
- Continue using this machinery (for e.g. entailment)

SPARQL with Property Paths

Egor V. Kostylev, **Juan L. Reutter**,
Miguel Romero, Domagoj Vrgoc

Oxford - PUC Chile - U. of Chile

Center for **Semantic Web** Research

