From Shapes to Shapes

Inferring SHACL Shapes for Results of SPARQL CONSTRUCT Queries

Philipp Seifer · *Daniel Hernández* · *Ralf Lämmel* · *Steffen Staab*

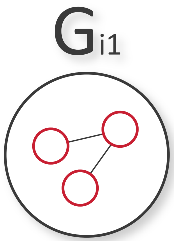
university of koblenz

University of Stuttgart
Germany

Artifacts Available acm

q

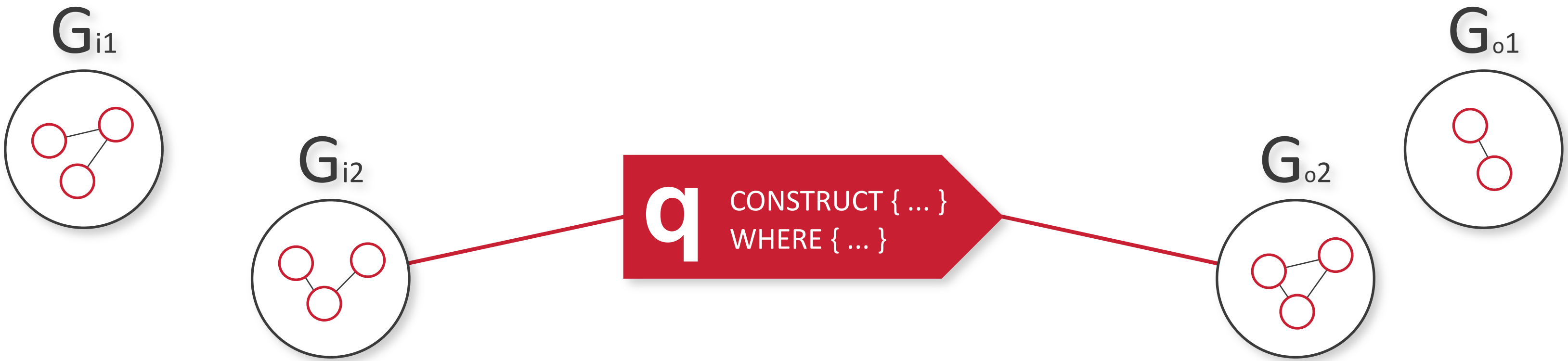
CONSTRUCT { ... }
WHERE { ... }



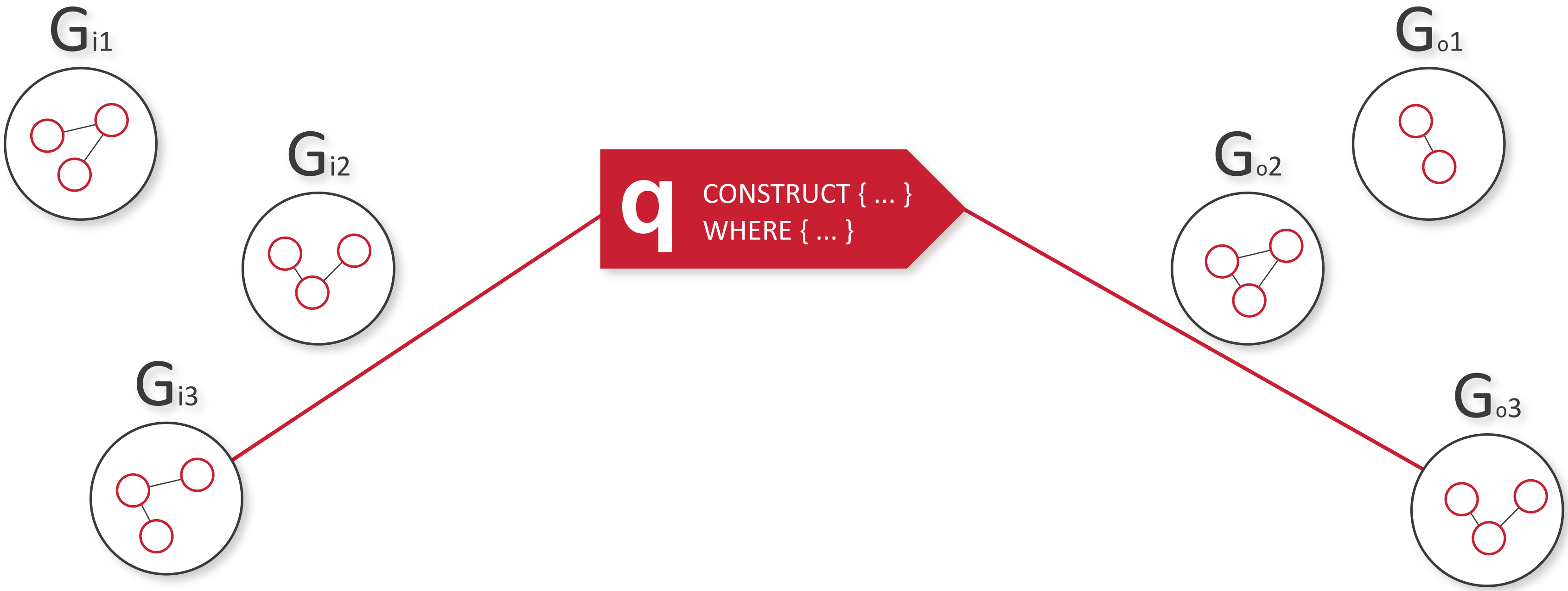
q CONSTRUCT { … }
WHERE { … }

university of koblenz

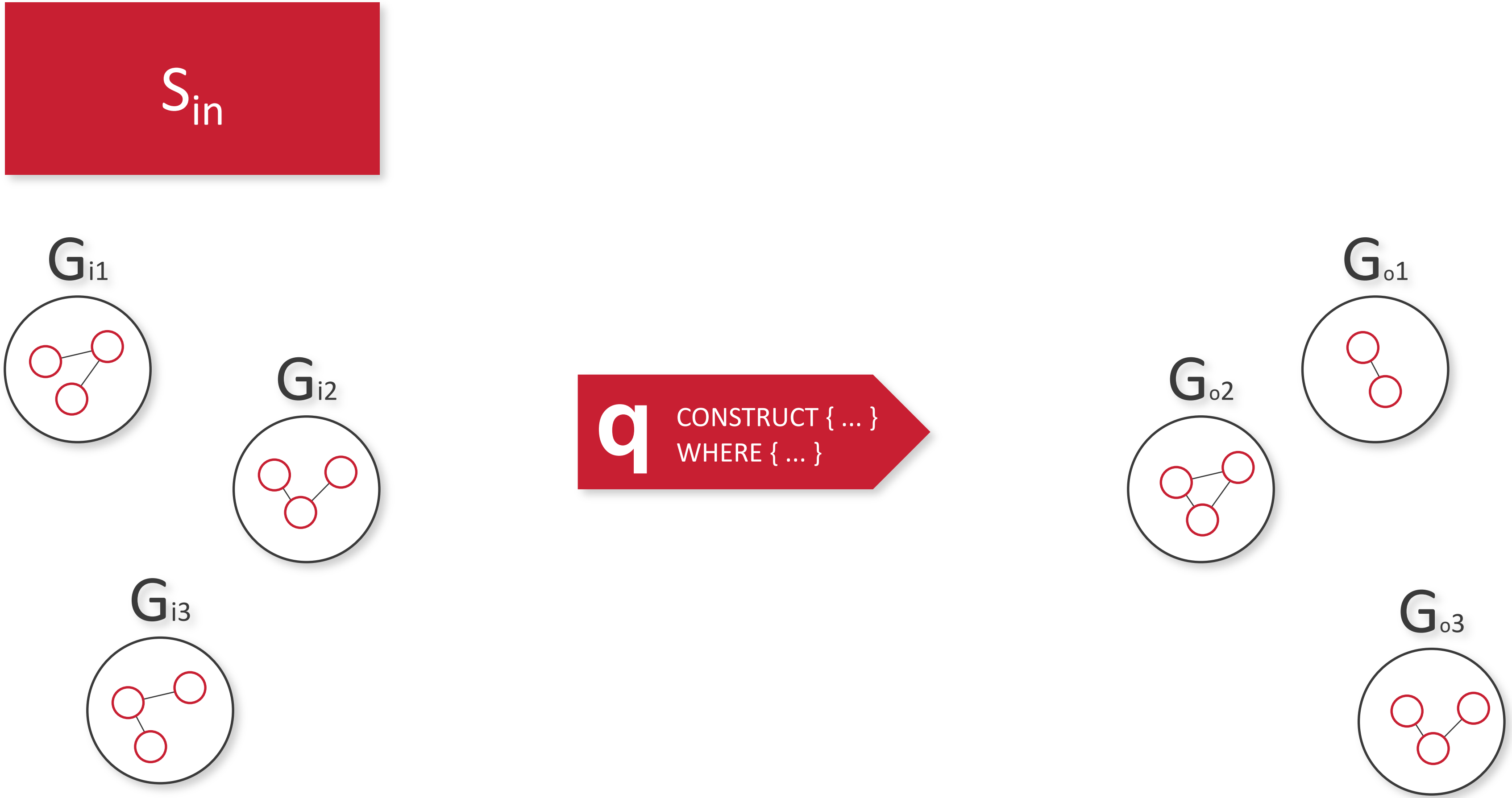

G_{i1}
G_{i2}
q
CONSTRUCT { ... }
WHERE { ... }
G_{o1}



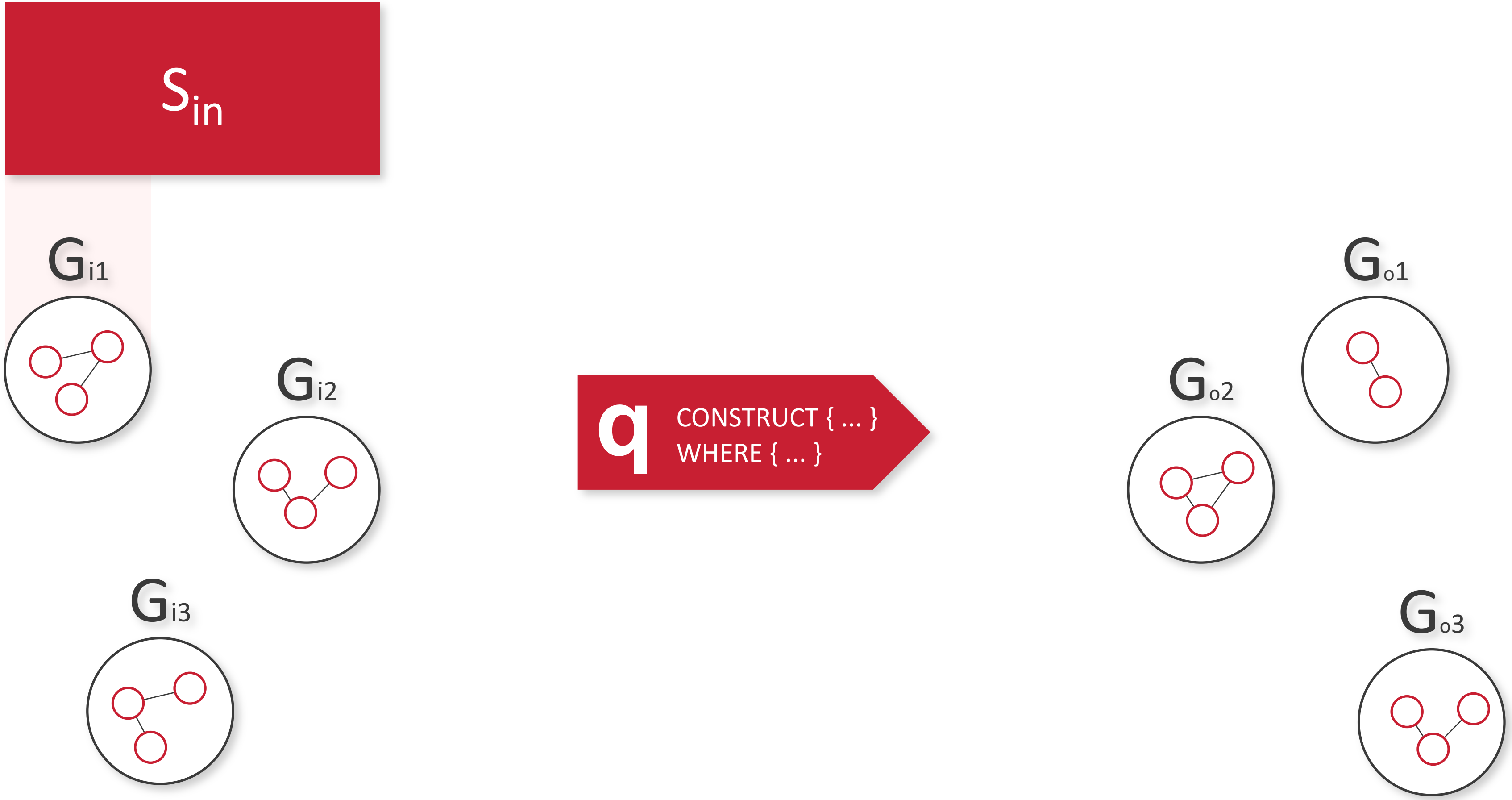
university of koblenz


Gi1
Gi2
Gi3
q
CONSTRUCT { ... }
WHERE { ... }
Go1
Go2
Go3

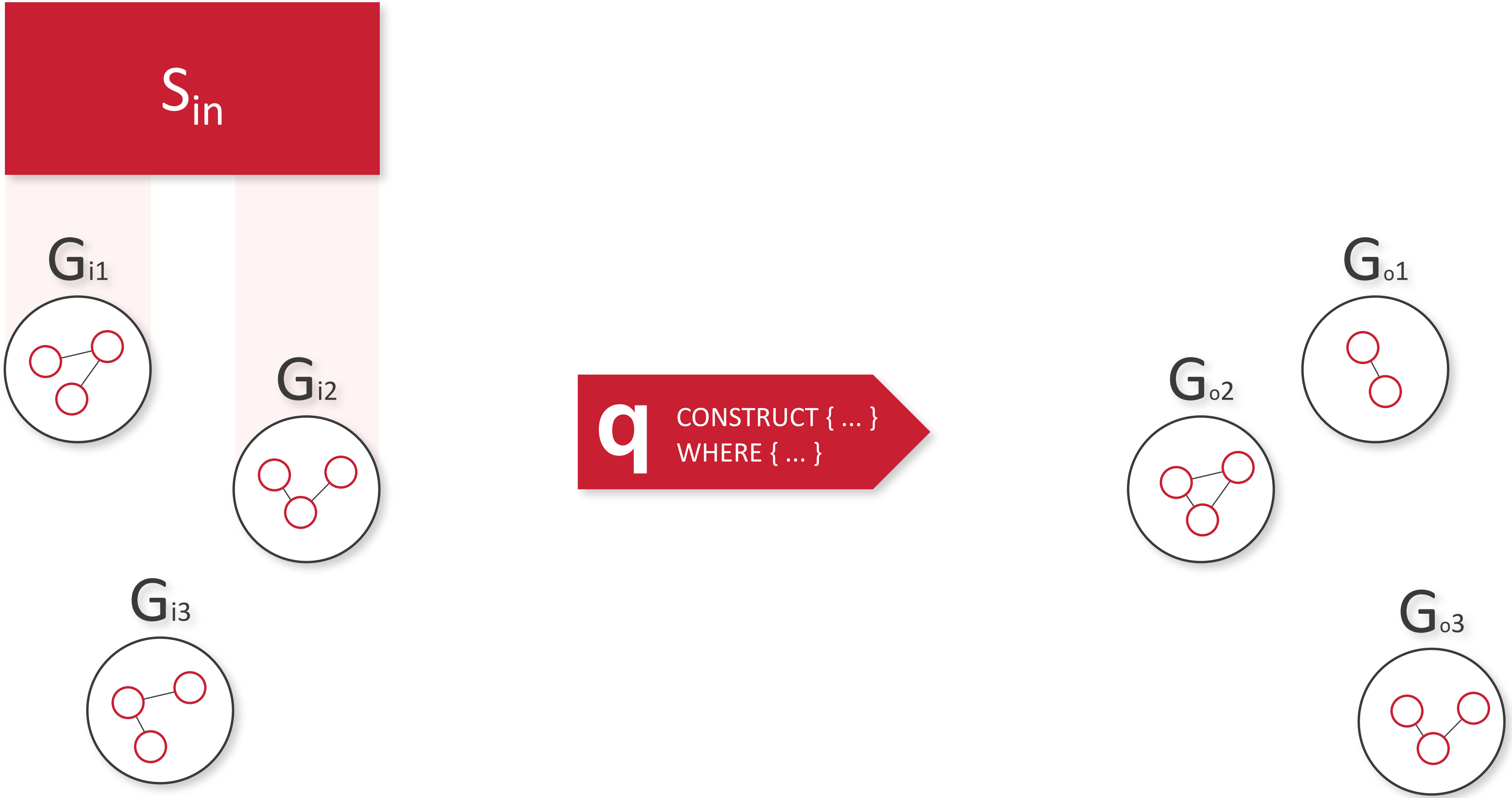

Gi1
Gi2
Gi3
q
CONSTRUCT { ... }
WHERE { ... }
Go1
Go2
Go3

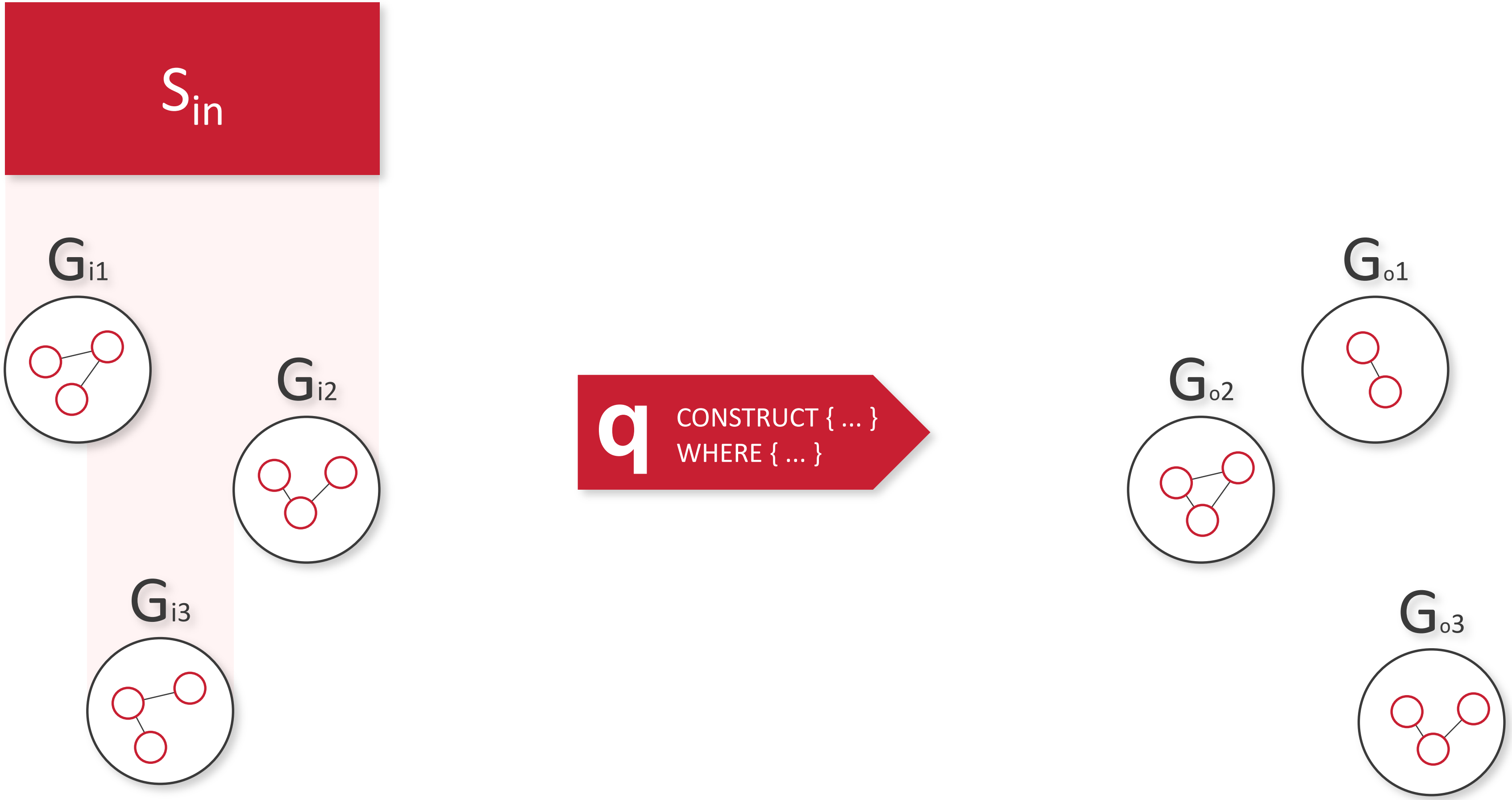


S_{in}
G_{i1}
G_{i2}
G_{i3}
q
CONSTRUCT { ... }
WHERE { ... }
G_{o1}
G_{o2}
G_{o3}



university of koblenz

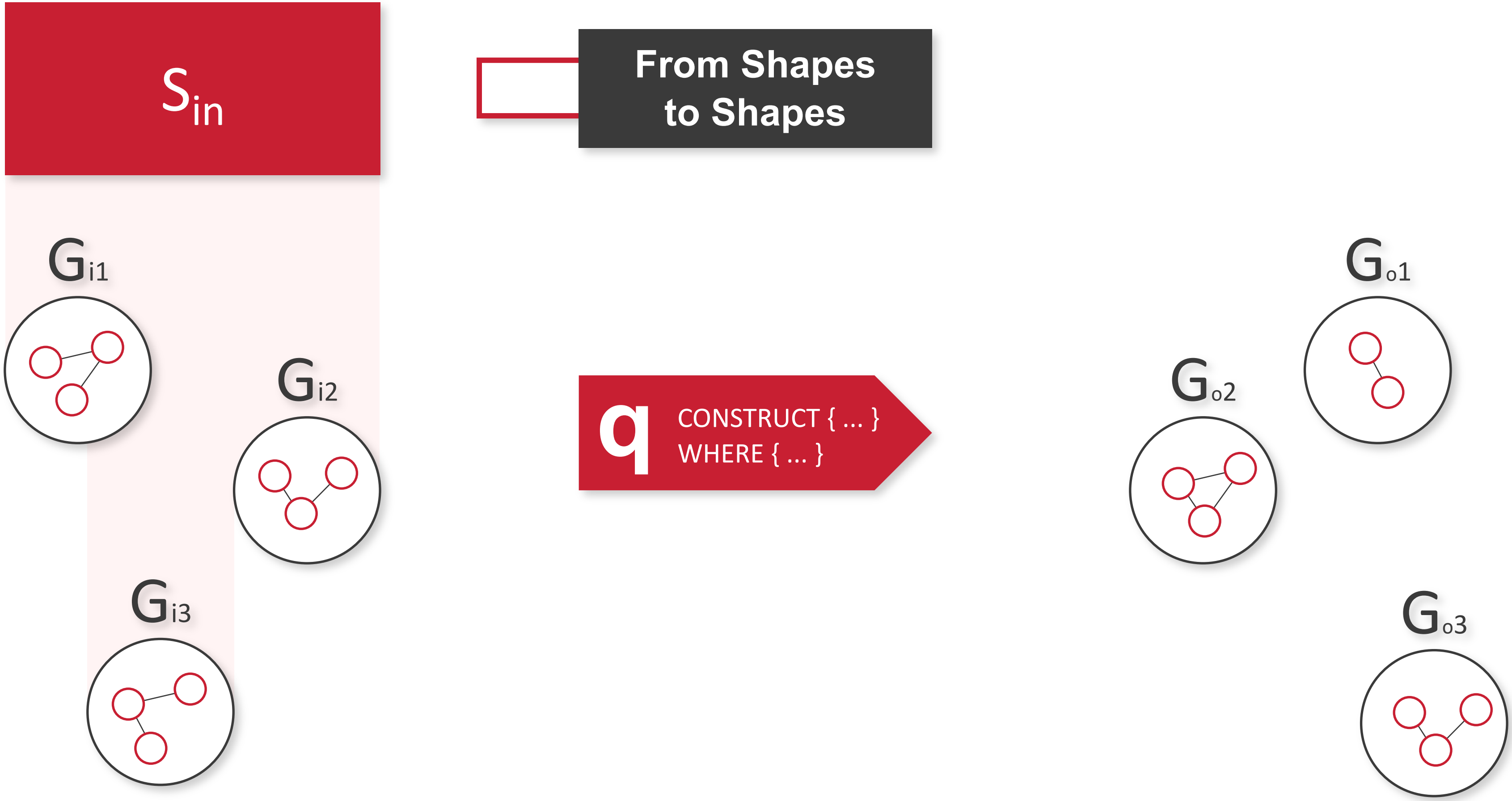

S_in
G_i1
G_i2
G_i3
q
CONSTRUCT { ... }
WHERE { ... }
G_o1
G_o2
G_o3

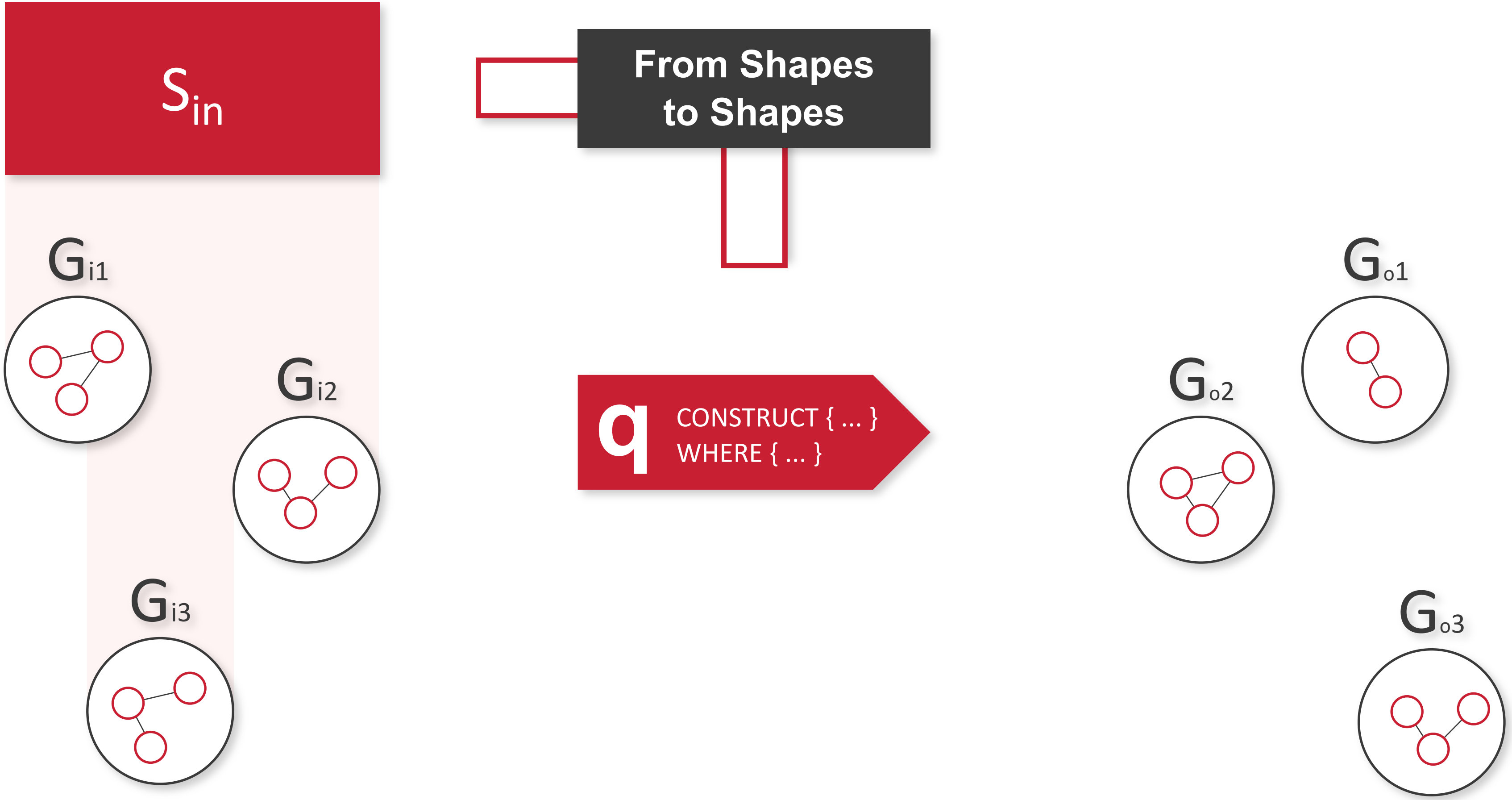


S_{in}
G_{i1}
G_{i2}
G_{i3}
q
CONSTRUCT { ... }
WHERE { ... }
G_{o1}
G_{o2}
G_{o3}

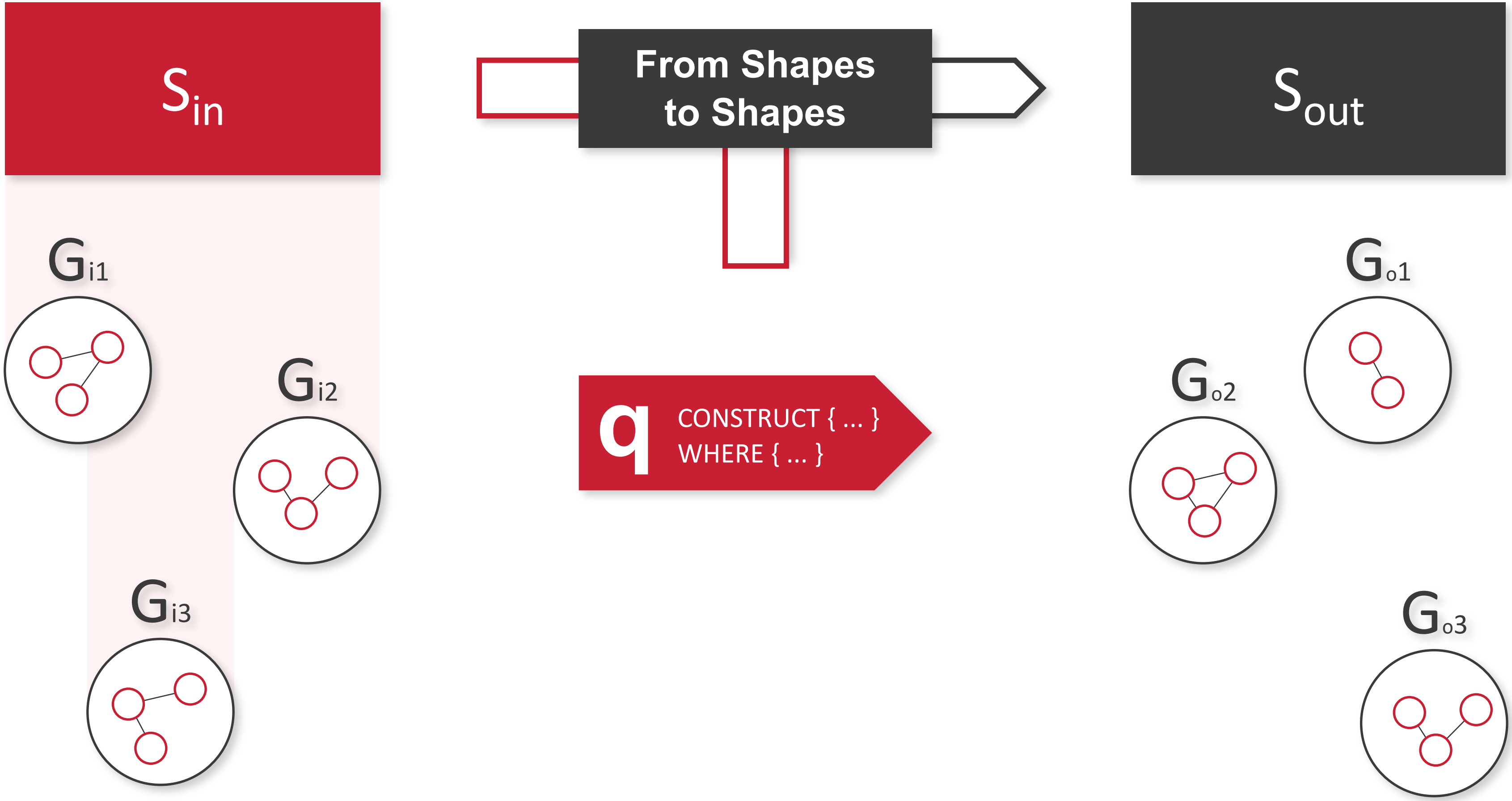
From Shapes to Shapes

From Shapes to Shapes

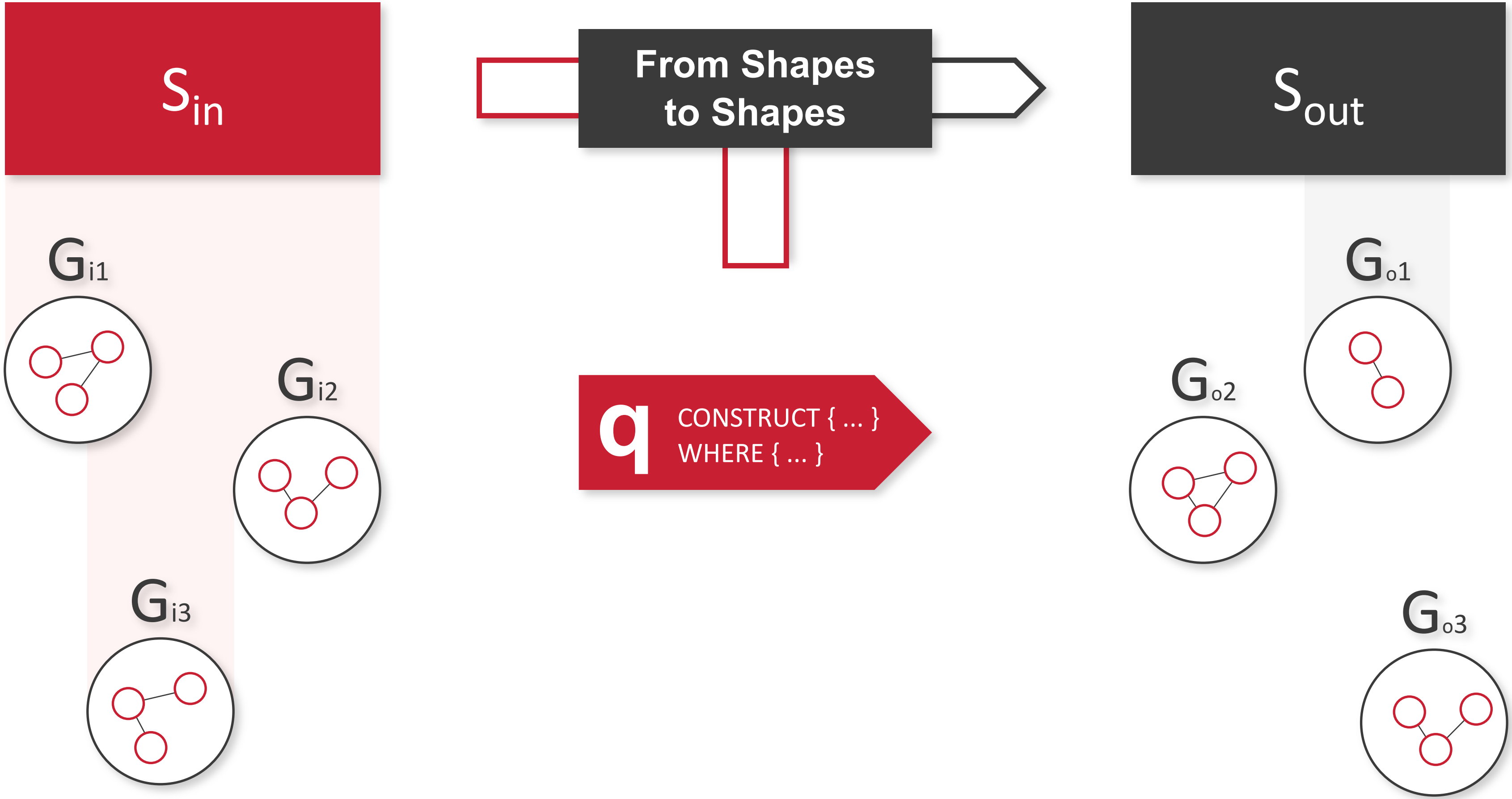


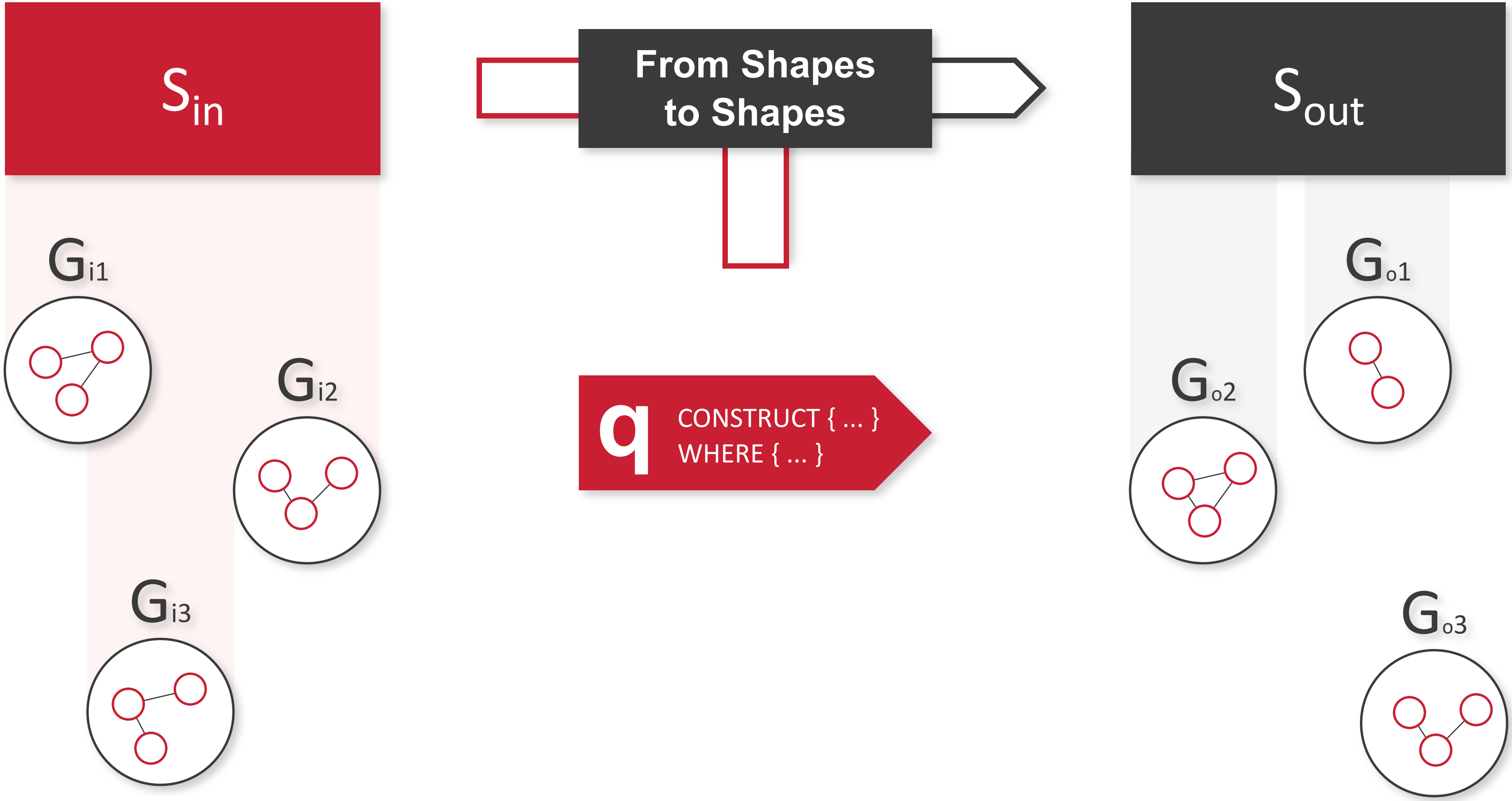


S in
From Shapes
to Shapes
G i1
G i2
G i3
q
CONSTRUCT { ... }
WHERE { ... }
G o1
G o2
G o3

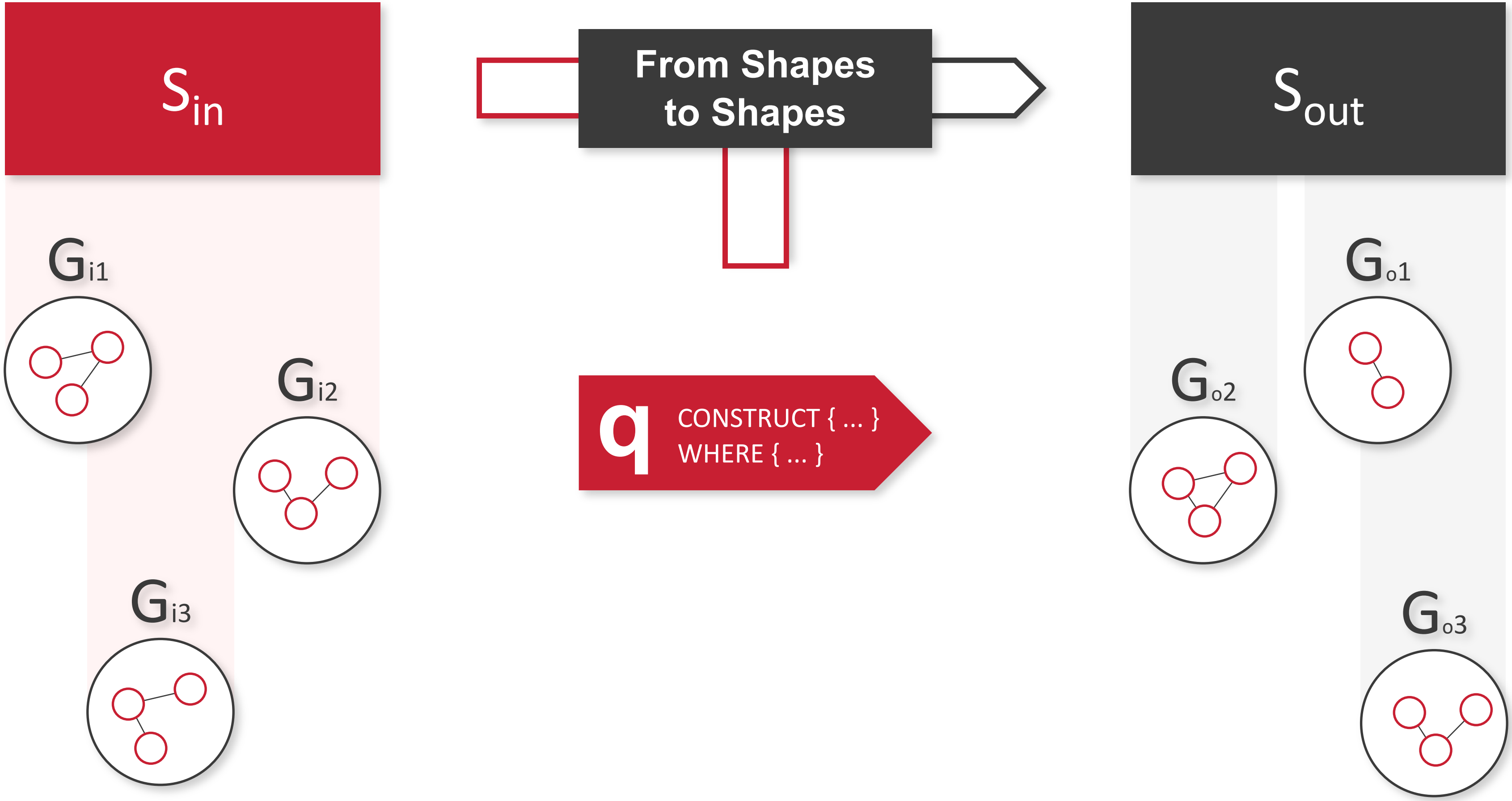


S in
From Shapes to Shapes
S out
G i1
G i2
G i3
q
CONSTRUCT { … }
WHERE { … }
G o1
G o2
G o3

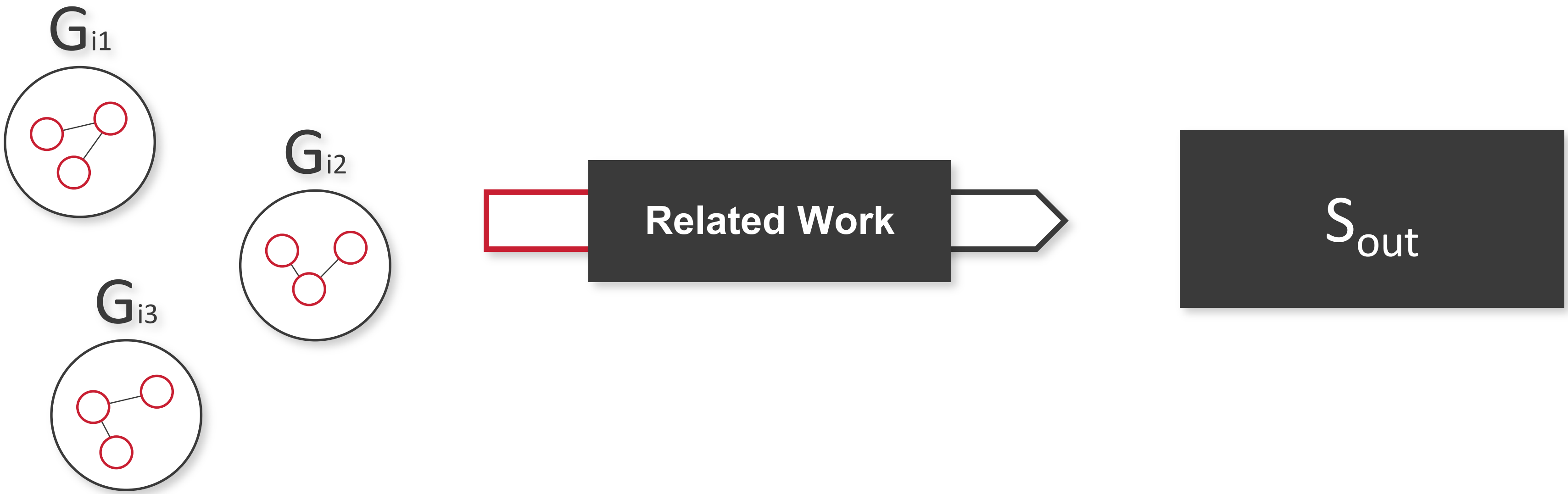

S_{in}
From Shapes to Shapes
S_{out}
G_{i1}
G_{i2}
G_{i3}
q
CONSTRUCT { ... }
WHERE { ... }
G_{o1}
G_{o2}
G_{o3}



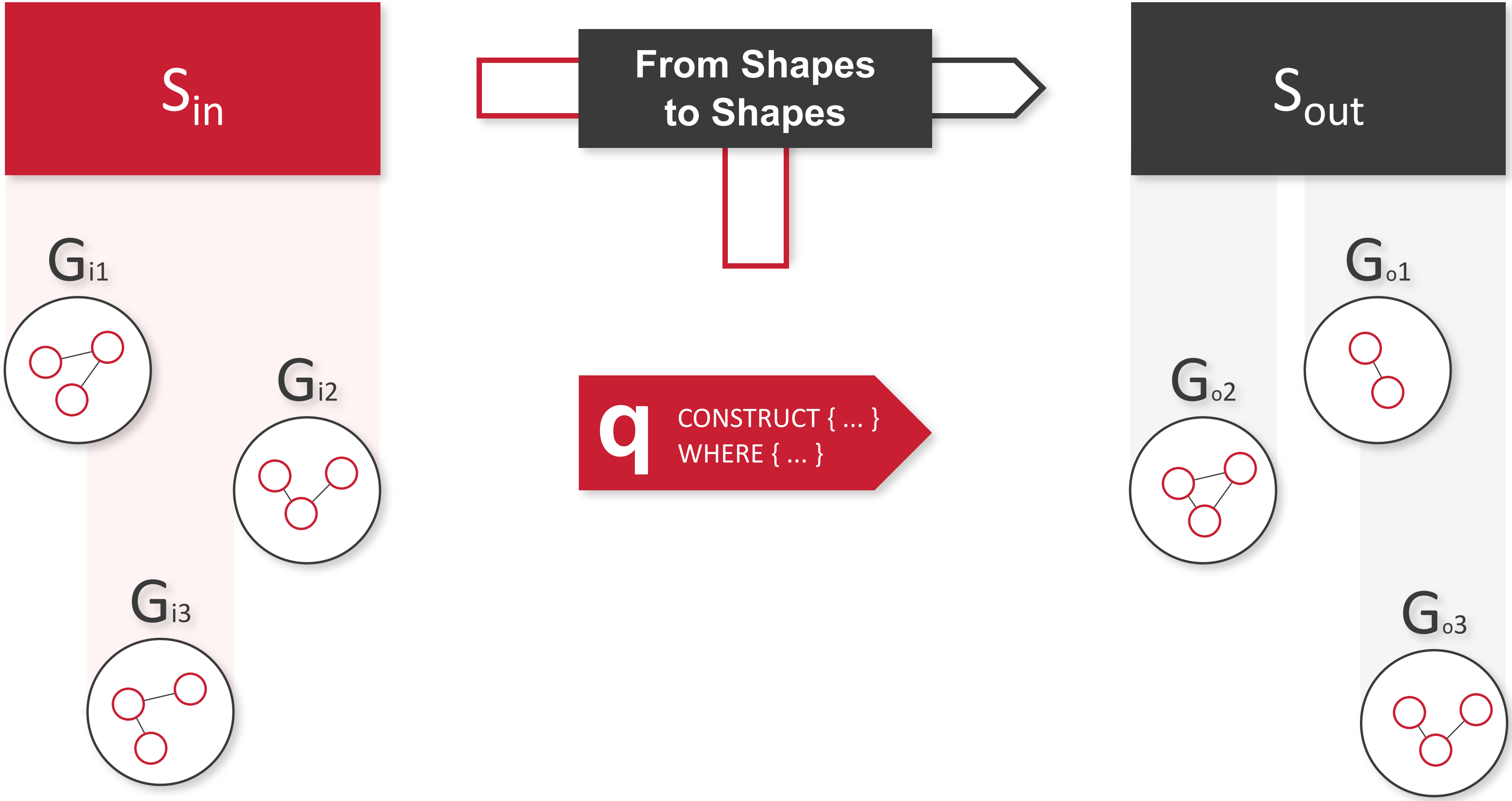
S_in
G_i1
G_i2
G_i3
From Shapes to Shapes
q
CONSTRUCT { ... }
WHERE { ... }
S_out
G_o1
G_o2
G_o3

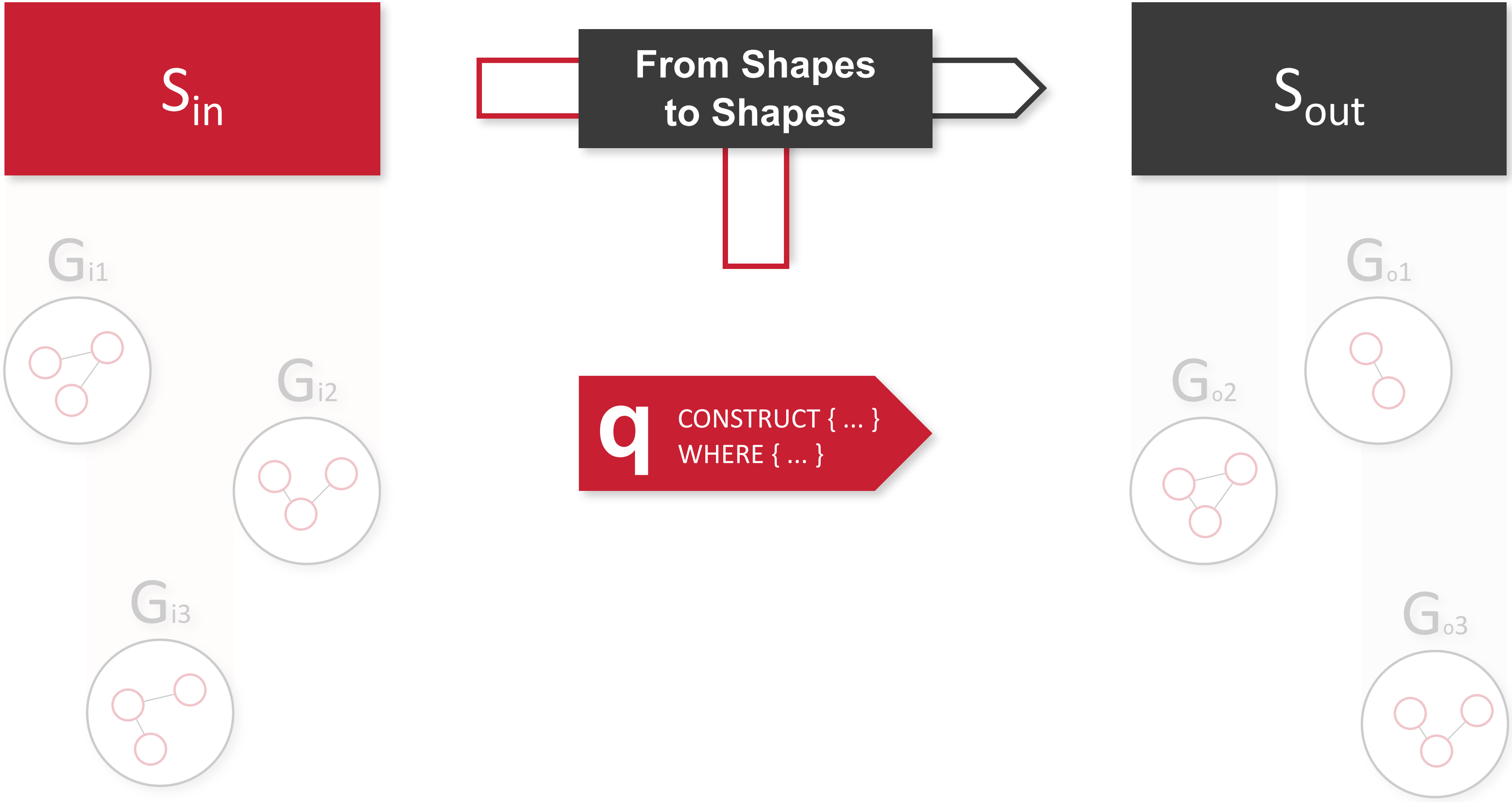


university of koblenz



G_{i1}
G_{i2}
G_{i3}
Related Work
S_{out}


S_{in}
From Shapes to Shapes
S_{out}
G_{i1}
G_{i2}
G_{i3}
q
CONSTRUCT { … }
WHERE { … }
G_{o1}
G_{o2}
G_{o3}


S in
From Shapes to Shapes
S out
G i1
G i2
G i3
q
CONSTRUCT { … }
WHERE { … }
G o1
G o2
G o3
university of koblenz

From Shapes to Shapes

```
CONSTRUCT {



} WHERE {
  ?w :p ?y .
  ?y  a :B .


}
```

From Shapes to Shapes

```
CONSTRUCT {



} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E
}
```

From Shapes to Shapes

```
CONSTRUCT {
  ?y  a :F .
  ?y :r ?z .
  ?z  a :G
} WHERE {
  ?w :p ?y .
  ?y  a :B .
  ?x :p ?z .
  ?z  a :E
}
```

S_{in}

"Every B is an E."

From Shapes to Shapes

```
CONSTRUCT {
   ?y  a :F .
   ?y :r ?z .
   ?z  a :G
} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E
}
```

S_{in}

“Every B is an E.”

From Shapes to Shapes

```
CONSTRUCT {
   ?y  a :F .
   ?y :r ?z .
   ?z  a :G
} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E
}
```

S_{in}

"Every B is an E."

From Shapes to Shapes

```
CONSTRUCT {
   ?y  a :F .
   ?y :r ?z .
   ?z  a :G
} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E
}
```

S$_{in}$

"Every B is an E."

From Shapes to Shapes

```
CONSTRUCT {
   ?y  a :F .
   ?y :r ?z .
   ?z  a :G
} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E
}
```

S_{in}

“Every B is an E.”

“?y” ⊆ “?z”

From Shapes to Shapes

```
CONSTRUCT {
   ?y  a :F .
   ?y :r ?z .
   ?z  a :G
} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E
}
```

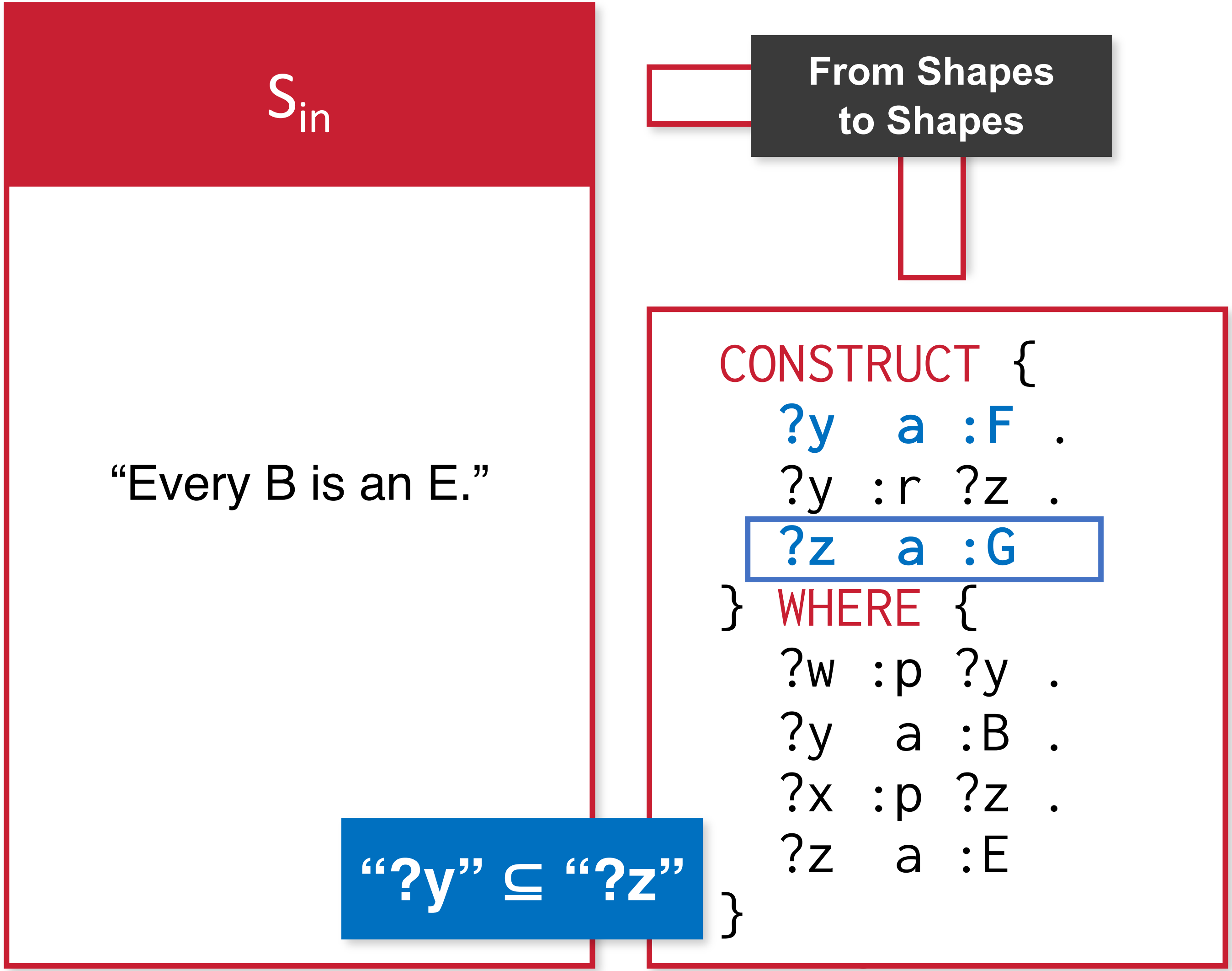
S_{in}

"Every B is an E."

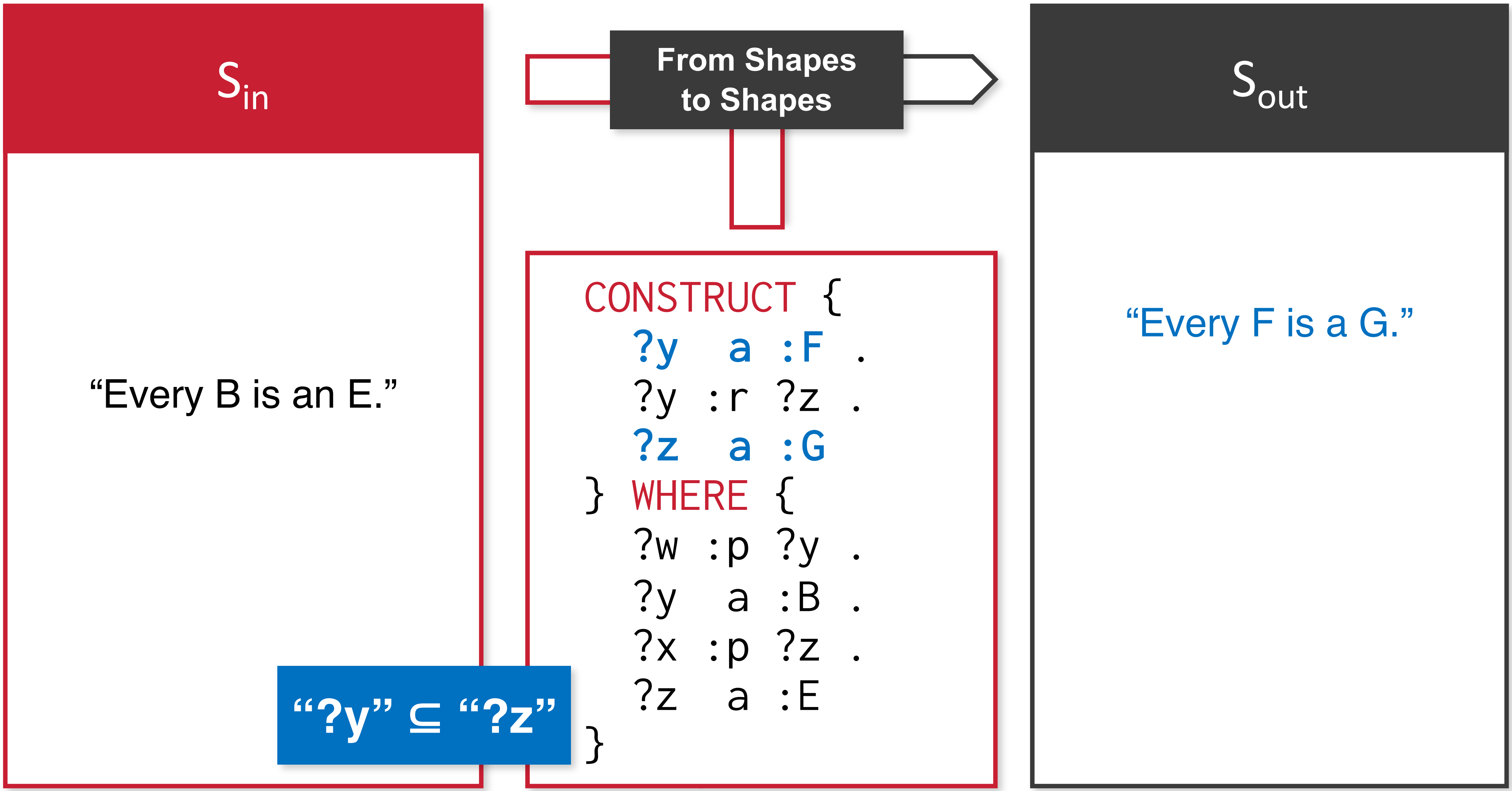
"?y" ⊆ "?z"

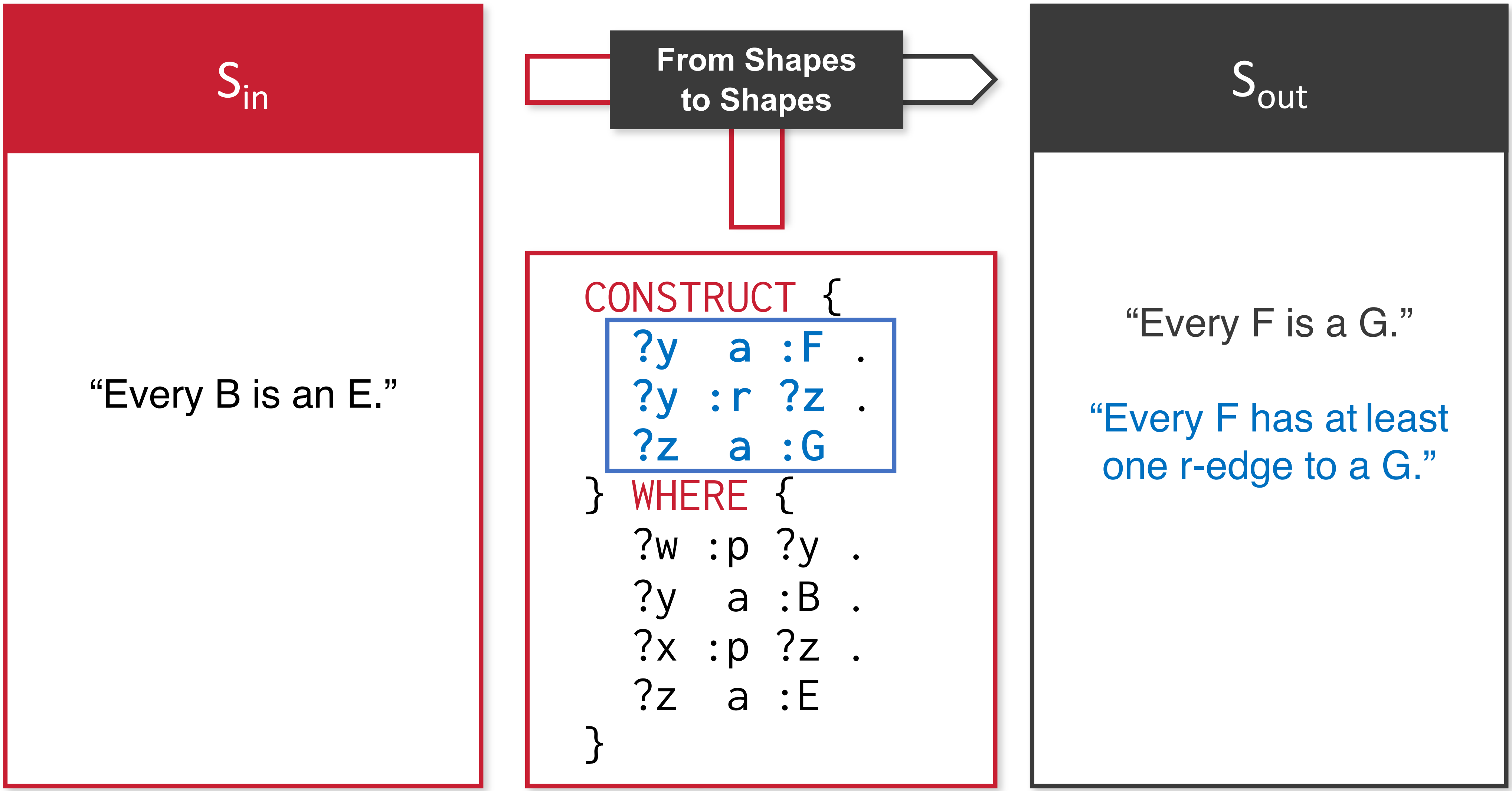
From Shapes to Shapes

```
CONSTRUCT {
  ?y  a :F .
  ?y :r ?z .
  ?z  a :G
} WHERE {
  ?w :p ?y .
  ?y  a :B .
  ?x :p ?z .
  ?z  a :E
}
```

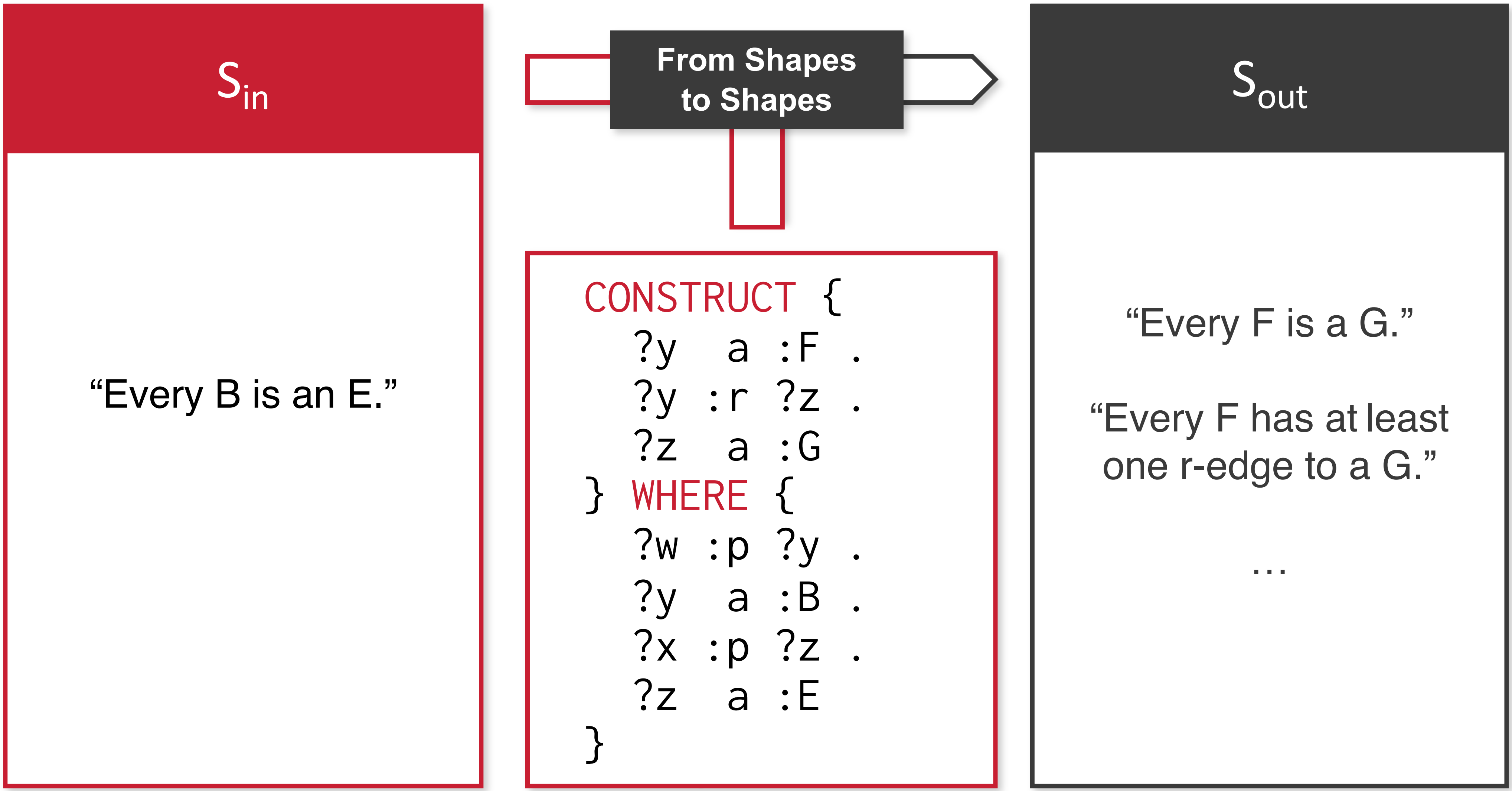


S_{in}
"Every B is an E."
"?y" ⊆ "?z"
From Shapes to Shapes
CONSTRUCT {
?y a :F .
?y :r ?z .
?z a :G
} WHERE {
?w :p ?y .
?y a :B .
?x :p ?z .
?z a :E
}


S_in
From Shapes to Shapes
S_out
“Every B is an E.”
CONSTRUCT {
?y a :F .
?y :r ?z .
?z a :G
} WHERE {
?w :p ?y .
?y a :B .
?x :p ?z .
?z a :E
}
“Every F is a G.”
“?y” ⊆ “?z”



S_in
"Every B is an E."
From Shapes to Shapes
CONSTRUCT {
?y a :F .
?y :r ?z .
?z a :G
} WHERE {
?w :p ?y .
?y a :B .
?x :p ?z .
?z a :E
}
S_out
"Every F is a G."
"Every F has at least one r-edge to a G."



S_in
"Every B is an E."
From Shapes to Shapes
CONSTRUCT {
?y a :F .
?y :r ?z .
?z a :G
} WHERE {
?w :p ?y .
?y a :B .
?x :p ?z .
?z a :E
}
S_out
"Every F is a G."
"Every F has at least one r-edge to a G."
...

“Every B is an E.”

“Every B is an E.”

Bogaerts et. al, 2022.
SHACL: A Description Logic in Disguise.
LPNMR. Springer

“Every B is an E.”

Bogaerts et. al, 2022.
SHACL: A Description Logic in Disguise.
LPNMR. Springer

$$B \sqsubseteq E$$

$\mathcal{ALCHOI}$

$\Sigma :=$

$$\Sigma := S_{in}$$

$$\Sigma := S_{in} \cup q_{infer}$$

$$\Sigma := S_{in} \cup q_{infer}$$

if $\Sigma \vdash s$

$$\Sigma := S_{in} \cup q_{infer}$$

$$\text{if } \Sigma \vdash s \text{ then } s \in S_{out}$$

(modulo namespaces)

$\Sigma := S_{in} \cup q_{infer}$

$B \sqsubseteq E$

```
CONSTRUCT {
   ?y  a :F .
   ?y :r ?z .
   ?z  a :G
} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E
}
```

$\Sigma := S_{in} \cup q_{infer}$

$B \sqsubseteq E$

X encoding **?x**
Y encoding **?y**
Z encoding **?z**

```
CONSTRUCT {
   ?y  a  :F  .
   ?y :r  ?z  .
   ?z  a  :G
} WHERE {
   ?w :p  ?y  .
   ?y  a  :B  .
   ?x :p  ?z  .
   ?z  a  :E
}
```

$\Sigma := S_{in} \cup q_{infer}$

$B \sqsubseteq E$

$X \equiv \exists p.Z$

$W \equiv \exists p.Y$

$Z \equiv E \sqcap (\exists p^-.X)$

$Y \equiv B \sqcap (\exists p^-.W)$

```
CONSTRUCT {
   ?y  a :F .
   ?y :r ?z .
   ?z  a :G
} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E
}
```

$\Sigma := S_{in} \cup q_{infer}$

$B \sqsubseteq E$

$X \equiv \exists p.Z$

$W \equiv \exists p.Y$

$Z \equiv E \sqcap (\exists p^{-}.X)$

$Y \equiv B \sqcap (\exists p^{-}.W)$

```
CONSTRUCT {
   ?y  a :F .
   ?y :r ?z .
   ?z  a :G
} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E
}
```

$\Sigma := S_{in} \cup q_{infer}$

$B \sqsubseteq E$

$X \equiv \exists p.Z$

$W \equiv \exists p.Y$

$Z \equiv E \sqcap (\exists p^{-}.X)$

$Y \equiv B \sqcap (\exists p^{-}.W)$

…

$G \equiv Z$

$\exists r.Z \equiv Y$

$\exists r^{-}.Y \equiv Z$

$F \equiv Y$

$\exists r.T \equiv Y \sqcap (\exists r.Z)$

$\exists r^{-}.T \equiv Z \sqcap (\exists p^{-}.Y)$

```
CONSTRUCT {
   ?y  a :F .
   ?y :r ?z .
   ?z  a :G
} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E
}
```

$\Sigma := S_{in} \cup q_{infer}$

$B \sqsubseteq E$

$X \equiv \exists p.Z$

$W \equiv \exists p.Y$

$G \equiv Z$

$\exists r.Z \equiv Y$

$\exists r^{-}.Y \equiv Z$

…

$Z \equiv E \sqcap (\exists p^{-}.X)$

$Y \equiv B \sqcap (\exists p^{-}.W)$

$F \equiv Y$

$\exists r.T \equiv Y \sqcap (\exists r.Z)$

$\exists r^{-}.T \equiv Z \sqcap (\exists p^{-}.Y)$

```
CONSTRUCT {
  ?y  a  :F  .
  ?y :r ?z  .
  ?z  a  :G
} WHERE {
  ?w :p ?y  .
  ?y  a  :B  .
  ?x :p ?z  .
  ?z  a  :E
}
```

Definition 16 (CWA-encoding). The *CWA-encoding* for a SCCQ $q = (H \leftarrow P)$, denoted $\mathrm{CWA}(q)$, is the minimal set of $\mathcal{ALCHOI}$ axioms including:

1. For each concept name A in P, $\dot{A} \equiv A \sqcap \bigsqcup_{u:A\in P} \mathrm{C}_u$.
2. For each concept name A in H, $\ddot{A} \equiv \bigsqcup_{u:A\in H} \mathrm{C}_u$.
3. For each variable x in $\mathrm{var}(q)$ the axiom

$$V_x \sqsubseteq \bigsqcap_{x:A\in P} A \sqcap \bigsqcap_{(x,u):p\in P} \exists p.\,\mathrm{C}_u \sqcap \bigsqcap_{(u,x):p\in P} \exists p^{-}.\,\mathrm{C}_u,$$

and if $\mathrm{vcg}(P)$ is acyclic w.r.t x, then also the axiom

$$V_x \sqsupseteq \bigsqcap_{x:A\in P} A \sqcap \bigsqcap_{(x,u):p\in P} \exists p.\,\mathrm{C}_u \sqcap \bigsqcap_{(u,x):p\in P} \exists p^{-}.\,\mathrm{C}_u\,.$$

4. For each role name p in pattern P the axioms

$$\exists \dot{p}.\, C_v \equiv \bigsqcup_{(u,v):p\in P} C_u, \qquad \exists \dot{p}.\top \equiv \bigsqcup_{(u,v):p\in P} C_u \sqcap \exists \dot{p}.\, C_v,$$

$$\exists \dot{p}^-.\, C_u \equiv \bigsqcup_{(u,v):p\in P} C_v, \qquad \exists \dot{p}^-.\top \equiv \bigsqcup_{(u,v):p\in P} C_v \sqcap \exists \dot{p}^-.\, C_u\,.$$

5. For each role name p in template H the axioms

$$\exists \ddot{p}.\, C_v \equiv \bigsqcup_{(u,v):p\in H} C_u, \qquad \exists \ddot{p}.\top \equiv \bigsqcup_{(u,v):p\in H} C_u \sqcap \exists \ddot{p}.\, C_v,$$

$$\exists \ddot{p}^-.\, C_u \equiv \bigsqcup_{(u,v):p\in H} C_v, \qquad \exists \ddot{p}^-.\top \equiv \bigsqcup_{(u,v):p\in P} C_v \sqcap \exists \ddot{p}^-.\, C_u\,.$$

$\Sigma := S_{in} \cup q_{infer}$

$B \sqsubseteq E$

$X \equiv \exists p.Z$

$W \equiv \exists p.Y$

$Z \equiv E \sqcap (\exists p^{-}.X)$

$Y \equiv B \sqcap (\exists p^{-}.W)$

…

$G \equiv Z$

$\exists r.Z \equiv Y$

$\exists r^{-}.Y \equiv Z$

$F \equiv Y$

$\exists r.\top \equiv Y \sqcap (\exists r.Z)$

$\exists r^{-}.\top \equiv Z \sqcap (\exists p^{-}.Y)$

```
CONSTRUCT {
   ?y  a  :F  .
   ?y :r ?z  .
   ?z  a  :G
} WHERE {
   ?w :p ?y  .
   ?y  a  :B  .
   ?x :p ?z  .
   ?z  a  :E
}
```

$$\Sigma := S_{in} \cup q_{infer}$$

$B \sqsubseteq E$

$Y \sqsubseteq Z$?

$X \equiv \exists p.Z$

$W \equiv \exists p.Y$

$Z \equiv E \sqcap (\exists p^{-}.X)$

$Y \equiv B \sqcap (\exists p^{-}.W)$

…

$G \equiv Z$

$\exists r.Z \equiv Y$

$\exists r^{-}.Y \equiv Z$

$F \equiv Y$

$\exists r.T \equiv Y \sqcap (\exists r.Z)$

$\exists r^{-}.T \equiv Z \sqcap (\exists p^{-}.Y)$

```
CONSTRUCT {
   ?y  a  :F  .
   ?y :r ?z  .
   ?z  a  :G
} WHERE {
   ?w :p ?y  .
   ?y  a  :B  .
   ?x :p ?z  .
   ?z  a  :E
}
```

```
CONSTRUCT {
  …
} WHERE {
  ?w :p ?y .
  ?y  a :B .
  ?x :p ?z .
  ?z  a :E .
}
```

```
CONSTRUCT {
  ...
} WHERE {
  ?w :p ?y .
  ?y  a :B .
  ?x :p ?z .
  ?z  a :E .
}
```

```
?w :p ?y .
?y  a :B .
```

1

```
CONSTRUCT {
   …
} WHERE {
   ?w :p ?y .
   ?y  a :B .
   ?x :p ?z .
   ?z  a :E .
}
```

```
?w :p ?y .
?y  a :B .
```

1

```
?x :p ?z .
?z  a :E .
```

2

```
?w :p ?y .
?y  a :B .
```

```
?x :p ?z .
?z  a :E .
```

Find a **mapping** **h** **of query variables**, such that **one component** is subset of **the other**.

```
?w  :p ?y  .
?y   a :B  .
```

Find a **mapping** **h** **of query variables**, such that **one component** is subset of **the other**.

```
?x  :p ?z  .
?z   a :E  .
```

Then a mapping such as h(x) = w implies W ⊑ X.

```
?w :p ?y .
?y  a :B .
```

```
?x :p ?z .
?z  a :E .
```

```
?w  :p ?y  .
?y   a :B  .
```

$B \sqsubseteq E$

```
?x  :p ?z  .
?z   a :E  .
```

```
?w  :p  ?y  .
?y   a  :B  .
```

$B \sqsubseteq E$

```
?w  :p  ?y  .
?y   a  :B  .
?y   a  :E  .
```

```
?x  :p  ?z  .
?z   a  :E  .
```

```
?x  :p  ?z  .
?z   a  :E  .
```

```
?w  :p  ?y  .
?y   a  :B  .
?y   a  :E  .
```

```
?x  :p  ?z  .
?z   a  :E  .
```

$h(x) = w$

$h(z) = y$

```
?w :p ?y .
?y  a :B .
?y  a :E .
```

```
?x :p ?z .
?z  a :E .
```

h(x) = w
h(z) = y

```
?w :p ?y .
?y  a :E .
```

```
?w :p ?y .
?y  a :B .
?y  a :E .
```

```
?w :p ?y .
?y  a :B .
?y  a :E .
```

⊆

```
?x :p ?z .
?z  a :E .
```

h(x) = w
h(z) = y

```
?w :p ?y .
?y  a :E .
```

```
?w :p ?y .
?y  a :B .
?y  a :E .
```

$h(x) = w$
$h(z) = y$

$\sqsubseteq$

$W \sqsubseteq X$
$Y \sqsubseteq Z$

```
?w :p ?y .
?y  a :E .
```

$\Sigma := S_{in} \cup q_{infer}$

$B \sqsubseteq E$

$W \sqsubseteq X$
$Y \sqsubseteq Z$

$X \equiv \exists p.Z$ $\quad$ $Z \equiv E \sqcap (\exists p^{-}.X)$

$W \equiv \exists p.Y$ $\quad$ $Y \equiv B \sqcap (\exists p^{-}.W)$

…

$G \equiv Z$ $\quad$ $F \equiv Y$

$\exists r.Z \equiv Y$ $\quad$ $\exists r.T \equiv Y \sqcap (\exists r.Z)$

$\exists r^{-}.Y \equiv Z$ $\quad$ $\exists r^{-}.T \equiv Z \sqcap (\exists p^{-}.Y)$

```
CONSTRUCT {
  ?y  a  :F  .
  ?y :r  ?z  .
  ?z  a  :G
} WHERE {
  ?w :p  ?y  .
  ?y  a  :B  .
  ?x :p  ?z  .
  ?z  a  :E
}
```

$\Sigma := S_{in} \cup q_{infer}$

$B \sqsubseteq E$

$W \sqsubseteq X$

$Y \sqsubseteq Z$

$X \equiv \exists p.Z$

$Z \equiv E \sqcap (\exists p^{-}.X)$

$W \equiv \exists p.Y$

$Y \equiv B \sqcap (\exists p^{-}.W)$

…

$G \equiv Z$

$F \equiv Y$

$\exists r.Z \equiv Y$

$\exists r.T \equiv Y \sqcap (\exists r.Z)$

$\exists r^{-}.Y \equiv Z$

$\exists r^{-}.T \equiv Z \sqcap (\exists p^{-}.Y)$

$\vdash$

S_{out}

$$\Sigma := S_{in} \cup q_{infer}$$

$B \sqsubseteq E$

$W \sqsubseteq X$

$Y \sqsubseteq Z$

$X \equiv \exists p.Z$

$W \equiv \exists p.Y$

$Z \equiv E \sqcap (\exists p^{-}.X)$

$Y \equiv B \sqcap (\exists p^{-}.W)$

…

$G \equiv Z$

$\exists r.Z \equiv Y$

$\exists r^{-}.Y \equiv Z$

$F \equiv Y$

$\exists r.T \equiv Y \sqcap (\exists r.Z)$

$\exists r^{-}.T \equiv Z \sqcap (\exists p^{-}.Y)$

$\vdash$

$$S_{out}$$

$F \sqsubseteq G$

$$\Sigma := S_{in} \cup q_{infer}$$

$B \sqsubseteq E$

$W \sqsubseteq X$

$Y \sqsubseteq Z$

$X \equiv \exists p.Z$

$W \equiv \exists p.Y$

$Z \equiv E \sqcap (\exists p^{-}.X)$

$Y \equiv B \sqcap (\exists p^{-}.W)$

...

$G \equiv Z$

$\exists r.Z \equiv Y$

$\exists r^{-}.Y \equiv Z$

$F \equiv Y$

$\exists r.T \equiv Y \sqcap (\exists r.Z)$

$\exists r^{-}.T \equiv Z \sqcap (\exists p^{-}.Y)$

$\vdash$

$$S_{out}$$

$F \sqsubseteq G$

$F \sqsubseteq \exists r.G$

$$\Sigma := S_{in} \cup q_{infer}$$

$B \sqsubseteq E$

$W \sqsubseteq X$

$Y \sqsubseteq Z$

$X \equiv \exists p.Z$

$W \equiv \exists p.Y$

$Z \equiv E \sqcap (\exists p^{-}.X)$

$Y \equiv B \sqcap (\exists p^{-}.W)$

…

$G \equiv Z$

$\exists r.Z \equiv Y$

$\exists r^{-}.Y \equiv Z$

$F \equiv Y$

$\exists r.T \equiv Y \sqcap (\exists r.Z)$

$\exists r^{-}.T \equiv Z \sqcap (\exists p^{-}.Y)$

$\vdash$

$$S_{out}$$

$F \sqsubseteq G$

$F \sqsubseteq \exists r.G$

$\exists r.T \sqsubseteq F$

$$\Sigma := S_{in} \cup q_{infer}$$

$$B \sqsubseteq E$$

$$W \sqsubseteq X$$

$$Y \sqsubseteq Z$$

$$X \equiv \exists p.Z$$

$$W \equiv \exists p.Y$$

$$Z \equiv E \sqcap (\exists p^{-}.X)$$

$$Y \equiv B \sqcap (\exists p^{-}.W)$$

…

$$G \equiv Z$$

$$\exists r.Z \equiv Y$$

$$\exists r^{-}.Y \equiv Z$$

$$F \equiv Y$$

$$\exists r.\top \equiv Y \sqcap (\exists r.Z)$$

$$\exists r^{-}.\top \equiv Z \sqcap (\exists p^{-}.Y)$$

$$\vdash$$

$$S_{out}$$

$$F \sqsubseteq G$$

$$F \sqsubseteq \exists r.G$$

$$\exists r.\top \sqsubseteq F$$

$$G \sqsubseteq \forall r.G$$

$$\Sigma := S_{in} \cup q_{infer}$$

$B \sqsubseteq E$

$W \sqsubseteq X$

$Y \sqsubseteq Z$

$X \equiv \exists p.Z$

$W \equiv \exists p.Y$

$Z \equiv E \sqcap (\exists p^{-}.X)$

$Y \equiv B \sqcap (\exists p^{-}.W)$

…

$G \equiv Z$

$\exists r.Z \equiv Y$

$\exists r^{-}.Y \equiv Z$

$F \equiv Y$

$\exists r.\top \equiv Y \sqcap (\exists r.Z)$

$\exists r^{-}.\top \equiv Z \sqcap (\exists p^{-}.Y)$

$$\vdash$$

$$S_{out}$$

$F \sqsubseteq G$

$F \sqsubseteq \exists r.G$

$\exists r.\top \sqsubseteq F$

$G \sqsubseteq \forall r.G$

…

Try out **Shapes 2 Shapes** at

*https://github.com/***softlang/s2s**

From Shapes to Shapes

Inferring SHACL Shapes for Results of SPARQL CONSTRUCT Queries

Philipp Seifer · *Daniel Hernández* · *Ralf Lämmel* · *Steffen Staab*

university of koblenz

University of Stuttgart
Germany

Artifacts Available acm