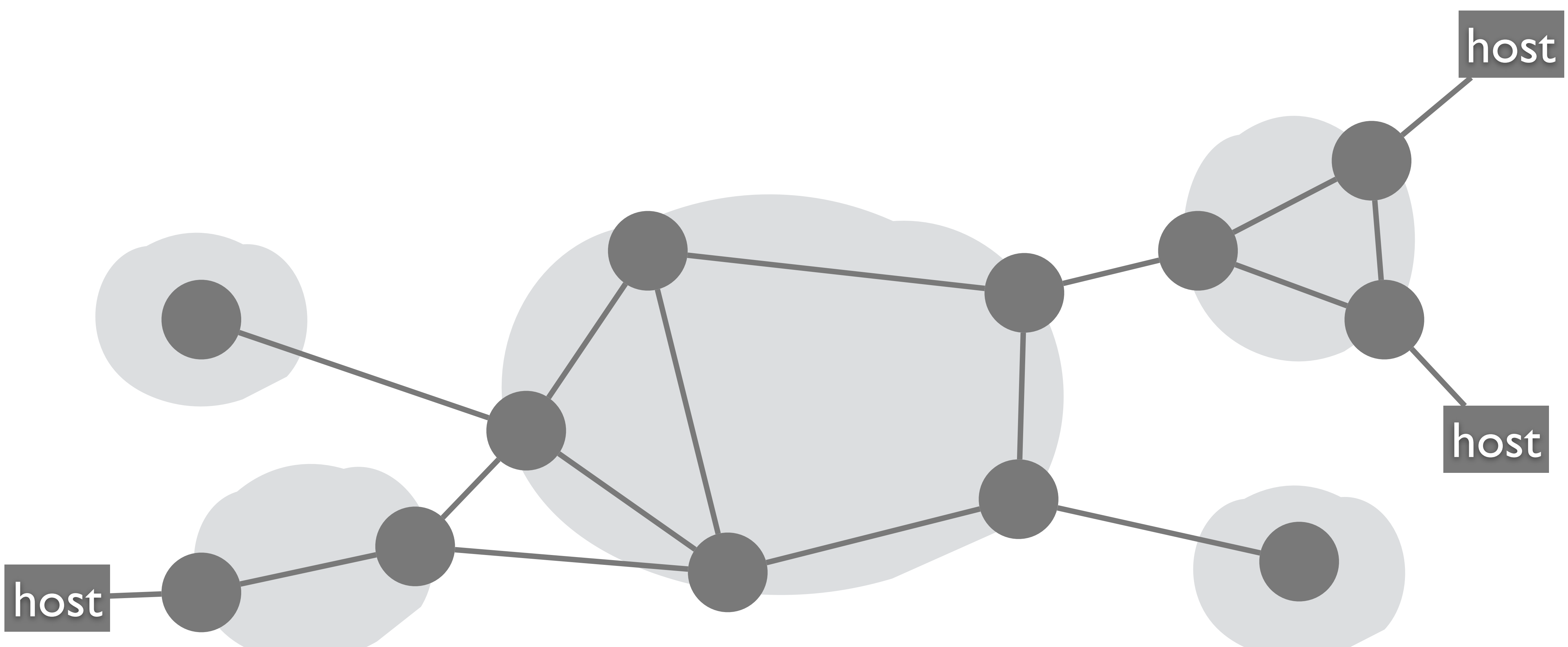
Rethinking Network Policy Coordination — A Database Perspective

Anduo Wang[*] Seungwon Shin[†] Eduard Dragut[*]
[*]Temple University [†]KAIST

APNet 2019 August 18, 2019, Beijing

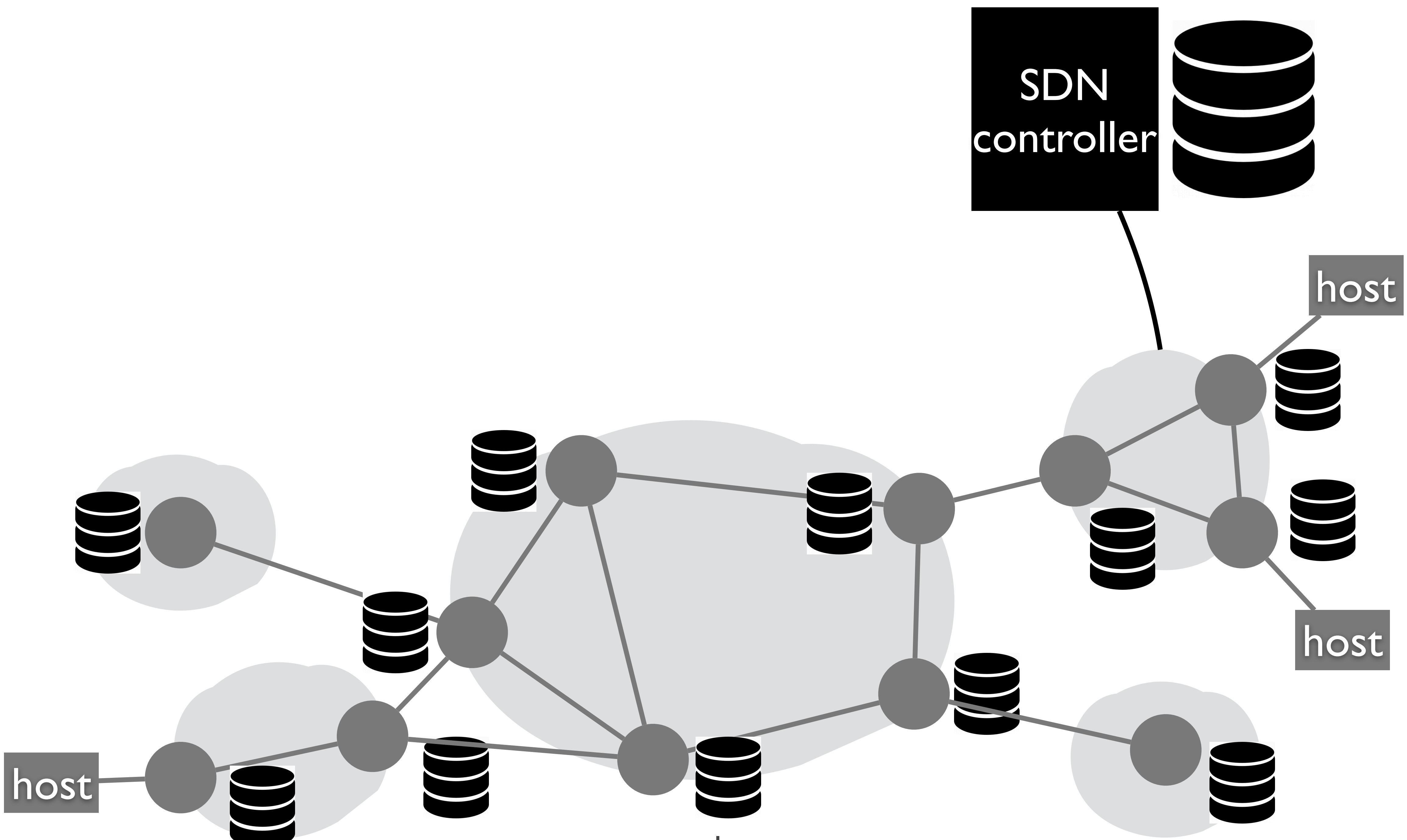
database usage in networking

great for managing network state



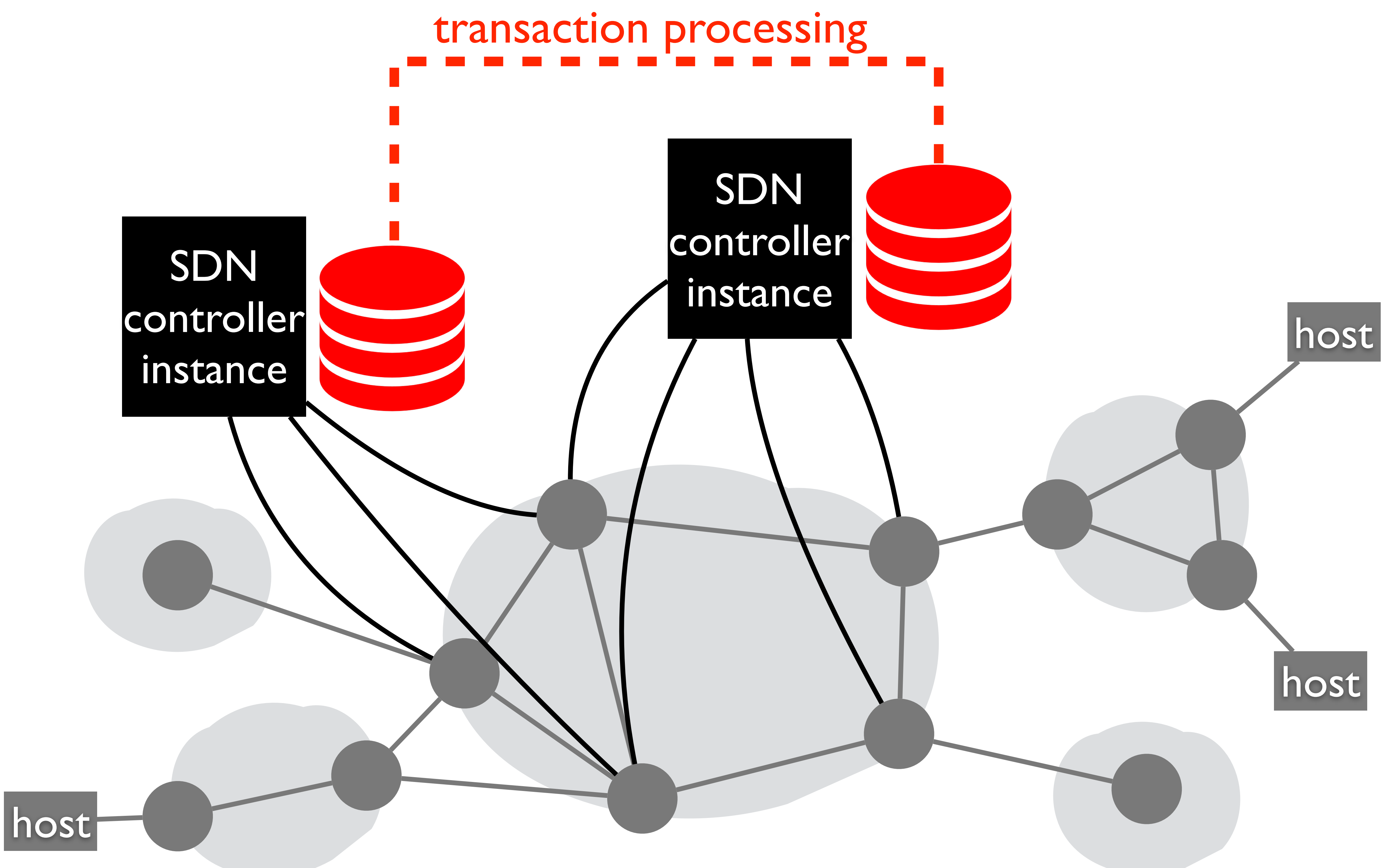
database usage in networking

great for managing network state — *factual* data



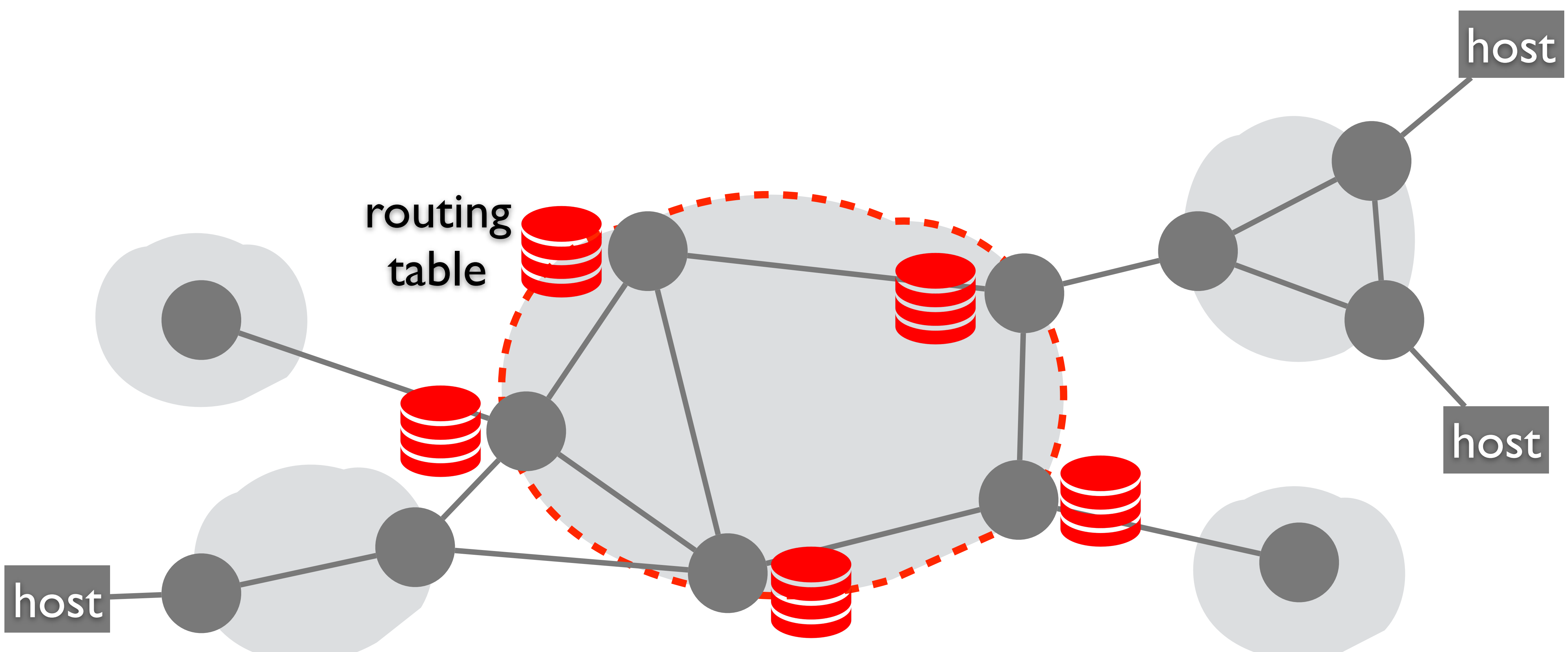
database usage in networking

data synchronization



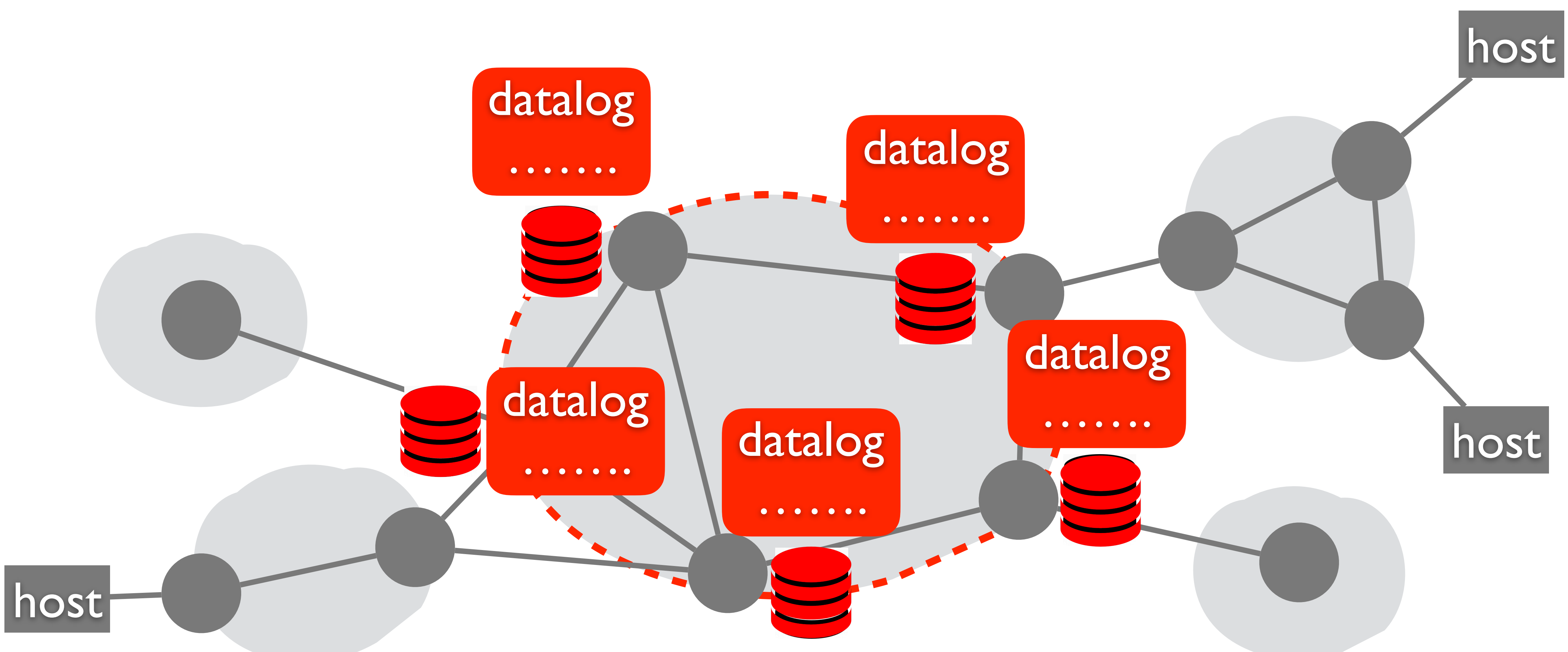
database usage in networking

data query



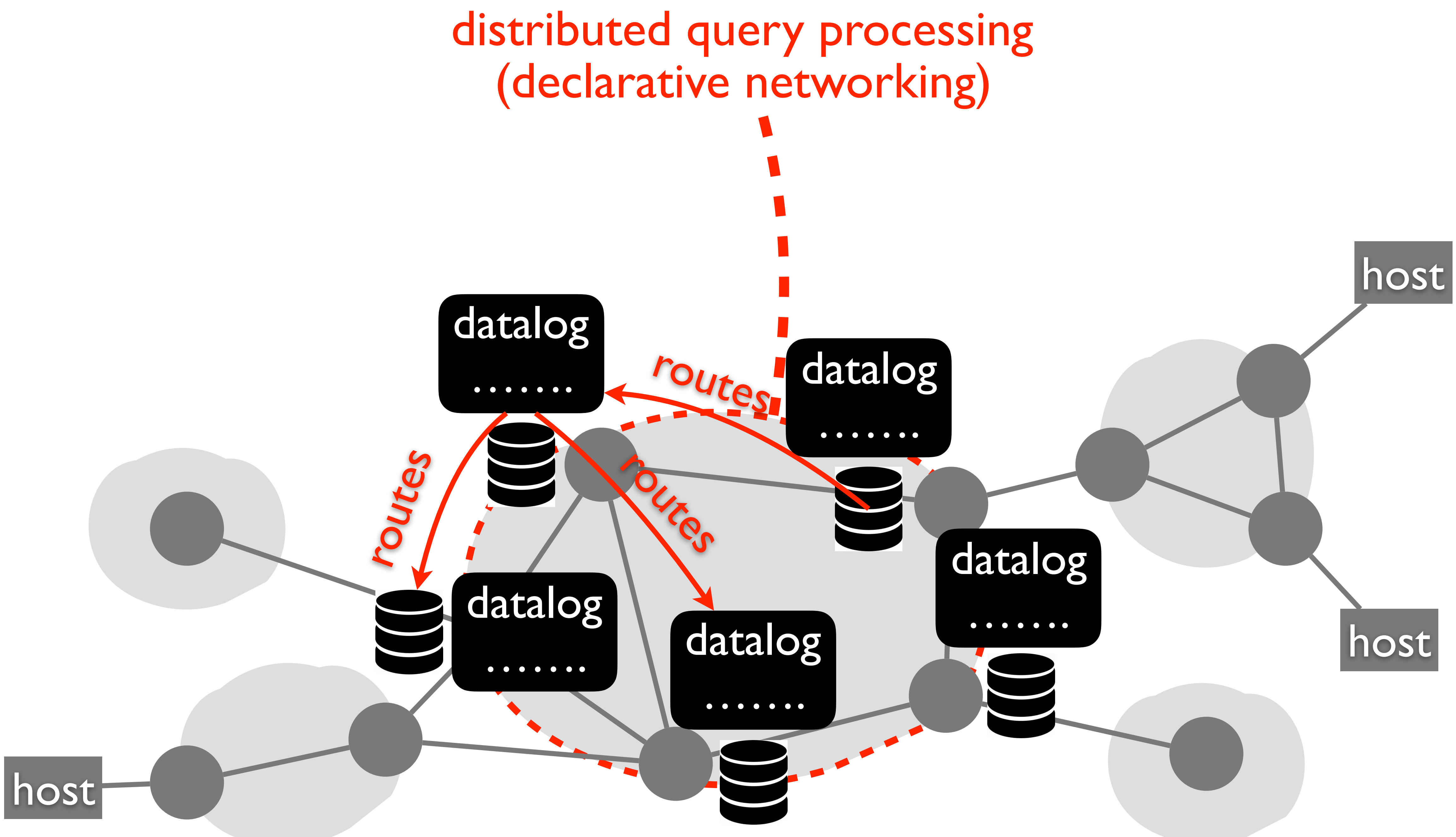
database usage in networking

data query



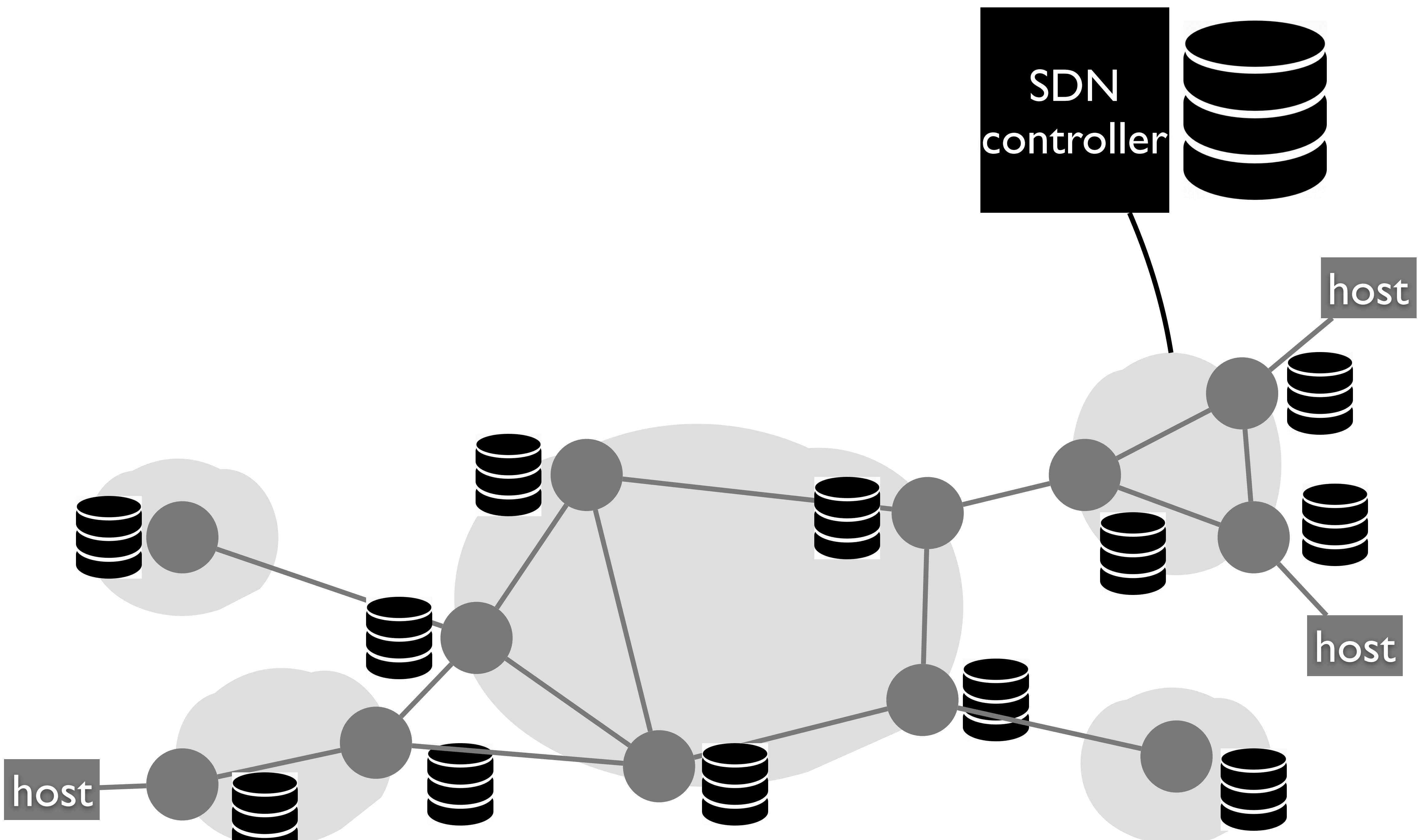
database usage in networking

data query



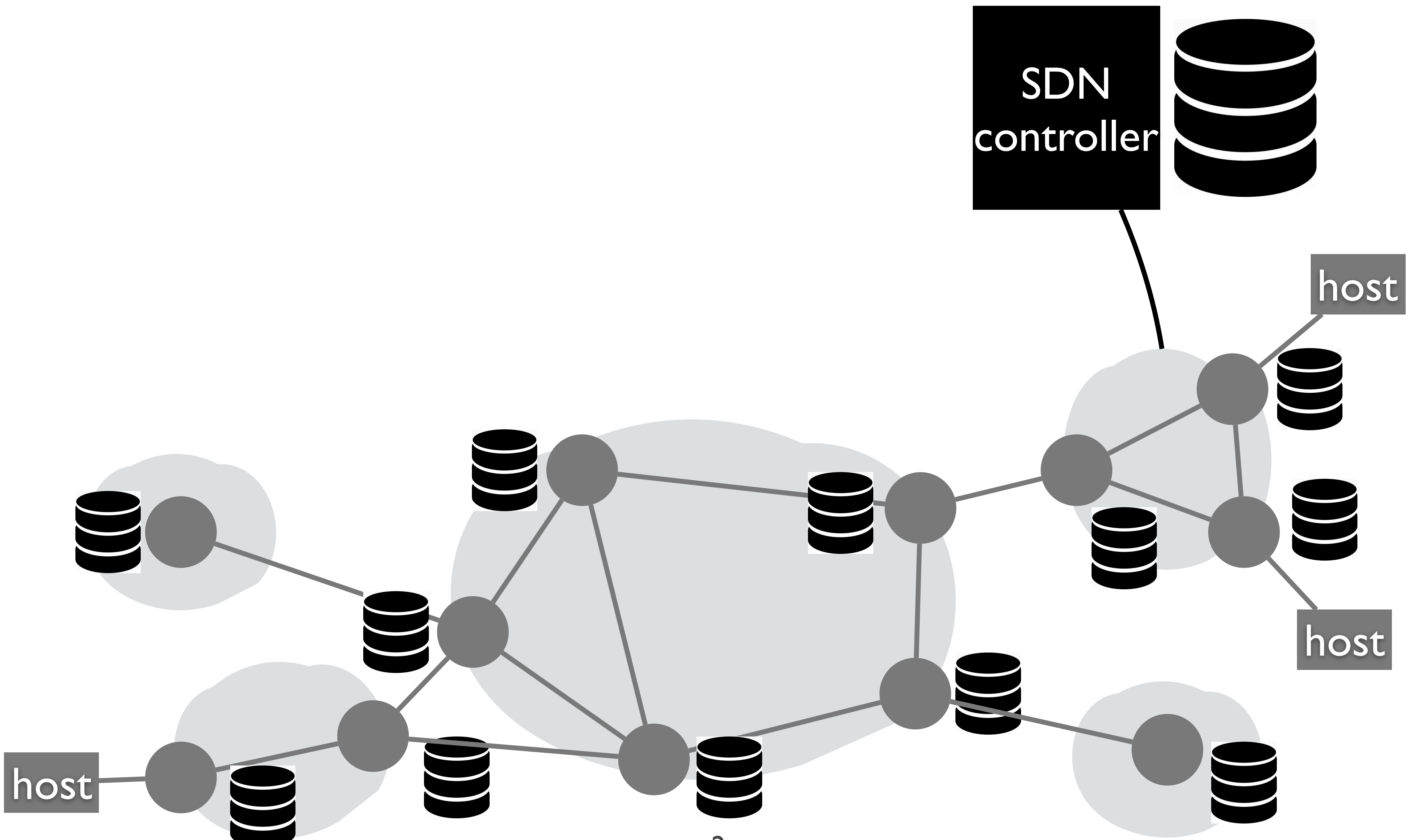
database usage in networking

focus on managing *factual data*



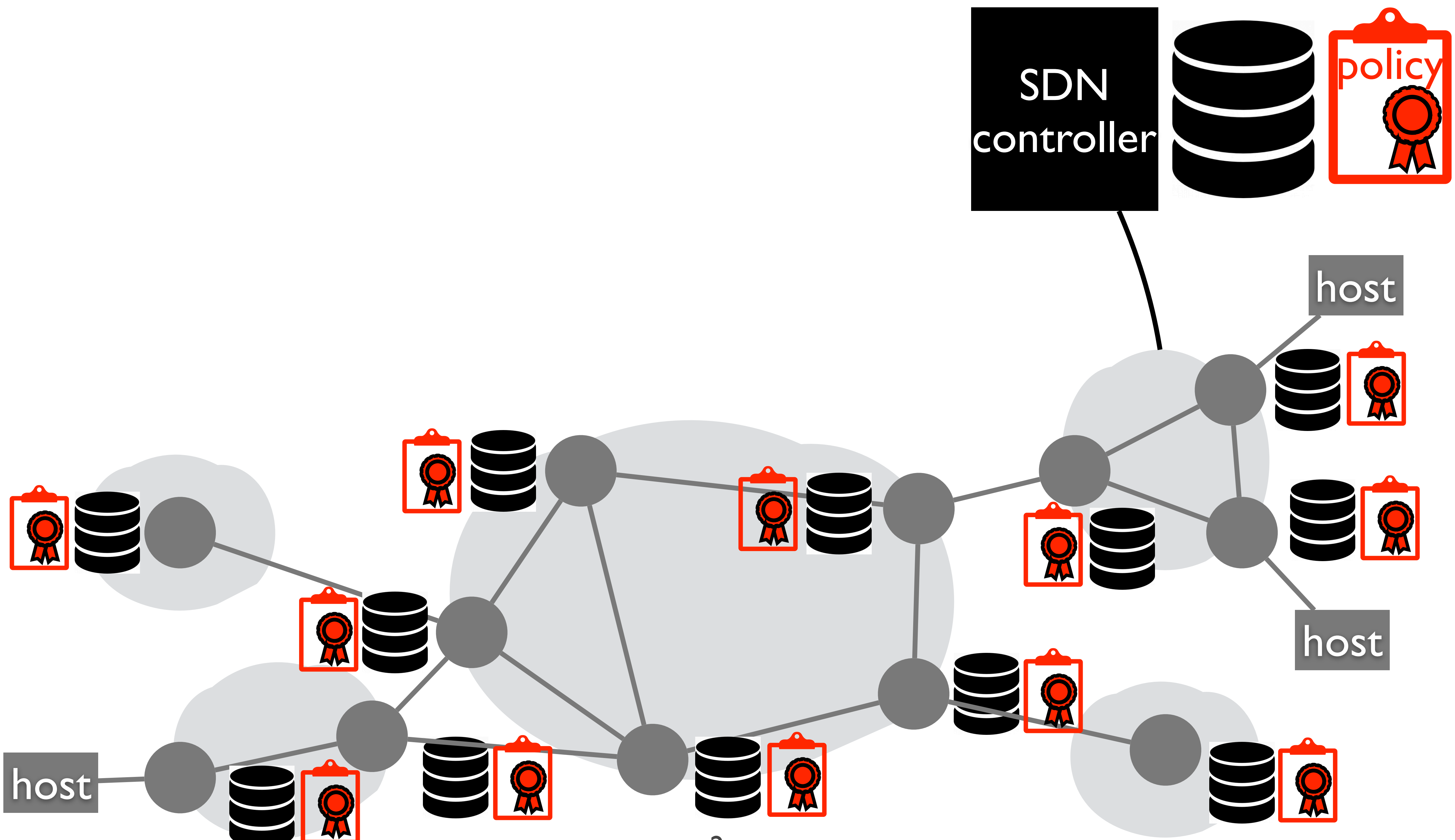
database usage in networking

but database is also renowned for *mediating semantic data*



database usage in networking

but database is also renowned for *mediating semantic data — policies* about what are the acceptable data



this talk

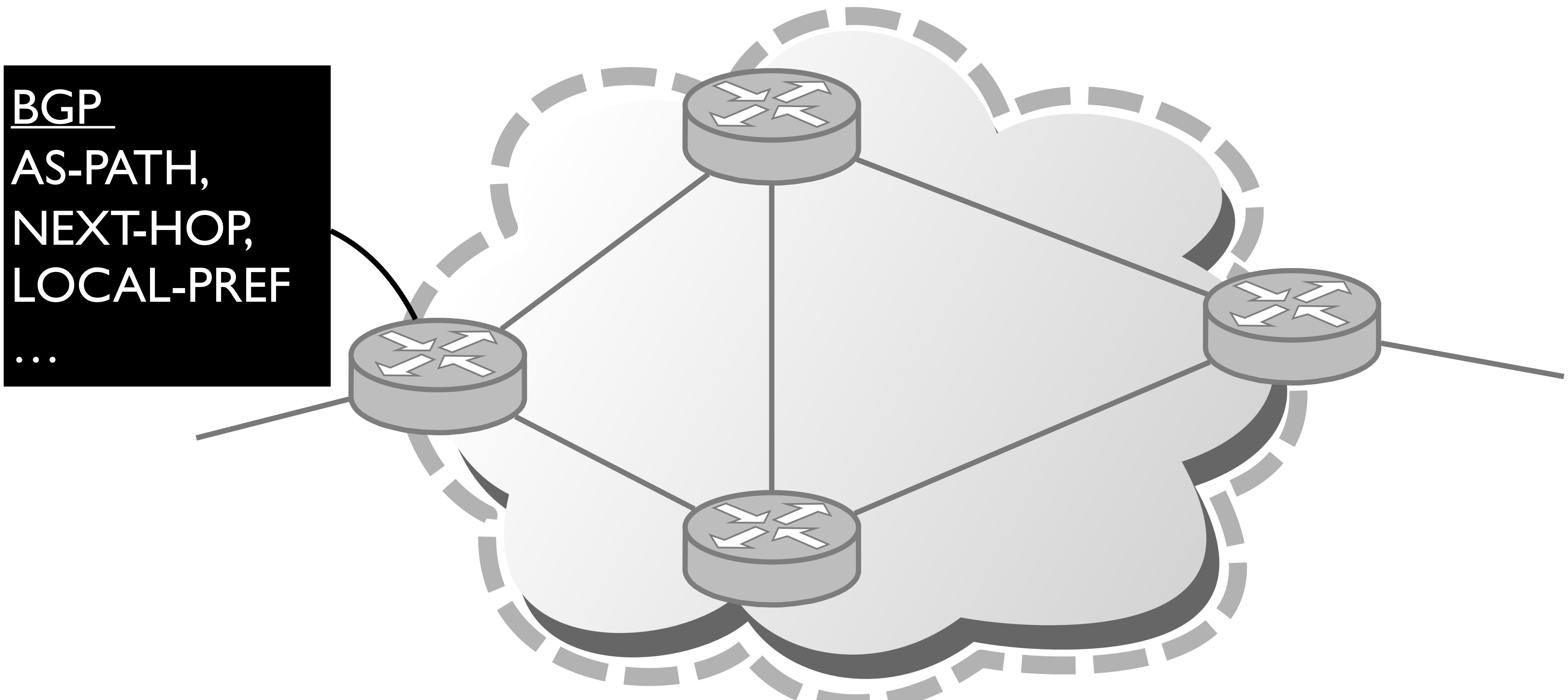
if and how can database systems help with managing network policies that can interact in complex ways

this talk

if and how can database systems help with managing network policies that can interact in complex ways

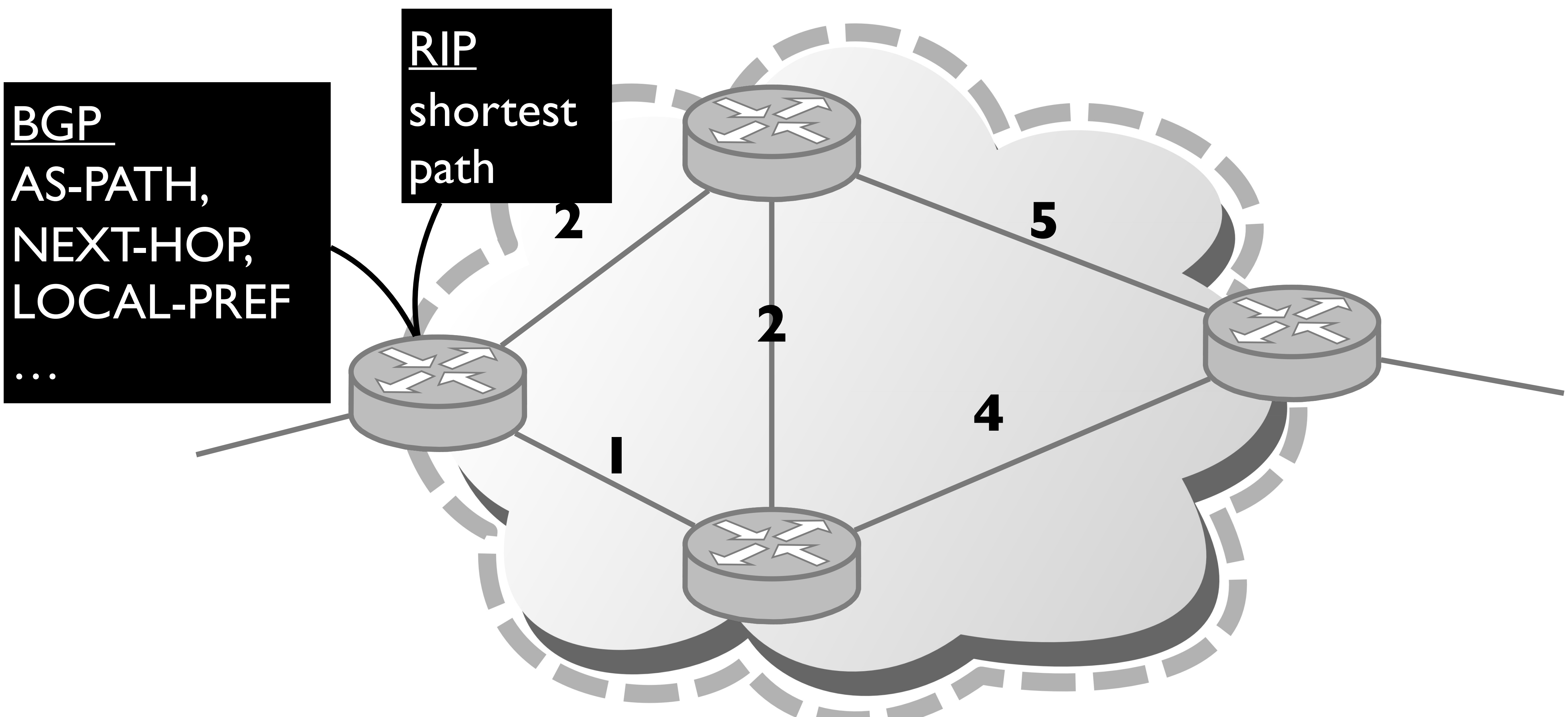
the policy interaction problem

disparate representations buried in the network



the policy interaction problem

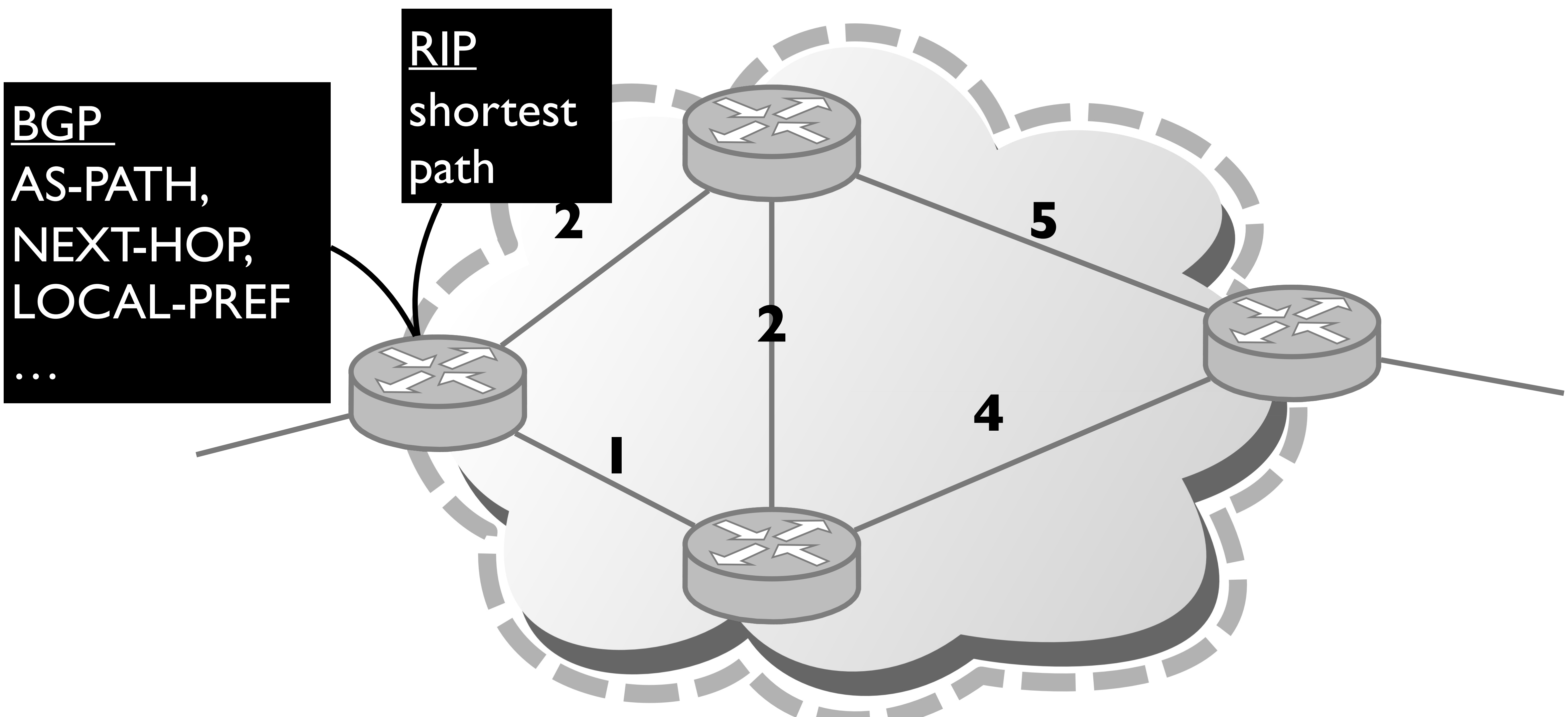
disparate representations buried in the network



the policy interaction problem

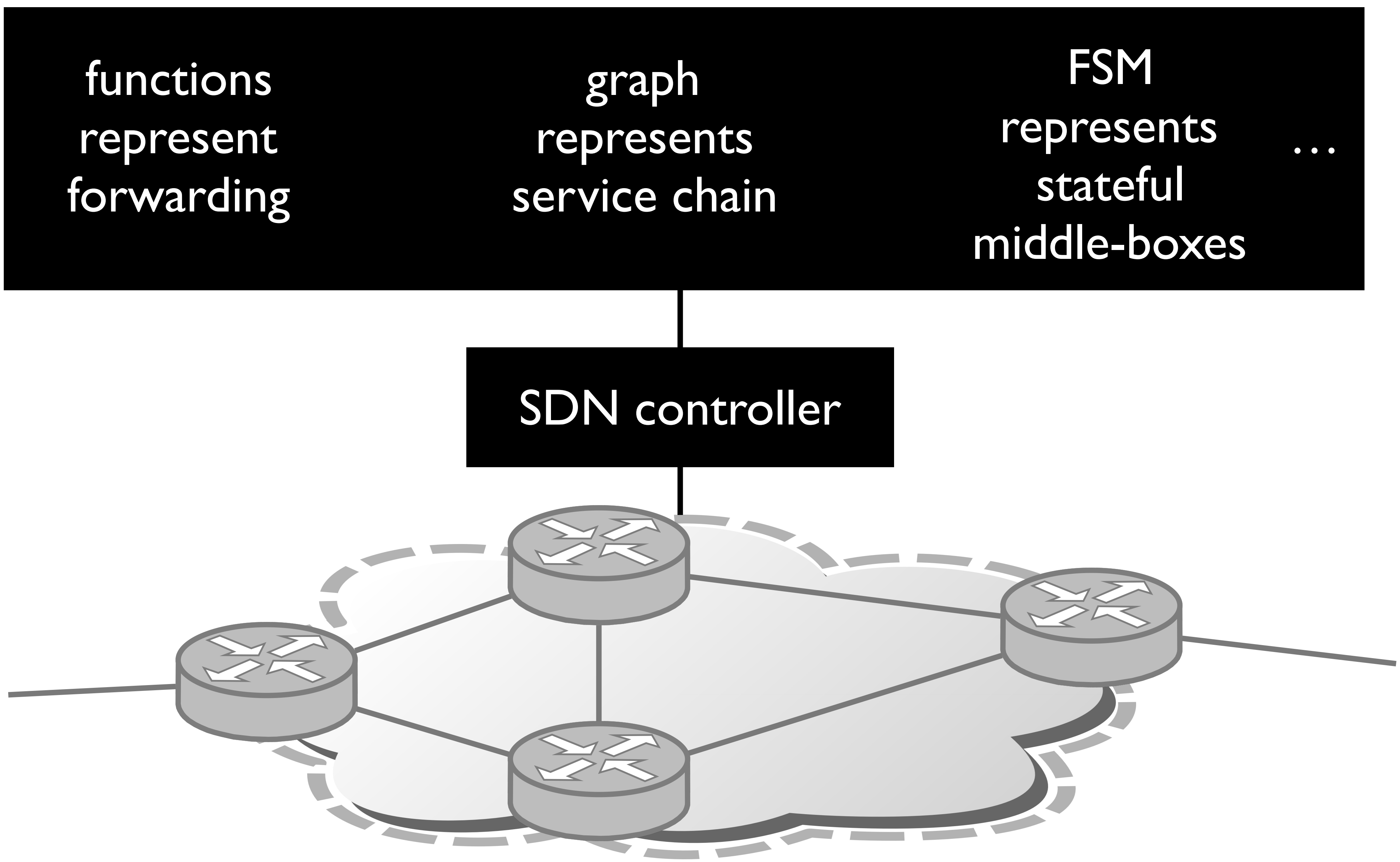
disparate representations buried in the network

- hinders rather than facilitates interaction



the policy interaction problem

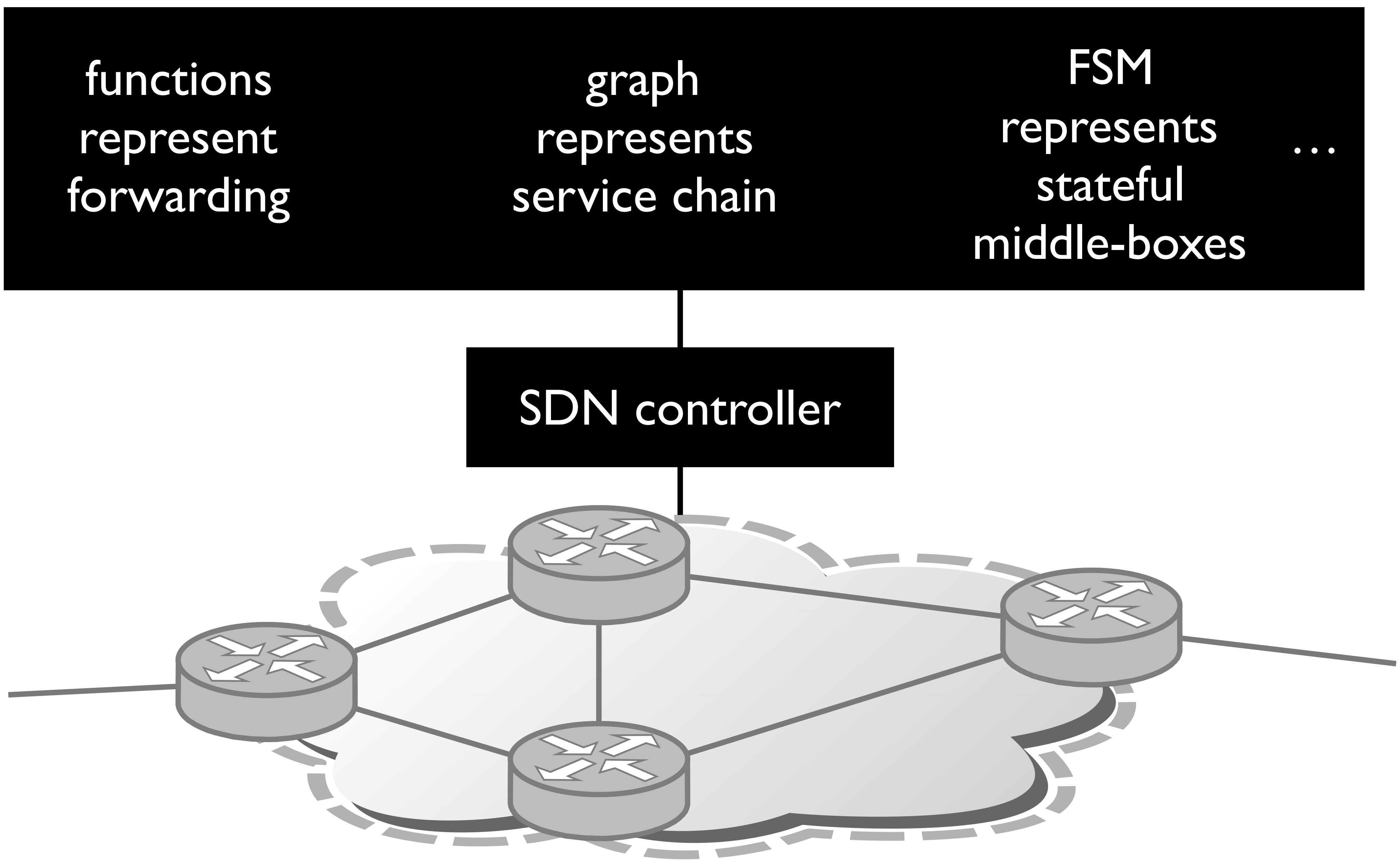
disparate representations buried in the network



the policy interaction problem

disparate representations buried in the network

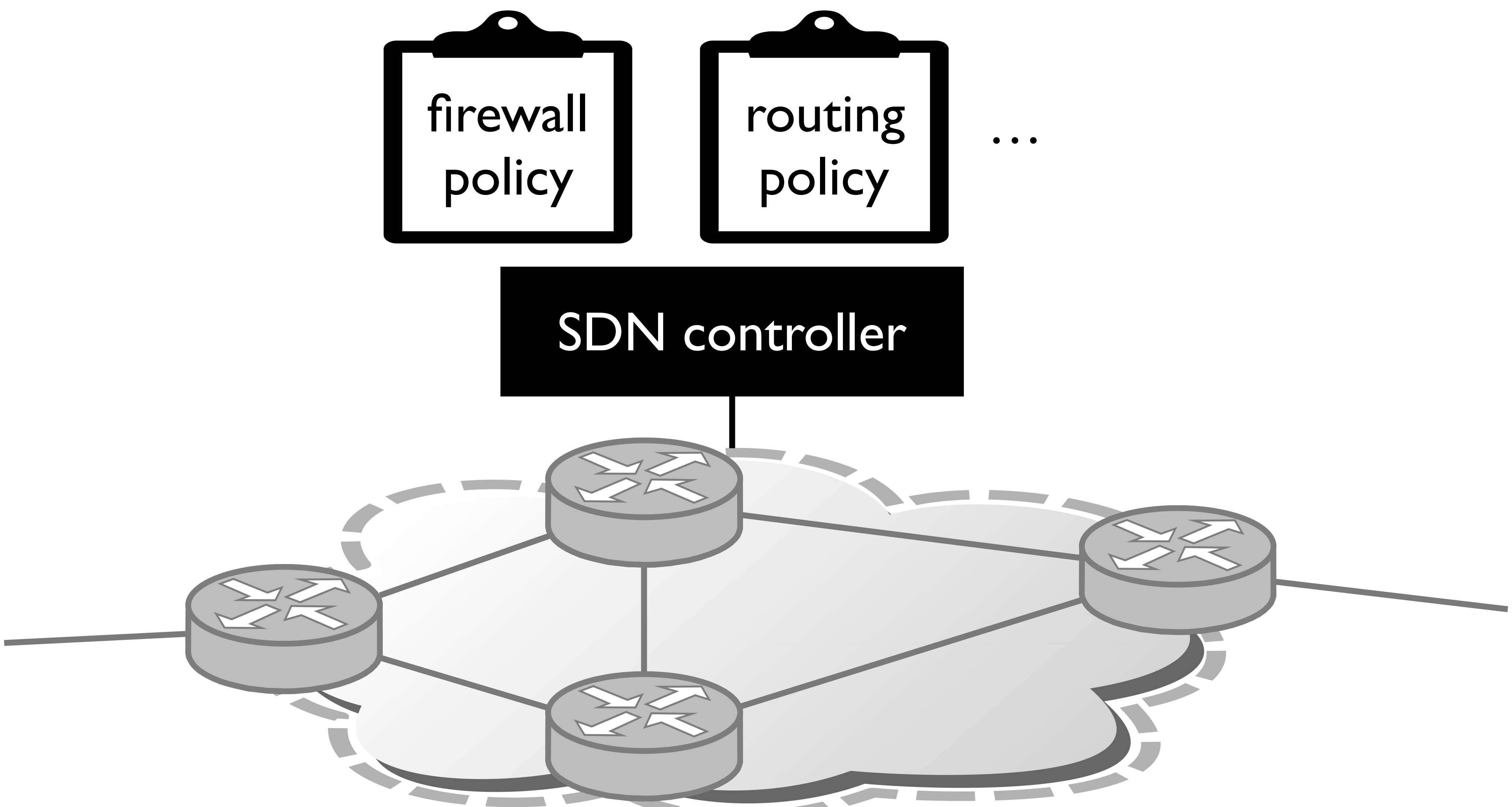
- hinders rather than facilitates interaction



the policy interaction problem

complementary policies in SDNs

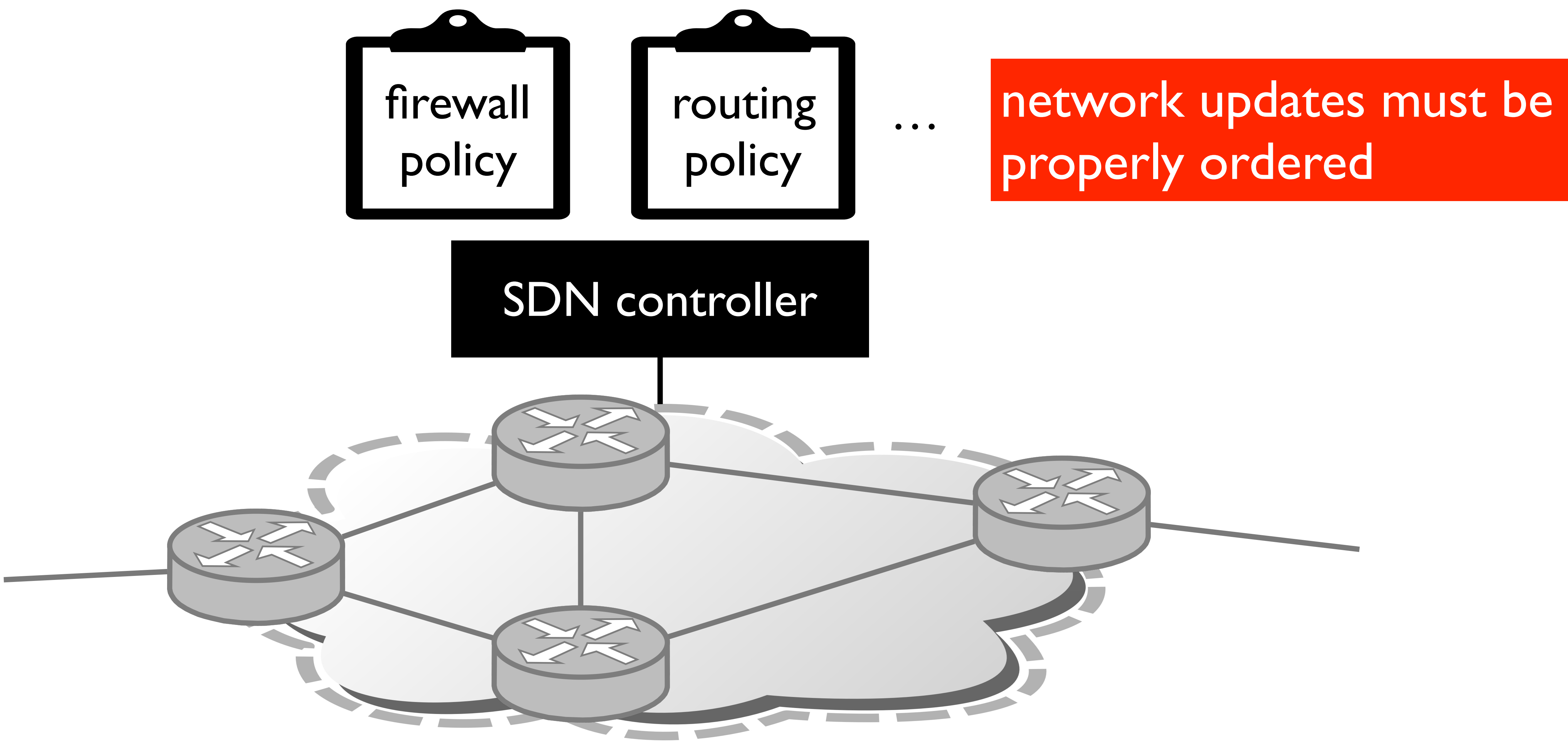
- jointly satisfiable



the policy interaction problem

complementary policies in SDNs

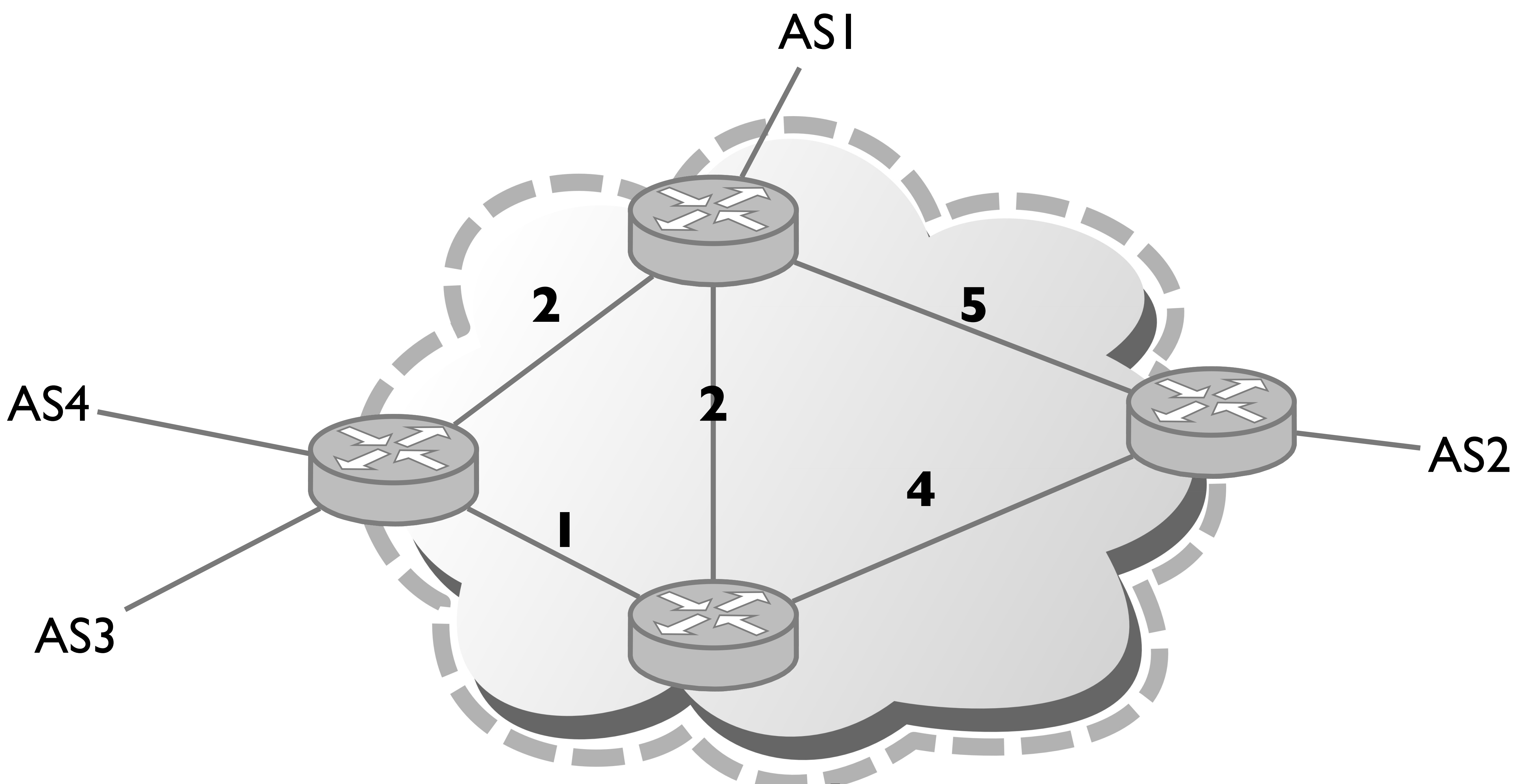
- jointly satisfiable *but not independent*



the policy interaction problem

conflicting policies in inter-domain routing

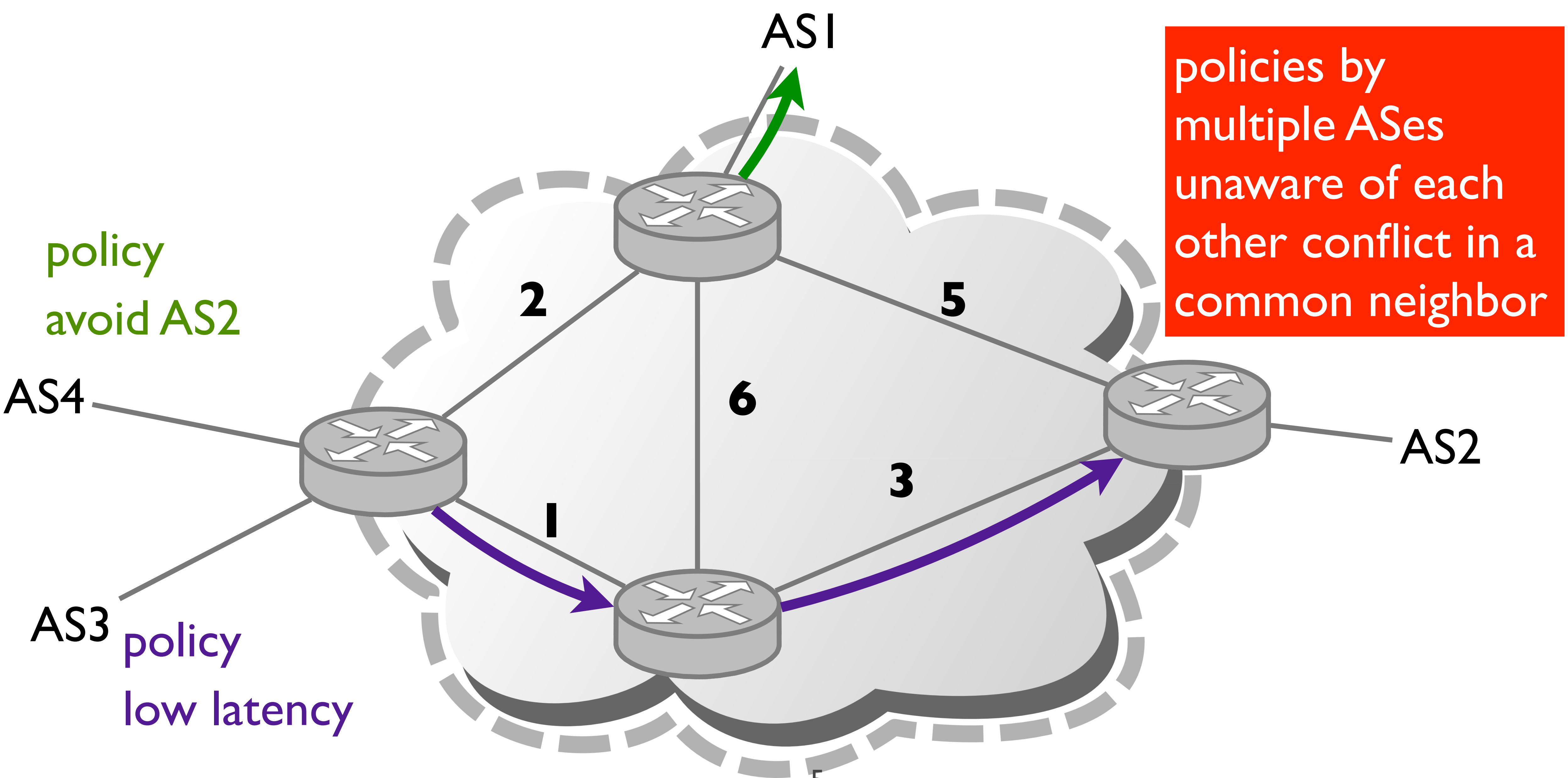
- overlooked conflicts within an autonomous system (AS)



the policy interaction problem

conflicting policies in inter-domain routing

- overlooked conflicts within an autonomous system (AS)
  - AS3 and AS4 attempt to influence route selection of the middle AS



the policy interaction problem

disparate representations buried in the network

complementary policies in SDNs

- jointly satisfiable but not independent

conflicting policies in inter-domain routing

- overlooked conflicts within an AS

a database solution

disparate representations buried in the network ➡ a unified knowledge representation

- policy as integrity constraints (ICs)

complementary policies in SDNs

- jointly satisfiable but not independent

conflicting policies in inter-domain routing

- overlooked conflicts within an AS

a database solution

disparate representations buried in the network ➡ a unified knowledge representation

- policy as integrity constraints (ICs)

complementary policies in SDNs ➡ update orchestrator

- jointly satisfiable but not independent
- dependency analysis of policy ICs

conflicting policies in inter-domain routing

- overlooked conflicts within an AS

policy interaction — a database solution

disparate representations buried in the network ➡ a unified knowledge representation

- policy as integrity constraints (ICs)

complementary policies in SDNs ➡ update orchestrator

- jointly satisfiable but not independent

- dependency analysis of policy ICs

conflicting policies in inter-domain routing ➡ policy negotiator

- overlooked conflicts within an AS

- deriving and merging impacts of policy ICs

a unifying representation

network state — factual data — as relations

- example schema

```
% intradomain tables
tp(sid,nid)                  % topology
rm(fid,sid,nid)              % end-to-end reachability (matrix)
cf(fid,sid,nid)              % configuration (forwarding table)
path(pv,cost,iid,eid)        % internal path

% interdomain tables
aspath(did,rid,apv)          % AS level path
```

a unifying representation

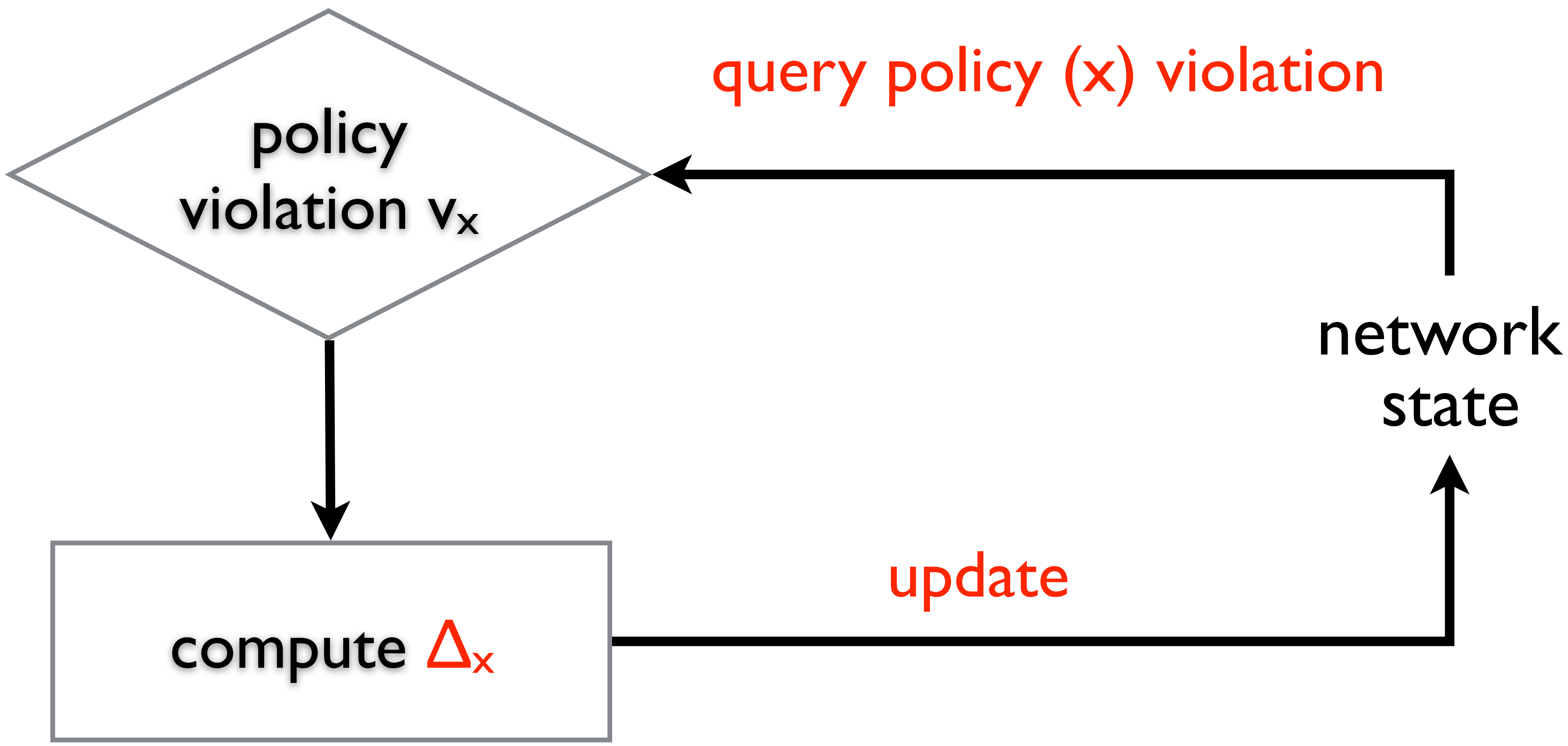
policies — semantic data — as integrity constraints (ICs)

- *denial* form: `:- b1, b2, ..., bn` ($\bot \leftarrow b_1 \wedge b_2 \wedge \ldots \wedge b_n$.)
- meaning: `b1,b2, ...,bn` cannot be simultaneously true.
- example

```
% routing policy
IC1 :- ¬rm(F,S,D),cf(F,X,Y).
IC2 :- rm(F,S,D),¬cf(F,X,Y).

% security policy
IC3 :- rm(F,S,D),blacklist(S,D).
```

relating network state & policy

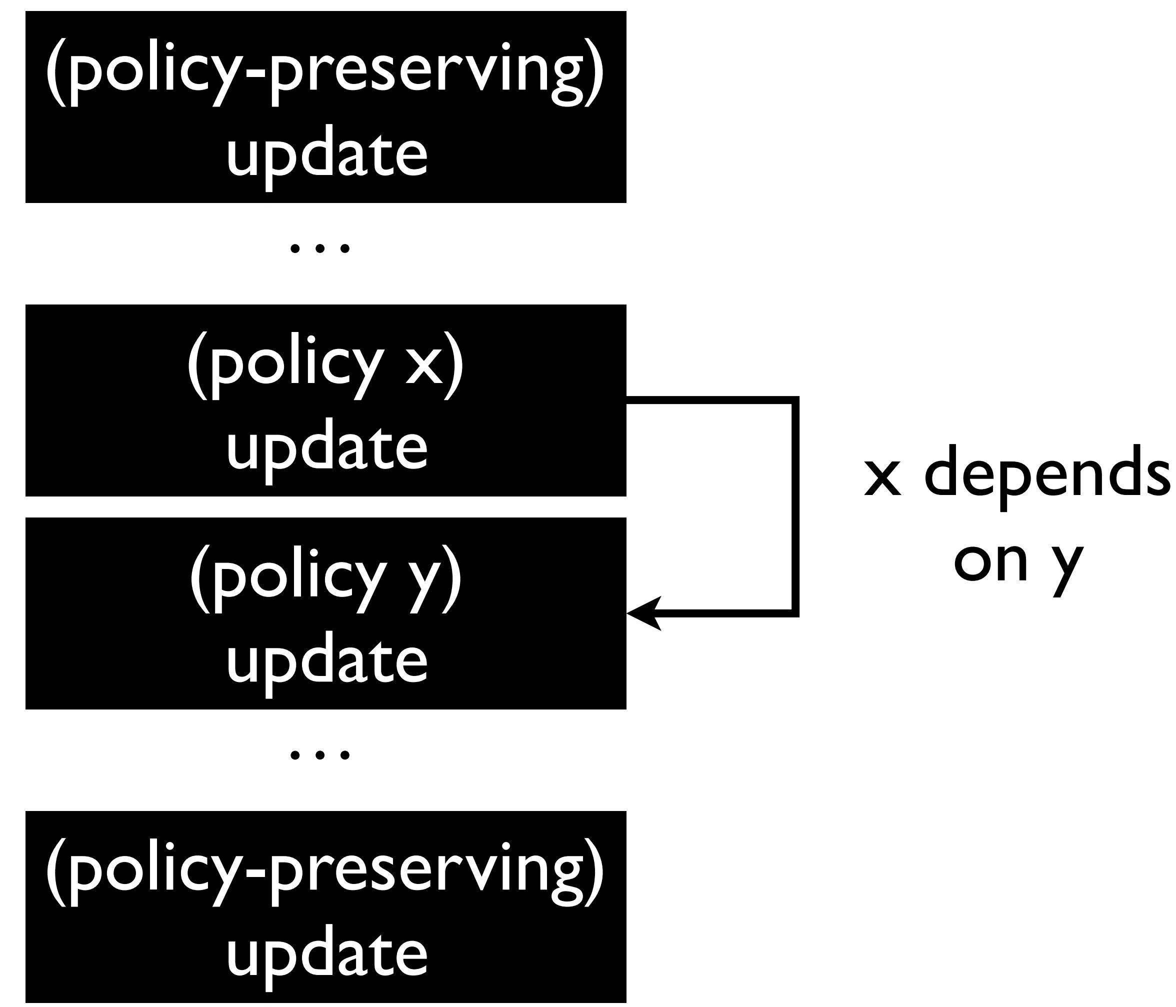


update orchestrator

manage complementary policies in SDNs

- multiple disjoint policies oversee a single task

arrange network updates into a semantic layering

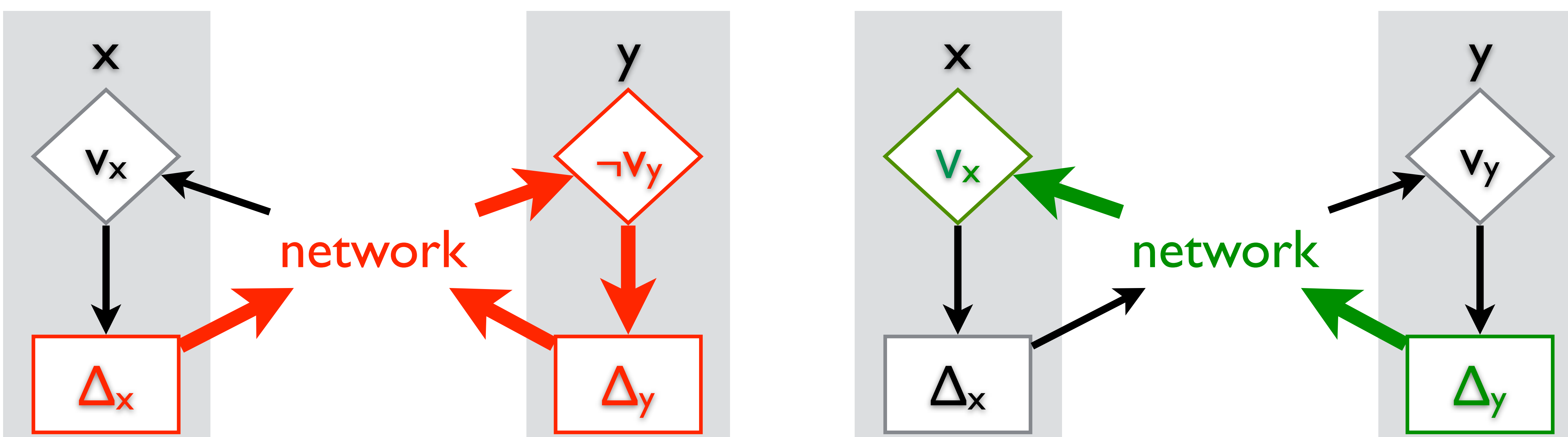


semantic dependency

policy x depends on y if

x update can violate y policy and trigger y action

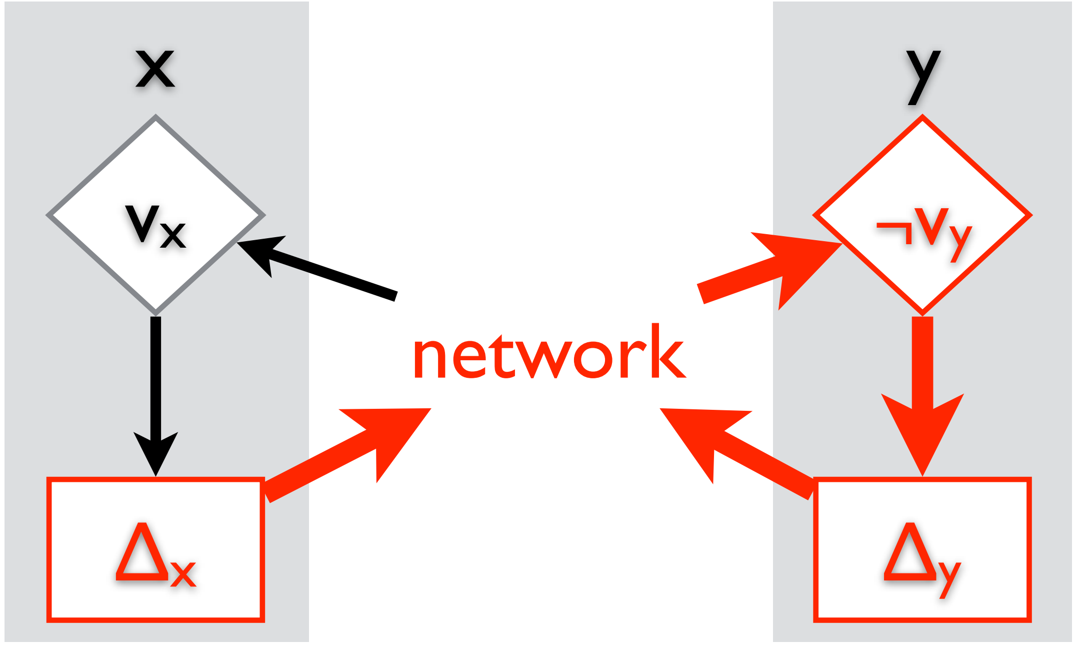
but y can not affect x



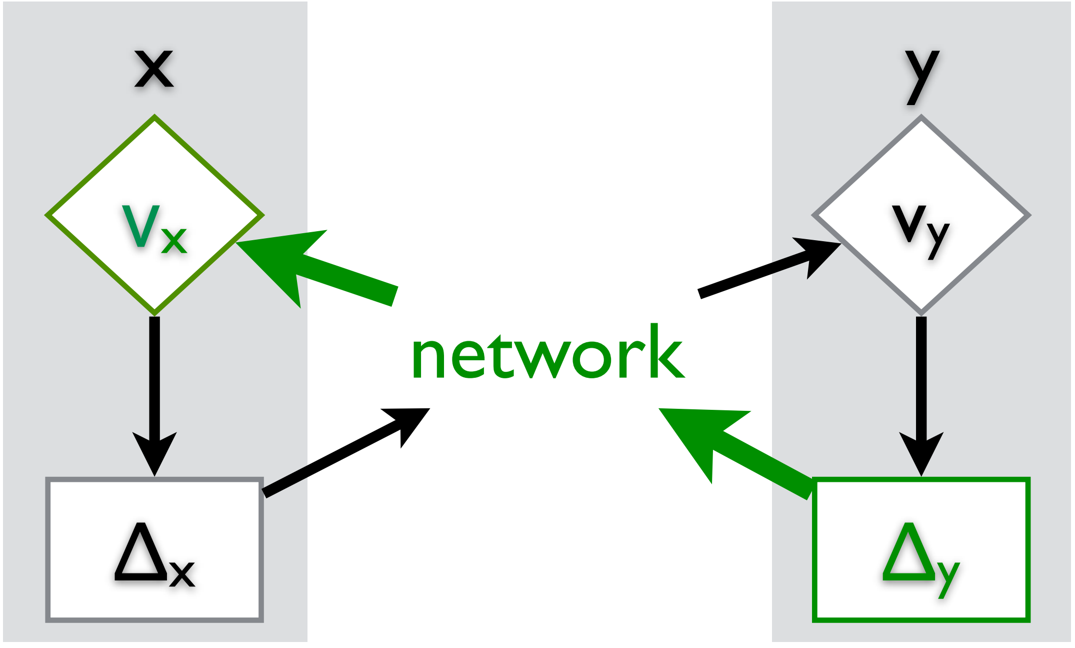
determine semantic dependency

dependency analysis by satisfiability reasoning

check whether Δ_x can alter the result of v_y



check whether Δ_y cannot alter v_x,

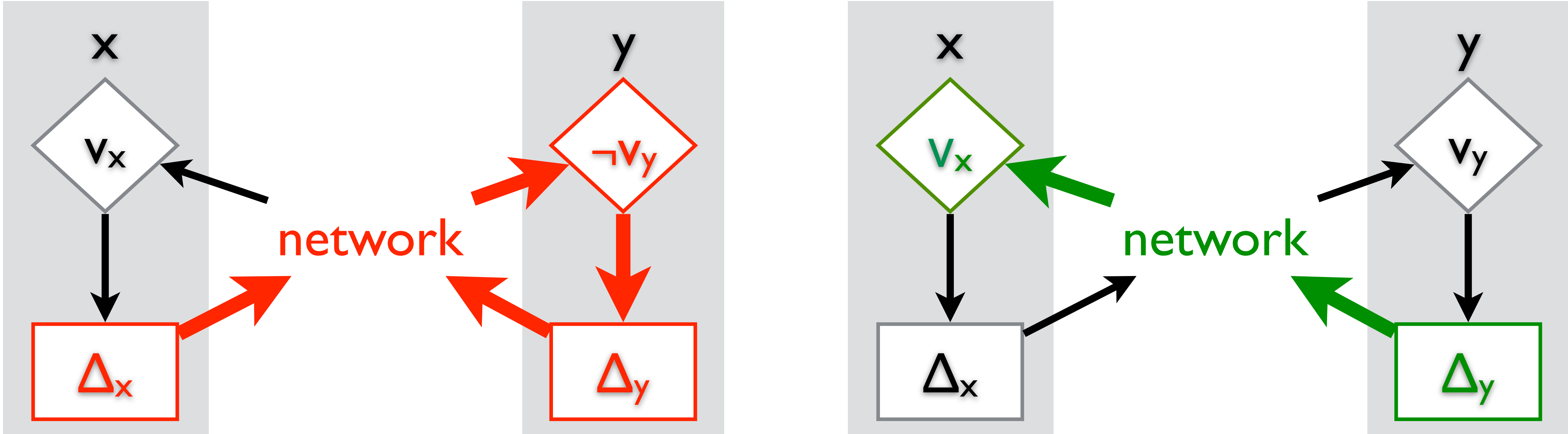


determine semantic dependency

dependency analysis by satisfiability reasoning

(Δ_x condition) $\wedge$ (v_y condition) is SAT

(Δ_y condition) $\wedge$ (v_x condition) is UNSAT



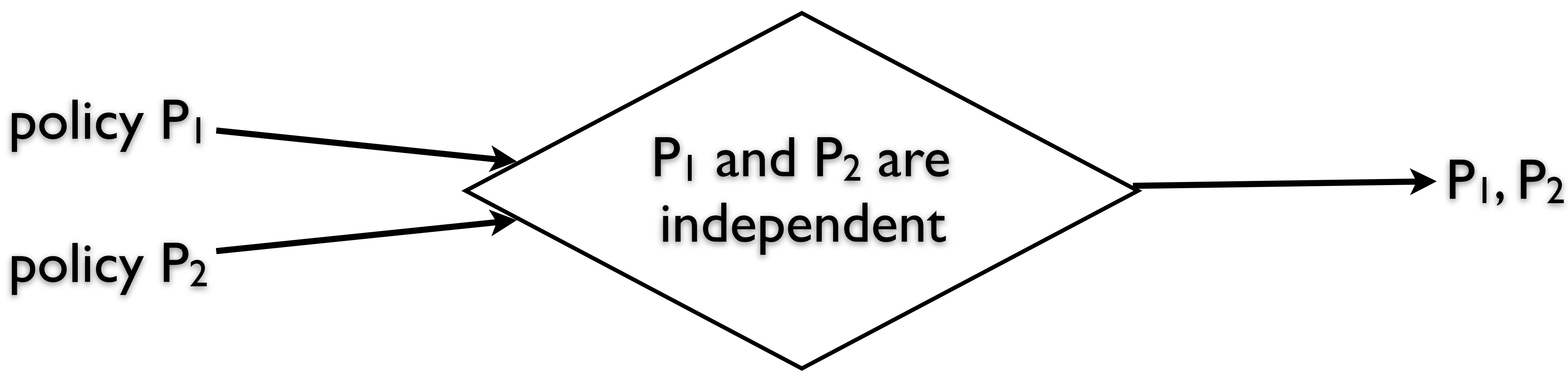
policy negotiator

manage conflicting policies within an AS

- under the influences of multiple neighbors unaware of each other
- derive and merge policy impacts

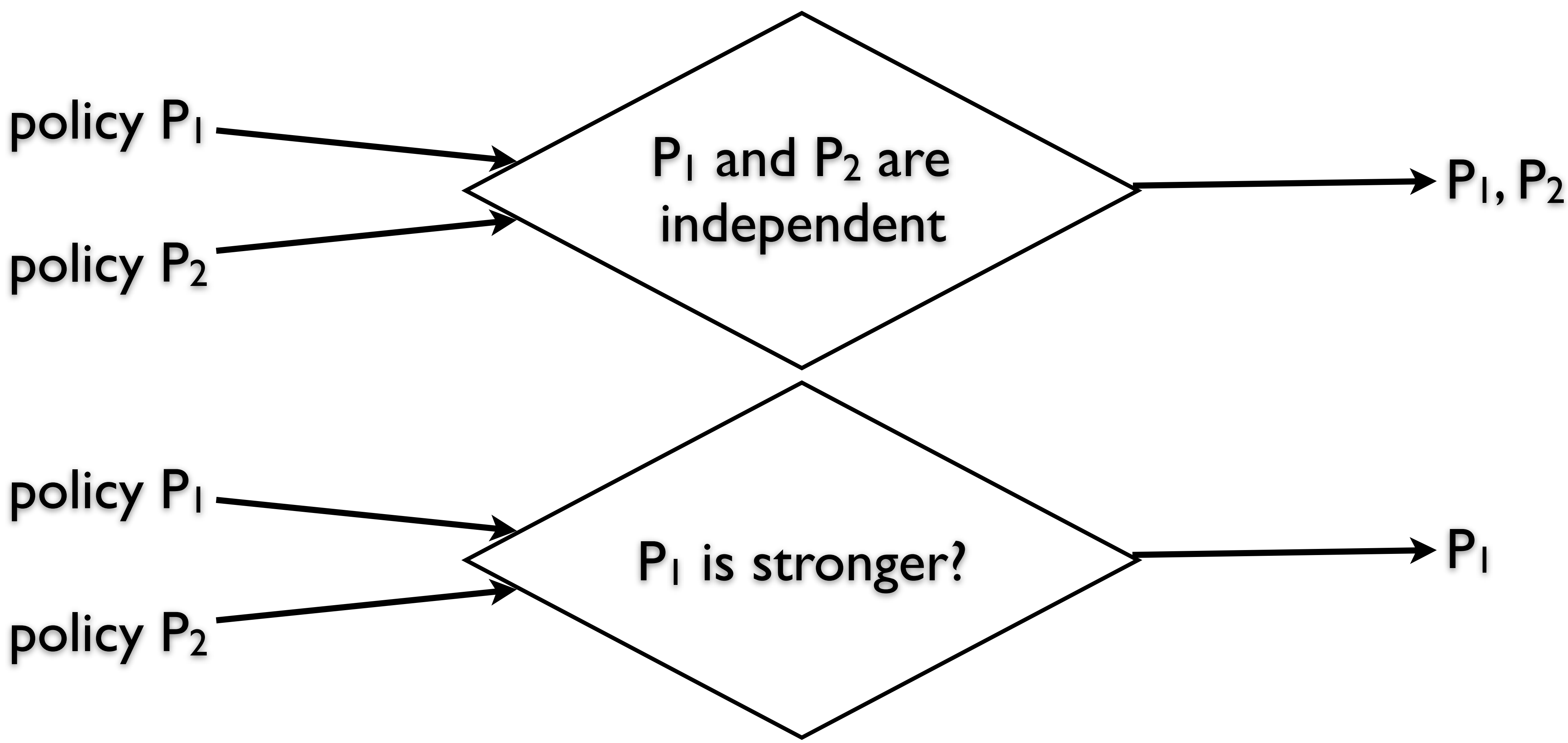
derive and merge policy impacts

an AS under the influences of p_1 and p_2



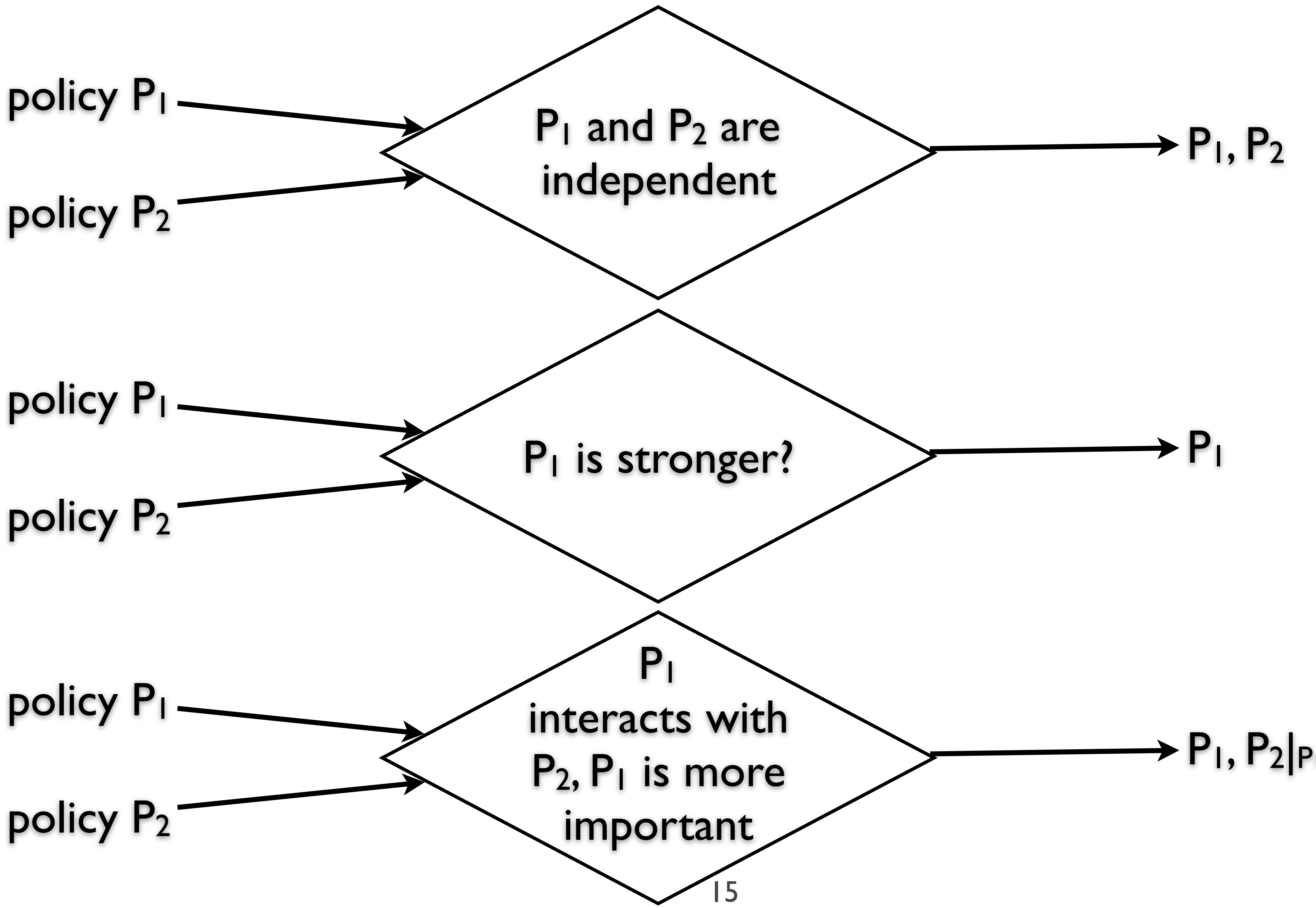
derive and merge policy impacts

an AS under the influences of p_1 and p_2



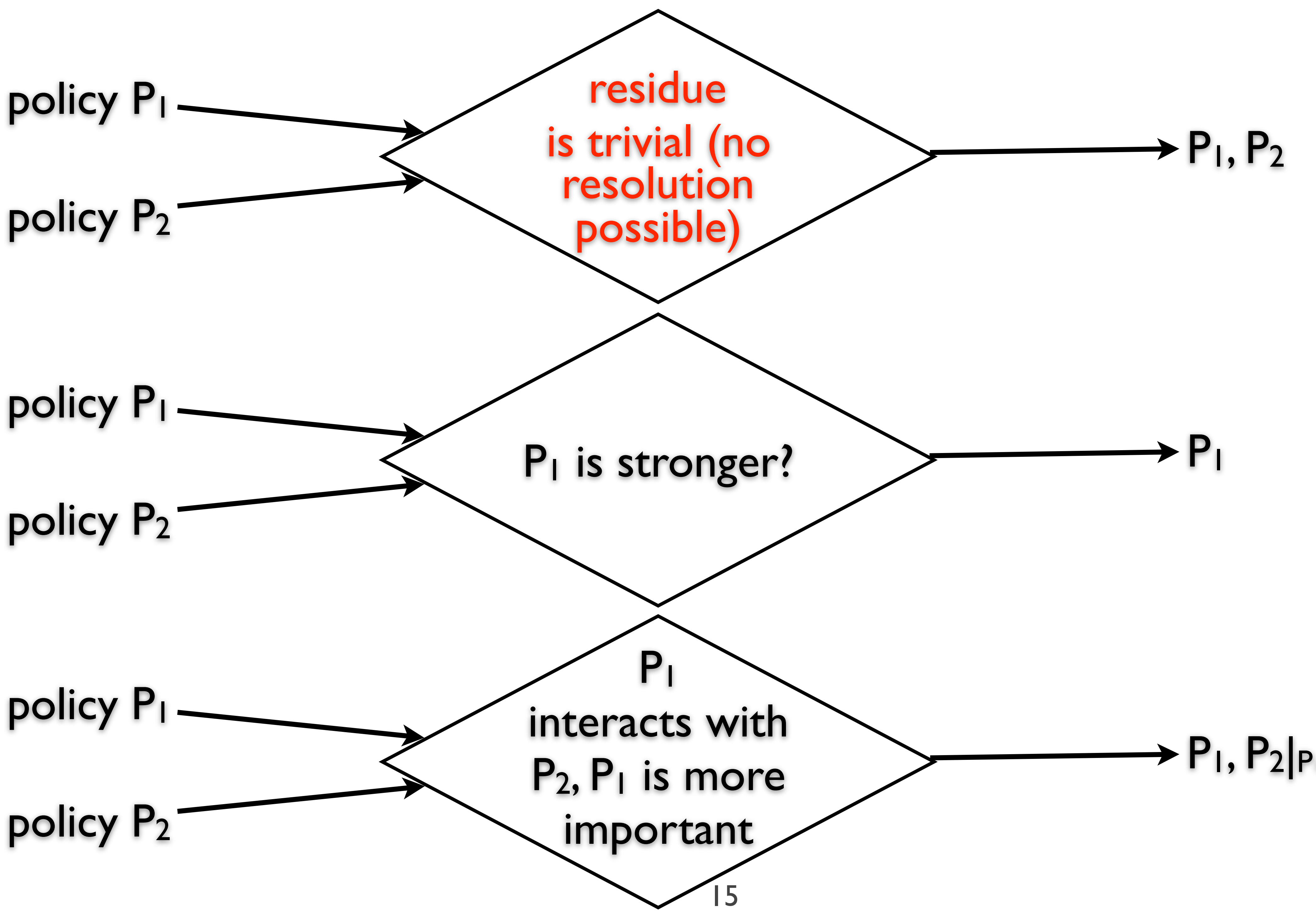
derive and merge policy impacts

an AS under the influences of p_1 and p_2



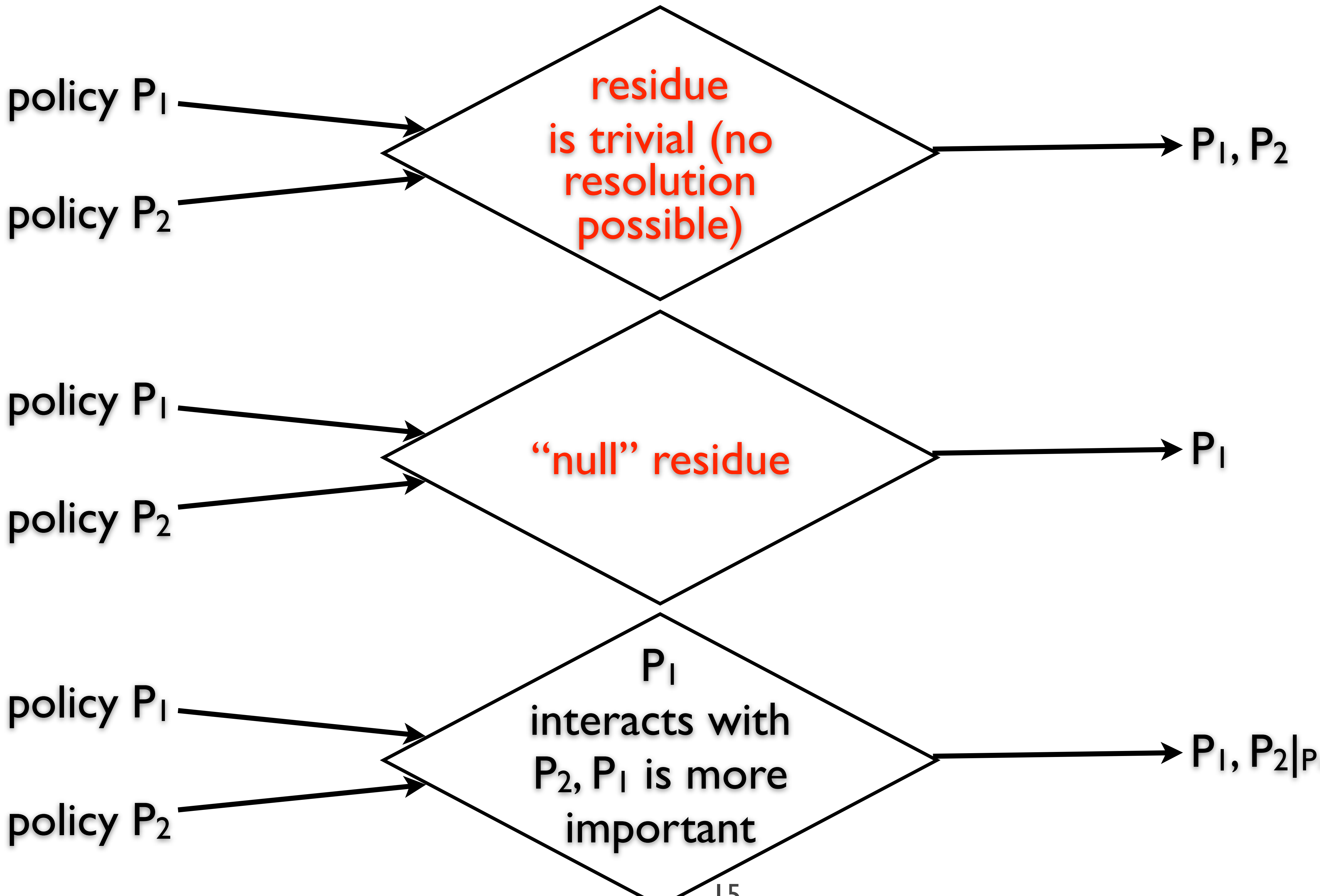
residue method

residue — syntactic fragment that anticipates impact, computed by partial subsumption



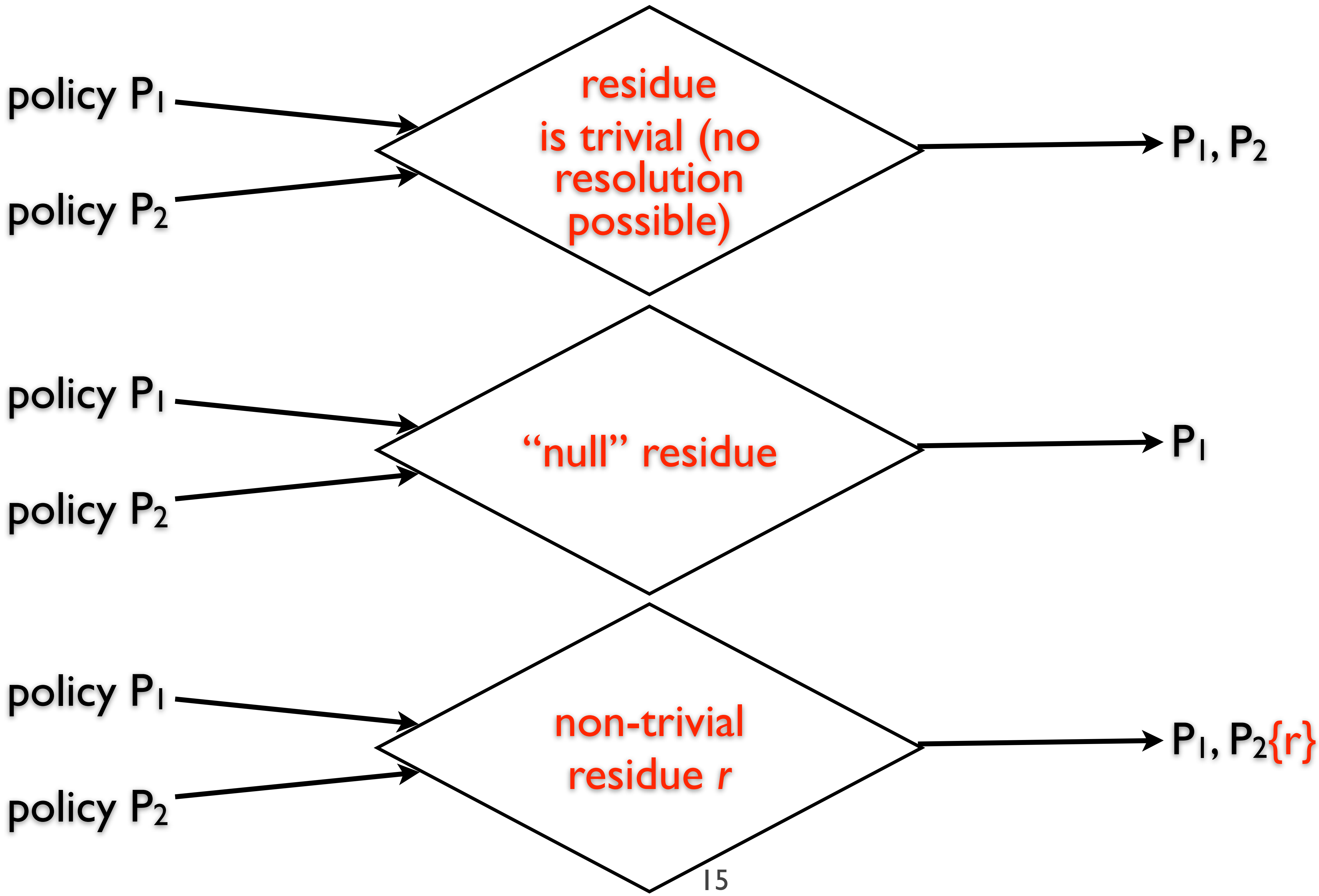
residue method

residue — syntactic fragment that anticipates impact, computed by partial subsumption



residue method

residue — syntactic fragment that anticipates impact, computed by partial subsumption



moving forward

expressiveness of the IC representation

- facilitating template, translating tool

cyclic dependency in SDN

- break cycles

policies are private in interdomain

- obfuscate policies

recap

disparate representations buried in the network ➡ a unified knowledge representation

- policy as integrity constraints (ICs)

complementary policies in SDNs ➡ update orchestrator

- jointly satisfiable but not independent

- dependency analysis of policy ICs

conflicting policies in inter-domain routing ➡ policy negotiator

- overlooked conflicts within an AS

- deriving and merging impacts of policy ICs

thank you

part of "Ravel database-defined networking" ravel-net.org

backup

residue computation by example

%% shortest path

$\mathtt{IC_{sp}}$ `:-` $\mathtt{r_o(x,y,z),\ r_i(x,y_2,z_2),\ l(z)>l(z_2).}$

%% explicit path policy

$\mathtt{IC_{ep}}$`:` `z=a :-` $\mathtt{r_o(x,y,z),}$ `x=d.`

$\mathtt{IC_{ep}}$

`z=a :-` $\mathtt{r_o(x,y,z),}$ `x=d.`

elements of $\mathtt{IC_{sp}}$

`:-` $\mathtt{r_o(x,y,z).}$

$\boxed{\mathtt{z=a\ :-\ x=d.}}$

a non-trivial residue prescribes the impact of $\mathtt{IC_{ep}}$ —
additional conditions that must be taken into account for $\mathtt{IC_{sp}}$

$\mathtt{IC_{sp}}$ affects $\mathtt{IC_{ep}}$, as anticipated by the residue