Ravel au
pupitre
« Boléro »
luc. albert Morea

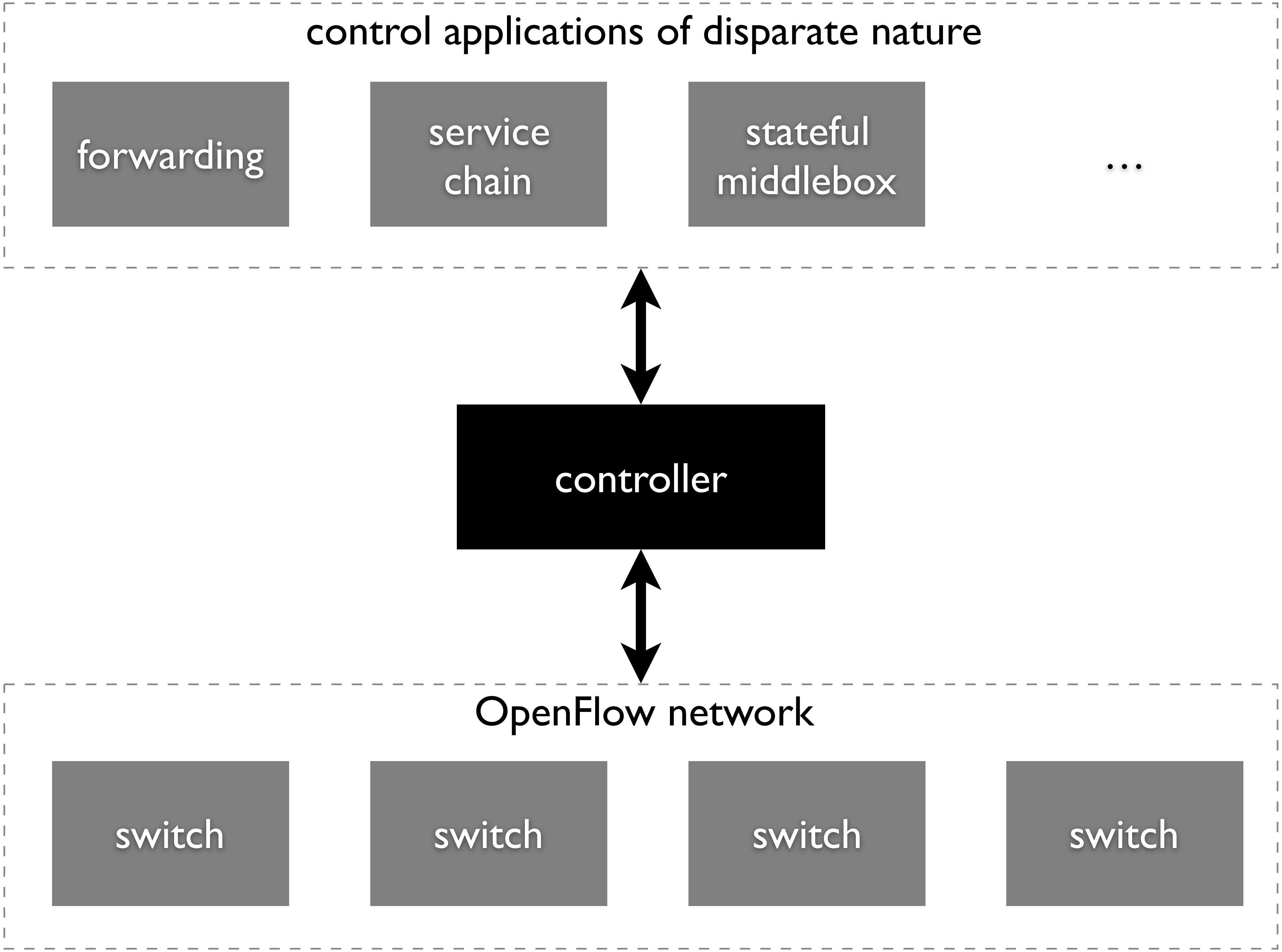
Ravel: a database-defined network

Anduo Wang* Xueyuan Mei† Jason Croft†
Matthew Caesar† Brighten Godfrey†

**Temple University* †*University of Illinois Urbana-Champaign*

March 14, 2016 SOSR'16

software-defined network



control applications of disparate nature

forwarding

service chain

stateful middlebox

…

controller

OpenFlow network

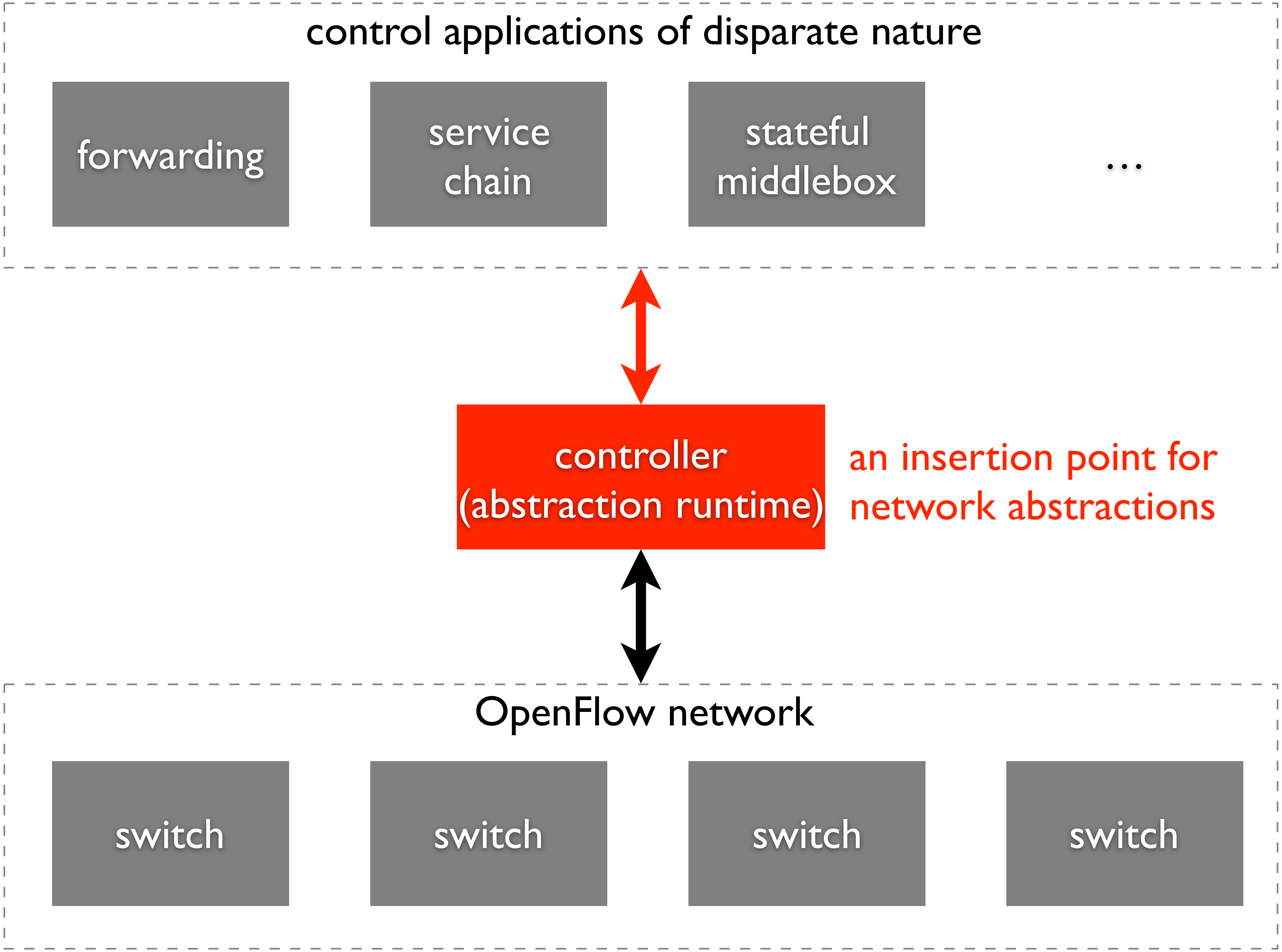
switch

switch

switch

switch

software-defined network



control applications of disparate nature

forwarding

service chain

stateful middlebox

...

controller (abstraction runtime)

an insertion point for network abstractions

OpenFlow network

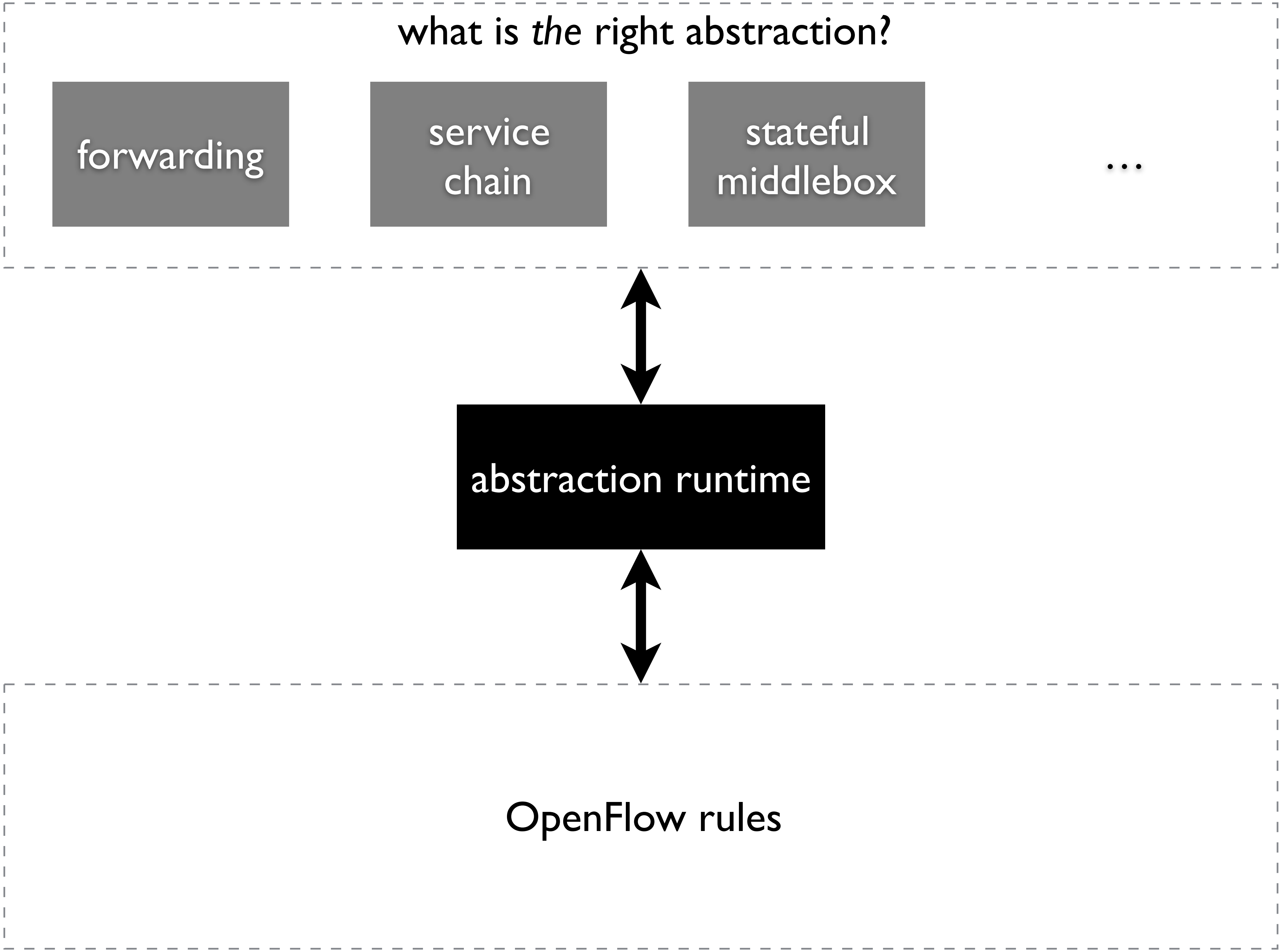
switch

switch

switch

switch

abstractions



what is *the* right abstraction?

forwarding

service chain

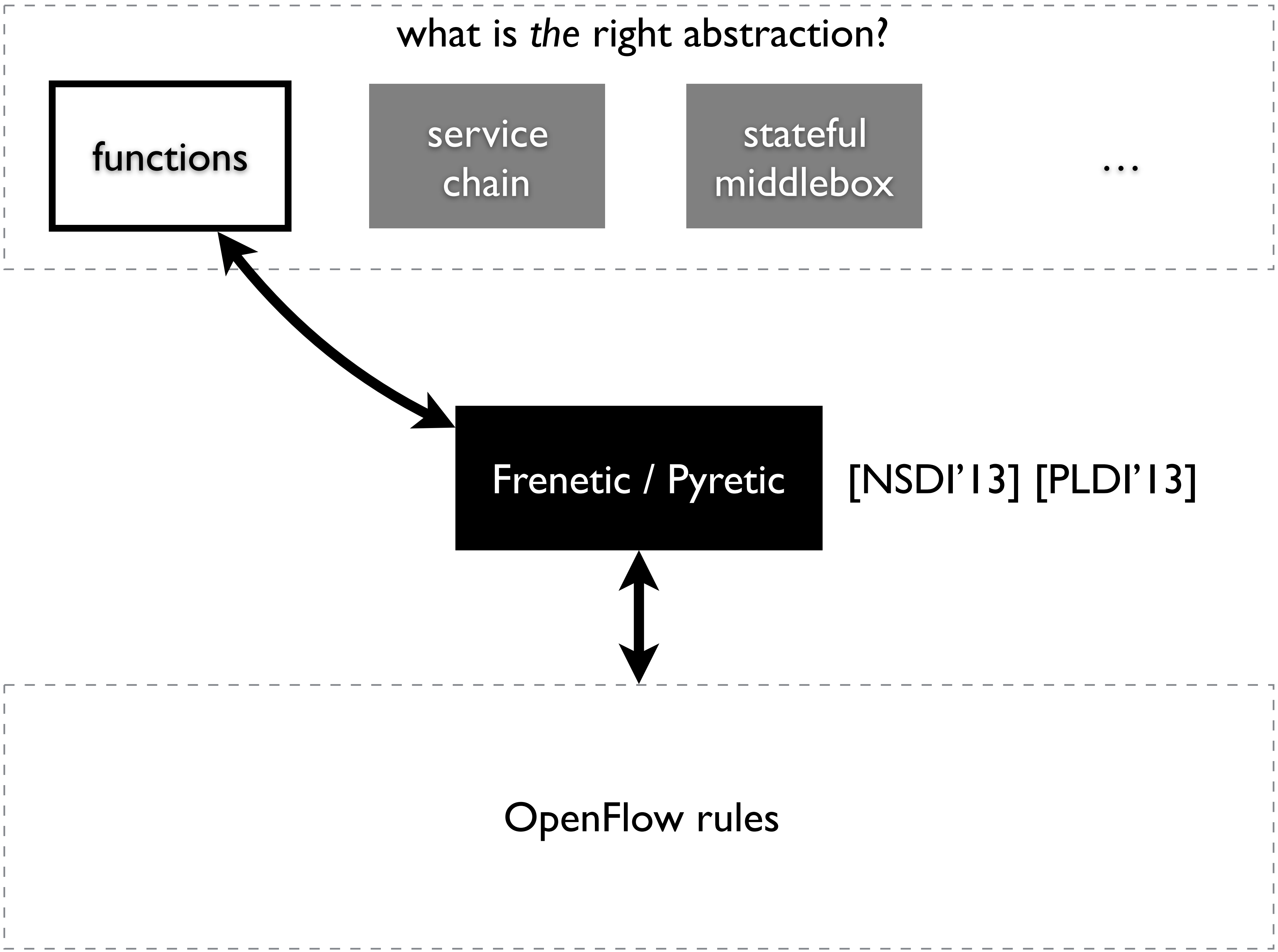
stateful middlebox

…

abstraction runtime

OpenFlow rules

abstractions



what is *the* right abstraction?

functions

service chain

stateful middlebox

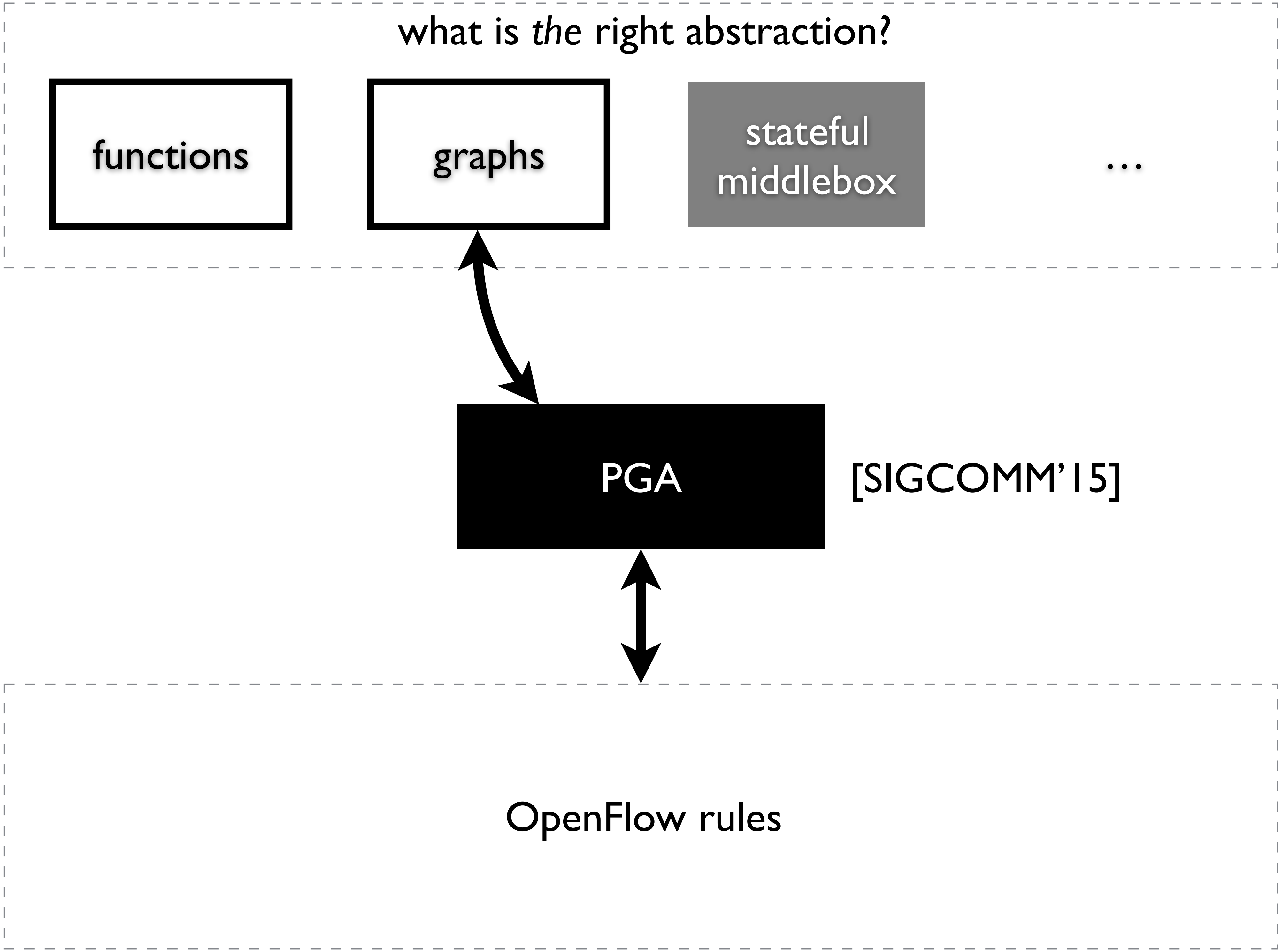
…

Frenetic / Pyretic

[NSDI’13] [PLDI’13]

OpenFlow rules

abstractions



what is *the* right abstraction?

functions

graphs

stateful middlebox

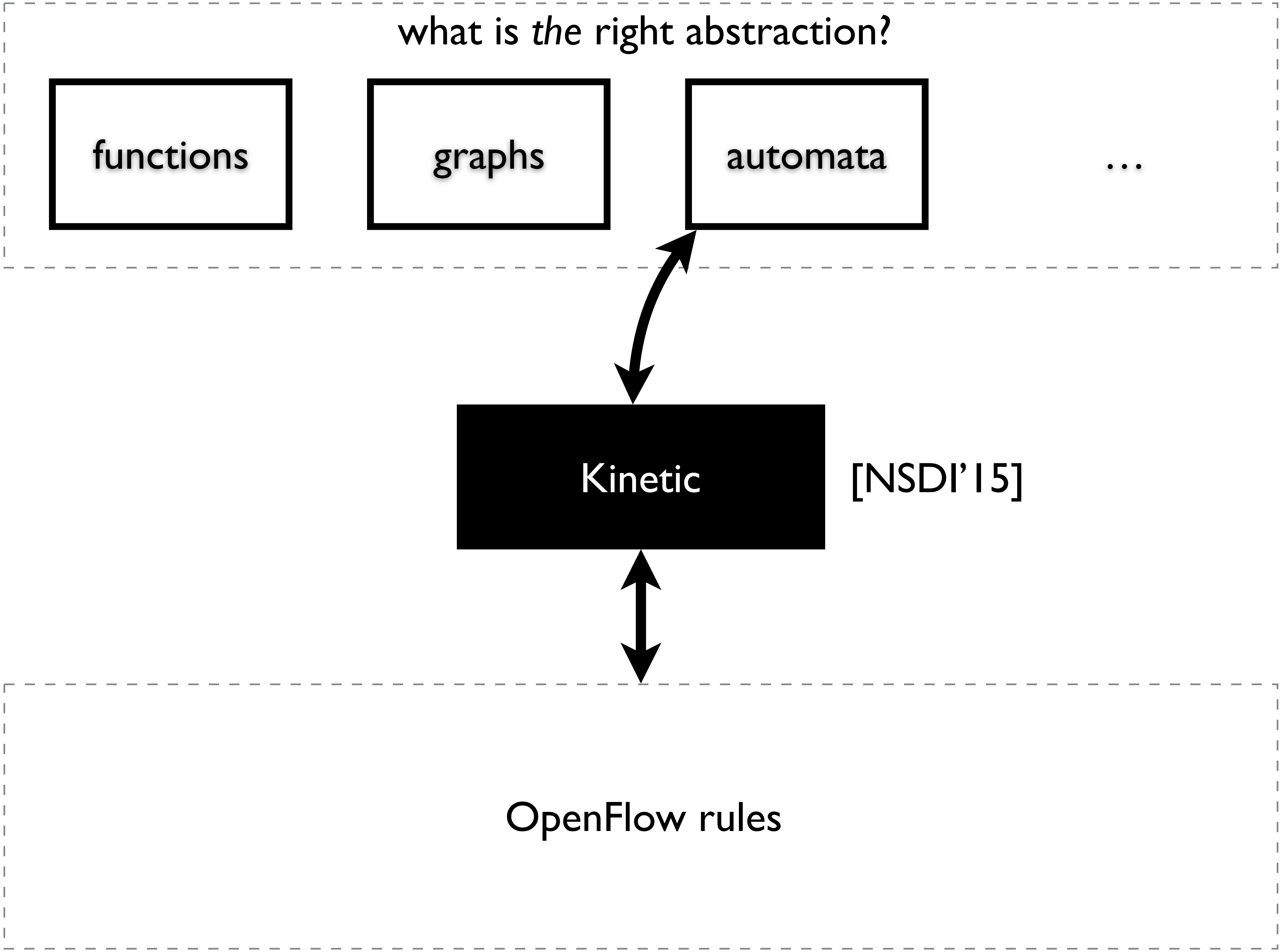
…

PGA

[SIGCOMM'15]

OpenFlow rules

abstractions



what is *the* right abstraction?

functions

graphs

automata

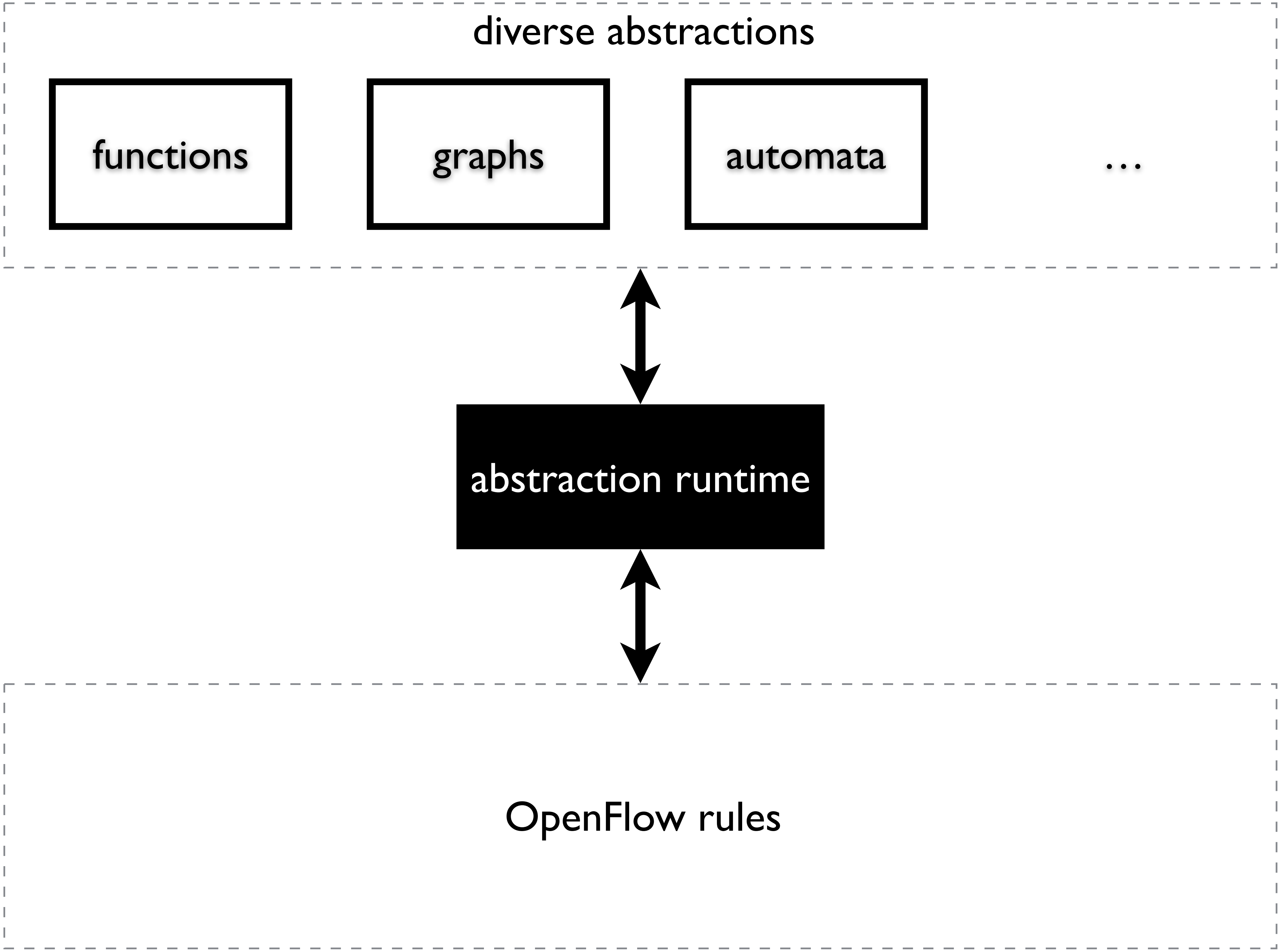
…

Kinetic

[NSDI’15]

OpenFlow rules

abstractions



diverse abstractions

functions

graphs

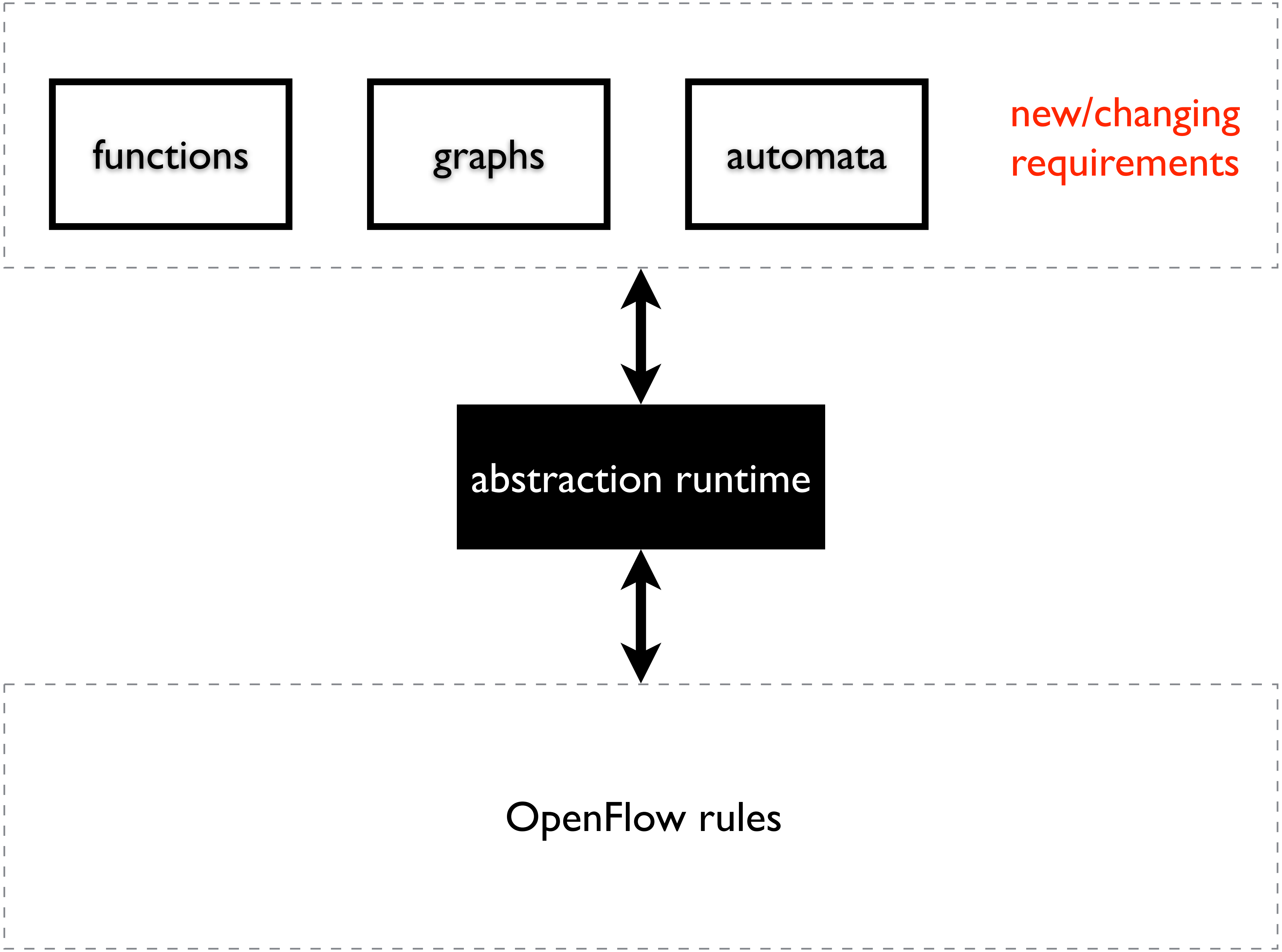
automata

…

abstraction runtime

OpenFlow rules

but network keeps evolving



functions

graphs

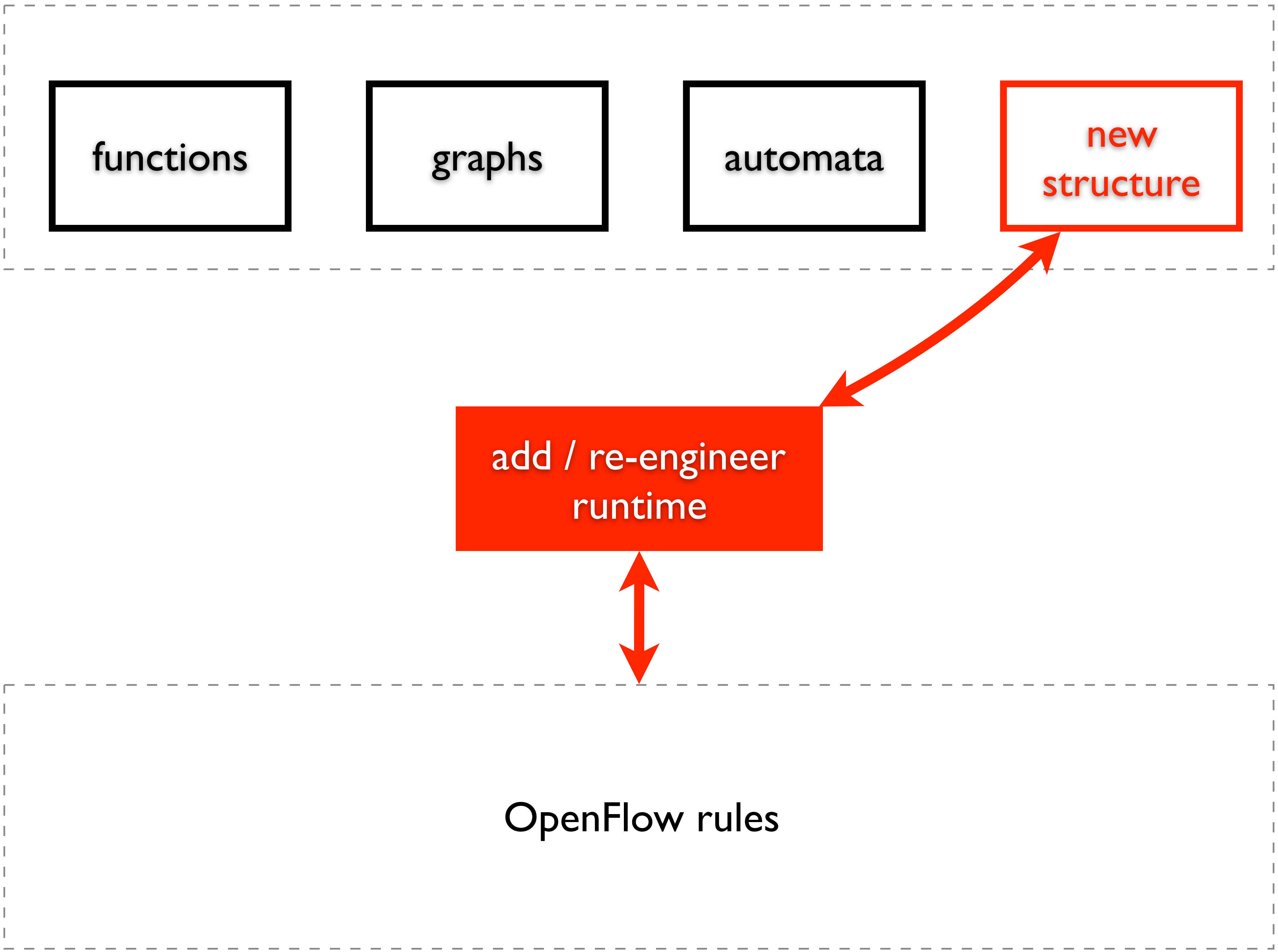
automata

new/changing requirements

abstraction runtime

OpenFlow rules

but network keeps evolving



functions

graphs

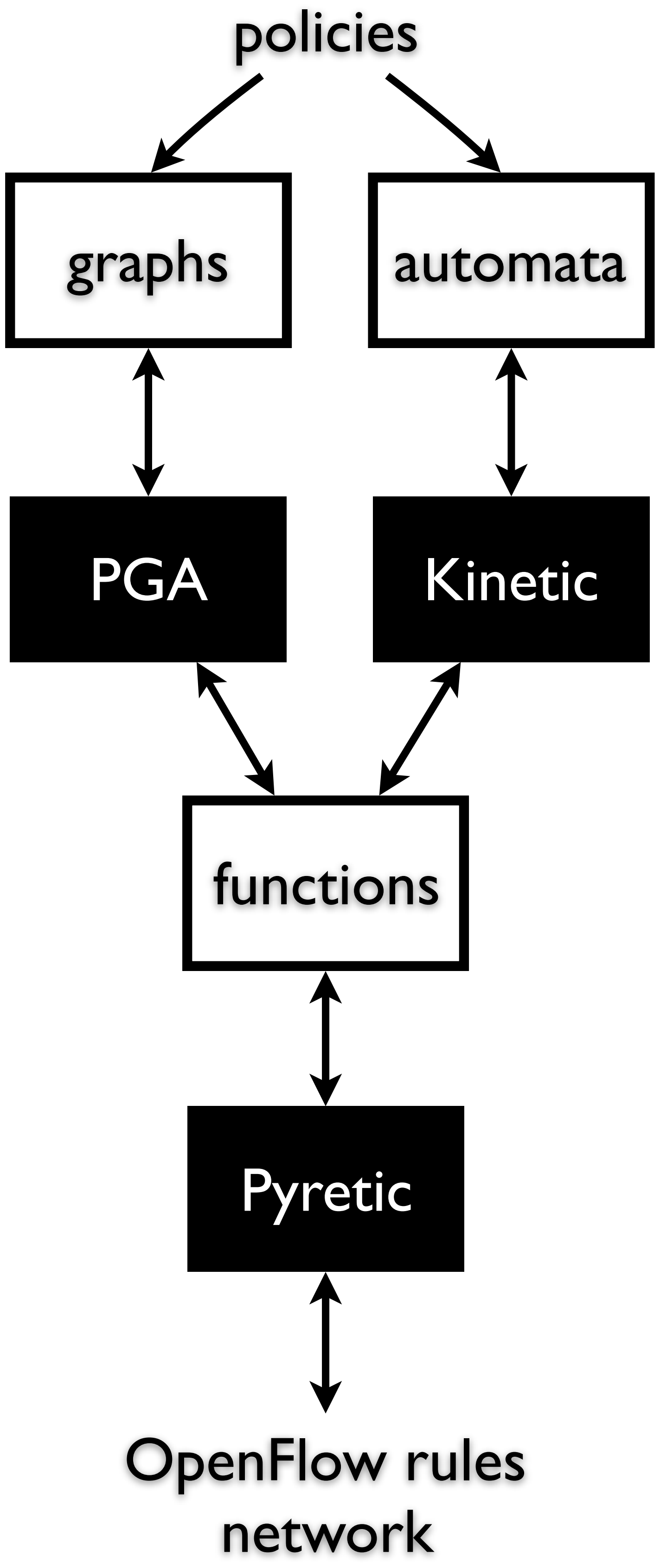
automata

new structure

add / re-engineer runtime

OpenFlow rules

and applications (components) interact



policies

graphs

automata

PGA

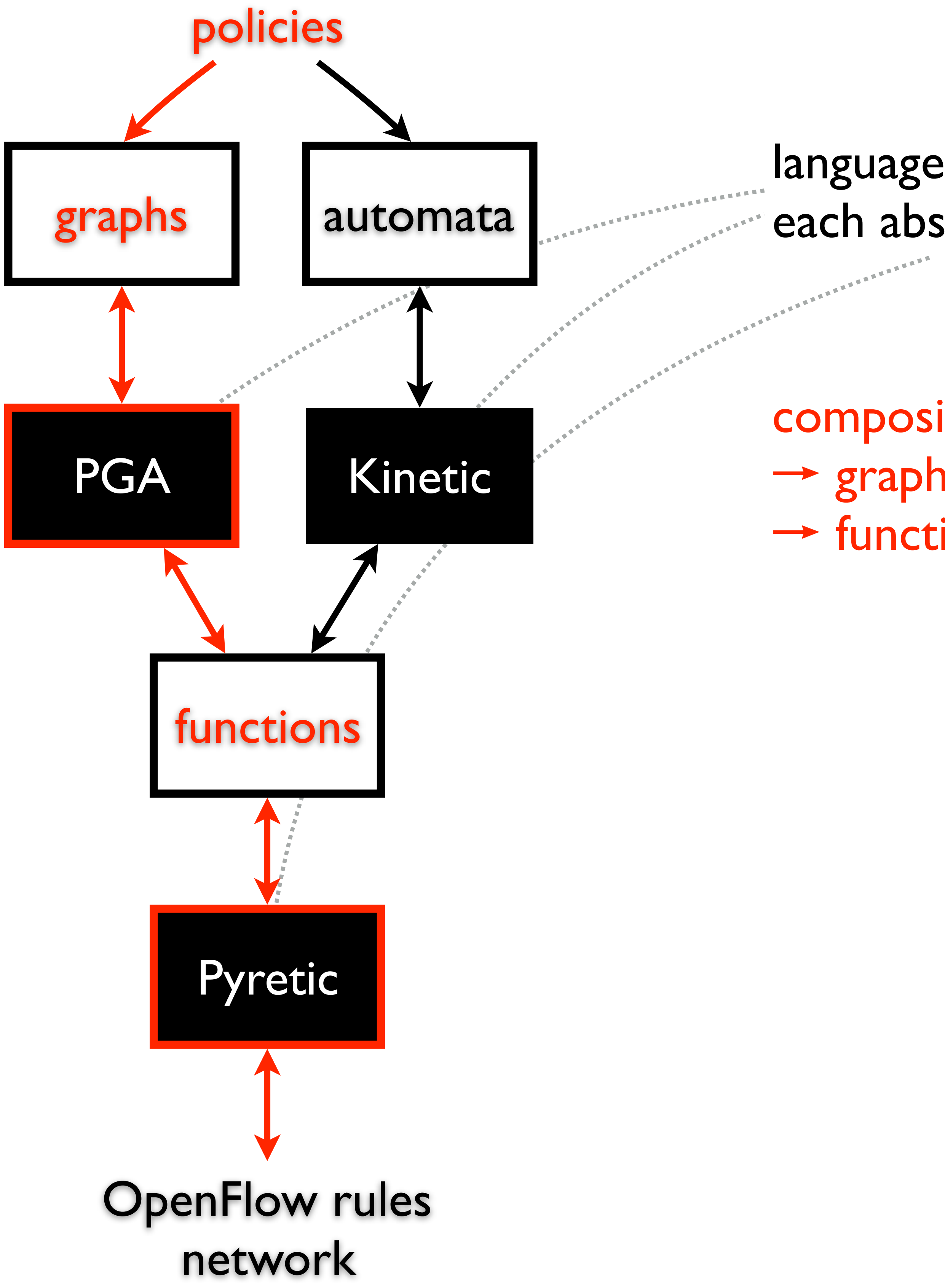
Kinetic

functions

Pyretic

OpenFlow rules
network

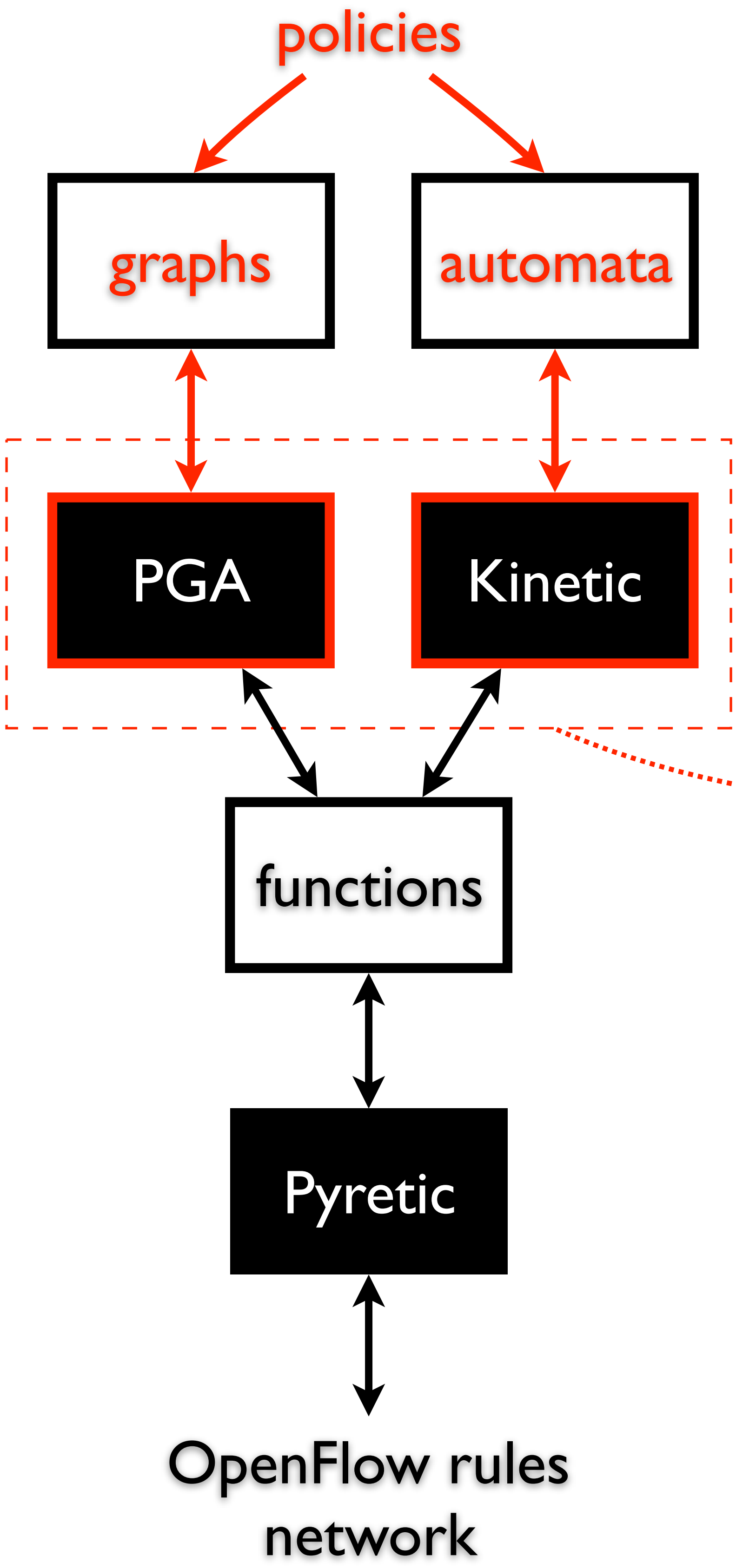
and applications (components) interact


policies
graphs
automata
PGA
Kinetic
functions
Pyretic
OpenFlow rules
network

language-level orchestration restricted to each abstraction

composing (+) policy
→ graph $+_{\text{PGA}}$ graph
→ function $+_{\text{Pyretic}}$ function

and applications (components) interact



policies

graphs

automata

PGA

Kinetic

functions

Pyretic

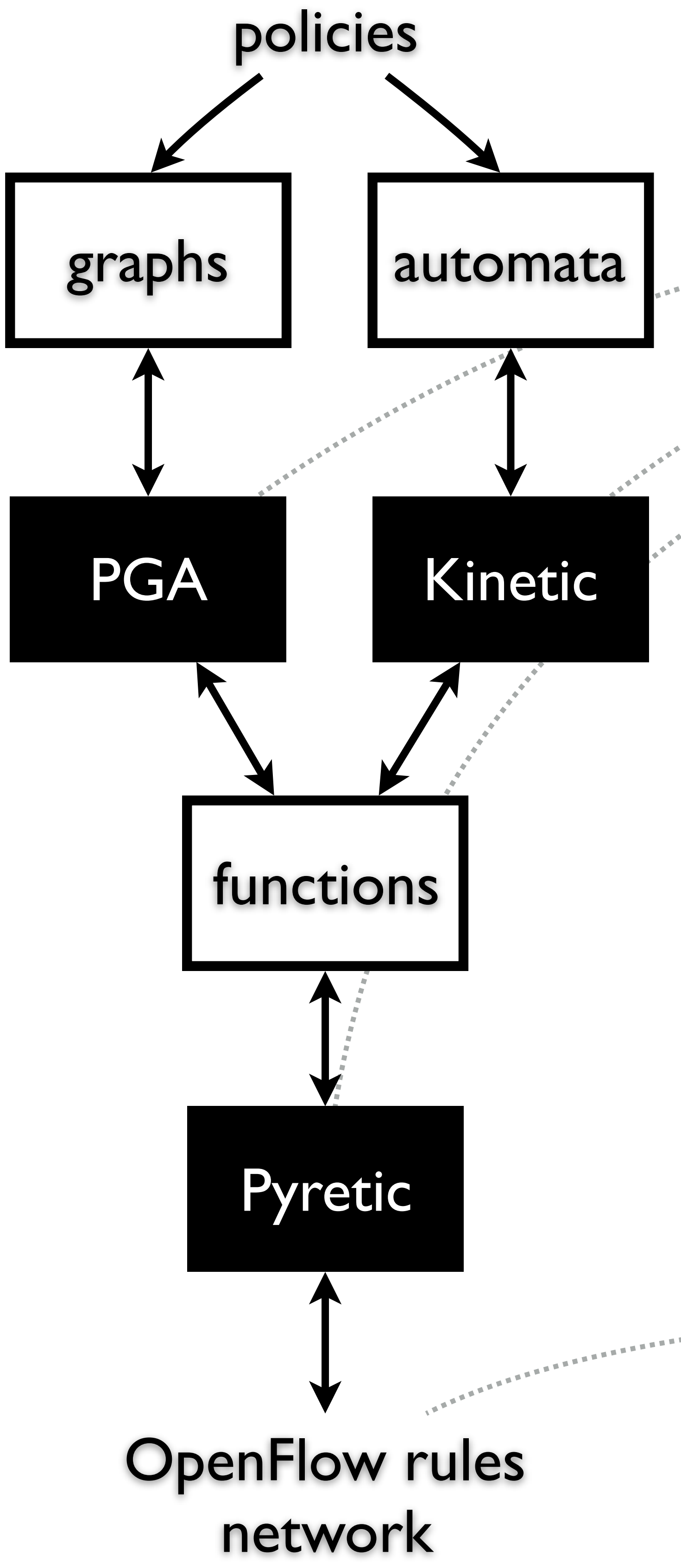
OpenFlow rules
network

language-level orchestration restricted to each abstraction

composing (+) policy
→ graph $+_?$ automata

how to integrate the runtime?
hard-wiring internals?

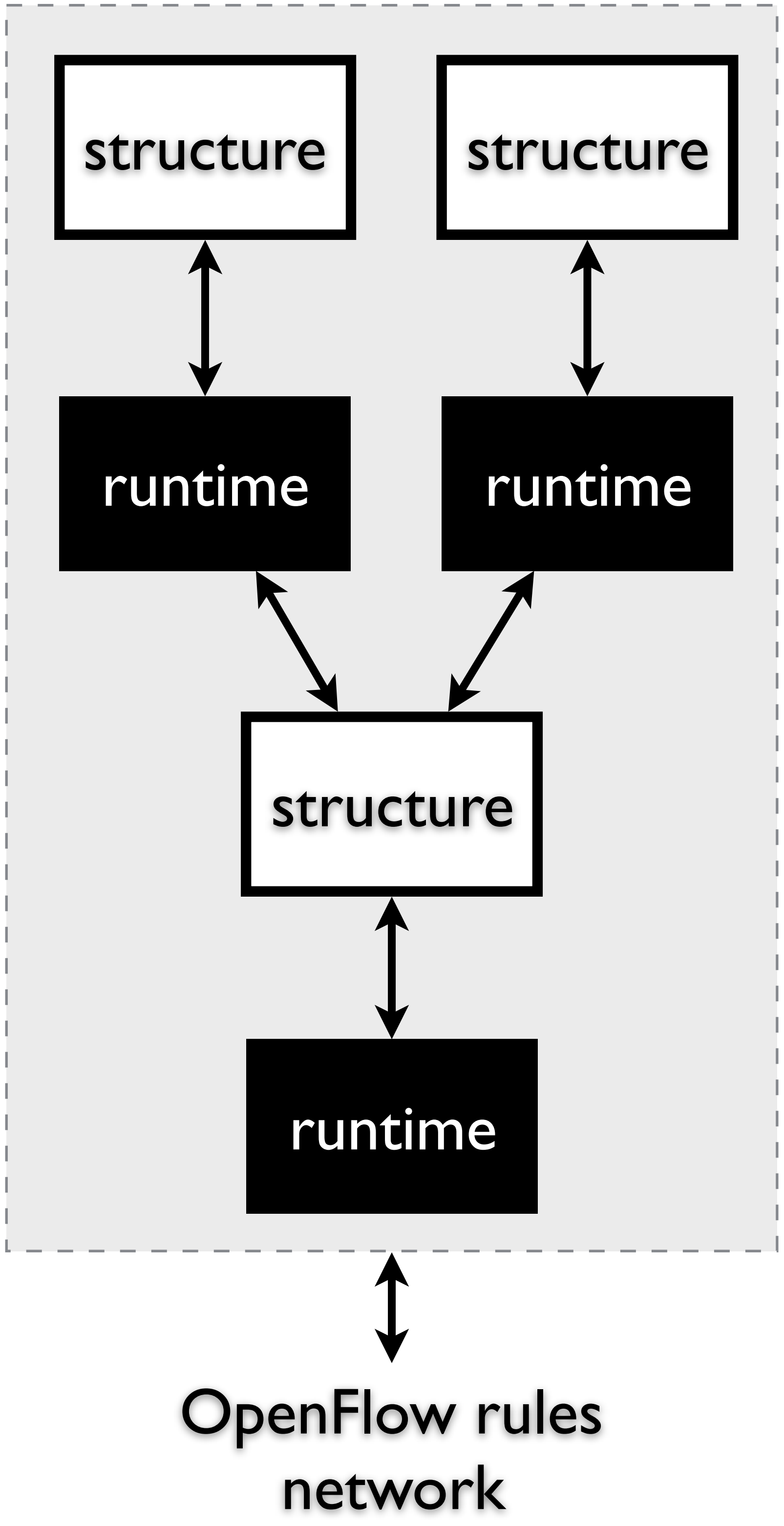
and applications (components) interact


policies
graphs
automata
PGA
Kinetic
functions
Pyretic
OpenFlow rules
network

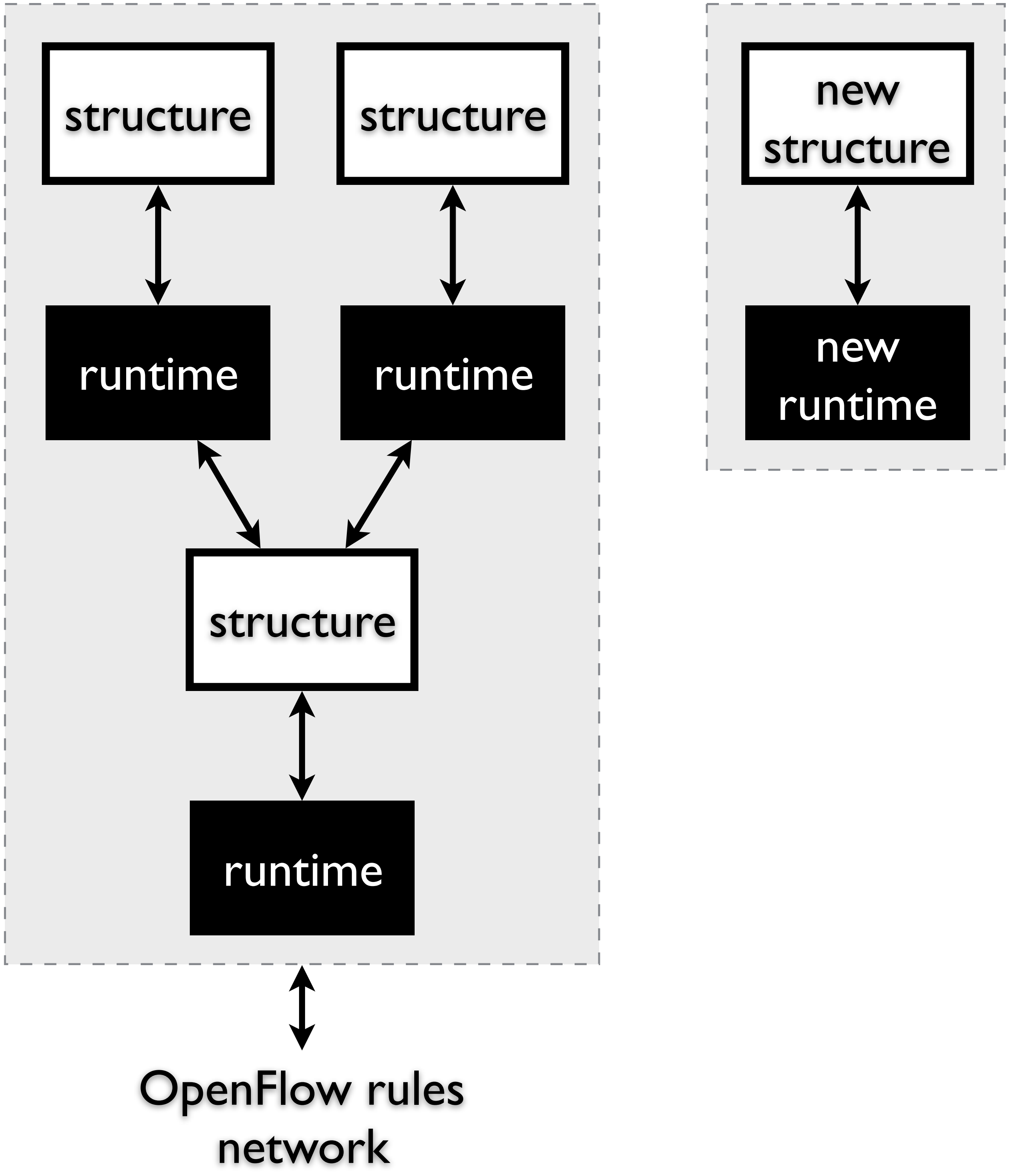
language-level orchestration restricted to each abstraction

abstraction-agonistic coordination often low-level
Co-visor [NSDI'15] statesman [SIGCOMM'14]

current state of abstraction research

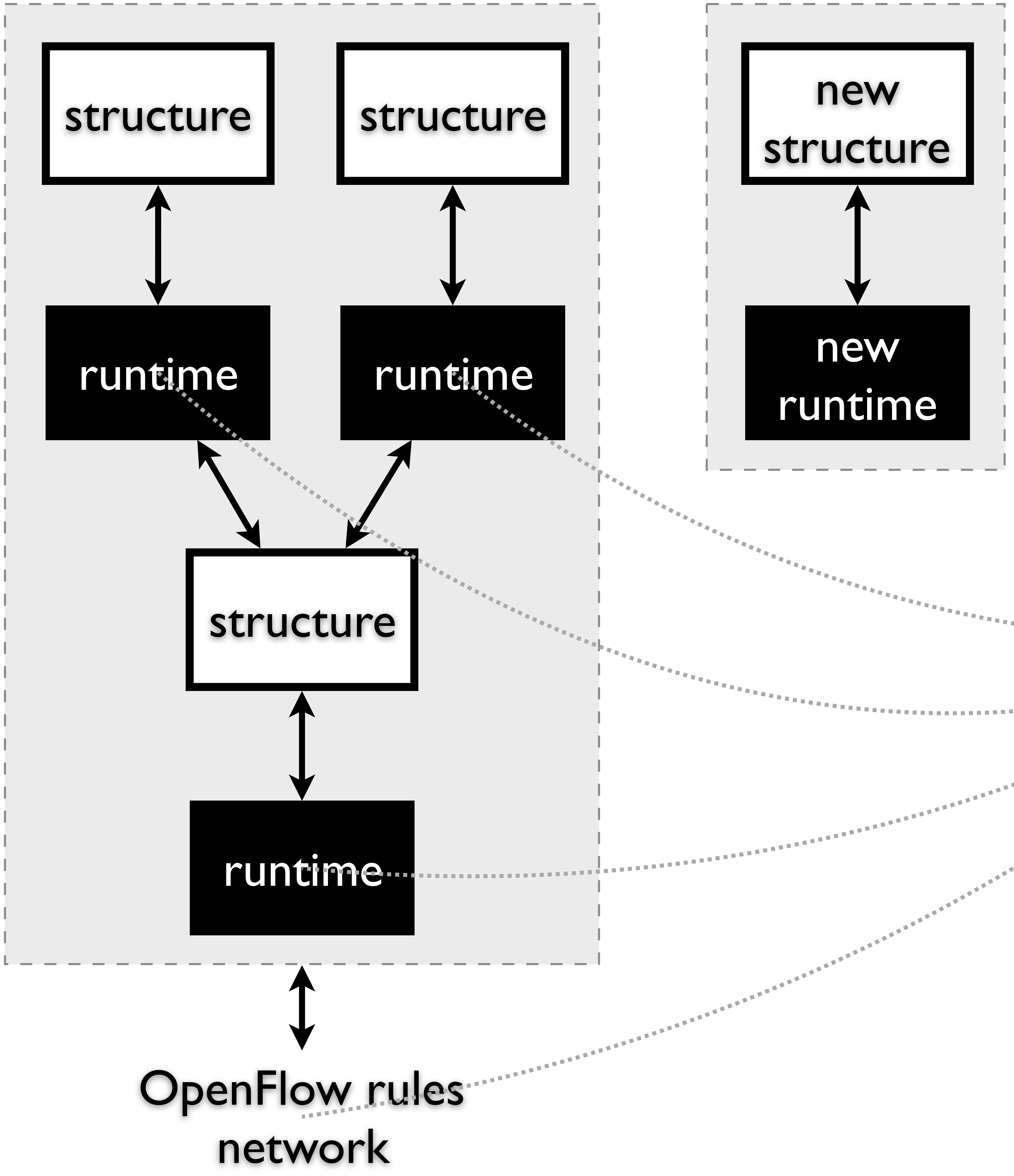

structure
structure
runtime
runtime
structure
runtime
OpenFlow rules
network

current state of abstraction research


structure
structure
runtime
runtime
structure
runtime
OpenFlow rules
network
new
structure
new
runtime

enlarging body of abstractions

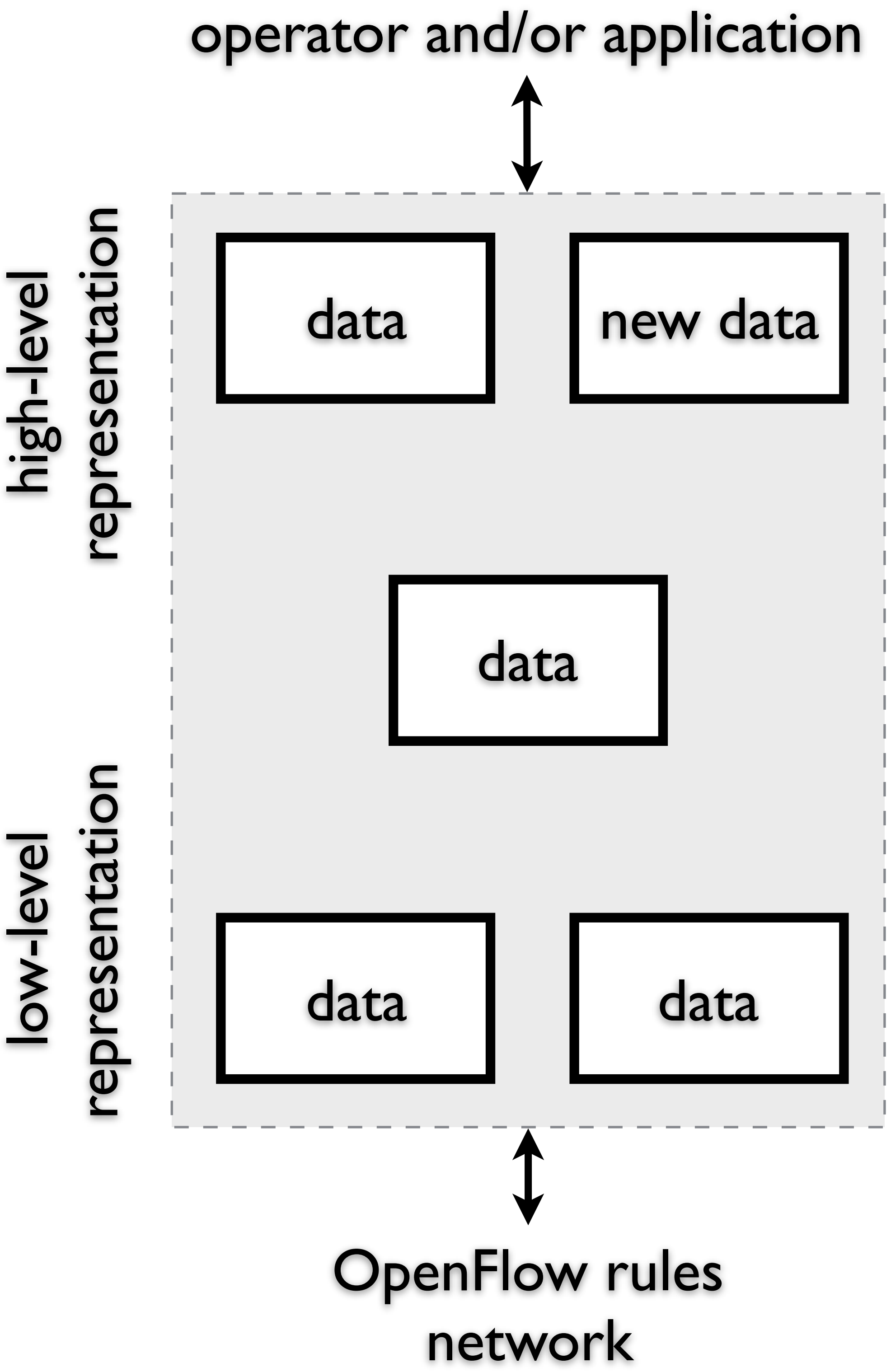
current state of abstraction research


structure
structure
new structure
runtime
runtime
new runtime
structure
runtime
OpenFlow rules
network

enlarging body of abstractions

fragmented orchestration

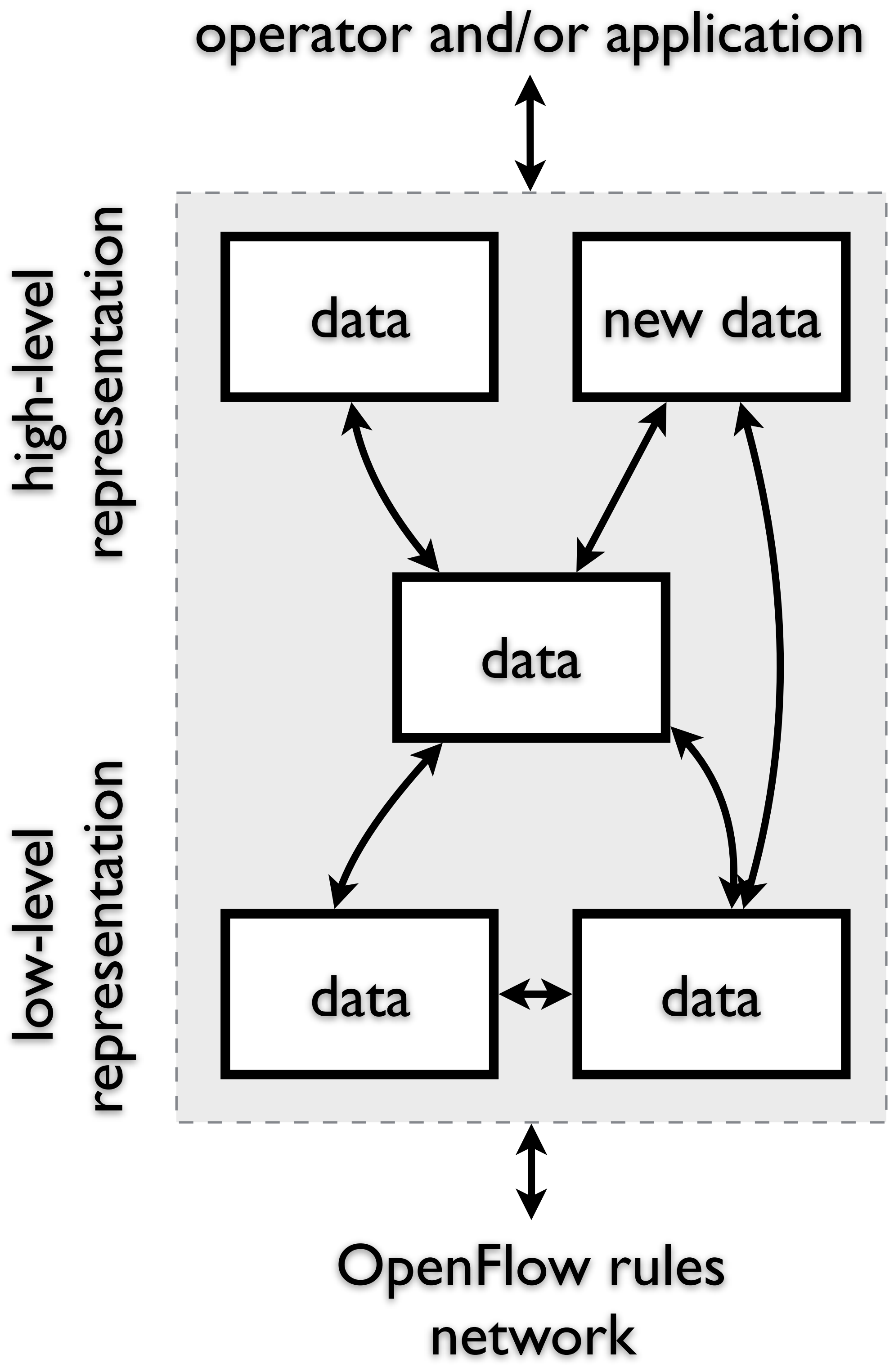
our perspective


operator and/or application
high-level representation
data
new data
data
low-level representation
data
data
OpenFlow rules
network

SDN control revolves around data representation

- discard specialized, pre-compiled, fixed structures
- adopt a *plain data representation*

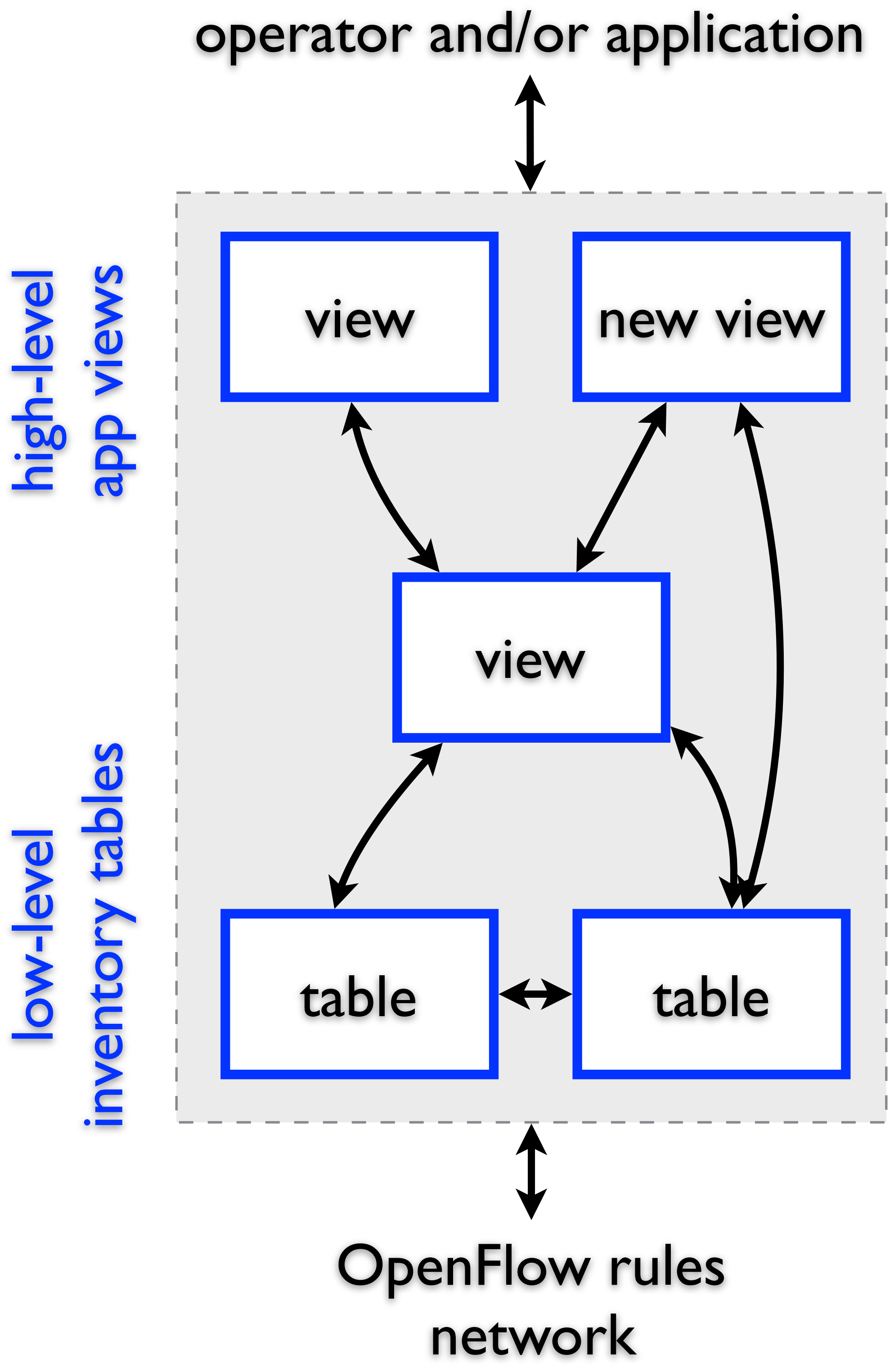
our perspective


operator and/or application
high-level representation
low-level representation
data
new data
data
data
data
OpenFlow rules
network

SDN control revolves around data representation

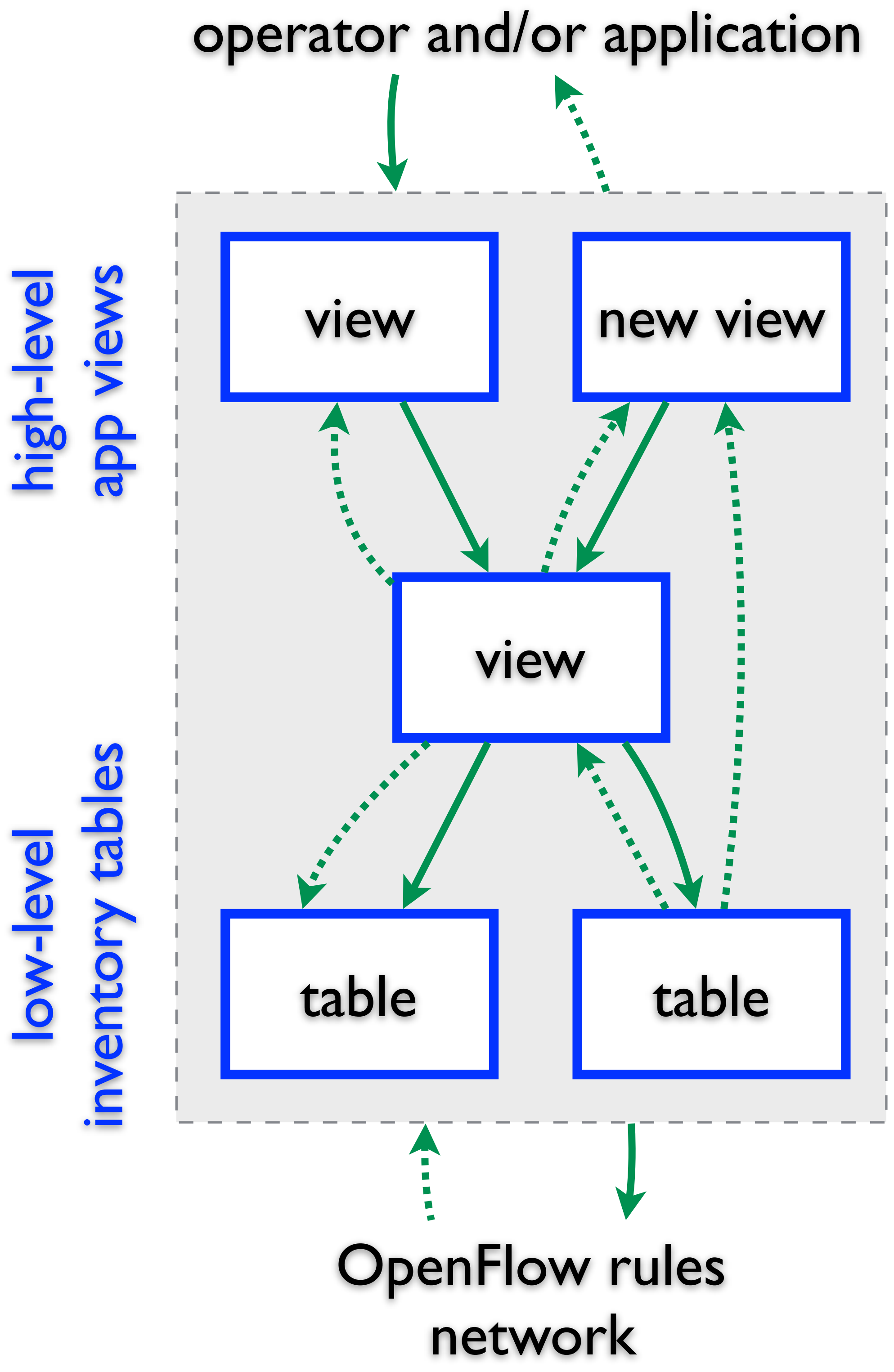
- discard specialized, pre-compiled, fixed structures
- adopt a *plain data representation*
- use a *universal data language*

a database-defined network


operator and/or application
high-level
app views
view
new view
view
low-level
inventory tables
table
table
OpenFlow rules
network

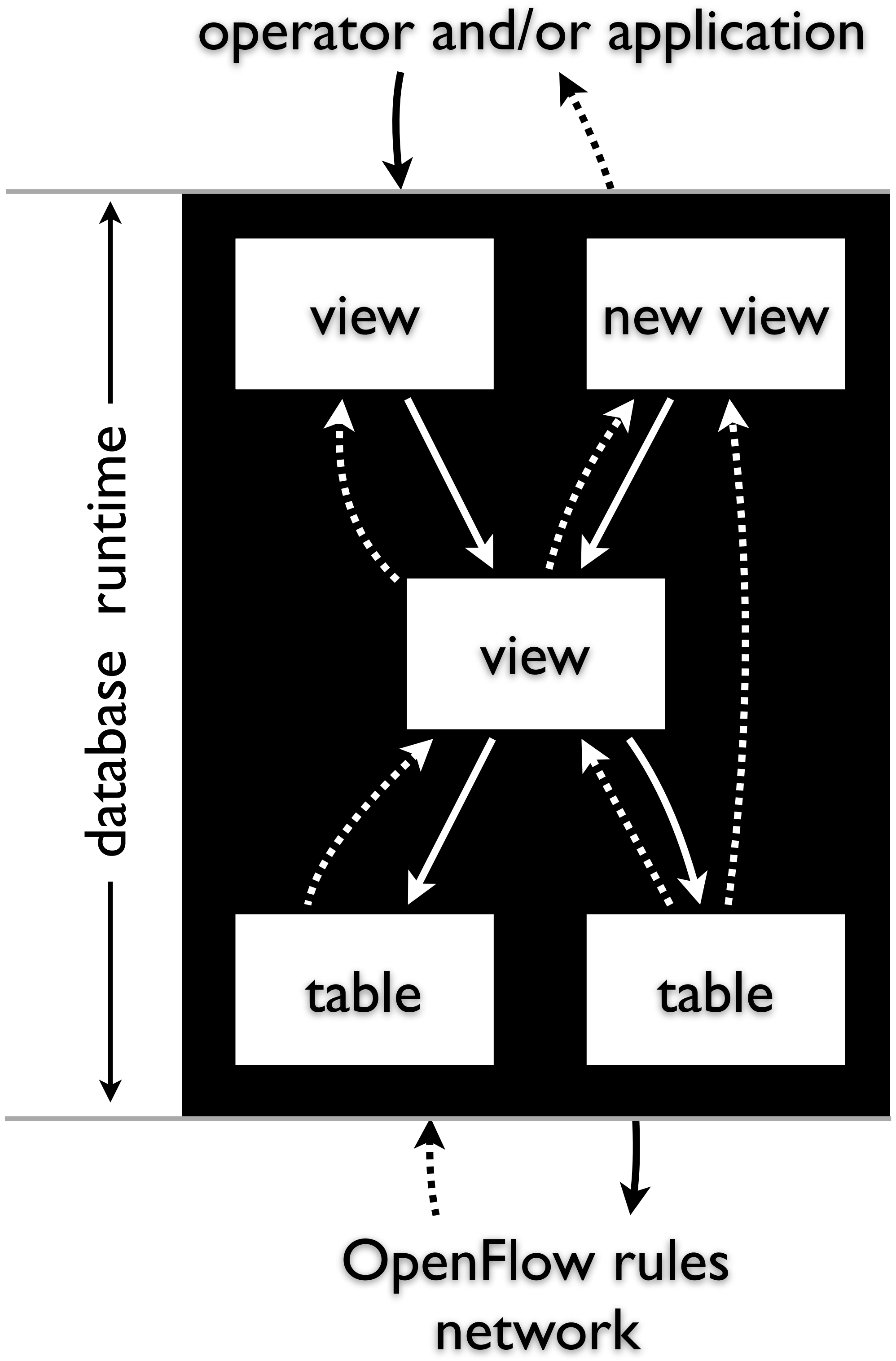
- relation — the plain data representation
  - table — stored relation
  - view — virtual relation

a database-defined network


operator and/or application
high-level
app views
view
new view
view
low-level
inventory tables
table
table
OpenFlow rules
network

- relation — the plain data representation
  - table — stored relation
  - view — virtual relation
- SQL — the universal data language
  - query, update, trigger, rule

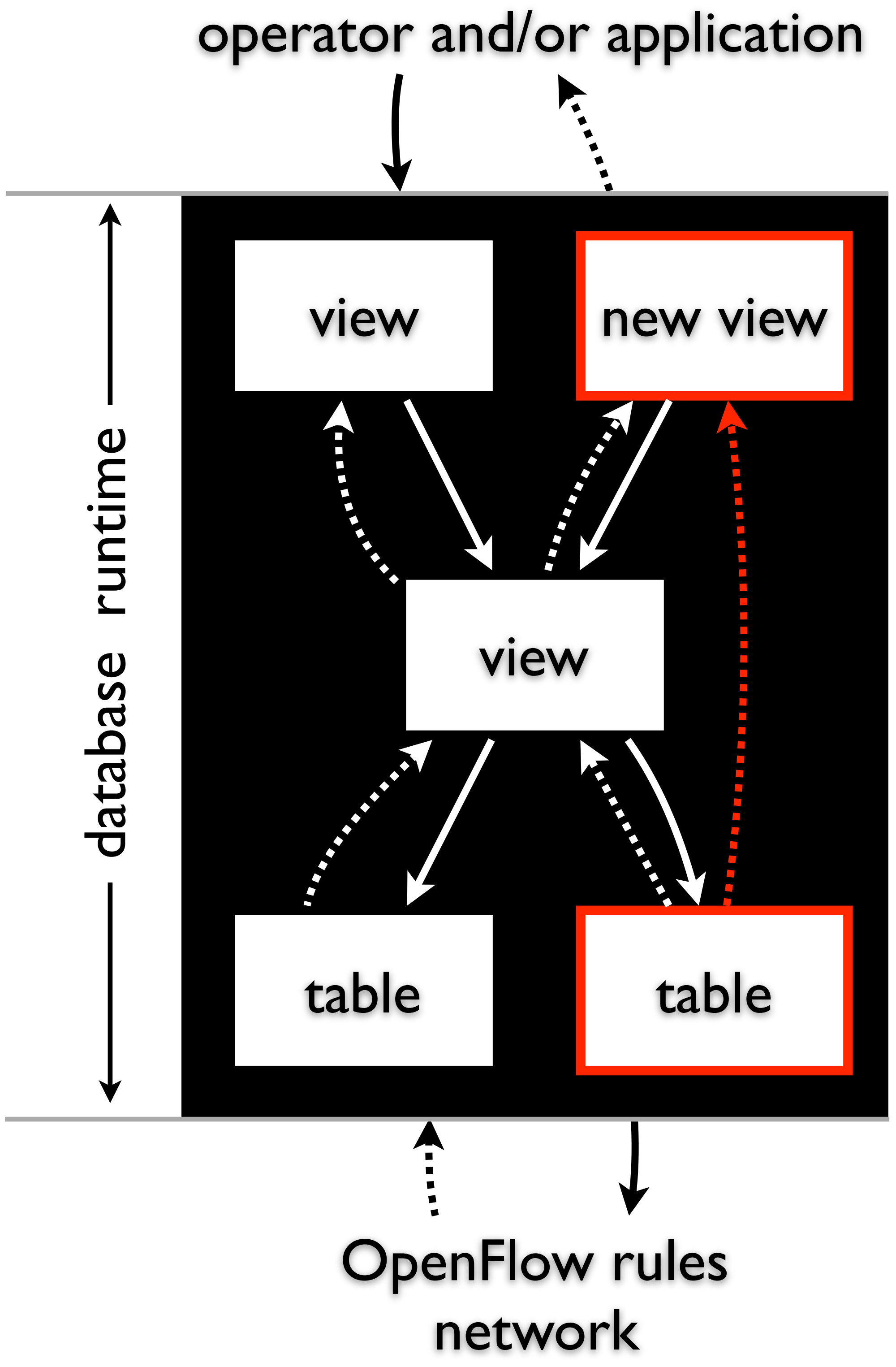
Ravel: a realization with SQL database


operator and/or application
view
new view
view
table
table
database runtime
OpenFlow rules
network

attractive features

- ad-hoc programmable abstraction via views
- orchestration across abstractions via view mechanism
- orchestration across applications via data mediation
- network control via SQL

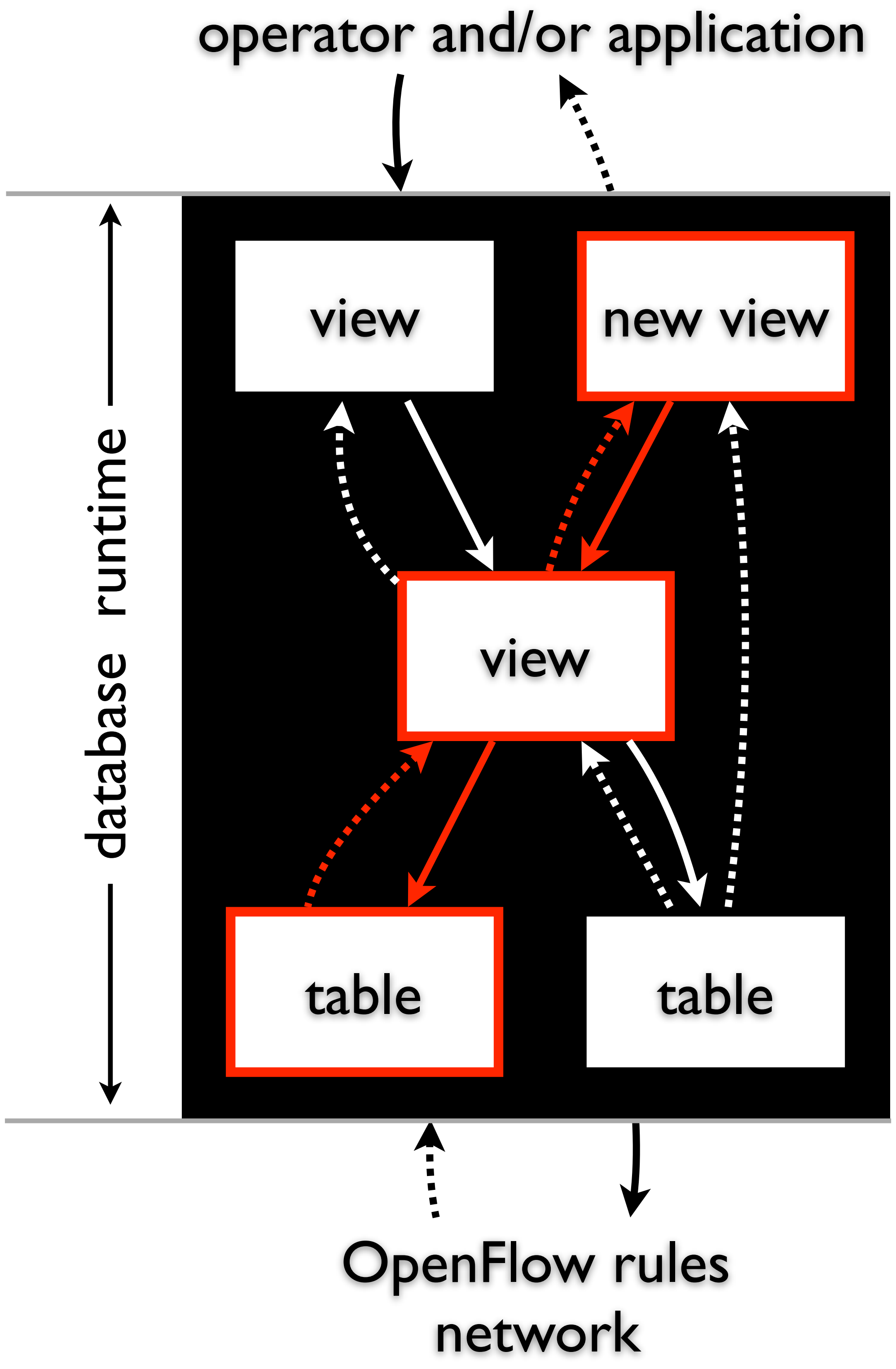
Ravel: a realization with SQL database


operator and/or application
database runtime
view
new view
view
table
table
OpenFlow rules
network

attractive features

- ad-hoc programmable abstraction via views
- orchestration across abstractions via view mechanism
- orchestration across applications via data mediation
- network control via SQL

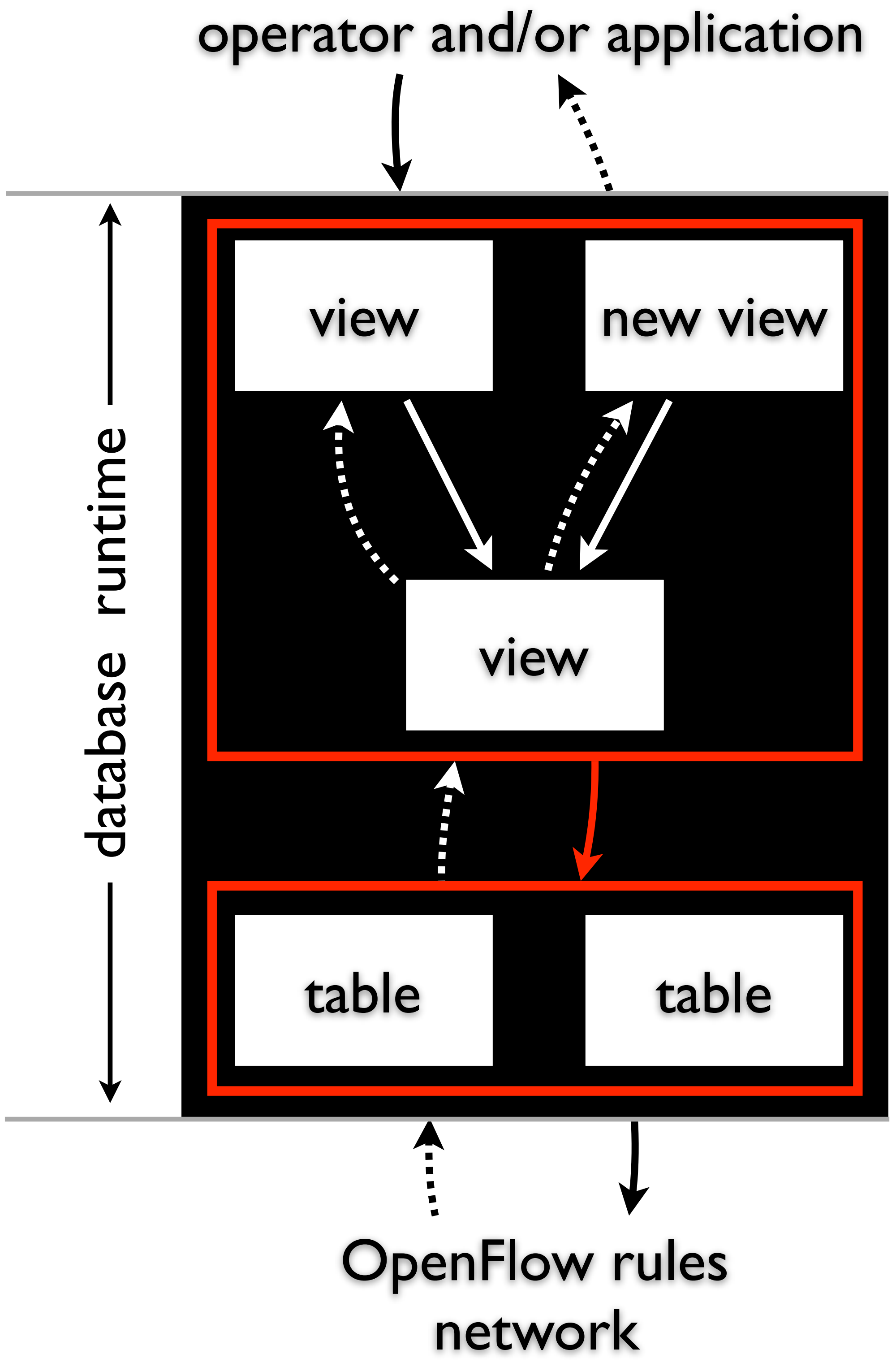
Ravel: a realization with SQL database


operator and/or application
view
new view
view
table
table
database runtime
OpenFlow rules
network

attractive features

- ad-hoc programmable abstraction via views
- orchestration across abstractions via view mechanism
- orchestration across applications via data mediation
- network control via SQL

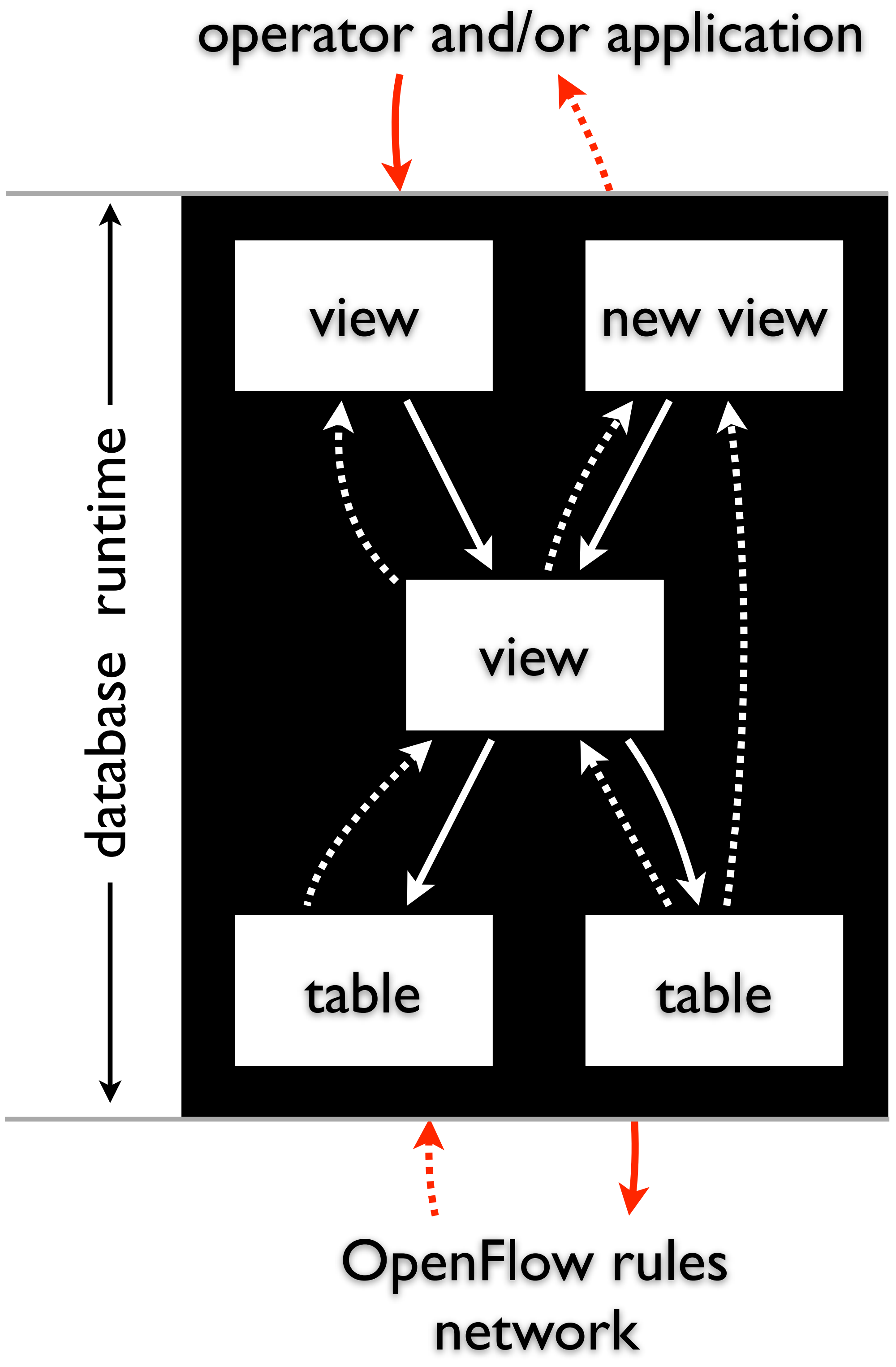
Ravel: a realization with SQL database


operator and/or application
view
new view
view
table
table
database runtime
OpenFlow rules
network

attractive features

- ad-hoc programmable abstraction via views
- orchestration across abstractions via view mechanism
- orchestration across applications via data mediation
- network control via SQL

Ravel: a realization with SQL database


operator and/or application
view
new view
view
table
table
database runtime
OpenFlow rules
network

attractive features

- ad-hoc programmable abstraction via views
- orchestration across abstractions via view mechanism
- orchestration across applications via data mediation
- network control via SQL

abstraction: network tables

reachability matrix

fid	src	dst	vol	…
1	h_1	h_4	5	
2	h_2	h_3	9	

…

topology

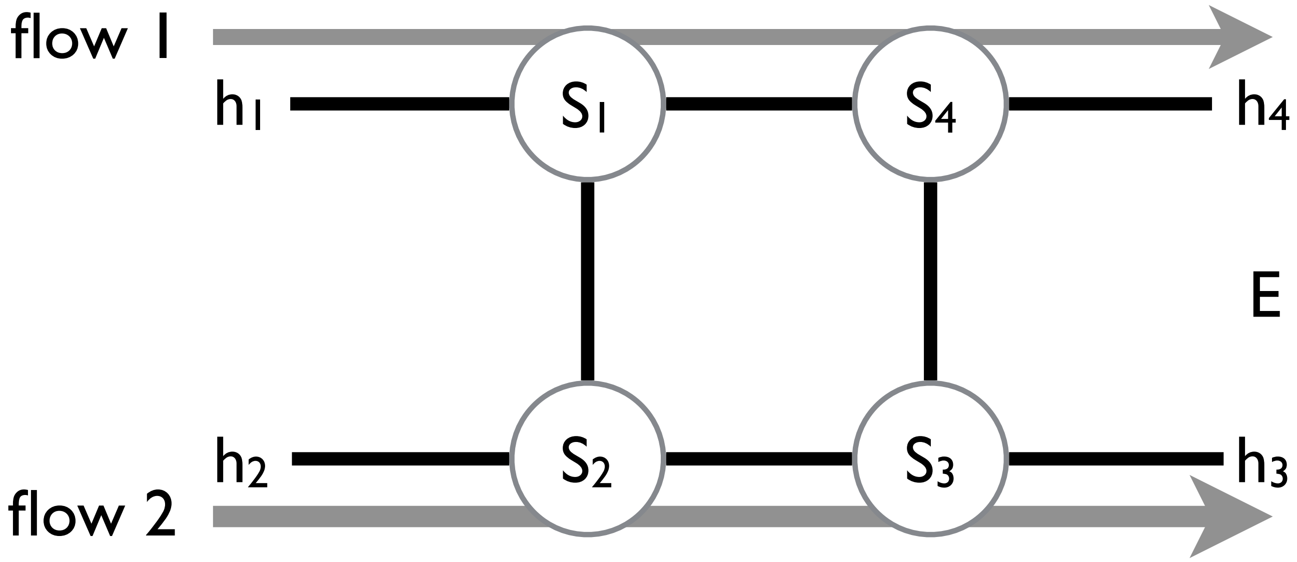
sid	nid
S_1	S_2
S_1	S_3
S_1	h_1

…

configuration

fid	sid	nid
1	S_1	S_4
1	S_4	h_4

…


flow 1
h_1
S_1
S_4
h_4
E
h_2
S_2
S_3
h_3
flow 2

abstraction: application view

firewall view: monitoring unsafe flows violating acl policy

```
CREATE VIEW acl_violation AS (
   SELECT fid
   FROM rm
   WHERE FW = 1  AND
     (src, dst) NOT IN
     (SELECT end1, end2 FROM acl
                        WHERE allow = 1)
);
```

```
CREATE TABLE acl (
  end1 integer, end2 integer, allow integer
);
```

abstraction: application view

firewall view: monitoring unsafe flows violating acl policy

```
CREATE VIEW acl_violation AS (
   SELECT fid
   FROM rm
   WHERE FW = 1  AND
     (src, dst) NOT IN
     (SELECT end1, end2 FROM acl
                        WHERE allow = 1)
);
```

```
CREATE TABLE acl (
  end1 integer, end2 integer, allow integer
);
```

firewall control: repairing violation

```
CREATE RULE acl_repair AS
   ON DELETE TO acl_violation
   DO INSTEAD
         DELETE FROM rm WHERE fid = OLD.fid;
```

abstraction: application view

firewall view: monitoring unsafe flows violating acl policy

```
CREATE VIEW acl_violation AS (
   SELECT fid
   FROM rm
   WHERE FW = 1  AND
     (src, dst) NOT IN
     (SELECT end1, end2 FROM acl
                       WHERE allow = 1)
);
```

```
CREATE TABLE acl (
  end1 integer, end2 integer, allow integer
);
```

firewall control: repairing violation

```
CREATE RULE acl_repair AS
   ON DELETE TO acl_violation
   DO INSTEAD
         DELETE FROM rm WHERE fid = OLD.fid;
```

- many more
  - routing, stateful firewall, service chain policy between subdomains …

abstraction: application view

firewall view: monitoring unsafe flows violating acl policy

```
CREATE VIEW acl_violation AS (
   SELECT fid
   FROM rm
   WHERE FW = 1  AND
     (src, dst) NOT IN
     (SELECT end1, end2 FROM acl
                        WHERE allow = 1)
);
```

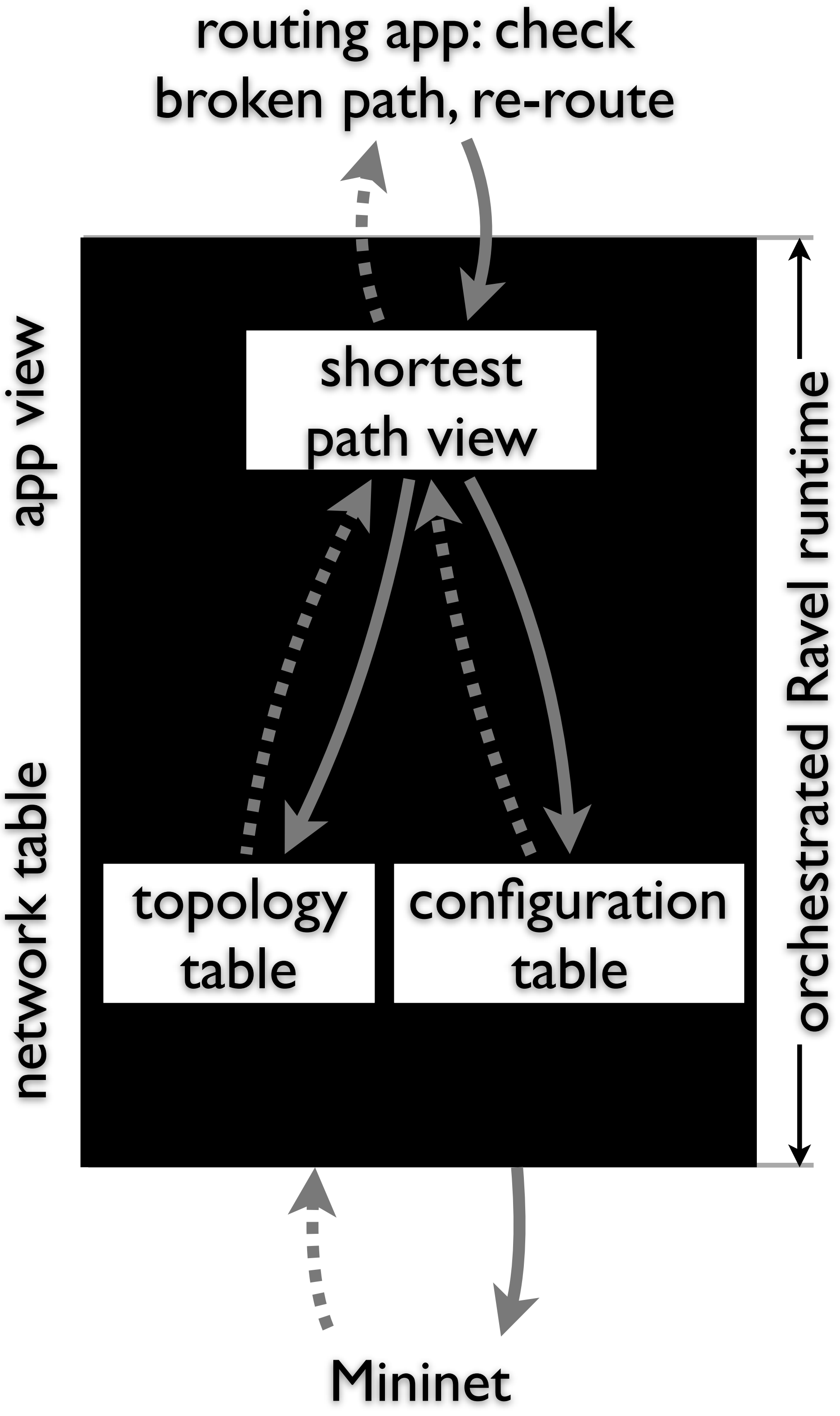
```
CREATE TABLE acl (
  end1 integer, end2 integer, allow integer
);
```

firewall control: repairing violation

```
CREATE RULE acl_repair AS
   ON DELETE TO acl_violation
   DO INSTEAD
         DELETE FROM rm WHERE fid = OLD.fid;
```

- many more
  - routing, stateful firewall, service chain policy between subdomains …
- optimizing application by materializing views
  - (one order of magnitude) faster access with small maintenance overhead (.01~10ms)

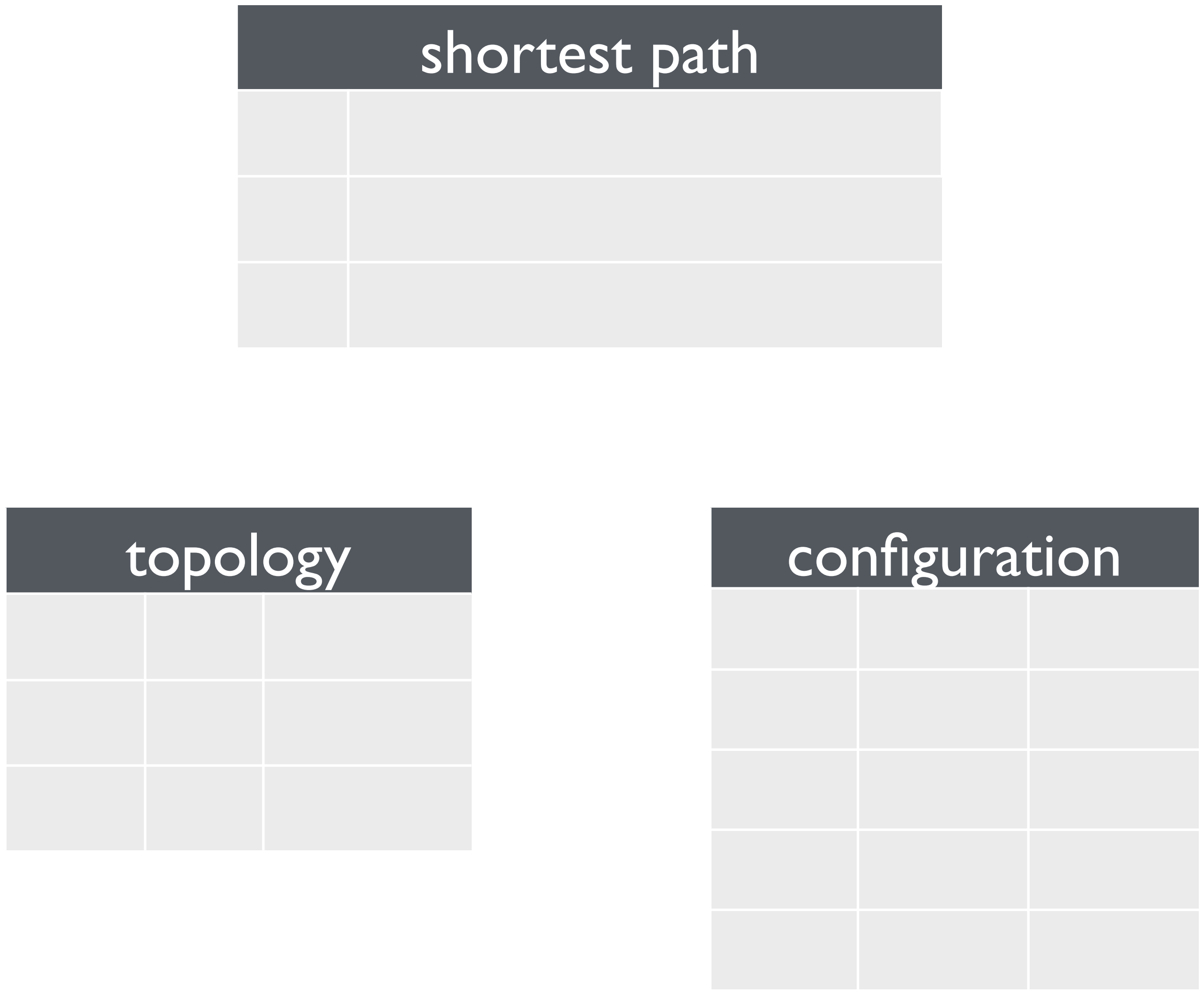
orchestration across representations



routing app: check broken path, re-route

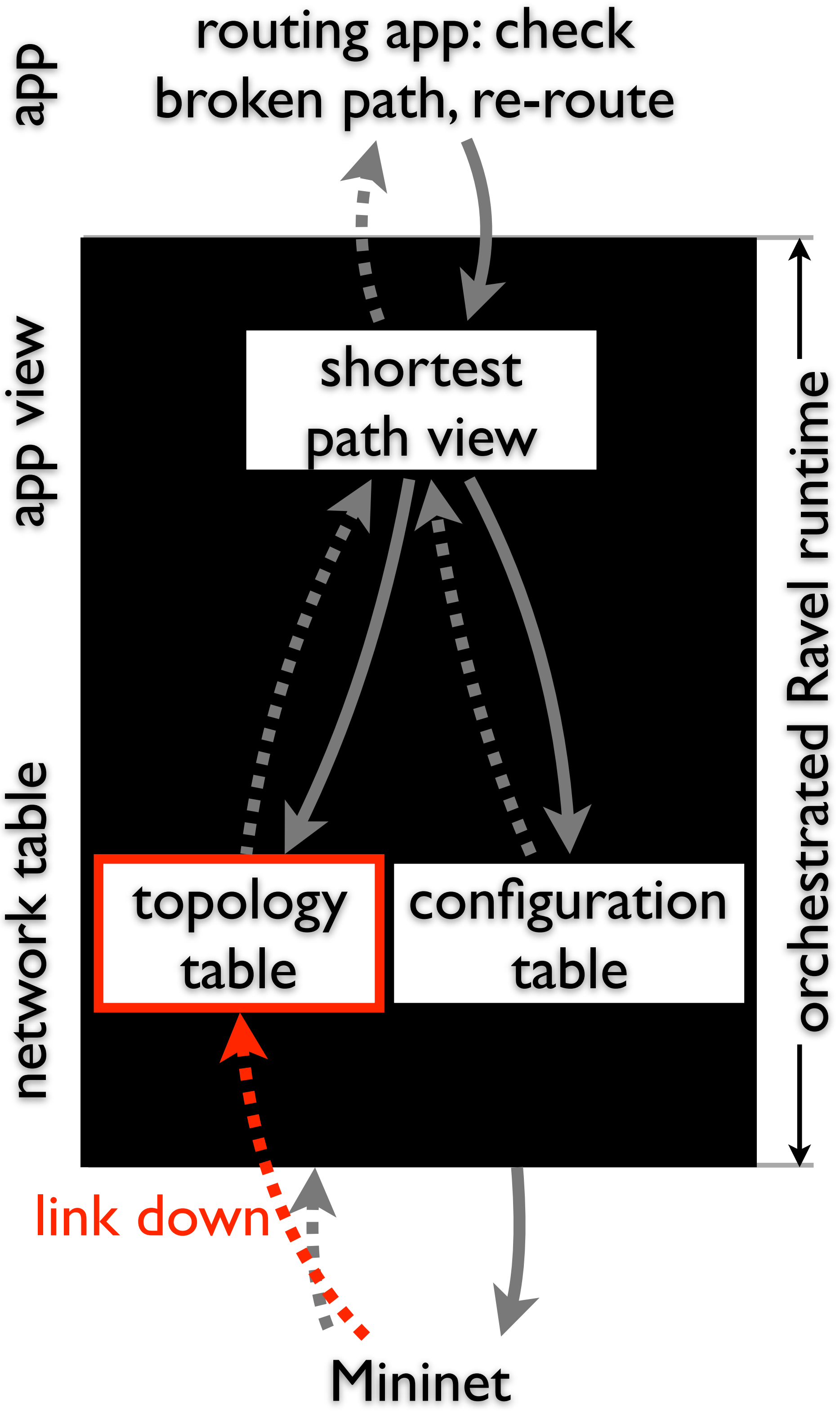
app view
network table
shortest path view
topology table
configuration table
orchestrated Ravel runtime
Mininet

SQL rule:
upon broken path, re-route



shortest path
topology
configuration

orchestration across representations


app
routing app: check broken path, re-route
app view
shortest path view
network table
topology table
configuration table
orchestrated Ravel runtime
link down
Mininet

SQL rule:
upon broken path, re-route

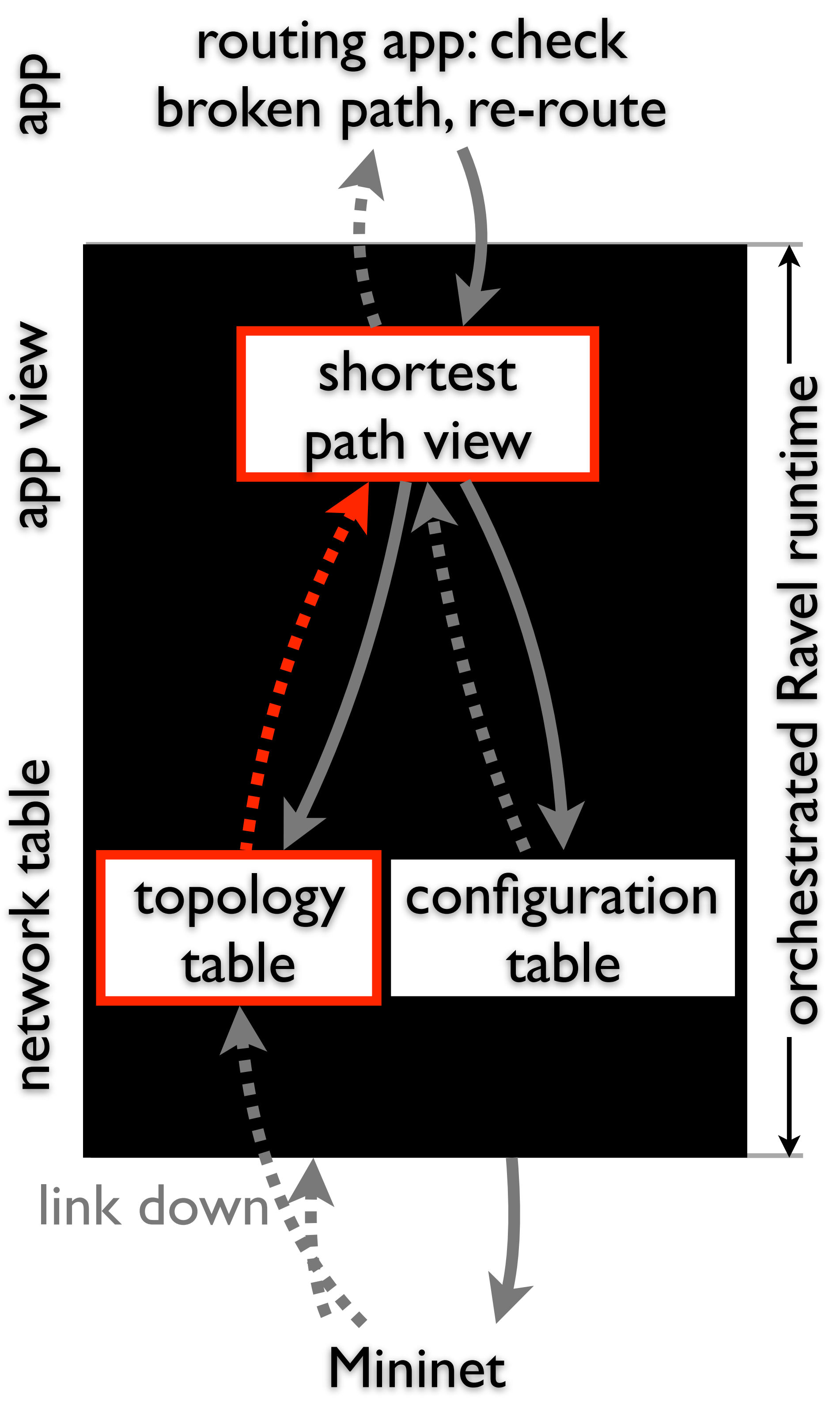
shortest path	

	topology		
	sid	nid	active
-	172	39	1
+	172	39	0

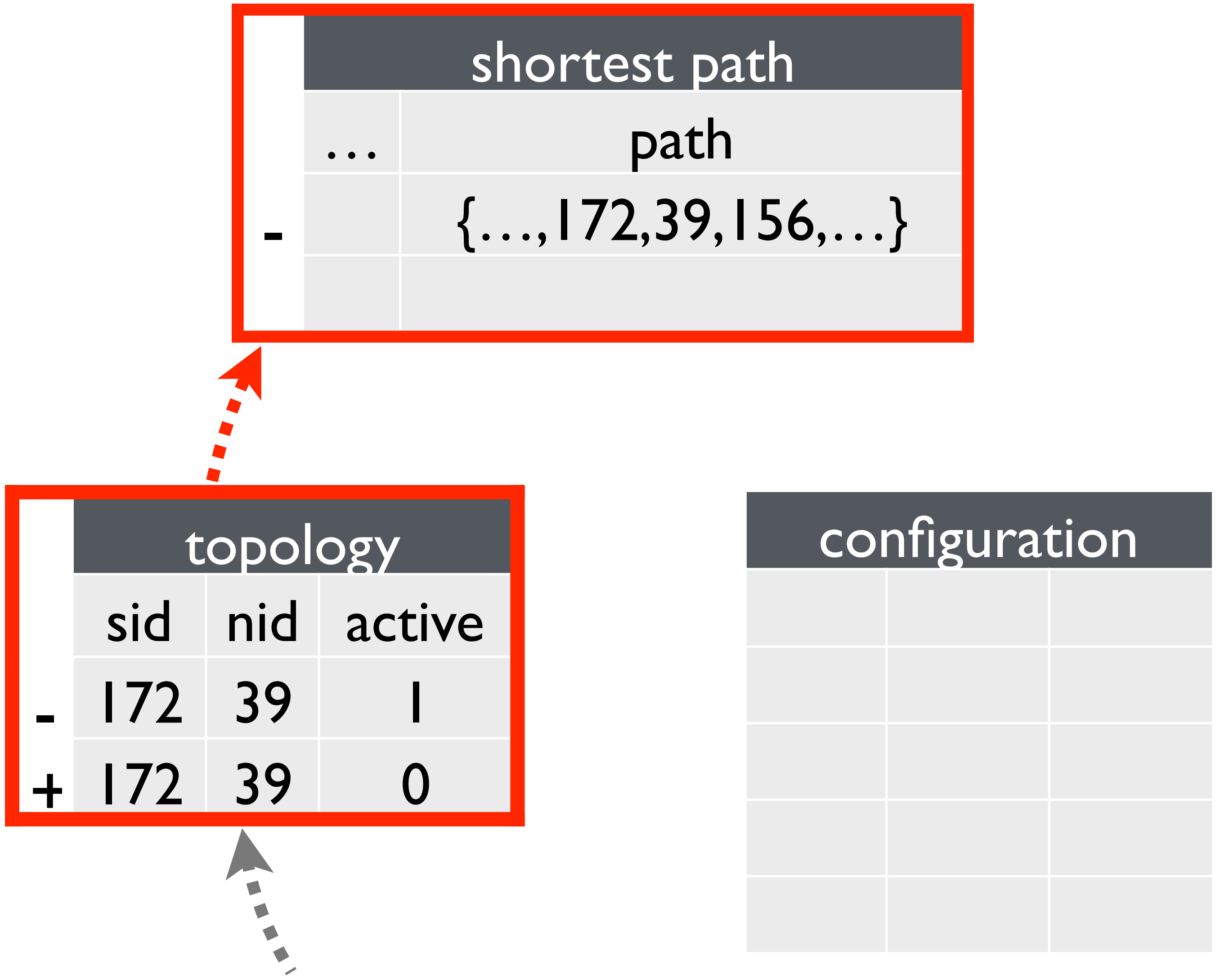
configuration		

Mininet link (172,39) down

orchestration across representations


app
routing app: check broken path, re-route
app view
shortest path view
network table
topology table
configuration table
orchestrated Ravel runtime
link down
Mininet

SQL rule:
upon broken path, re-route



shortest path

	...	path
-		{...,172,39,156,...}

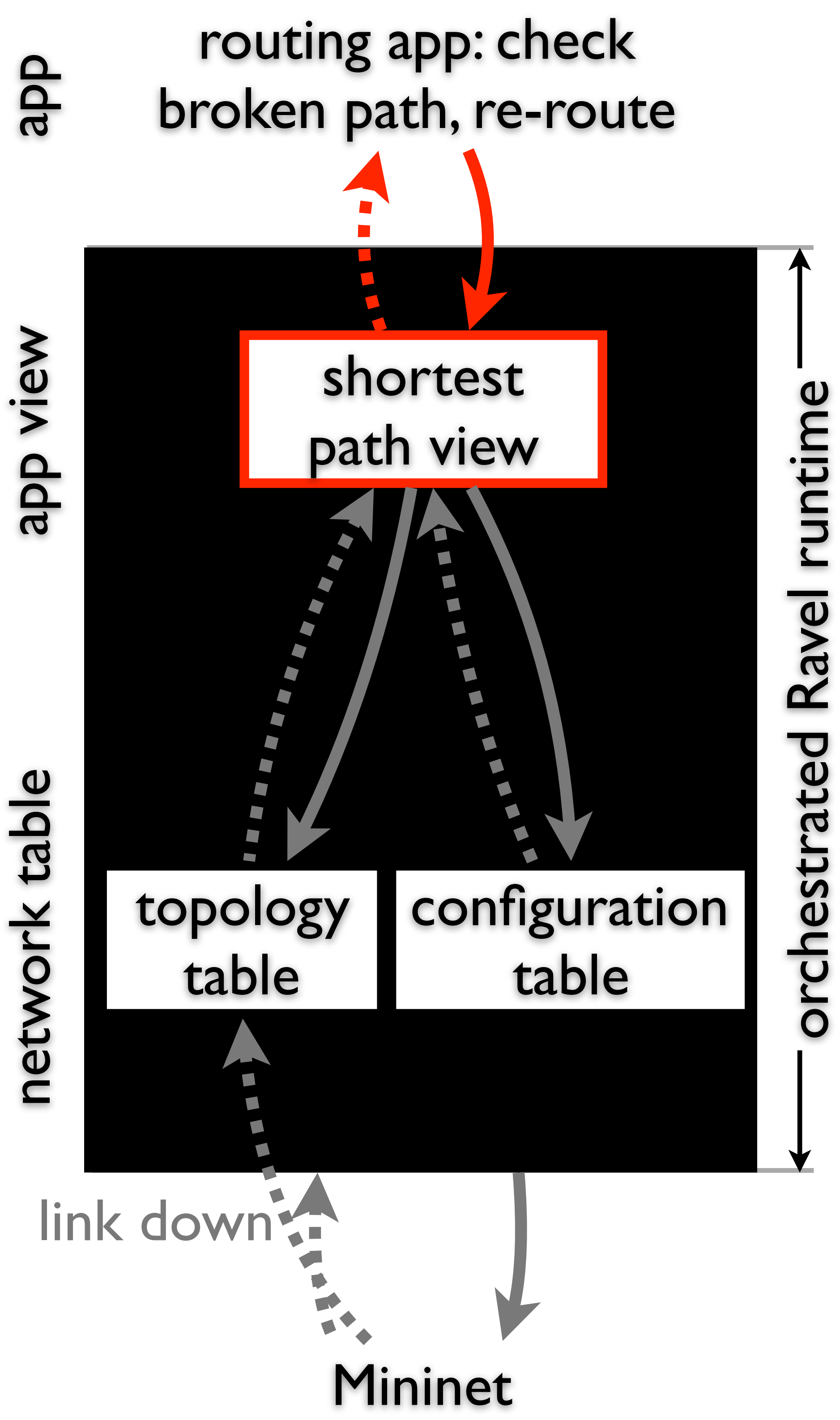
topology

	sid	nid	active
-	172	39	1
+	172	39	0

configuration

Mininet link (172,39) down

orchestration across representations



app

routing app: check broken path, re-route

app view

shortest path view

network table

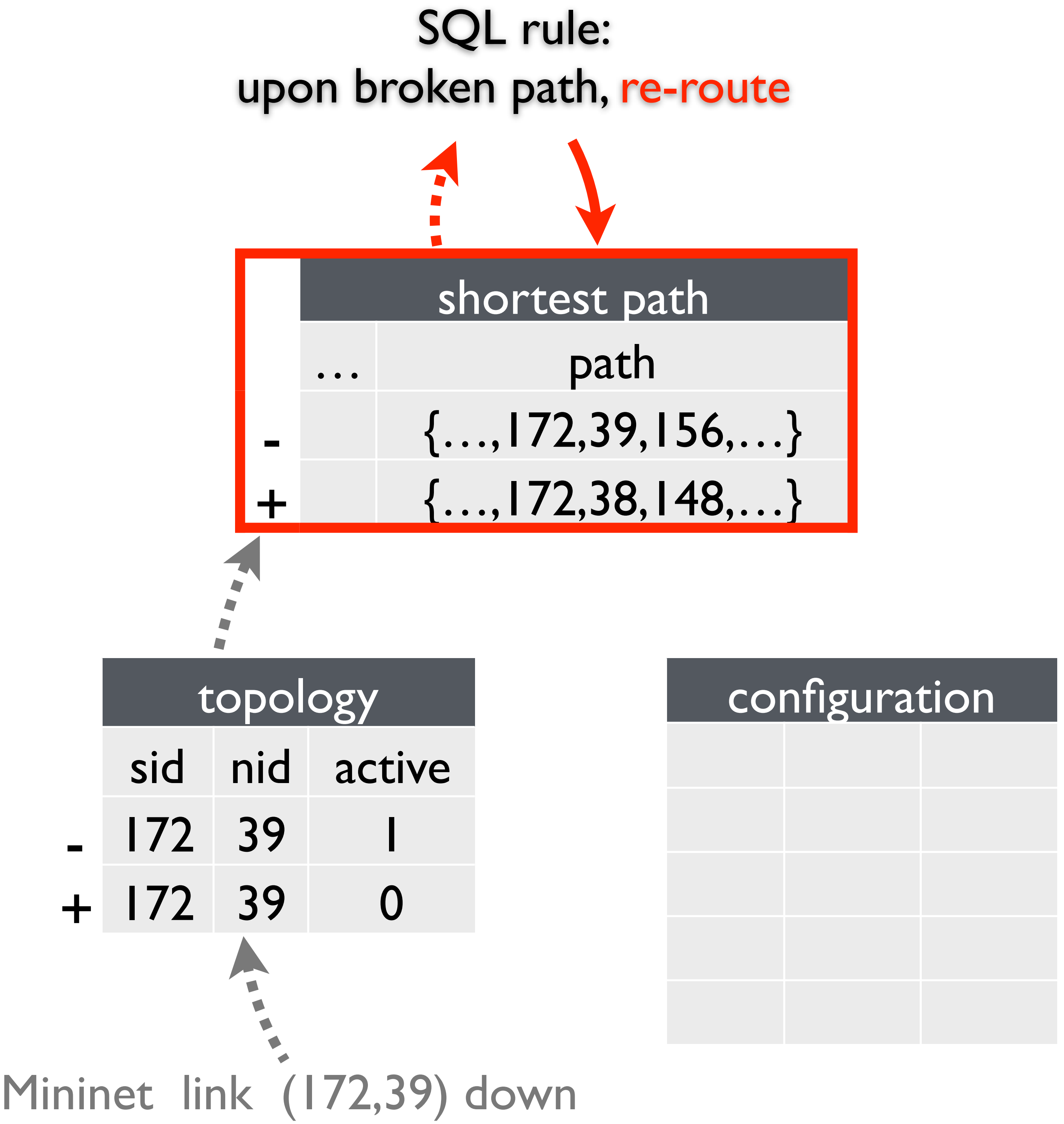
topology table

configuration table

orchestrated Ravel runtime

link down

Mininet



SQL rule:
upon broken path, re-route

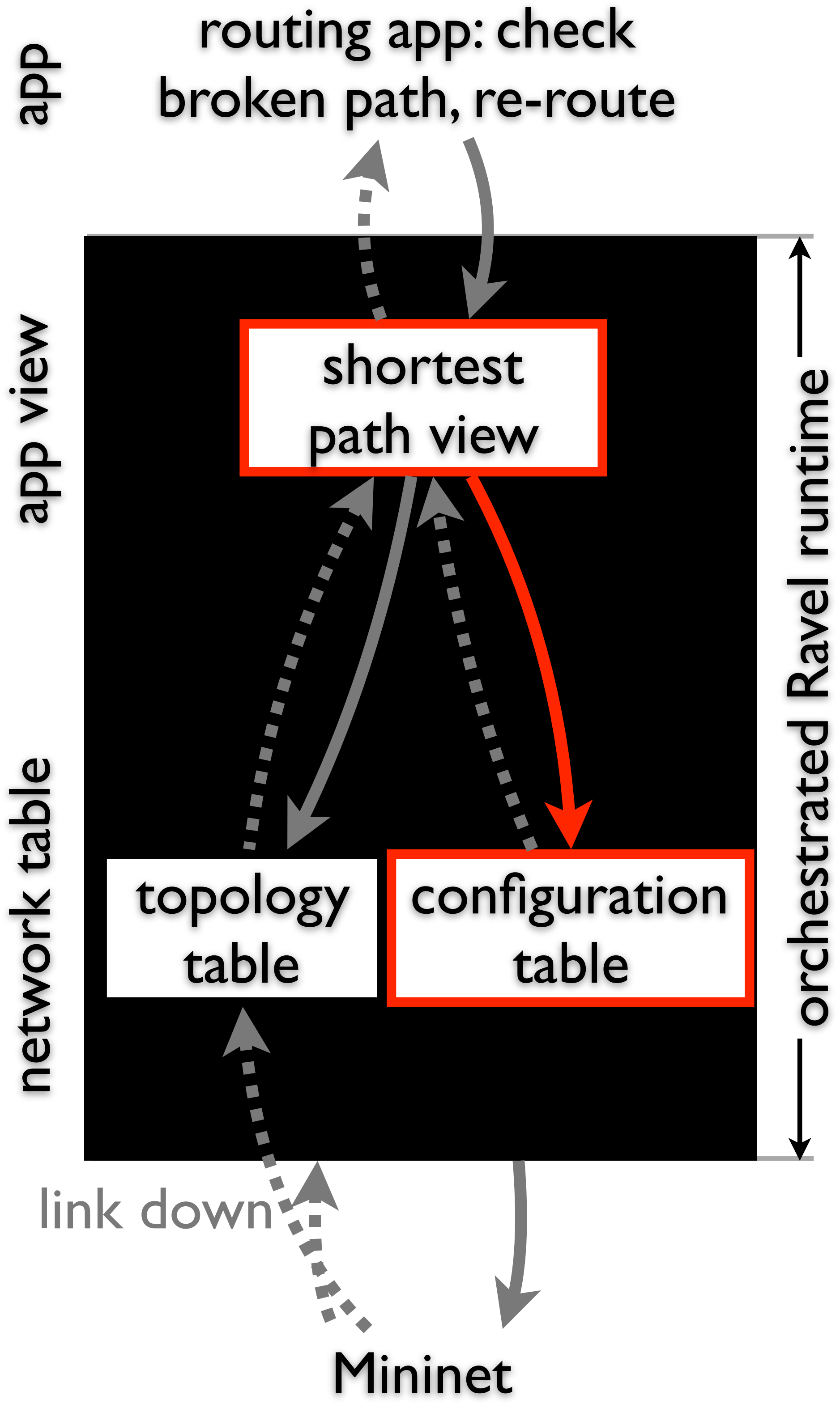
	shortest path	
	…	path
-		{…,172,39,156,…}
+		{…,172,38,148,…}

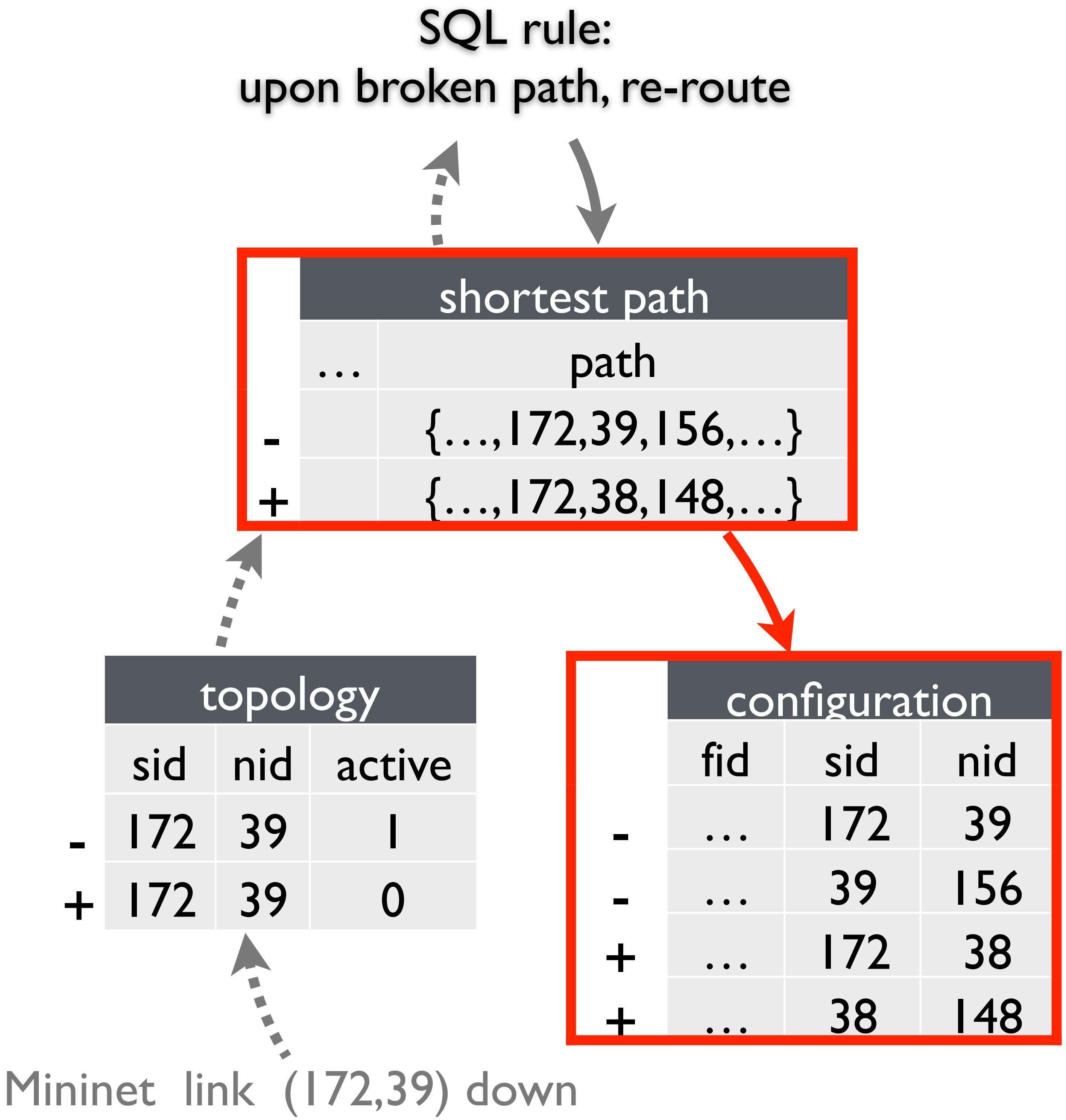
	topology		
	sid	nid	active
-	172	39	1
+	172	39	0

configuration		

Mininet link (172,39) down

orchestration across representations


app
routing app: check broken path, re-route
app view
shortest path view
network table
topology table
configuration table
orchestrated Ravel runtime
link down
Mininet


SQL rule:
upon broken path, re-route

shortest path

	…	path
-		{…,172,39,156,…}
+		{…,172,38,148,…}

topology

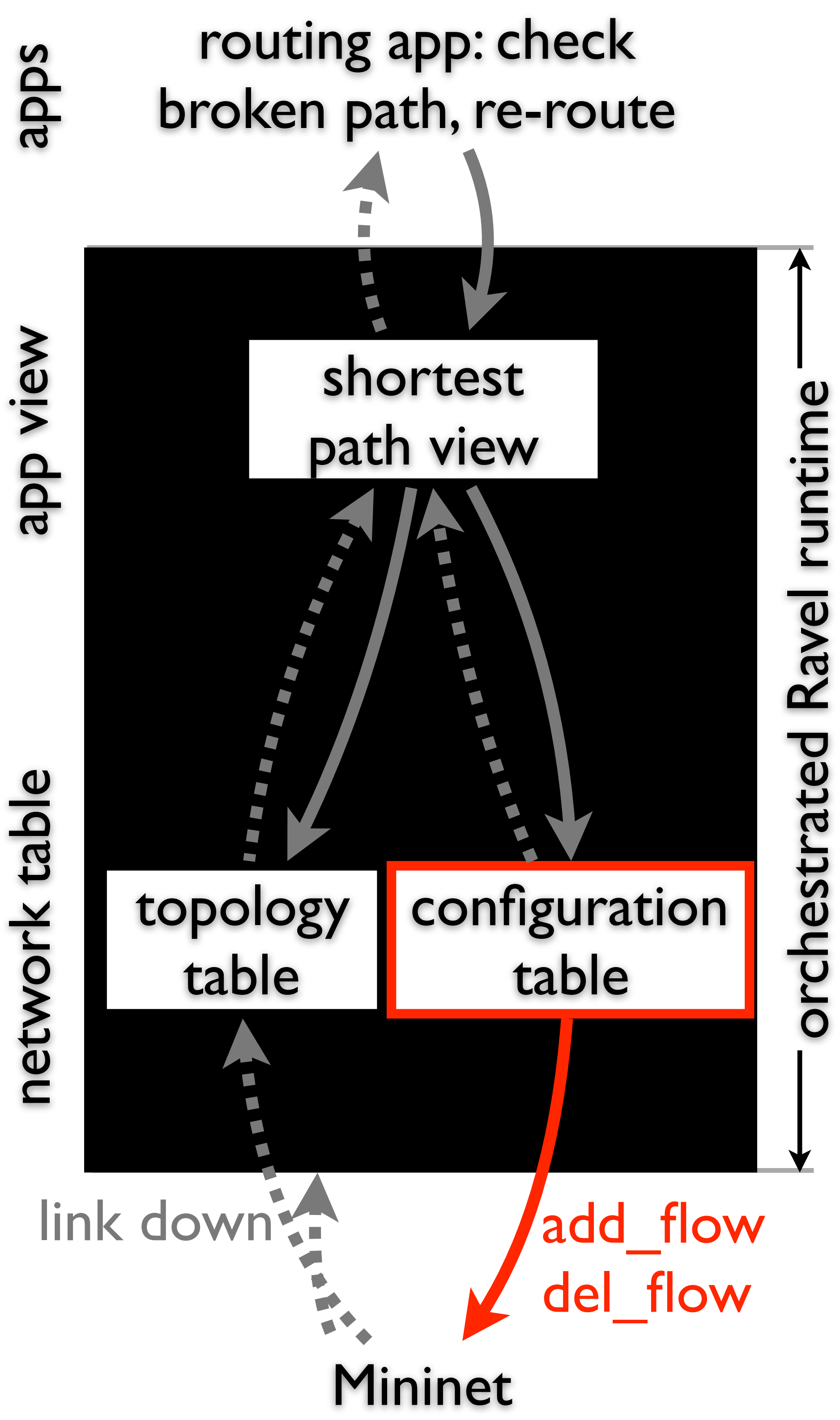
	sid	nid	active
-	172	39	1
+	172	39	0

configuration

	fid	sid	nid
-	…	172	39
-	…	39	156
+	…	172	38
+	…	38	148

Mininet link (172,39) down

orchestration across representations



apps

app view

network table

routing app: check broken path, re-route

shortest path view

topology table

configuration table

orchestrated Ravel runtime

link down

add_flow

del_flow

Mininet

SQL rule:
upon broken path, re-route

shortest path

	…	path
-		{…,172,39,156,…}
+		{…,172,38,148,…}

topology

	sid	nid	active
-	172	39	1
+	172	39	0

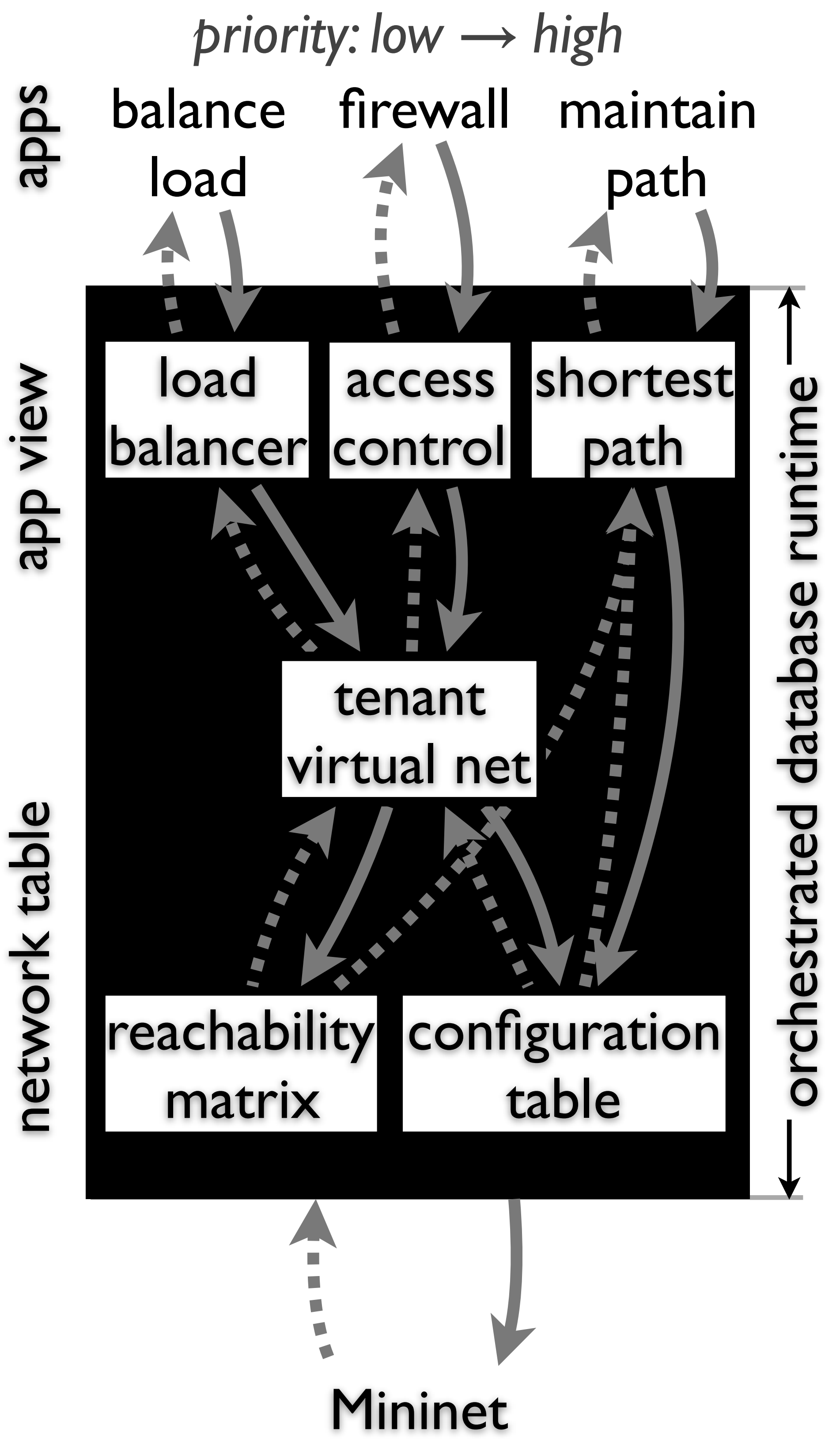
Mininet link (172,39) down

configuration

	fid	sid	nid
-	…	172	39
-	…	39	156
+	…	172	38
+	…	38	148

orchestrated updates:
re-route via (172, 38)

orchestration across applications


priority: low → high
apps
balance load
firewall
maintain path
app view
load balancer
access control
shortest path
tenant virtual net
network table
reachability matrix
configuration table
orchestrated database runtime
Mininet

load balancer	

access control		

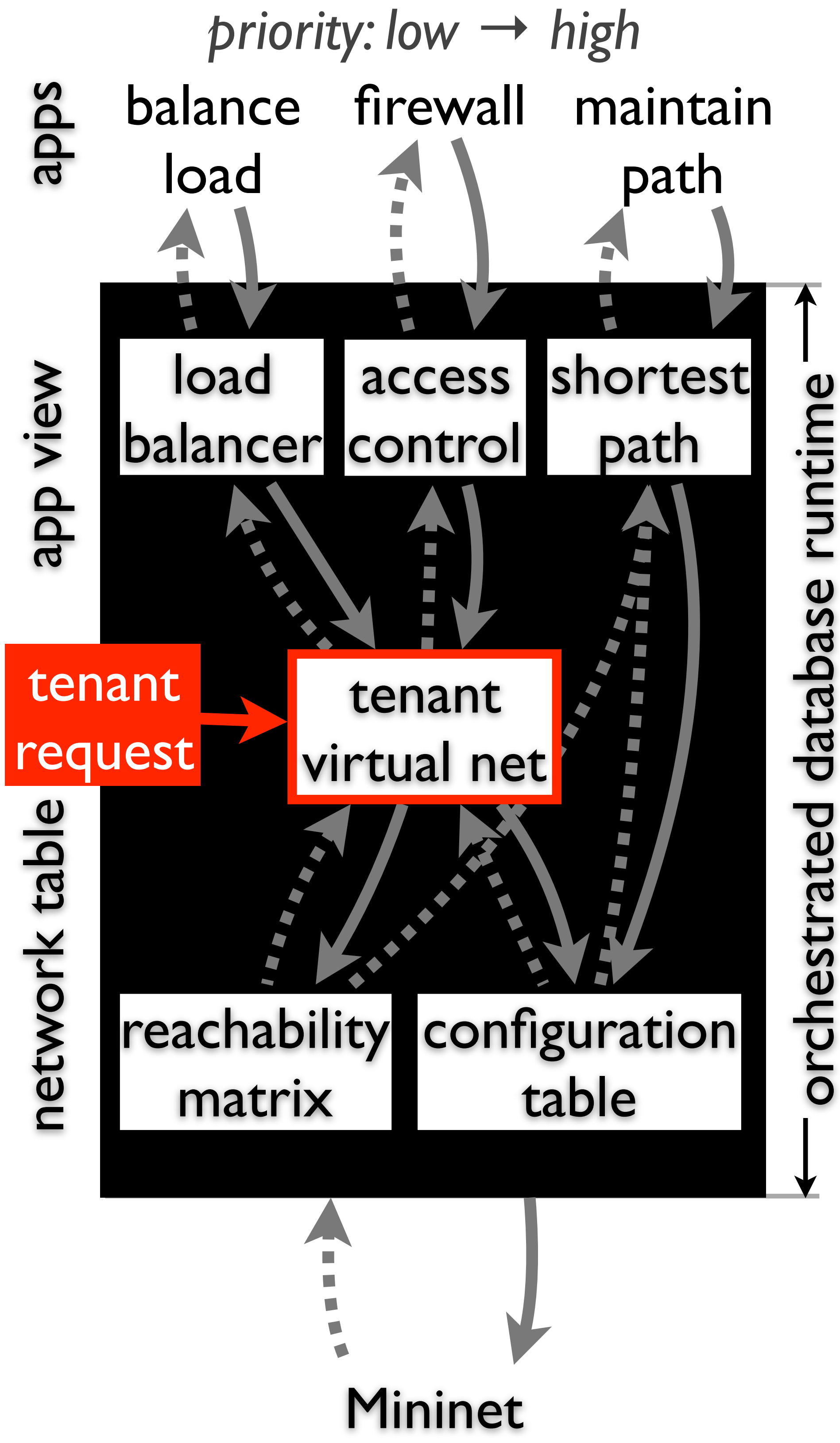
shortest path	

tenant virtual net		

reachability matrix		

configuration		

orchestration across applications


priority: low → high
apps
balance load
firewall
maintain path
app view
load balancer
access control
shortest path
tenant request
tenant virtual net
network table
reachability matrix
configuration table
orchestrated database runtime
Mininet

load balancer	

access control		

shortest path	

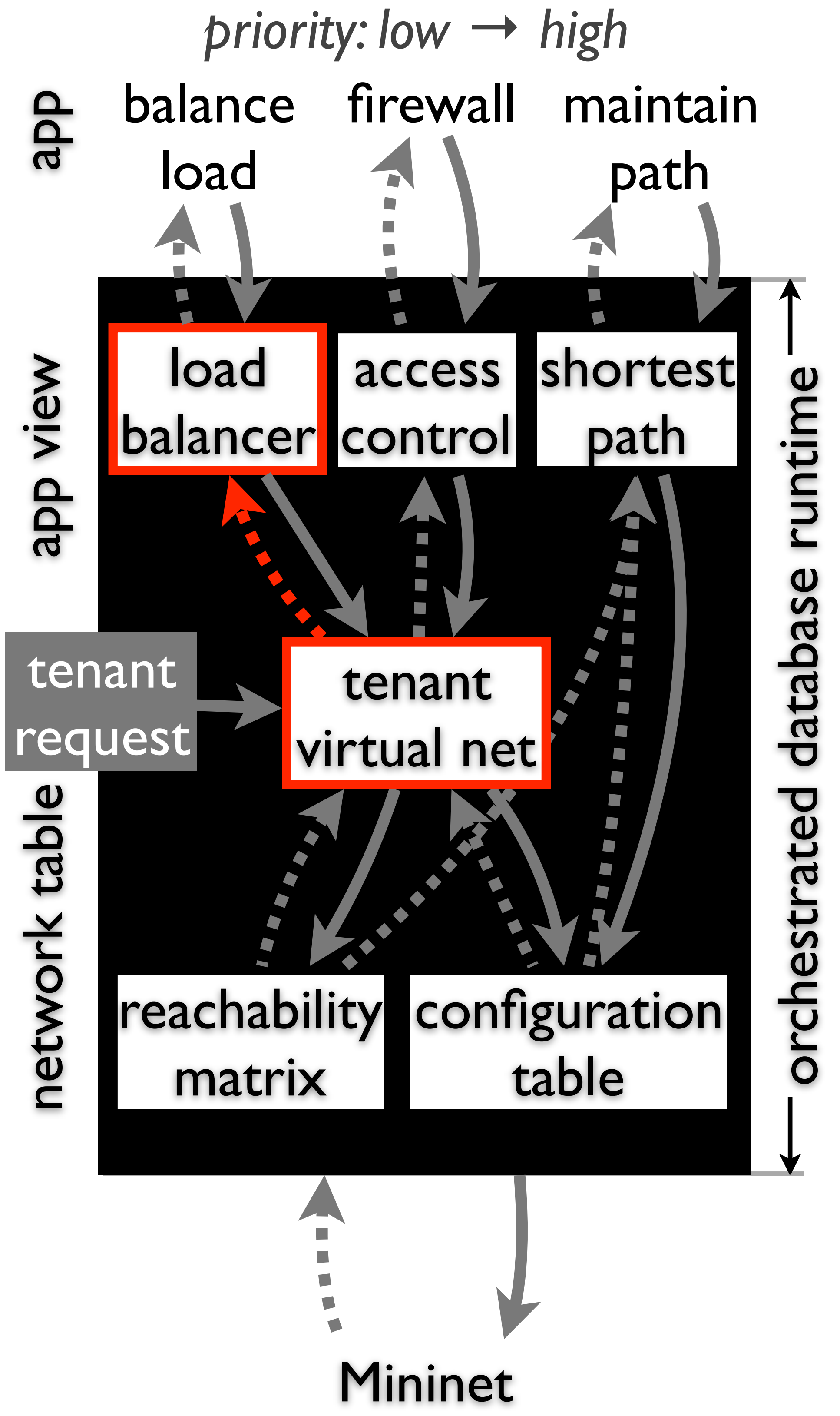
tenant request host 1238 to server 1003

	tenant virtual net		
	…	host	server
+	…	1238	1003

reachability matrix		

configuration		

orchestration across applications


priority: low → high
app
balance load
firewall
maintain path
app view
load balancer
access control
shortest path
tenant request
tenant virtual net
network table
reachability matrix
configuration table
orchestrated database runtime
Mininet

load

	sid	load
+	1003	4
-	1003	3

access control

shortest path

tenant request host 1238 to server 1003

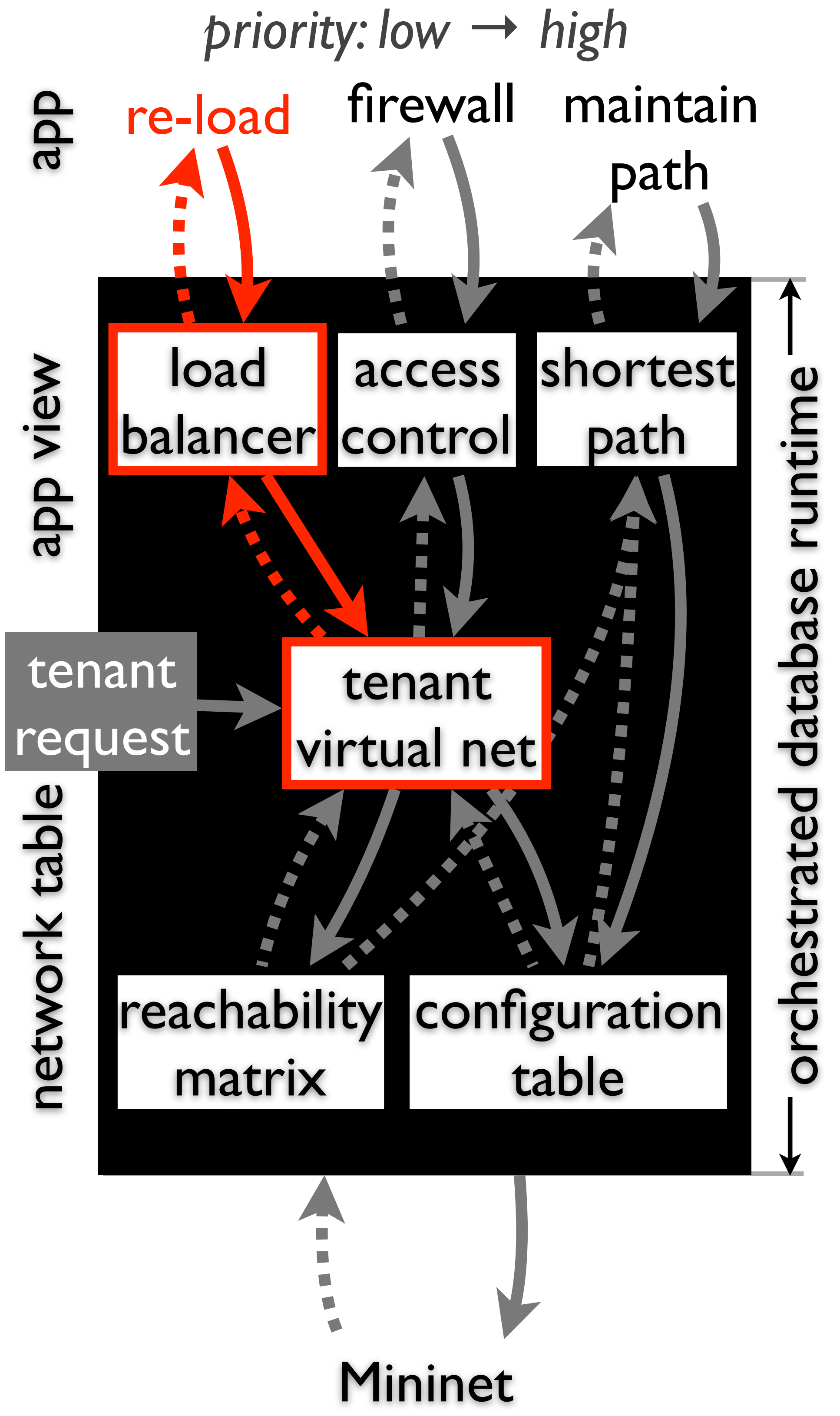
tenant virtual net

	…	host	server
+	…	1238	1003

reachability matrix

configuration

orchestration across applications



priority: low → high

app

re-load

firewall

maintain path

app view

load balancer

access control

shortest path

tenant request

tenant virtual net

network table

reachability matrix

configuration table

orchestrated database runtime

Mininet

	load	
	sid	load
+	~~1003~~	4
-	~~1003~~	~~3~~
-	1034	1
+	1034	2

access control		

shortest path	

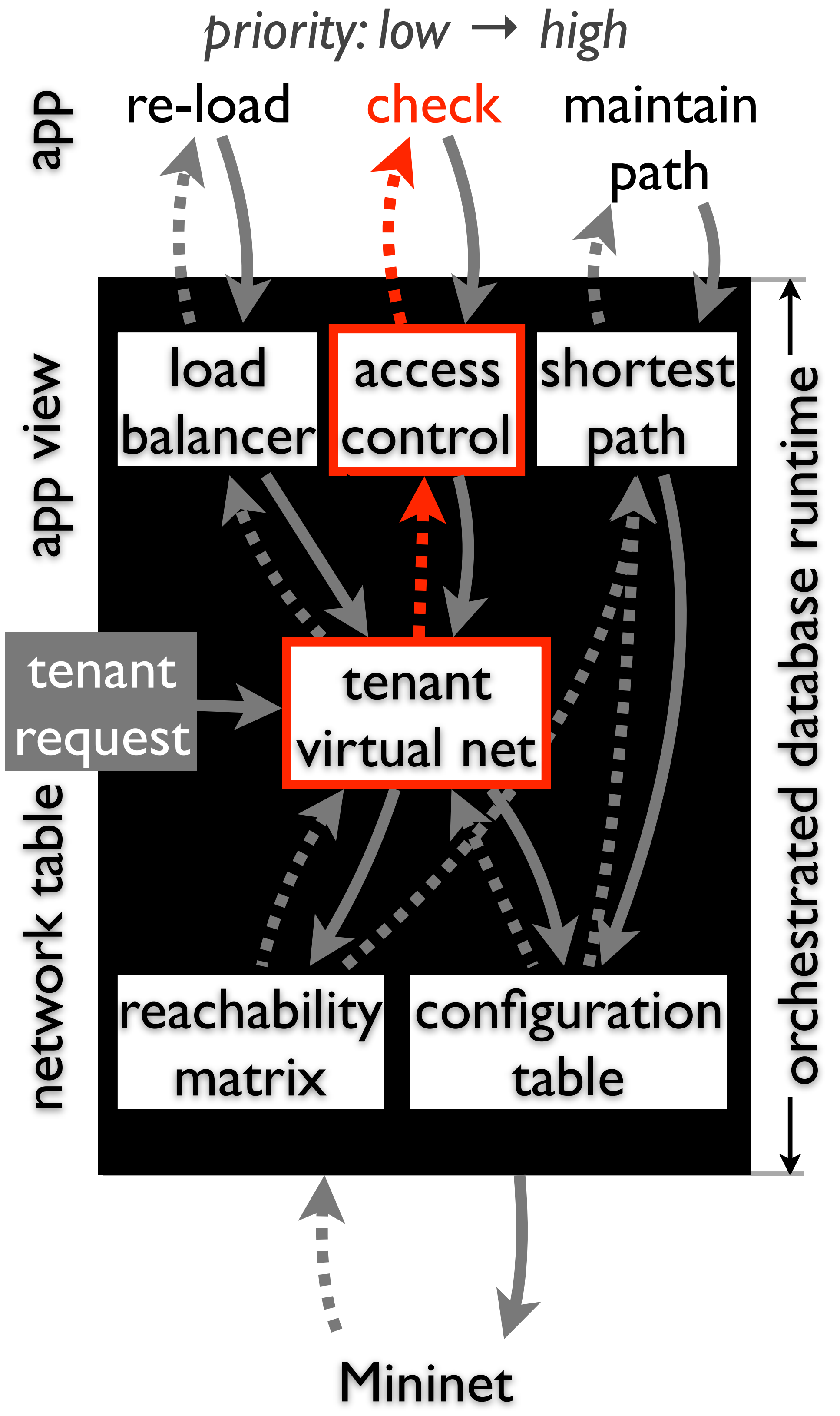
tenant request host 1238 to server 1003

	tenant virtual net		
	…	host	server
+	~~…~~	~~1238~~	~~1003~~
+		1238	1034

reachability matrix		

configuration		

orchestration across applications


priority: low → high
app
re-load
check
maintain path
app view
load balancer
access control
shortest path
tenant request
tenant virtual net
network table
reachability matrix
configuration table
orchestrated database runtime
Mininet

	load	
	sid	load
+	~~1003~~	4
-	~~1003~~	~~3~~
-	1034	1
+	1034	2

access control		
src	dst	allow
1238	1034	1
1238	1003	0

shortest path

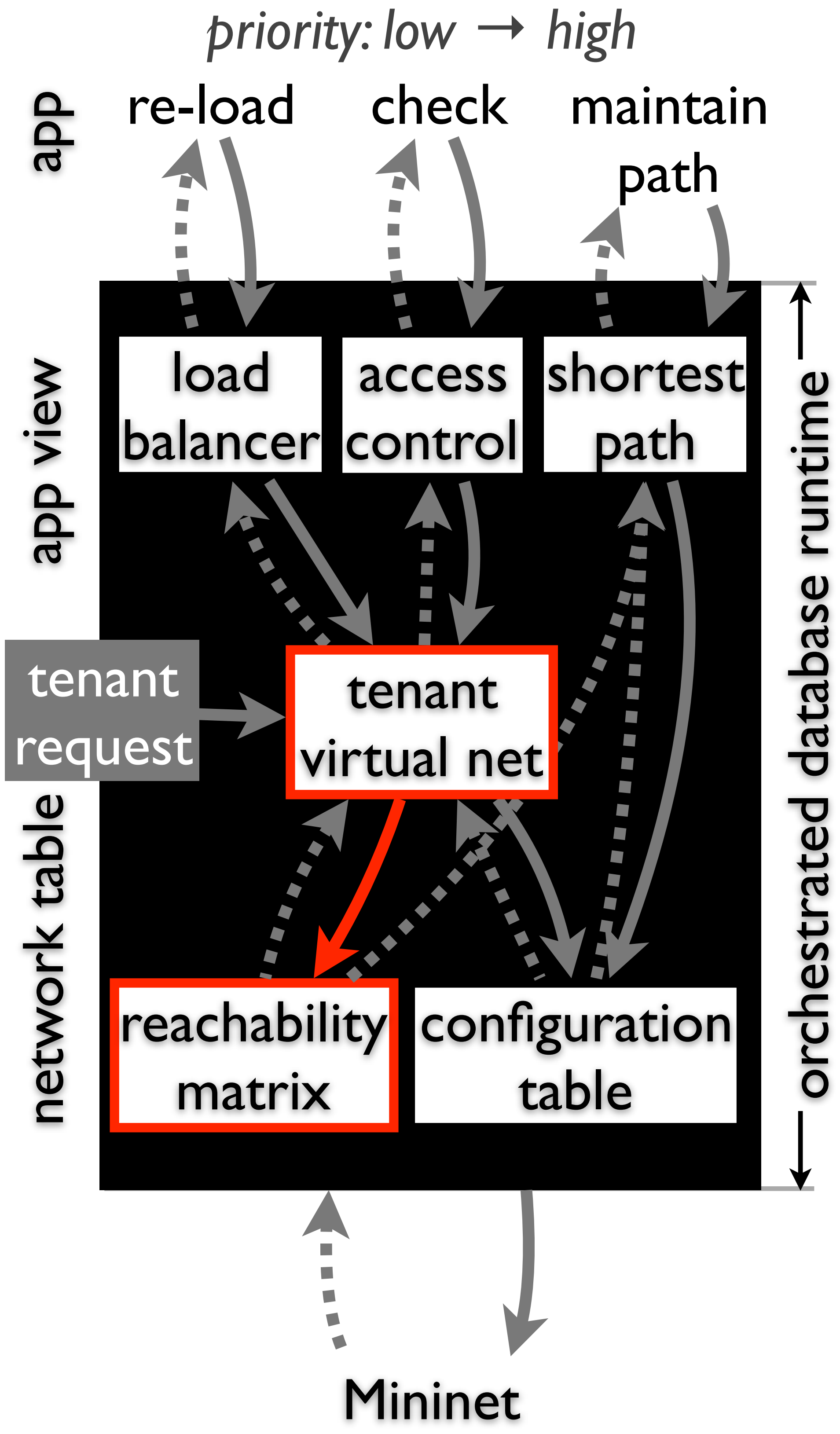
tenant request host 1238 to server 1003

	tenant virtual net		
	…	host	server
+	~~…~~	~~1238~~	~~1003~~
+		1238	1034

reachability matrix

configuration

orchestration across applications


priority: low → high
app
re-load
check
maintain path
app view
load balancer
access control
shortest path
tenant request
tenant virtual net
network table
reachability matrix
configuration table
orchestrated database runtime
Mininet

load

	sid	load
+	~~1003~~	4
-	~~1003~~	~~3~~
-	1034	1
+	1034	2

access control

src	dst	allow
1238	1034	1
1238	1003	0

shortest path

tenant virtual net

	…	host	server
+	~~…~~	~~1238~~	~~1003~~
+		1238	1034

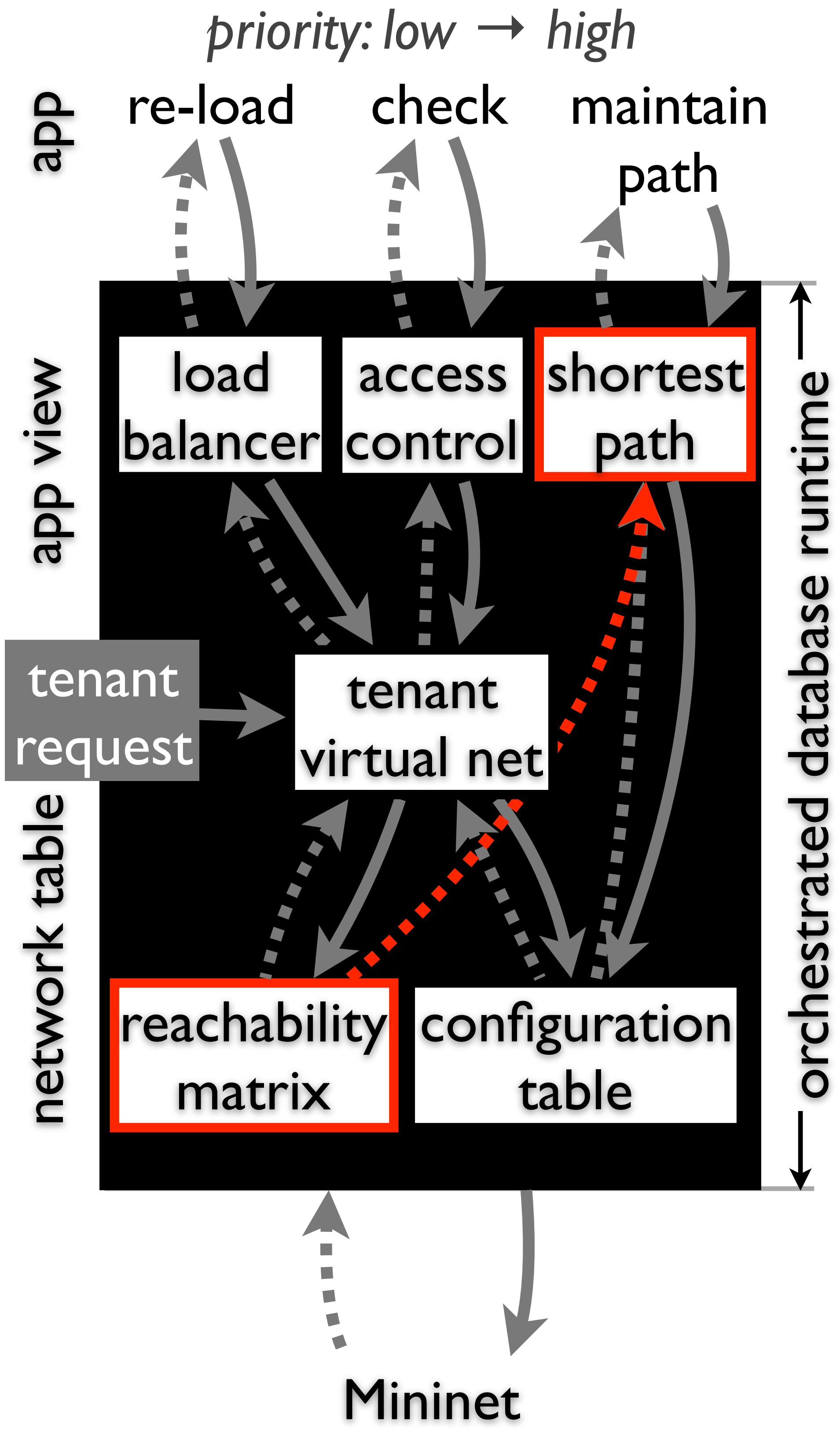
tenant request host 1238 to server 1003

reachability matrix

	fid	sid	nid
+	…	1238	1034

configuration

orchestration across applications


priority: low → high
app
re-load
check
maintain path
app view
load balancer
access control
shortest path
tenant request
tenant virtual net
network table
reachability matrix
configuration table
orchestrated database runtime
Mininet

load

	sid	load
+	~~1003~~	4
-	~~1003~~	~~3~~
-	1034	1
+	1034	2

access control

src	dst	allow
1238	1034	1
1238	1003	0

shortest path

	…	path
+	…	{1238,…,1034}

tenant request host 1238 to server 1003

tenant virtual net

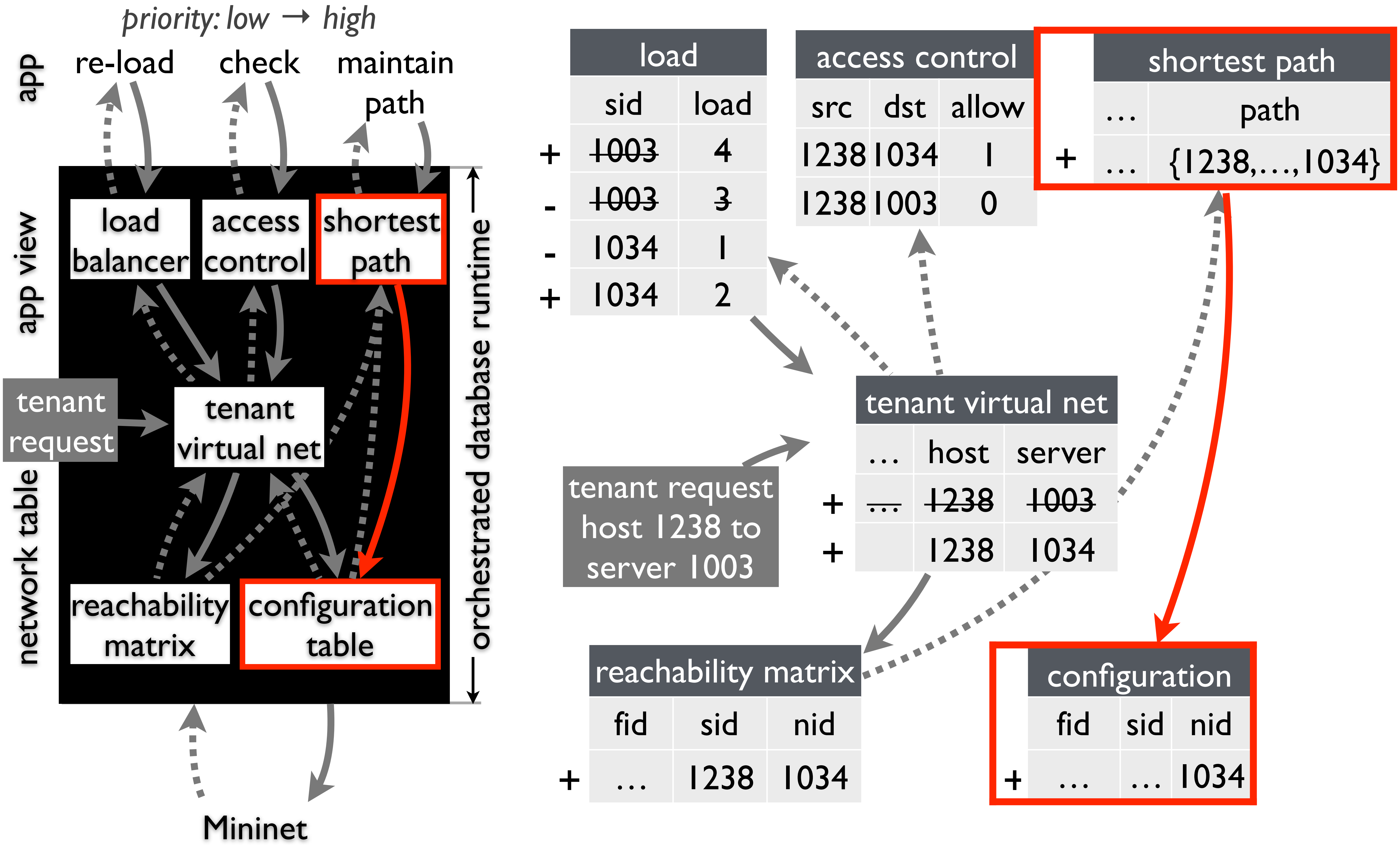
	…	host	server
~~+~~	~~…~~	~~1238~~	~~1003~~
+		1238	1034

reachability matrix

	fid	sid	nid
+	…	1238	1034

configuration

orchestration across applications



priority: low → high
app
re-load
check
maintain path
app view
load balancer
access control
shortest path
network table
tenant request
tenant virtual net
reachability matrix
configuration table
orchestrated database runtime
Mininet

	load	
	sid	load
+	~~1003~~	4
-	~~1003~~	~~3~~
-	1034	1
+	1034	2

access control		
src	dst	allow
1238	1034	1
1238	1003	0

	shortest path	
	…	path
+	…	{1238,…,1034}

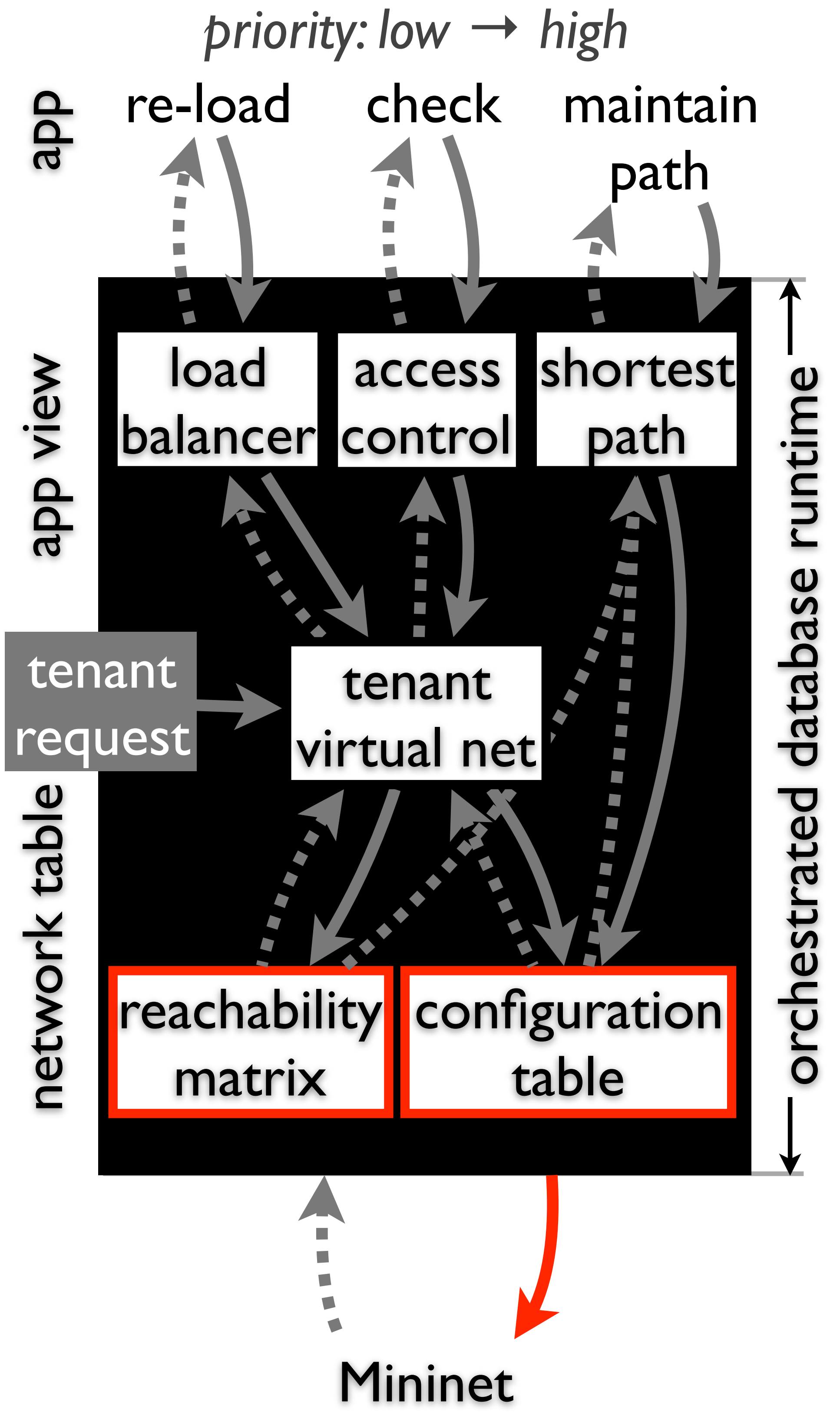
tenant request host 1238 to server 1003

	tenant virtual net		
	…	host	server
+	~~…~~	~~1238~~	~~1003~~
+		1238	1034

	reachability matrix		
	fid	sid	nid
+	…	1238	1034

	configuration		
	fid	sid	nid
+	…	…	1034

orchestration across applications



priority: low → high

app

re-load

check

maintain path

app view

load balancer

access control

shortest path

tenant request

tenant virtual net

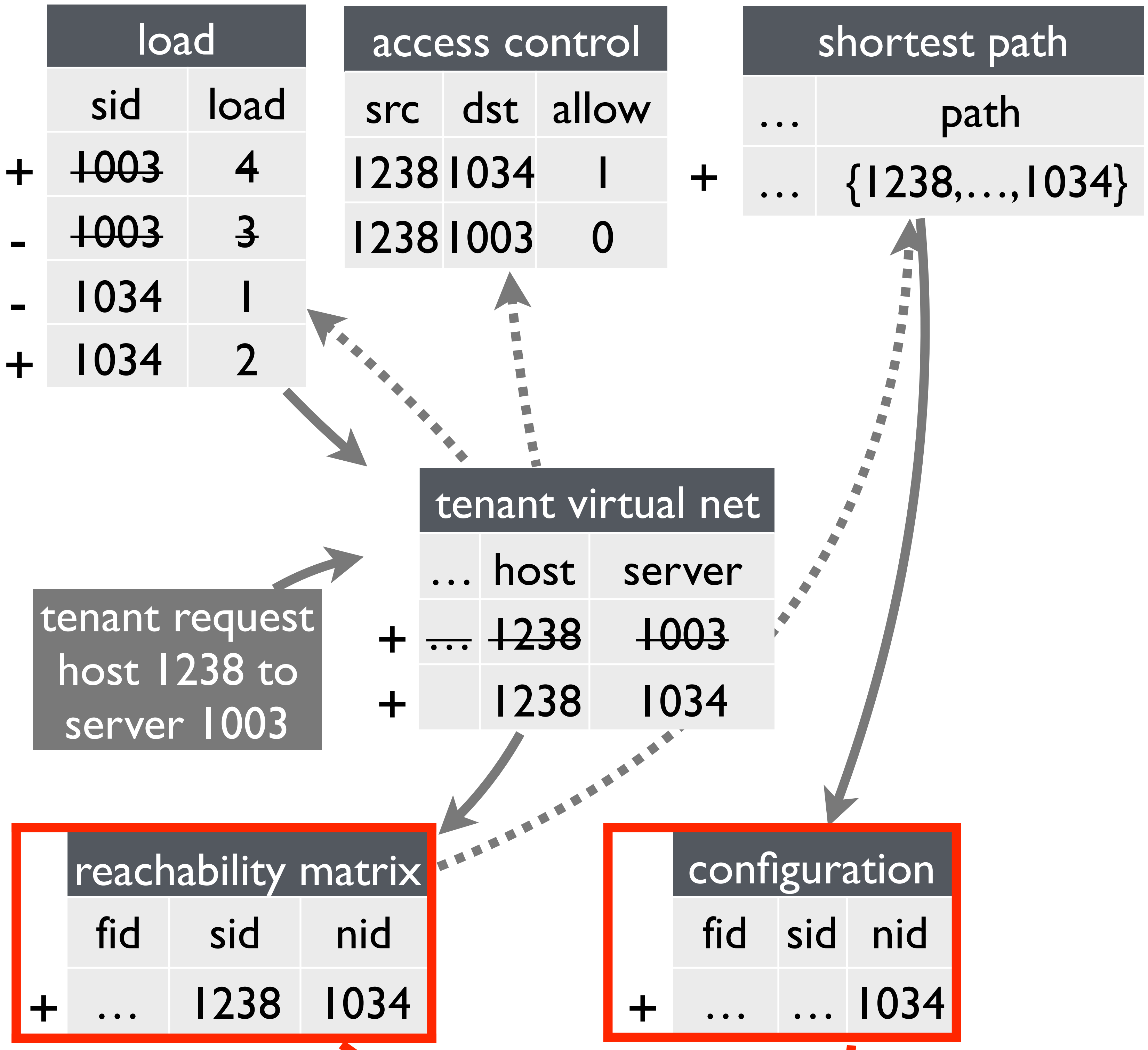
network table

reachability matrix

configuration table

orchestrated database runtime

Mininet



	load	
	sid	load
+	~~1003~~	4
-	~~1003~~	~~3~~
-	1034	1
+	1034	2

access control		
src	dst	allow
1238	1034	1
1238	1003	0

	shortest path	
	…	path
+	…	{1238,…,1034}

tenant request host 1238 to server 1003

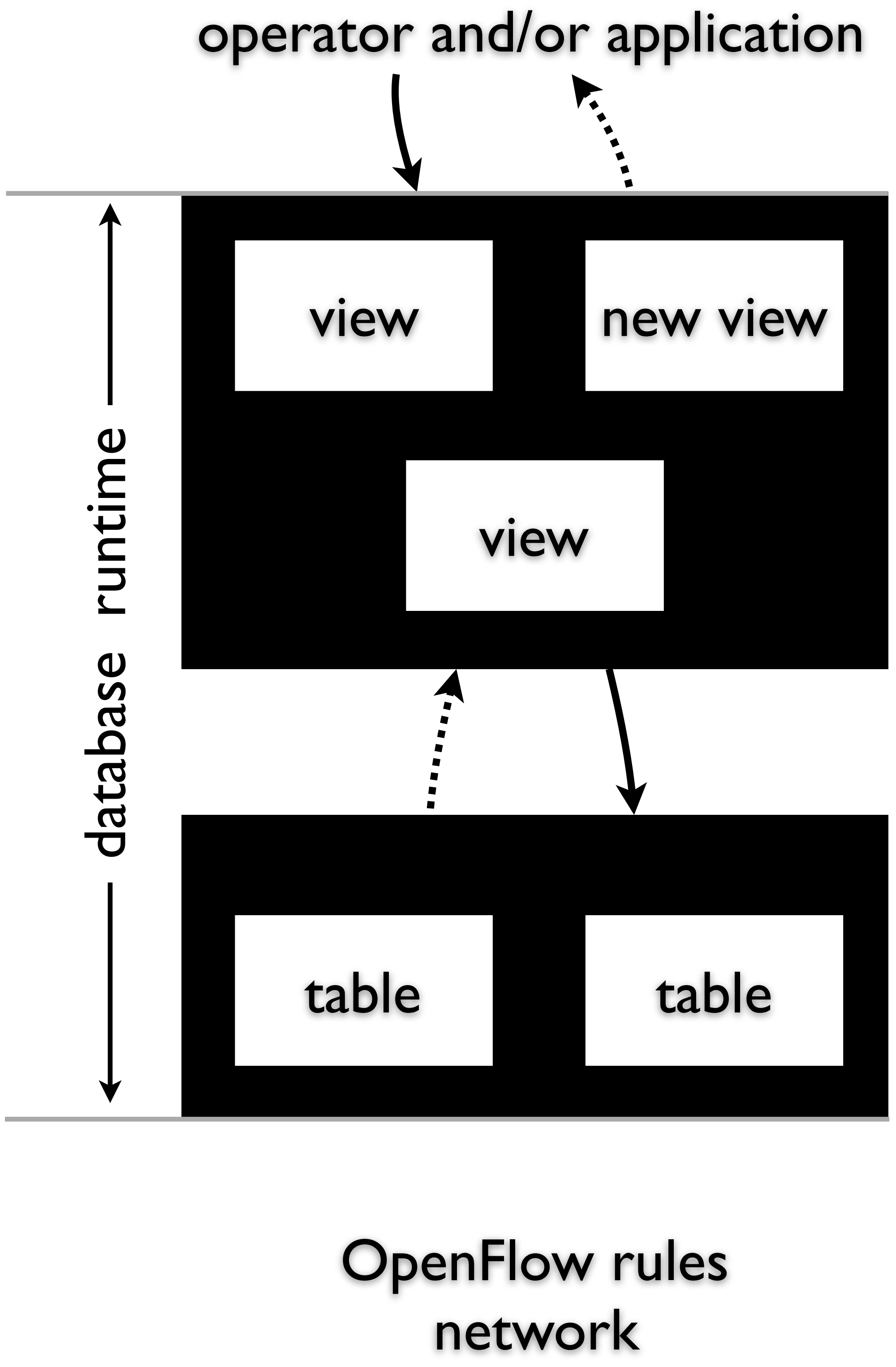
	tenant virtual net		
	…	host	server
+	~~…~~	~~1238~~	~~1003~~
+		1238	1034

	reachability matrix		
	fid	sid	nid
+	…	1238	1034

	configuration		
	fid	sid	nid
+	…	…	1034

orchestrated updates: install alternative route that is load-balanced and safe

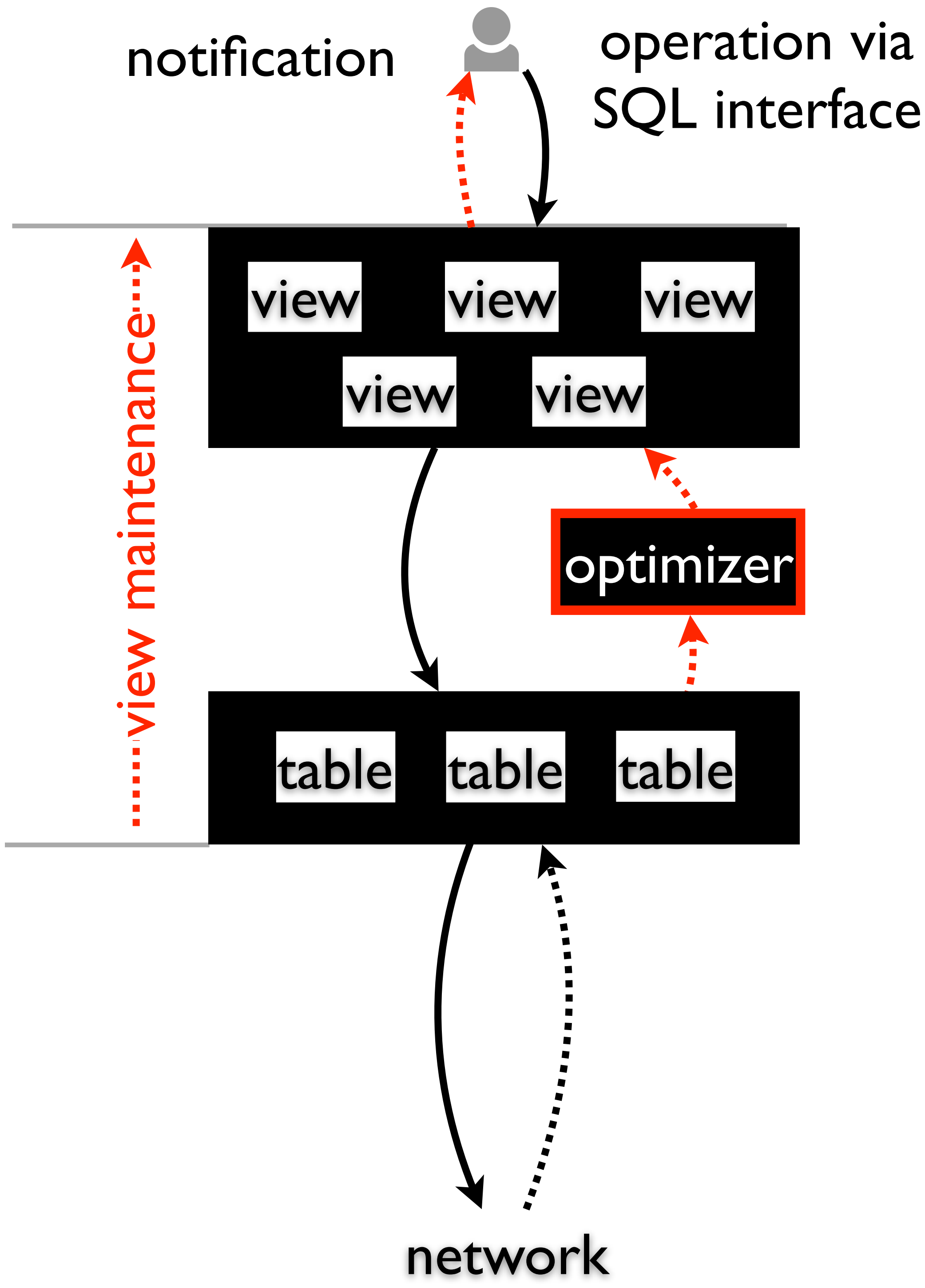
achieving *Ravel* advantages


operator and/or application
view
new view
view
database runtime
table
table
OpenFlow rules
network

attractive features

- ad-hoc programmable abstraction via views
- orchestration across abstractions via view mechanism
- orchestration across applications via data mediation
- network control via SQL

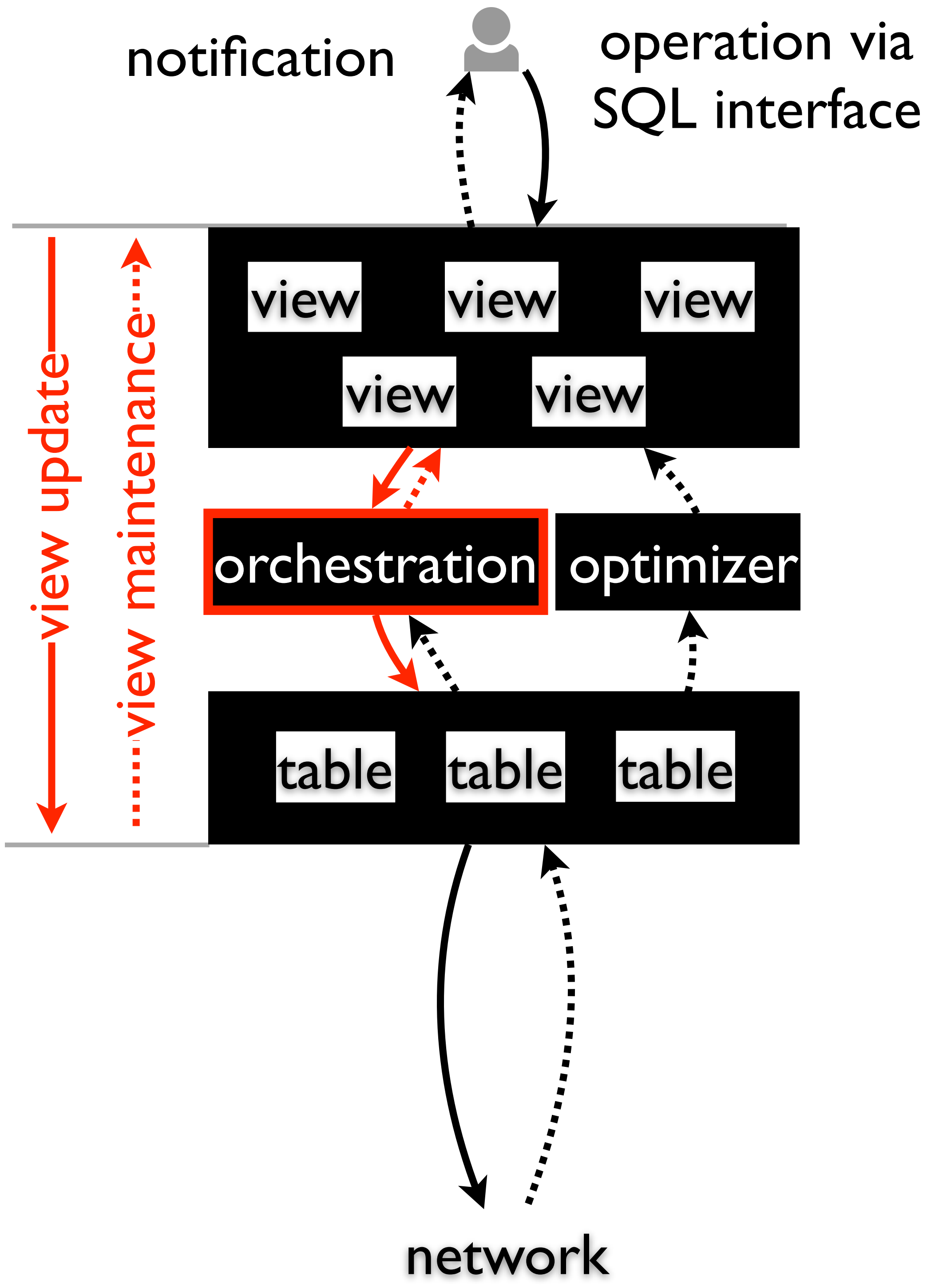
runtime


notification
operation via SQL interface
view
view
view
view
view
view maintenance
optimizer
table
table
table
network

ad-hoc programmable abstraction via views

- challenge: inefficient user view
- solution: optimizer
  - materialize user view with fast maintenance algorithm

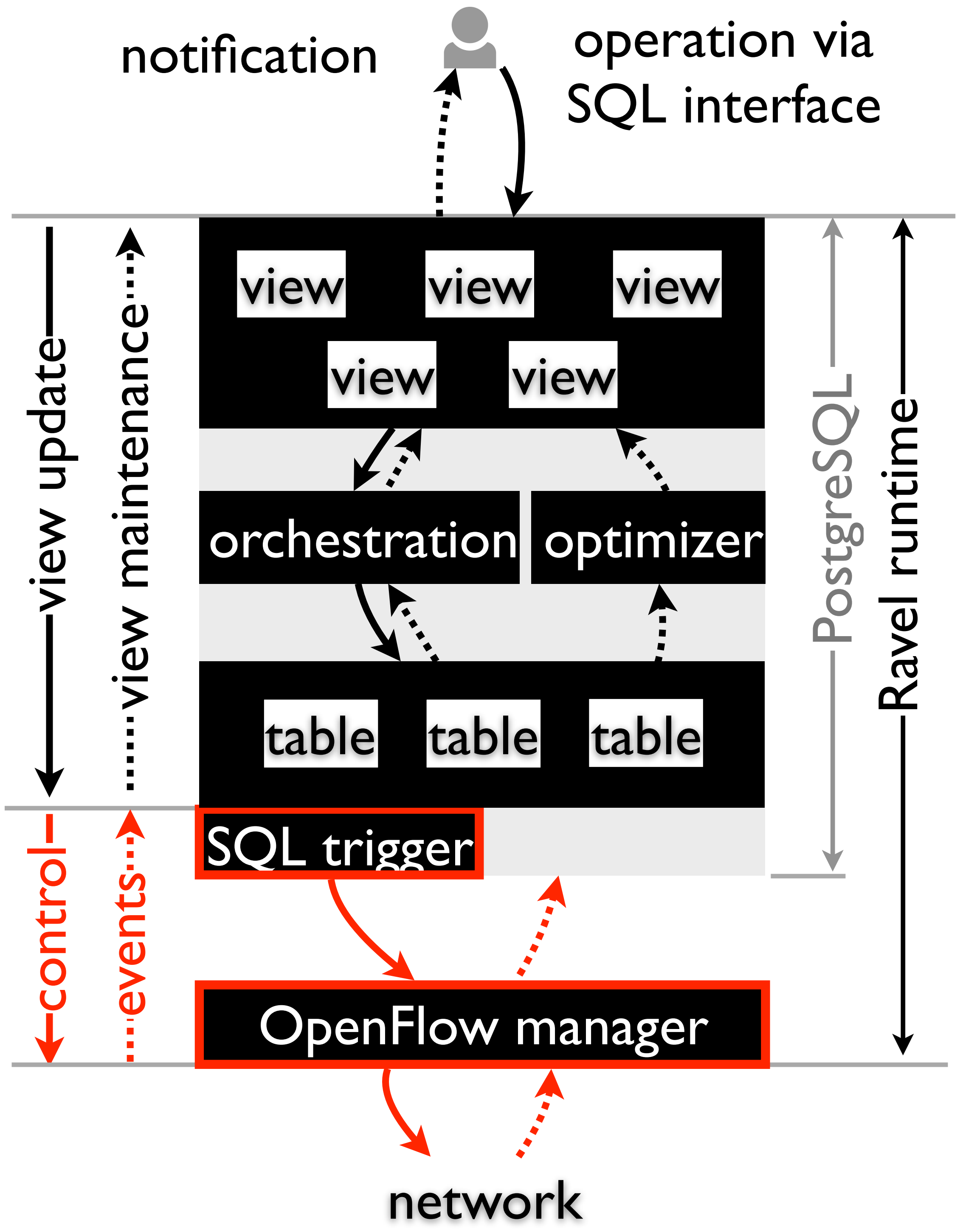
runtime


notification
operation via SQL interface
view update
view maintenance
view
view
view
view
view
orchestration
optimizer
table
table
table
network

orchestration across applications

- challenge: database lacking inter-view support
- solution: mediation protocol
  - translate app priority into view updates that dynamically merge into a coherent data plane

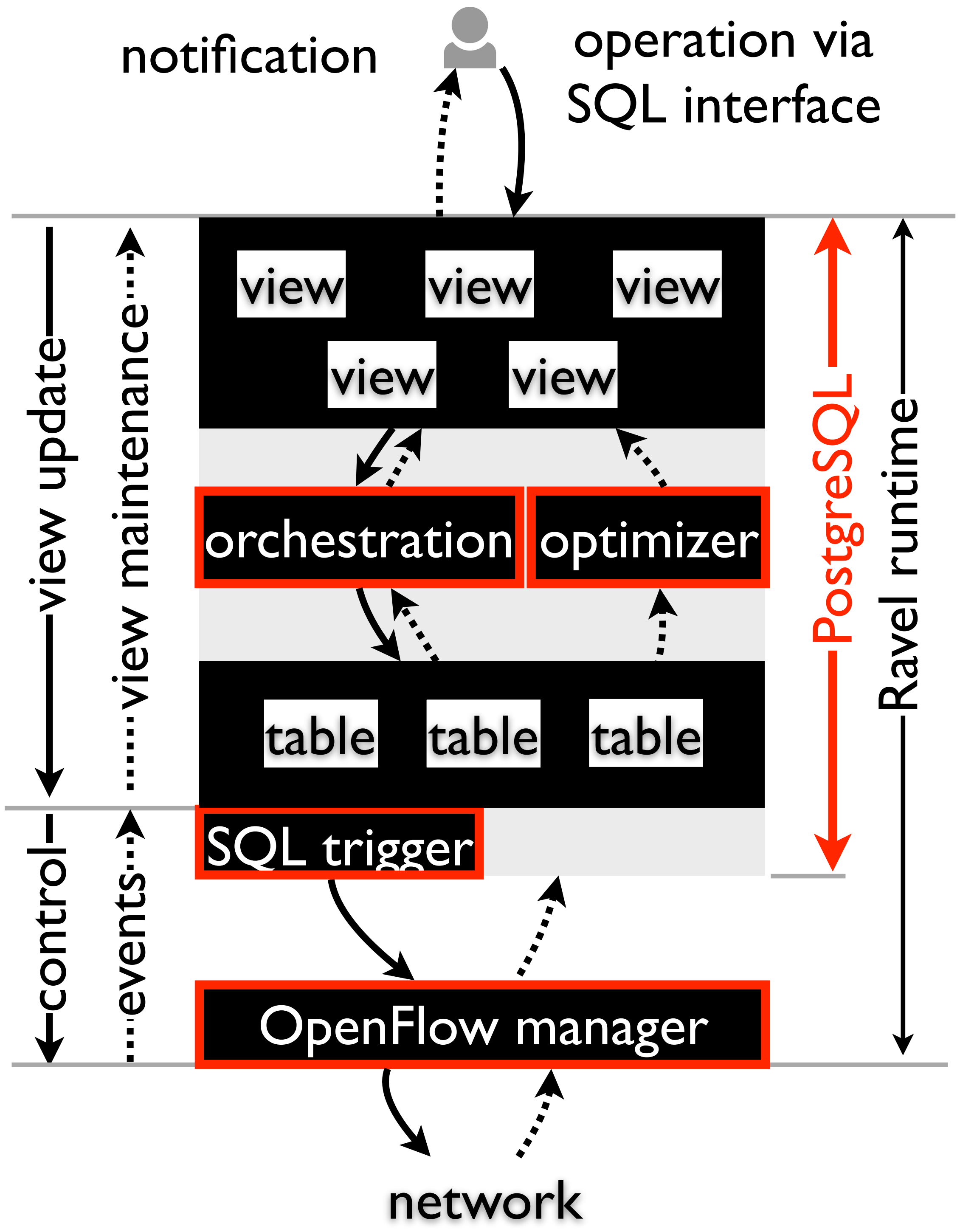
runtime


notification
operation via SQL interface
view
view
view
view
view
orchestration
optimizer
table
table
table
SQL trigger
OpenFlow manager
network
view update
view maintenance
control
events
PostgreSQL
Ravel runtime

SDN control via SQL

- challenge: database lacks connection to network data plane
- solution: SQL trigger + OF manager

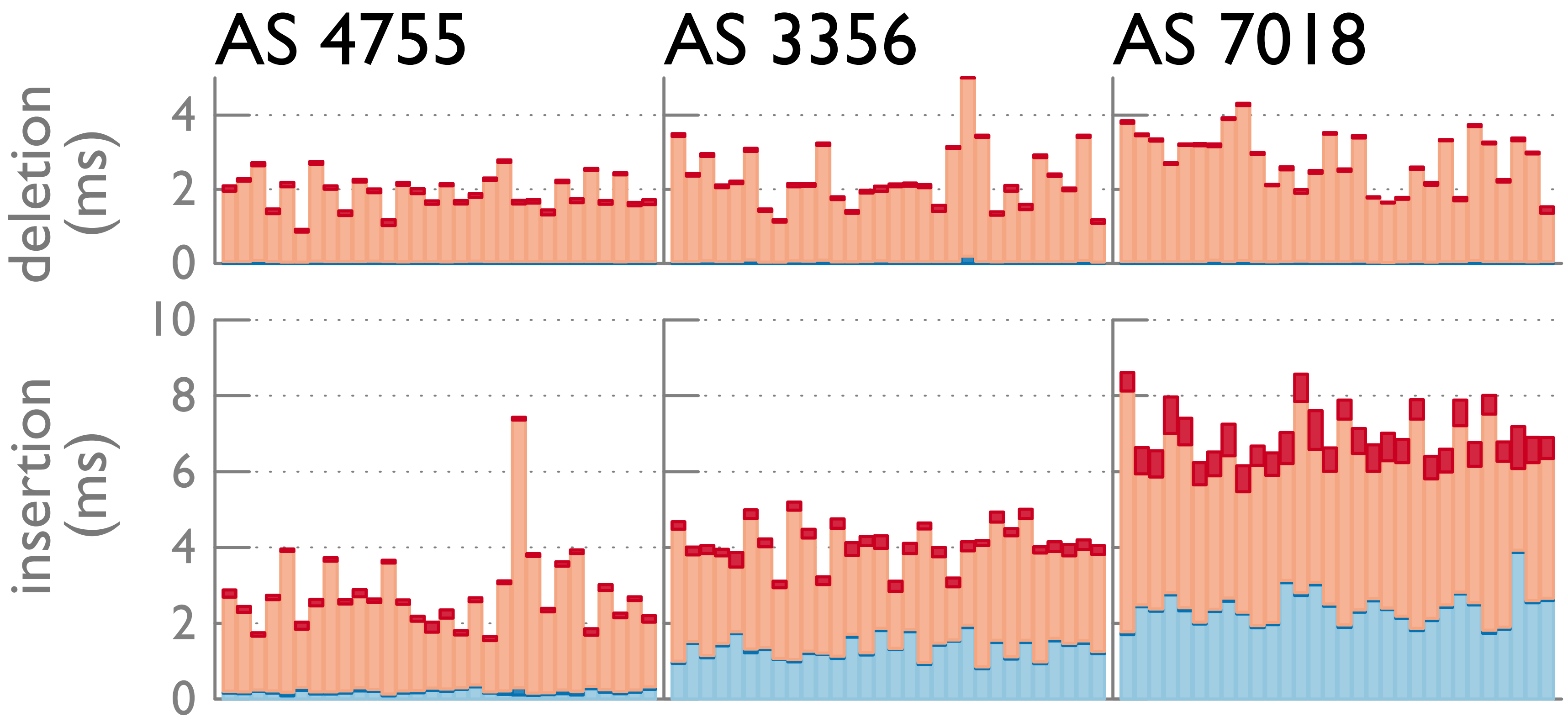
runtime


notification
operation via SQL interface
view
view
view
view
view
orchestration
optimizer
table
table
table
SQL trigger
OpenFlow manager
network
view update
view maintenance
control
events
PostgreSQL
Ravel runtime

a high-performance runtime

- PostgreSQL
- orchestration
- optimizer
- SQL trigger and OF manager

evaluation

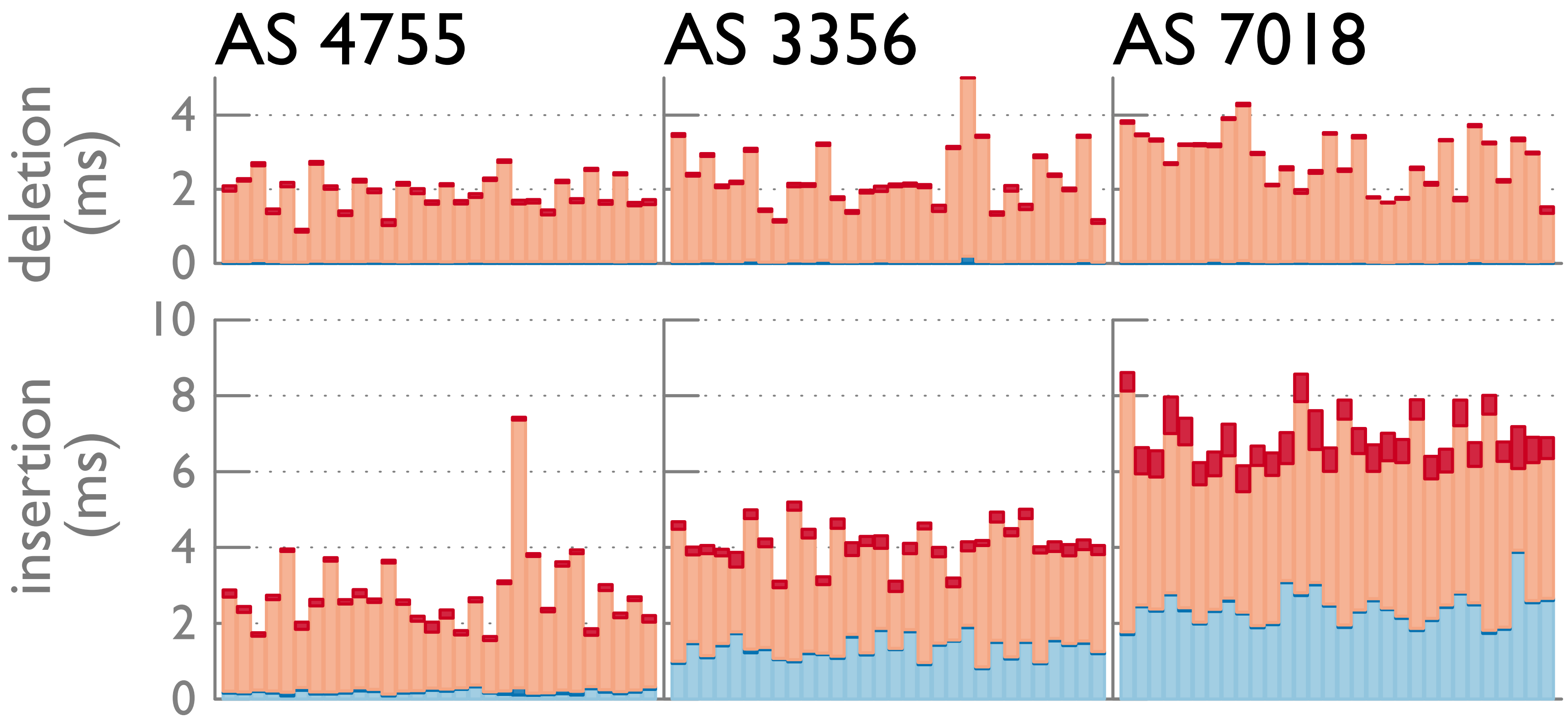


profile end to end delay (normalized per-rule, 30 rounds)
for route insertion and deletion

Rocketfuel ISP topology

AS#	nodes	links
4755	142	258
3356	1772	13640
7018	25382	11292

evaluation

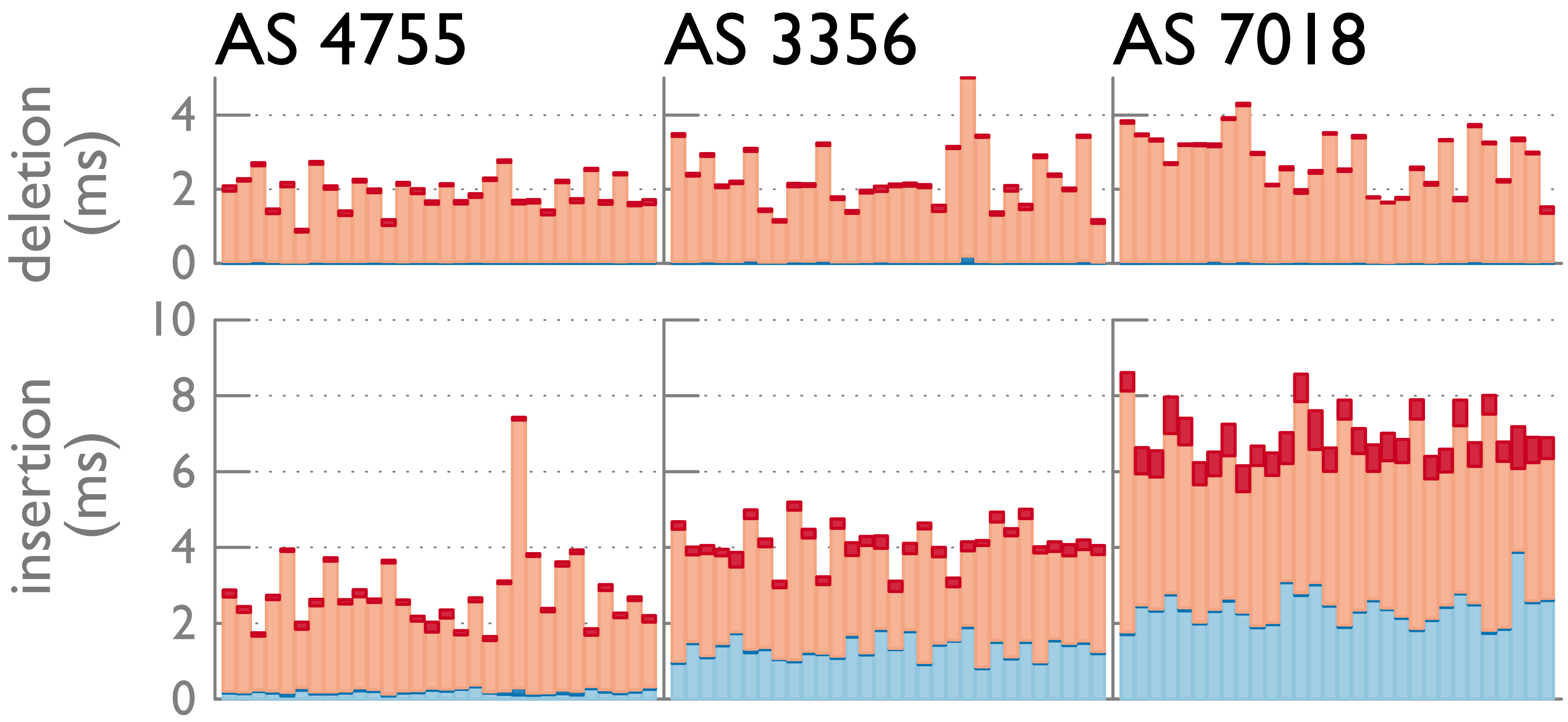


profile end to end delay (normalized per-rule, 30 rounds) for route insertion and deletion

Rocketfuel ISP topology

AS#	nodes	links
4755	142	258
3356	1772	13640
7018	25382	11292

evaluation

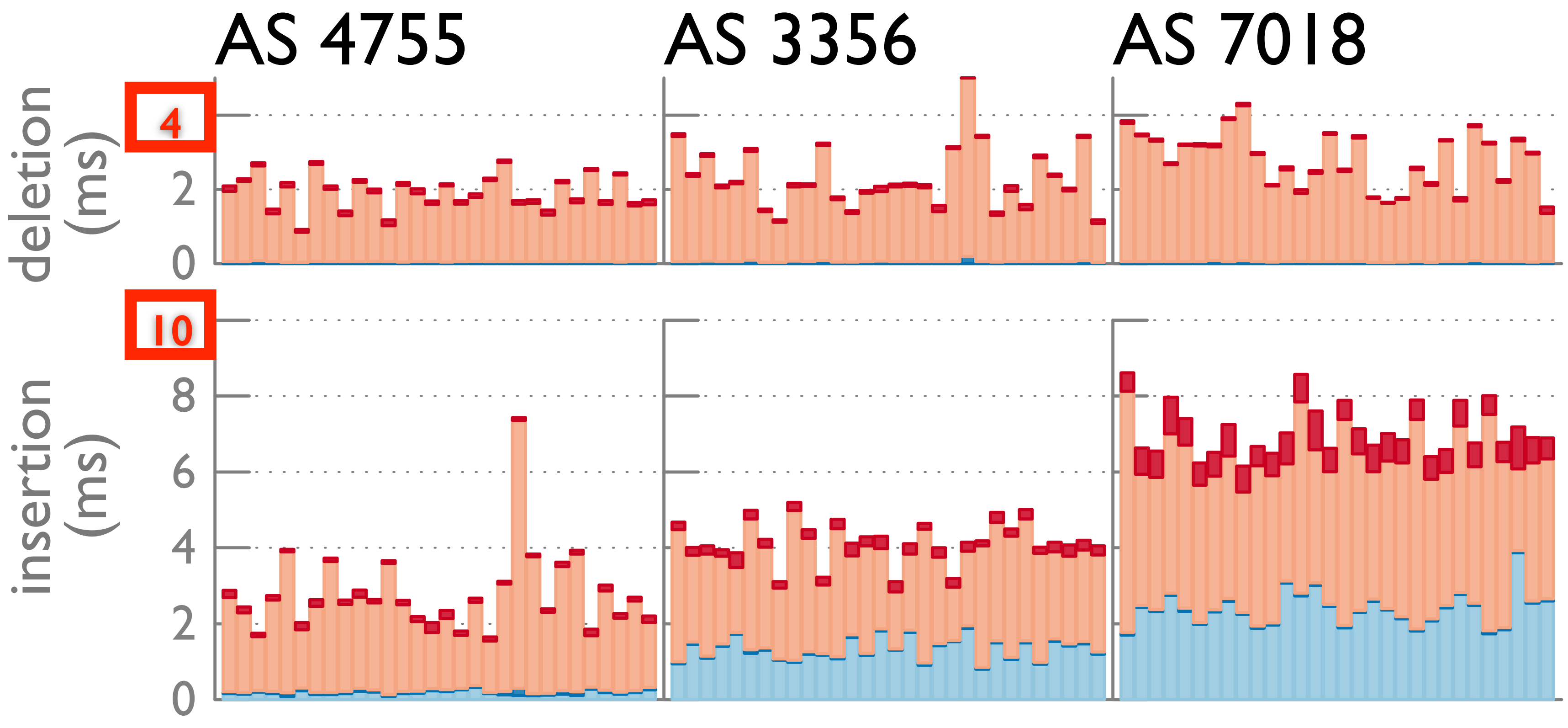


profile end to end delay (normalized per-rule, 30 rounds)
for route insertion and deletion

Rocketfuel ISP topology

AS#	nodes	links
4755	142	258
3356	1772	13640
7018	25382	11292

evaluation



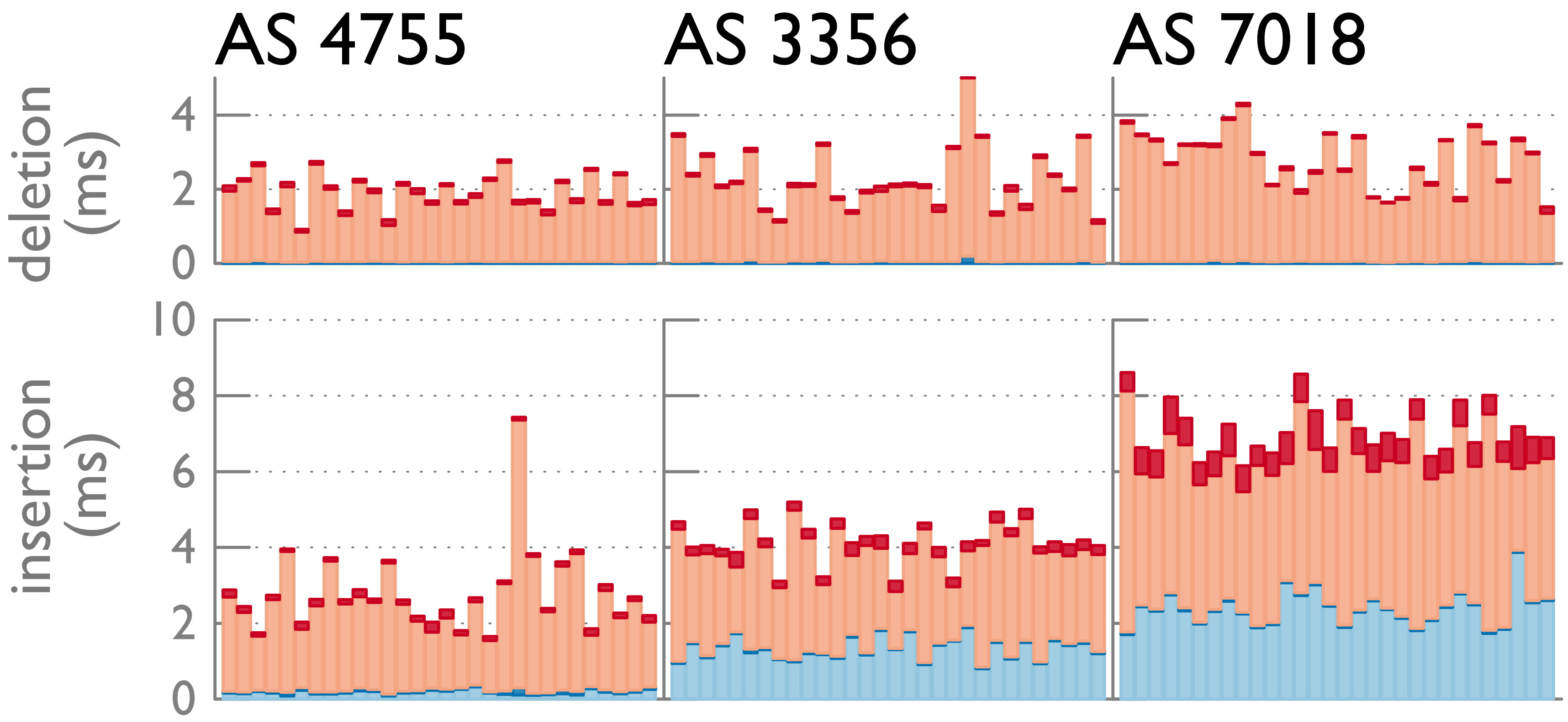
profile end to end delay (normalized per-rule, 30 rounds)
for route insertion and deletion

Rocketfuel ISP topology

AS#	nodes	links
4755	142	258
3356	1772	13640
7018	25382	11292

compute path 🟦
lookup ports 🟦
write to table 🟧
trigger/rule 🟥

evaluation



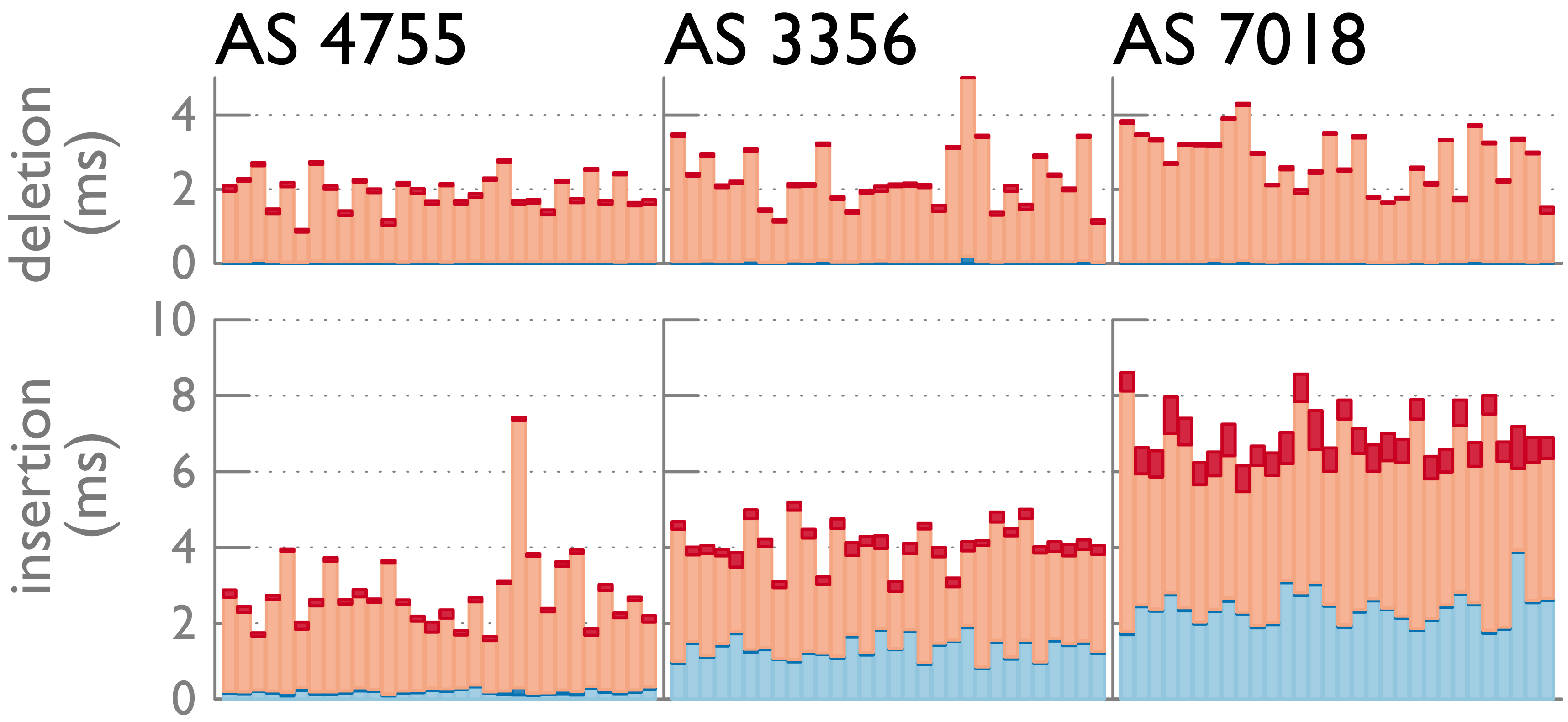
Rocketfuel ISP topology

AS#	nodes	links
4755	142	258
3356	1772	13640
7018	25382	11292

compute path 🟦
lookup ports 🟦
write to table 🟧
trigger/rule 🟥

profile end to end delay (normalized per-rule, 30 rounds)
for route insertion and deletion

evaluation



profile end to end delay (normalized per-rule, 30 rounds) for route insertion and deletion

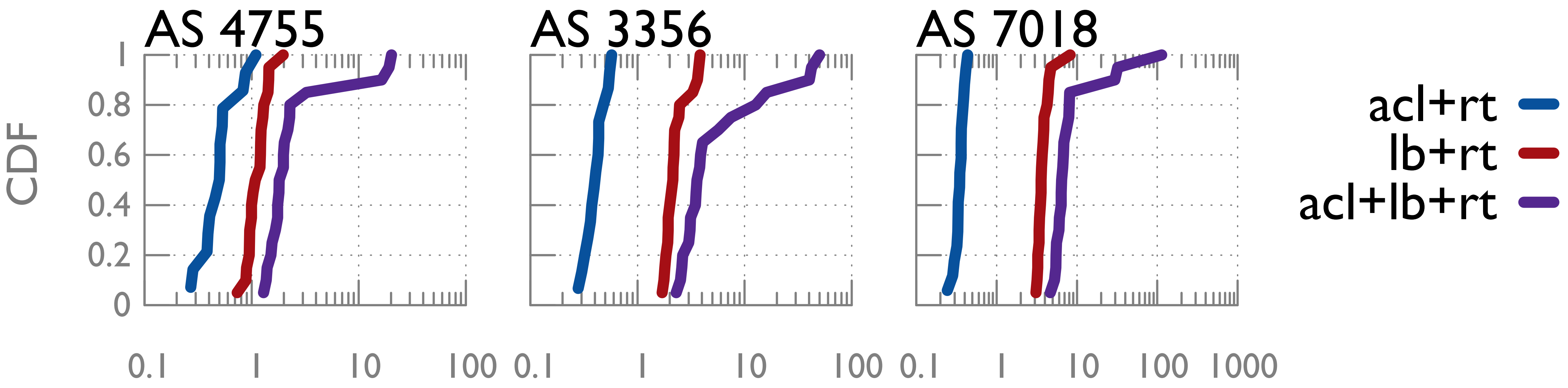
Rocketfuel ISP topology

AS#	nodes	links
4755	142	258
3356	1772	13640
7018	25382	11292

similar profile on fat-tree topology (fewer nodes, more links)

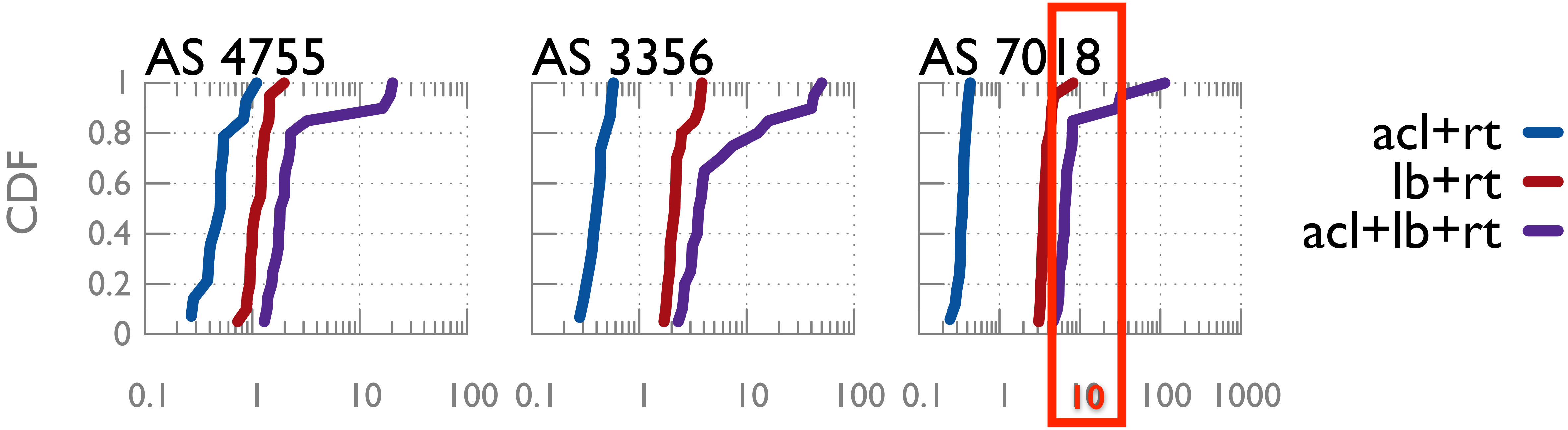
- total delay < 30ms for fat-tree with 5120 switches and 196608 links

evaluation



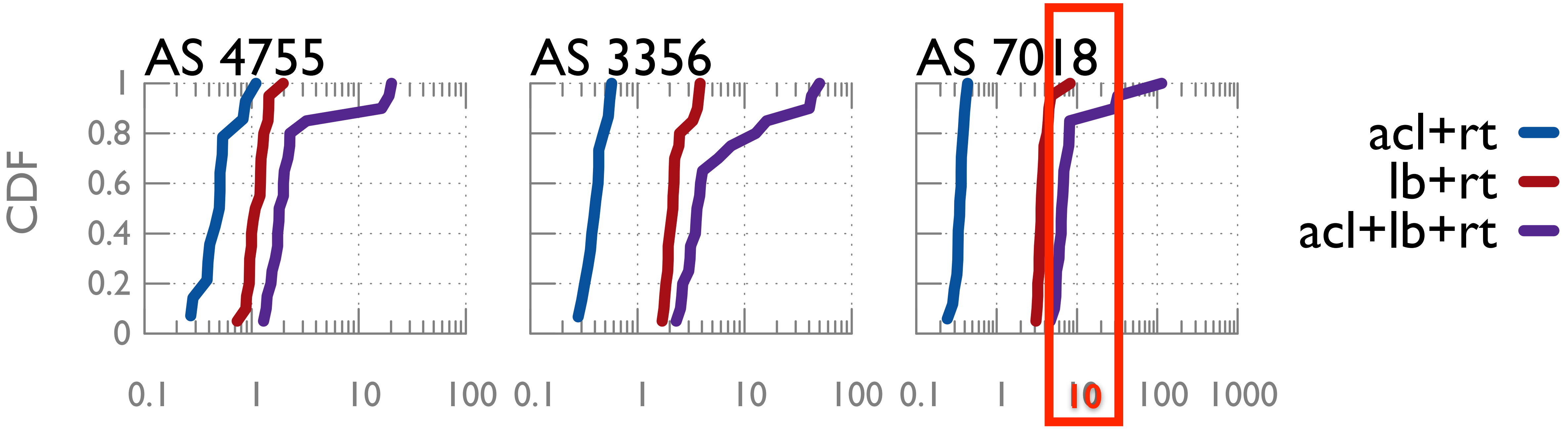
orchestration delay (ms) normalized per-rule for 3 scenarios:
access control and routing (acl+rt), load balancing and routing (lb+rt), access control, load balancing, and routing (acl+lb+rt)

evaluation



orchestration delay (ms) normalized per-rule for 3 scenarios:
access control and routing (acl+rt), load balancing and routing (lb+rt), access control, load balancing, and routing (acl+lb+rt)

evaluation

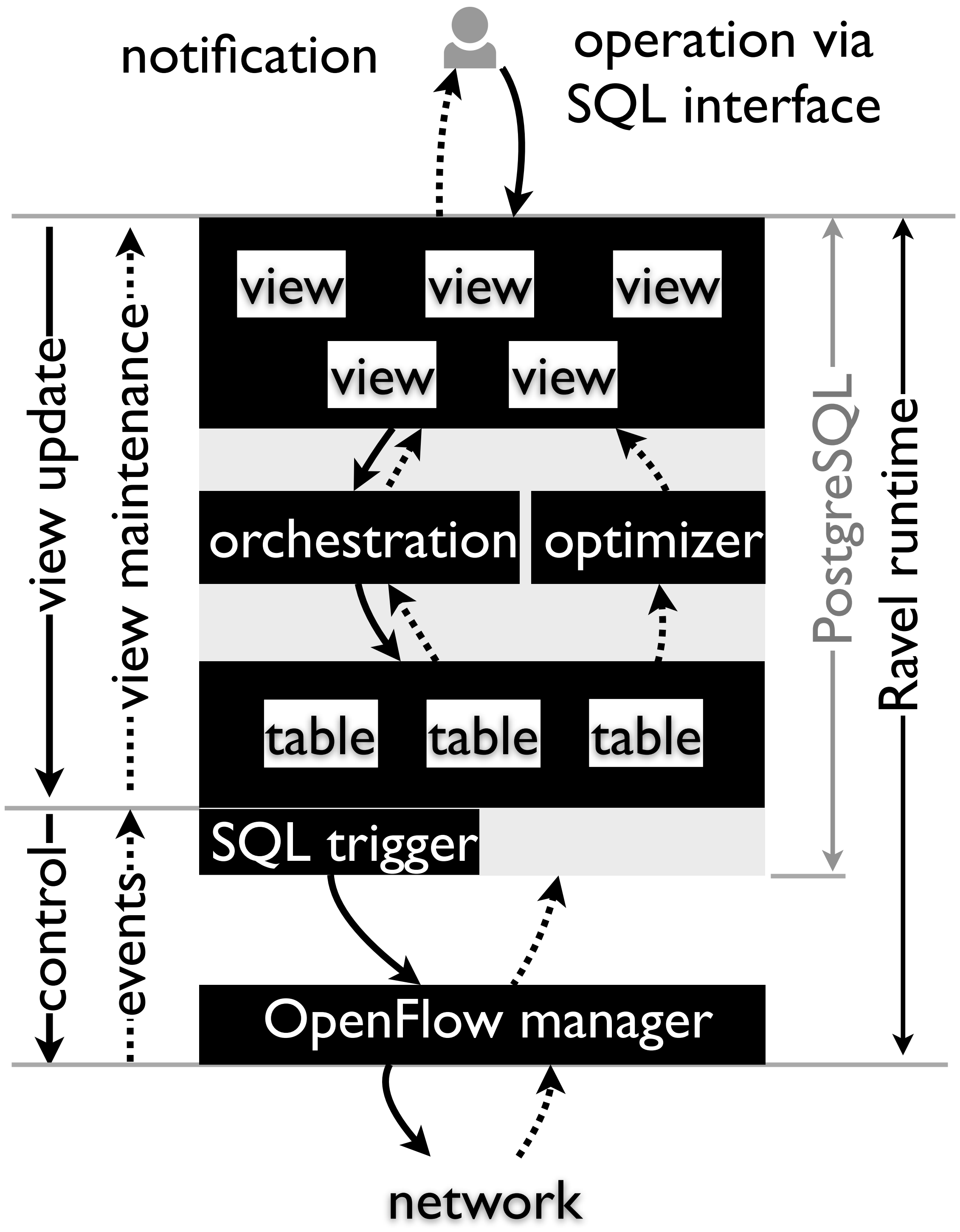


orchestration delay (ms) normalized per-rule for 3 scenarios:
access control and routing (acl+rt), load balancing and routing (lb+rt), access control, load balancing, and routing (acl+lb+rt)

orchestration also scales gracefully on fat-tree

- < 30ms for fat-tree with 5120 switches and 196608 links

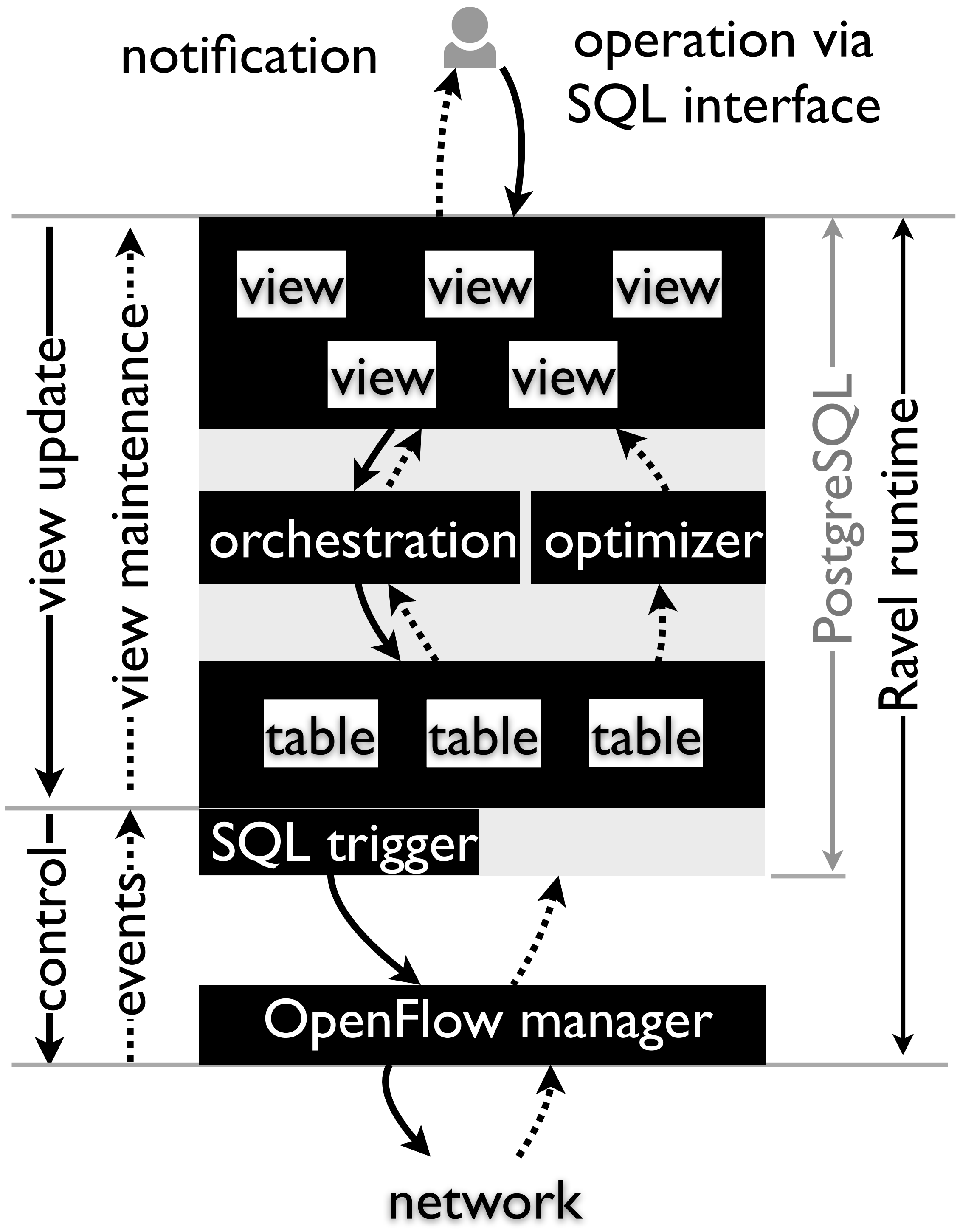
conclusion


notification
operation via
SQL interface
view
view
view
view
view
orchestration
optimizer
table
table
table
SQL trigger
OpenFlow manager
network
view update
view maintenance
control
events
PostgreSQL
Ravel runtime

this talk

- flexible abstraction via SQL:
  ad-hoc extensible, orchestratable
  promising performance

conclusion


notification
operation via SQL interface
view
view
view
view
view
orchestration
optimizer
table
table
table
SQL trigger
OpenFlow manager
network
view update
view maintenance
control
events
PostgreSQL
Ravel runtime

this talk

- flexible abstraction via SQL:
  ad-hoc extensible, orchestratable
  promising performance

looking forward

- application of database features
  - network-wide transaction
  - bootstrapping legacy networks
- enhancing database
  - better runtime: orchestration
  - better control decision: view analysis
- interpretability
  - integrate foreign applications, plug-n-play 3rd party solvers

demo

```
[ravel@ravelvm ravel]$
```

demo

```
[ravel@ravelvm ravel]$
```

Ravel au pupitre « Boléro »

playtime

download *Ravel*

ravel-net.org/download

start playing: tutorials, add your own app

ravel-net.org

explore more

github.com/ravel-net