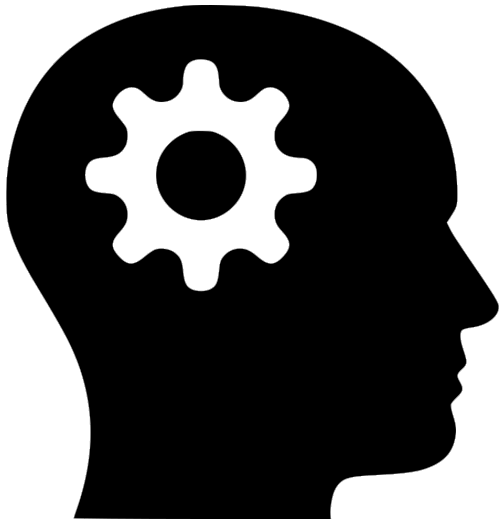
Untangling the Service Mesh

Dominik Tornow

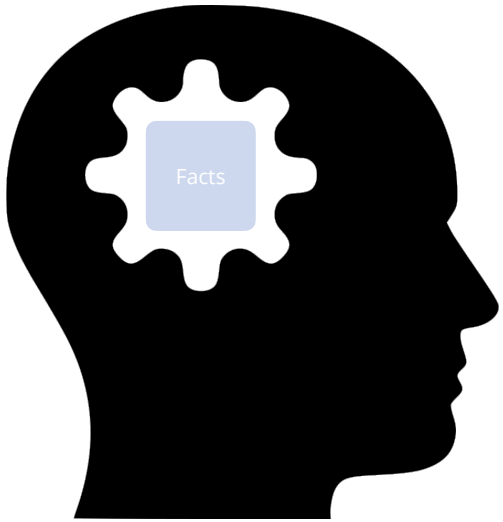
Dominik Tornow

Principal Engineer @ Cisco

dominik.tornow@gmail.com



Facts



Facts about the System

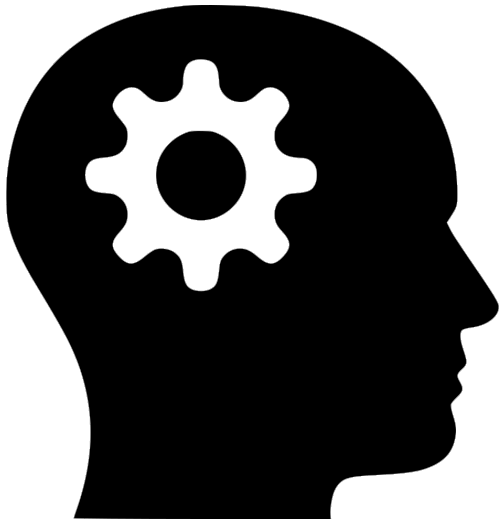
Traditional Definition

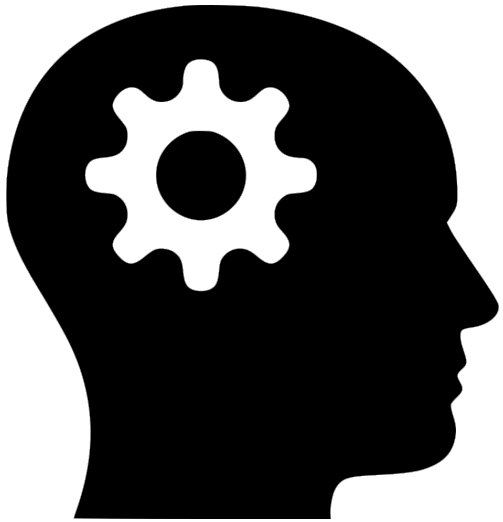
What is a Service Mesh?

A Service Mesh is a dedicated infrastructure layer for handling service to service communication

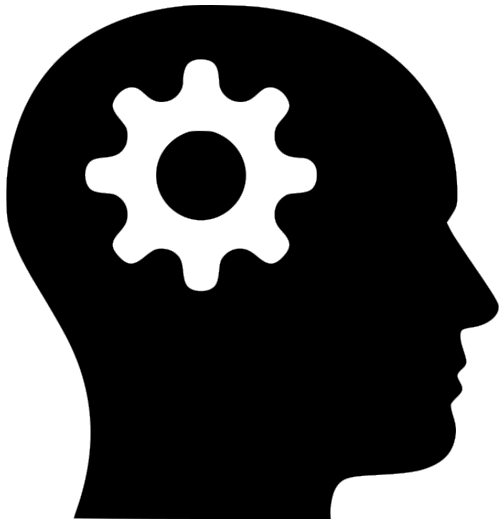
A Service Mesh is a set of user space proxies executing alongside your services and a set of management processes

Part I



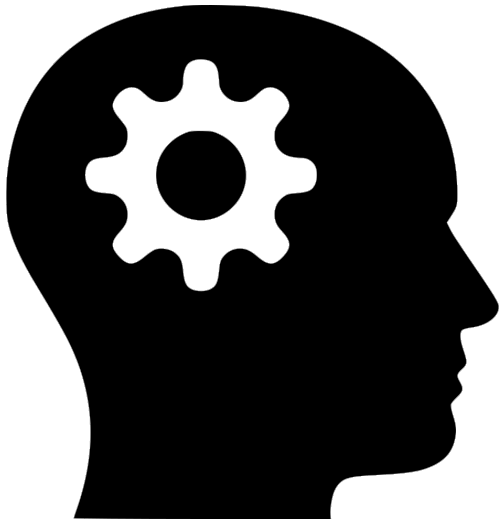


Networking



Networking

Services



Networking

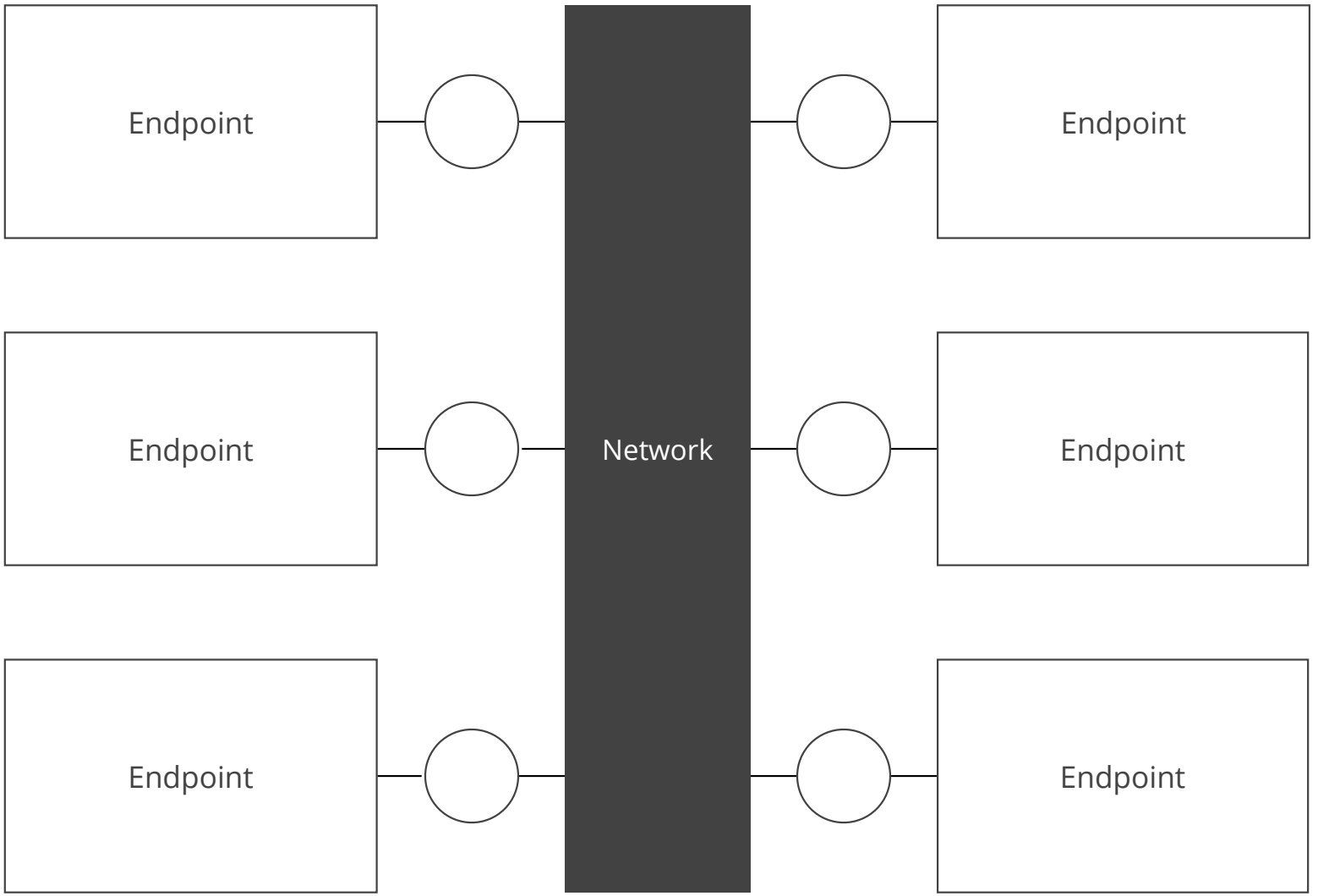
Services

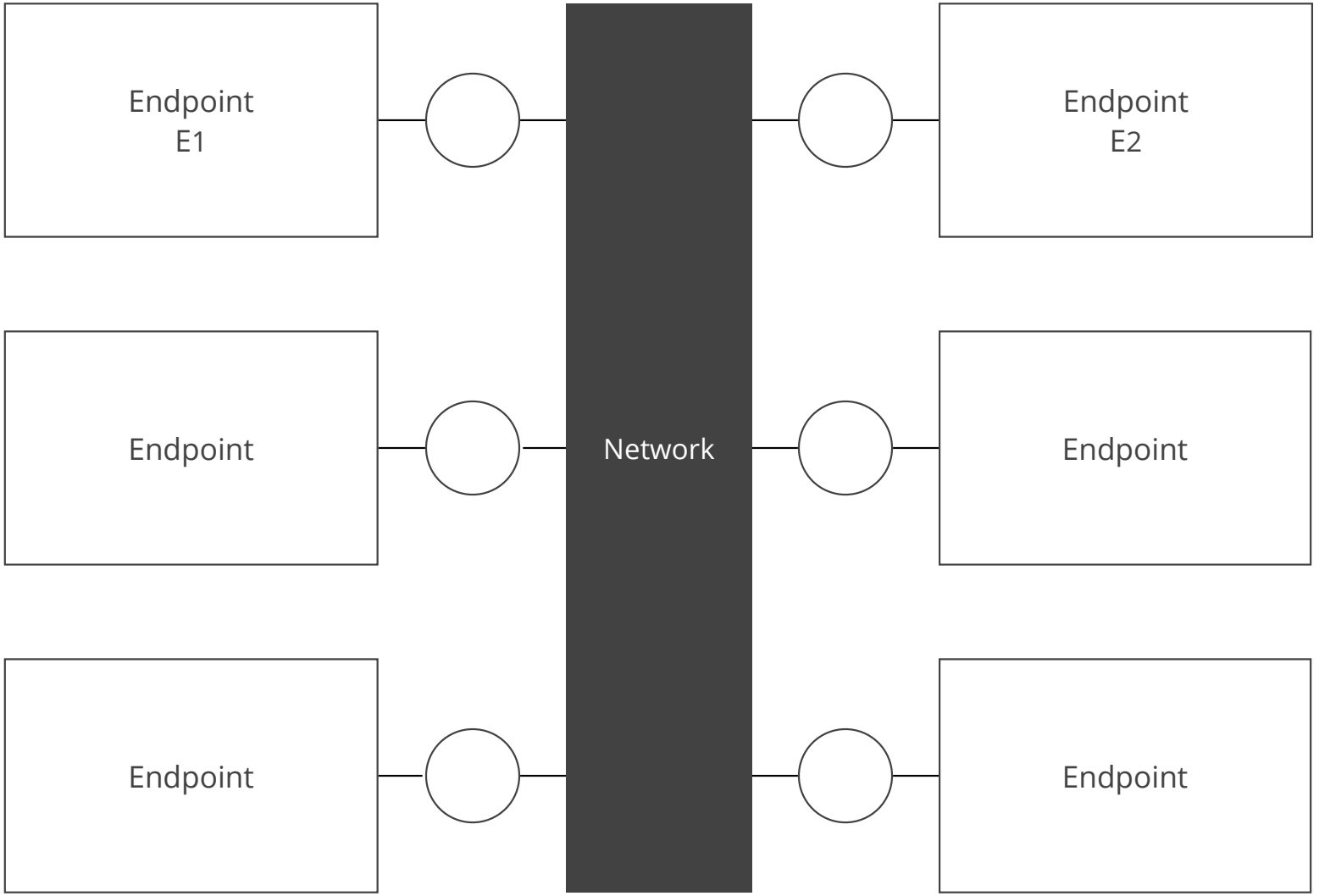
Service Meshes

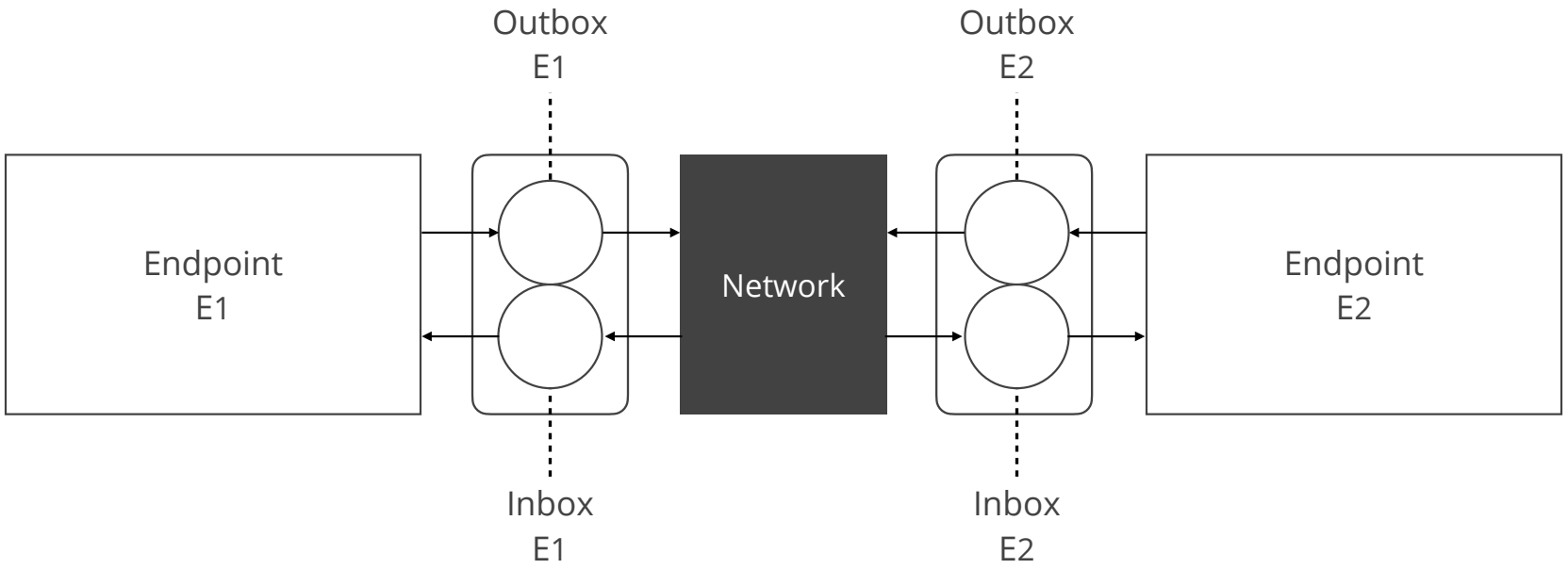
cisco

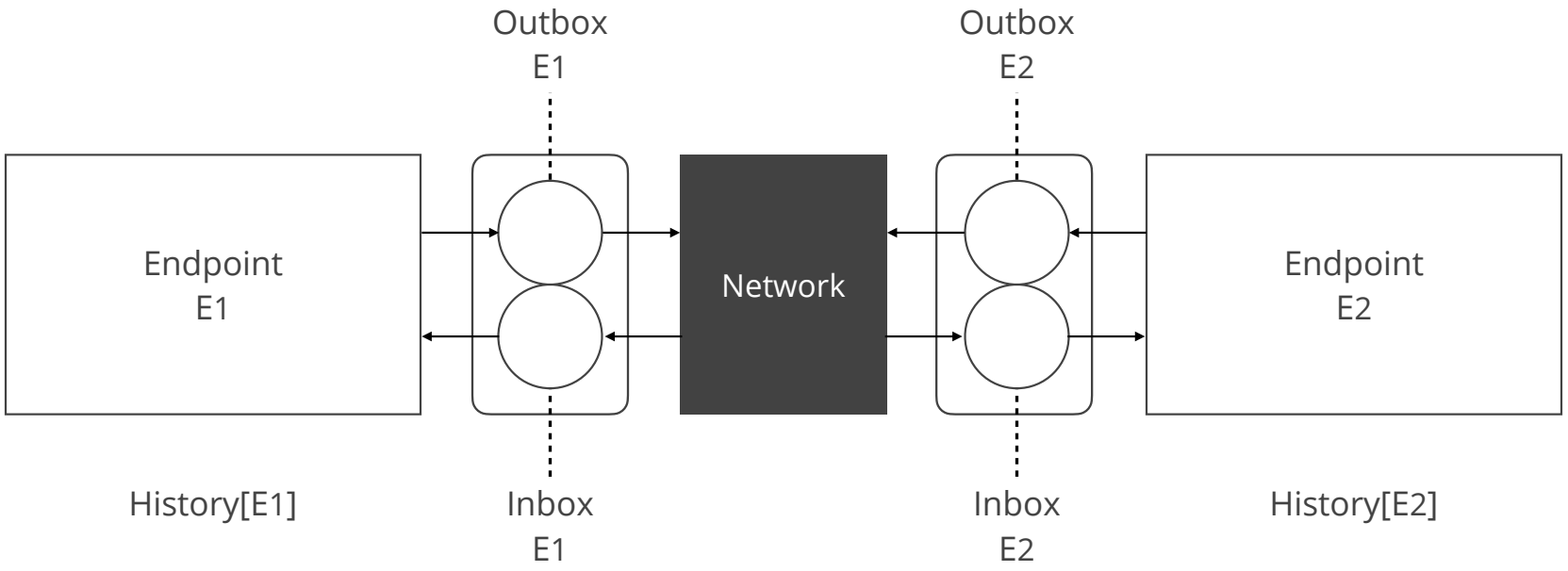
Networking Model

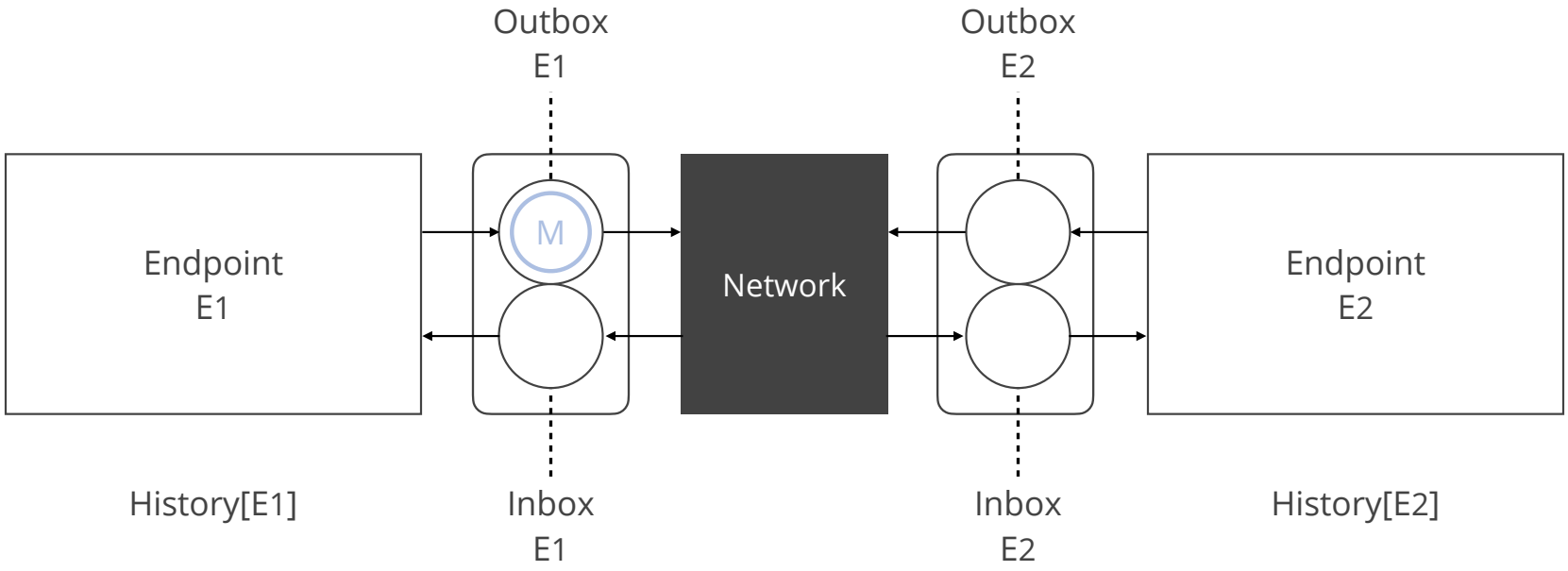
What is the Network?

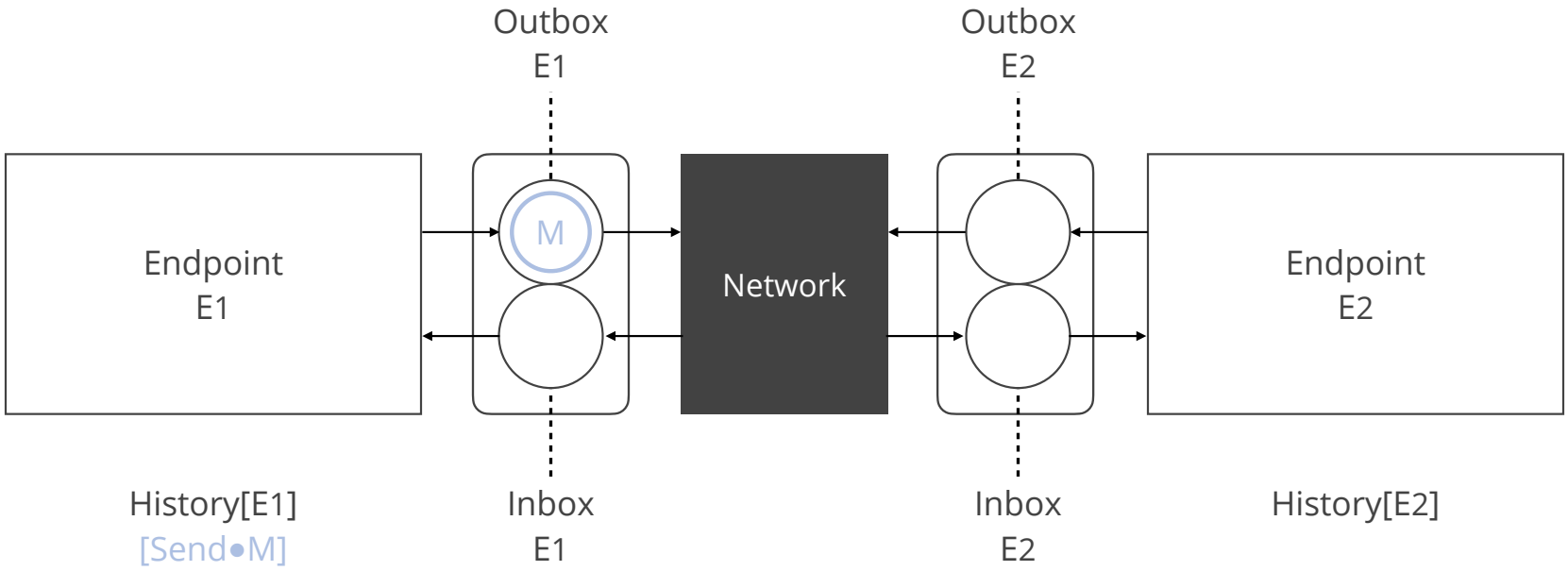

Endpoint
Endpoint
Endpoint
Network
Endpoint
Endpoint
Endpoint
Endpoint

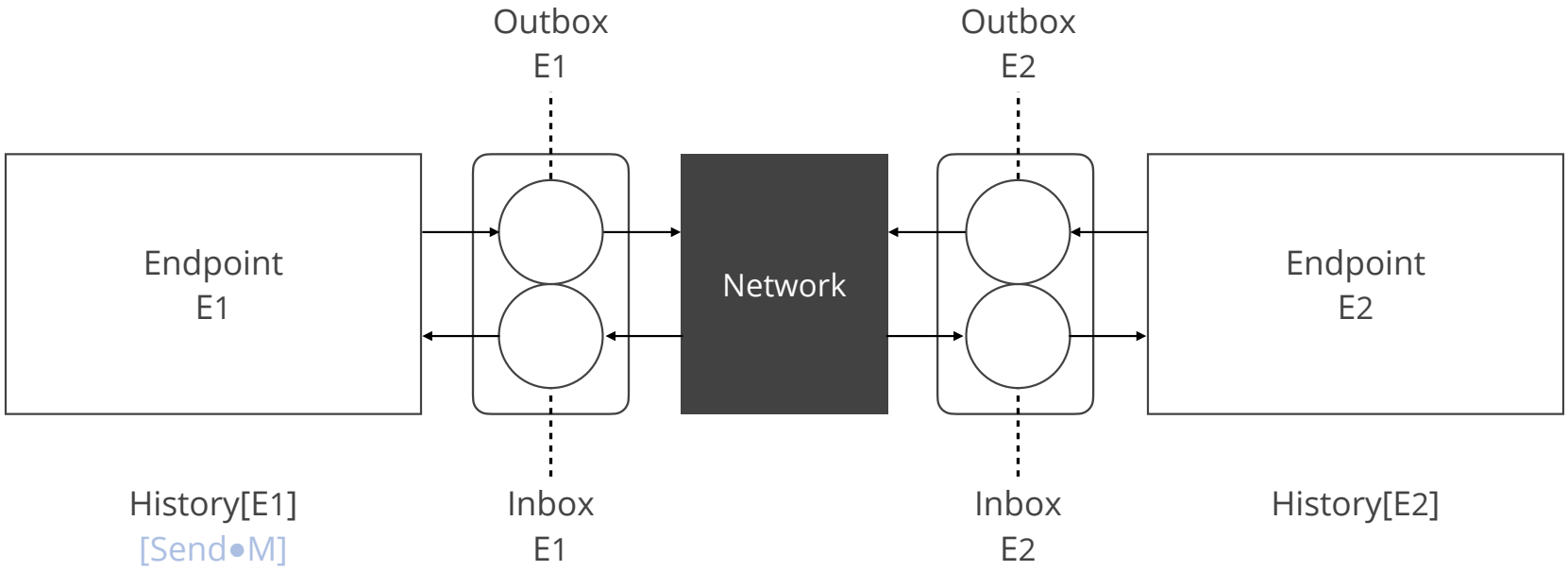

Endpoint
E1
Endpoint
E2
Endpoint
Network
Endpoint
Endpoint
Endpoint

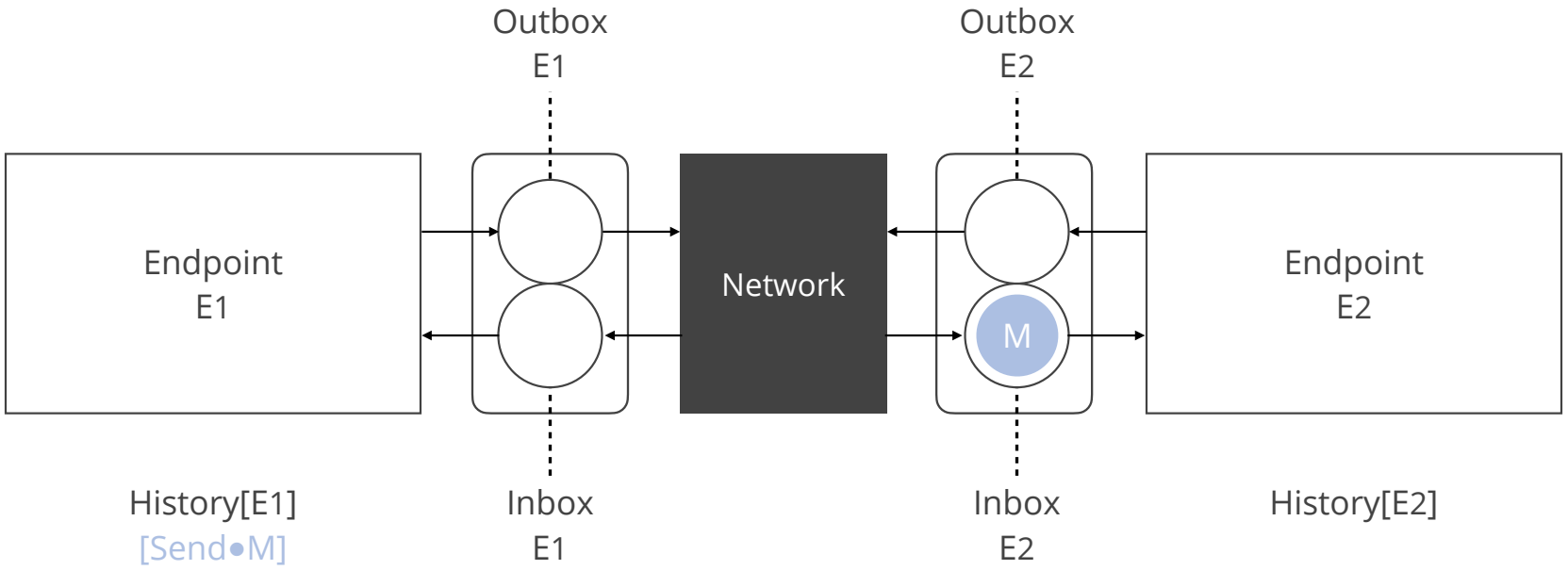

Outbox
E1
Outbox
E2
Endpoint
E1
Network
Endpoint
E2
Inbox
E1
Inbox
E2

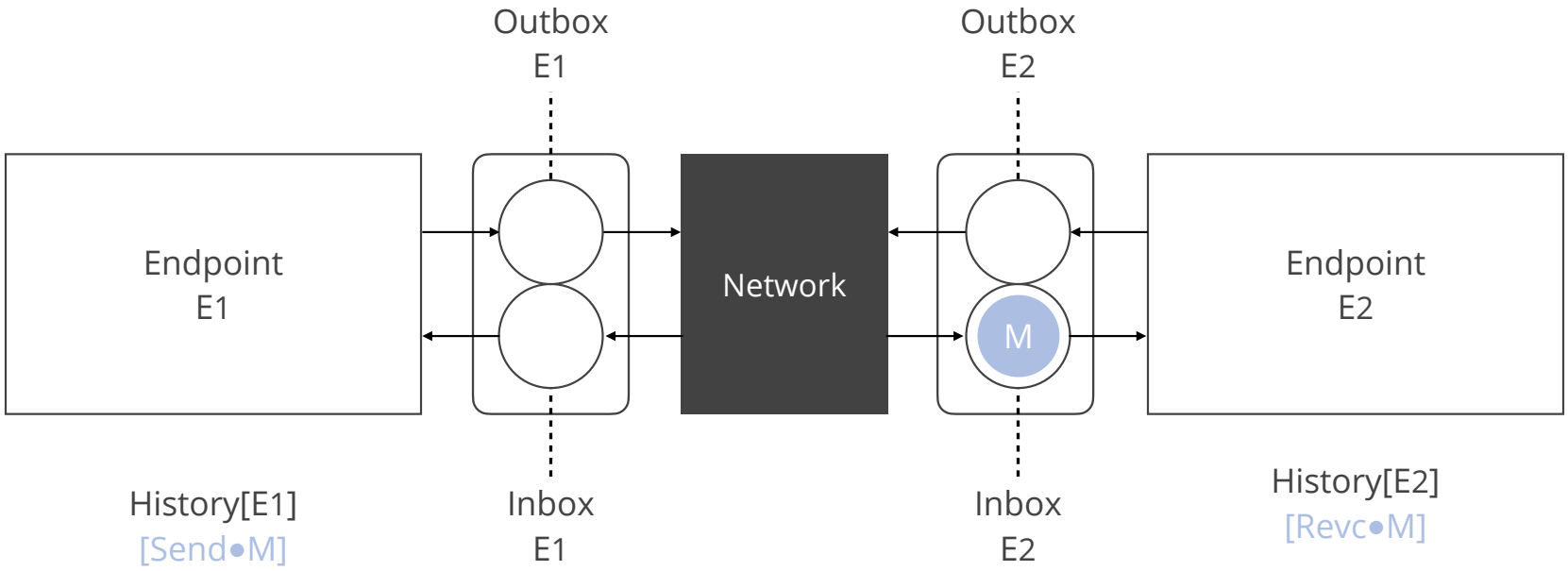

Outbox
E1
Outbox
E2
Endpoint
E1
Network
Endpoint
E2
History[E1]
Inbox
E1
Inbox
E2
History[E2]

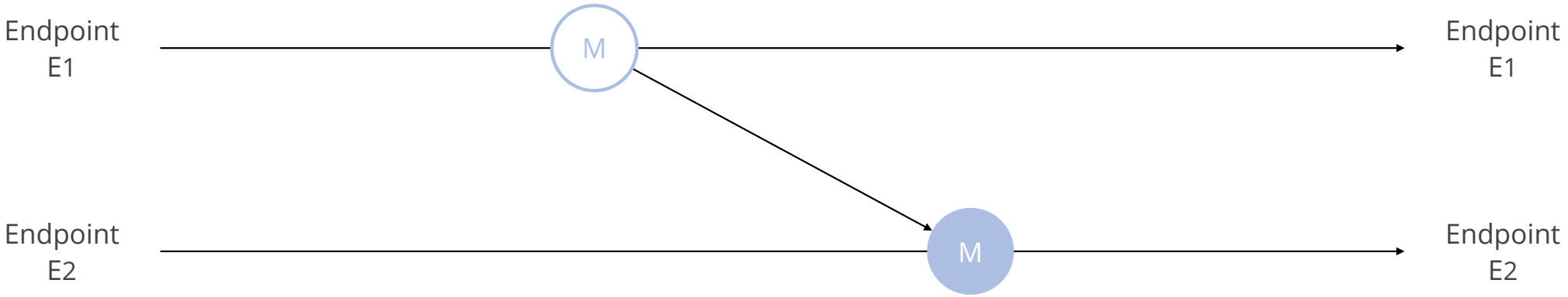

Outbox
E1
Outbox
E2
Endpoint
E1
M
Network
Endpoint
E2
History[E1]
Inbox
E1
Inbox
E2
History[E2]

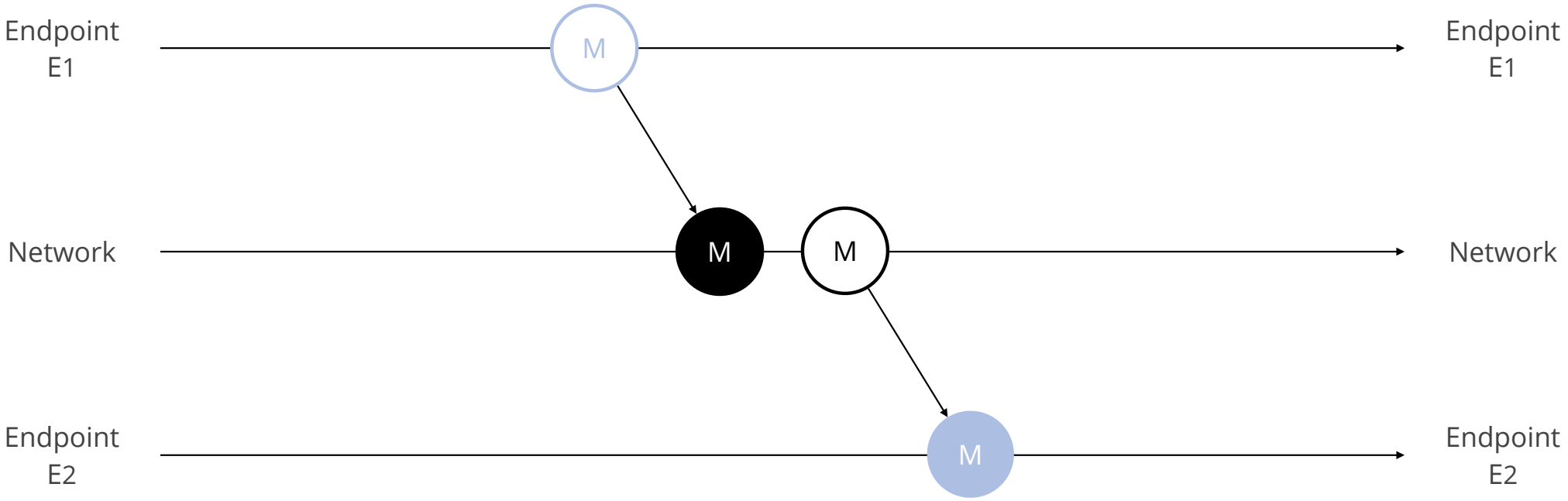

Outbox
E1
Outbox
E2
Endpoint
E1
M
Network
Endpoint
E2
History[E1]
[Send•M]
Inbox
E1
Inbox
E2
History[E2]

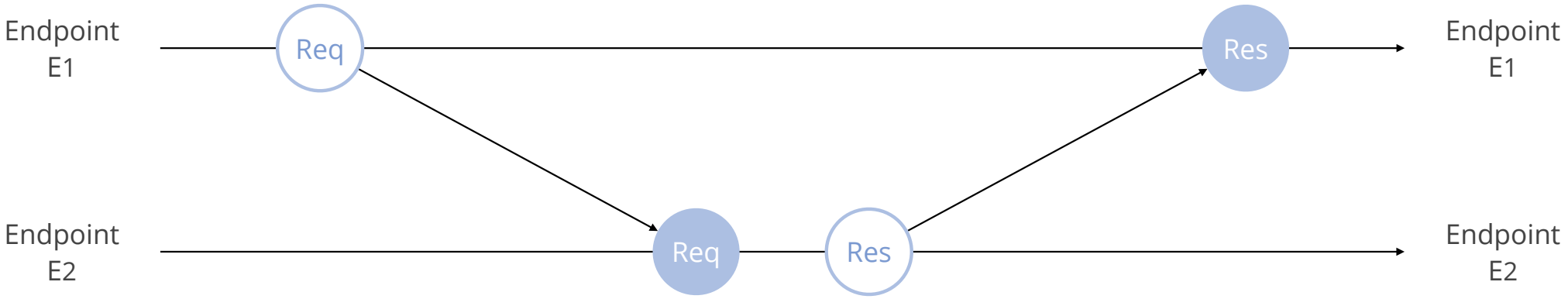

Outbox
E1
Outbox
E2
Endpoint
E1
Network
Endpoint
E2
History[E1]
[Send•M]
Inbox
E1
Inbox
E2
History[E2]

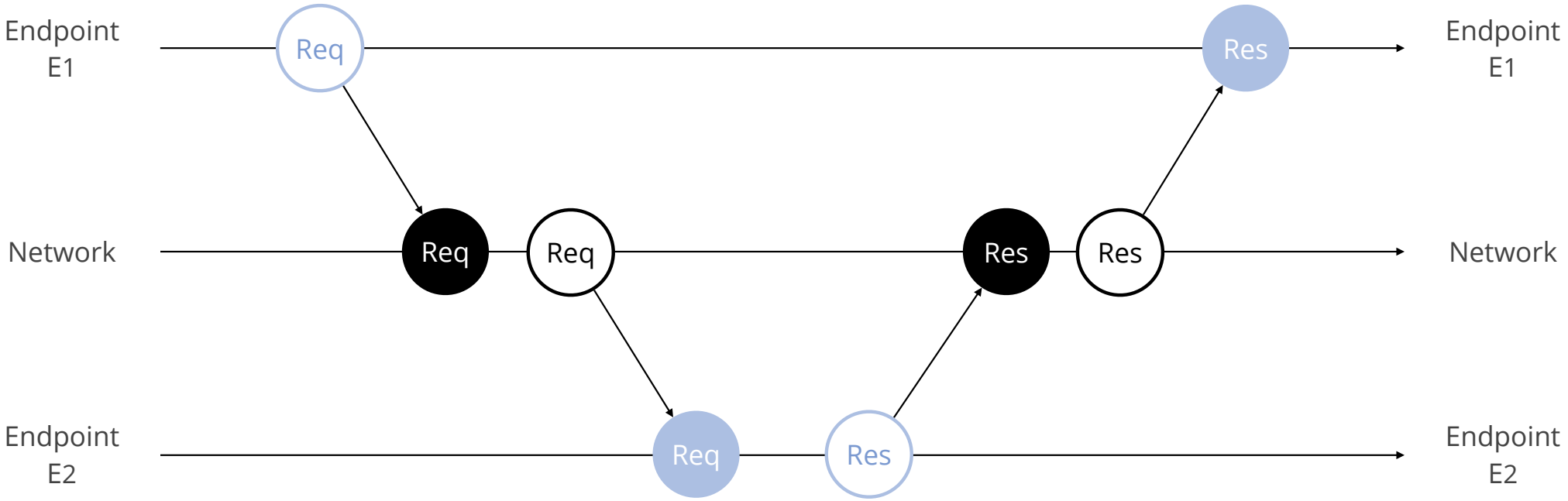

Outbox
E1
Outbox
E2
Endpoint
E1
Network
M
Endpoint
E2
History[E1]
[Send•M]
Inbox
E1
Inbox
E2
History[E2]


Outbox
E1
Outbox
E2
Endpoint
E1
Network
M
Endpoint
E2
History[E1]
[Send•M]
Inbox
E1
Inbox
E2
History[E2]
[Revc•M]

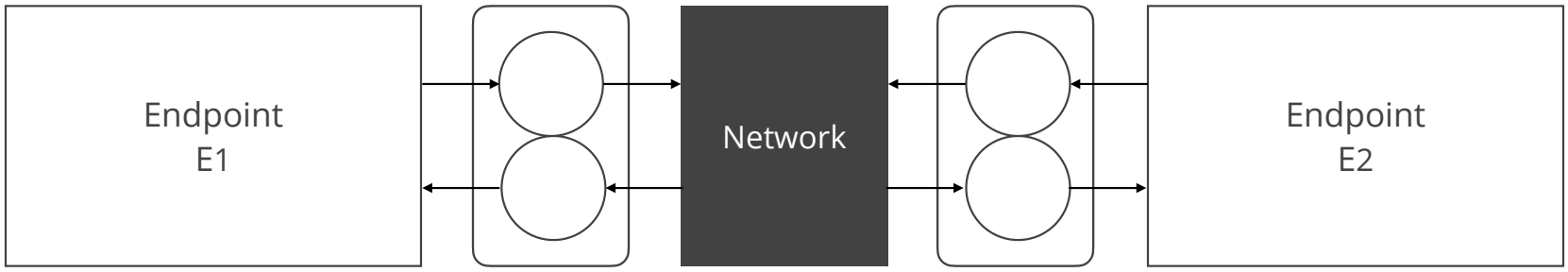

Endpoint
E1
M
Endpoint
E1
Endpoint
E2
M
Endpoint
E2

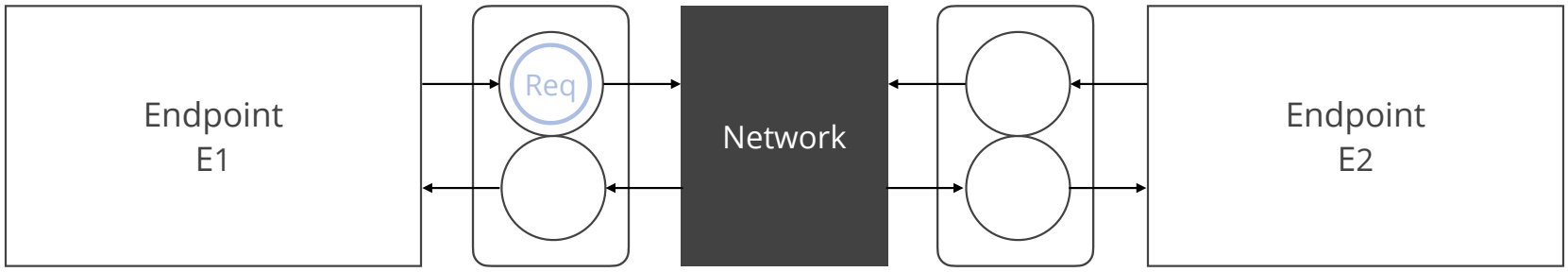

Endpoint
E1
M
Endpoint
E1
Network
M
M
Network
Endpoint
E2
M
Endpoint
E2

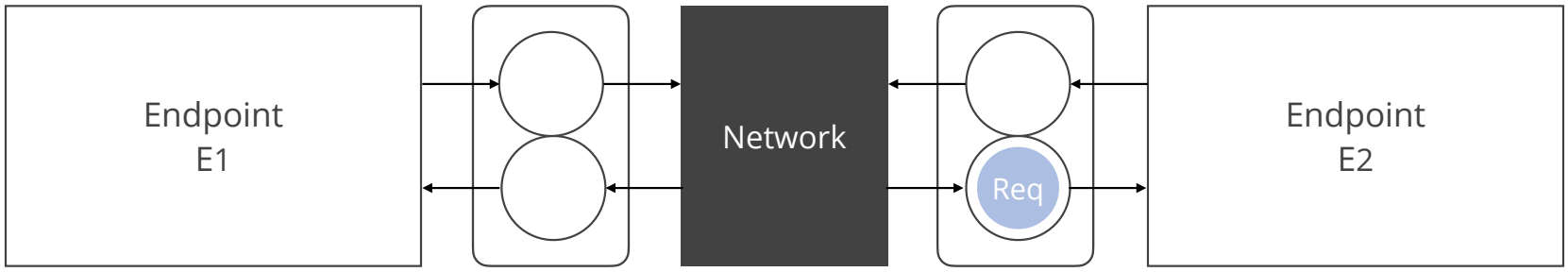

Endpoint
E1
Req
Res
Endpoint
E1
Endpoint
E2
Req
Res
Endpoint
E2

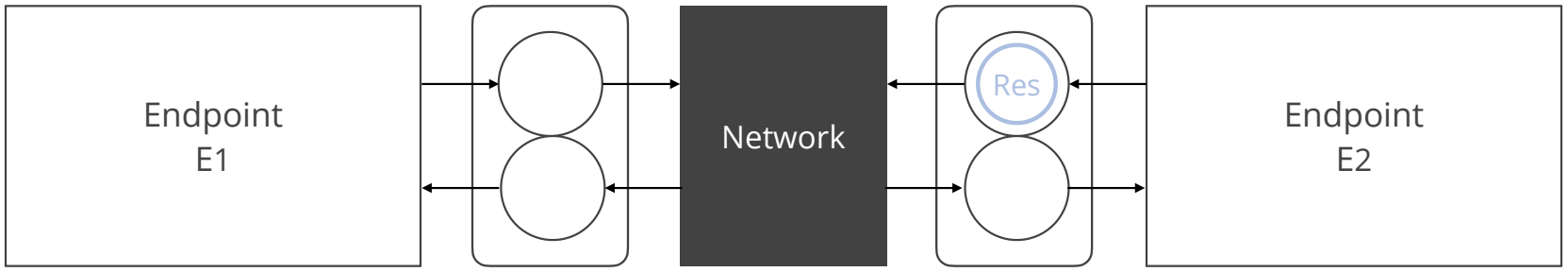

Endpoint
E1
Req
Res
Endpoint
E1
Network
Req
Req
Res
Res
Network
Endpoint
E2
Req
Res
Endpoint
E2

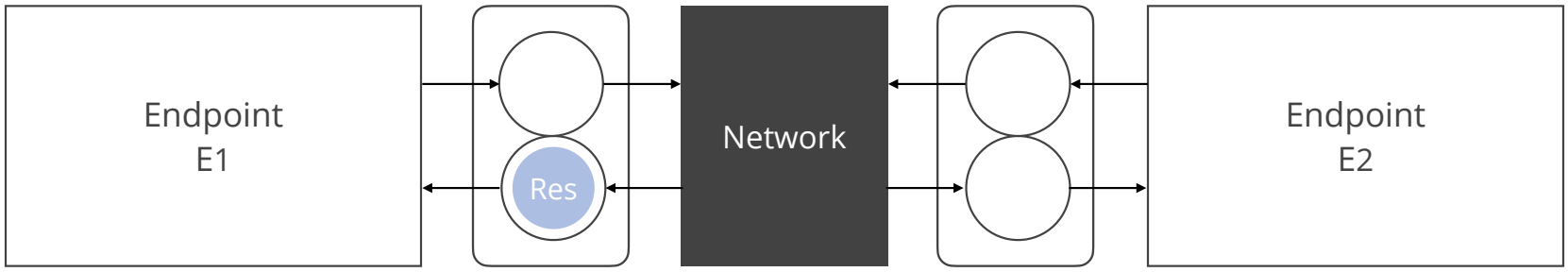
Global vs Local PoV

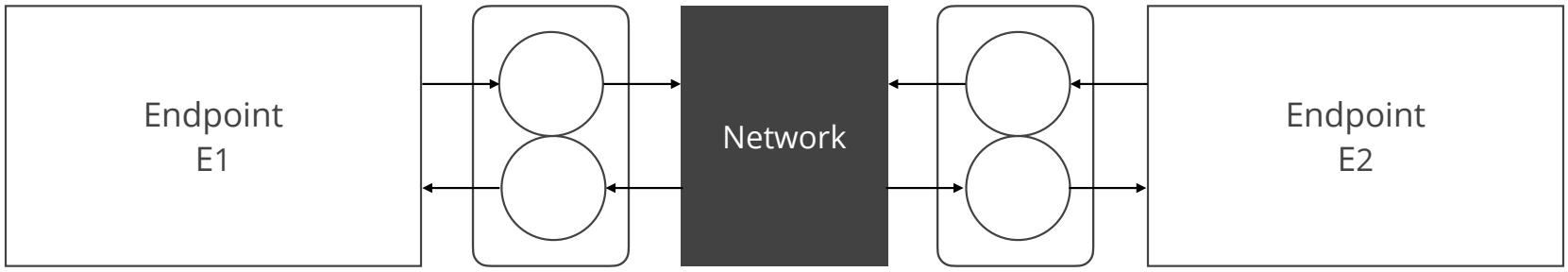

Endpoint
E1
Network
Endpoint
E2

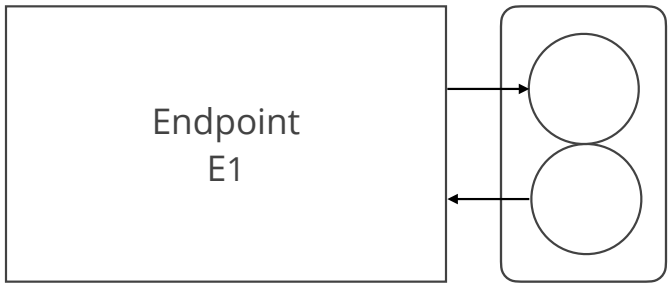

Endpoint
E1
Req
Network
Endpoint
E2


Endpoint
E1
Network
Req
Endpoint
E2

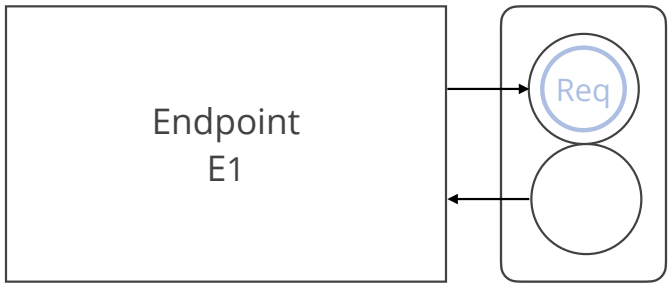

Endpoint
E1
Network
Res
Endpoint
E2


Endpoint
E1
Res
Network
Endpoint
E2

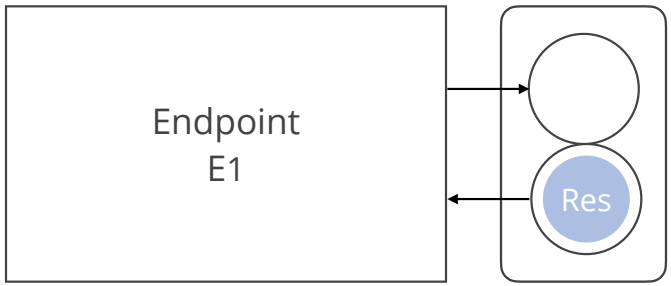

Endpoint
E1
Network
Endpoint
E2



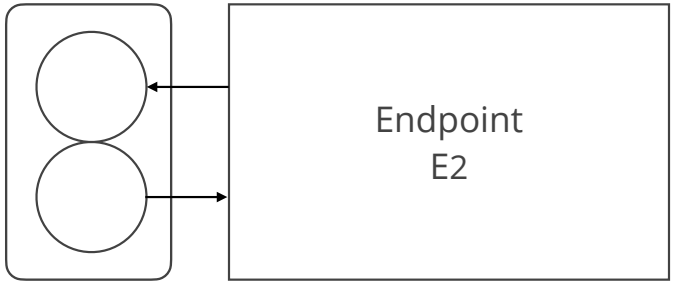
History[E1]



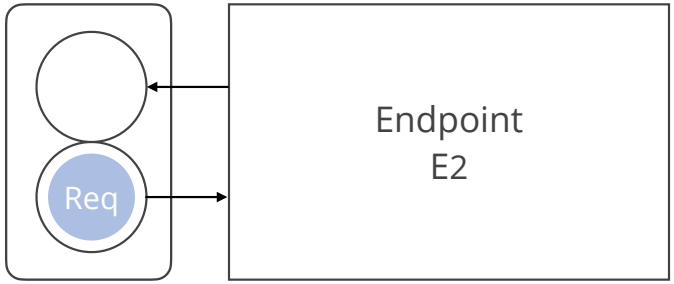
History[E1]
[Send•Req]



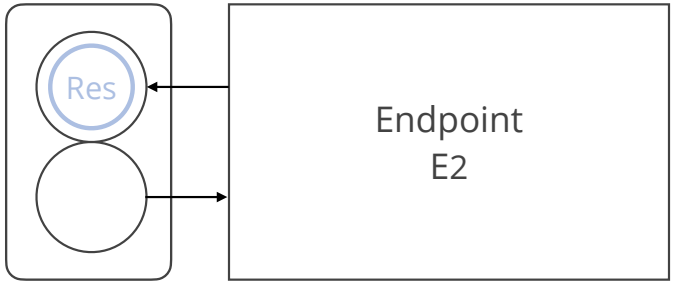
History[E1]
[Send•Req, Recv•Res]



History[E2]

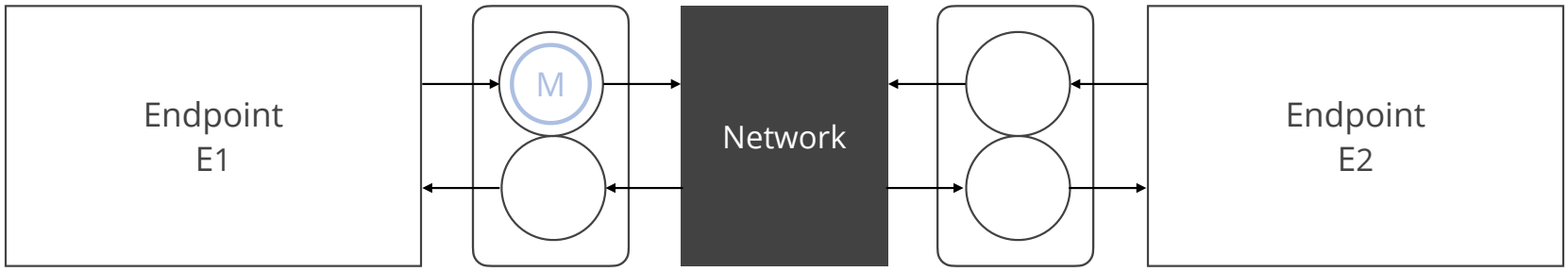


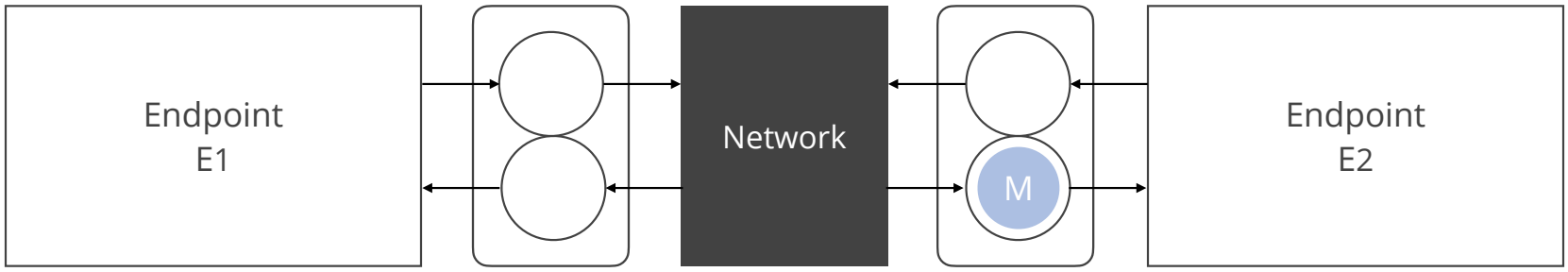
History[E2]
[Revc•Req]

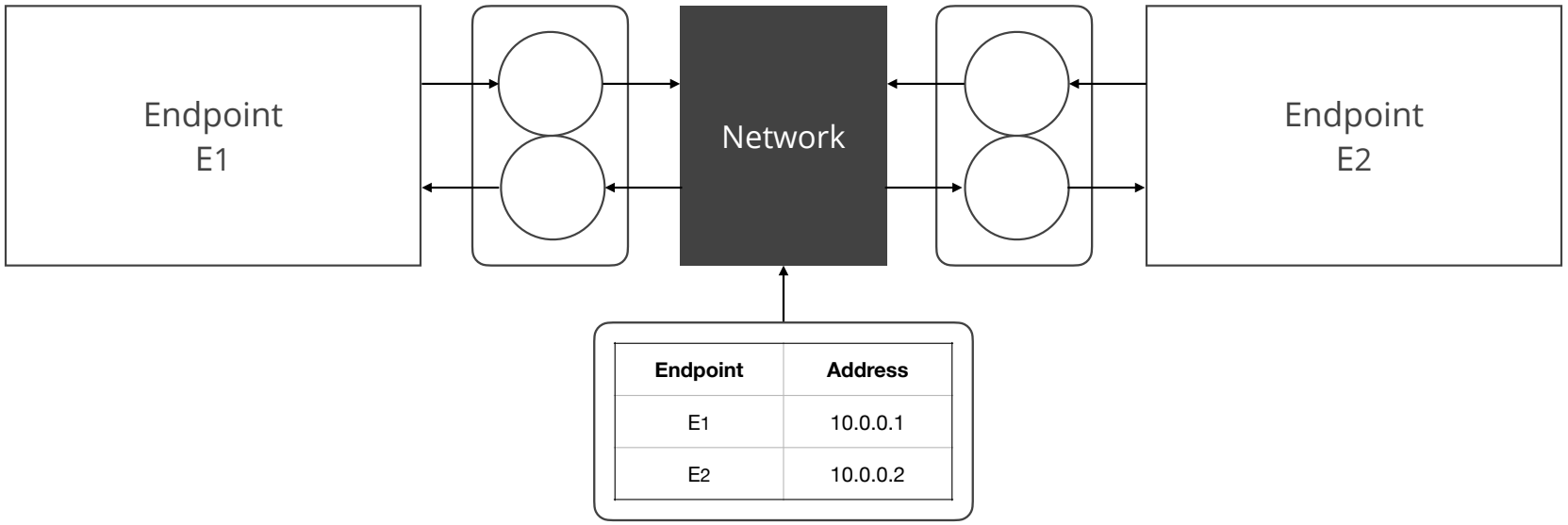


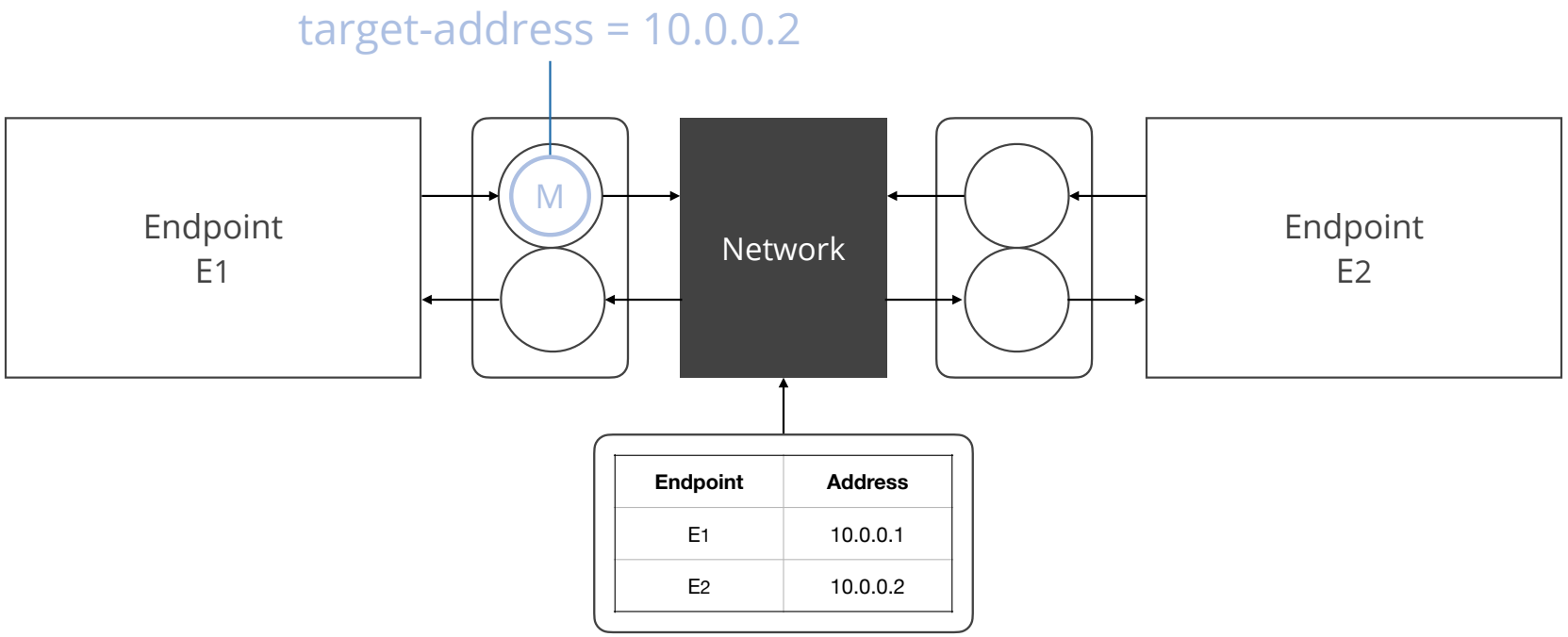
History[E2]
[Revc•Req, Send•Res]

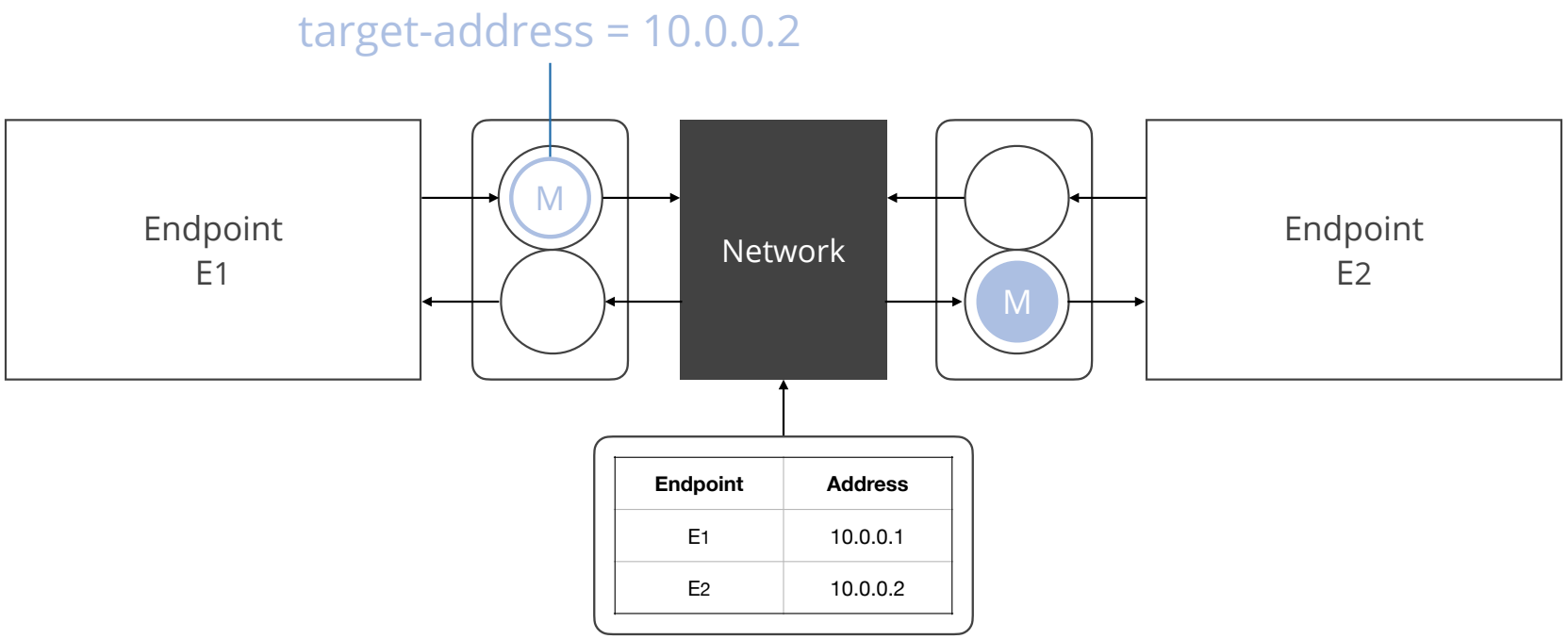
Addressing


Endpoint
E1
M
Network
Endpoint
E2


Endpoint
E1
Network
M
Endpoint
E2

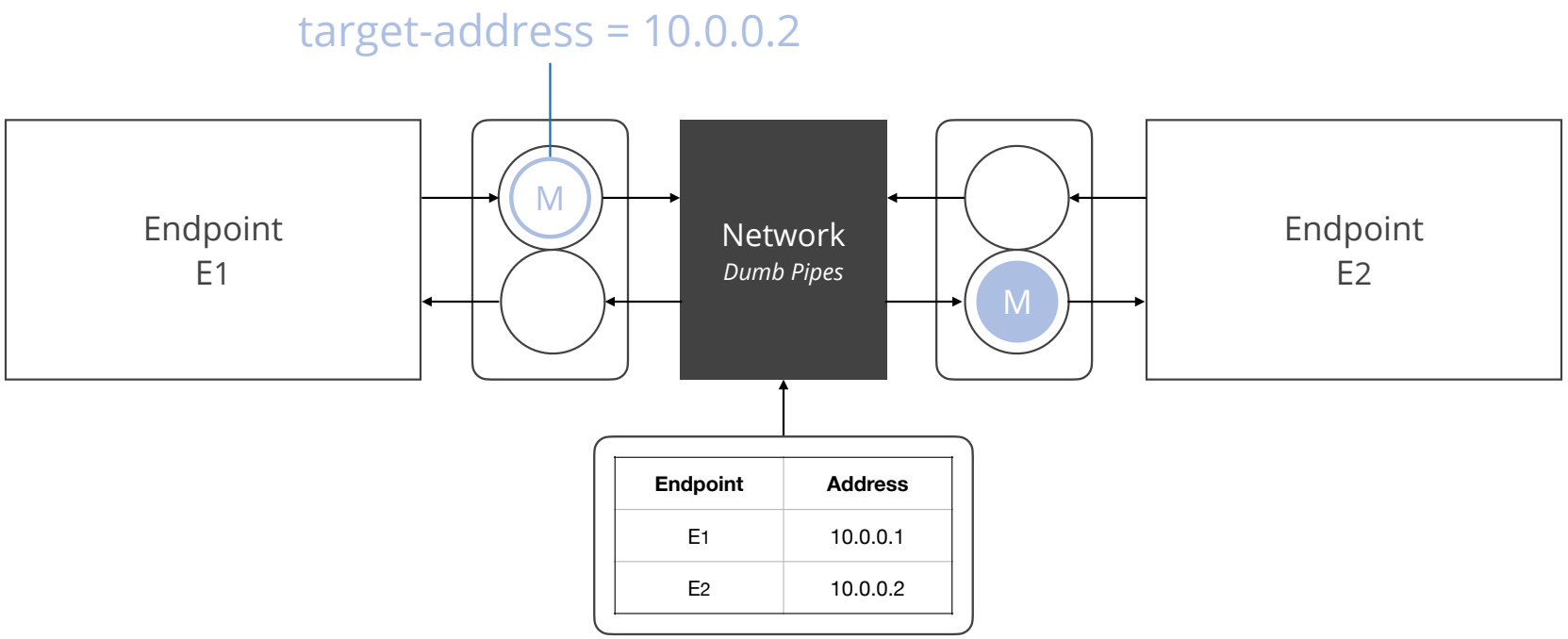

Endpoint
E1
Network
Endpoint
E2
Endpoint
Address
E1
10.0.0.1
E2
10.0.0.2


target-address = 10.0.0.2
Endpoint
E1
M
Network
Endpoint
E2
Endpoint
Address
E1
10.0.0.1
E2
10.0.0.2



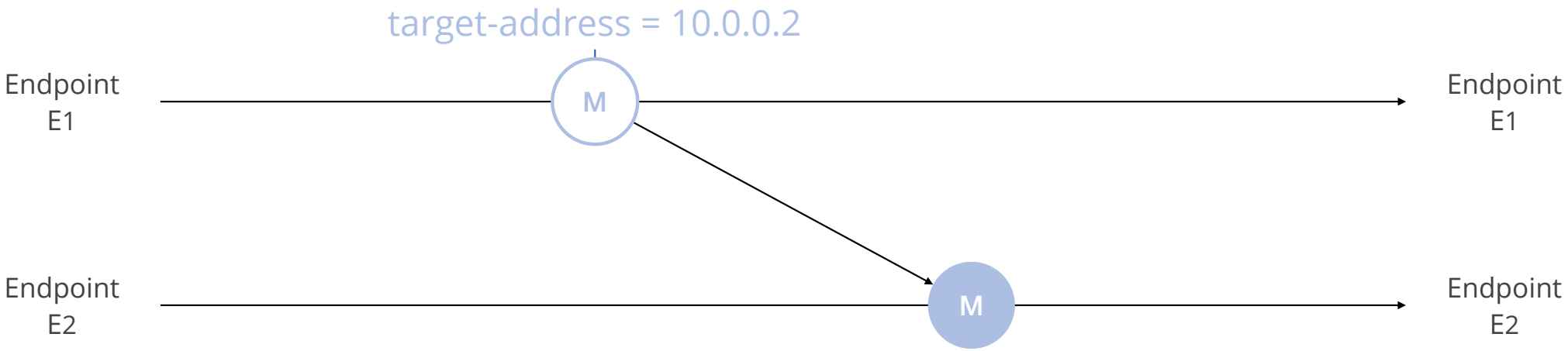
$$\text{Send} \bullet M \in \text{History}[E1] \wedge \text{Recv} \bullet M \in \text{History}[E2]$$

$$\Longrightarrow \text{target-address}[M] = \text{address}[E2]$$



$$\text{Send} \bullet M \in \text{History}[E1] \wedge \text{Recv} \bullet M \in \text{History}[E2]$$

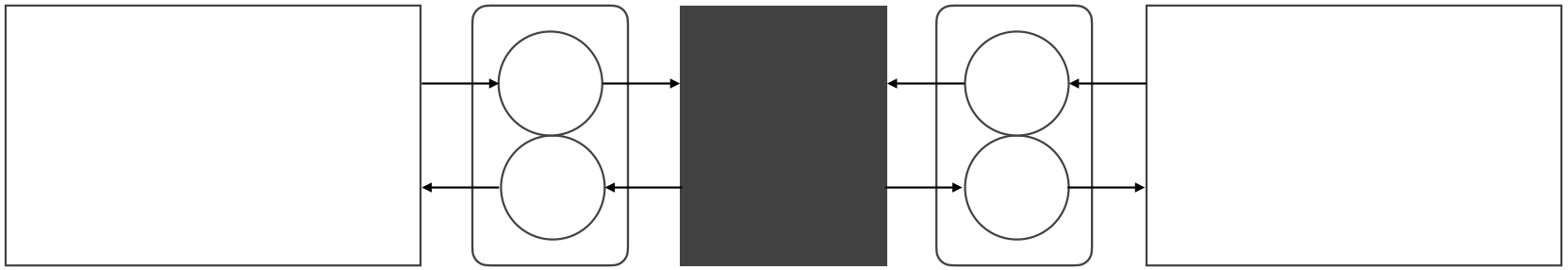
$$\Longrightarrow \text{target-address}[M] = \text{address}[E2]$$

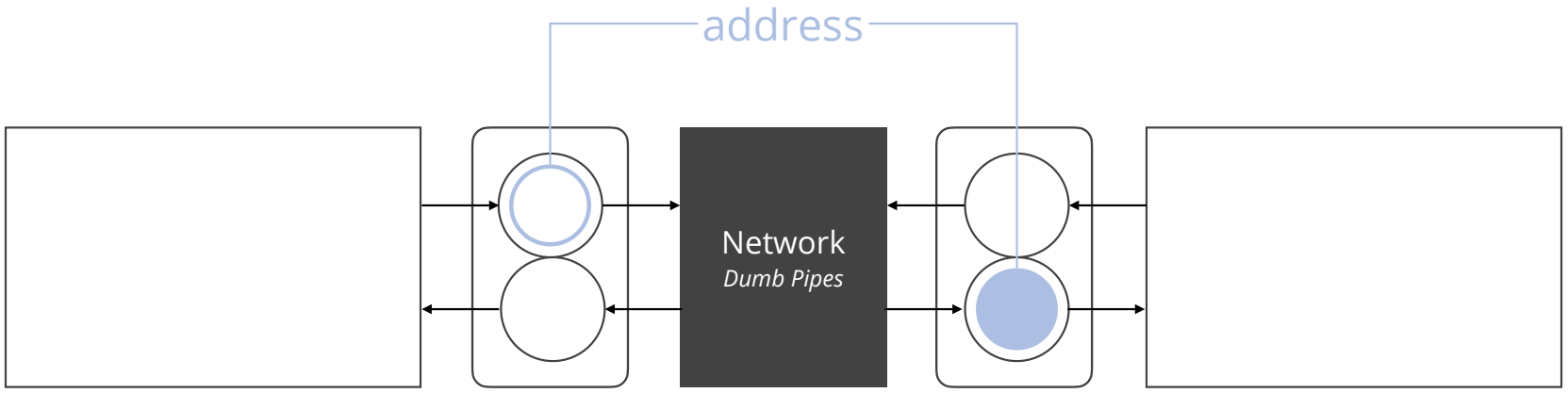


$$\text{Send} \bullet M \in \text{History}[E1] \wedge \text{Recv} \bullet M \in \text{History}[E2]$$
$$\Longrightarrow \text{target-address}[M] = \text{address}[E2]$$

Networking Model

Summary



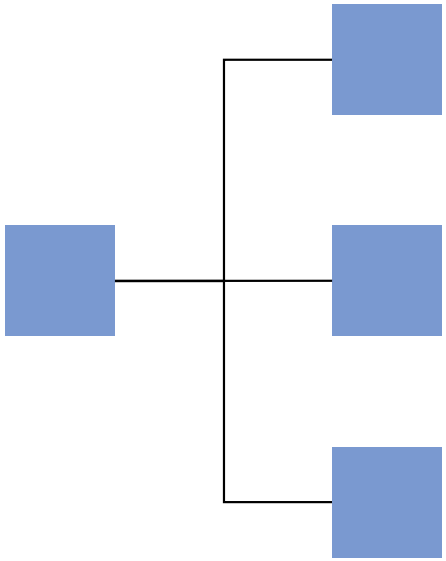

address
Network
Dumb Pipes

Service Model

What is a Service

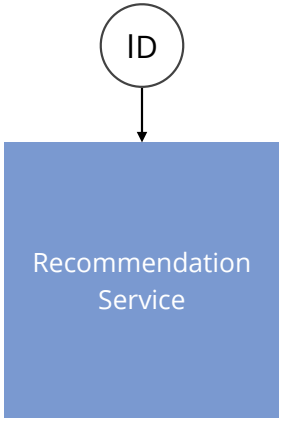
Service Model

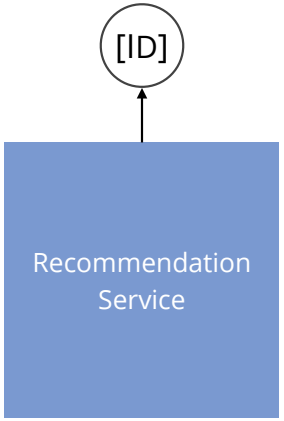
Intuition

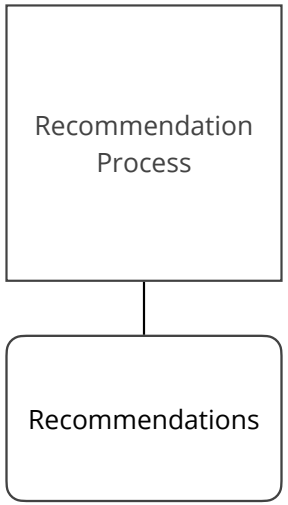


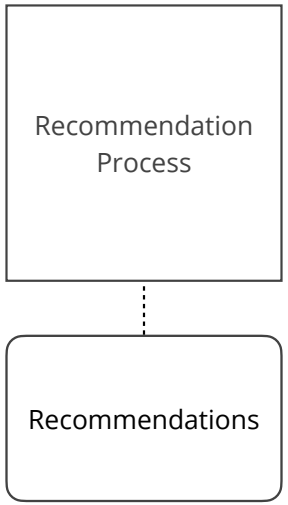
Recommendation Service
`get: ID ⟶ [ID]`

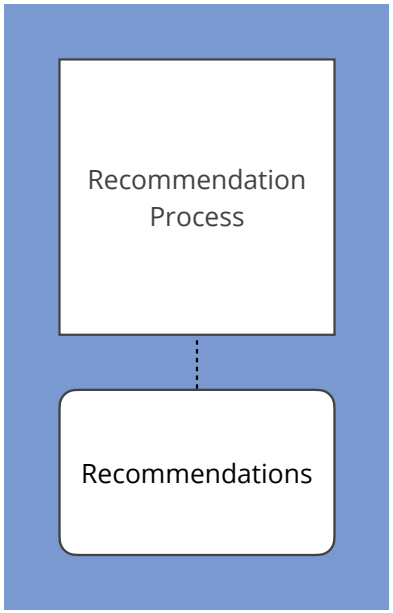

Recommendation
Service

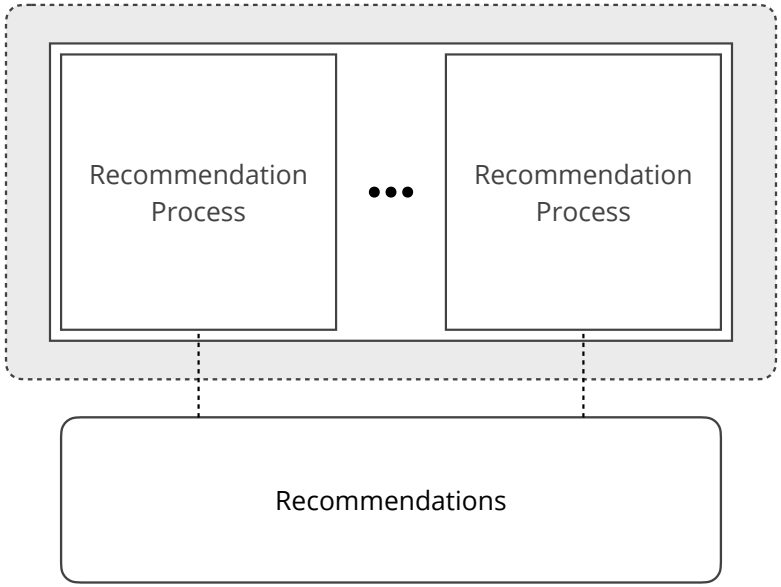

ID
Recommendation Service

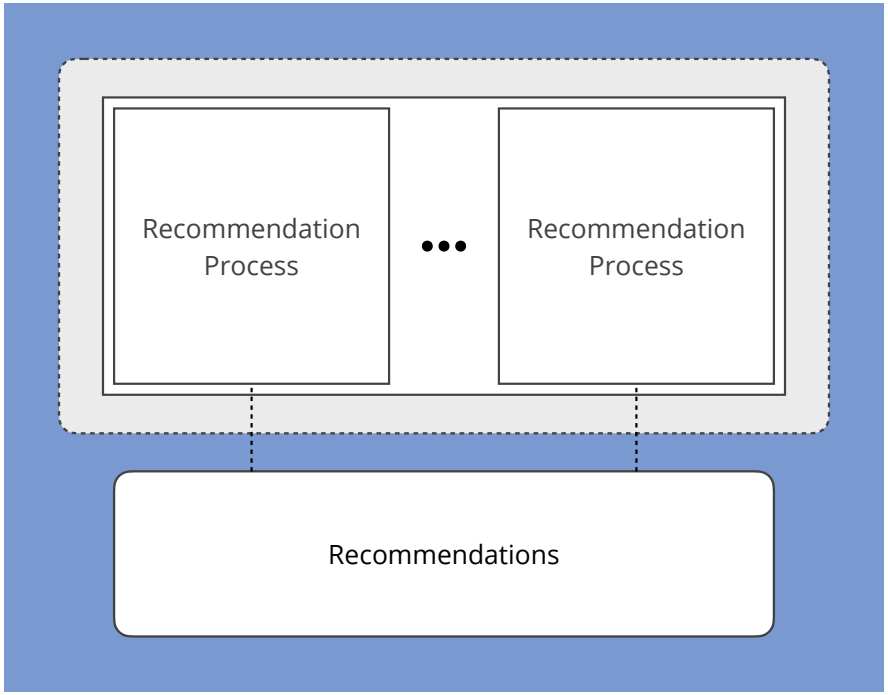

[ID]
Recommendation Service

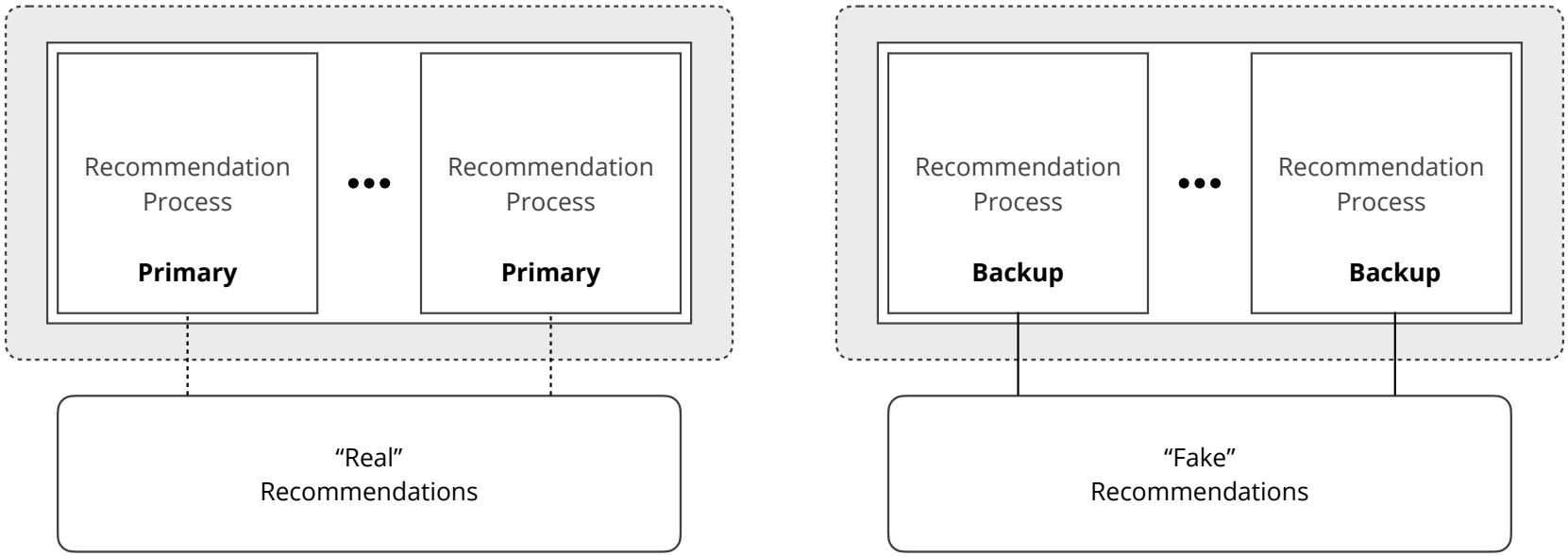

Recommendation Process
Recommendations

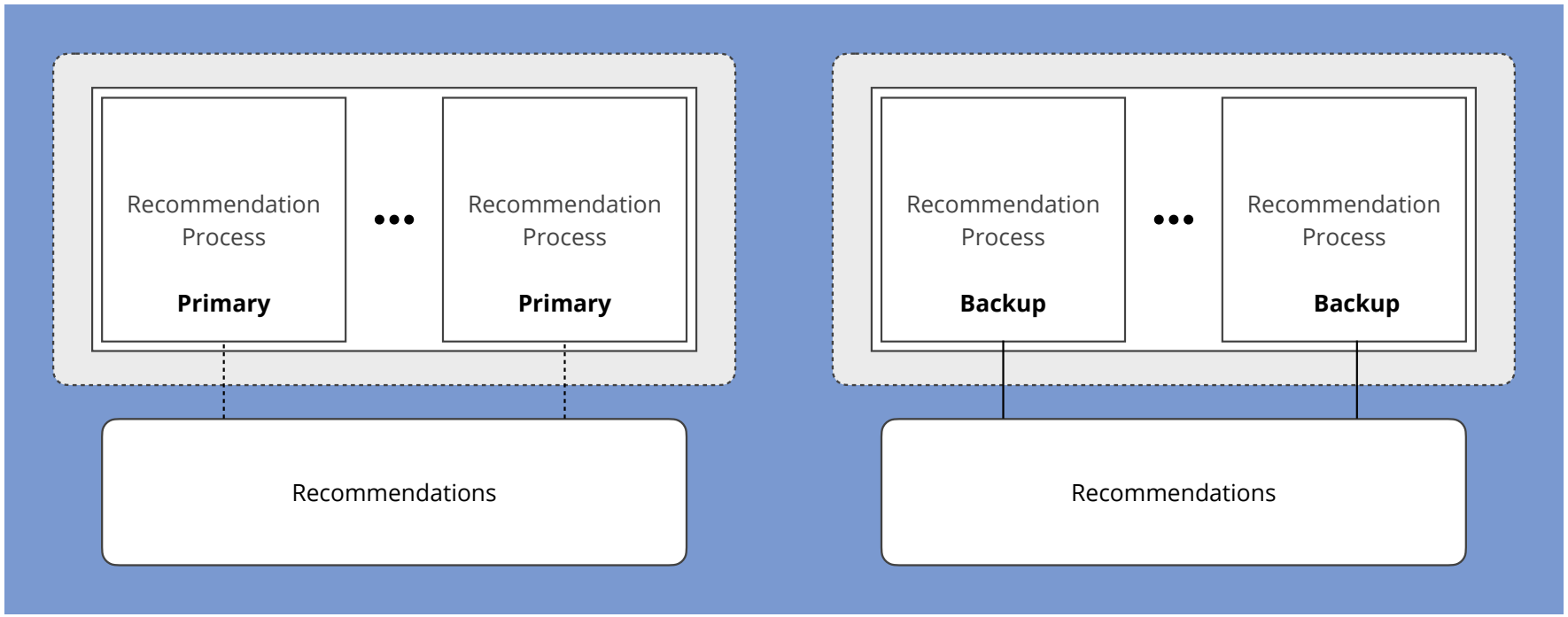

Recommendation Process
Recommendations

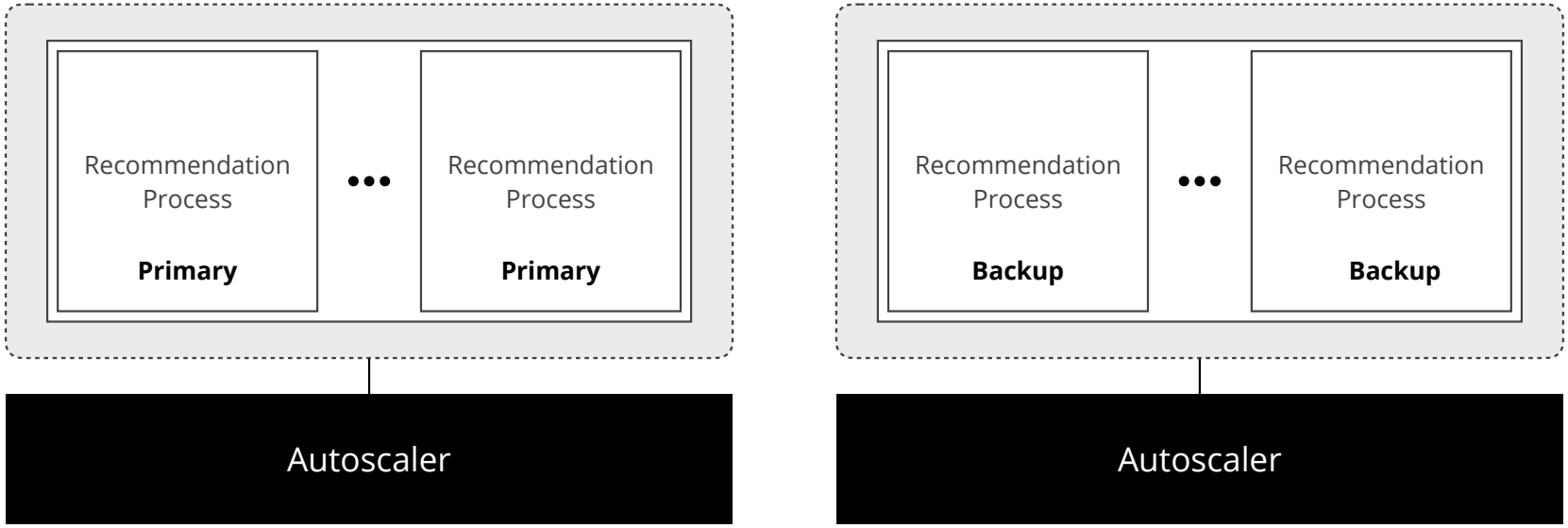

Recommendation Process
Recommendations

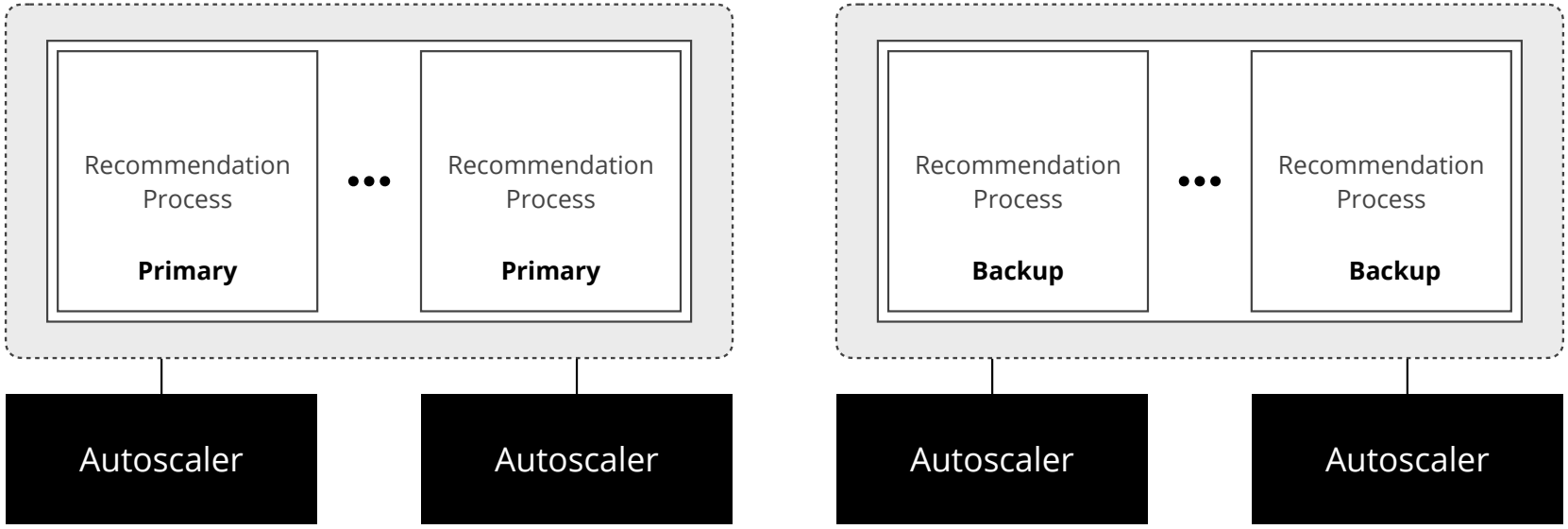

Recommendation Process
Recommendation Process
Recommendations

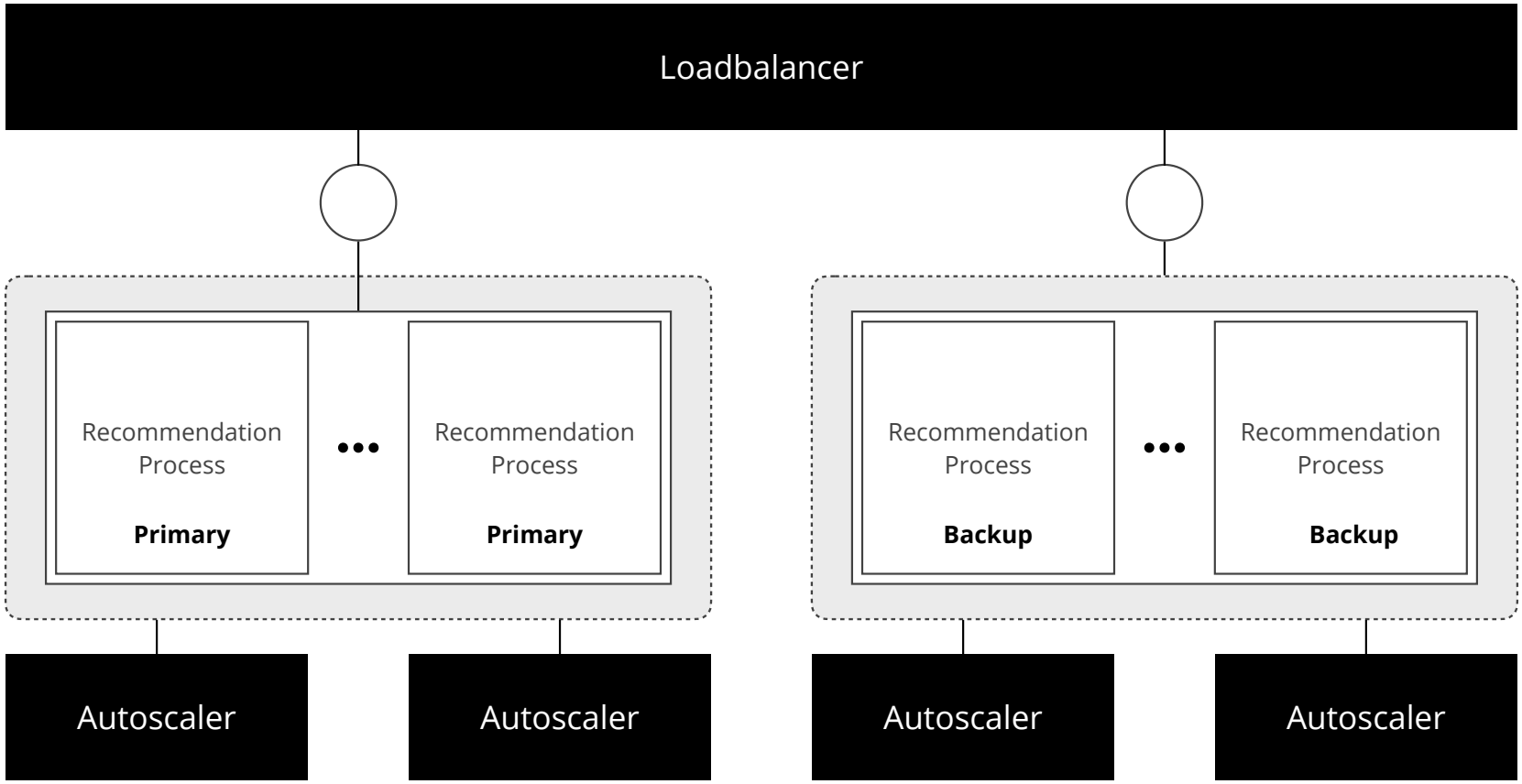

Recommendation Process
Recommendation Process
Recommendations

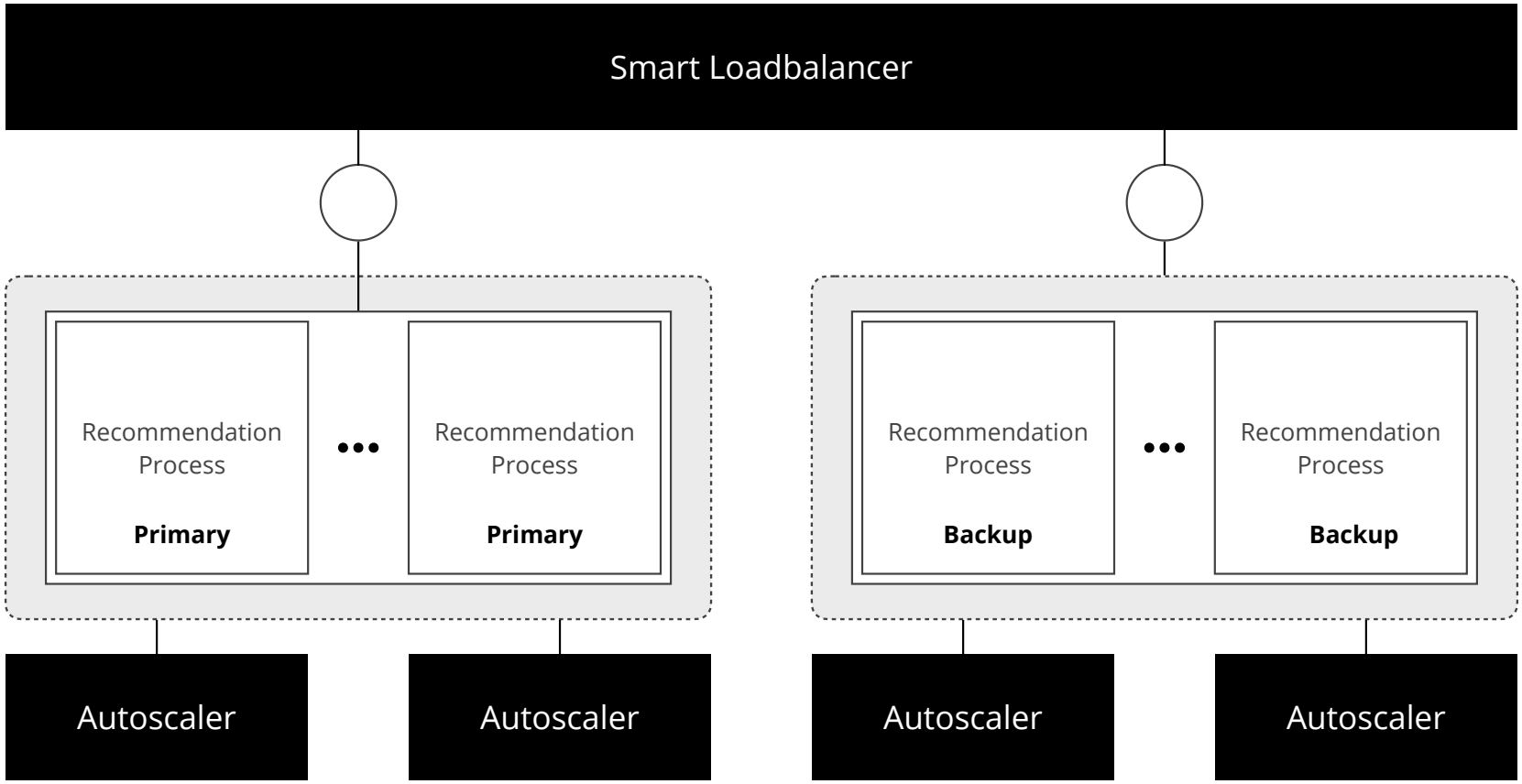

Recommendation Process
Primary
Recommendation Process
Primary
Recommendation Process
Backup
Recommendation Process
Backup
"Real" Recommendations
"Fake" Recommendations

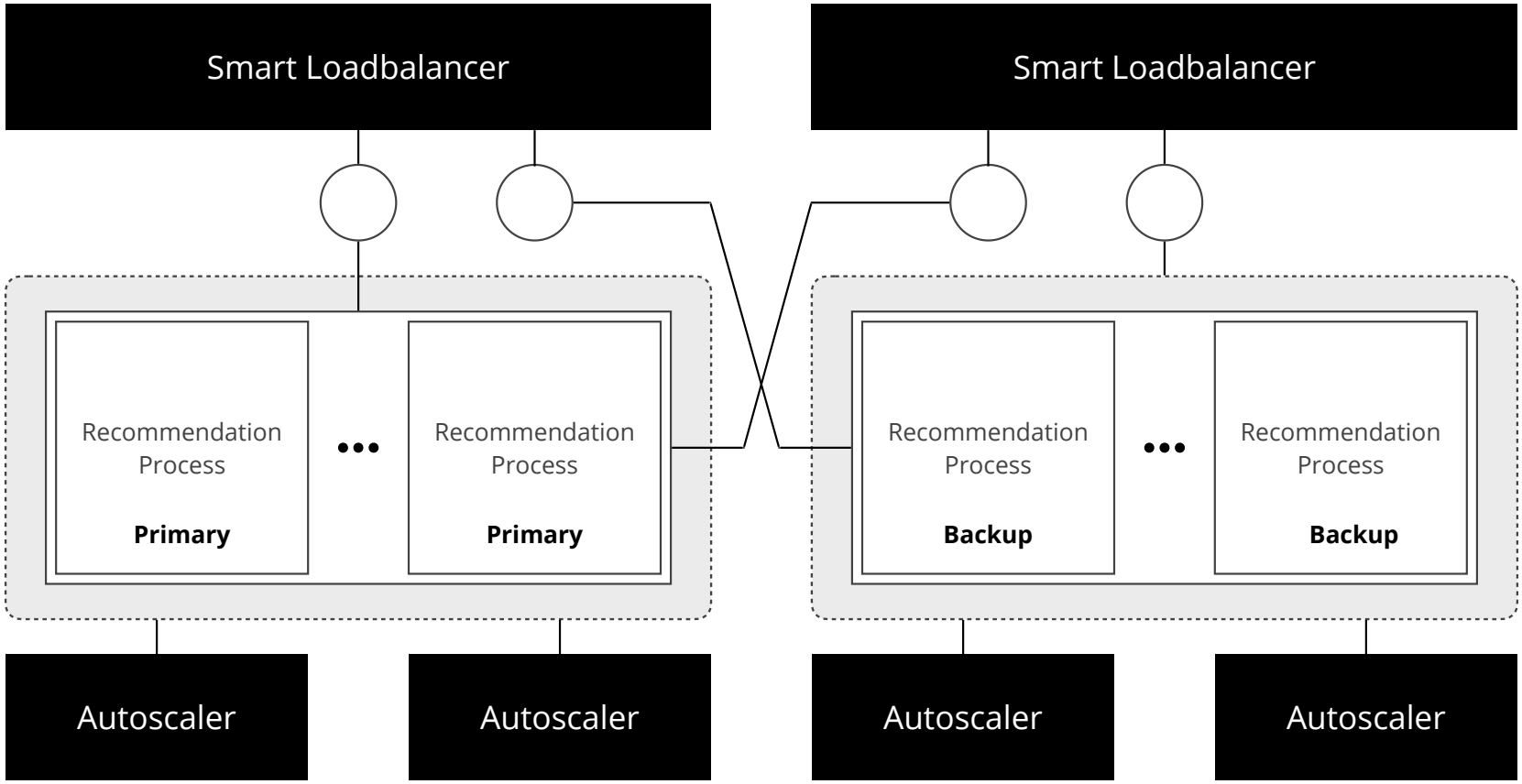

Recommendation Process
Primary
Recommendation Process
Primary
Recommendation Process
Backup
Recommendation Process
Backup
Recommendations
Recommendations

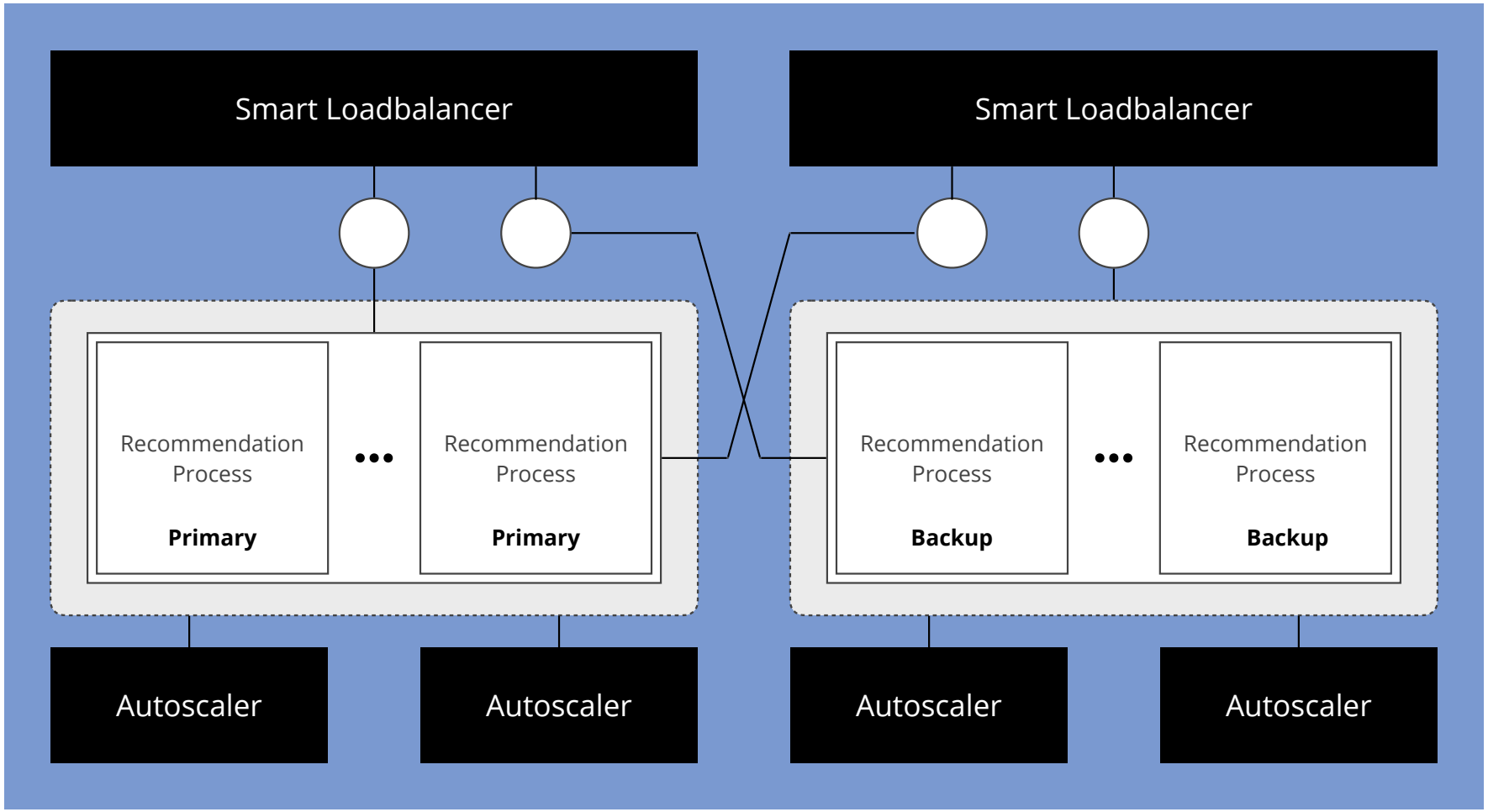

Recommendation Process
Primary
Recommendation Process
Primary
Autoscaler
Recommendation Process
Backup
Recommendation Process
Backup
Autoscaler

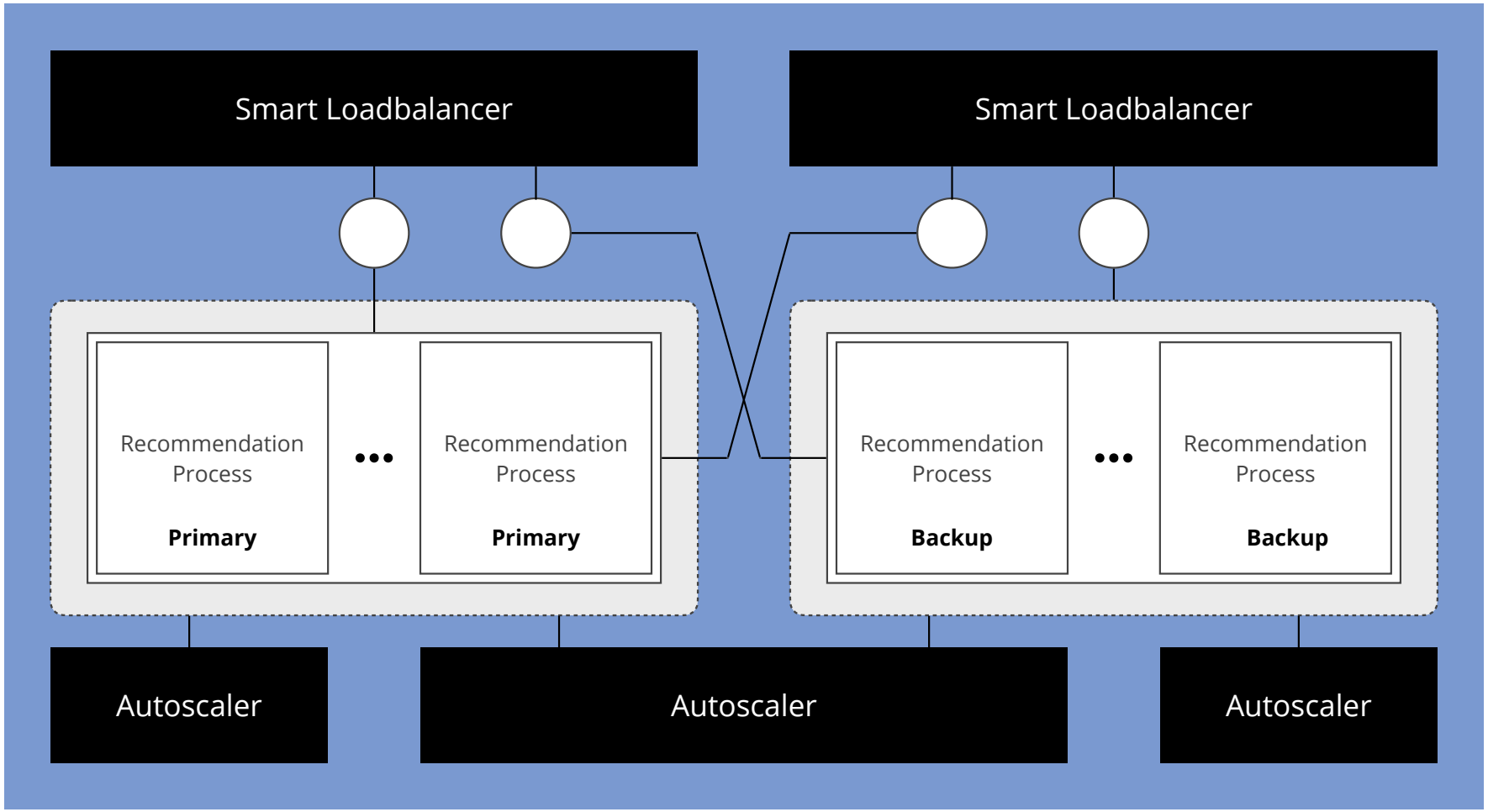

Recommendation Process
Primary
Recommendation Process
Primary
Recommendation Process
Backup
Recommendation Process
Backup
Autoscaler
Autoscaler
Autoscaler
Autoscaler

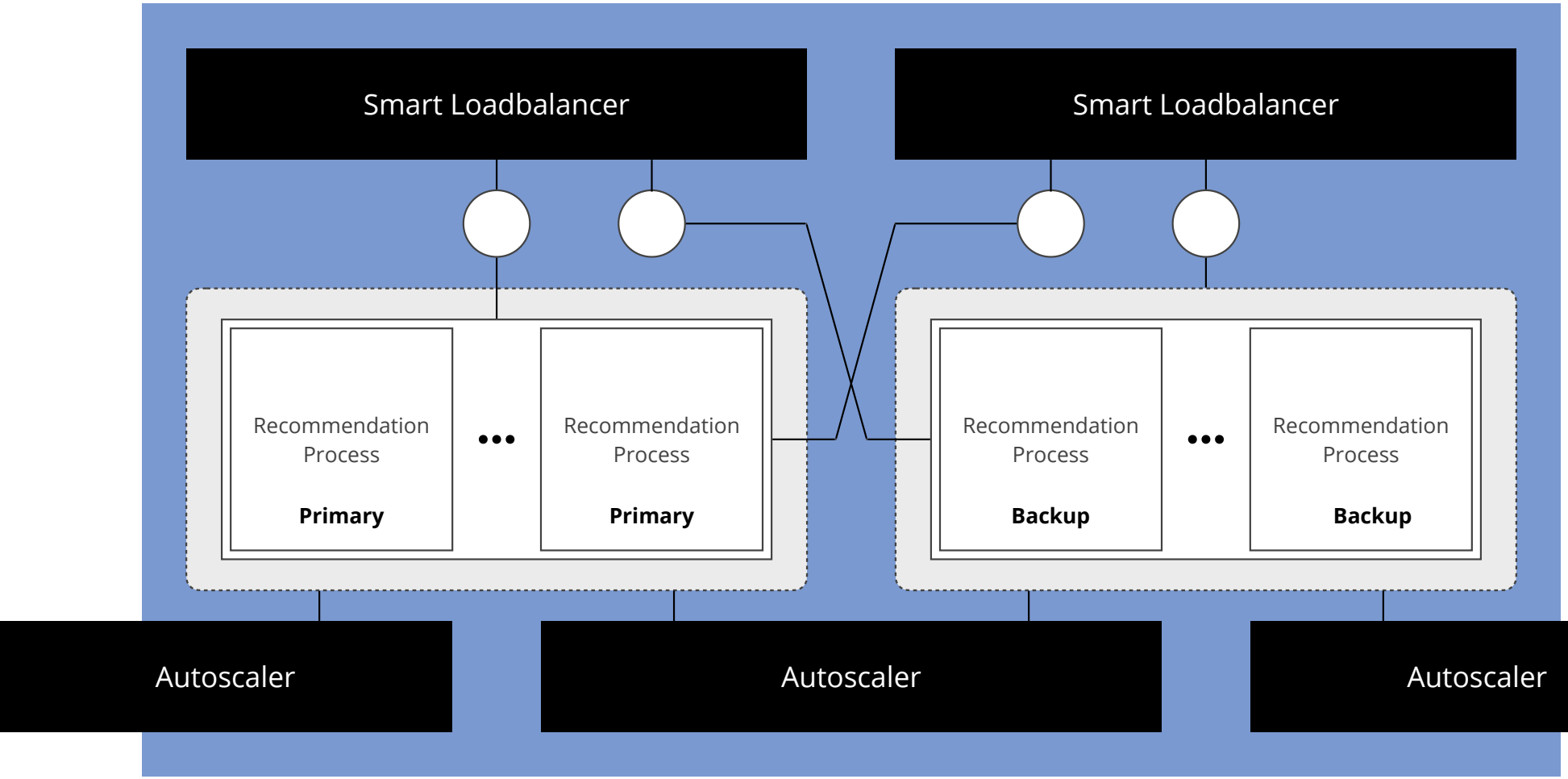

Loadbalancer
Recommendation Process
Primary
Recommendation Process
Primary
Recommendation Process
Backup
Recommendation Process
Backup
Autoscaler
Autoscaler
Autoscaler
Autoscaler


Smart Loadbalancer
Recommendation Process
Primary
Recommendation Process
Primary
Recommendation Process
Backup
Recommendation Process
Backup
Autoscaler
Autoscaler
Autoscaler
Autoscaler

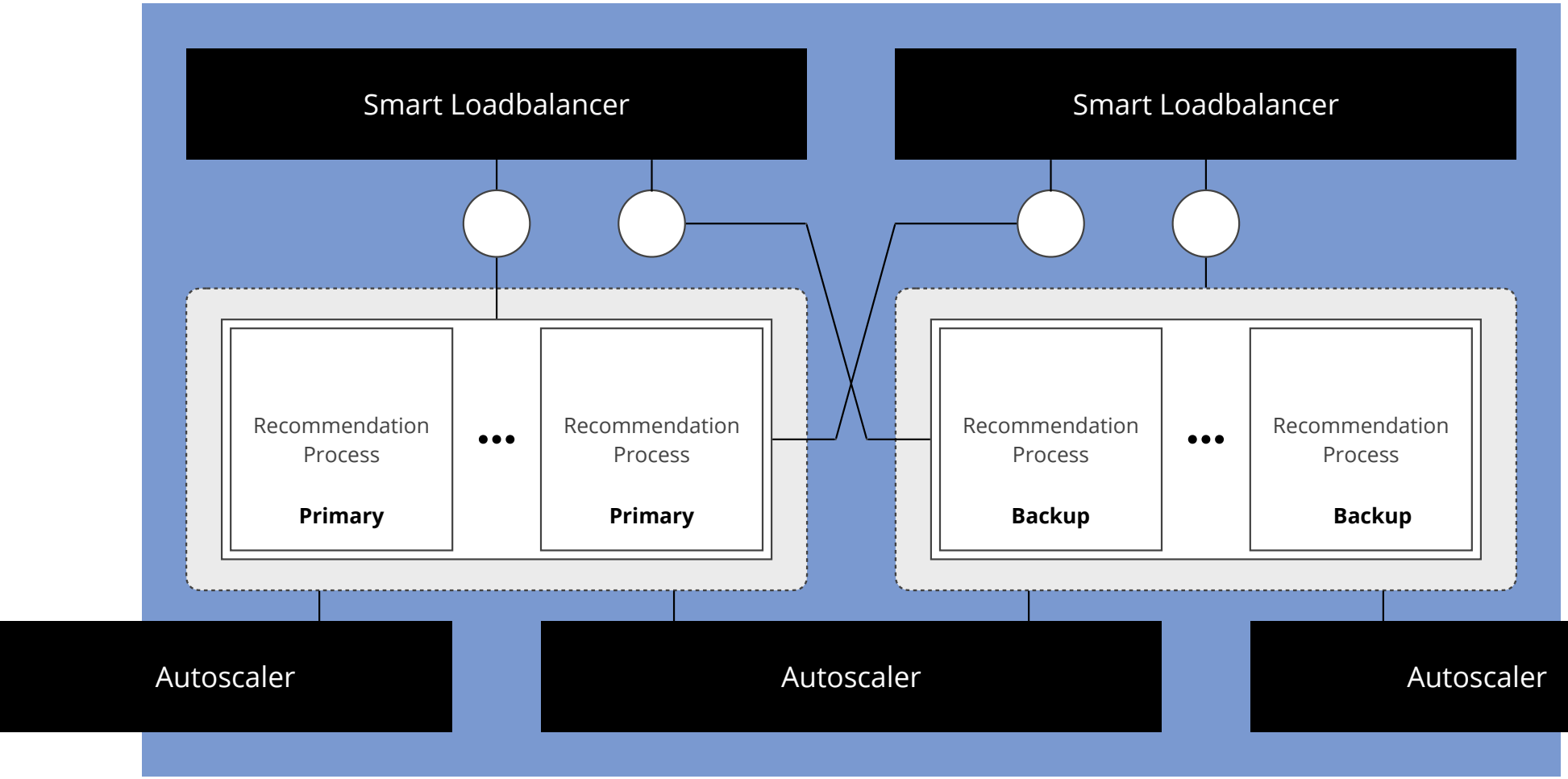

Smart Loadbalancer
Smart Loadbalancer
Recommendation Process
Primary
Recommendation Process
Primary
Recommendation Process
Backup
Recommendation Process
Backup
Autoscaler
Autoscaler
Autoscaler
Autoscaler

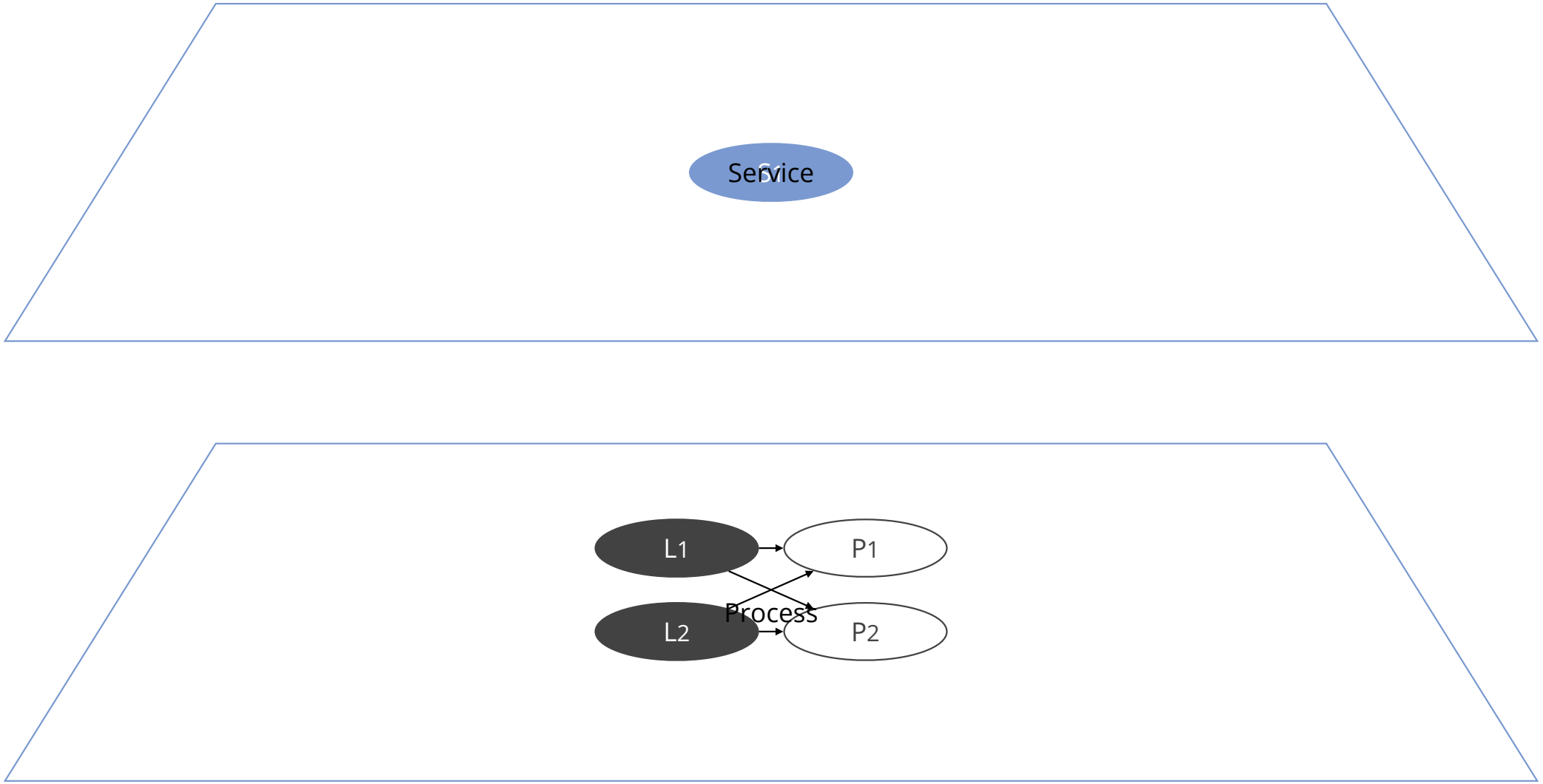

Smart Loadbalancer
Smart Loadbalancer
Recommendation Process
Primary
Recommendation Process
Primary
Recommendation Process
Backup
Recommendation Process
Backup
Autoscaler
Autoscaler
Autoscaler
Autoscaler

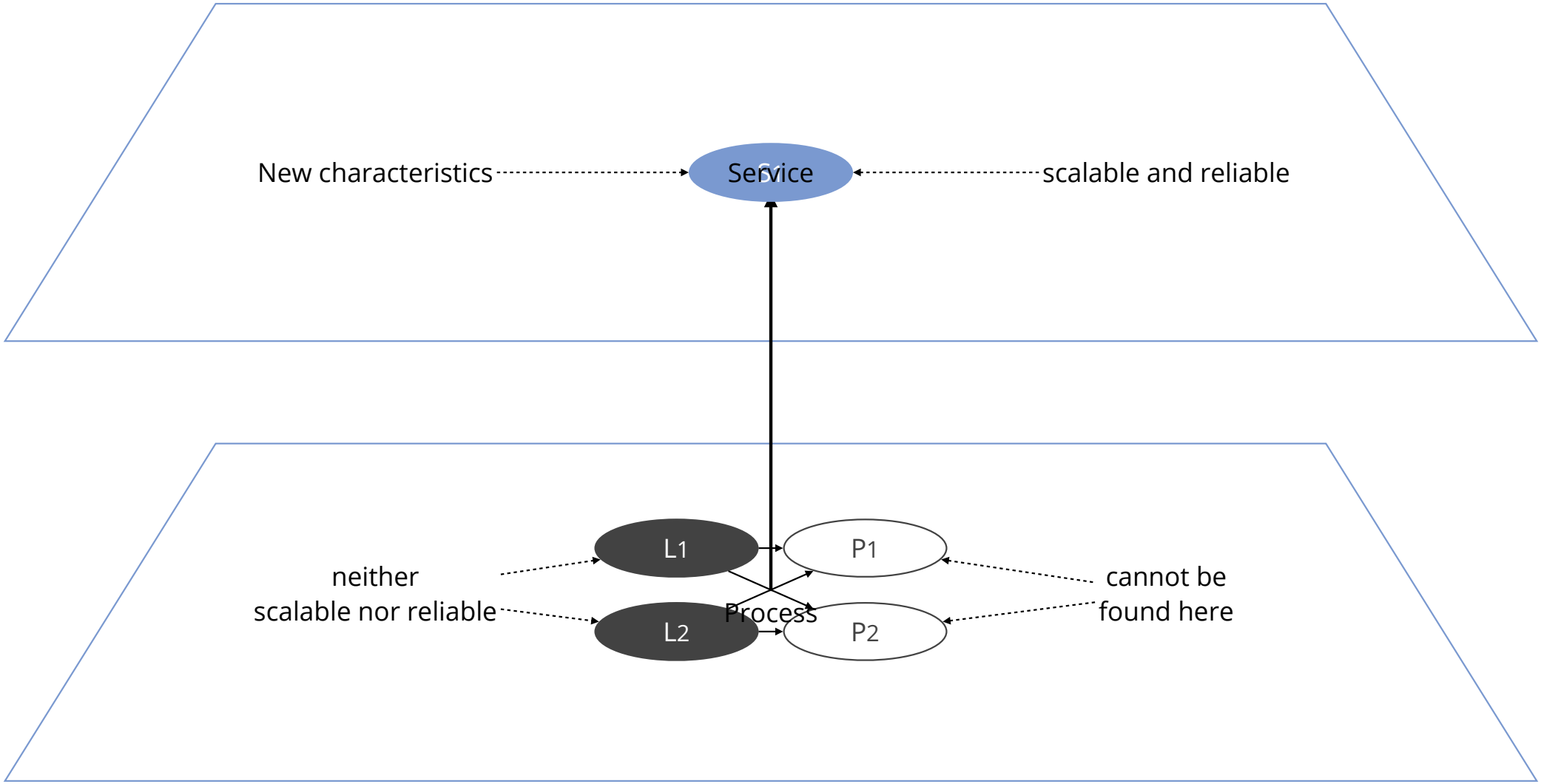

Smart Loadbalancer
Smart Loadbalancer
Recommendation Process
Primary
Recommendation Process
Primary
Recommendation Process
Backup
Recommendation Process
Backup
Autoscaler
Autoscaler
Autoscaler

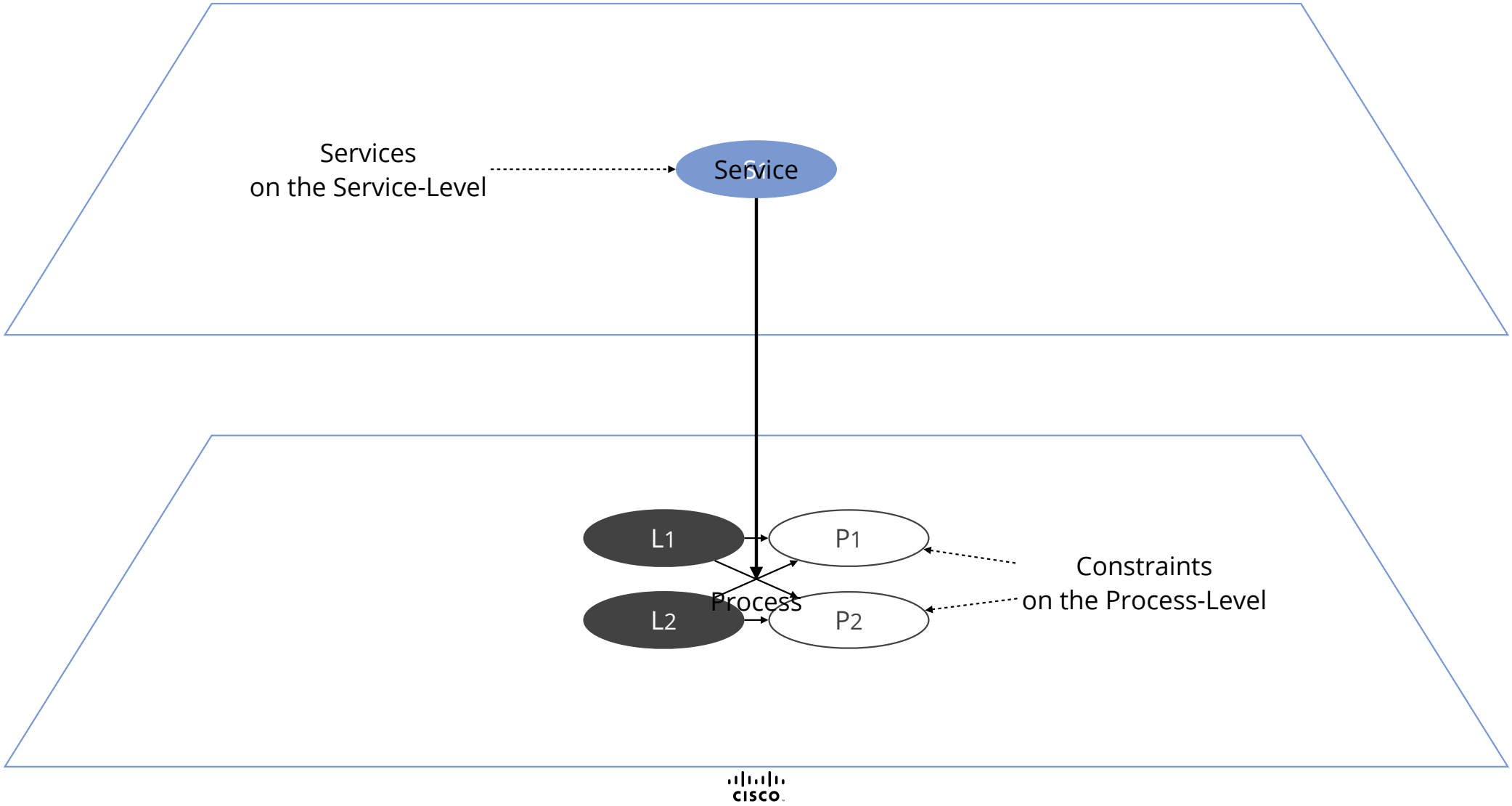

Smart Loadbalancer
Smart Loadbalancer
Recommendation Process
Primary
Recommendation Process
Primary
Recommendation Process
Backup
Recommendation Process
Backup
Autoscaler
Autoscaler
Autoscaler

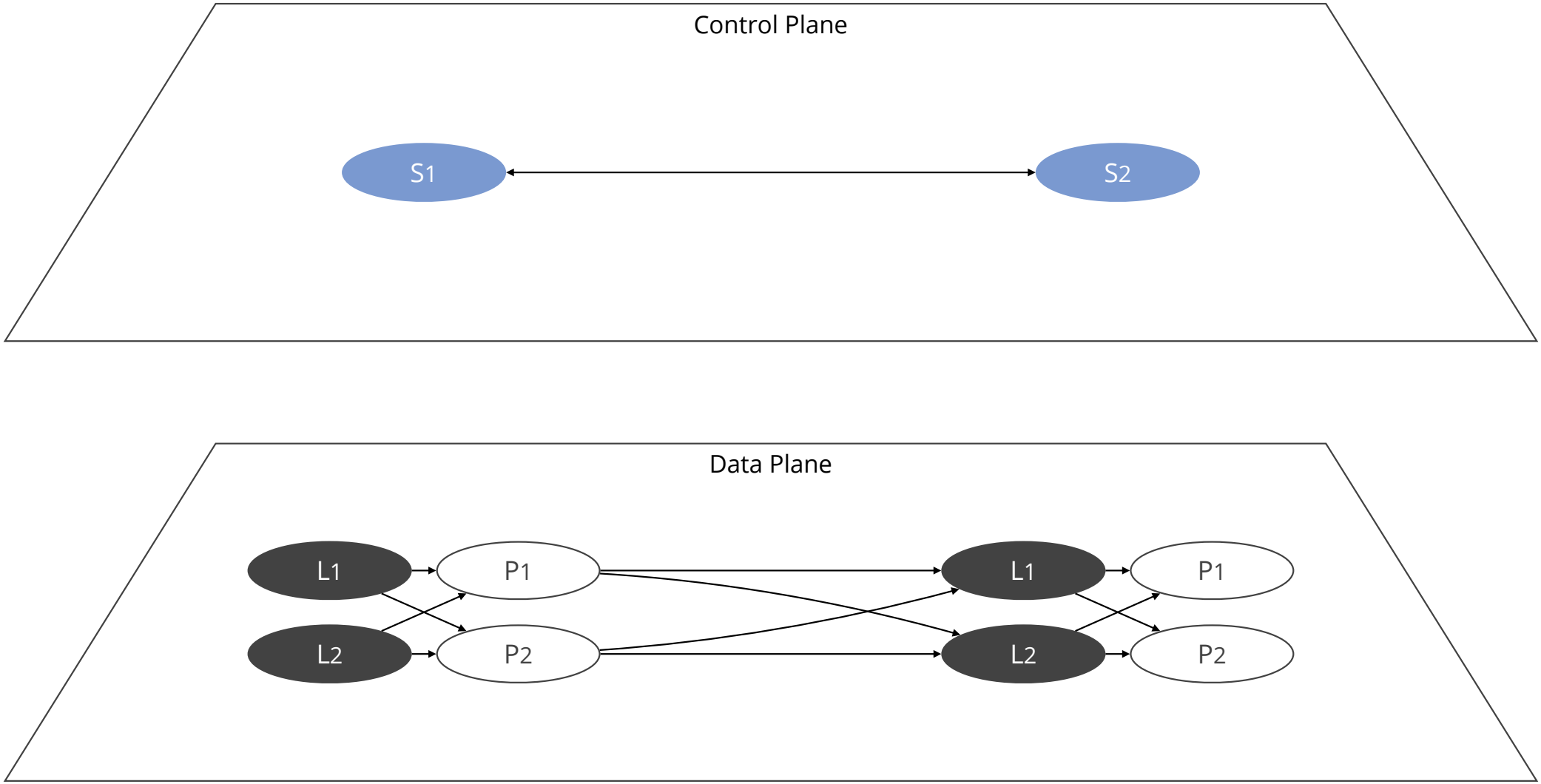
(Most Important Slide Of The Entire Presentation)


Smart Loadbalancer
Smart Loadbalancer
Recommendation Process
Primary
Recommendation Process
Primary
Recommendation Process
Backup
Recommendation Process
Backup
Autoscaler
Autoscaler
Autoscaler


Service
L1
L2
P1
P2
Process

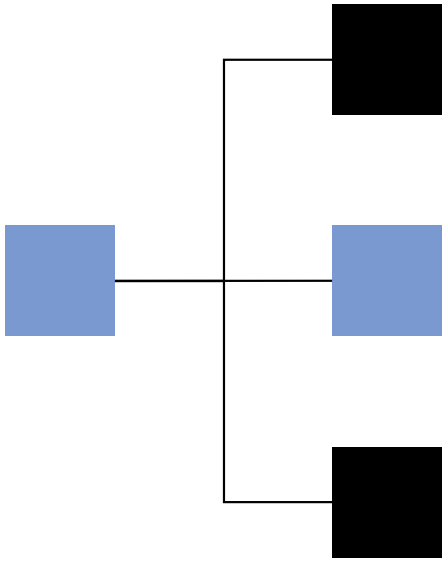

New characteristics
Service
scalable and reliable
L1
P1
neither
scalable nor reliable
cannot be
found here
Process
L2
P2


Services
on the Service-Level
Service
L1
L2
Process
P1
P2
Constraints
on the Process-Level


Control Plane
S1
S2
Data Plane
L1
P1
L2
P2
L1
P1
L2
P2

Service Model

Definition

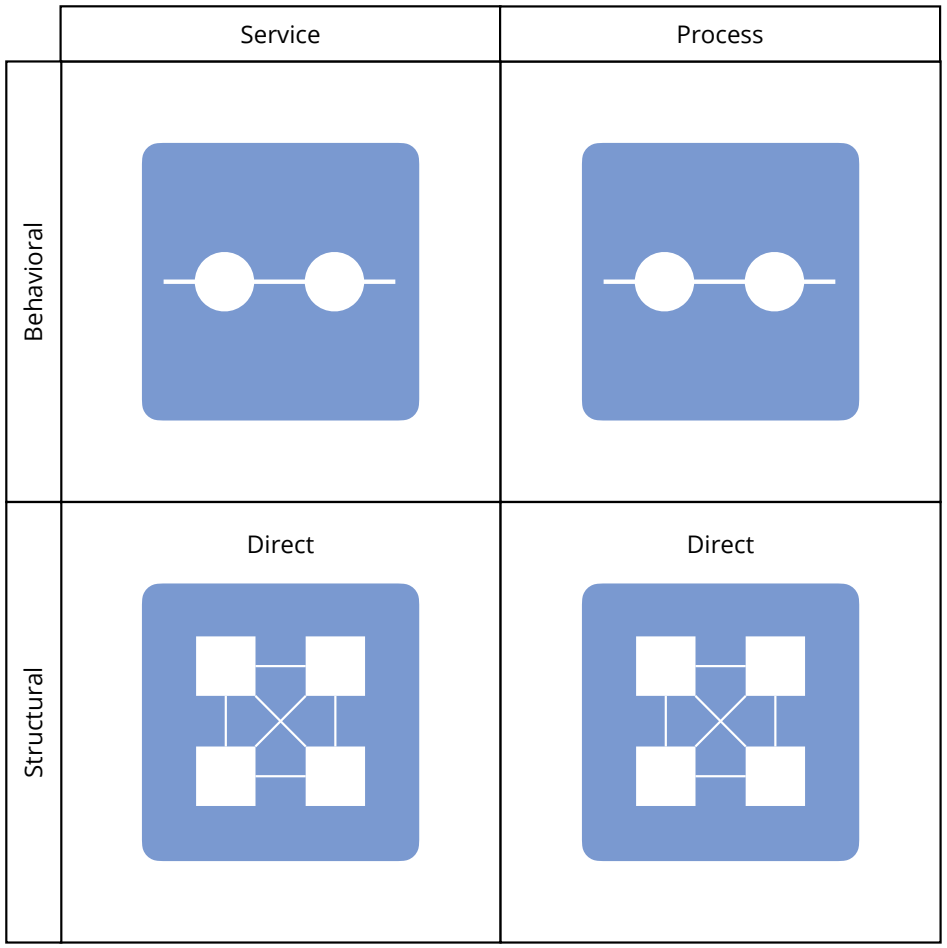


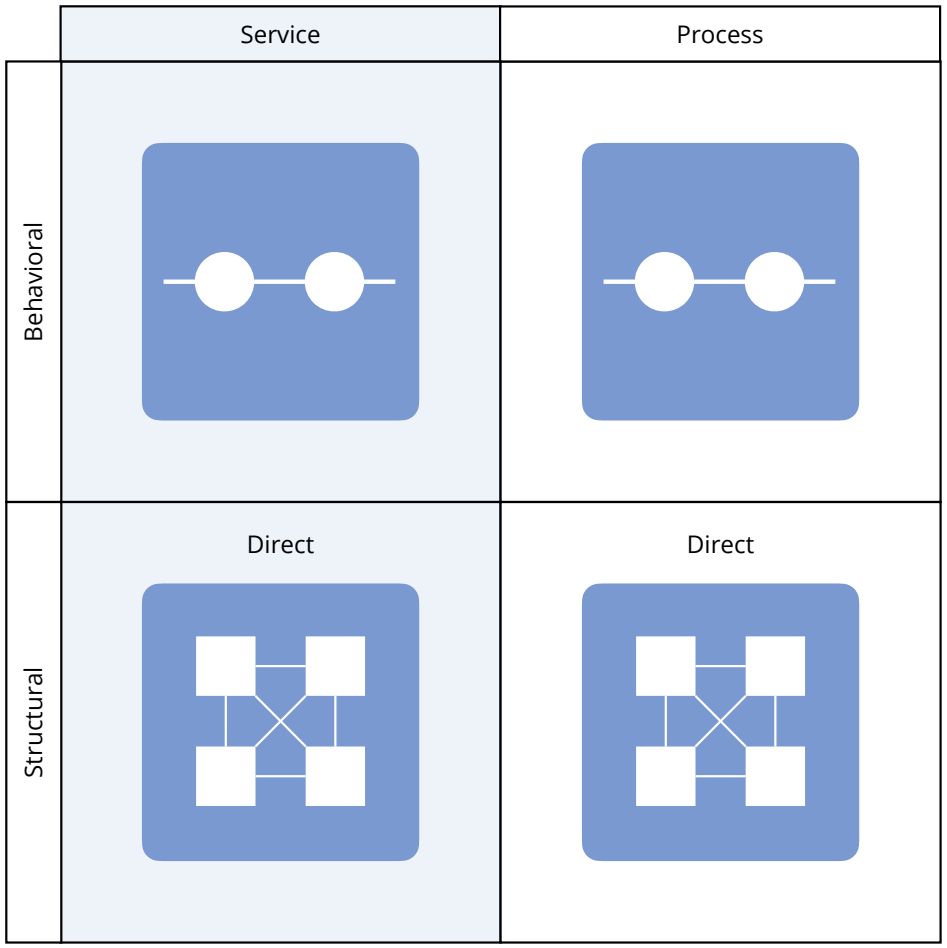
KVS

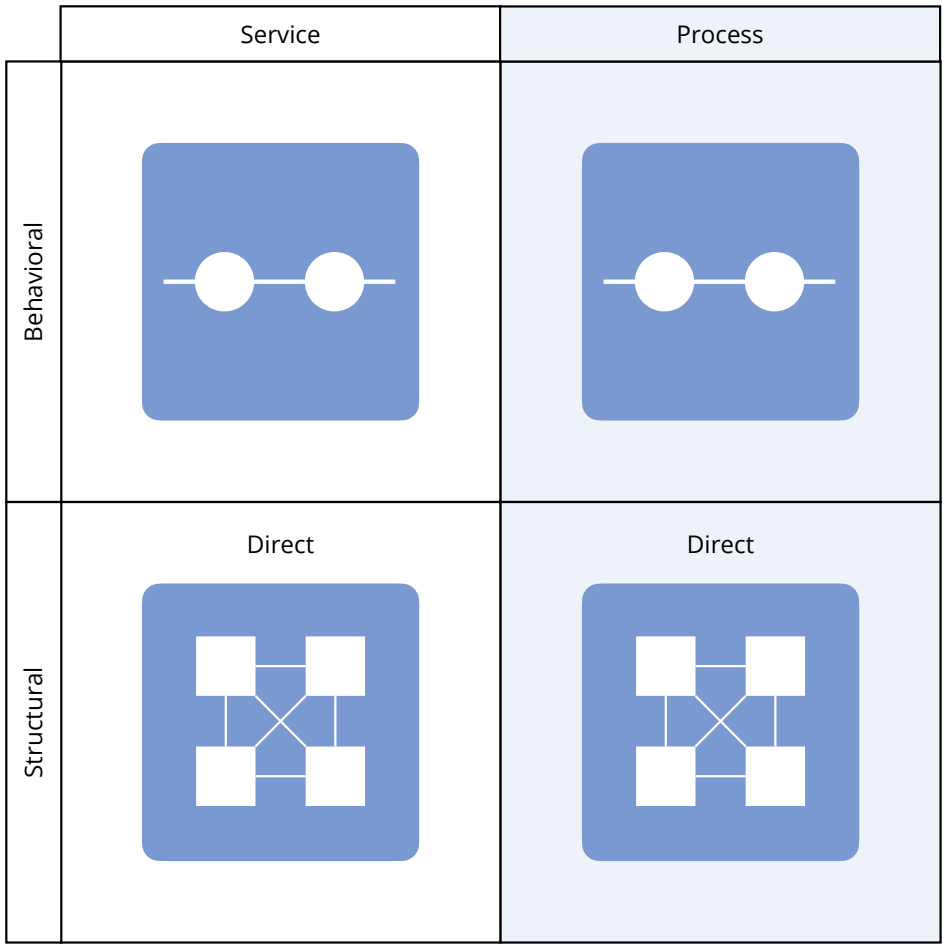
```
get: KVS ⟶ Int ⟶ String
put: KVS ⟶ Int ⟶ String ⟶ KVS
```

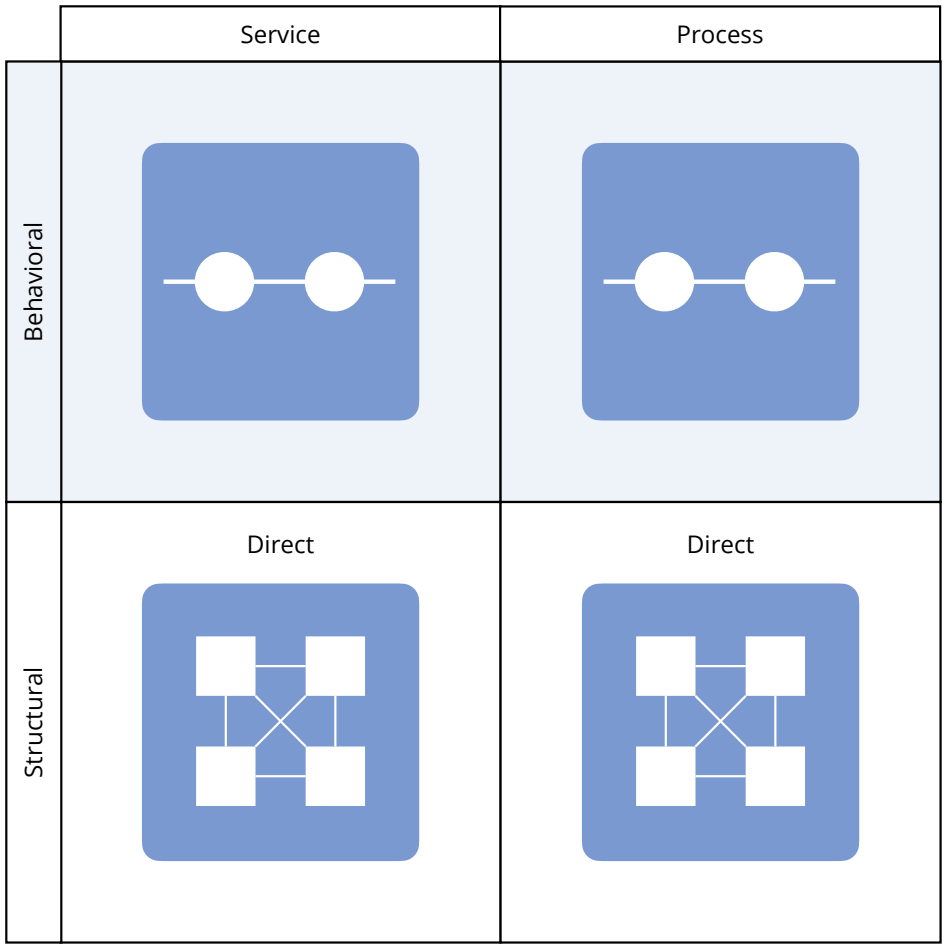
Axioms

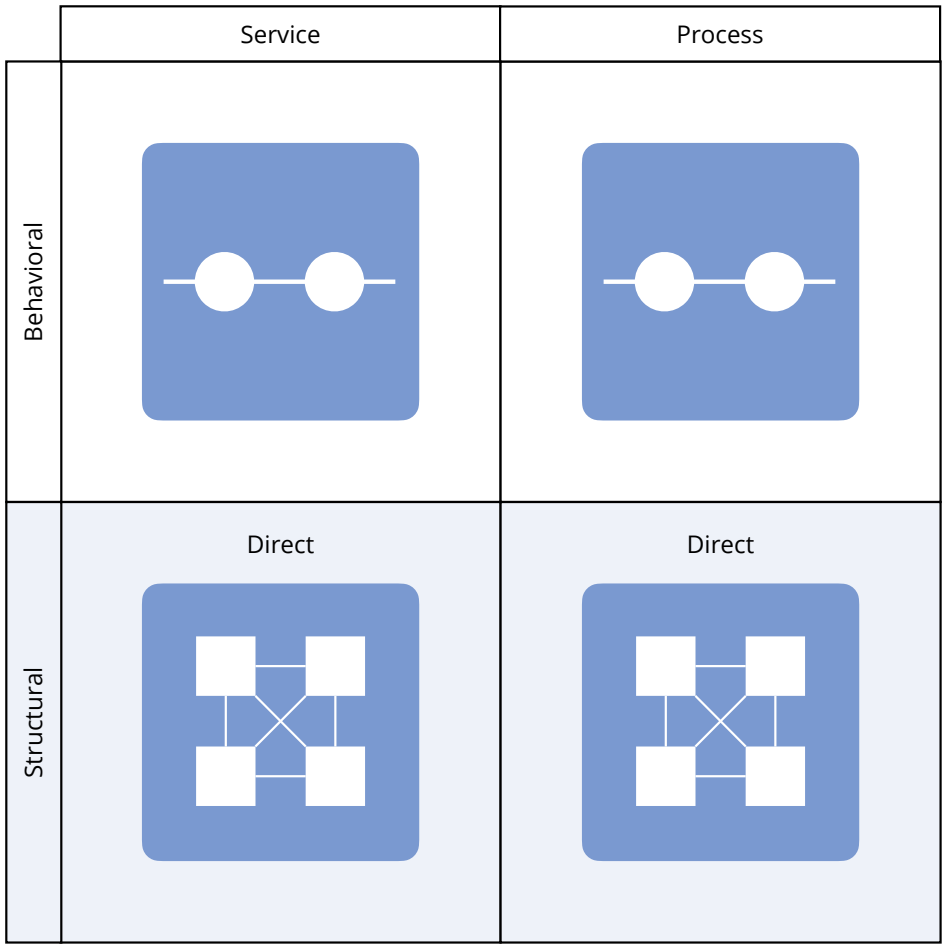
```
get (empty) k ⟶ ""
get (put (empty) k v) k ⟶ v
```

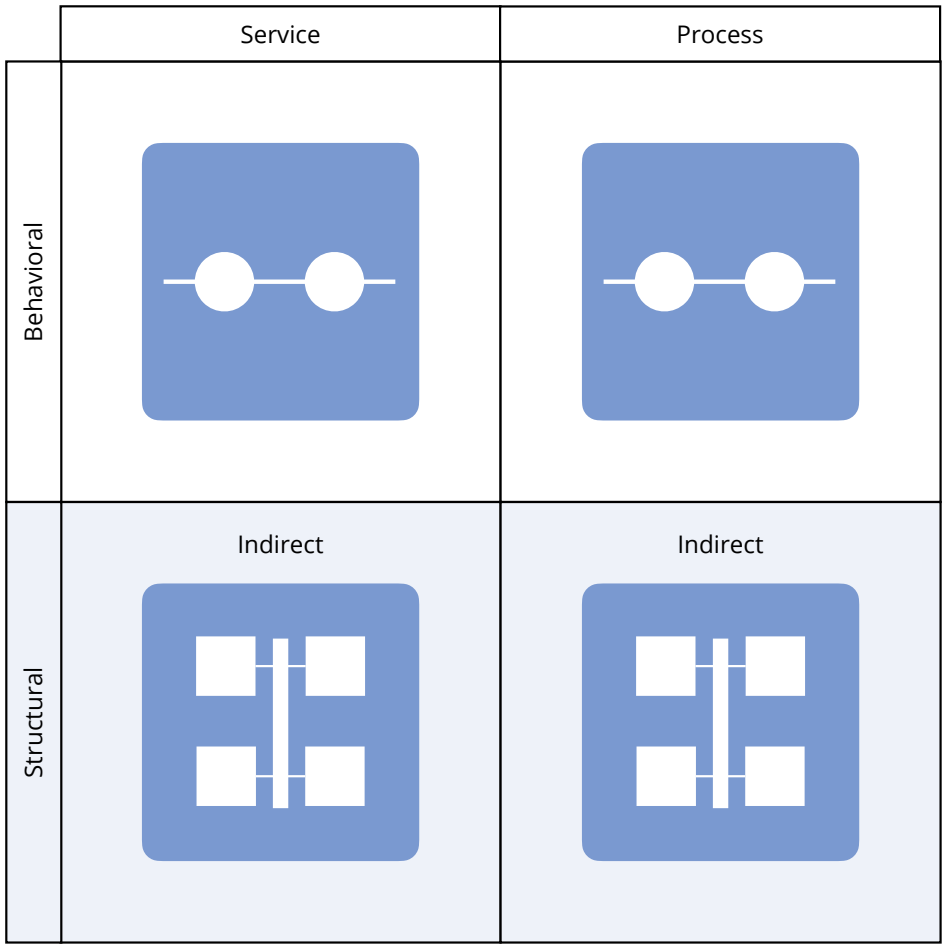

Service
Process
Behavioral
Structural
Direct
Direct


Service
Process
Behavioral
Structural
Direct
Direct


Service
Process
Behavioral
Structural
Direct
Direct

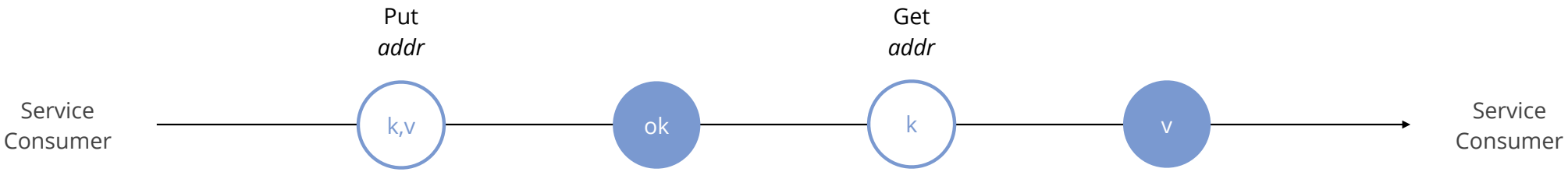

Service
Process
Behavioral
Structural
Direct
Direct

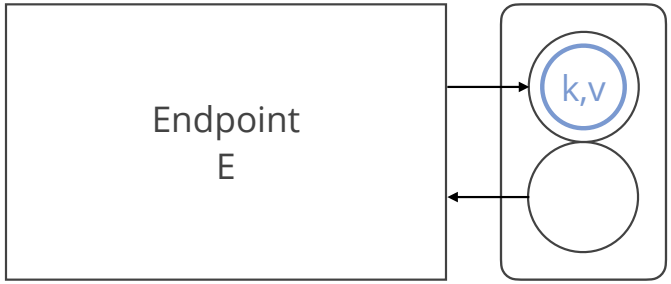

Service
Process
Behavioral
Structural
Direct
Direct


Service
Process
Behavioral
Structural
Indirect
Indirect

Service Model

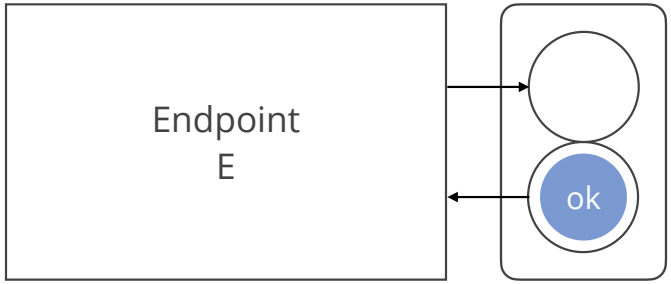
The Contract


Put
addr
Get
addr
Service
Consumer
k,v
ok
k
v
Service
Consumer



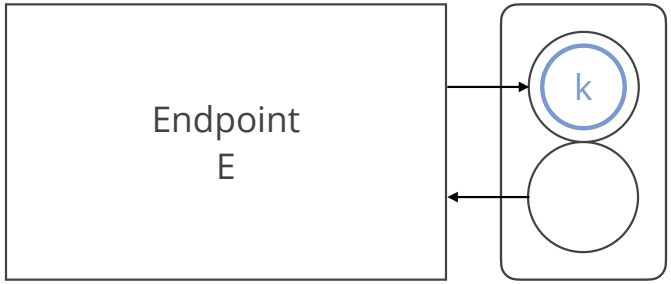
History[E]

[Send●Req=(addr, Put(k,v))]



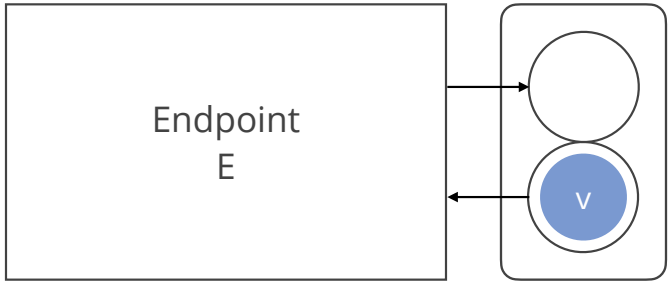
History[E]

[Send•Req=(addr, Put(k,v)), Recv•Res=ok]



History[E]

[Send•Req=(addr, Put(k,v)), Recv•Res=ok, Send•Req=(addr, Get(k))]



History[E]

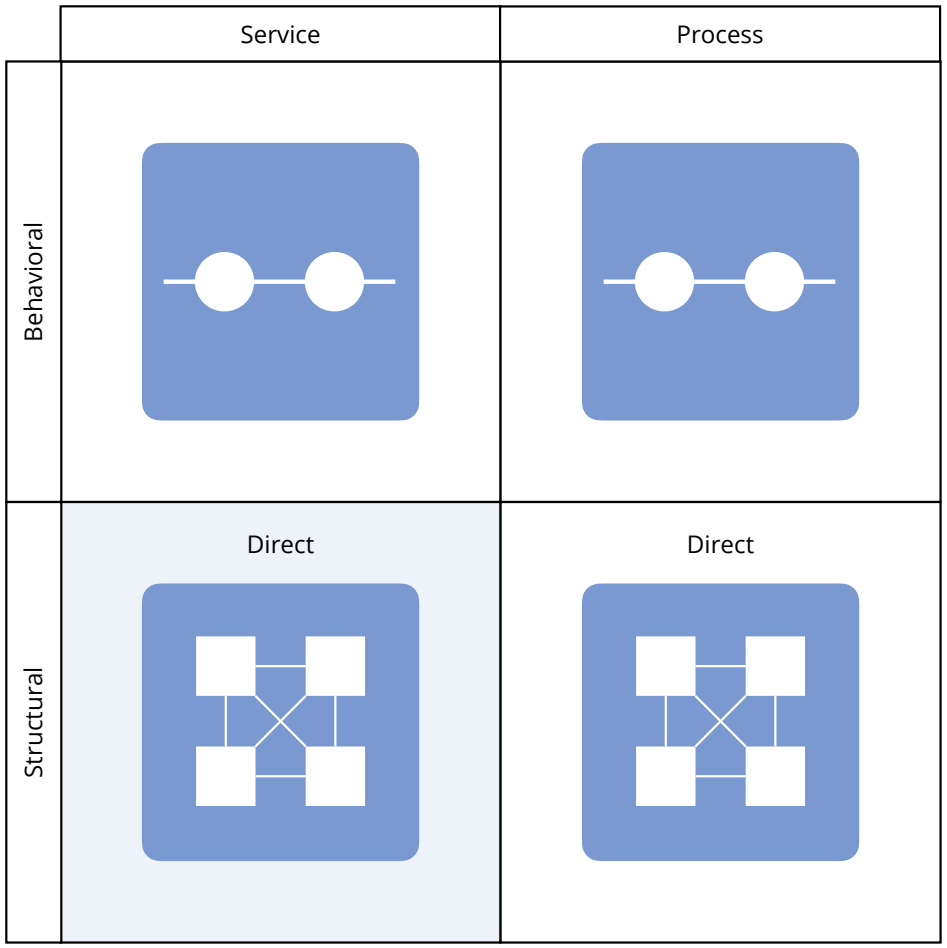
[Send•Req=(addr, Put(k,v)), Recv•Res=ok, Send•Req=(addr, Get(k)), Recv•Res=v]

Service Model

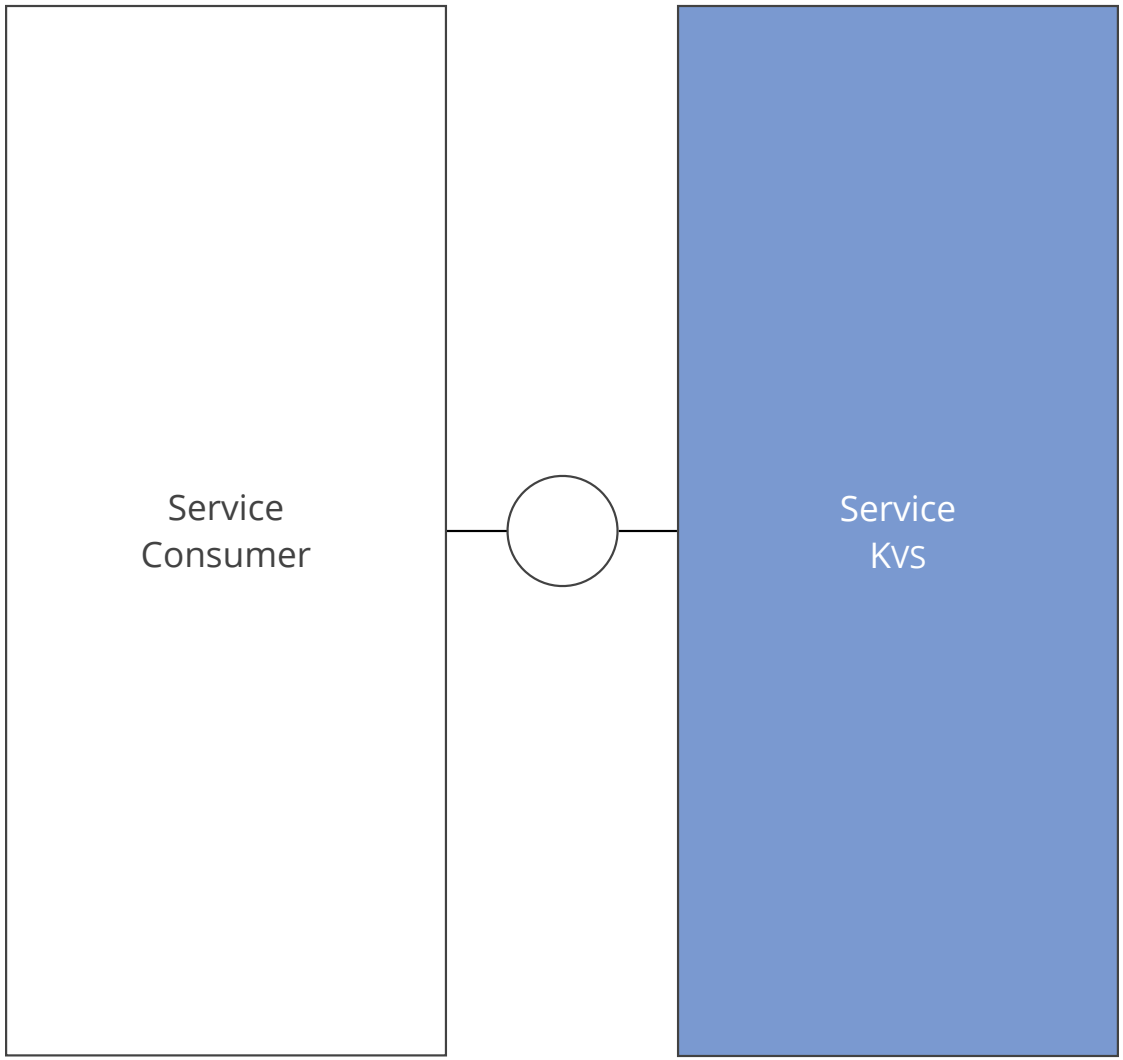
The Component

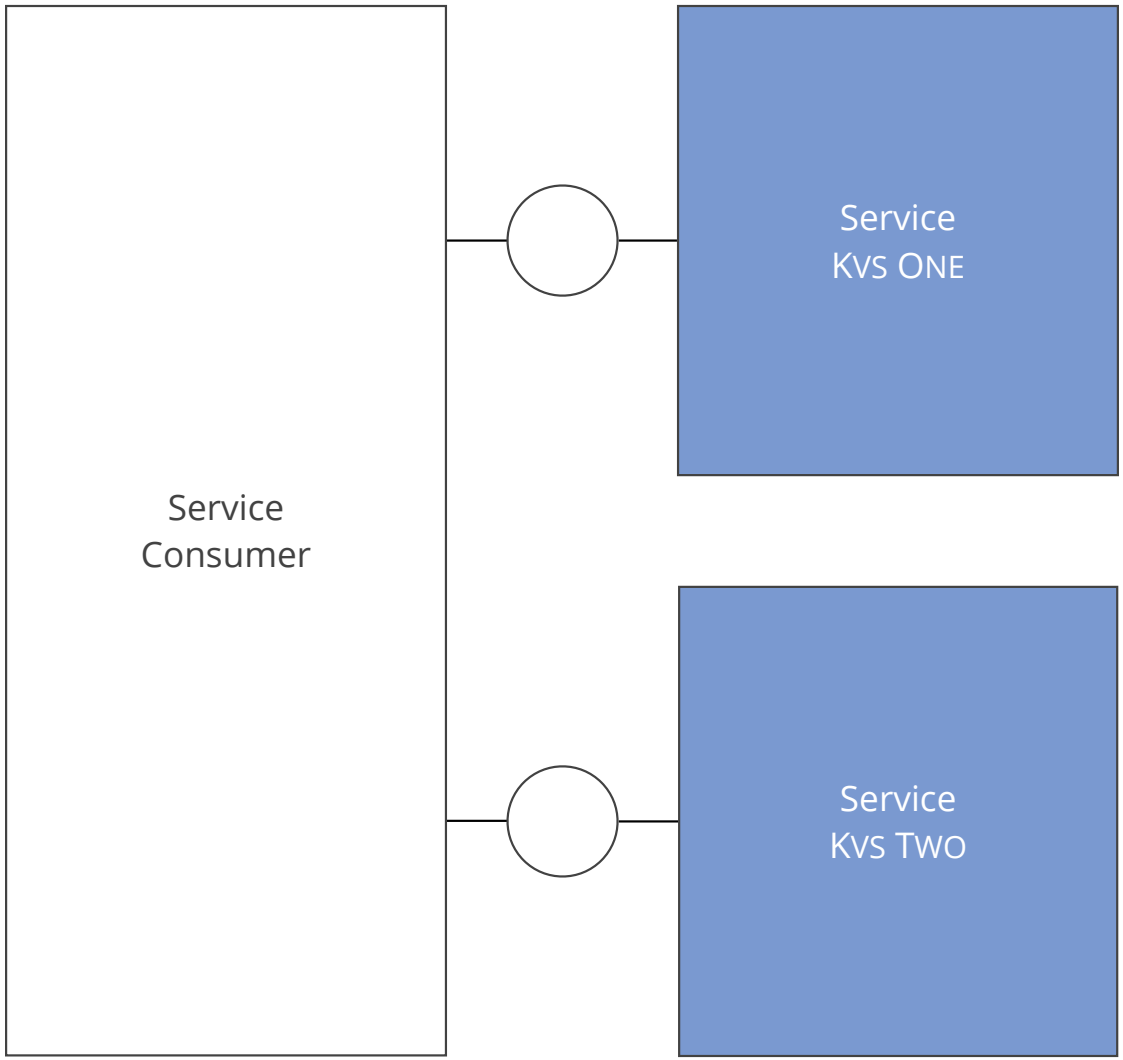
Service Model

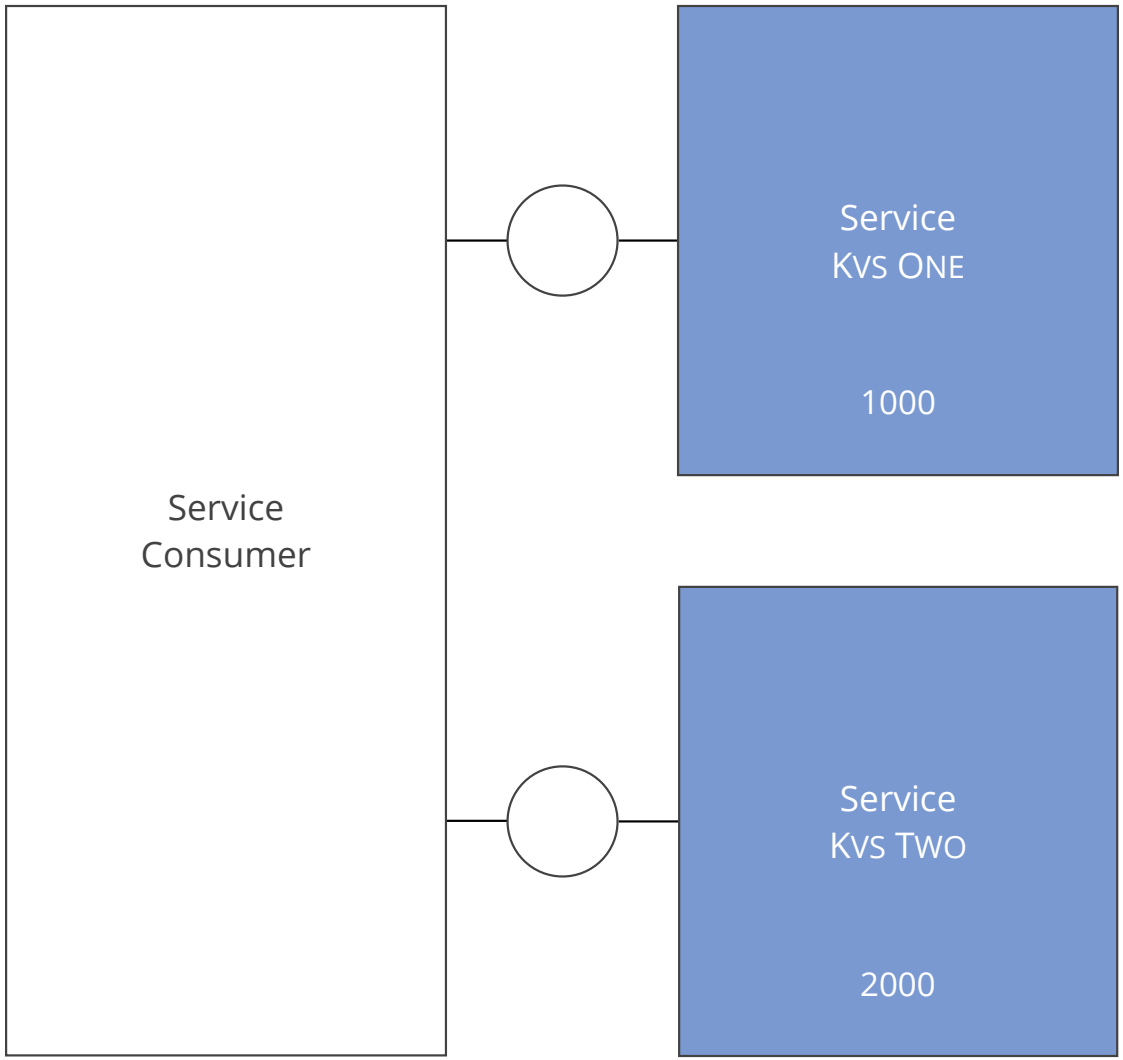
Service to Service

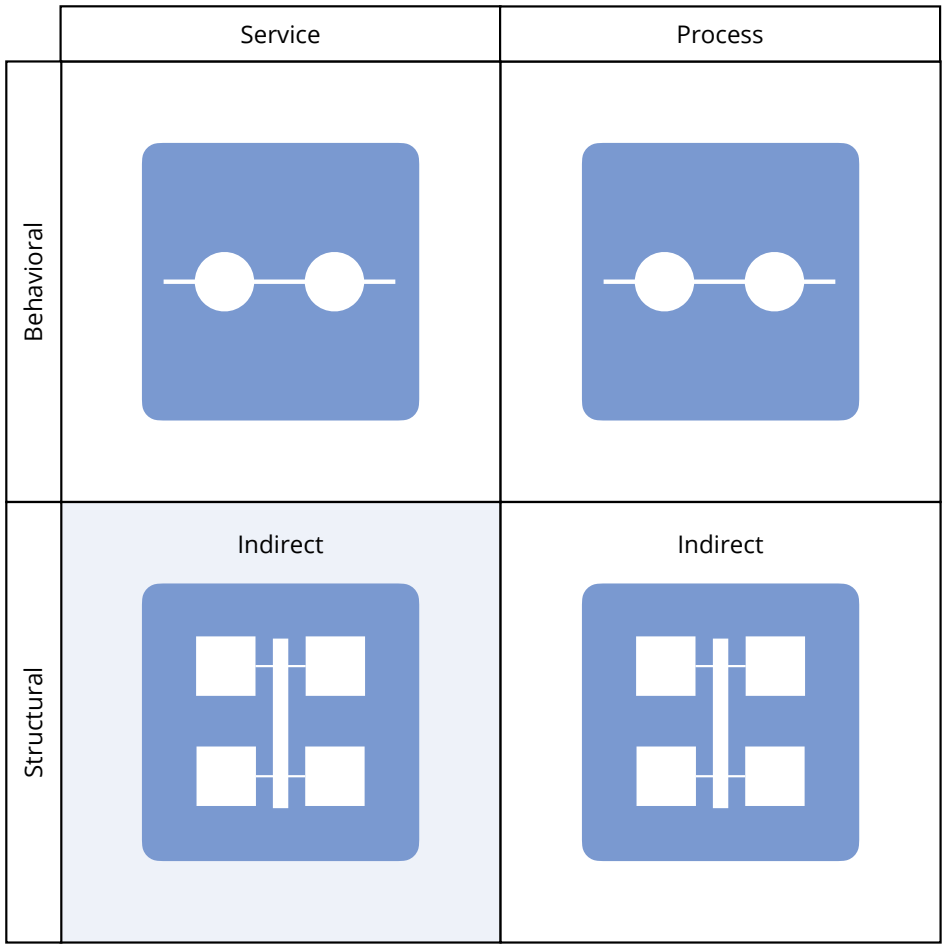

Service
Process
Behavioral
Structural
Direct
Direct

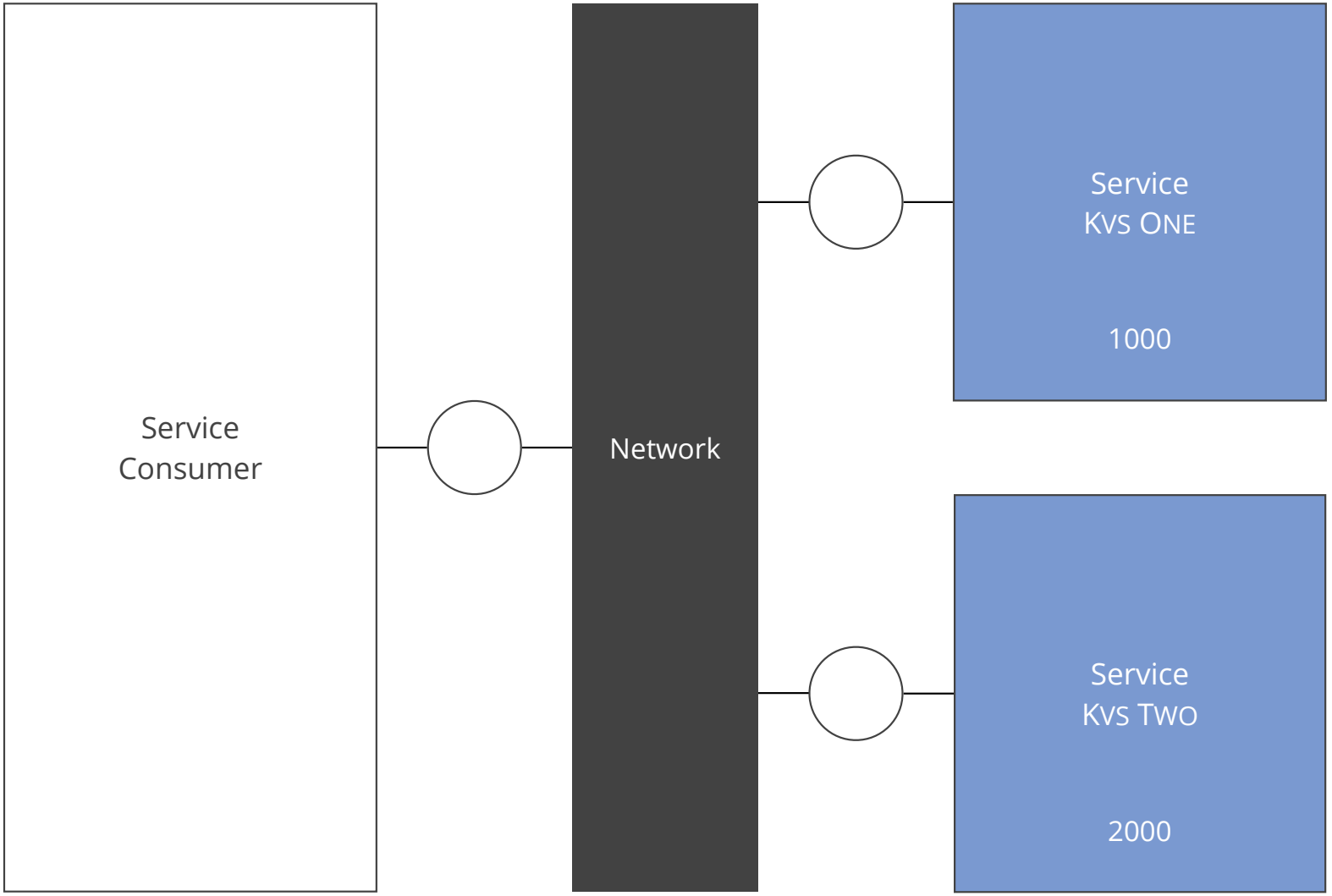
Service to Service • Structural • Direct Communication

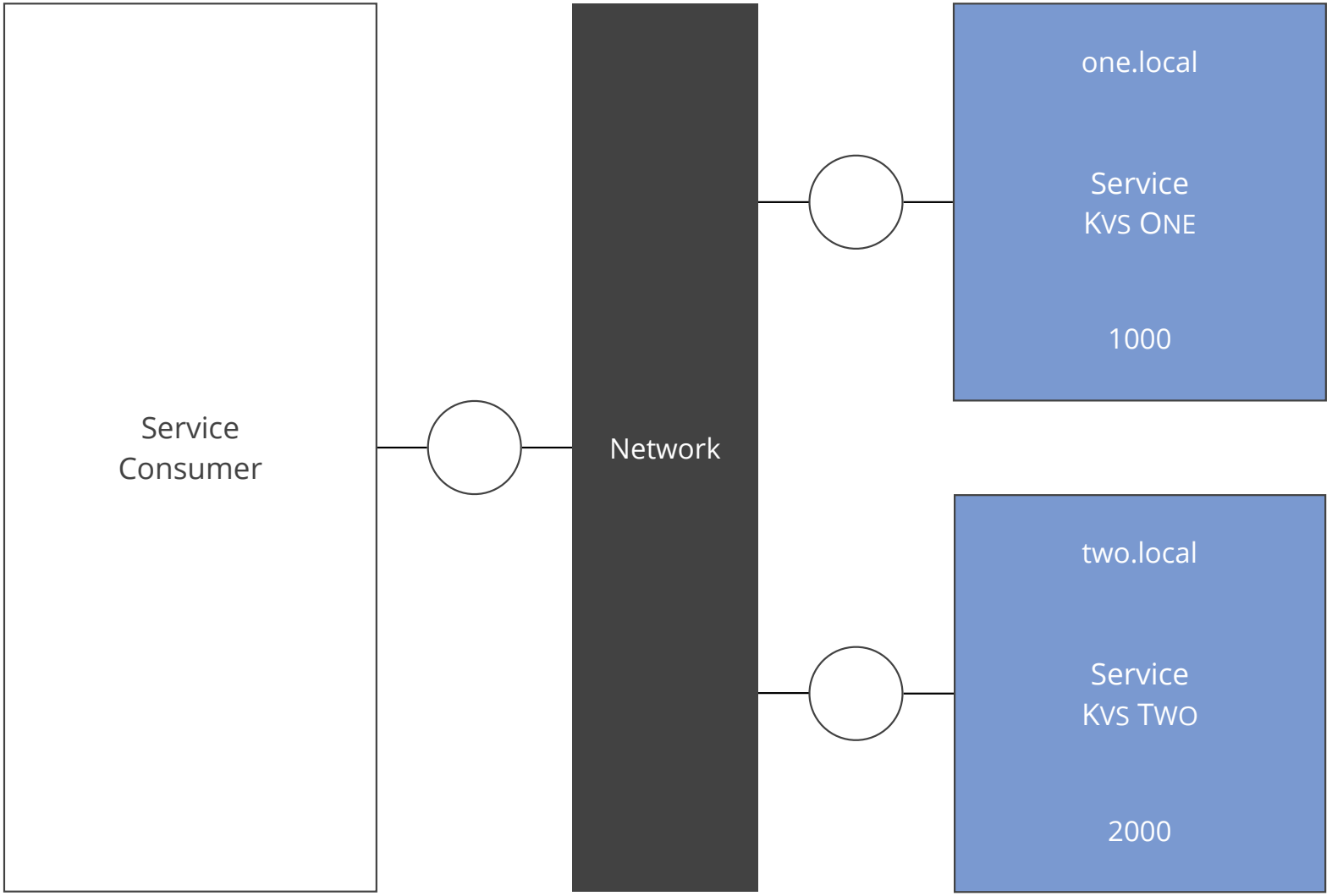


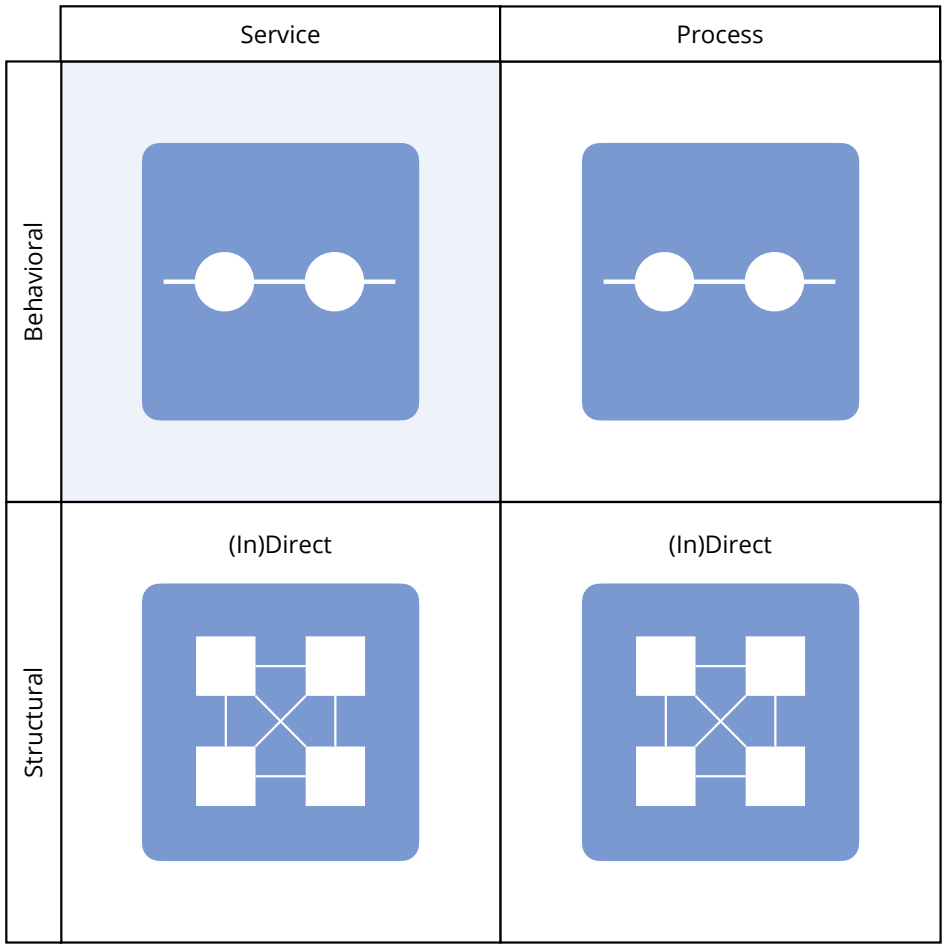

Service
Consumer
Service
KVS ONE
Service
KVS TWO

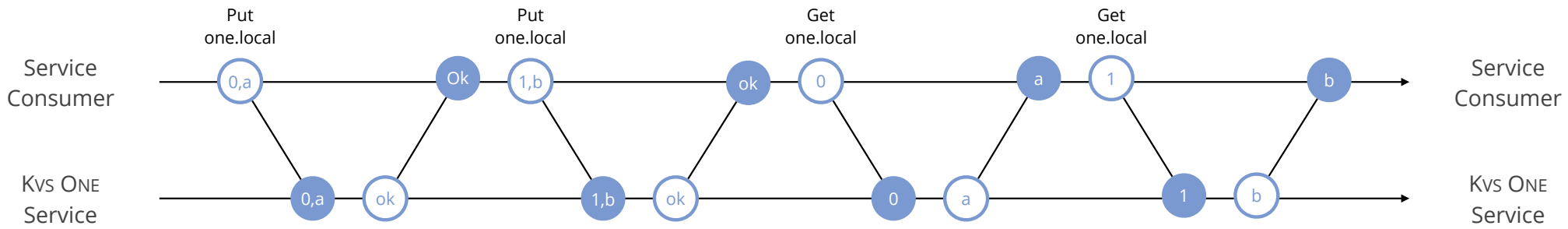

Service
Consumer
Service
KVS ONE
1000
Service
KVS TWO
2000

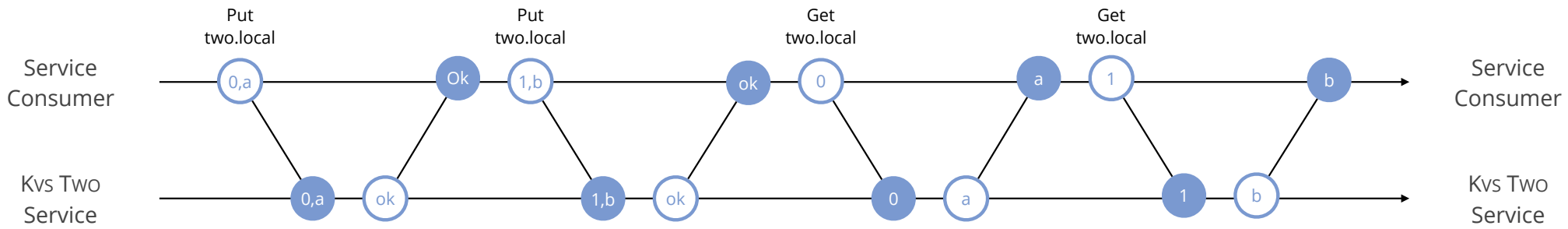

Service
Process
Behavioral
Structural
Indirect
Indirect

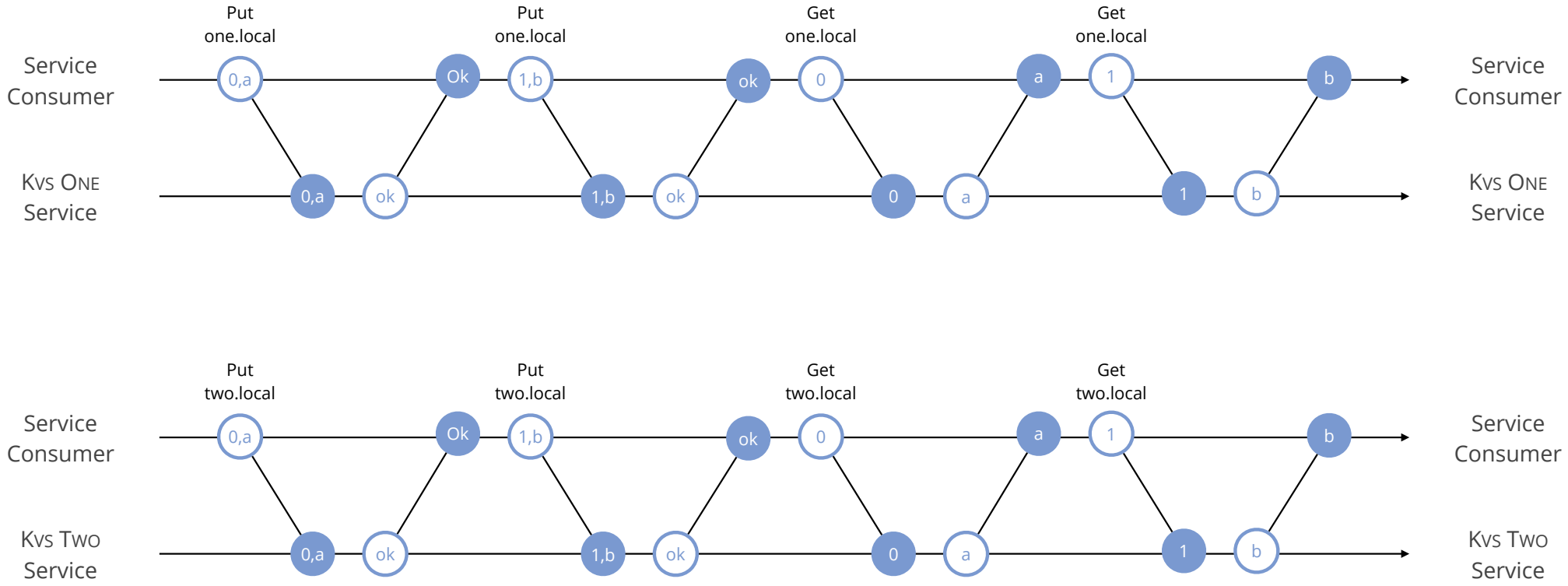

Service
Consumer
Network
Service
KVS ONE
1000
Service
KVS TWO
2000


Service
Consumer
Network
one.local
Service
KVS ONE
1000
two.local
Service
KVS TWO
2000


Service
Process
Behavioral
Structural
(In)Direct
(In)Direct


Put
one.local
Put
one.local
Get
one.local
Get
one.local
Service
Consumer
0,a
Ok
1,b
ok
0
a
1
b
Service
Consumer
KVS ONE
Service
0,a
ok
1,b
ok
0
a
1
b
KVS ONE
Service


Put
two.local
Put
two.local
Get
two.local
Get
two.local
Service
Consumer
0,a
Ok
1,b
ok
0
a
1
b
Service
Consumer
Kvs Two
Service
0,a
ok
1,b
ok
0
a
1
b
Kvs Two
Service


Put
one.local
Put
one.local
Get
one.local
Get
one.local
Service
Consumer
0,a
Ok
1,b
ok
0
a
1
b
Service
Consumer
KVS ONE
Service
0,a
ok
1,b
ok
0
a
1
b
KVS ONE
Service
Put
two.local
Put
two.local
Get
two.local
Get
two.local
Service
Consumer
0,a
Ok
1,b
ok
0
a
1
b
Service
Consumer
KVS TWO
Service
0,a
ok
1,b
ok
0
a
1
b
KVS TWO
Service

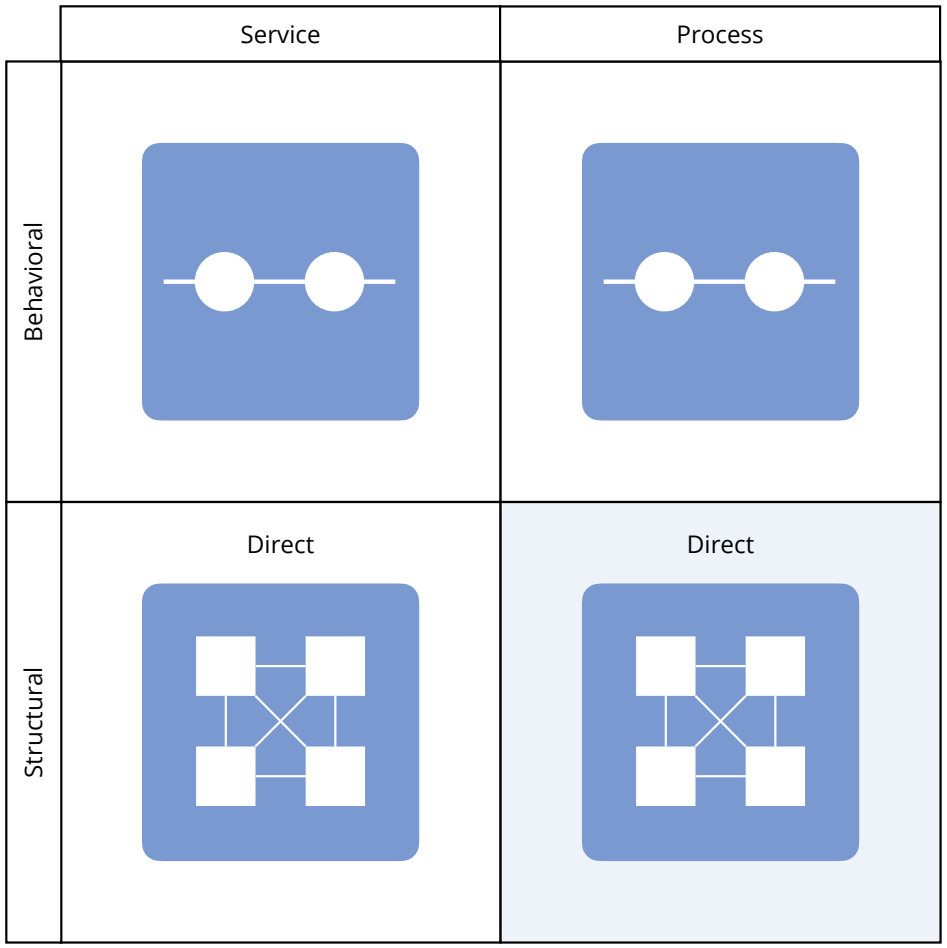
Service Model

Process to Process

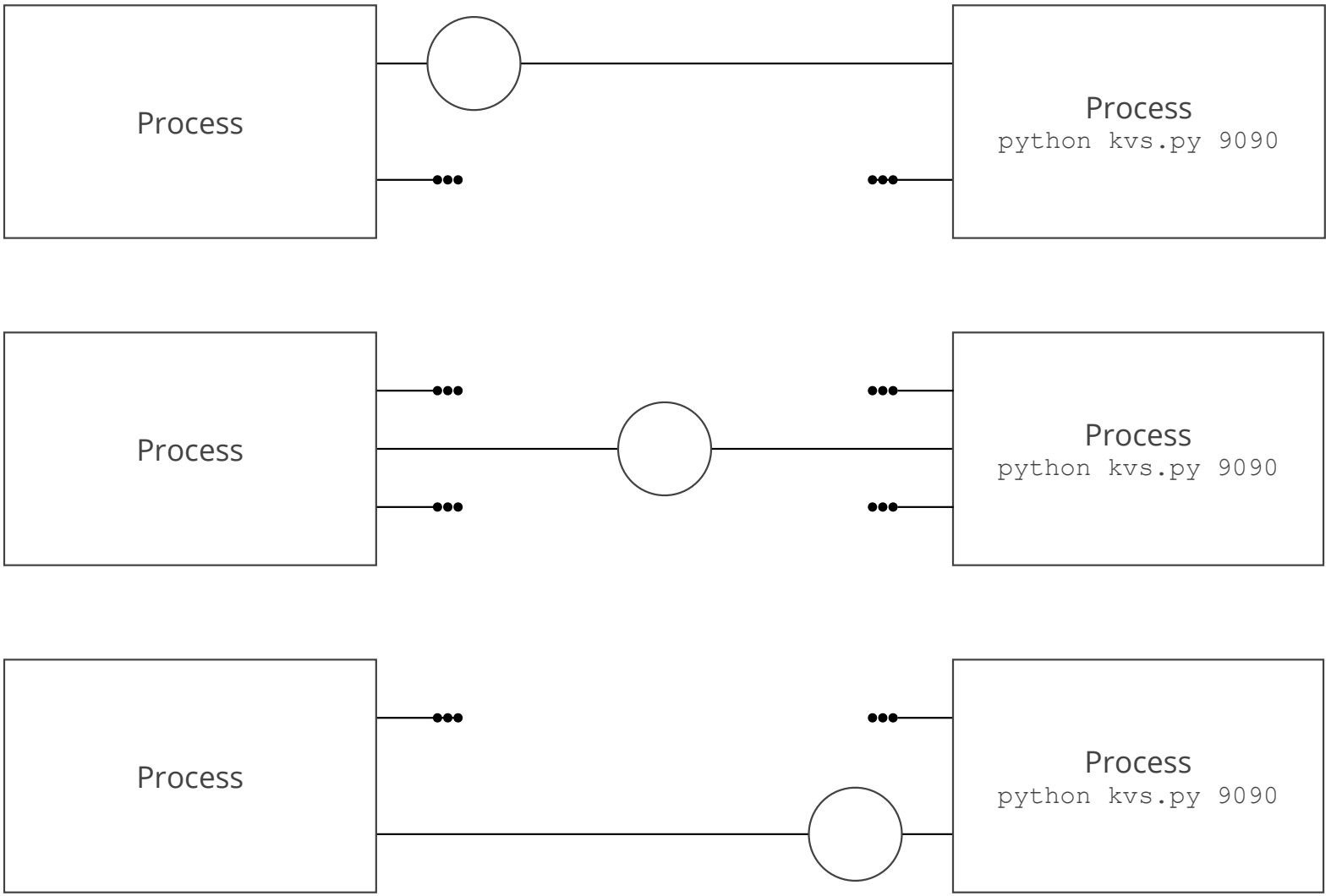
```python
app = Flask(__name__)

kvs = {}

@app.route('/<int:k>', methods=['GET', 'PUT'])
def handler(k):
    if request.method == "GET":
        return Response(kvs.get(k), status=200)
    if request.method == "PUT":
        # GOTCHA! No more than 1000 keys
        if 1000 <= len(kvs) and k not in kvs:
            return Response(status=503)
        else:
            kvs[k] = request.get_data()
            return Response(status=201)

if __name__ == '__main__':
    app.run(port=sys.argv[1])
```

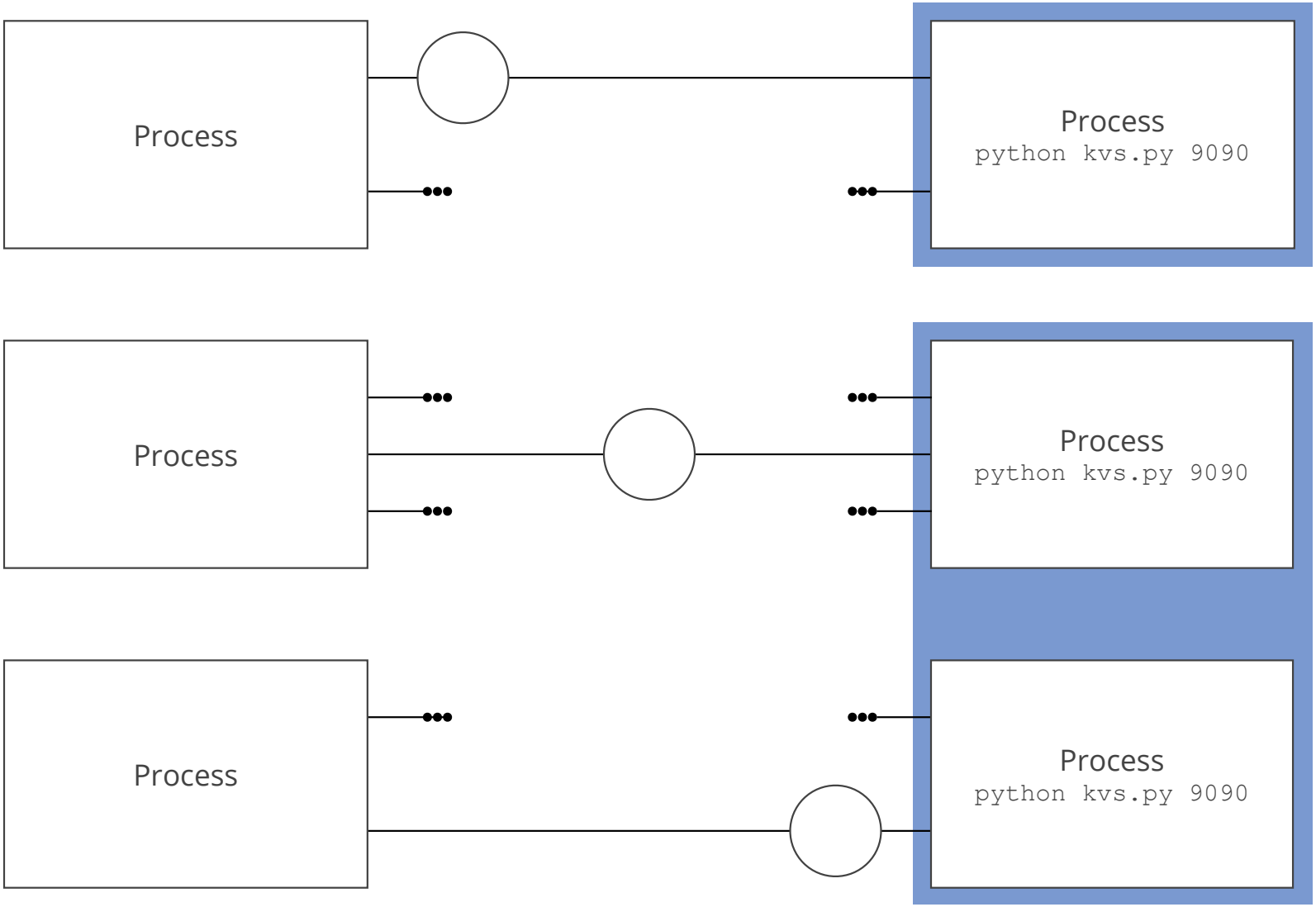
```python
app = Flask(__name__)

kvs = {}

@app.route('/<int:k>', methods=['GET', 'PUT'])
def handler(k):
    if request.method == "GET":
        return Response(kvs.get(k), status=200)
    if request.method == "PUT":
        # GOTCHA! No more than 1000 keys
        if 1000 <= len(kvs) and k not in kvs:
            return Response(status=503)
        else:
            kvs[k] = request.get_data()
            return Response(status=201)

if __name__ == '__main__':
    app.run(port=sys.argv[1])
```

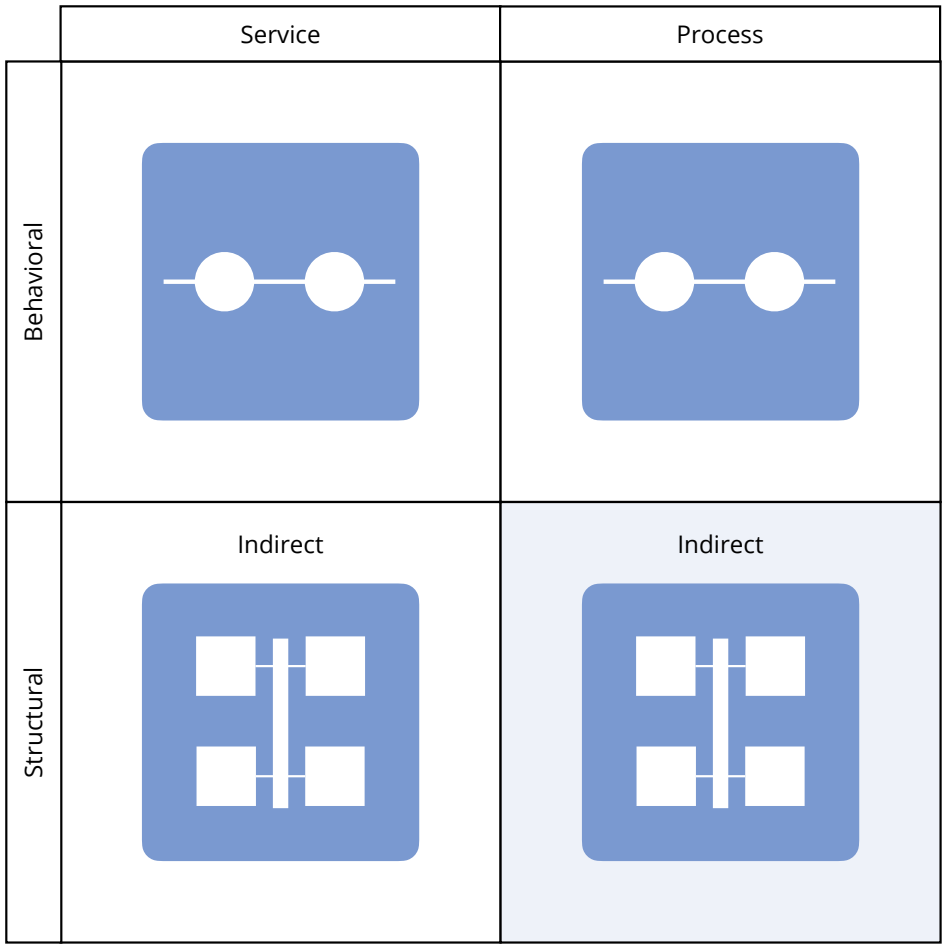
```python
app = Flask(__name__)

kvs = {}

@app.route('/<int:k>', methods=['GET', 'PUT'])
def handler(k):
    if request.method == "GET":
        return Response(kvs.get(k), status=200)
    if request.method == "PUT":
        # GOTCHA! No more than 1000 keys
        if 1000 <= len(kvs) and k not in kvs:
            return Response(status=503)
        else:
            kvs[k] = request.get_data()
            return Response(status=201)

if __name__ == '__main__':
    app.run(port=sys.argv[1])
```

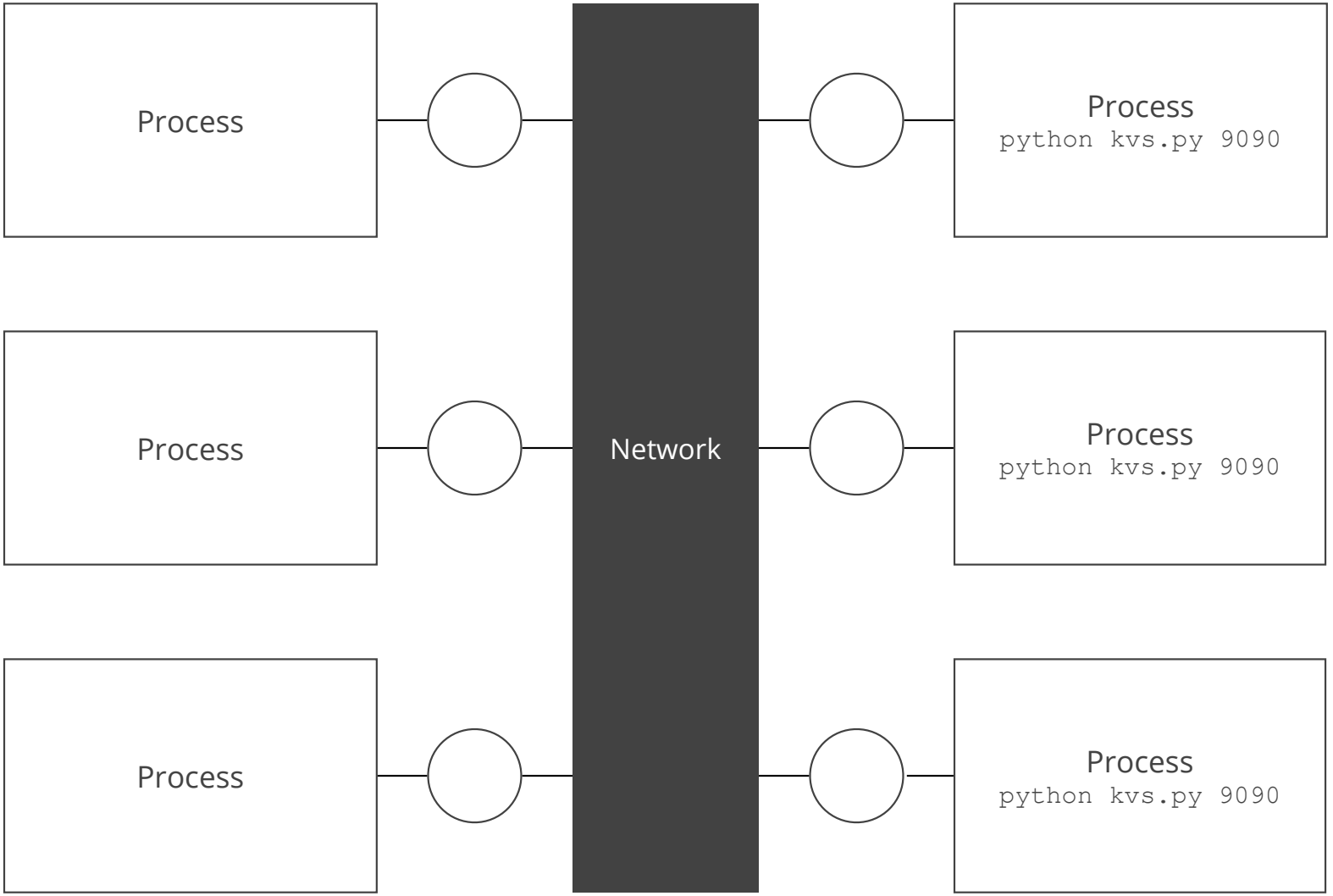

Service
Process
Behavioral
Structural
Direct
Direct

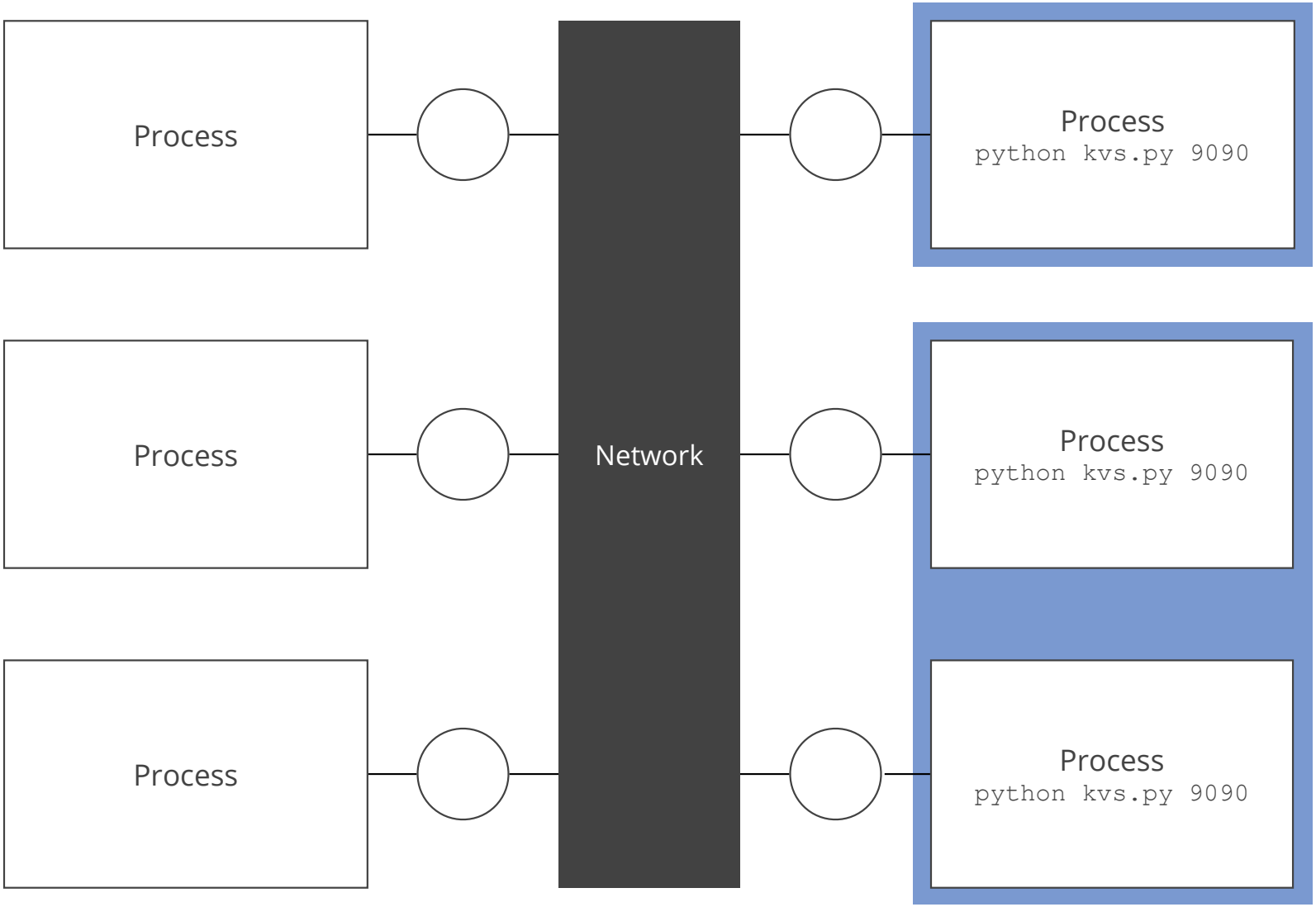
Process to Process • Structural • Direct Communication

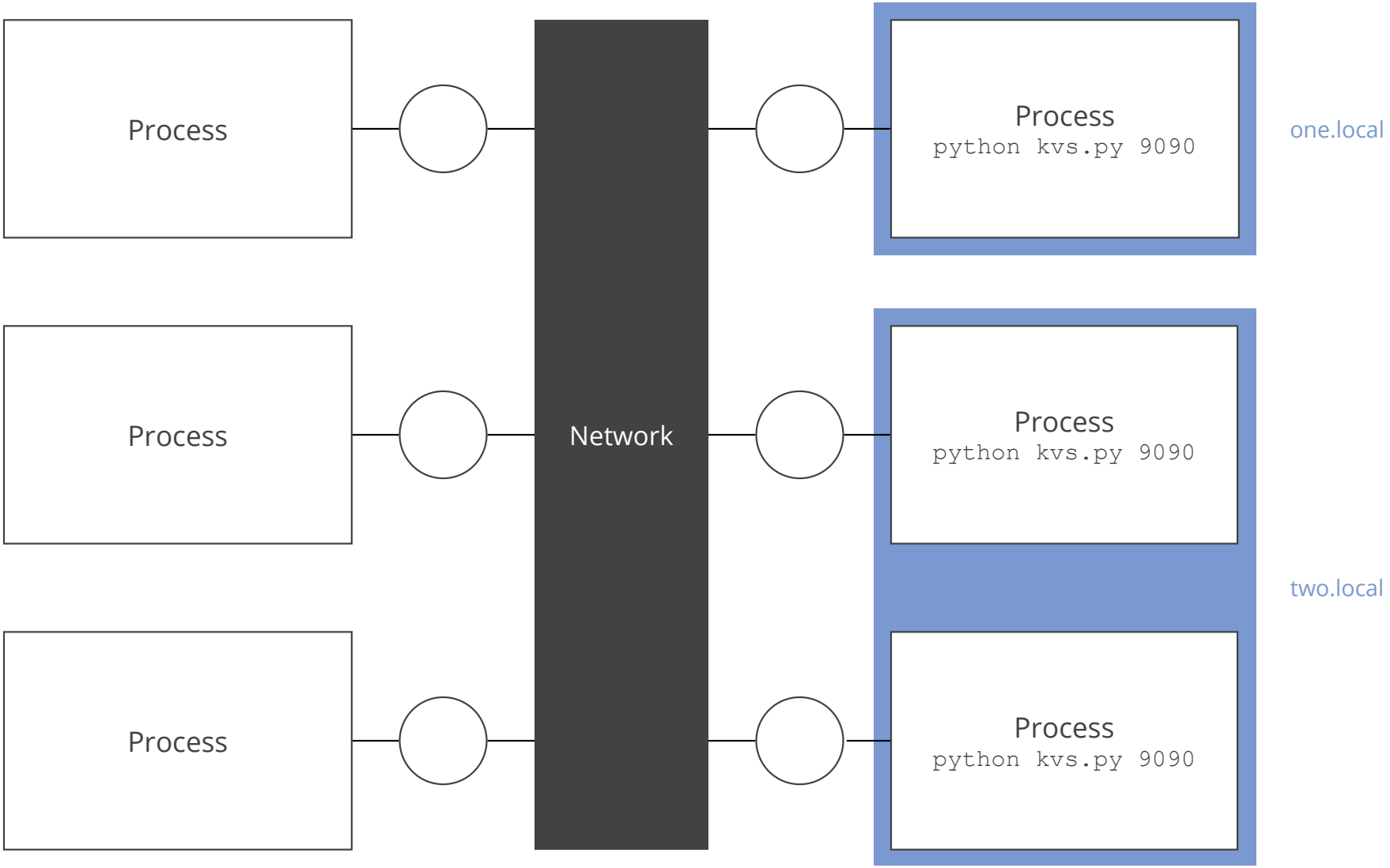


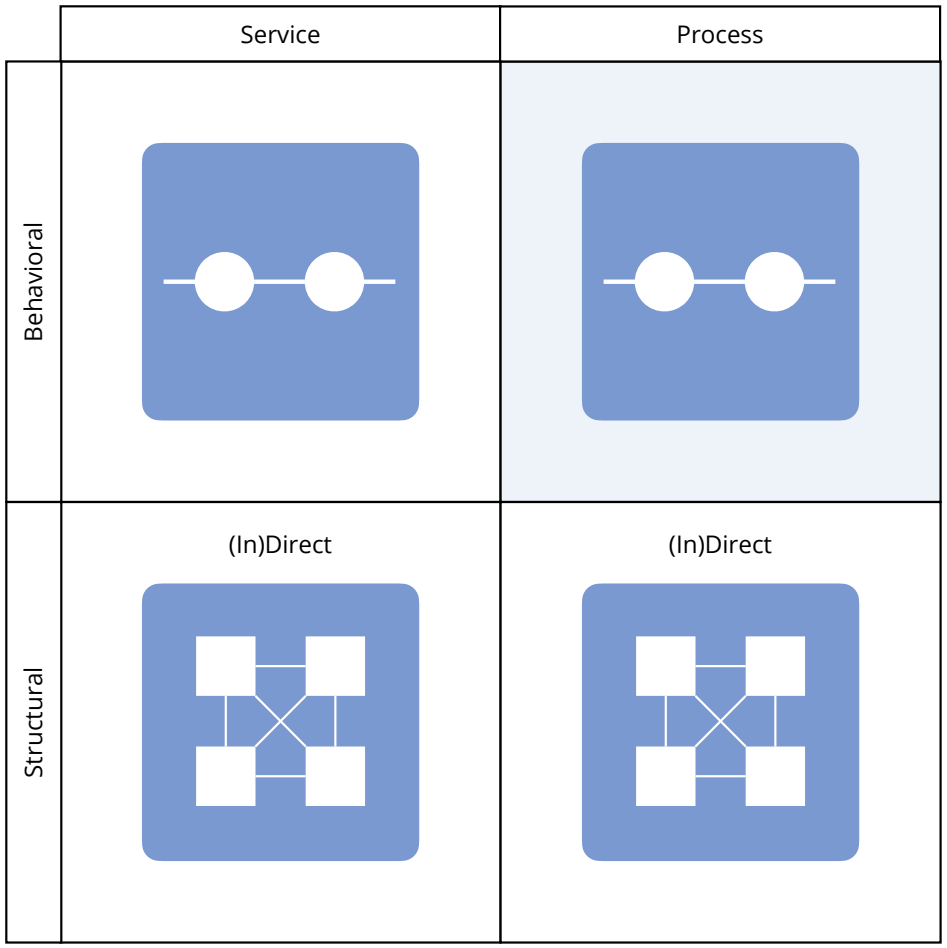

Process
Process
python kvs.py 9090
Process
Process
python kvs.py 9090
Process
Process
python kvs.py 9090

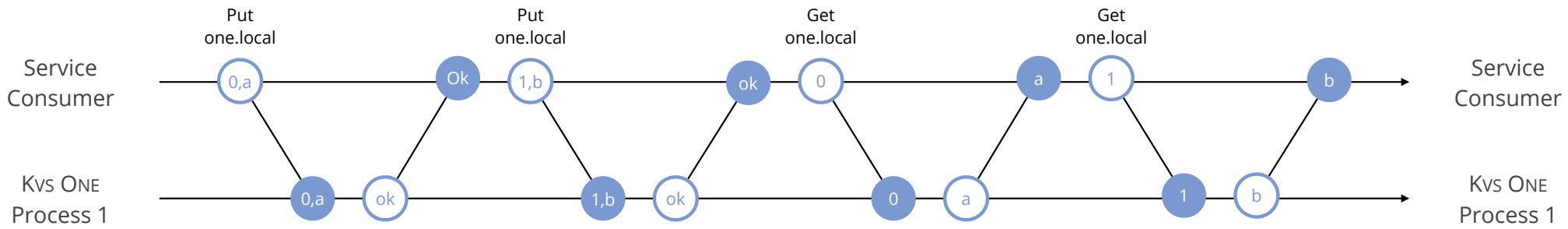

Service
Process
Behavioral
Structural
Indirect
Indirect

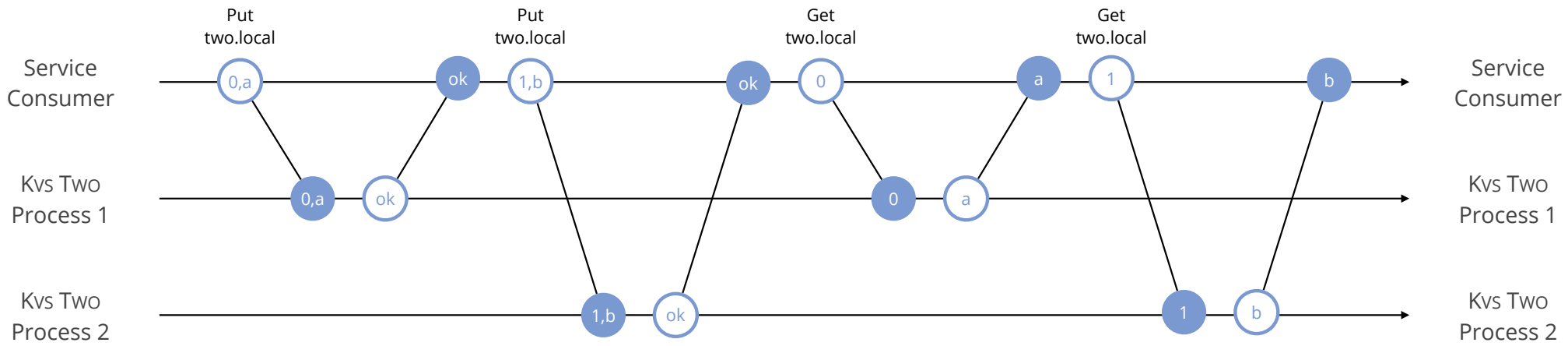

Process
Process
Process
Network
Process
python kvs.py 9090
Process
python kvs.py 9090
Process
python kvs.py 9090

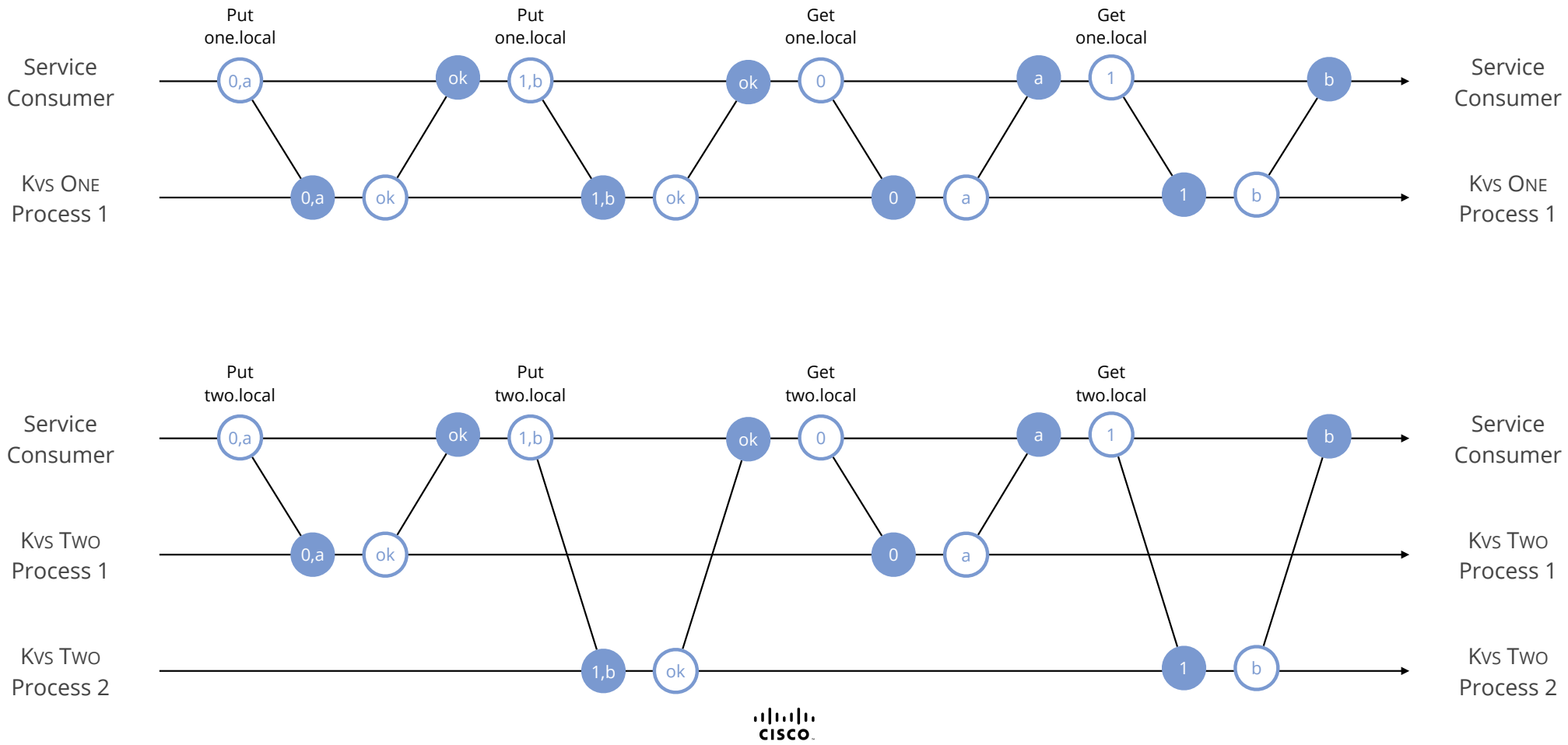

Process
Process
Process
Network
Process
python kvs.py 9090
Process
python kvs.py 9090
Process
python kvs.py 9090


Process
Process
Process
Network
Process
python kvs.py 9090
Process
python kvs.py 9090
Process
python kvs.py 9090
one.local
two.local


Service
Process
Behavioral
Structural
(In)Direct
(In)Direct

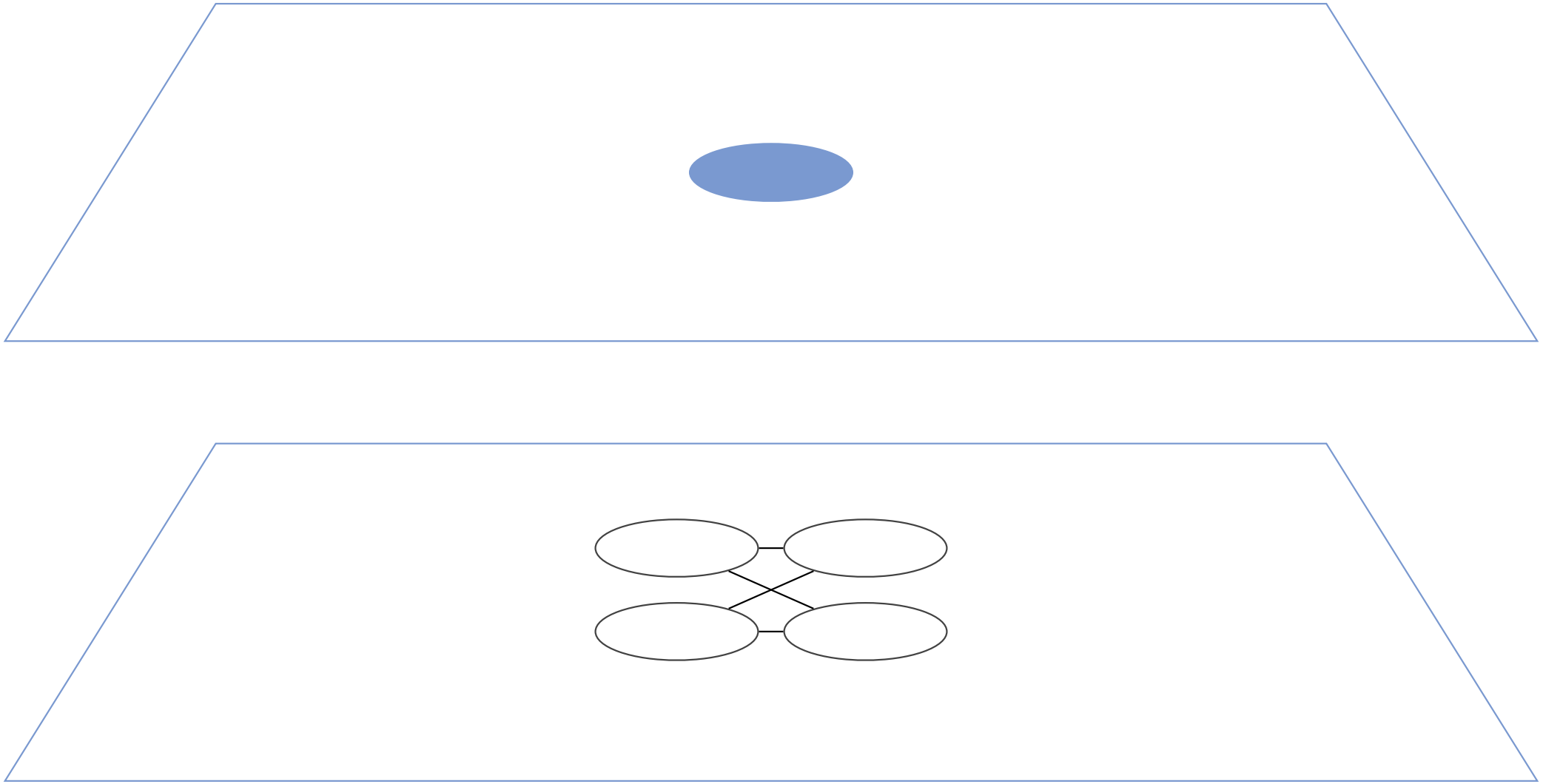

Put
one.local
Put
one.local
Get
one.local
Get
one.local
Service
Consumer
0,a
Ok
1,b
ok
0
a
1
b
Service
Consumer
KVS ONE
Process 1
0,a
ok
1,b
ok
0
a
1
b
KVS ONE
Process 1

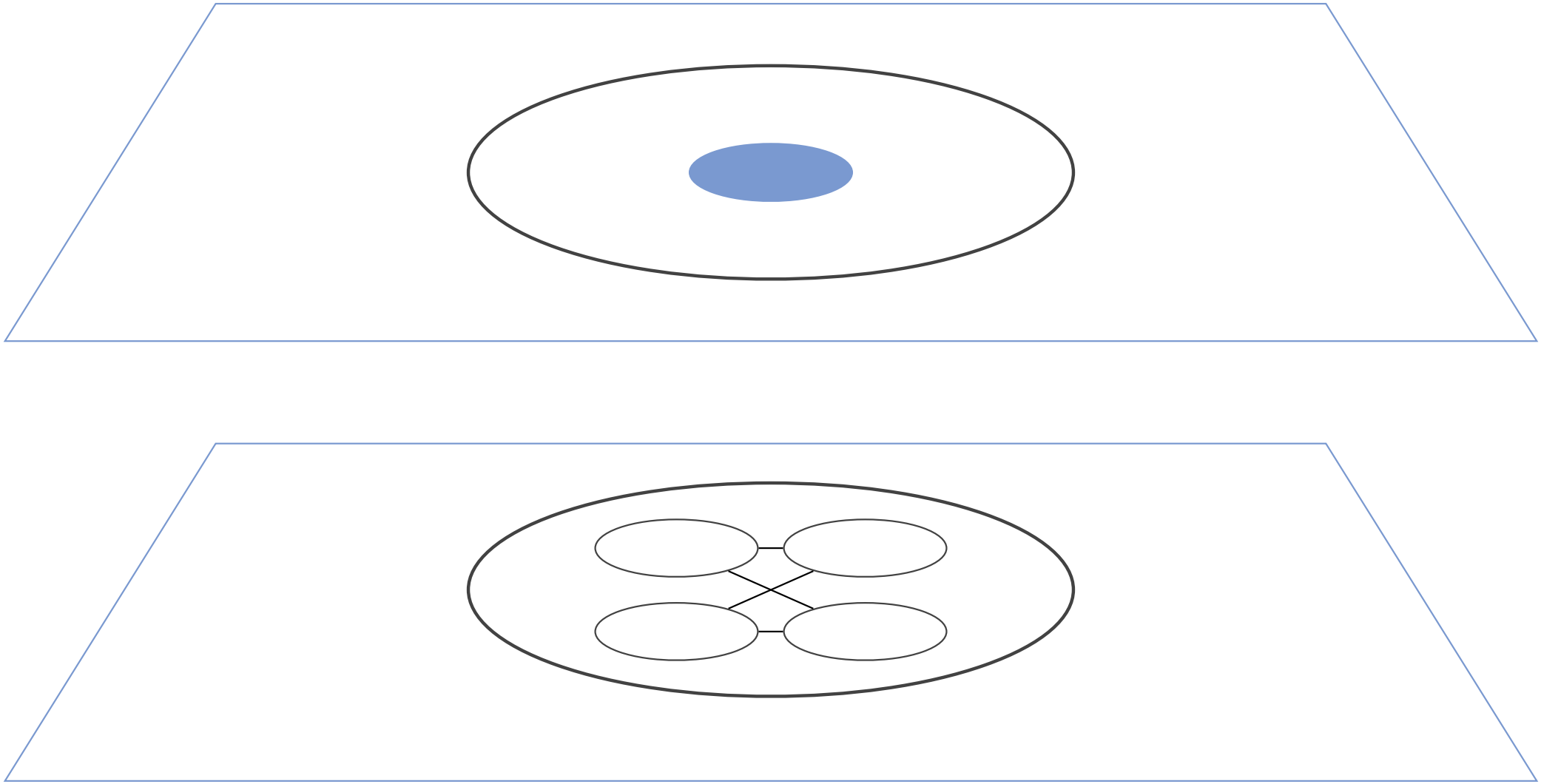

Put
two.local
Put
two.local
Get
two.local
Get
two.local
Service
Consumer
Kvs Two
Process 1
Kvs Two
Process 2
0,a
ok
1,b
ok
0
a
1
b
0,a
ok
0
a
1,b
ok
1
b
Service
Consumer
Kvs Two
Process 1
Kvs Two
Process 2


Put
one.local
Put
one.local
Get
one.local
Get
one.local
Service
Consumer
0,a
ok
1,b
ok
0
a
1
b
Service
Consumer
KVS ONE
Process 1
0,a
ok
1,b
ok
0
a
1
b
KVS ONE
Process 1
Put
two.local
Put
two.local
Get
two.local
Get
two.local
Service
Consumer
0,a
ok
1,b
ok
0
a
1
b
Service
Consumer
KVS TWO
Process 1
0,a
ok
0
a
KVS TWO
Process 1
KVS TWO
Process 2
1,b
ok
1
b
KVS TWO
Process 2

Service Model

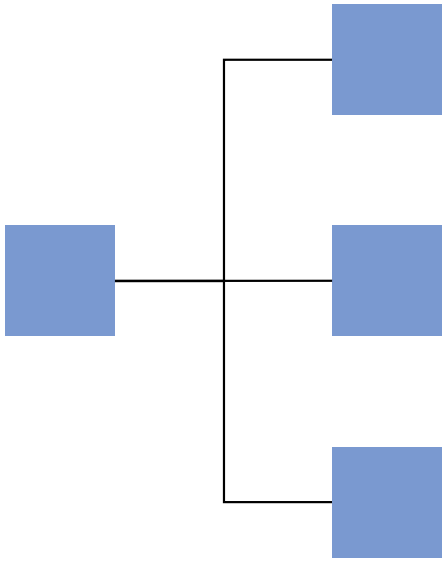
Summary

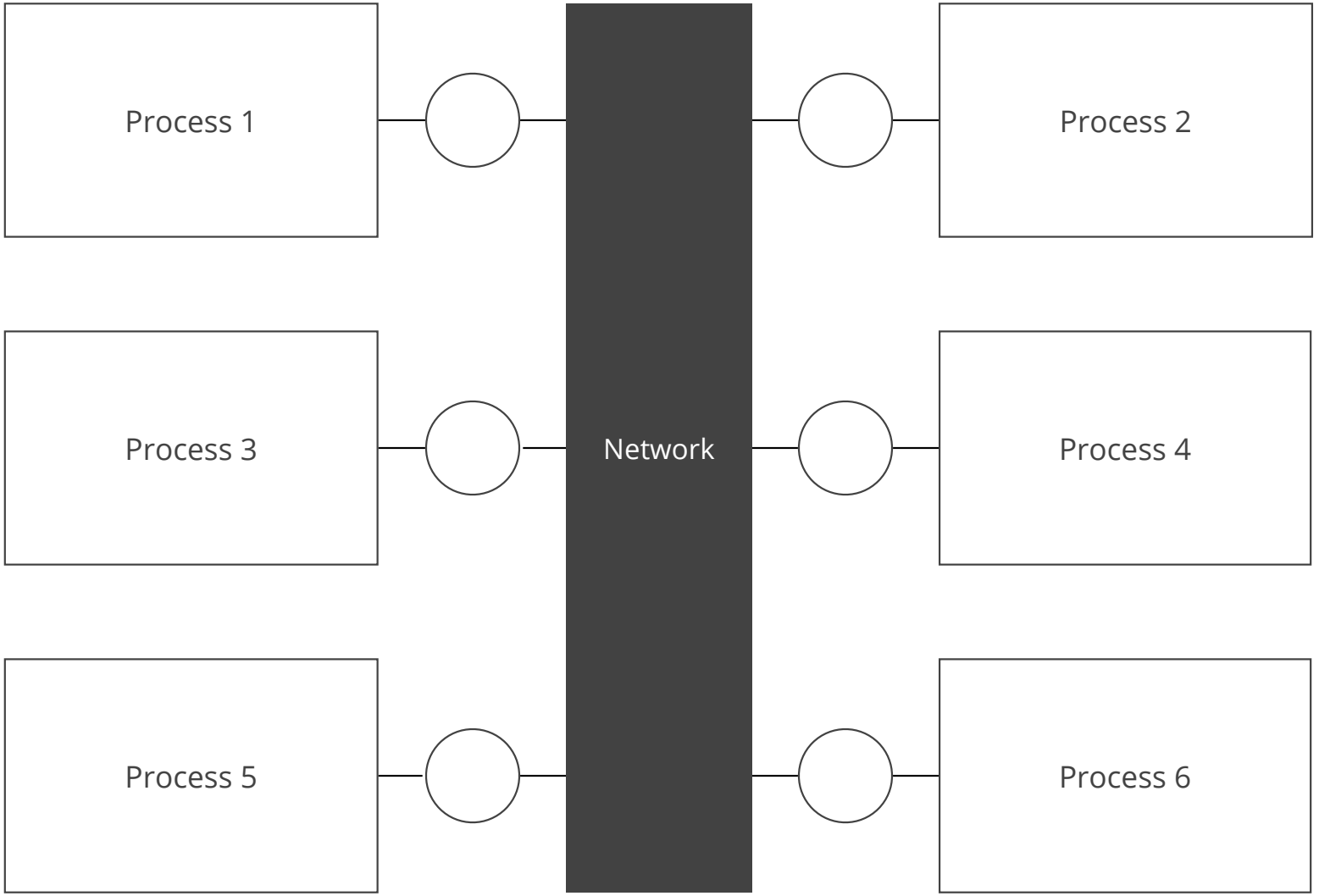


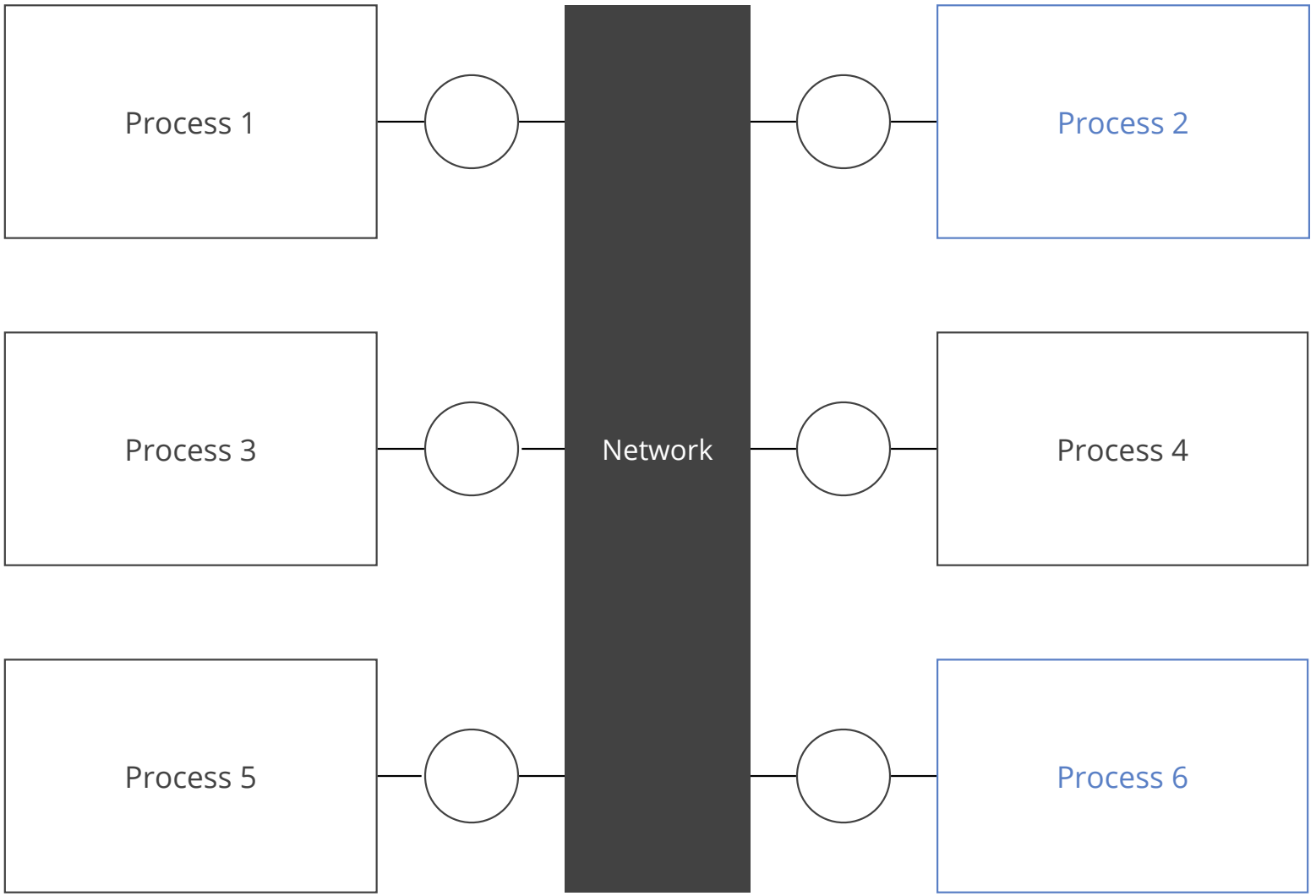


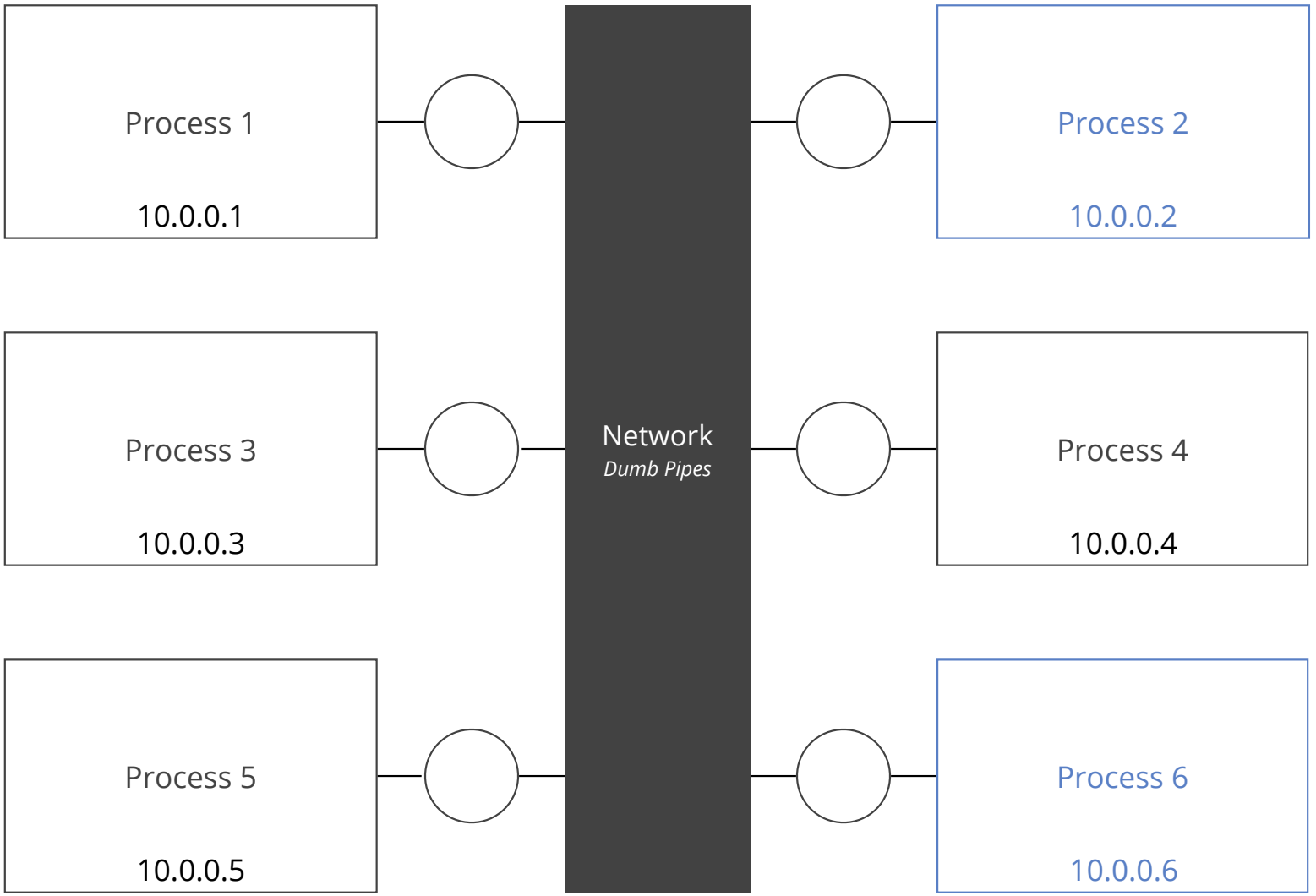
Service Mesh

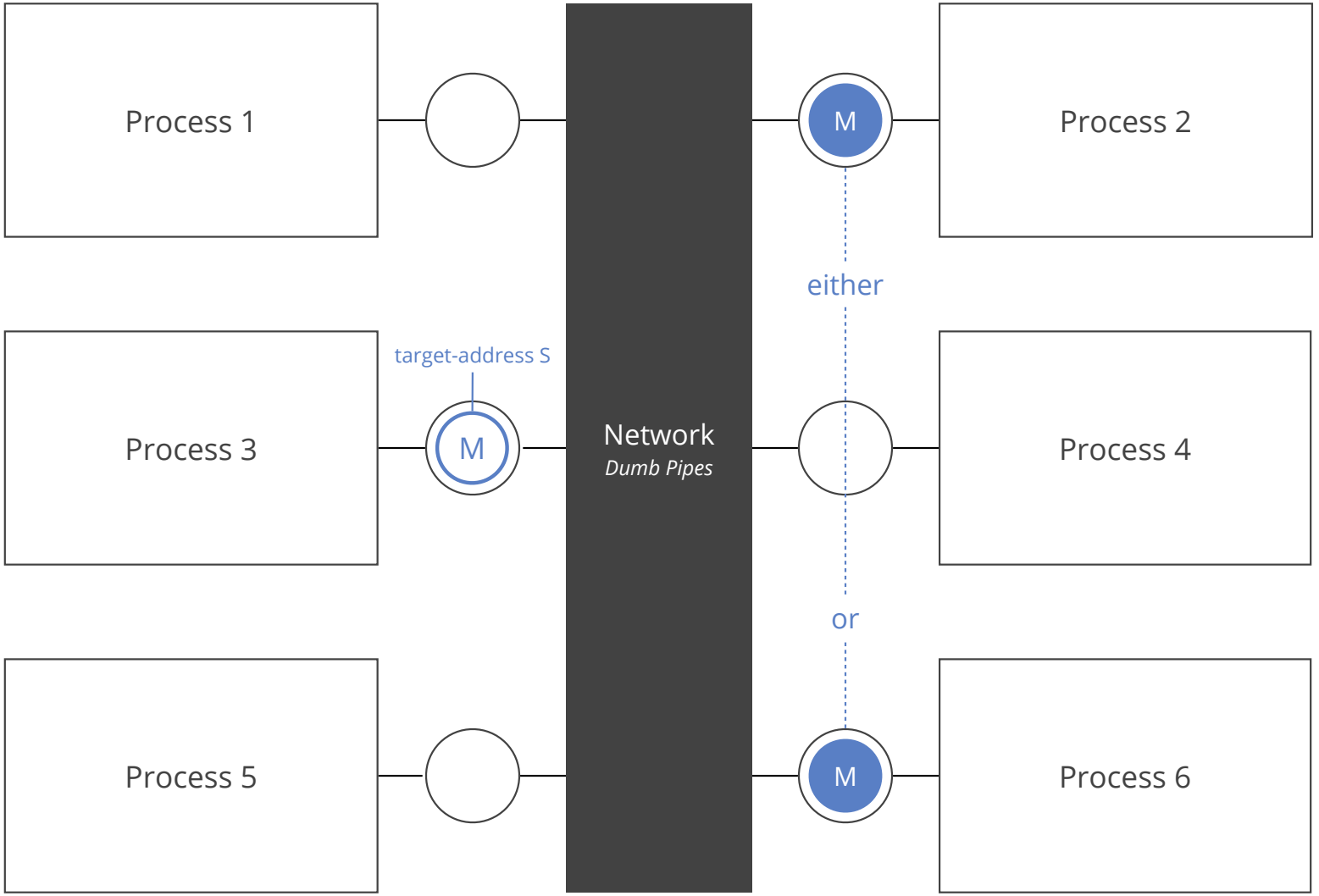
Overview



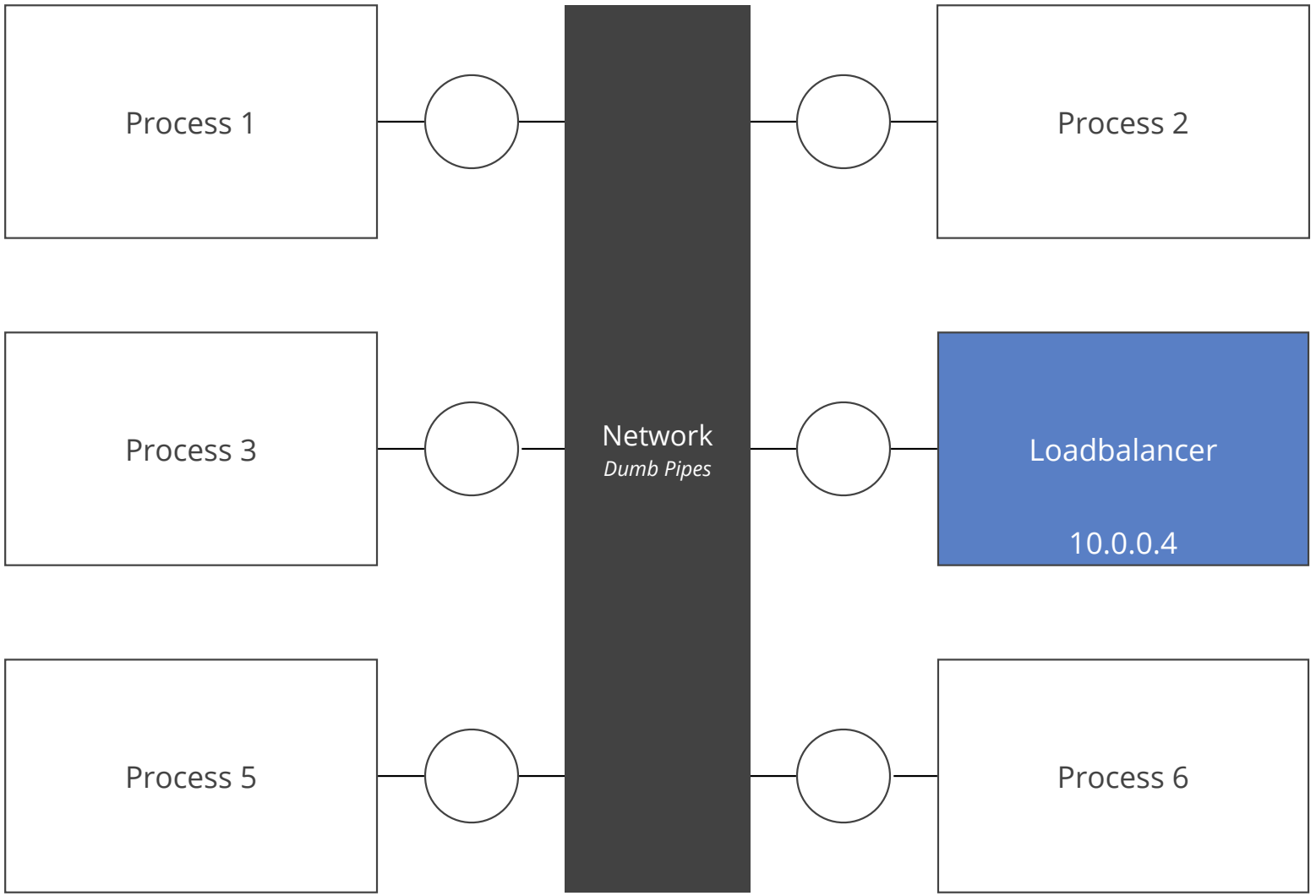

Process 1
Process 2
Process 3
Network
Process 4
Process 5
Process 6

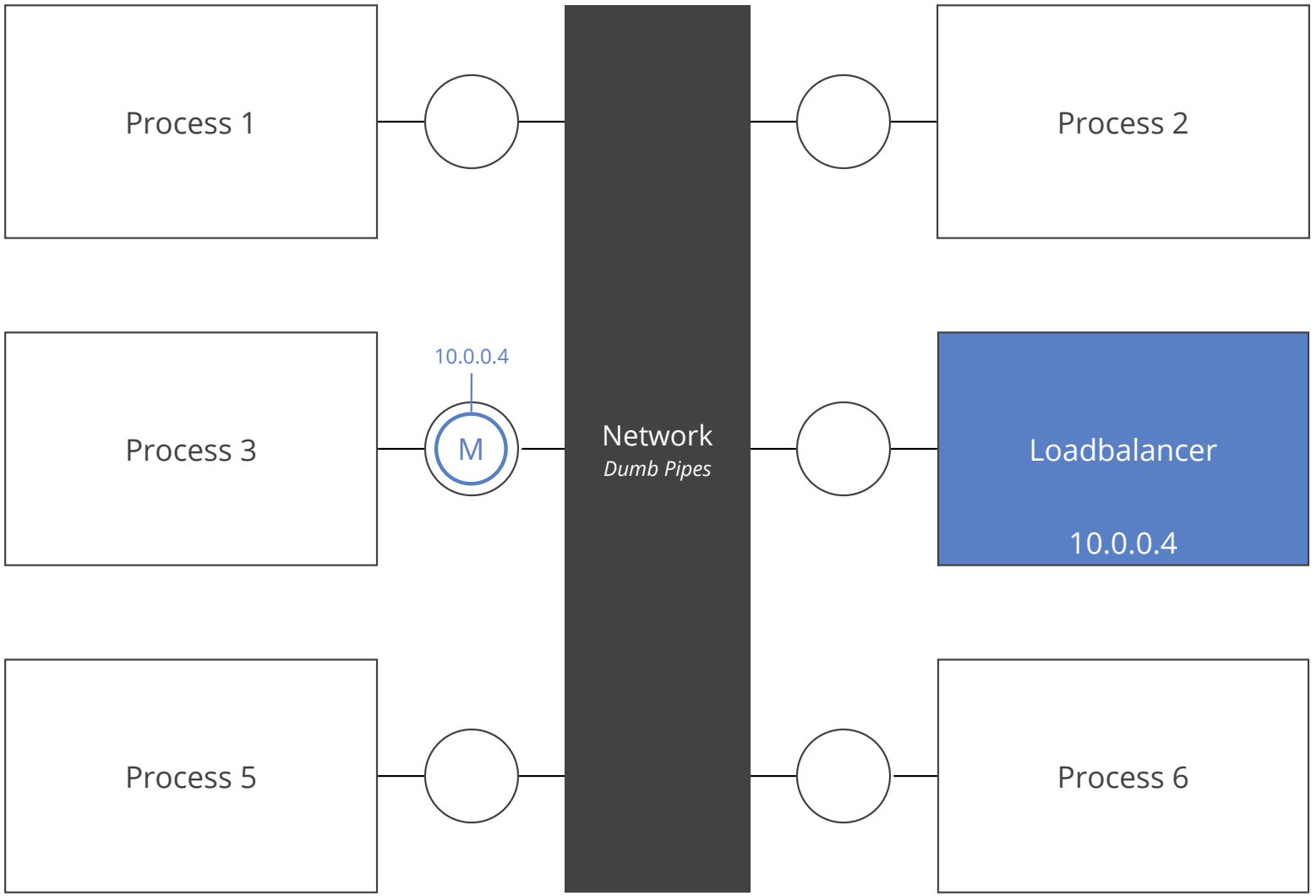

Process 1
Process 2
Process 3
Network
Process 4
Process 5
Process 6

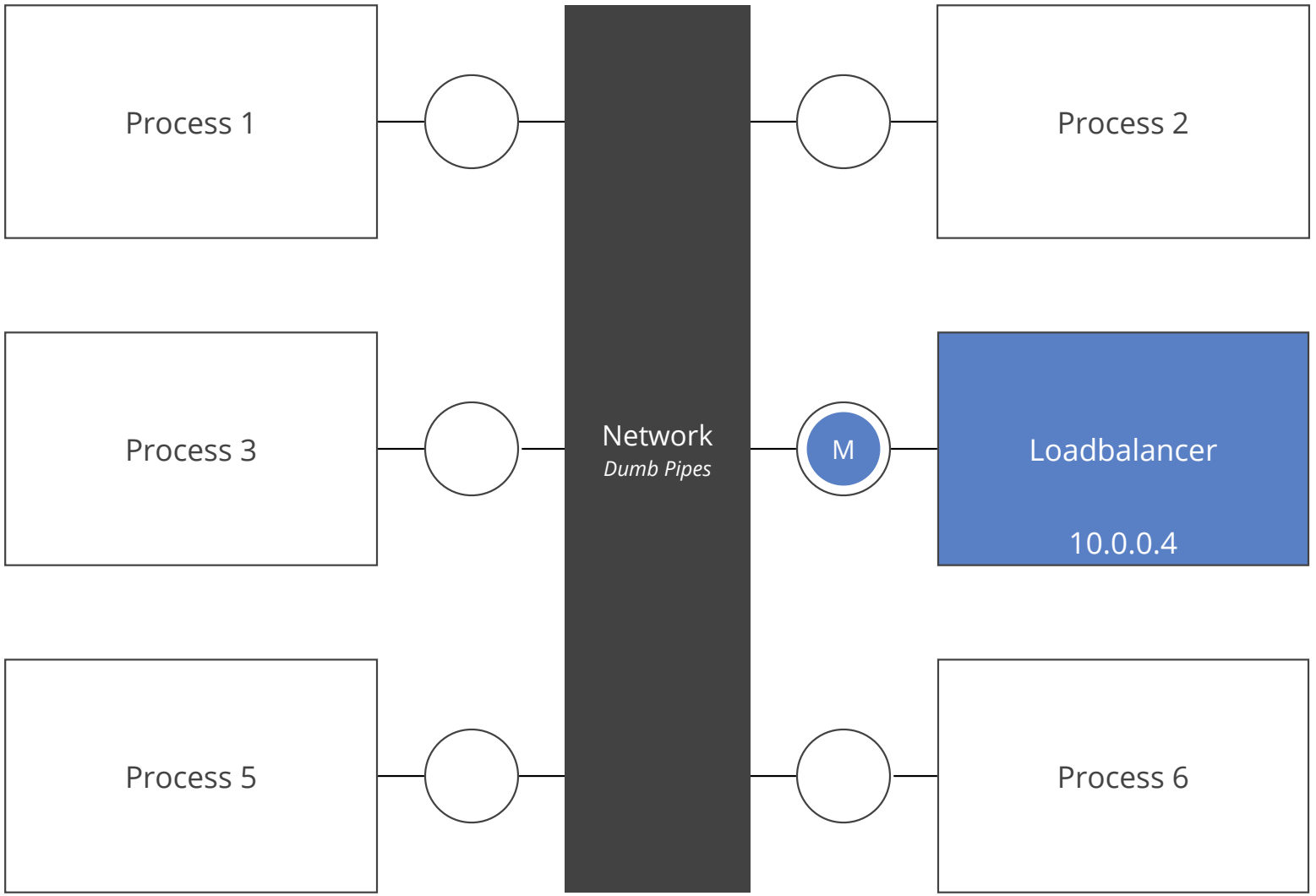

Process 1
10.0.0.1
Process 2
10.0.0.2
Process 3
10.0.0.3
Network
Dumb Pipes
Process 4
10.0.0.4
Process 5
10.0.0.5
Process 6
10.0.0.6

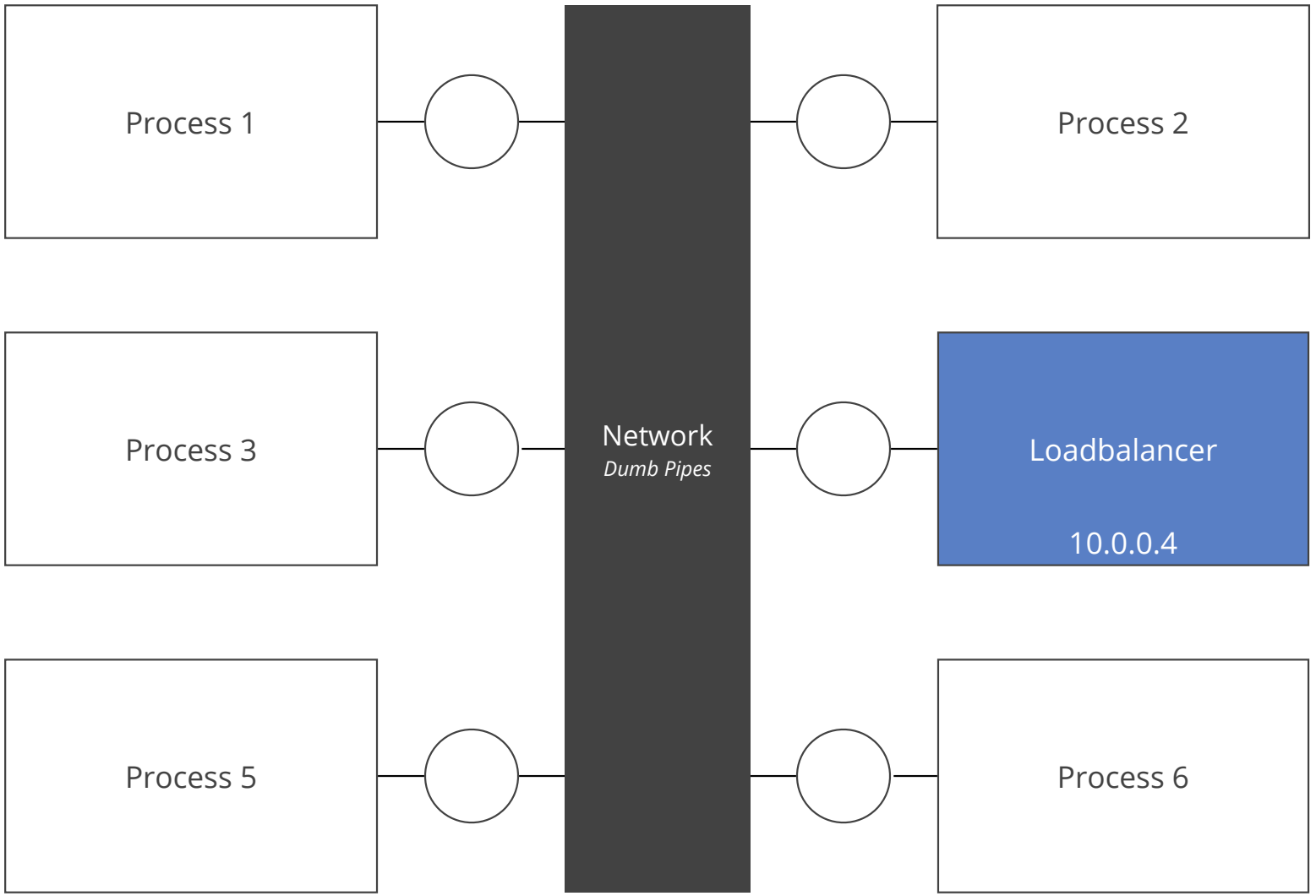

Process 1
Process 2
M
either
target-address S
Process 3
M
Network
Dumb Pipes
Process 4
or
Process 5
M
Process 6

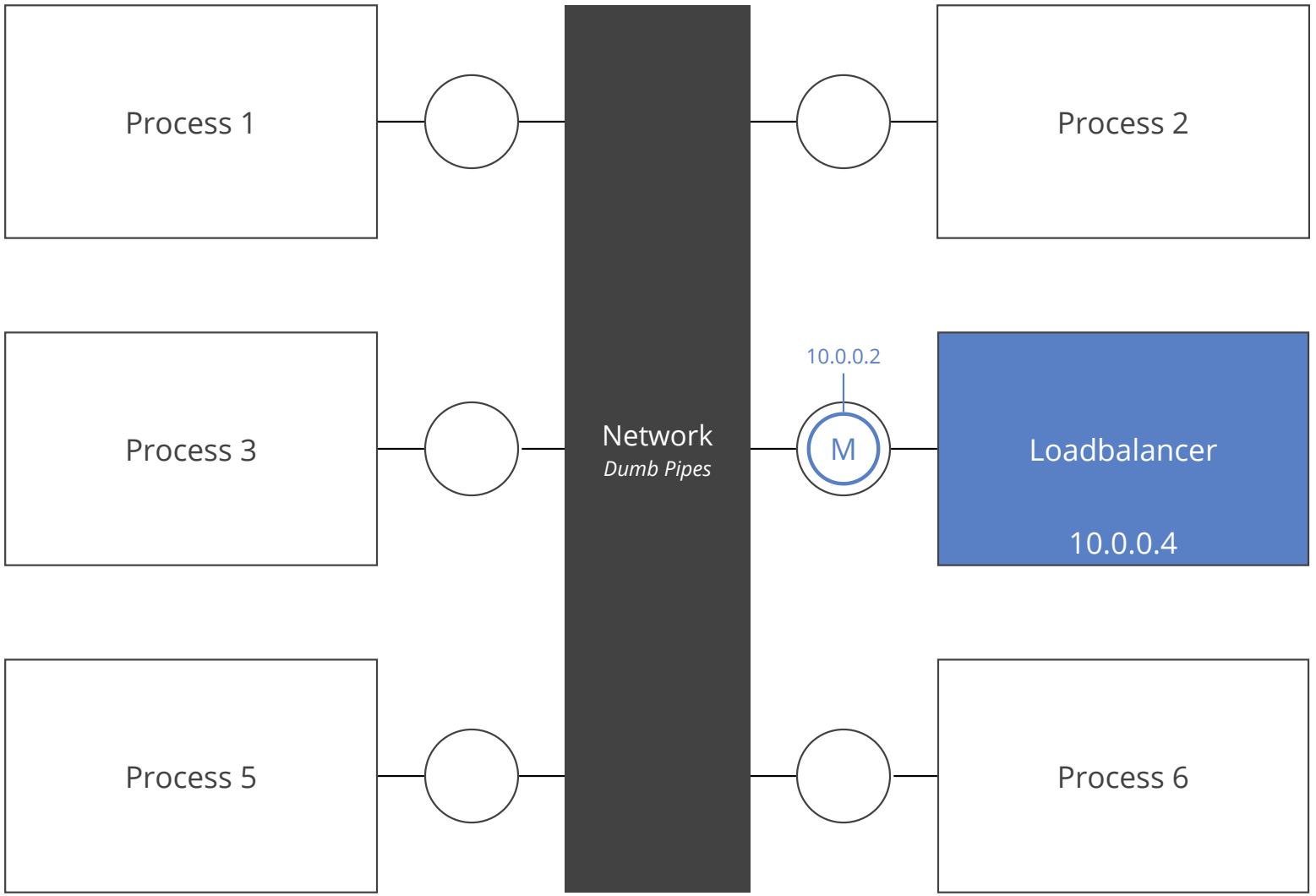
Crawl

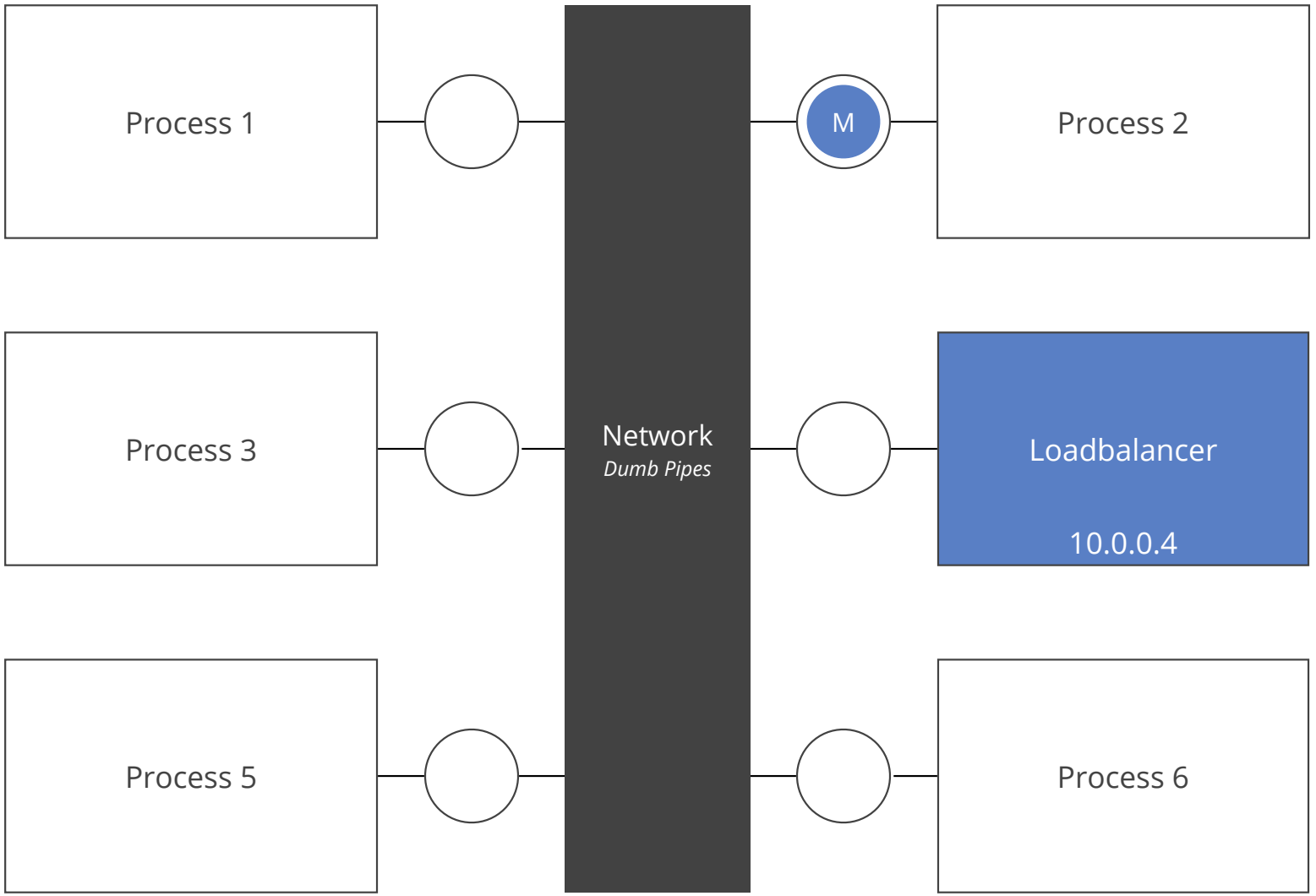

Process 1
Process 2
Process 3
Network
Dumb Pipes
Loadbalancer
10.0.0.4
Process 5
Process 6

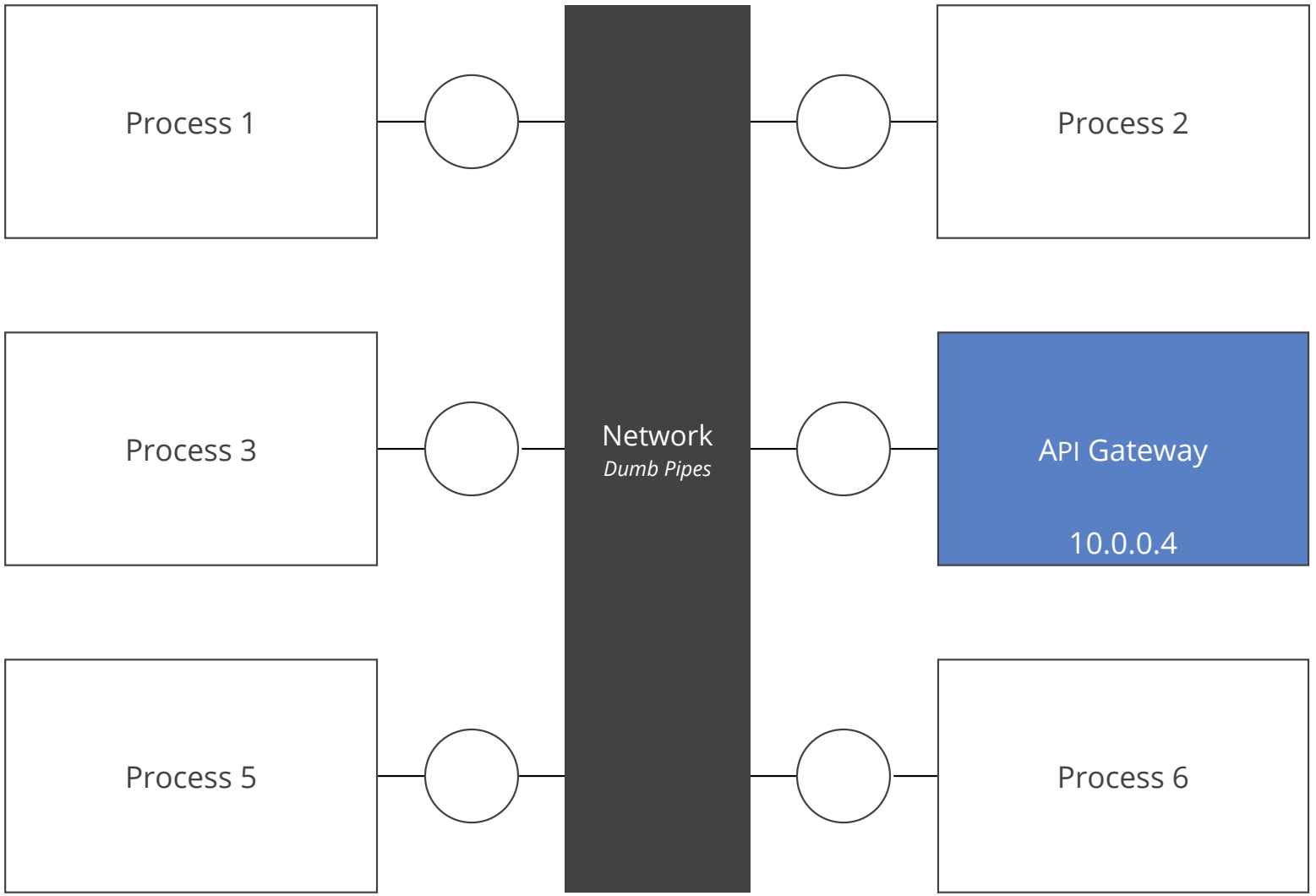

Process 1
Process 2
10.0.0.4
Process 3
M
Network
Dumb Pipes
Loadbalancer
10.0.0.4
Process 5
Process 6


Process 1
Process 2
Process 3
Network
Dumb Pipes
M
Loadbalancer
10.0.0.4
Process 5
Process 6

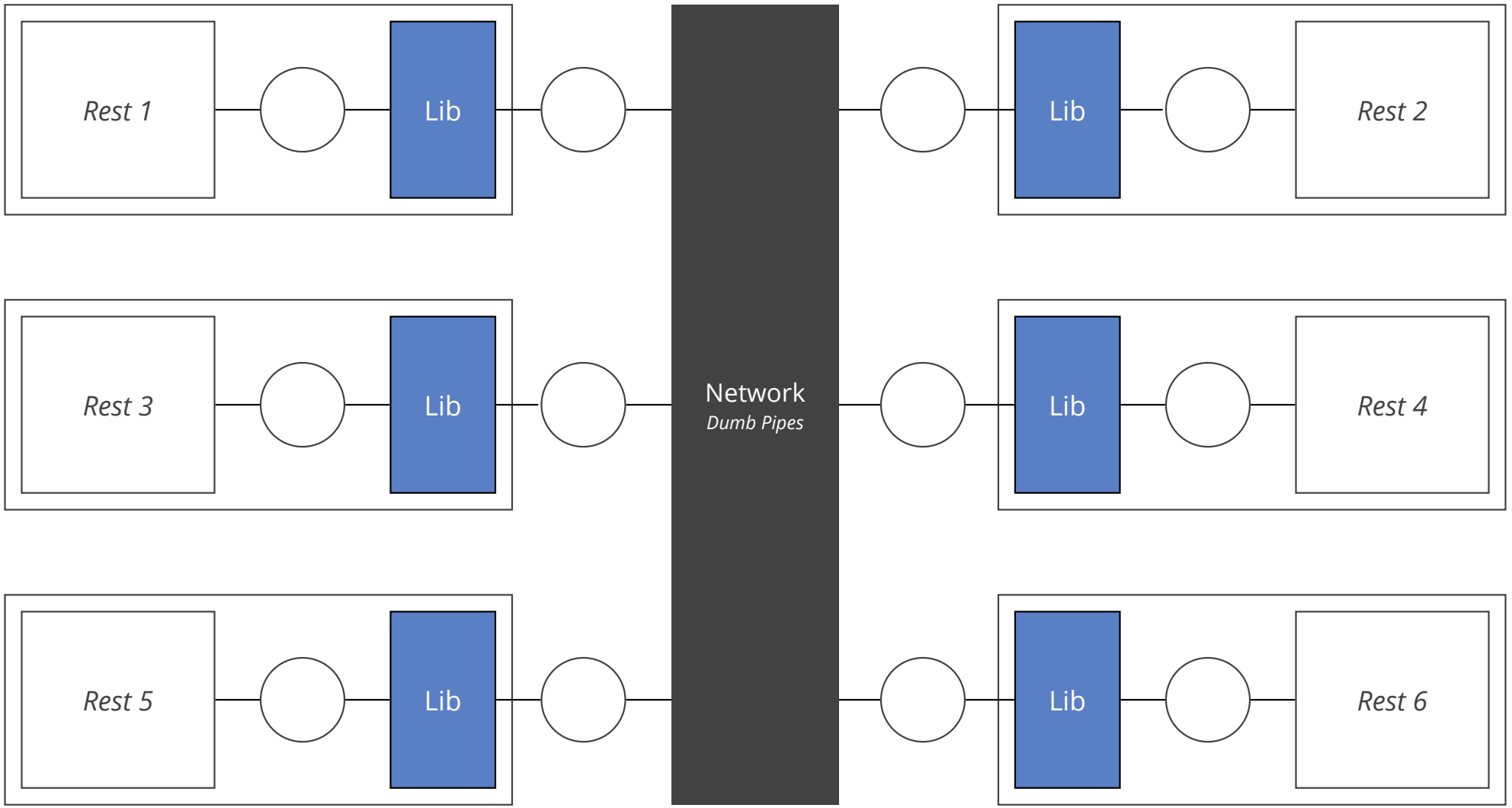

Process 1
Process 2
Process 3
Network
Dumb Pipes
Loadbalancer
10.0.0.4
Process 5
Process 6

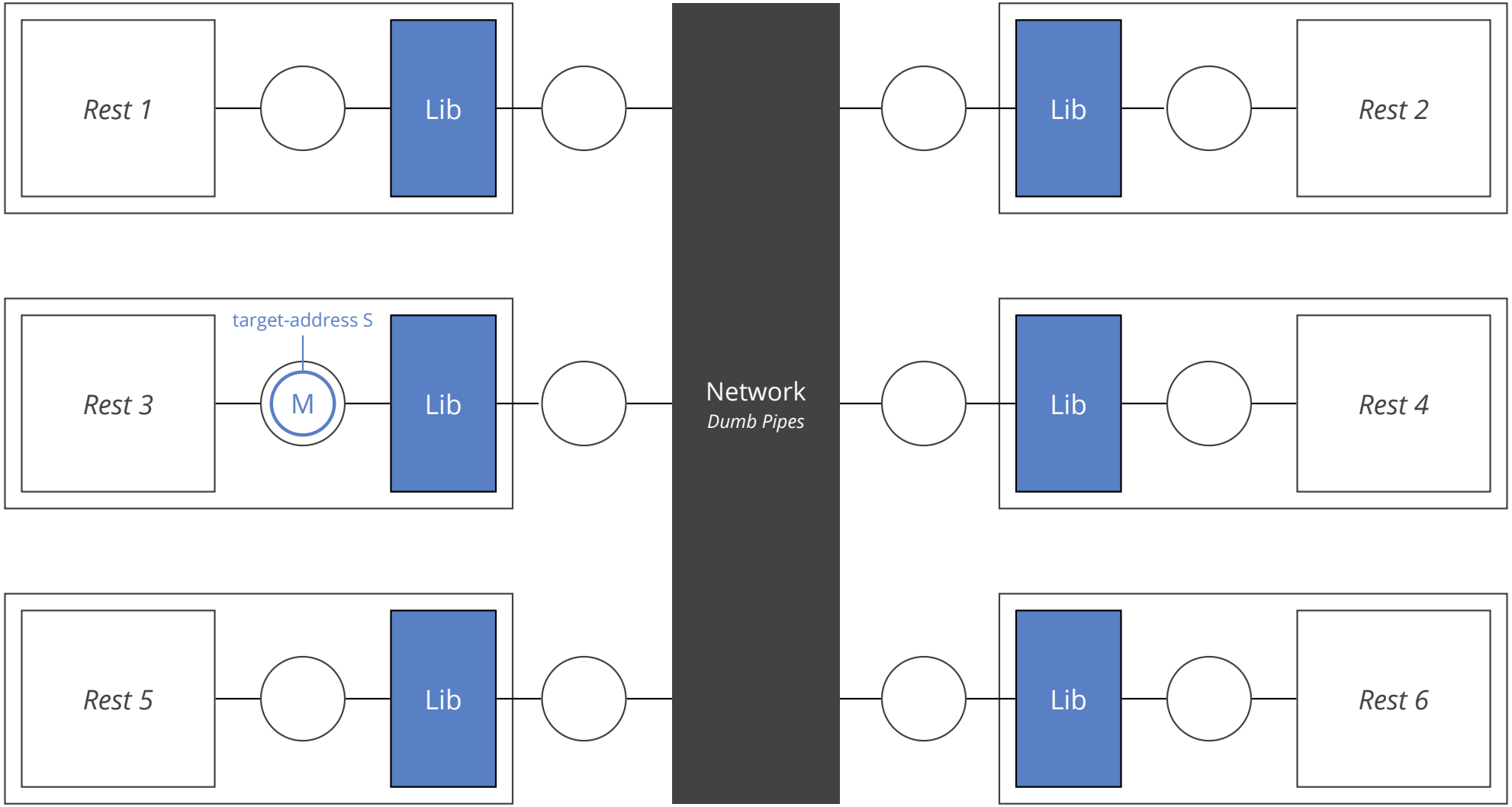

Process 1
Process 2
Process 3
Process 5
Process 6
Network
Dumb Pipes
10.0.0.2
M
Loadbalancer
10.0.0.4

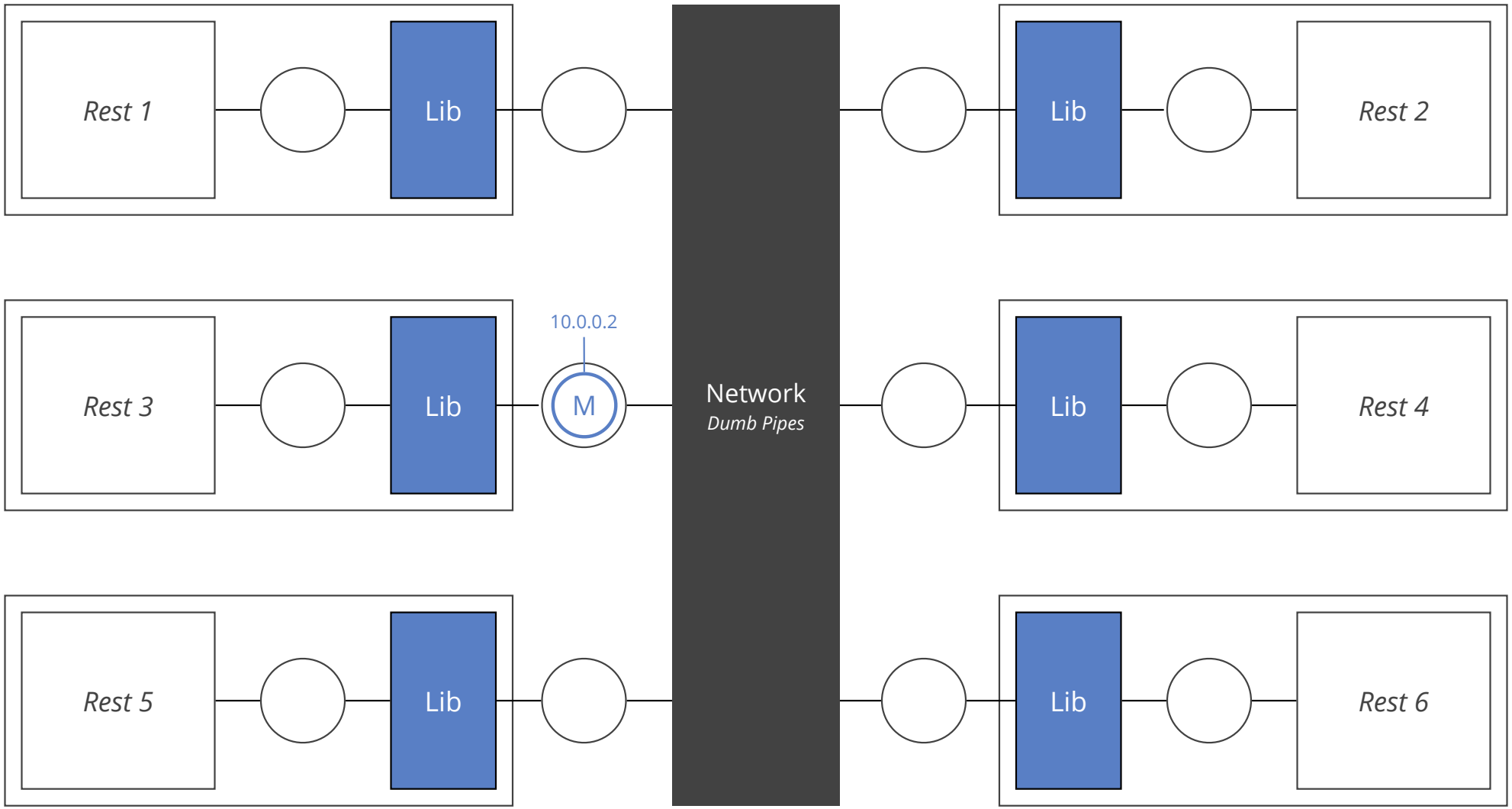

Process 1
M
Process 2
Process 3
Network
Dumb Pipes
Loadbalancer
10.0.0.4
Process 5
Process 6

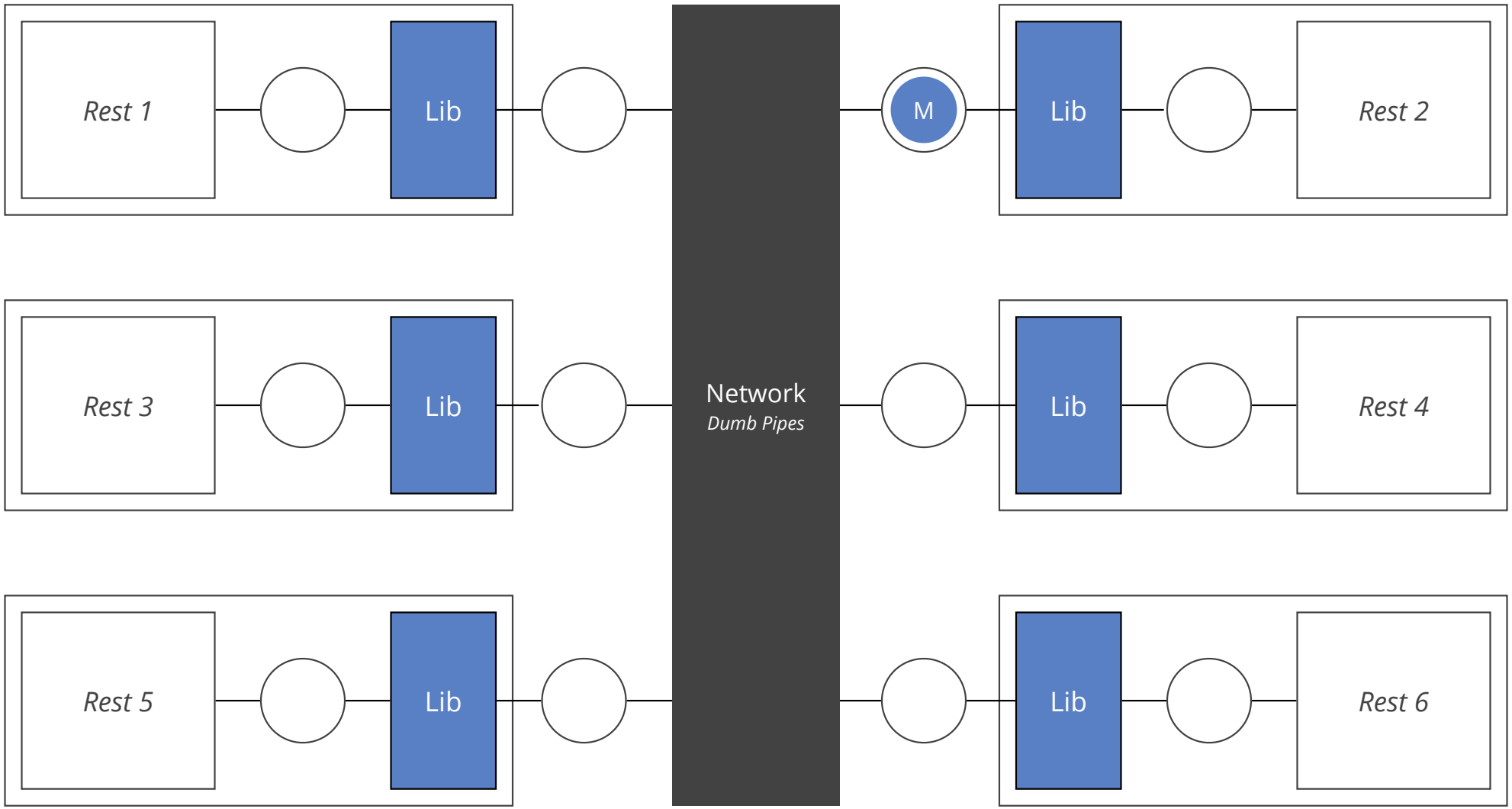

Process 1
Process 2
Process 3
Network
Dumb Pipes
API Gateway
10.0.0.4
Process 5
Process 6

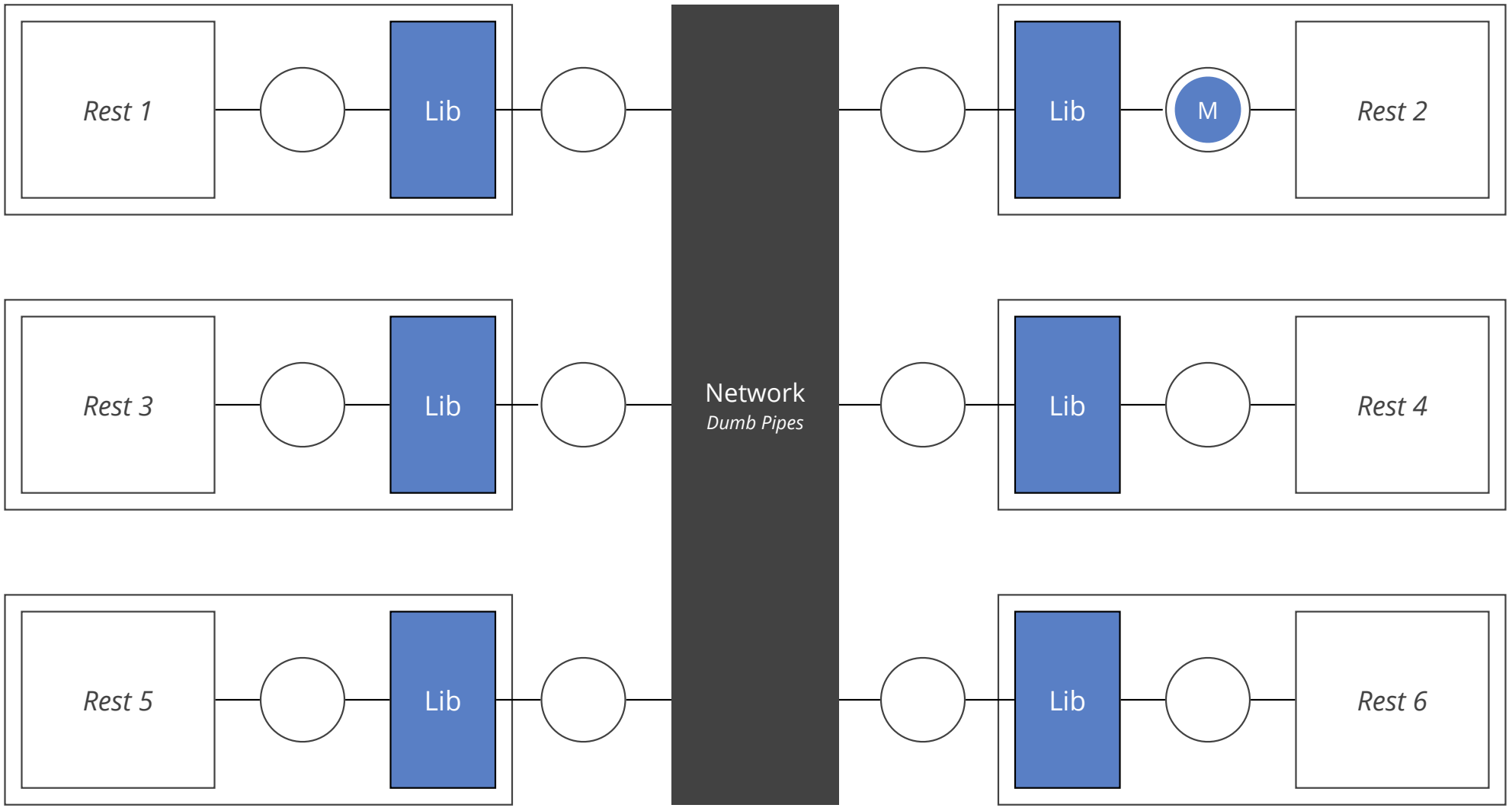
Walk

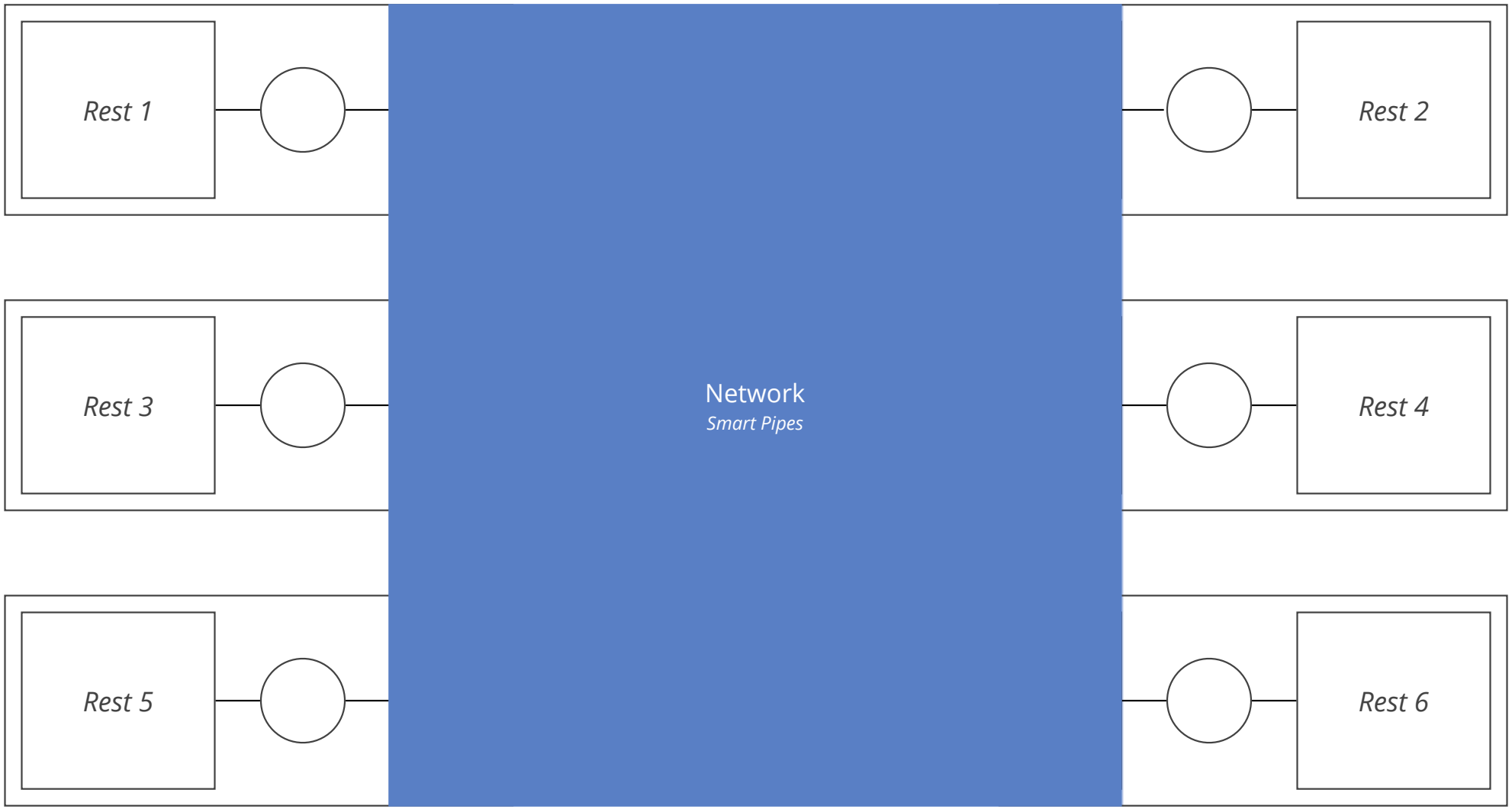

Rest 1
Lib
Lib
Rest 2
Rest 3
Lib
Network
Dumb Pipes
Lib
Rest 4
Rest 5
Lib
Lib
Rest 6


Rest 1
Lib
Lib
Rest 2
target-address S
Rest 3
M
Lib
Network
Dumb Pipes
Lib
Rest 4
Rest 5
Lib
Lib
Rest 6

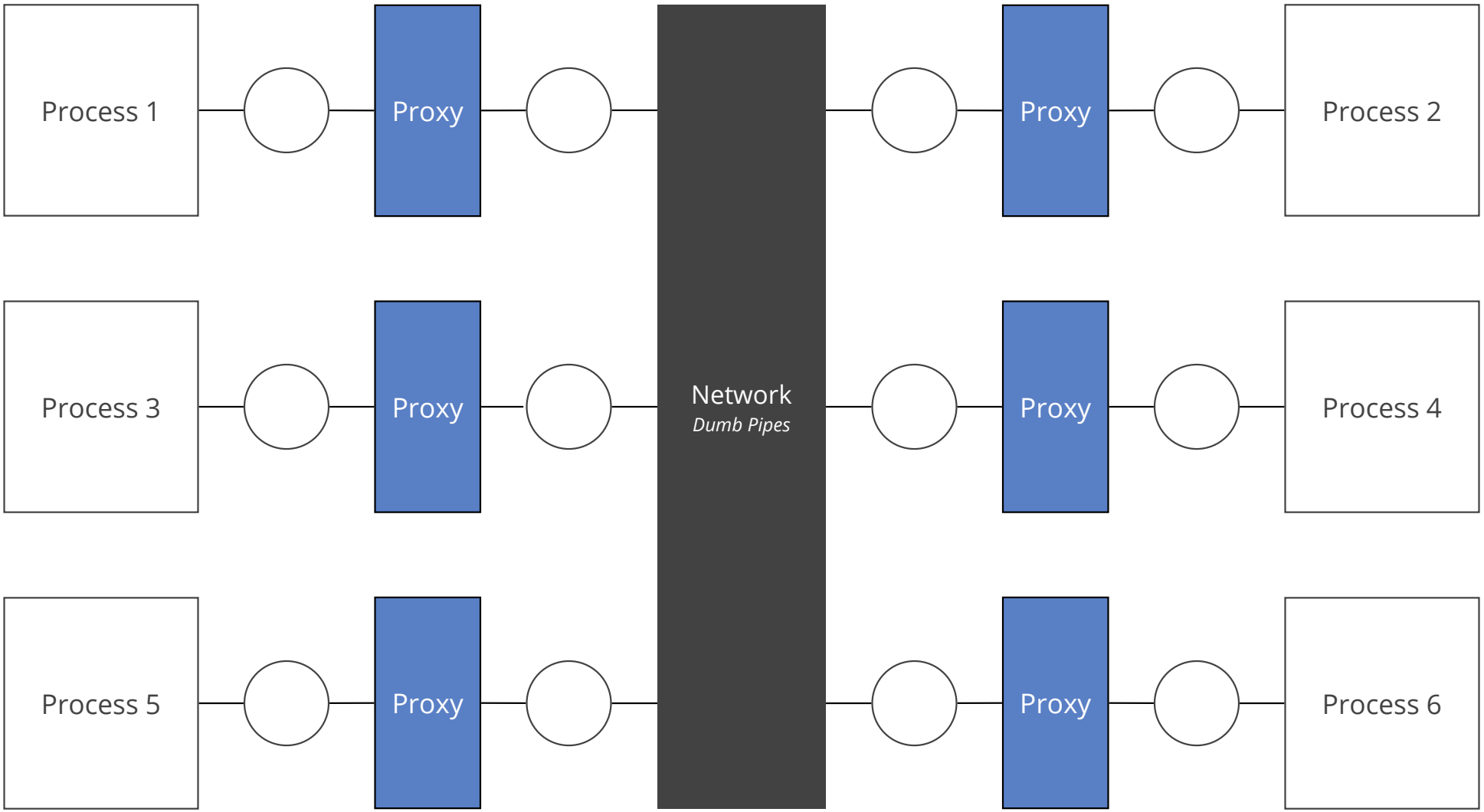

Rest 1
Lib
Lib
Rest 2
Rest 3
Lib
10.0.0.2
M
Network
Dumb Pipes
Lib
Rest 4
Rest 5
Lib
Lib
Rest 6

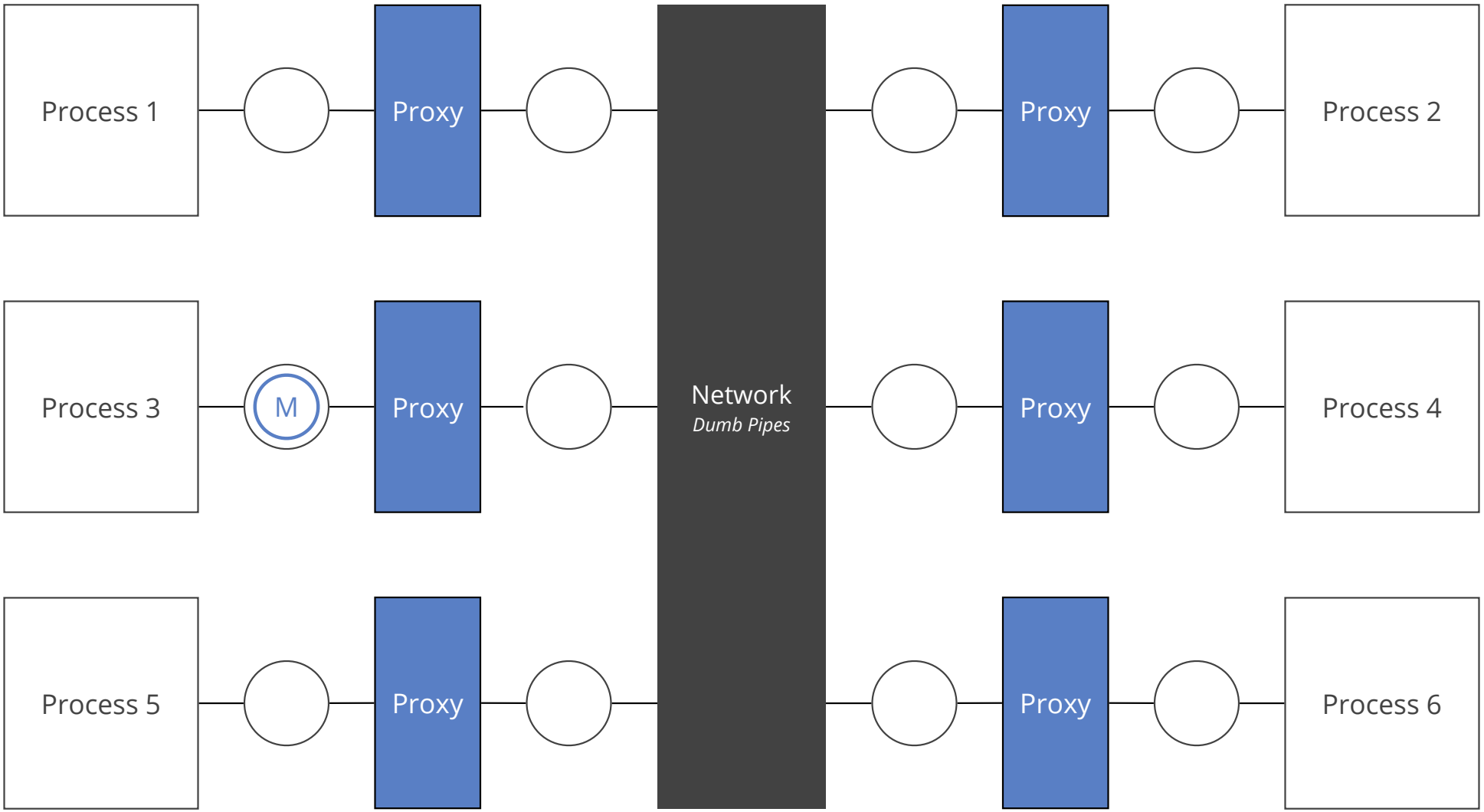

Rest 1
Lib
Rest 3
Lib
Rest 5
Lib
Network
Dumb Pipes
M
Lib
Rest 2
Lib
Rest 4
Lib
Rest 6

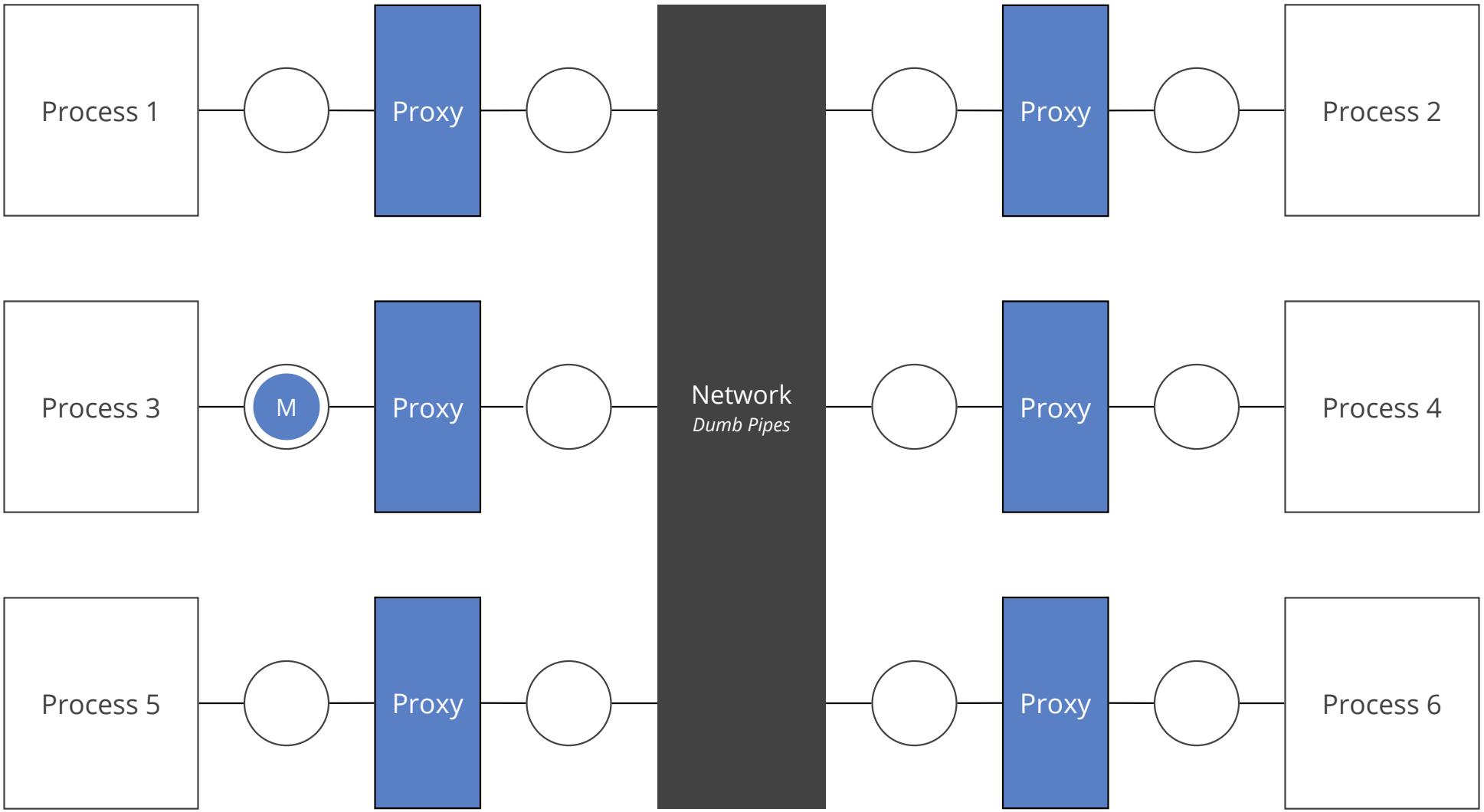

Rest 1
Lib
Network
Dumb Pipes
Lib
M
Rest 2
Rest 3
Lib
Lib
Rest 4
Rest 5
Lib
Lib
Rest 6

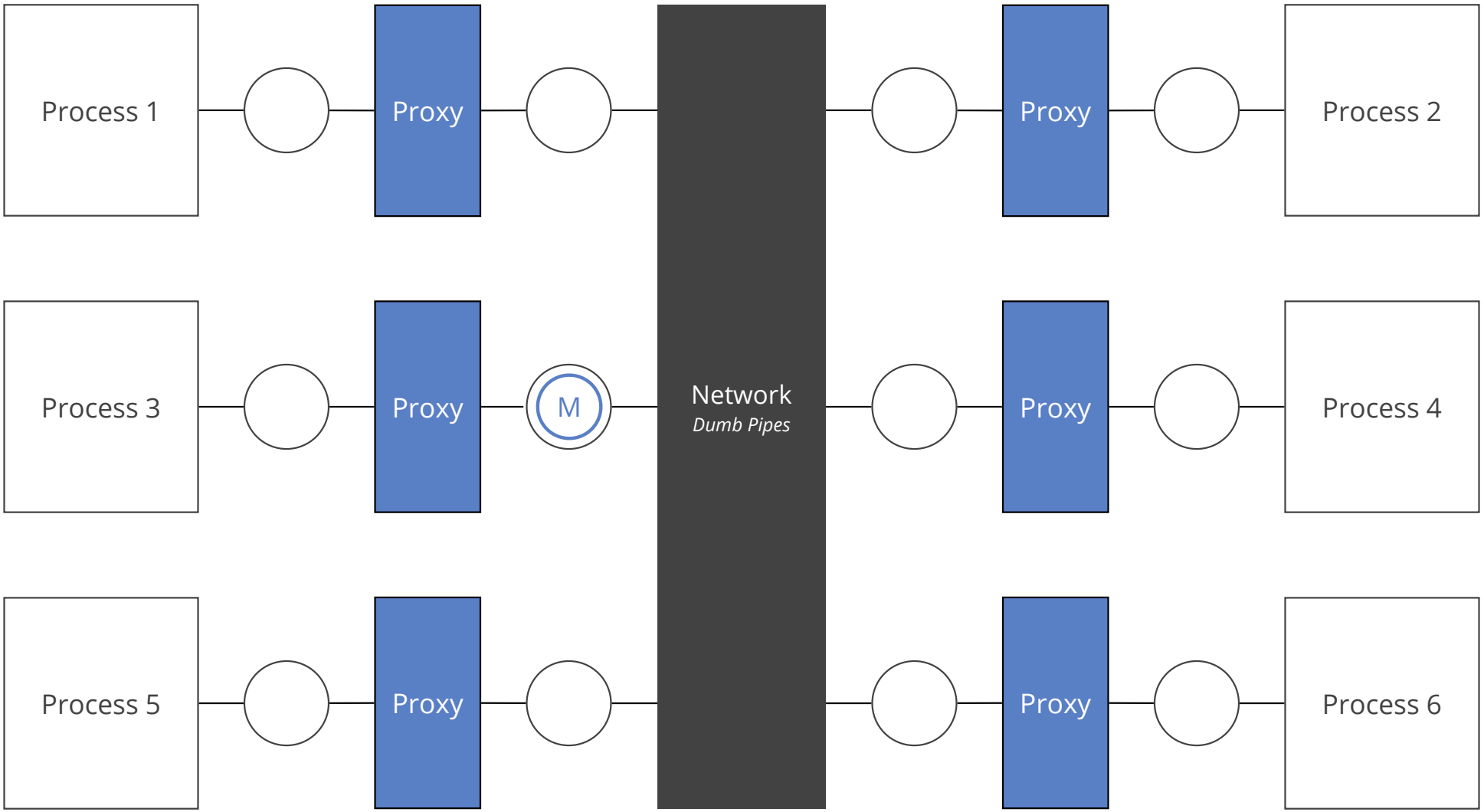

Rest 1
Rest 2
Rest 3
Network
Smart Pipes
Rest 4
Rest 5
Rest 6

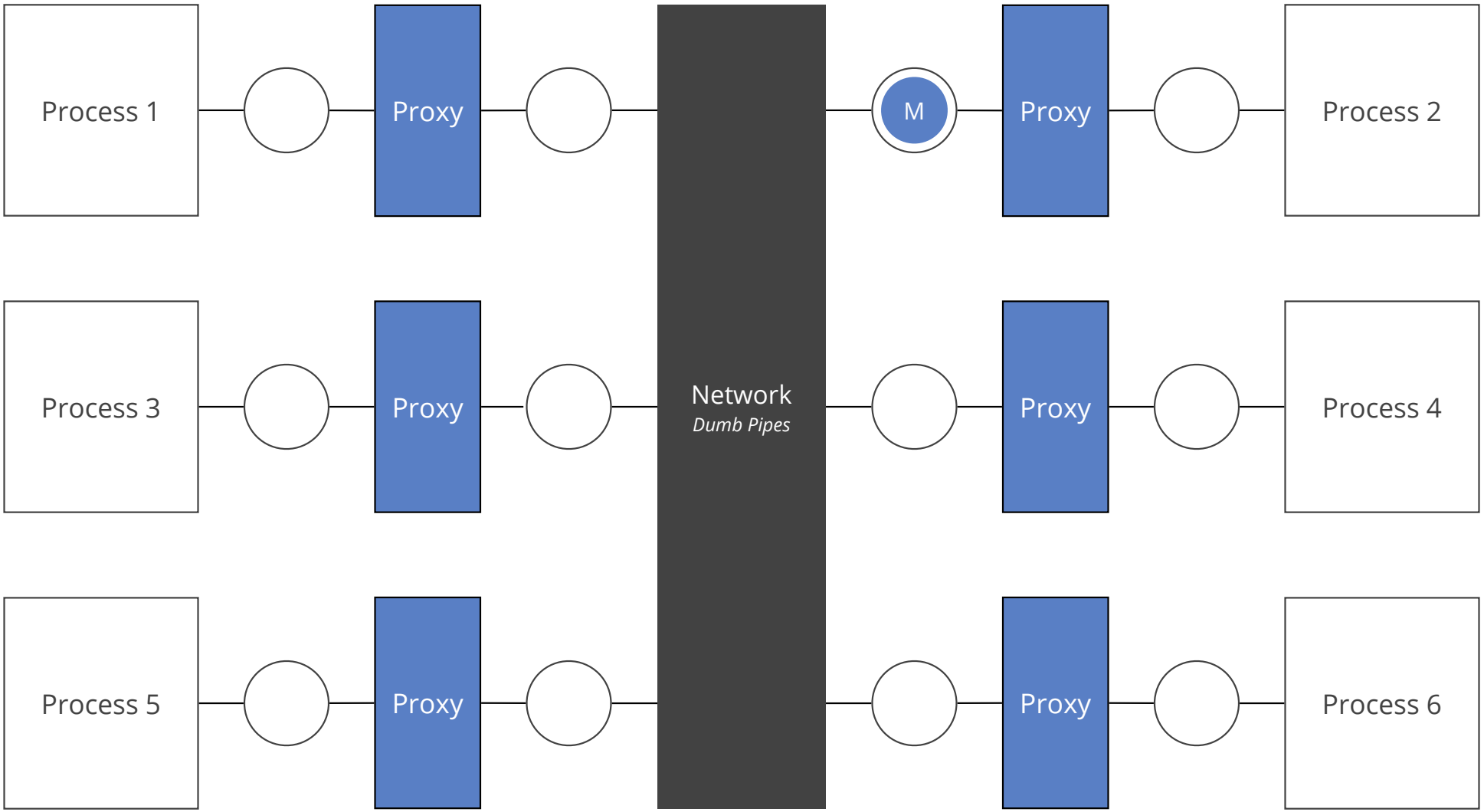
Run

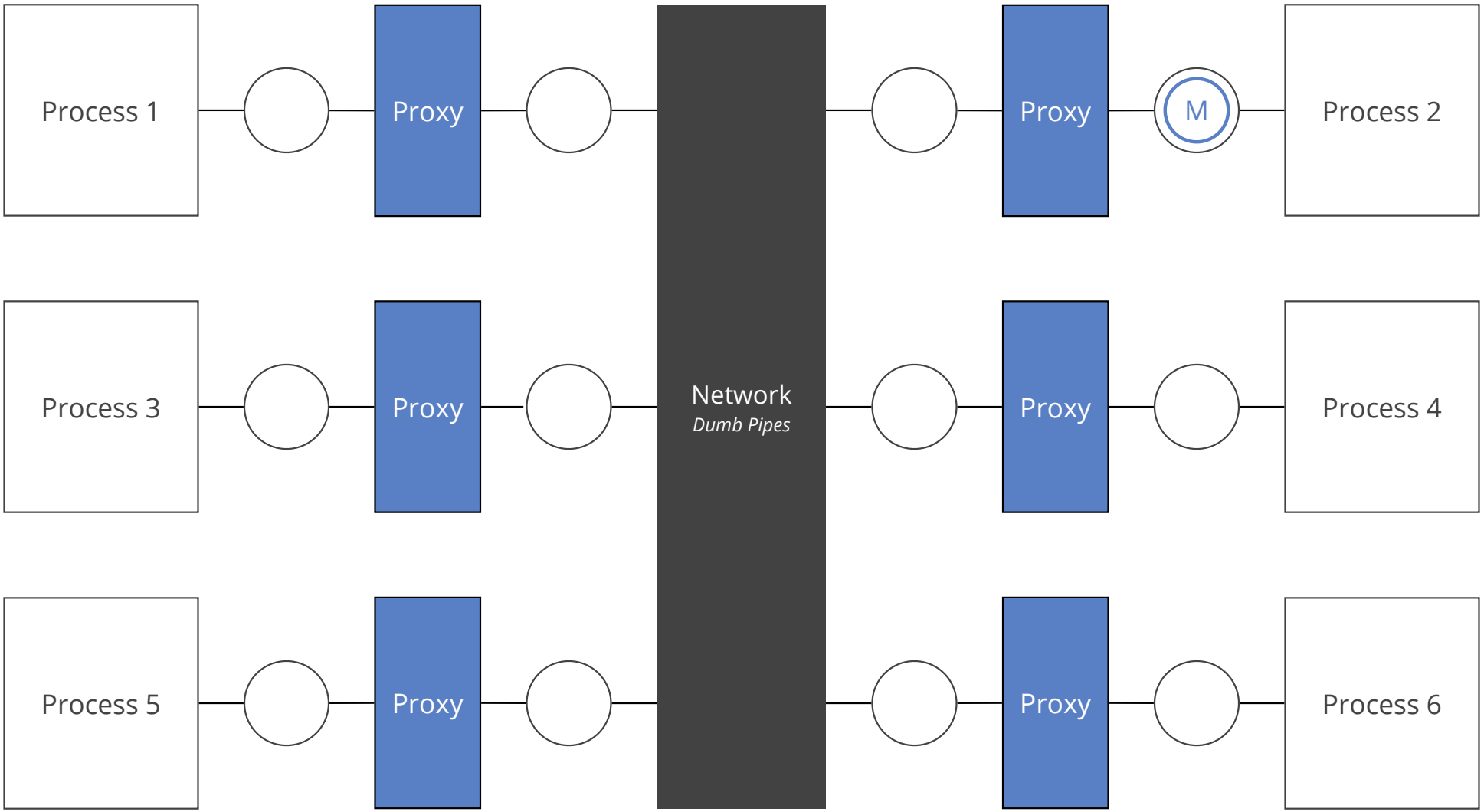

Process 1
Proxy
Network
Dumb Pipes
Proxy
Process 2
Process 3
Proxy
Proxy
Process 4
Process 5
Proxy
Proxy
Process 6

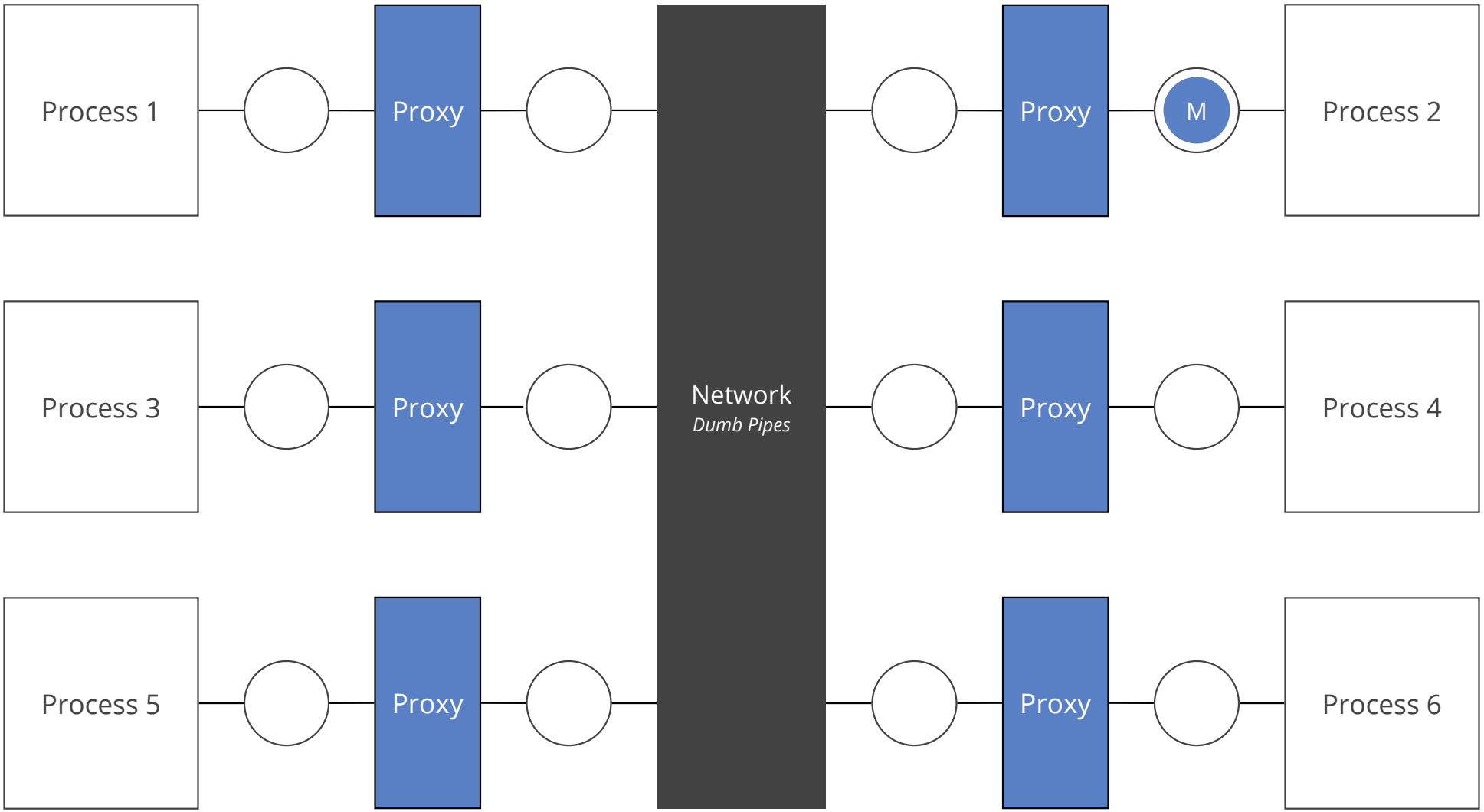

Process 1
Proxy
Proxy
Process 2
Process 3
M
Proxy
Network
Dumb Pipes
Proxy
Process 4
Process 5
Proxy
Proxy
Process 6

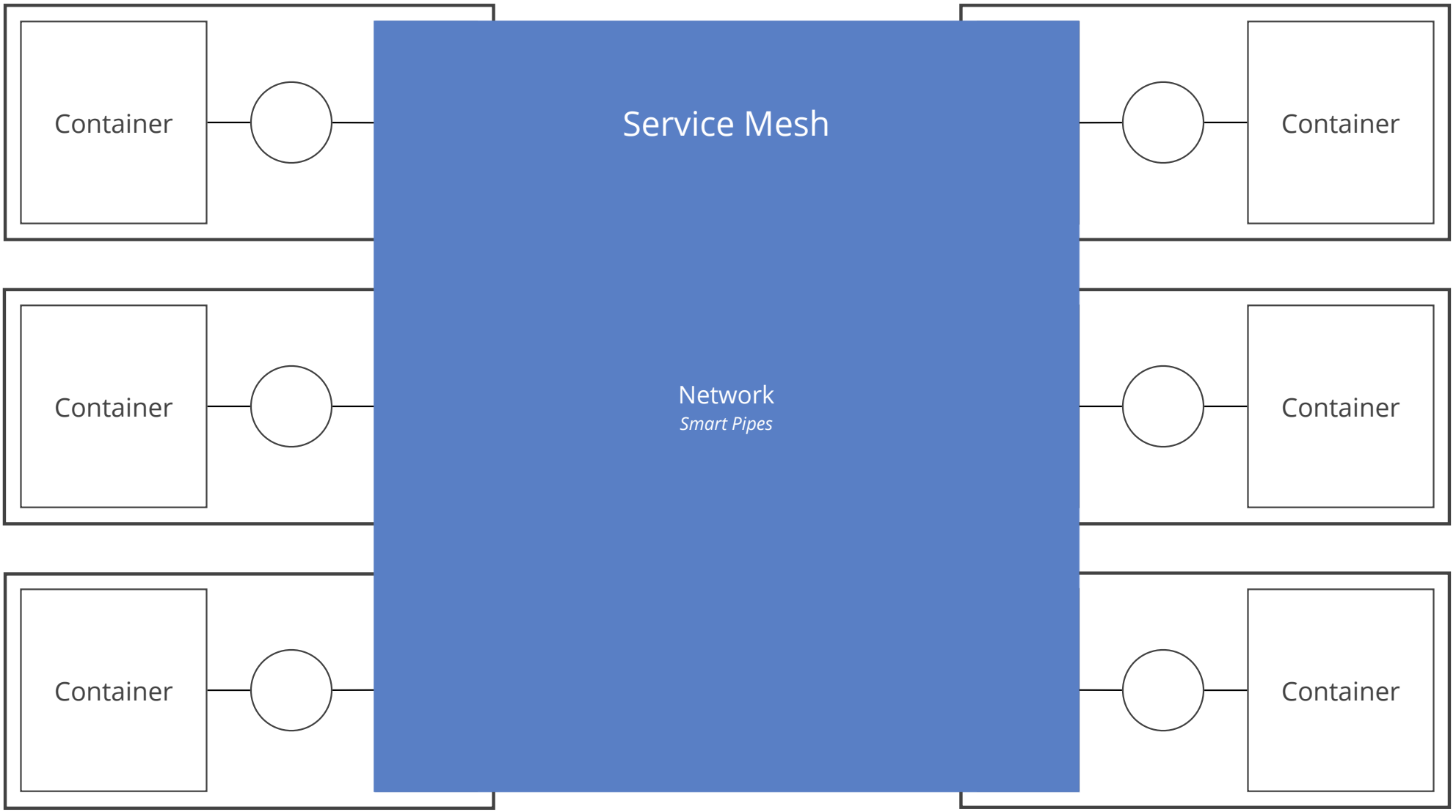

Process 1
Proxy
Proxy
Process 2
Process 3
M
Proxy
Network
Dumb Pipes
Proxy
Process 4
Process 5
Proxy
Proxy
Process 6


Process 1
Proxy
Proxy
Process 2
Process 3
Proxy
M
Network
Dumb Pipes
Proxy
Process 4
Process 5
Proxy
Proxy
Process 6


Process 1
Proxy
M
Proxy
Process 2
Process 3
Proxy
Network
Dumb Pipes
Proxy
Process 4
Process 5
Proxy
Proxy
Process 6

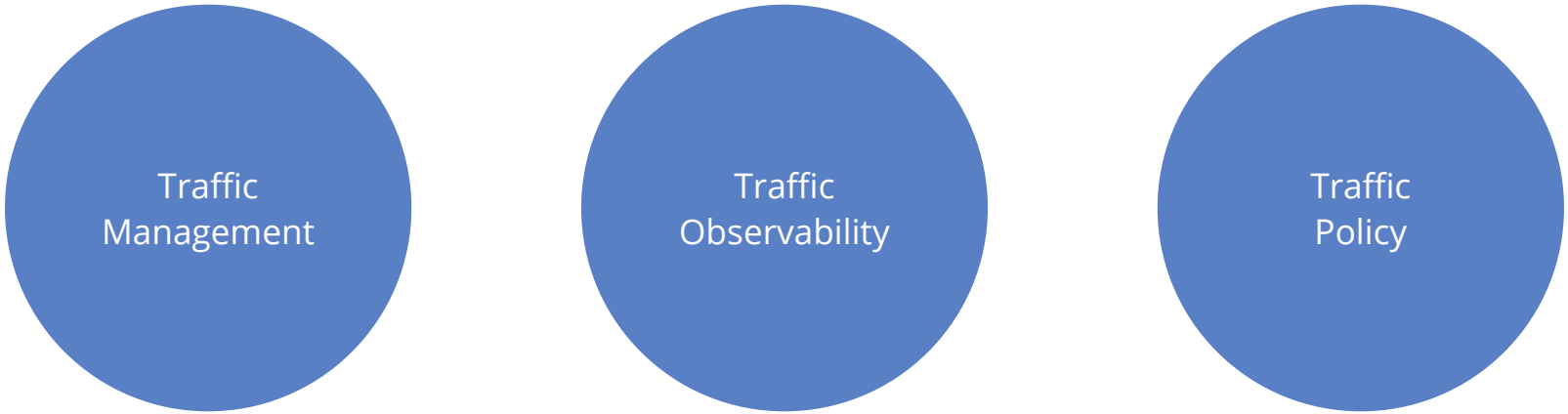

Process 1
Proxy
Proxy
M
Process 2
Process 3
Proxy
Network
Dumb Pipes
Proxy
Process 4
Process 5
Proxy
Proxy
Process 6

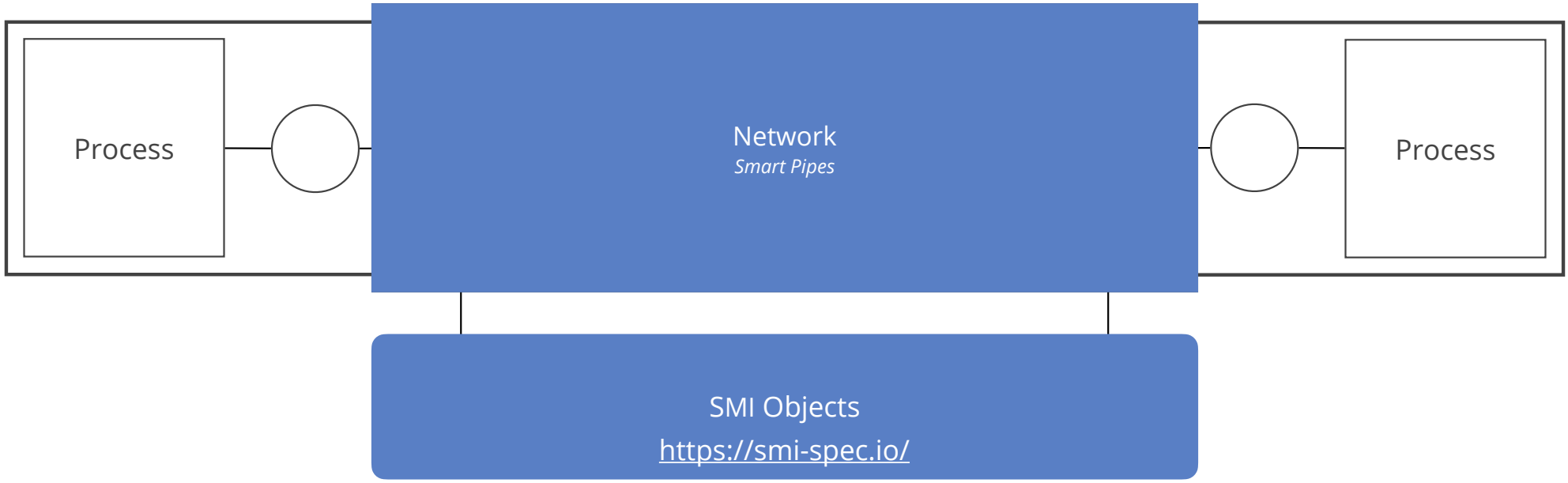

Process 1
Proxy
Network
Dumb Pipes
Proxy
M
Process 2
Process 3
Proxy
Proxy
Process 4
Process 5
Proxy
Proxy
Process 6


Container
Container
Container
Service Mesh
Network
Smart Pipes
Container
Container
Container

Service Mesh

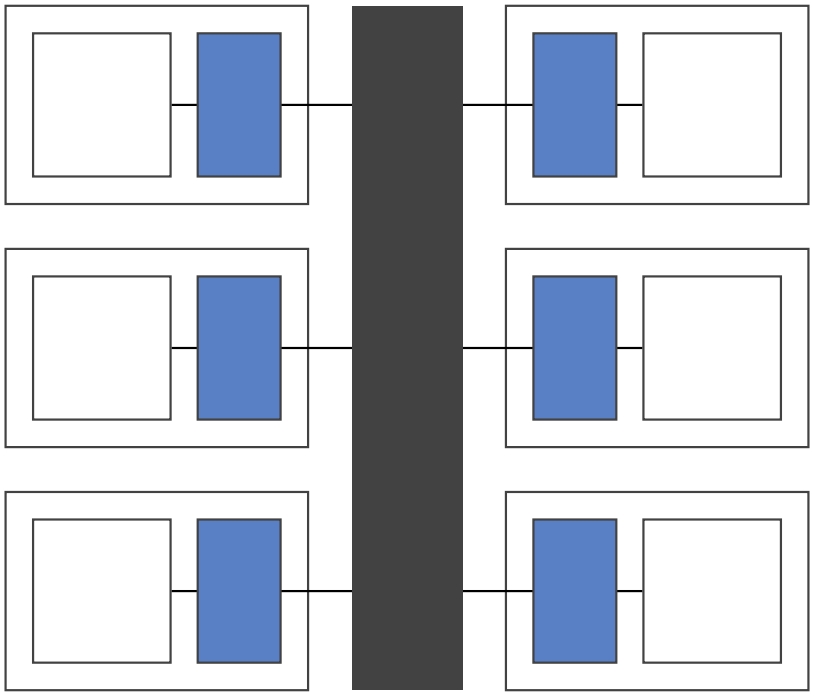
Service Mesh Interface

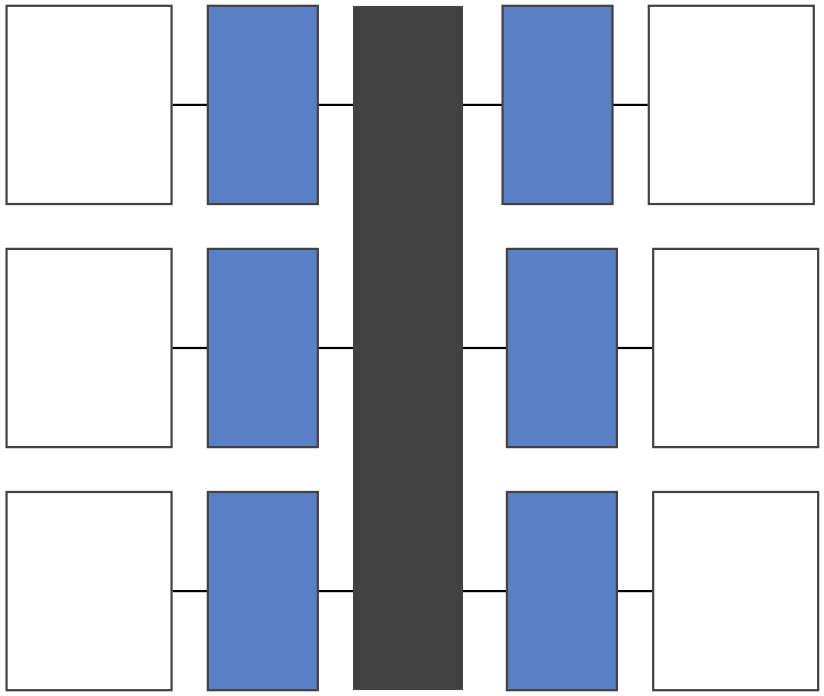

Traffic
Management
Traffic
Observability
Traffic
Policy

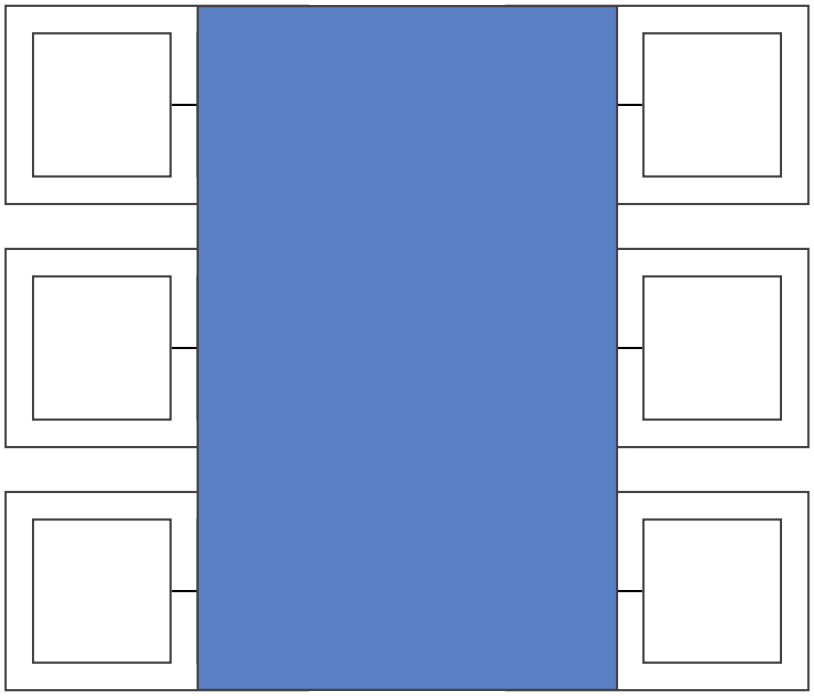

Process
Network
Smart Pipes
Process
SMI Objects
https://smi-spec.io/

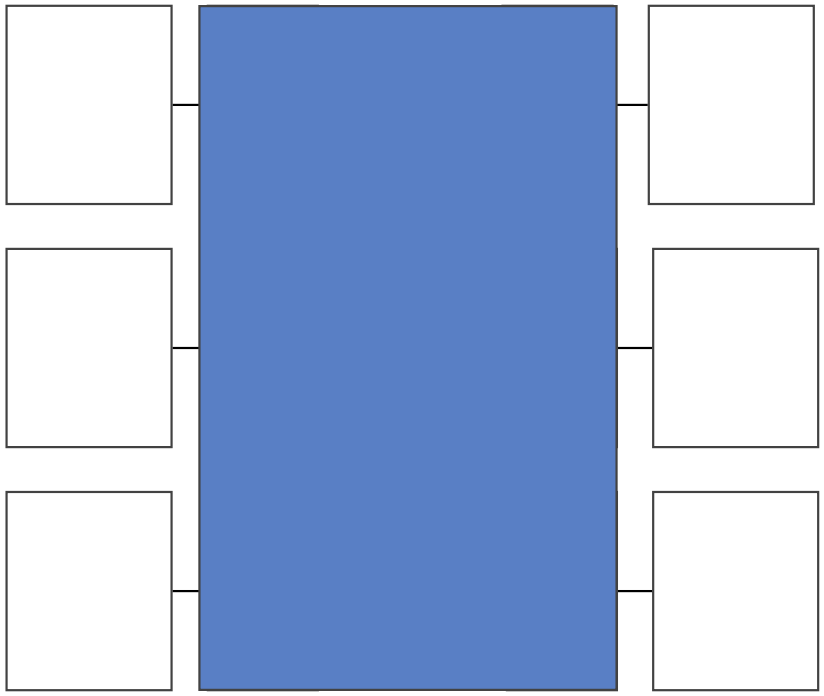
Service Mesh

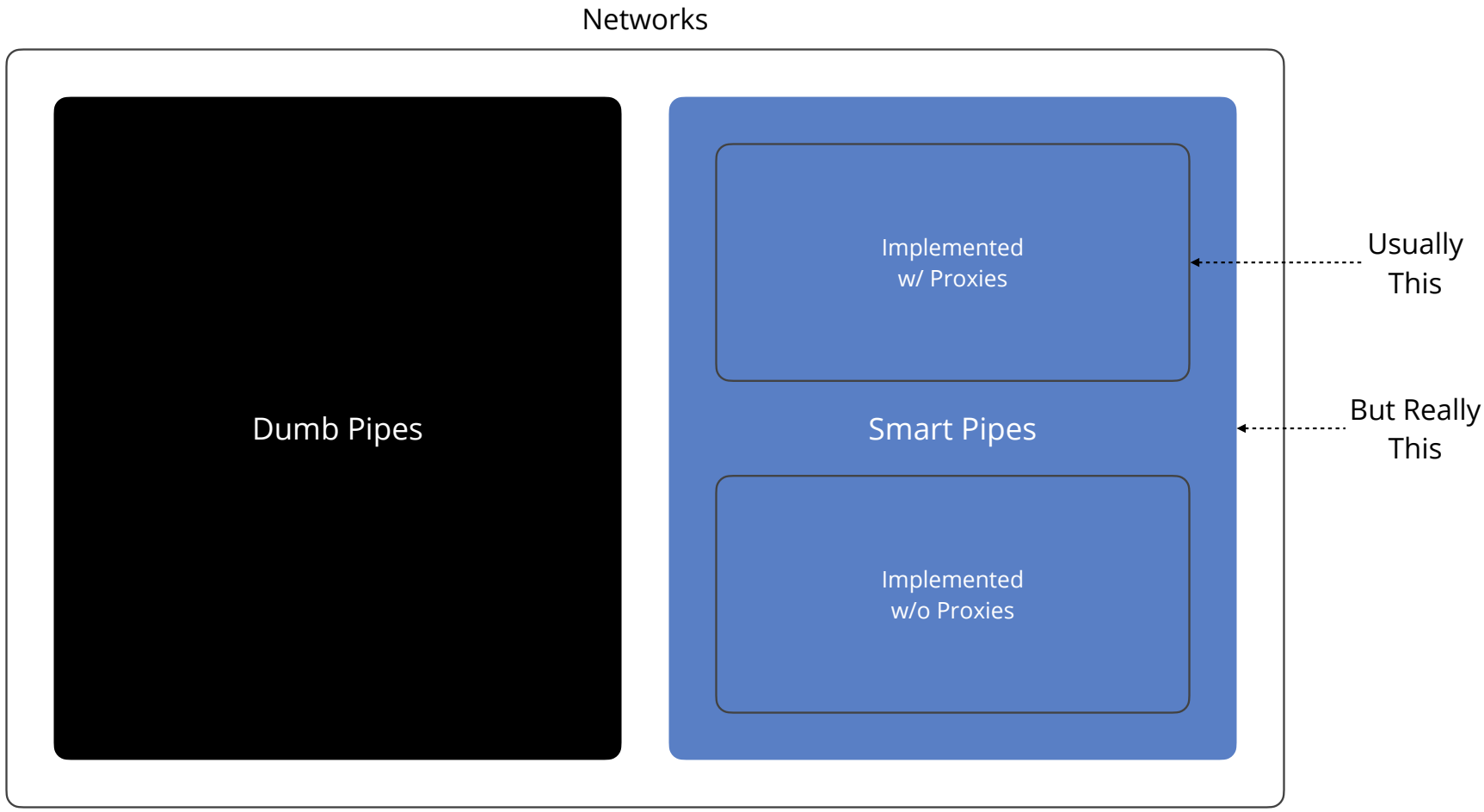
Summary



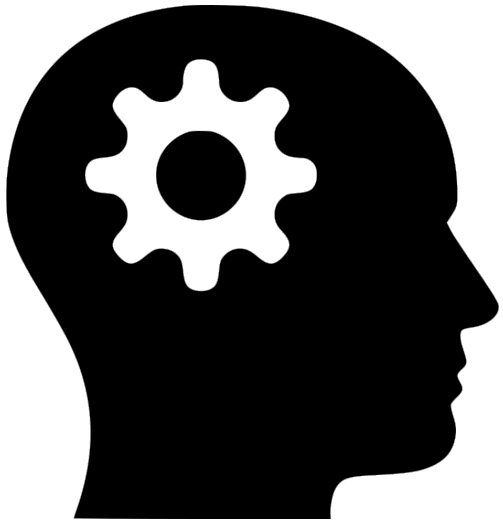


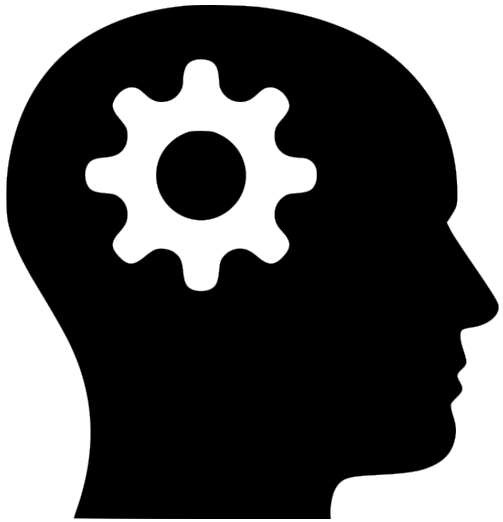





Networks
Dumb Pipes
Implemented
w/ Proxies
Smart Pipes
Implemented
w/o Proxies
Usually
This
But Really
This

Part II

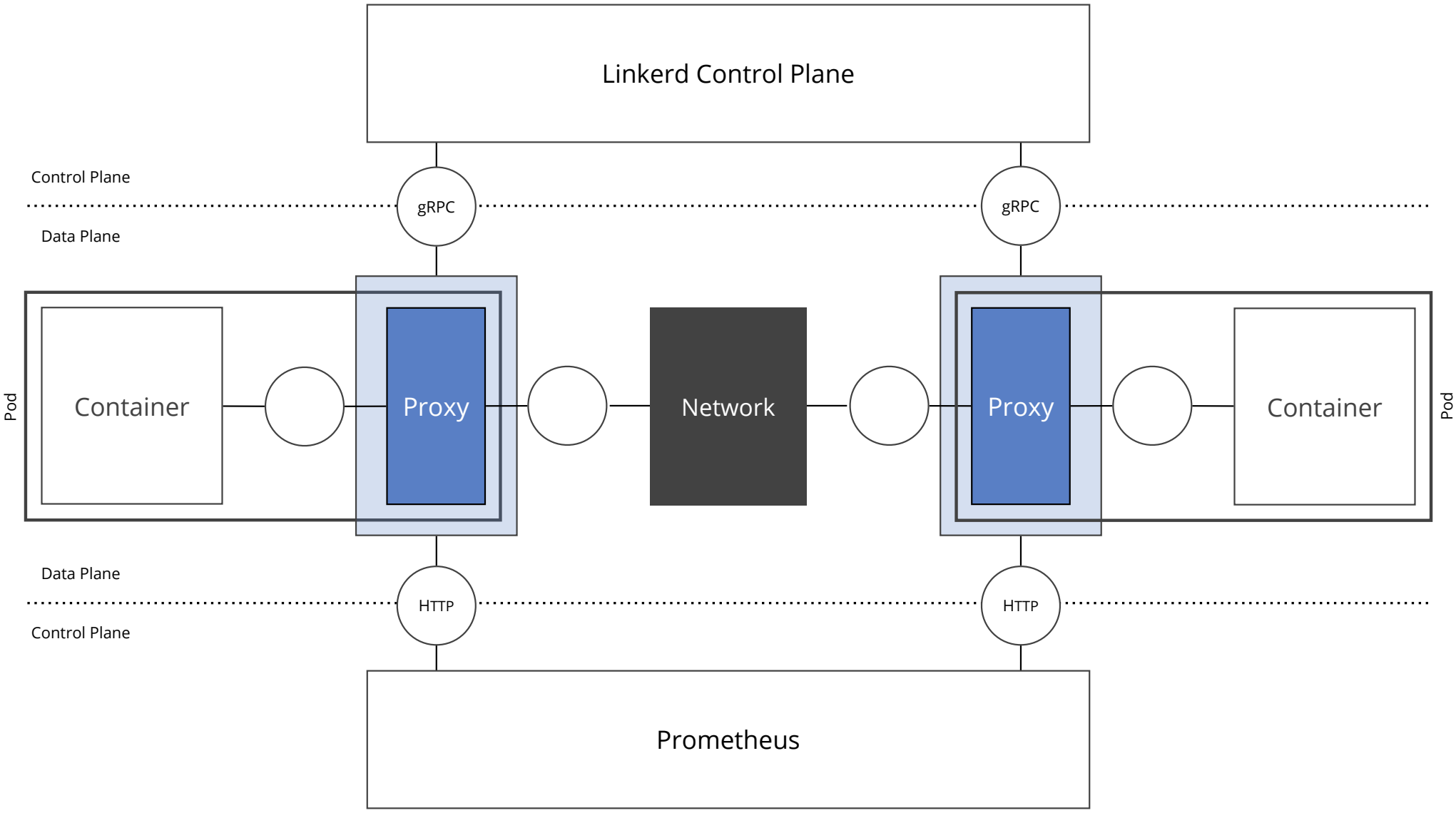


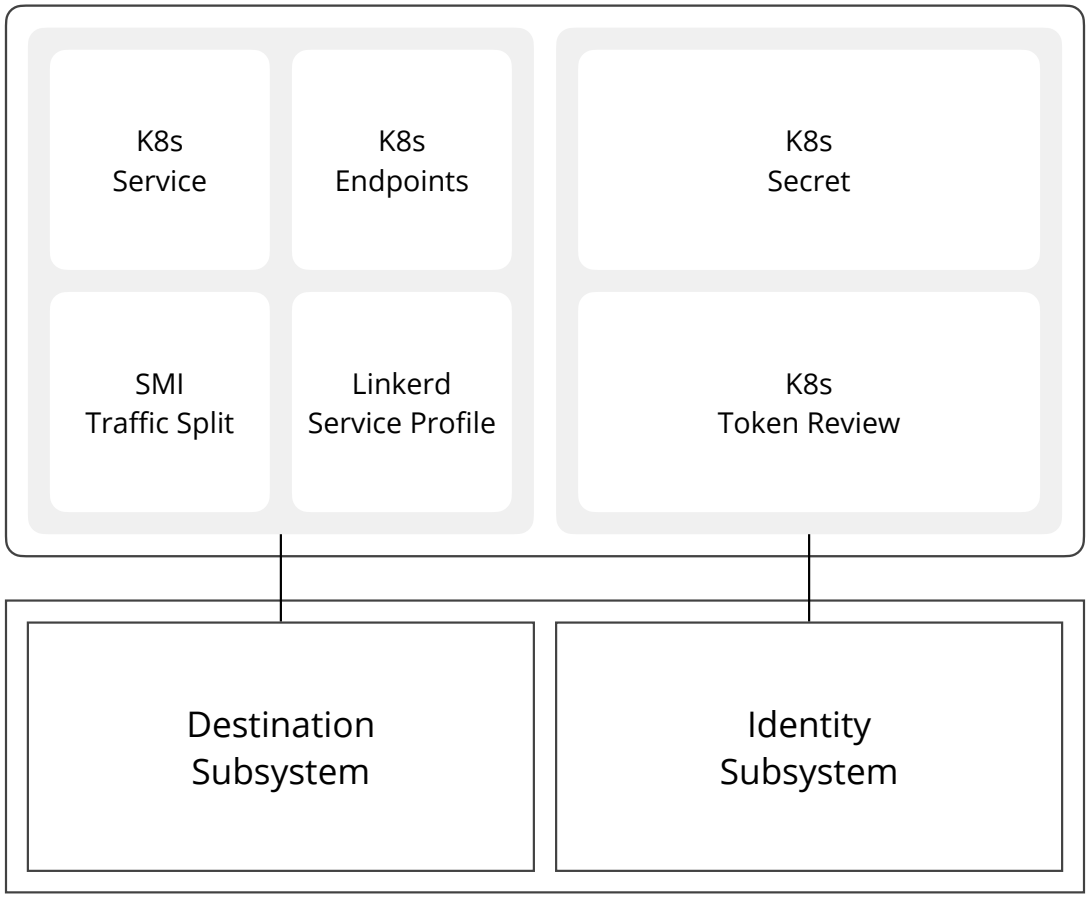


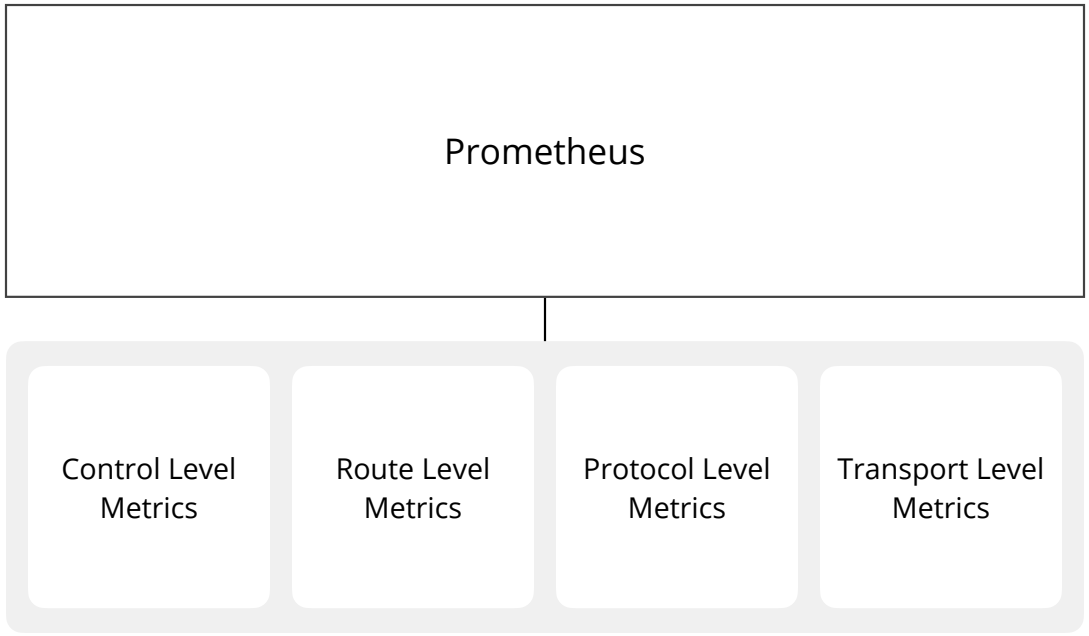

LINKERD

Linkerd

Overview


Linkerd Control Plane
Control Plane
gRPC
gRPC
Data Plane
Pod
Container
Proxy
Network
Proxy
Container
Pod
Data Plane
HTTP
HTTP
Control Plane
Prometheus


K8s
Service
K8s
Endpoints
K8s
Secret
SMI
Traffic Split
Linkerd
Service Profile
K8s
Token Review
Destination
Subsystem
Identity
Subsystem


Prometheus
Control Level Metrics
Route Level Metrics
Protocol Level Metrics
Transport Level Metrics

LINKERD

https://linkerd.io

cisco

Thank You

dominik.tornow@gmail.com

Thank You