

Today

- CDN (Coral)

Next Week

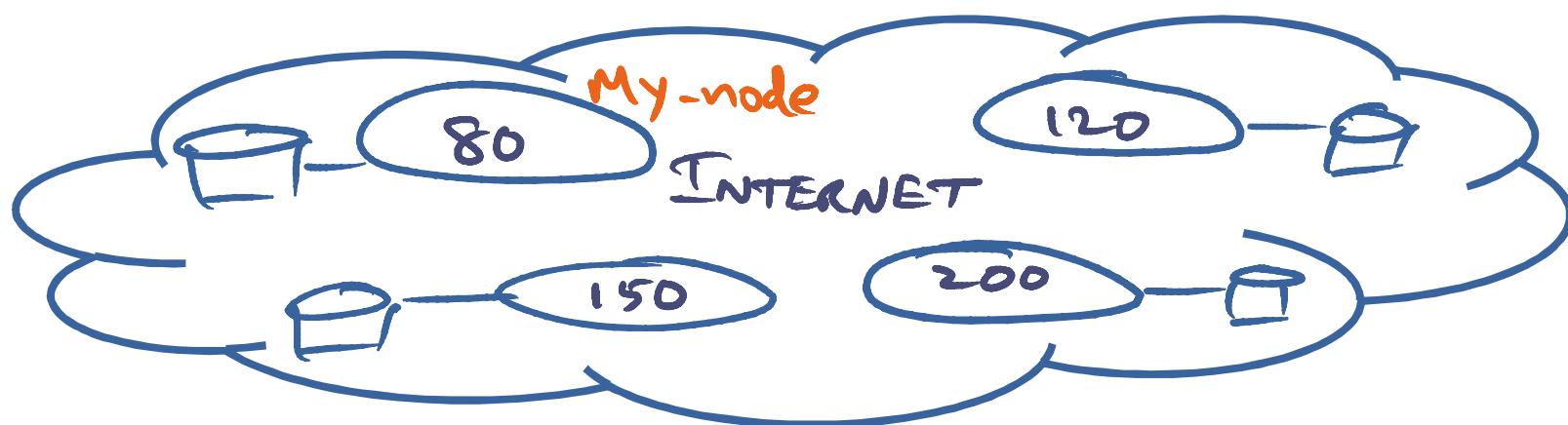
(Lesson 10)

- RT + multimedia

Lesson outline

- ✓ System issues in giant scale services
- ✓ Programming models for apps working on Big Data
- Content Distribution Networks

DHT Content Distribution Networks

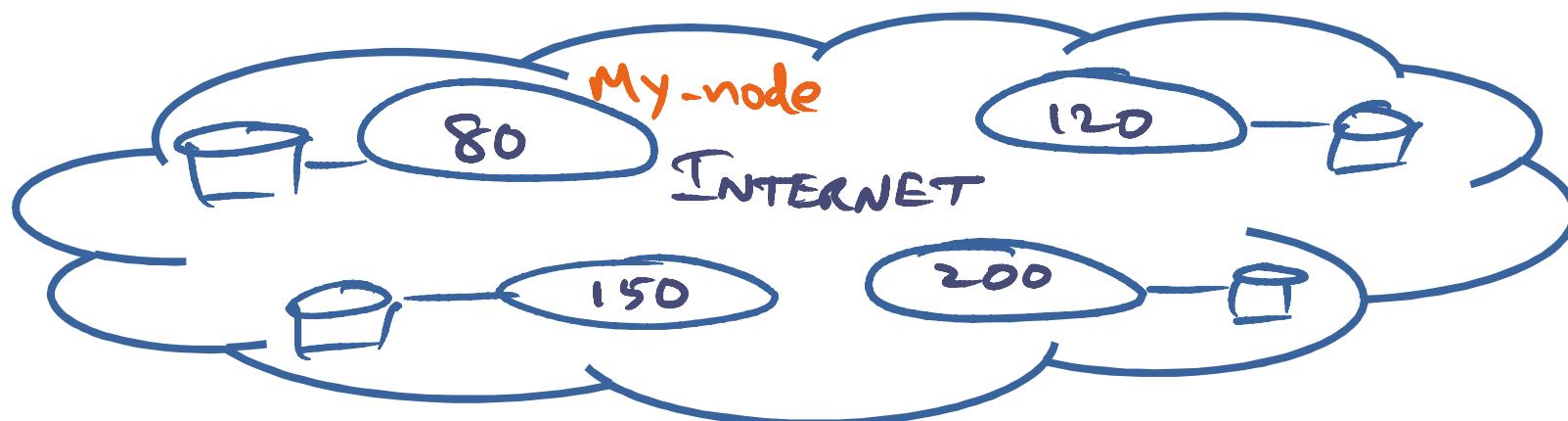


DHT

Content Distribution Networks

$\langle \text{Key}, \text{value} \rangle$

Content hash node-id where content
 stored
content hash = 149 $\Rightarrow \langle 149, 80 \rangle$



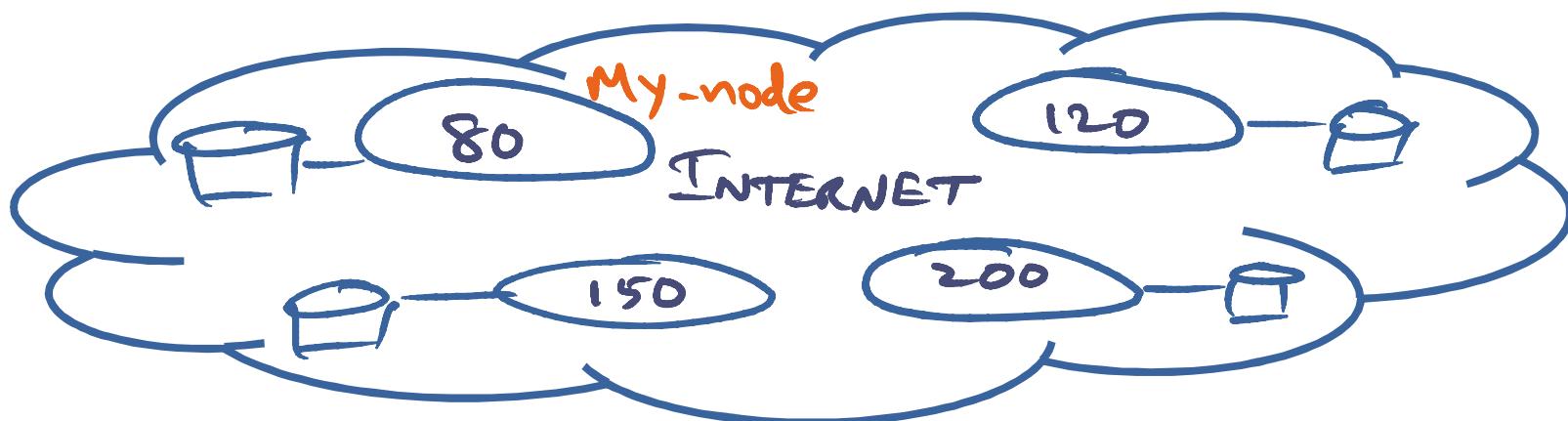
DHT

Content Distribution Networks

$\langle \text{Key}, \text{value} \rangle$

Content hash node-id where content
 stored

content hash = 149 $\Rightarrow \langle 149, 80 \rangle \Rightarrow$ Where to store?



DHT

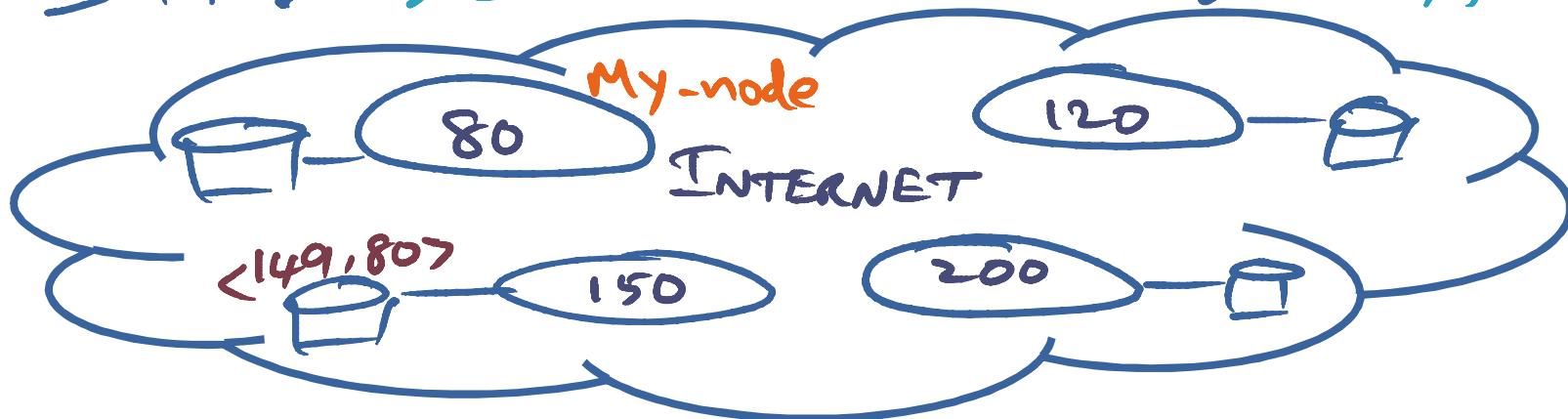
Content Distribution Networks

`<Key, value>`

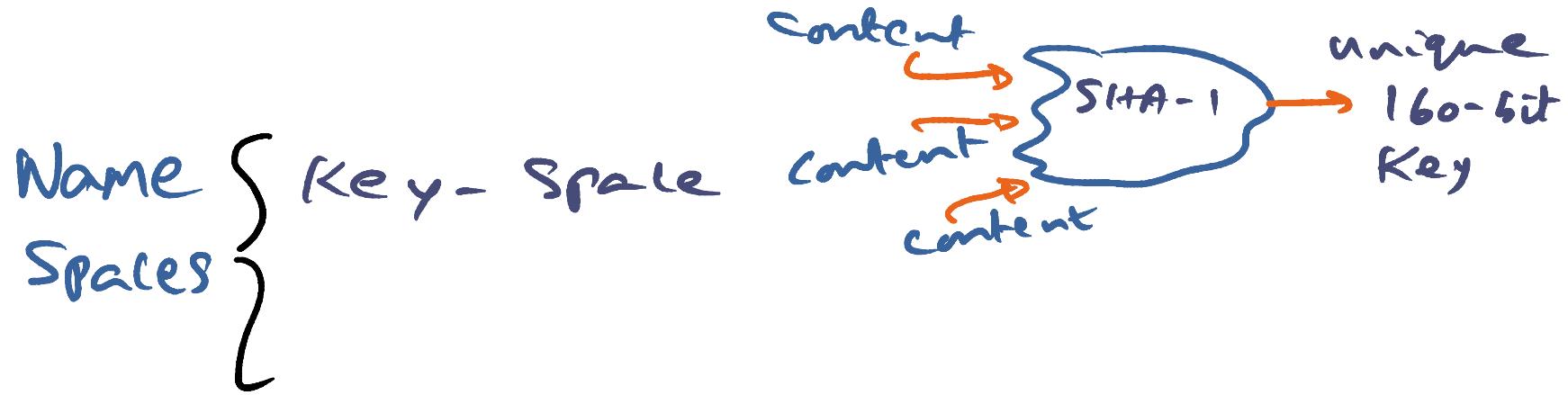
Content hash ↗ node-id where content
↑ stored

Content hash stored
content hash = 149 \Rightarrow <149, 80> \Rightarrow where to store?

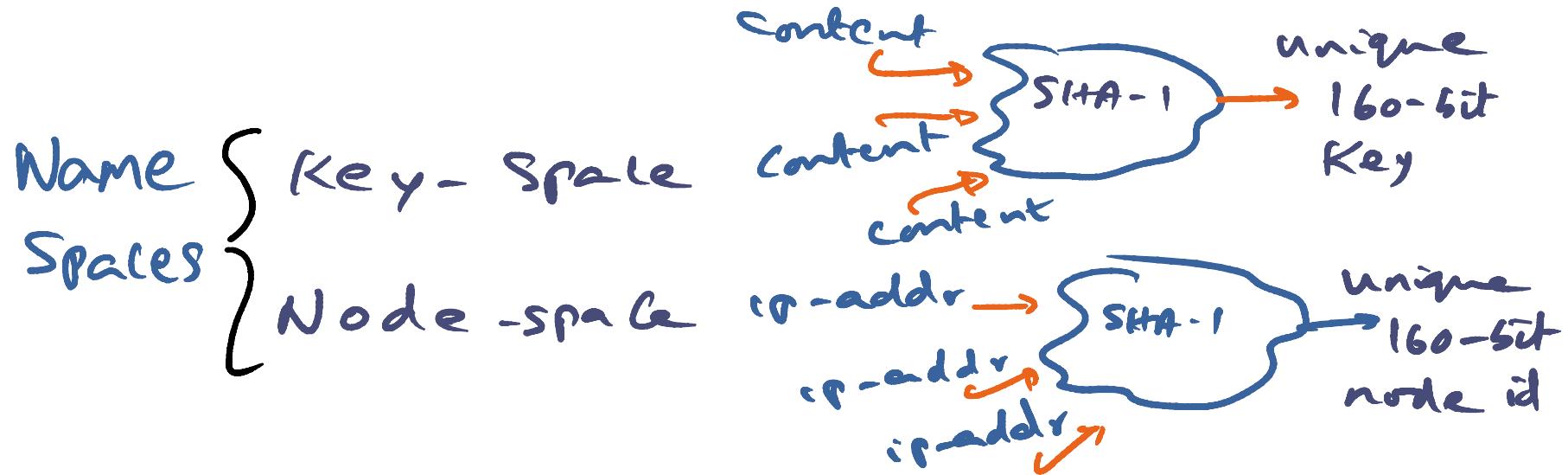
DHT: Key \cong node-id for storing $\langle \text{key}, \text{value} \rangle$



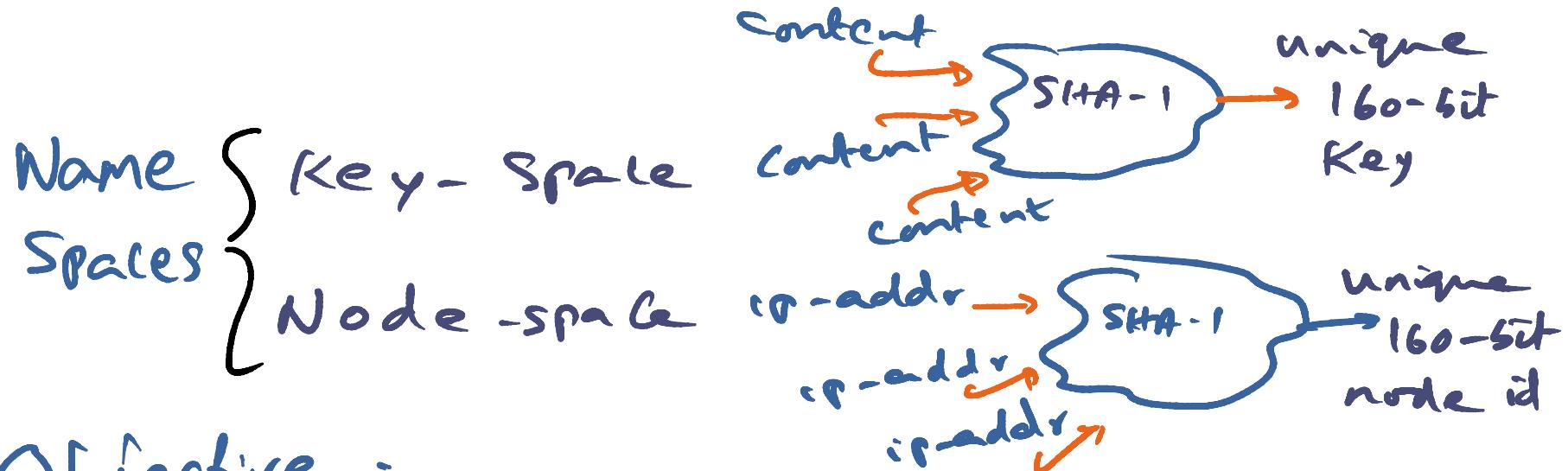
DHT Details



DHT Details



DHT Details

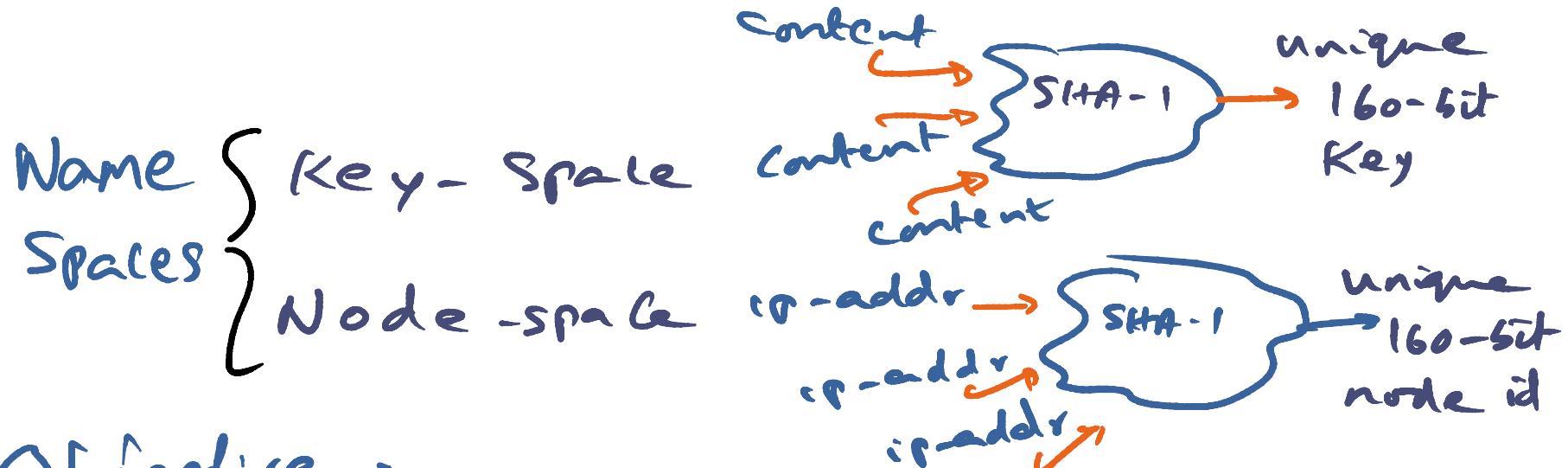


Objective :

$$\langle \text{Key} \rangle \rightarrow \text{nodeid } \langle N \rangle$$

such that $\langle \text{Key} \rangle \approx \langle N \rangle$

DHT Details



Objective :

$$\langle \text{Key} \rangle \rightarrow \text{nodeid } \langle N \rangle$$

such that $\langle \text{Key} \rangle \approx \langle N \rangle$

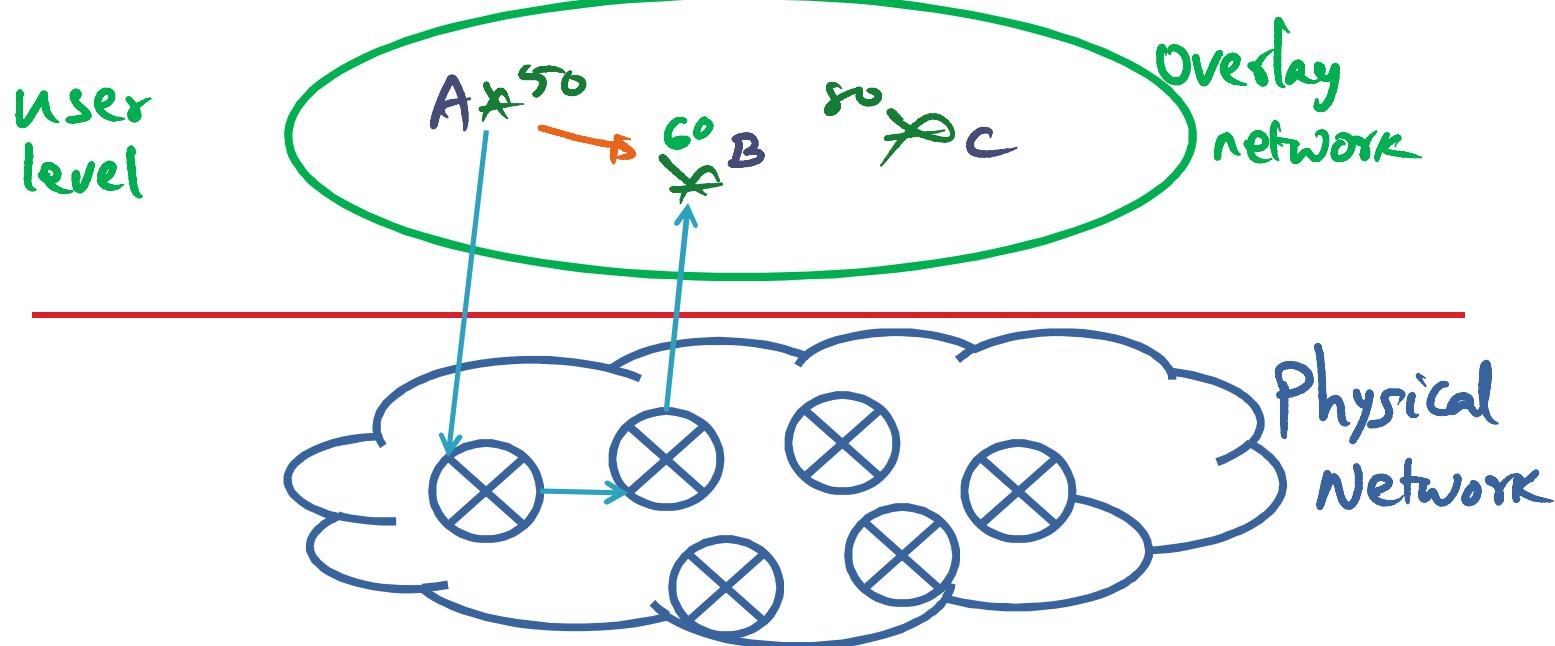
APIs

— Putkey, getkey

CDN - an overlay network

routing table at A
(node id = 50)

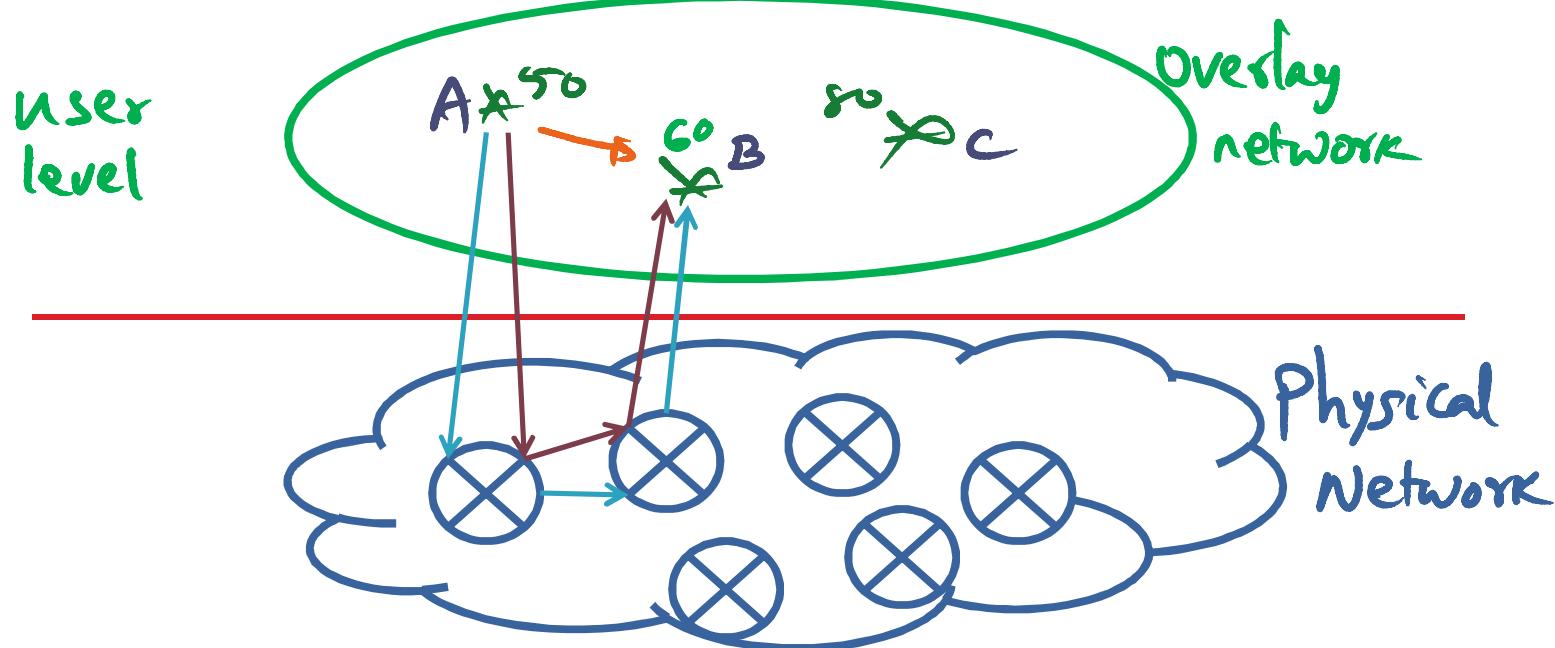
name	node id	next hop
B	60	60
C	80	60



CDN - an overlay network

routing table at A
(node id = 50)

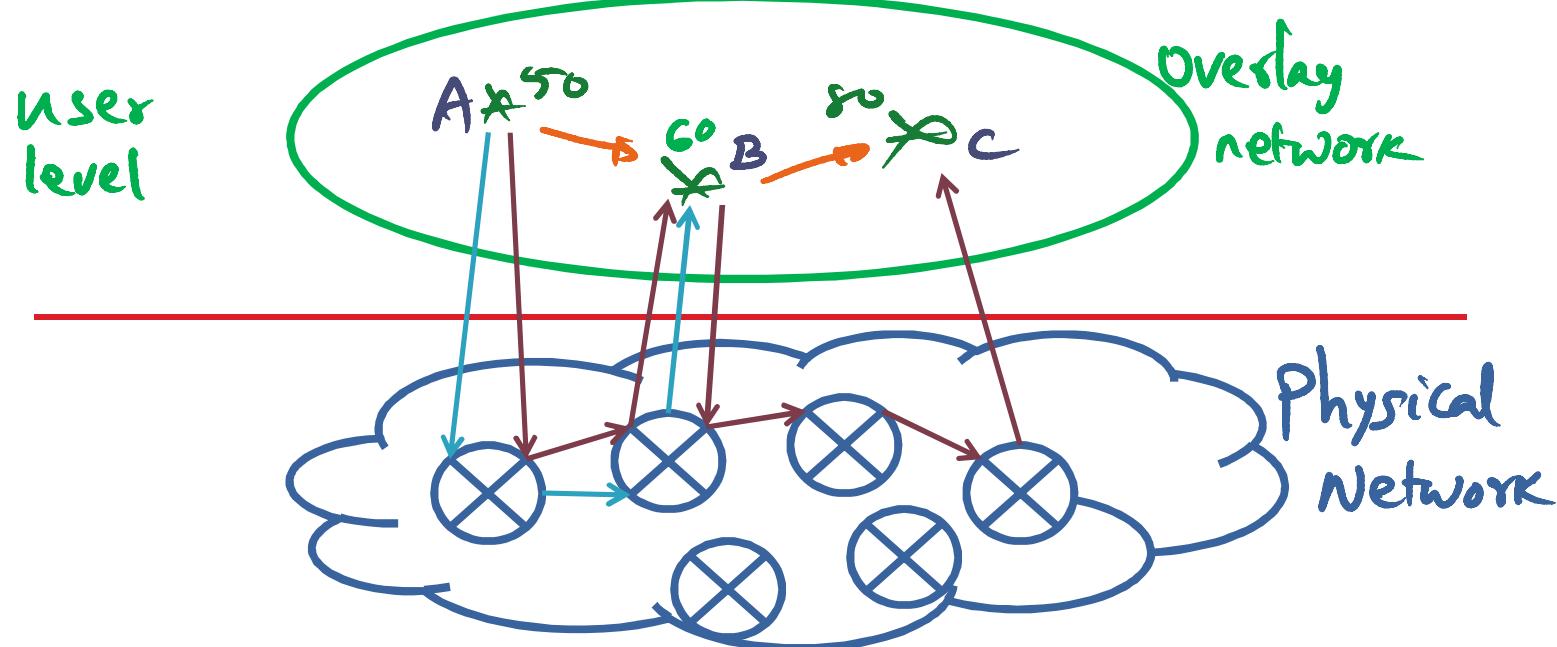
name	node id	next hop
B	60	60
C	80	60



CDN - an overlay network

routing table at A
(node id = 50)

name	node id	next hop
B	60	60
C	80	60



Overlay Networks in general

Examples of overlay

OS Level

— IP network is an overlay IP Addr | Mac Addr
on LAN

Overlay Networks in general

Examples of overlay

OS Level

- IP network is an overlay
on LAN

<u>IP Addr</u>	<u>Mac Addr</u>

App Level

- CDN is an overlay
on TCP/IP

<u>Node ID</u>	<u>IP Addr</u>

DHT + CDNs

Placement

- Put $\langle \text{key}, \text{value} \rangle$

Content hash ↗
node-id where content
 stored

Retrieval of value given key

- get $\langle \text{key} \rangle$

- get back $\langle \text{value} \rangle$

Traditional approach (greedy)

place $\langle \text{key}, \text{value} \rangle$ in
node N , where $N = \text{key}$

retrieve :

given key K goto node
 N ($\text{closest to Key } K$)

routing table @ A

Known Peers (N)	IP-addr
60
79

routing table @ B

Known Peers (N)	IP-addr
60	...
109	...

Traditional approach (greedy)

place $\langle \text{key}, \text{value} \rangle$ in
node N , where $N = \text{key}$

retrieve :

given key K go to node

(closest to key K)

Key = 58 }
59 }
routing table @ A

Known Peers (N)	IP-addr
60
109

routing table @ B

Known Peers (N)	IP-addr
60	...
109	...

Traditional approach (greedy)

place $\langle \text{key}, \text{value} \rangle$ in
node N , where $N = \text{key}$

retrieve :

given key K go to node

(closest to Key K)

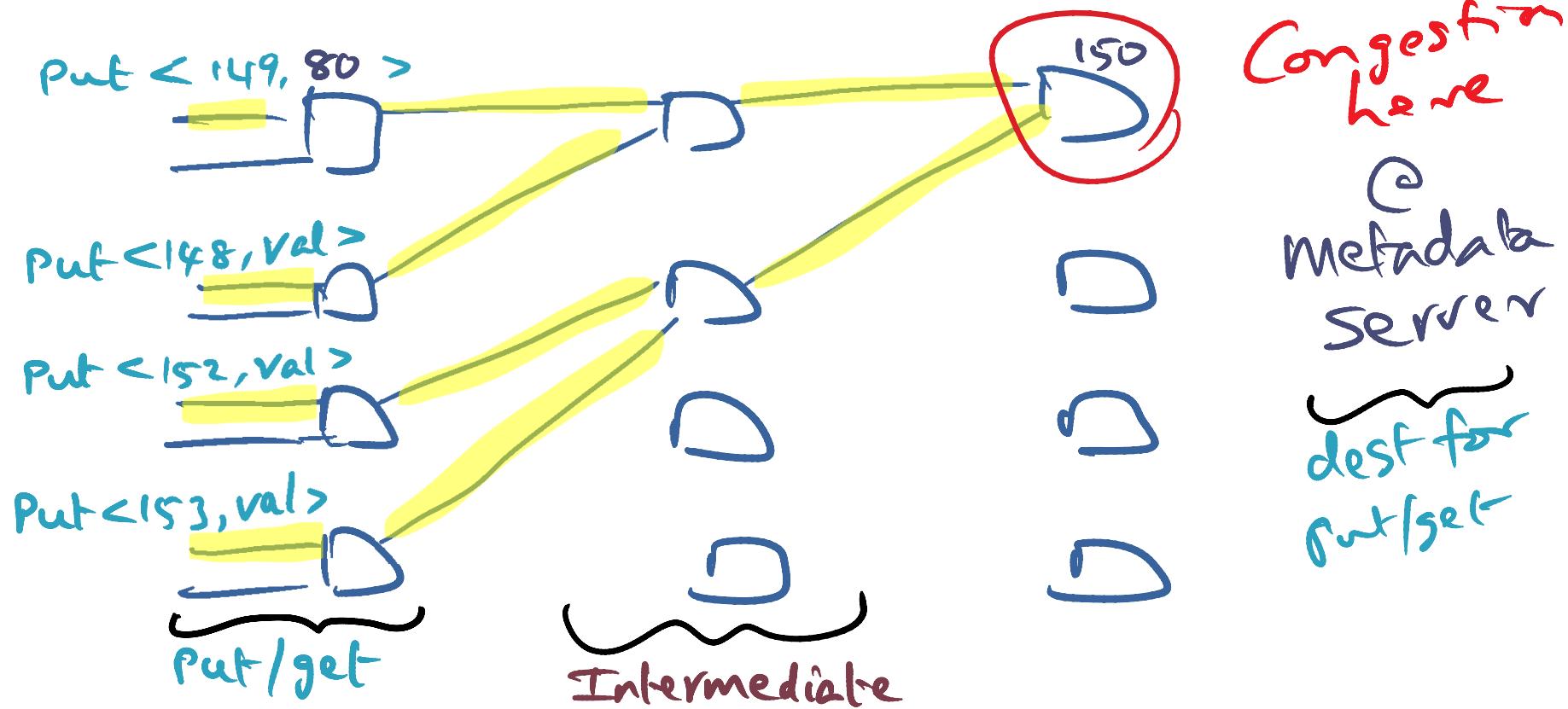
Routing table @ A

Known Peers (N)	IP-addr
60	...
79	...

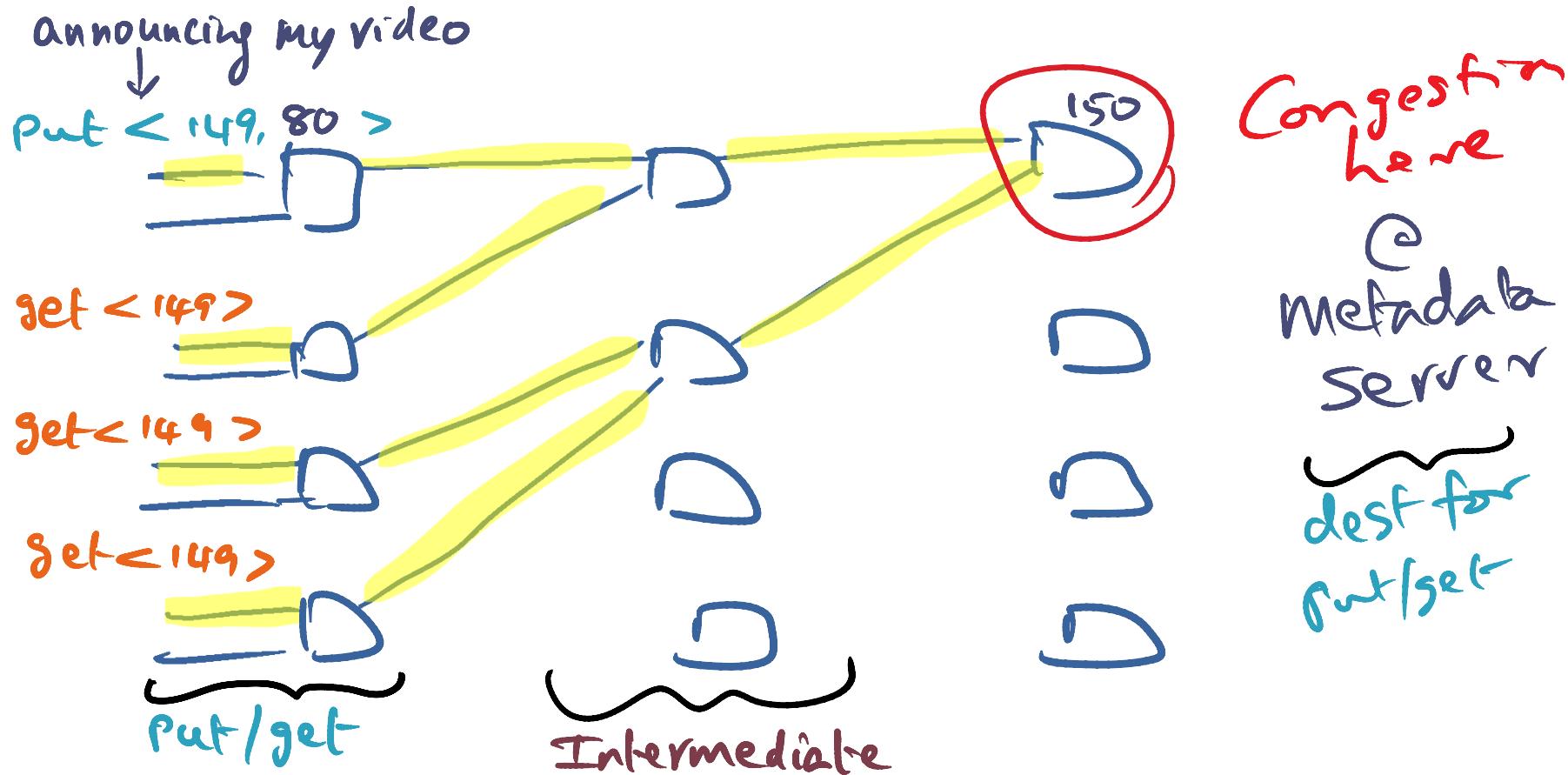
routing table @ B

Known Peers (N)	IP-addr
60	...
109	...

Greedy Approach leads to metadata server overload



Greedy Approach leads to metadata server overload



Origin Server Overload

