

Today

✓ Distributed System Basics

✓ Lamport's clock

✓ Efficient Communication Software

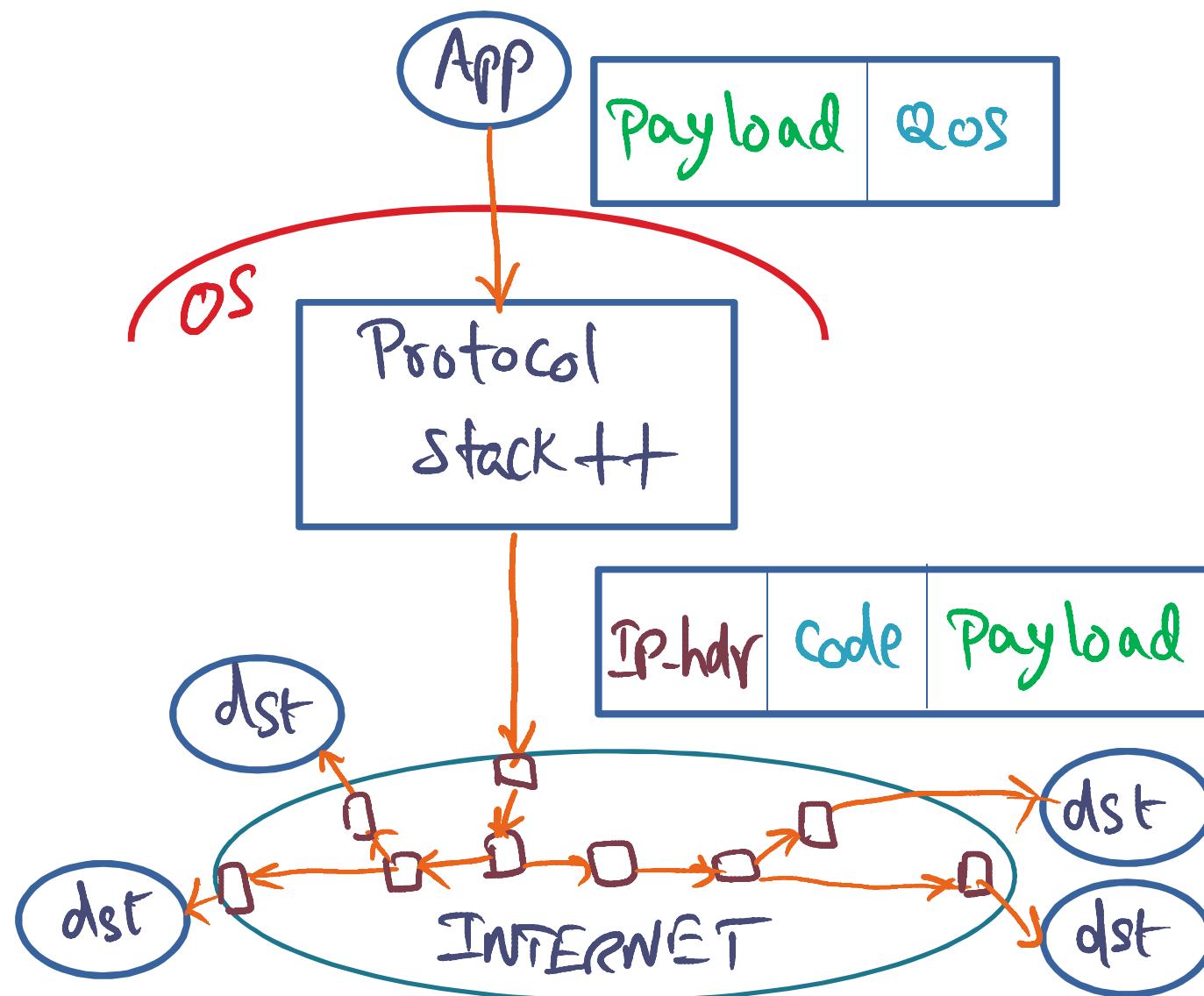
- * Application interface to the Kernel
- * Inside the Kernel

wetterall {+ End-to-End QoS via Active Networks

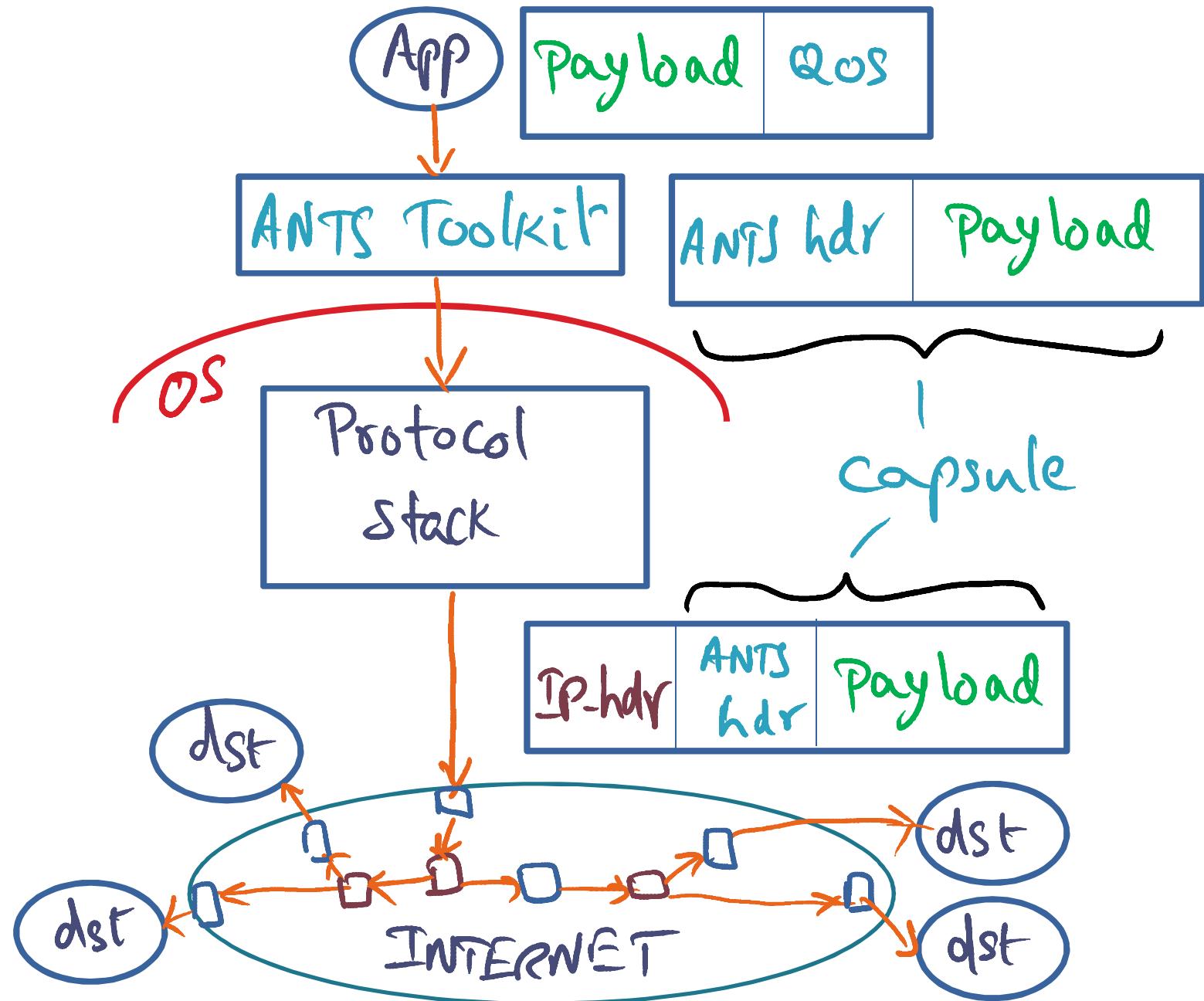
⇒ Synthesizing Network Protocol Stacks

next week Spring Kernel (Dist. Objects) Lesson 6

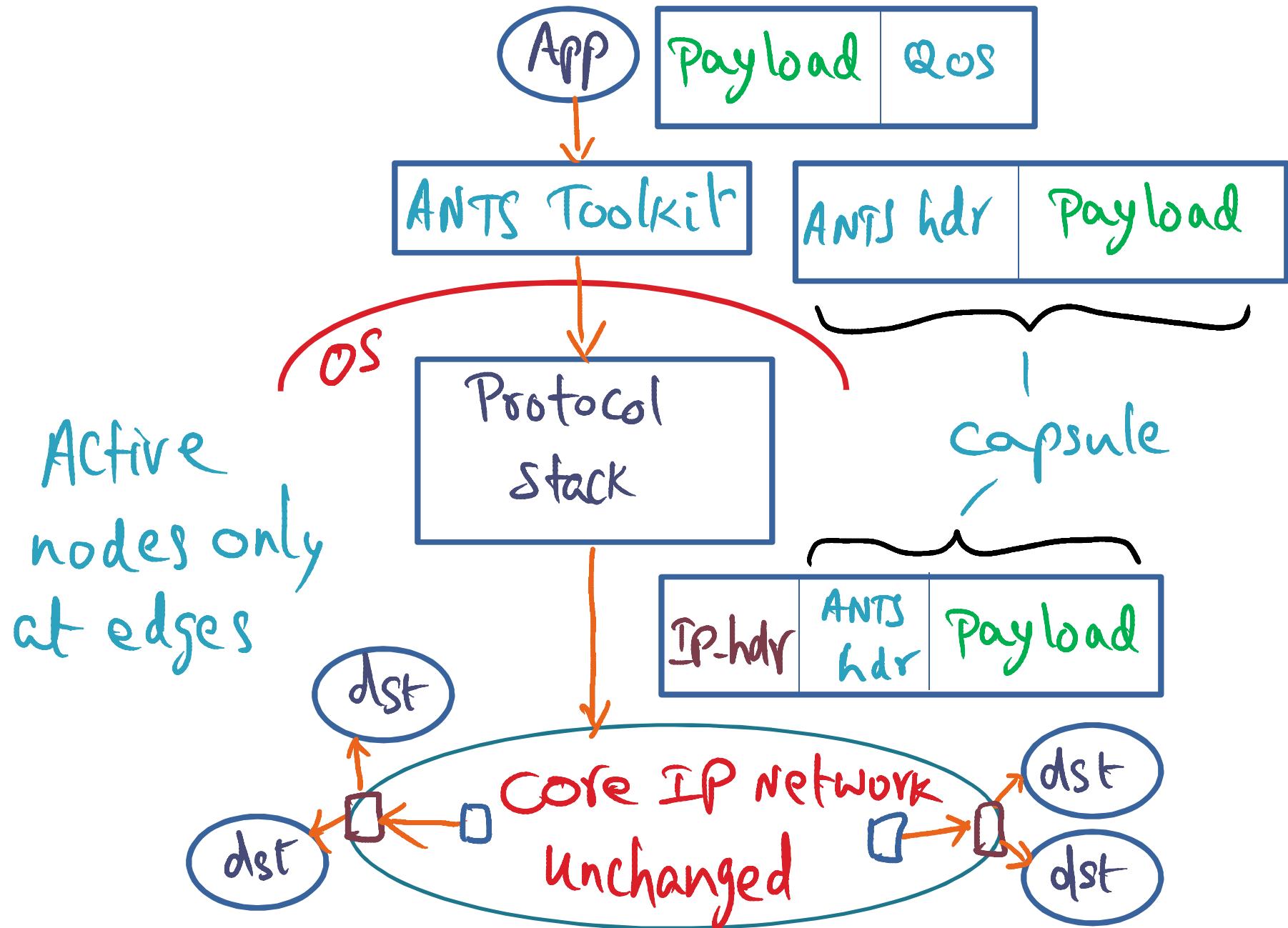
How to implement the vision?



ANTS Toolkit



ANTS Active Network



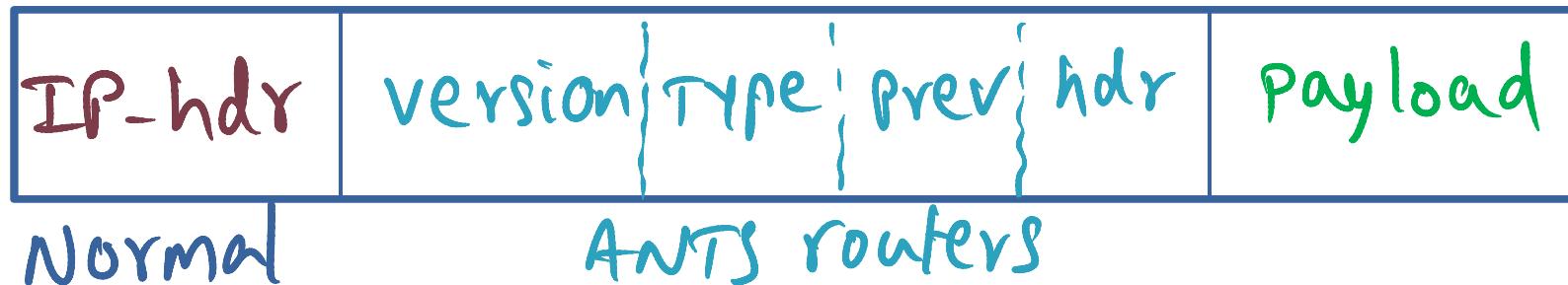
ANTS Capsule + API

← ANTS HDR →

IP-hdr	version TYPE prev hdr	Payload
Normal	ANTS routers	

ANTS Capsule + API

← ANTS HDR →



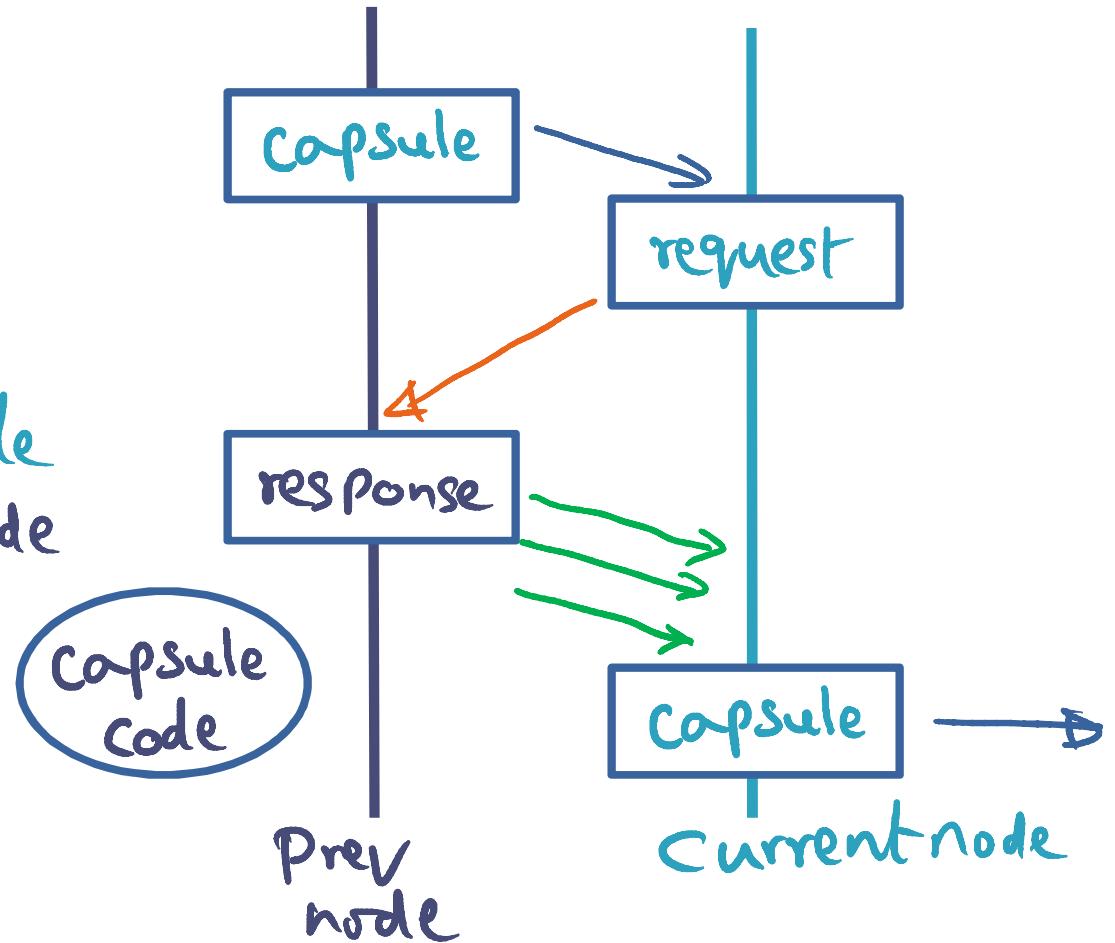
ANTS API

Method	Description
<code>int getAddress()</code>	Get local node address
<code>ChannelObject getChannel()</code>	Get receive channel
<code>Extension findExtension(String ext)</code>	Locate extended services
<code>long time()</code>	Get local time
<code>Object put(Object key, Object val, int age)</code>	Put object in soft-store
<code>Object get(Object key)</code>	Get object from soft-store
<code>Object remove(Object key)</code>	Remove object from soft-store
<code>void routeForNode(Capsule c, int n)</code>	Send capsule towards node
<code>void deliverToApp(Capsule c, int a)</code>	Deliver capsule to local application
<code>void log(String msg)</code>	Log a debugging message

Capsule Implementation

Action on Capsule arrival

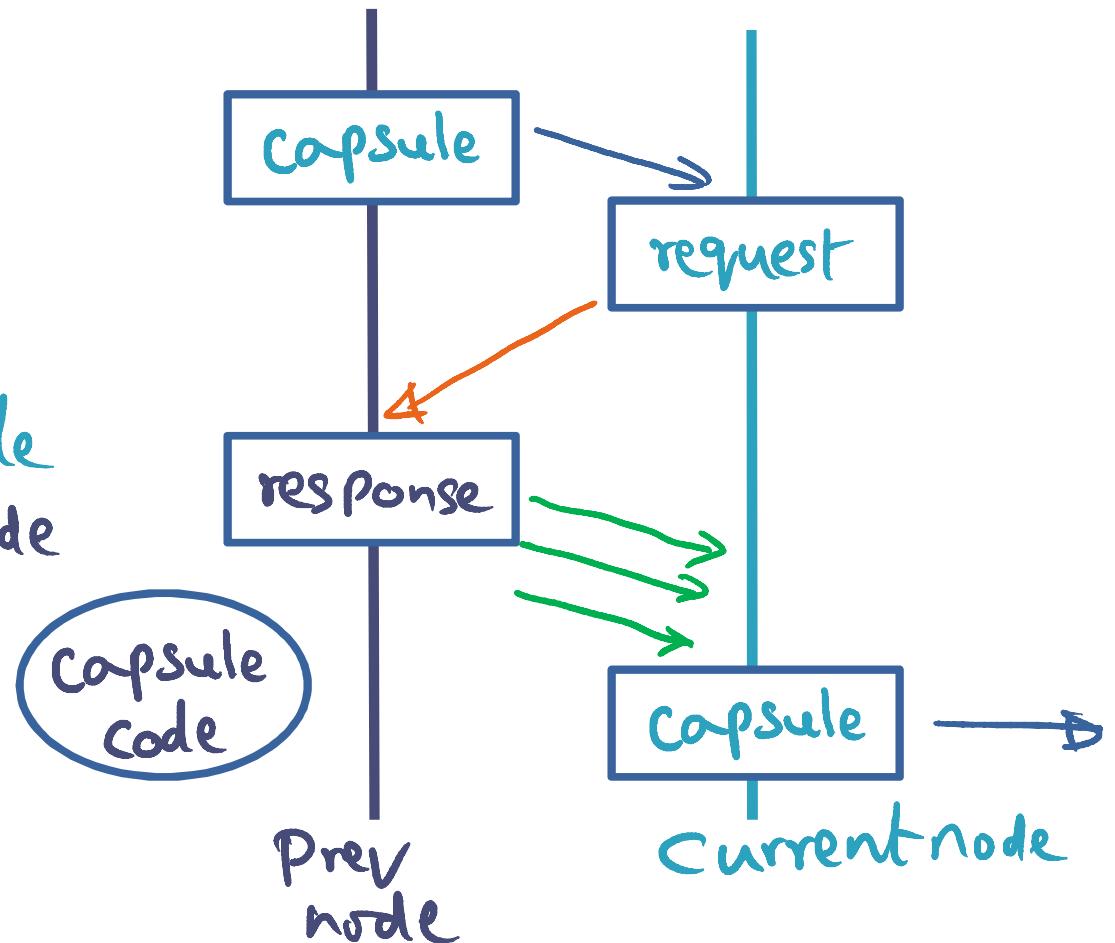
- type field:fingerprint
for capsule code
- demand load Capsule
code from prev node
by sending request



Capsule Implementation

Action on Capsule arrival

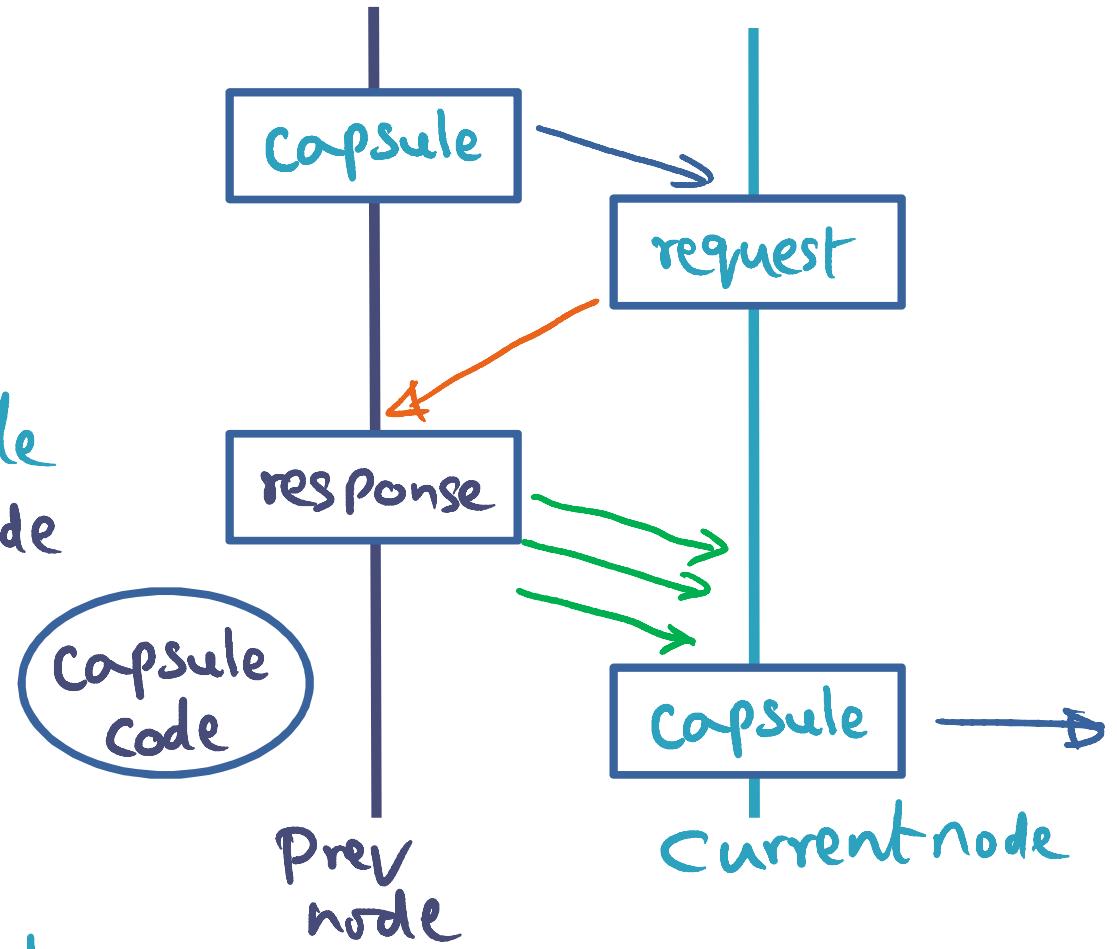
- type field: fingerprint for capsule code
- demand load Capsule code from prev node by sending request
- save in soft store for future use



Capsule Implementation

Action on Capsule arrival

- type field: fingerprint for capsule code
- demand load Capsule code from prev node by sending request
- save in soft store for future use
- drop capsule if code not in soft store or prev node



Potential Apps

Protocol independent multicast

Reliable multicast

Congestion notification

Private IP (PIP)

Anycasting

Pros & cons of Active Networks

Pro

- flexibility from APP perspective

Pros & cons of Active Networks

Pro

- flexibility from App perspective

Cons

- Protection threats

- * ANTS runtime safety \Rightarrow Java sandboxing
- * code spoofing \Rightarrow robust fingerprint
- * soft state integrity \Rightarrow restricted API

Pros & cons of Active Networks

Pro

- flexibility from APP perspective

Cons

- Protection threats

- * ANTS runtime safety \Rightarrow Java sandboxing
- * code spoofing \Rightarrow robust fingerprint
- * soft state integrity \Rightarrow restricted API

- Resource management threats

- * At each node \Rightarrow restricted API

- * Flooding the network \Rightarrow Internet already susceptible !!

roadblocks to the Active Networks vision?

roadblocks to the Active Networks vision?

- * Need buy in from router vendors
- * ANTS software routing cannot match throughput needed in Internet core

Feasible?

Router makers loath to opening up the network

⇒ only feasible at the edge of the network

Software routing cannot match hardware routing

⇒ only feasible at the edge of the network

Social + psychological reasons

⇒ hard for user community to accept
arbitrary code executing in the public
routing fabric

Key Takeaways

- Active Networks was ahead of its time and there was not a “killer app” to justify it. Further, it focused more on safety and less on performance. So in the 90's it seemed like a solution looking for a problem.
- Difficulties with network management, rise of virtualization, right hardware support, data center and cloud computing have all given Active Networks a new lease of life in the form of Software Defined Networking (SDN).
- Specifically, cloud computing promotes a model of utility computing where multiple tenants (i.e., businesses) can host their respective corporate networks simultaneously on the same computational resources of a data center. Imagine Coke and Pepsi using the same data center resources....
- What this means is a need for perfect isolation of the network traffic of one business from another even though each of their network traffic is flowing on the same physical infrastructure. This calls for virtualization of the physical network, hence the term software defined networking.