Computer Networks Overview, Day 2

Mark Allman
mark.allman@case.edu

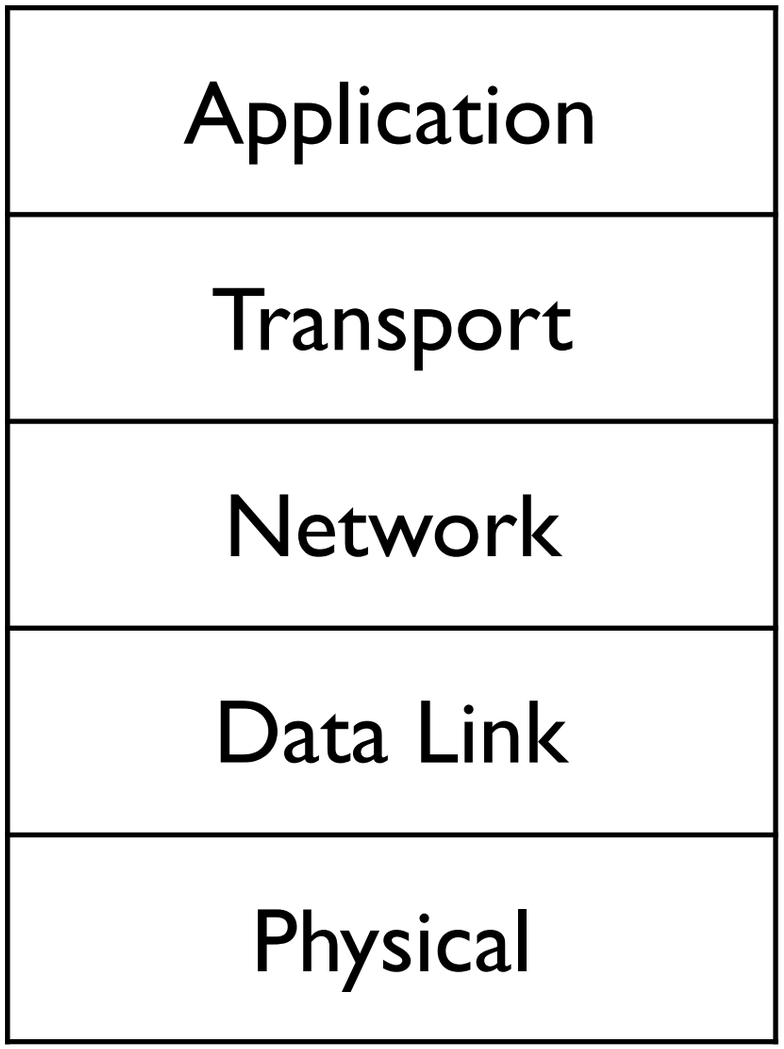
EECS 325/425
Fall 2018

"Keep your eyes on the road, your hands upon the wheel ..."

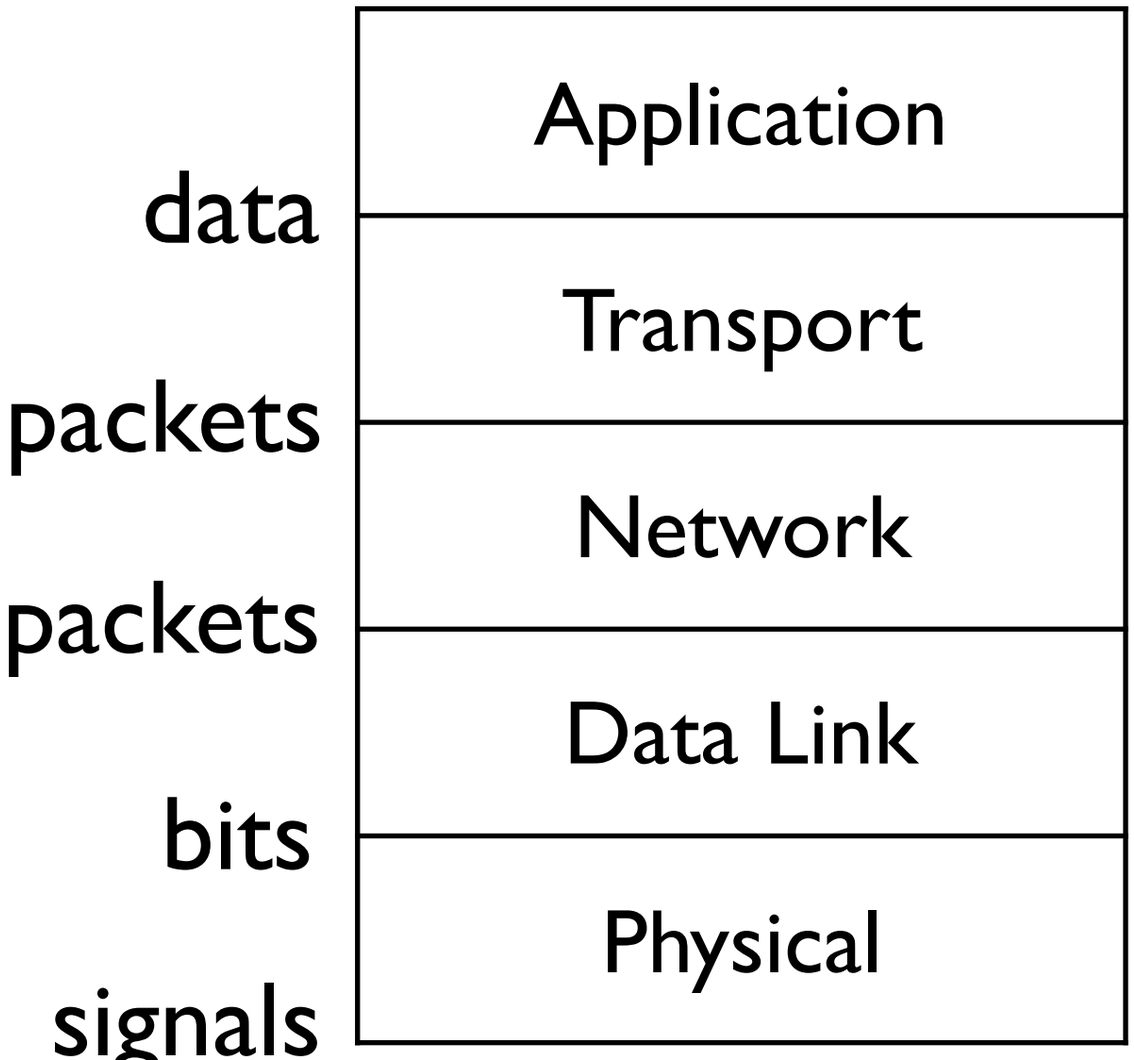
All material copyright 2011-2018
Mark Allman, All rights reserved.

Layering

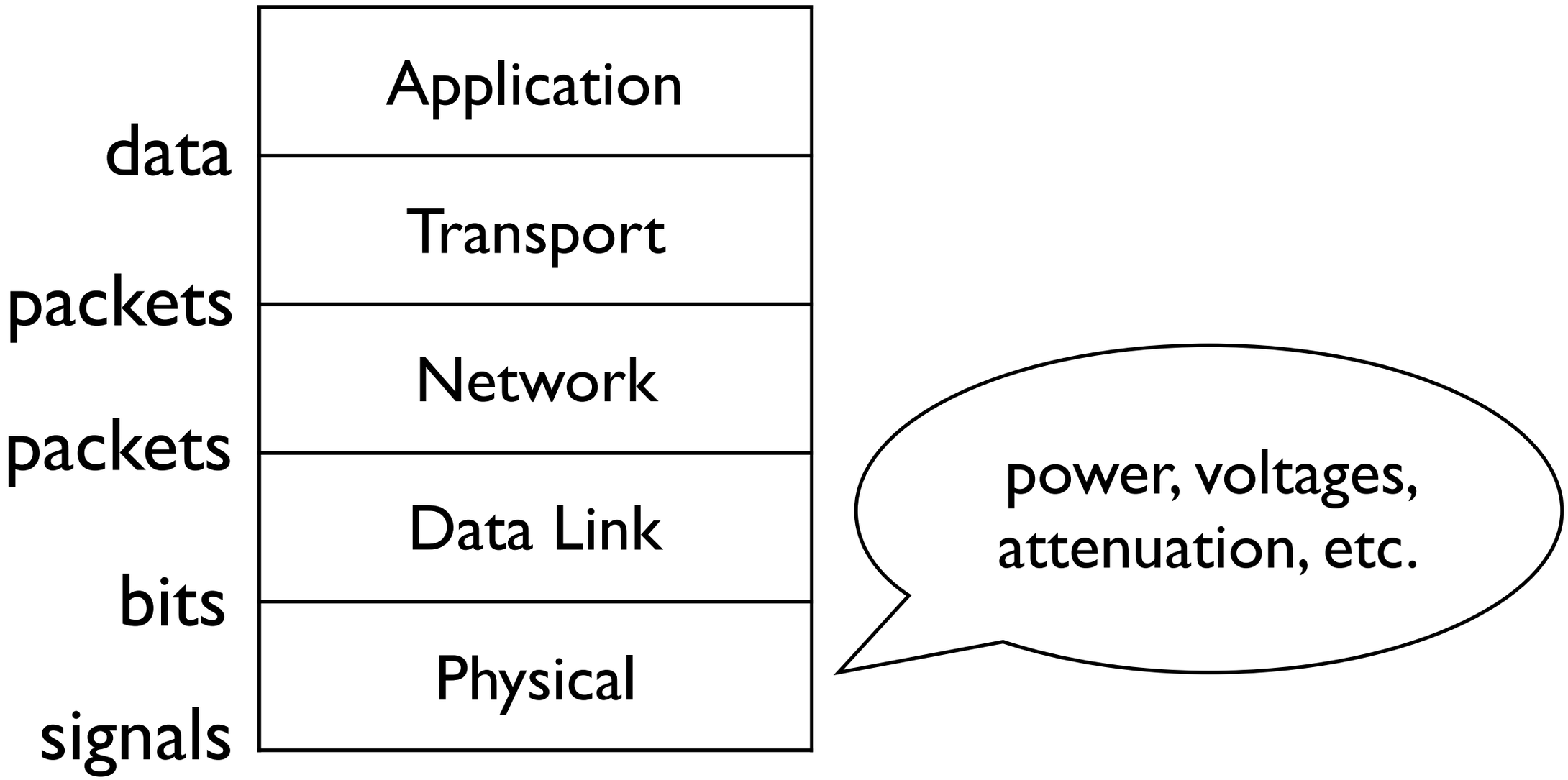
Layering



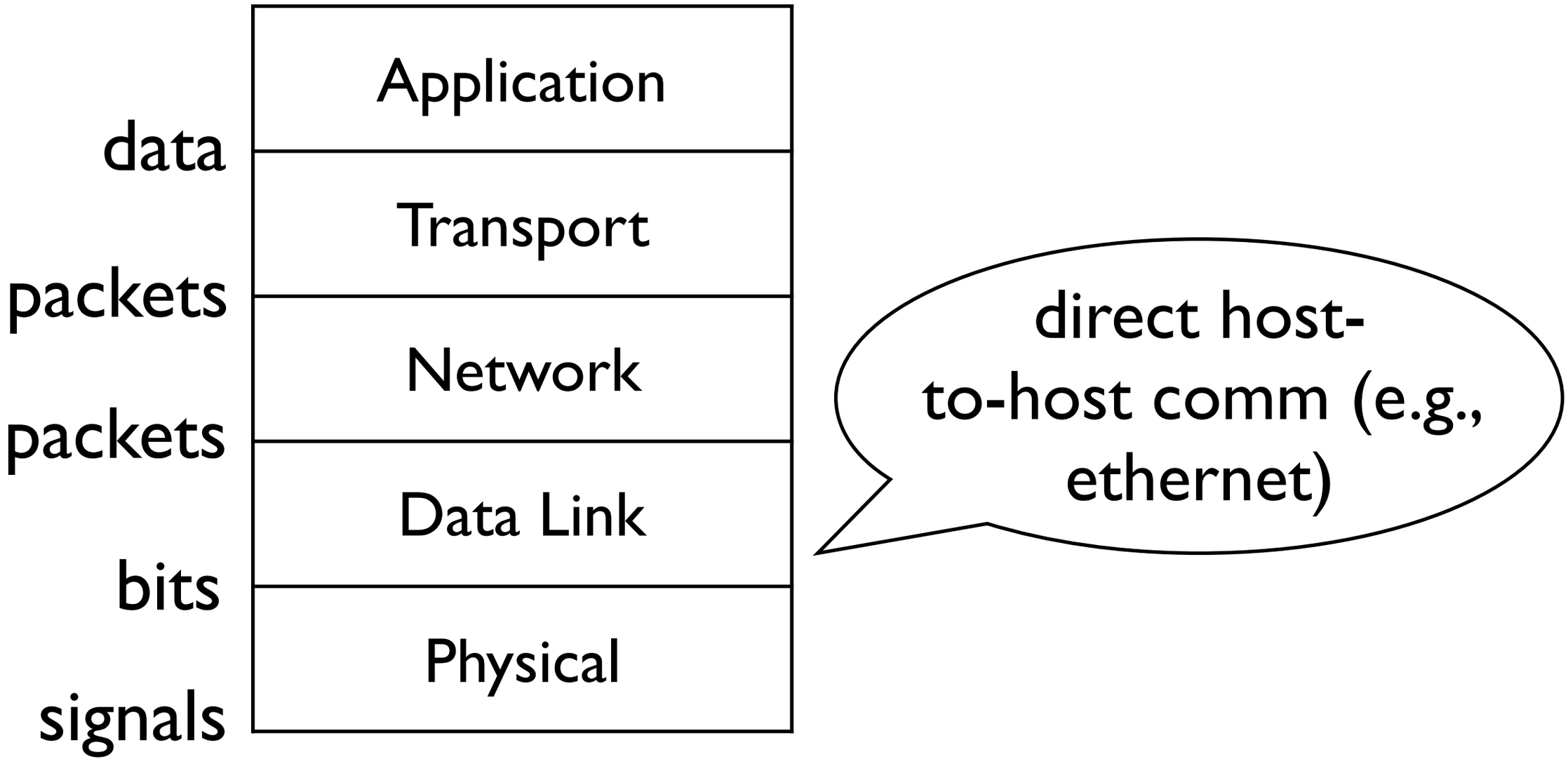
Layering


Application
Transport
Network
Data Link
Physical
data
packets
packets
bits
signals

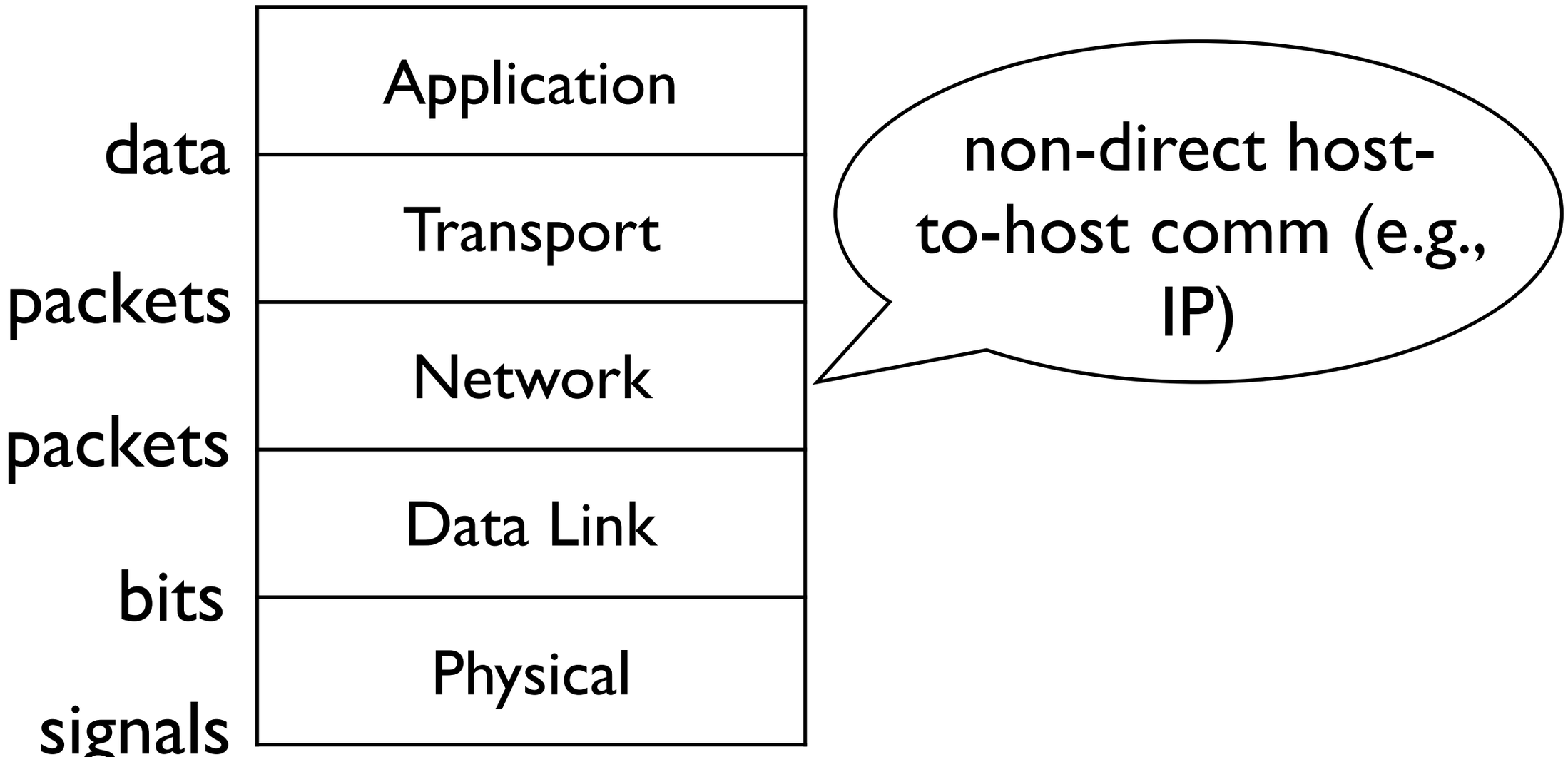
Layering


Application
Transport
Network
Data Link
Physical
data
packets
packets
bits
signals
power, voltages,
attenuation, etc.

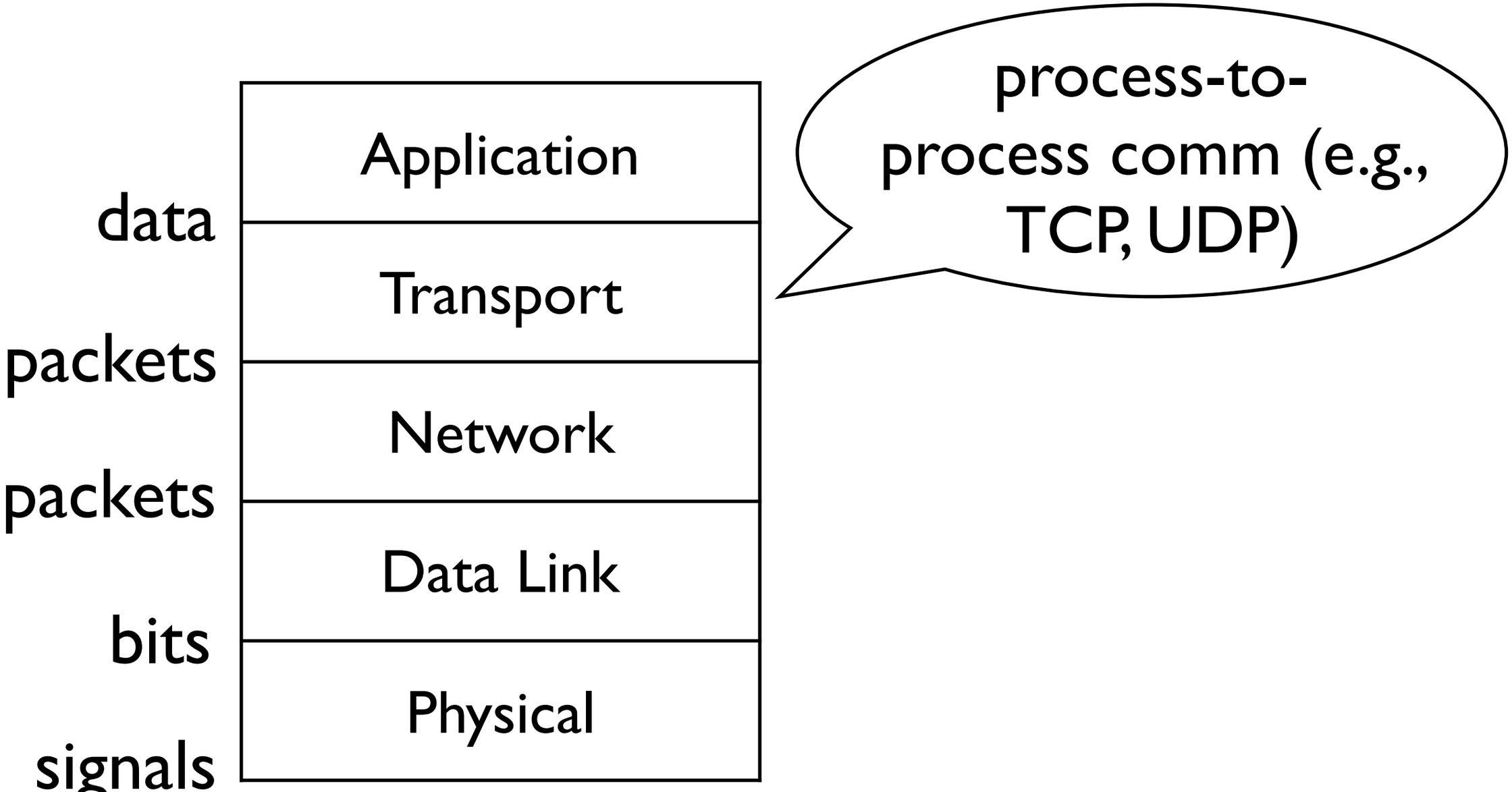
Layering


Application
Transport
Network
Data Link
Physical
data
packets
packets
bits
signals
direct host-to-host comm (e.g., ethernet)

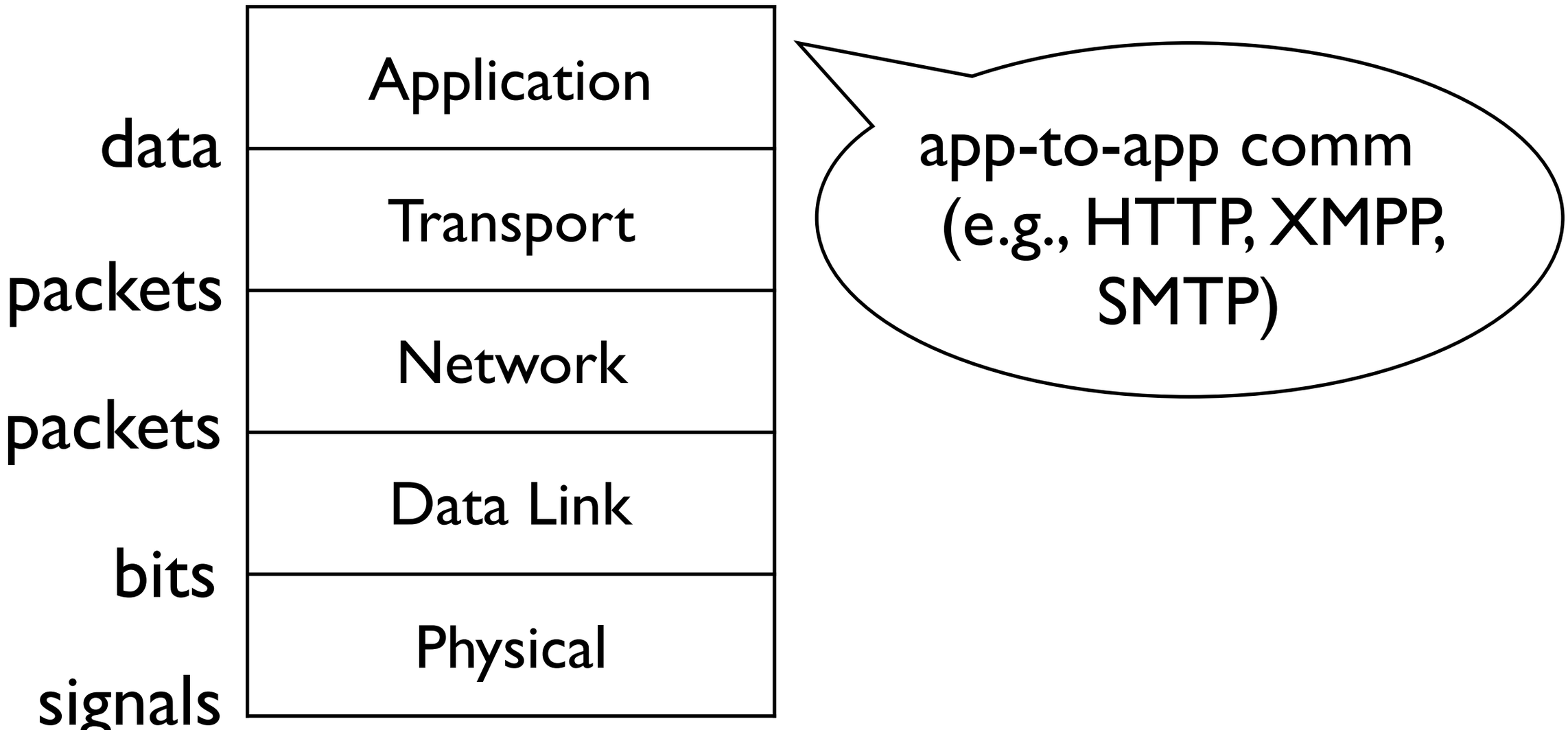
Layering


Application
Transport
Network
Data Link
Physical
data
packets
packets
bits
signals
non-direct host-to-host comm (e.g., IP)

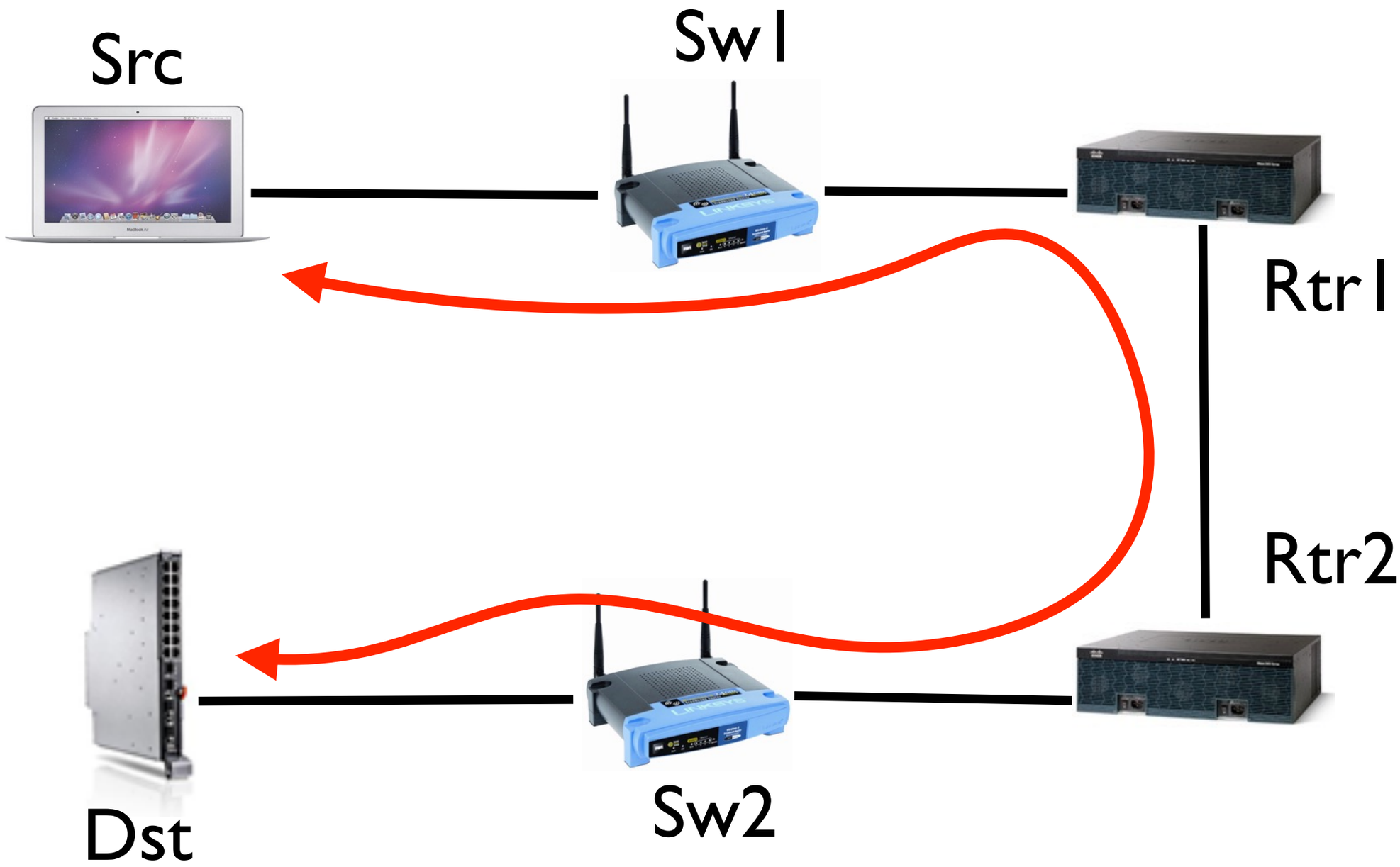
Layering


Application
Transport
Network
Data Link
Physical
data
packets
bits
signals
process-to-process comm (e.g., TCP, UDP)

Layering


Application
Transport
Network
Data Link
Physical
data
packets
packets
bits
signals
app-to-app comm
(e.g., HTTP, XMPP,
SMTP)

Sample Network Path



Layering, Take 2

Layering, Take 2

Src

Dst

Layering, Take 2

Src

- App
- Trans
- Net
- Link
- Phys

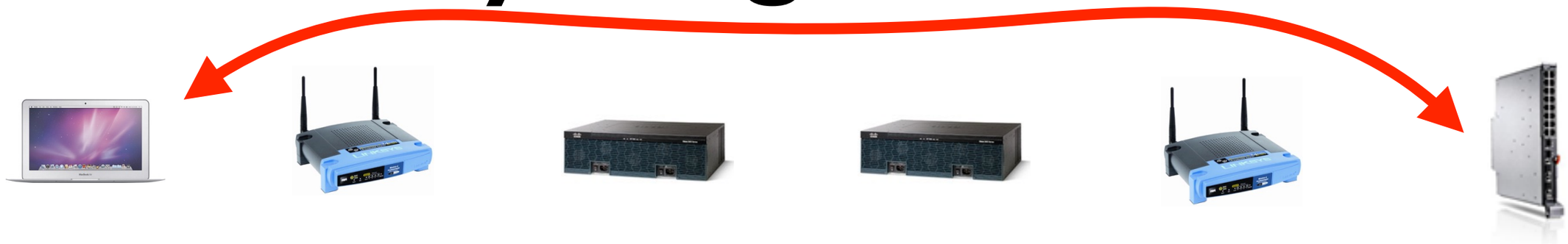
Dst

- App
- Trans
- Net
- Link
- Phys

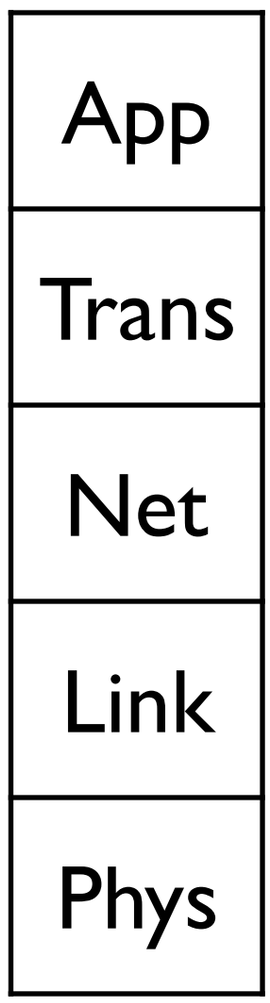
Allman

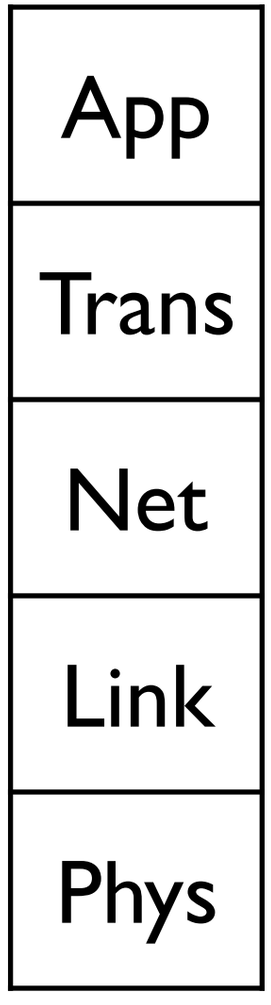
5

Layering, Take 2

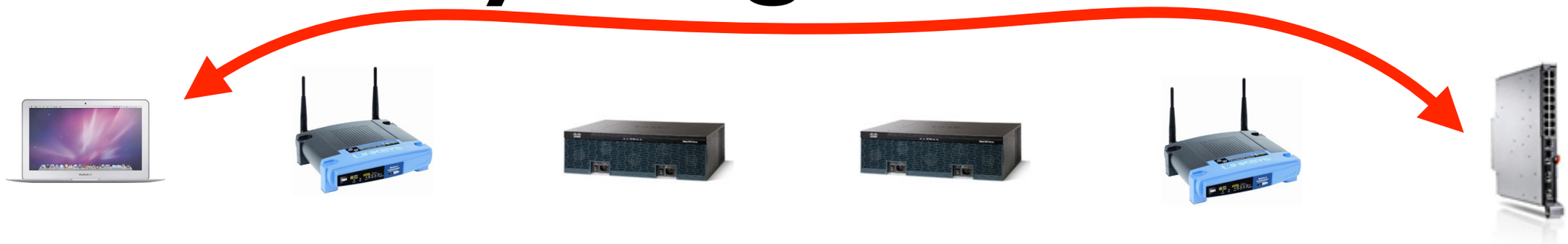


Src Sw1 Rtr1 Rtr2 Sw2 Dst

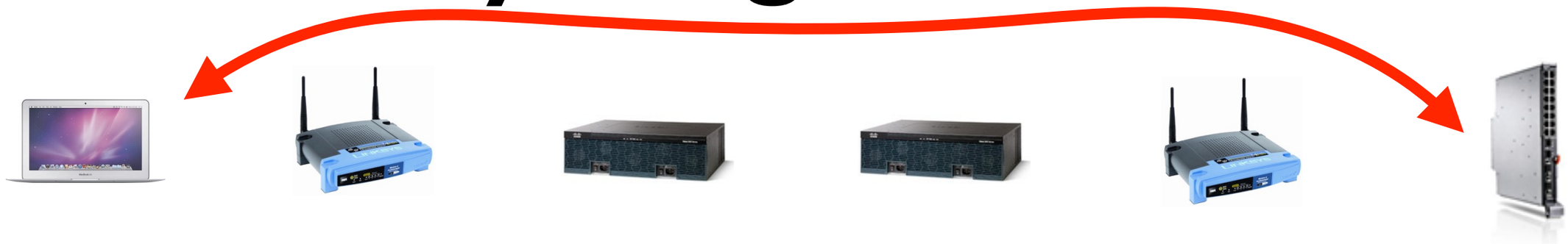




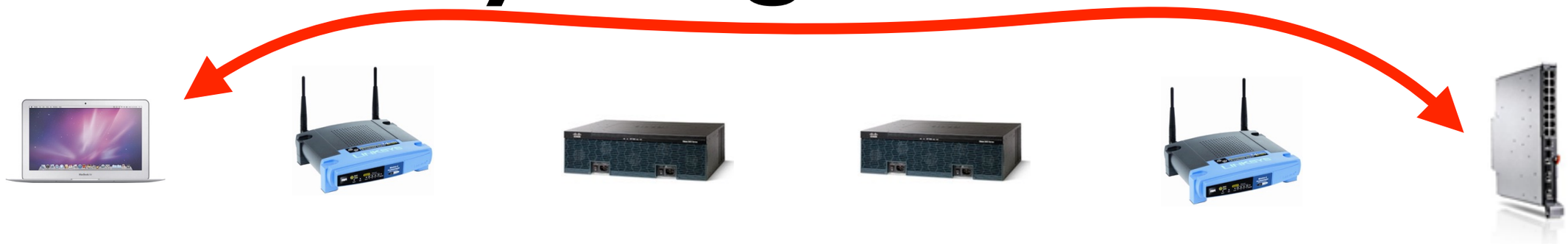
Layering, Take 2



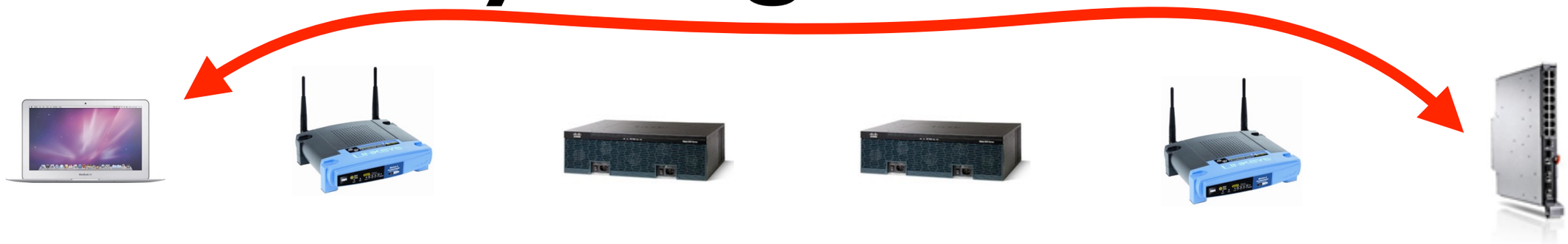
Layering, Take 2



Layering, Take 2



Layering, Take 2



Layering, Take 2

Layering, Take 2

Layering, Take 2

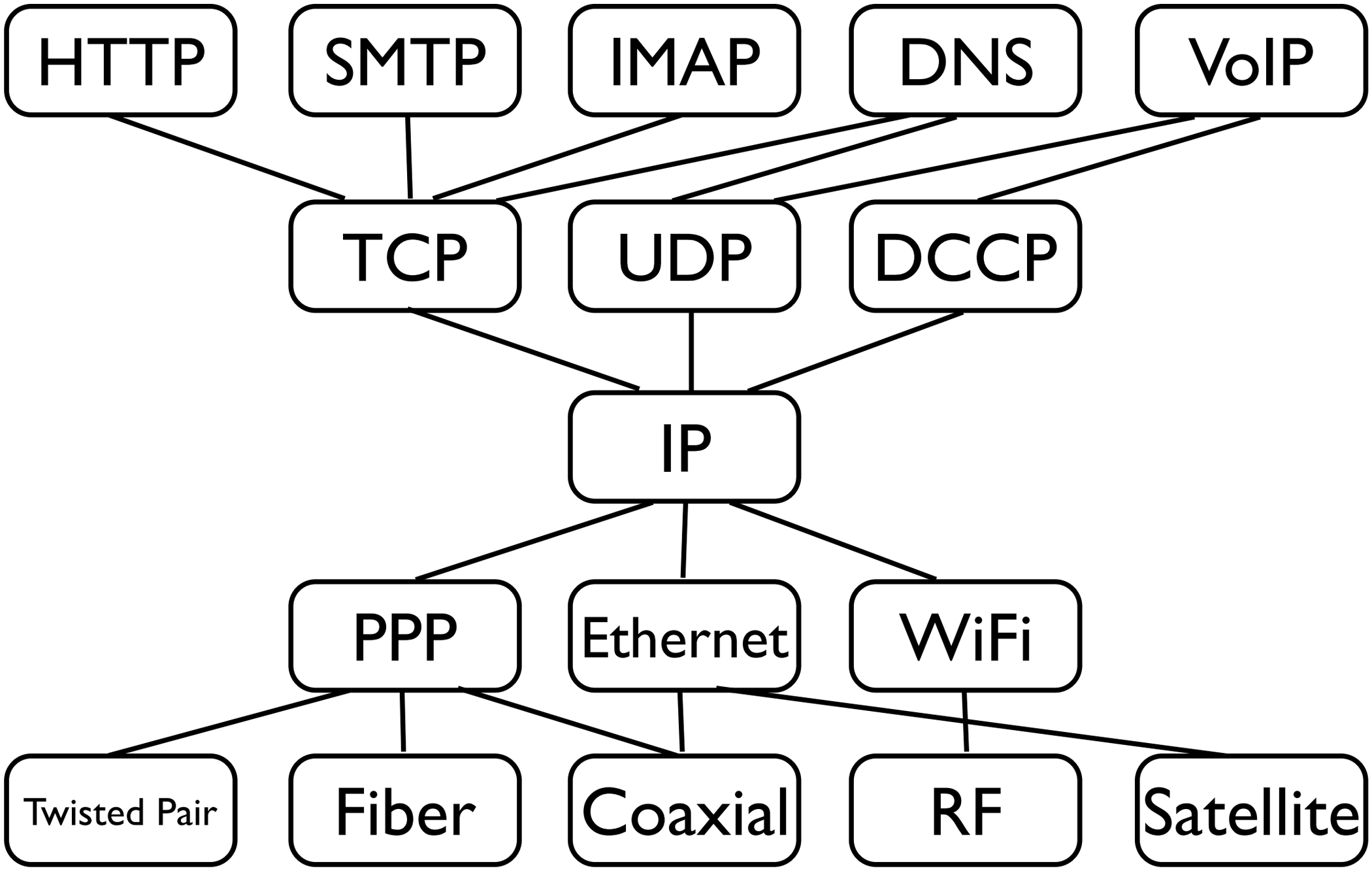
Layering, Take 2

Layering, Take 2

Layering, Take 2

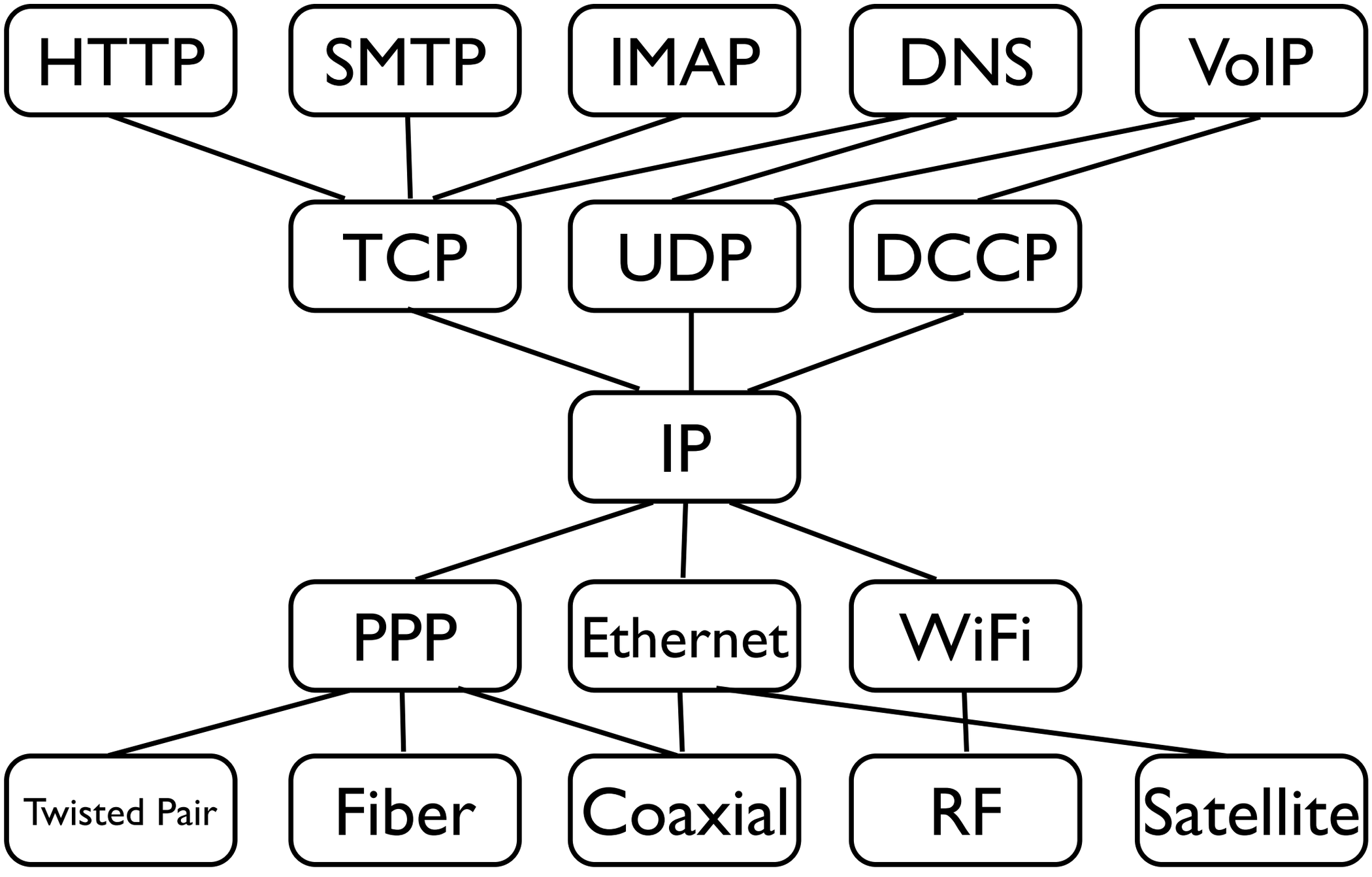
Layering, Take 2

Layering, Take 3

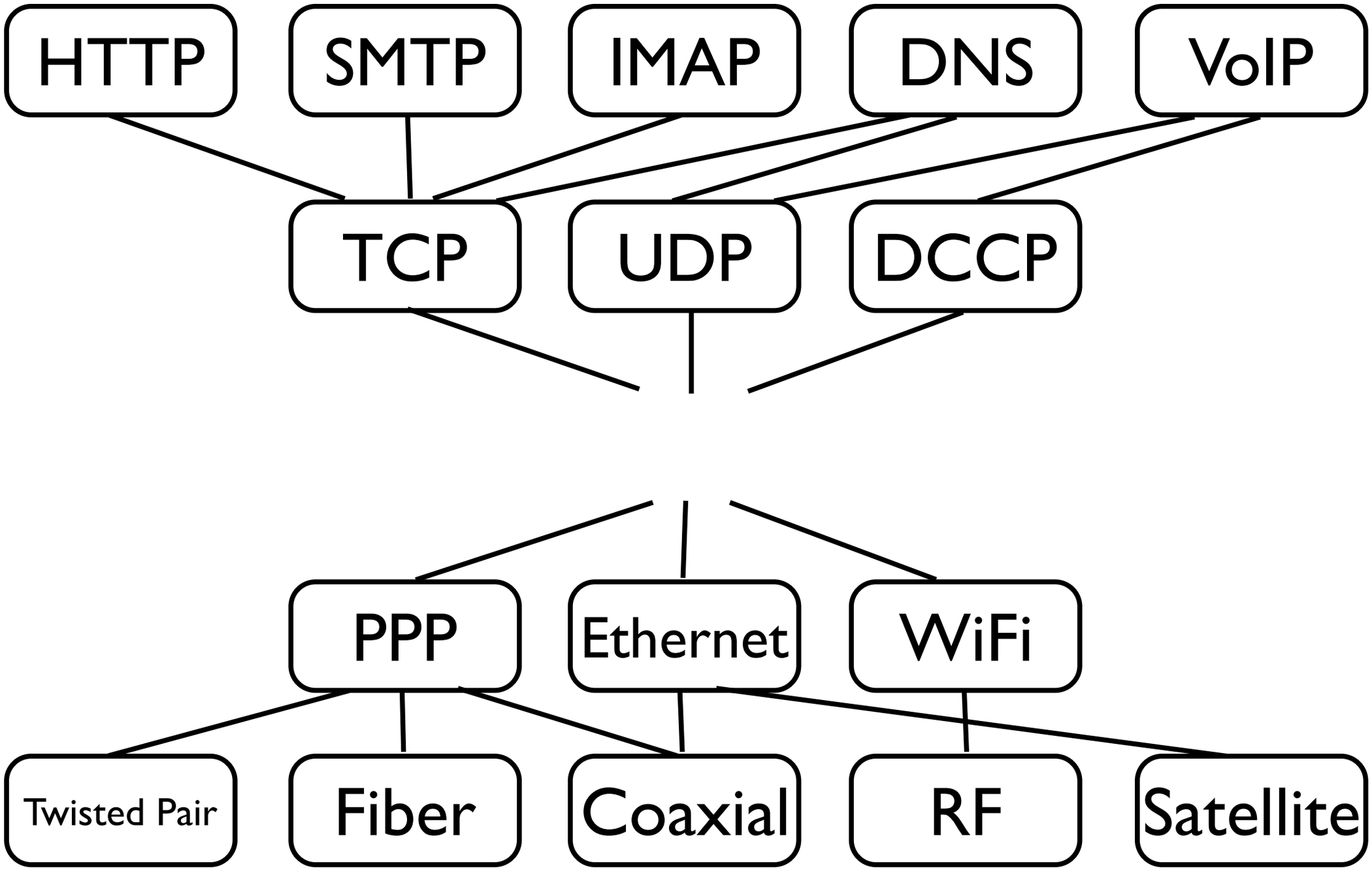


Layering, Take 3

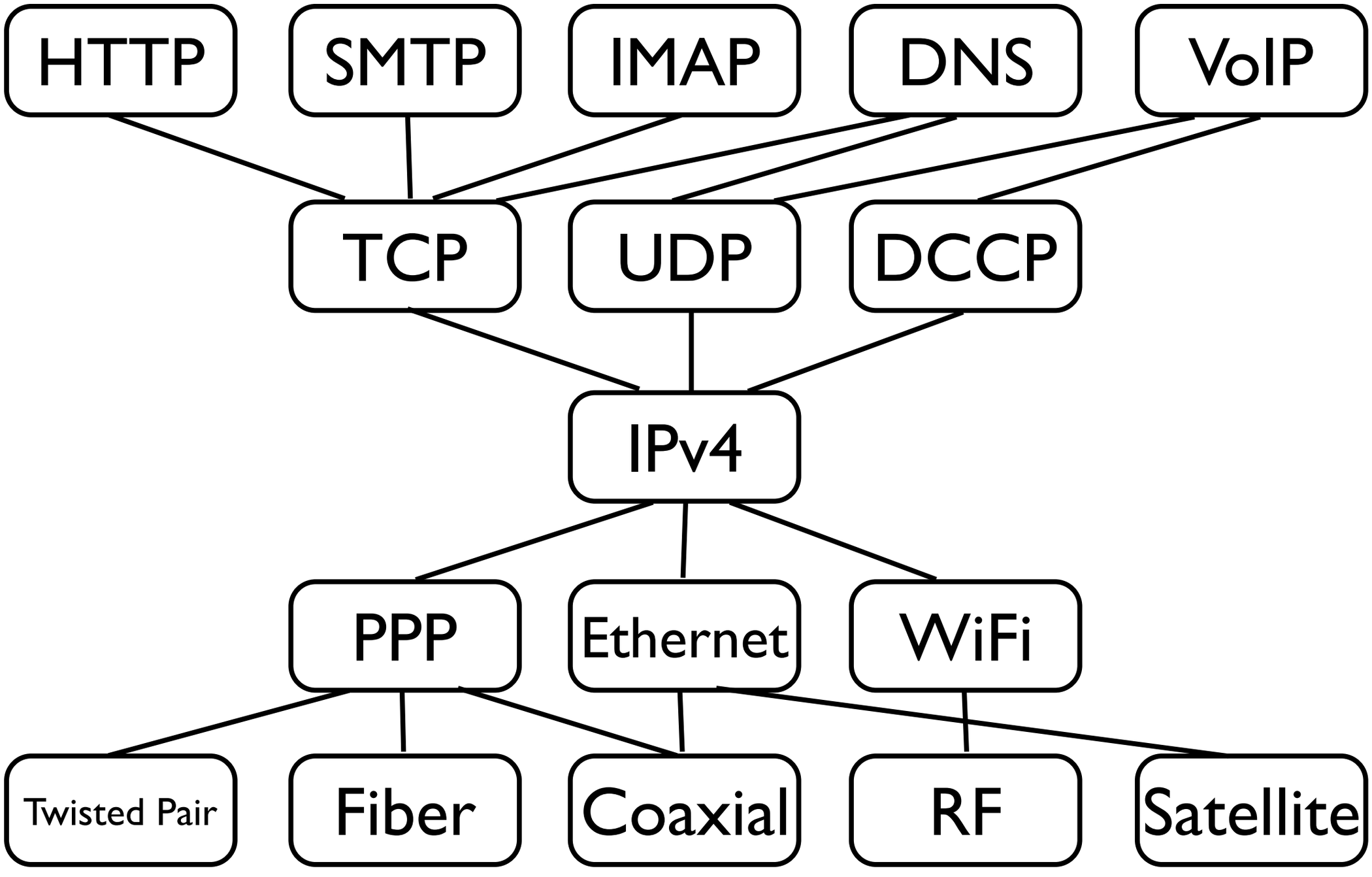
Layering, Take 3



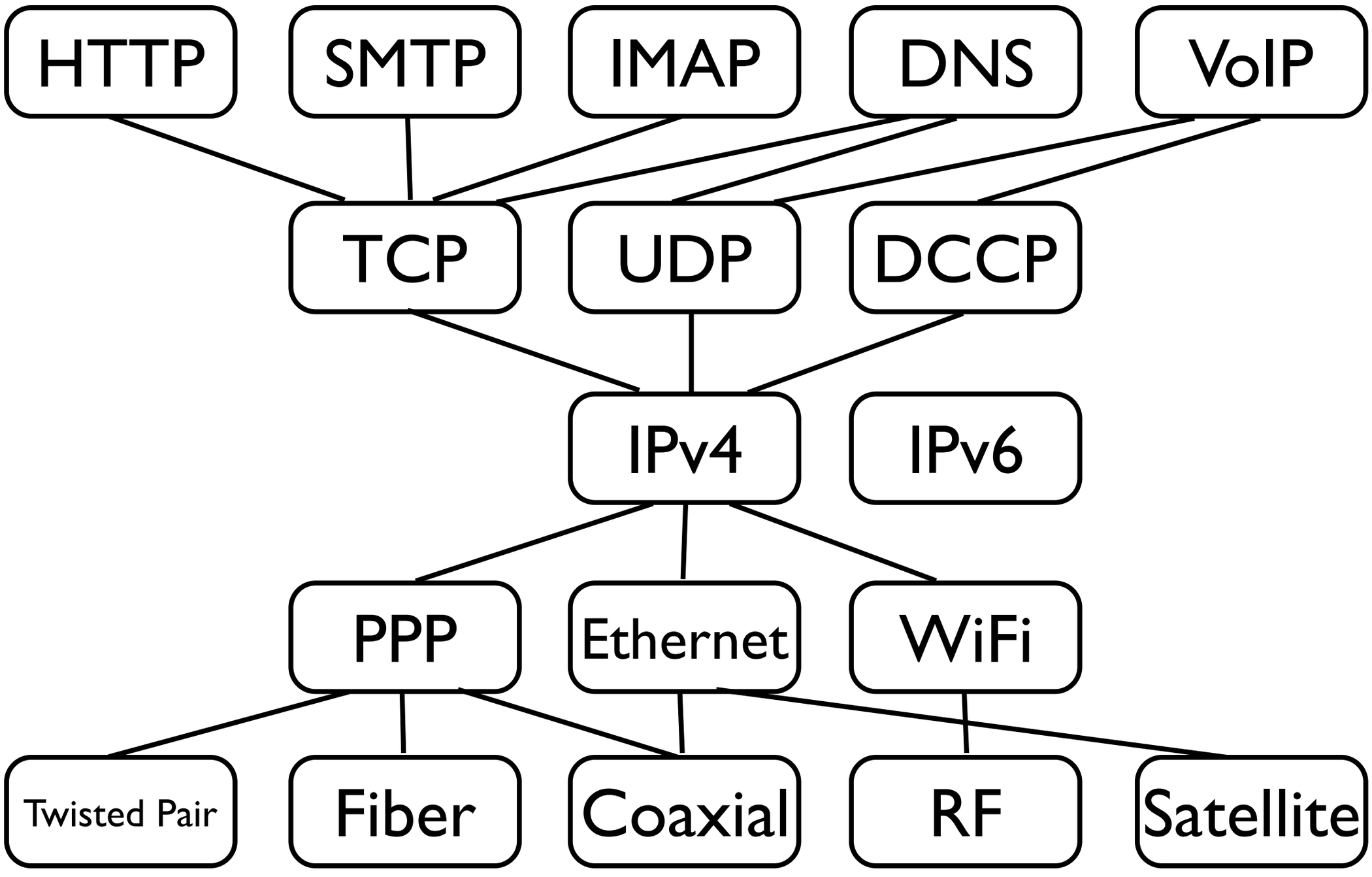
Layering, Take 3



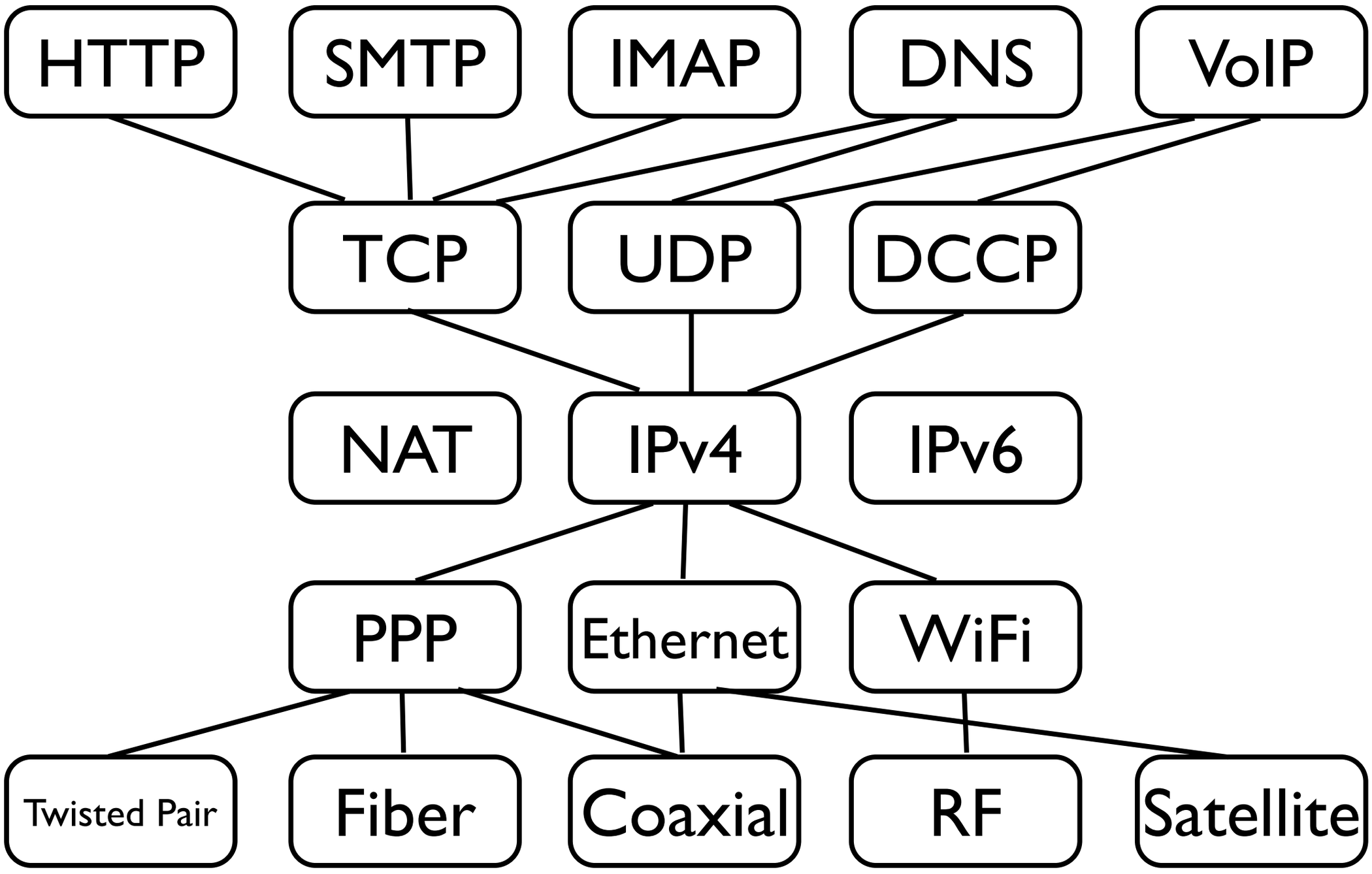
Layering, Take 3



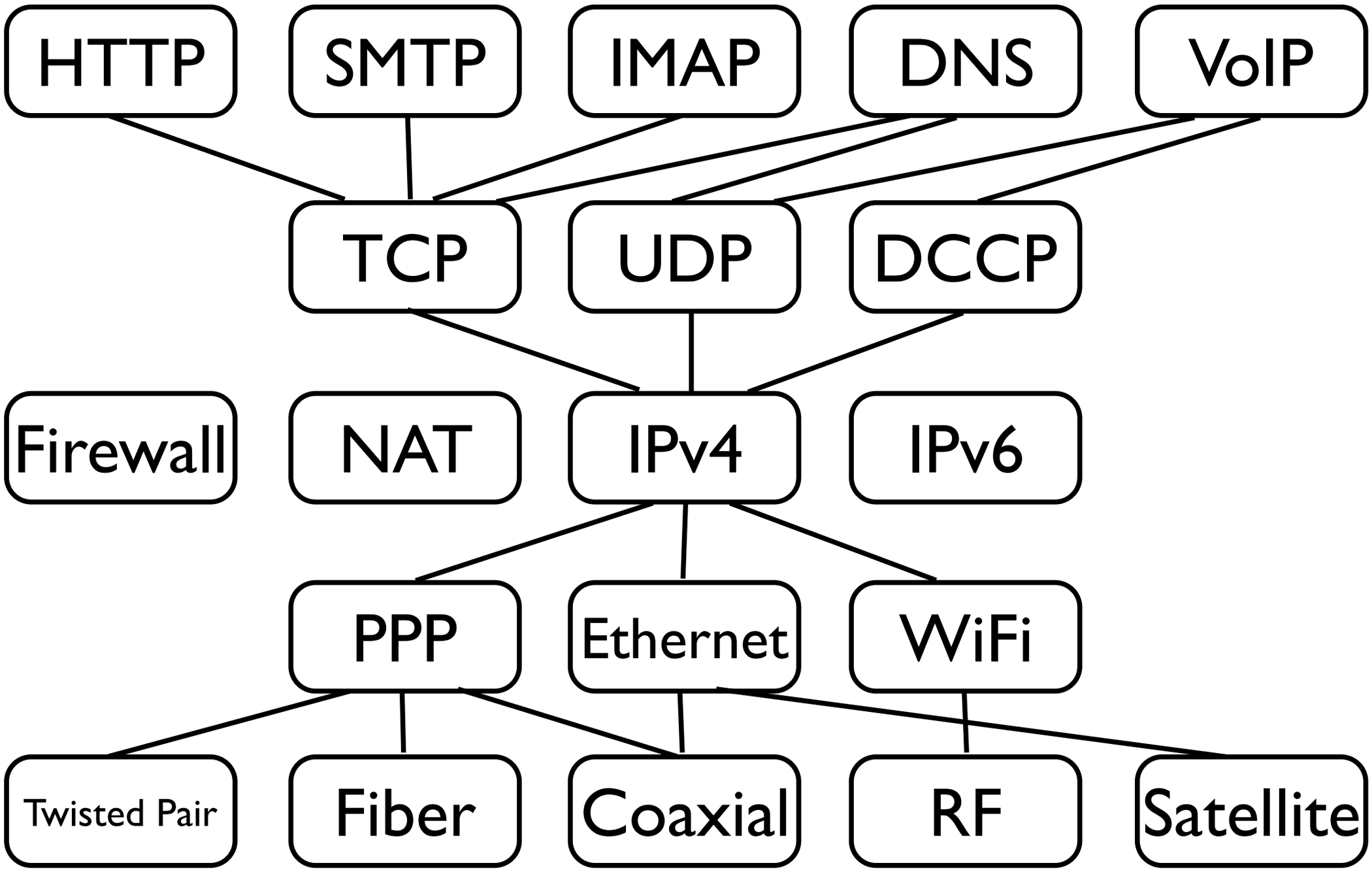
Layering, Take 3



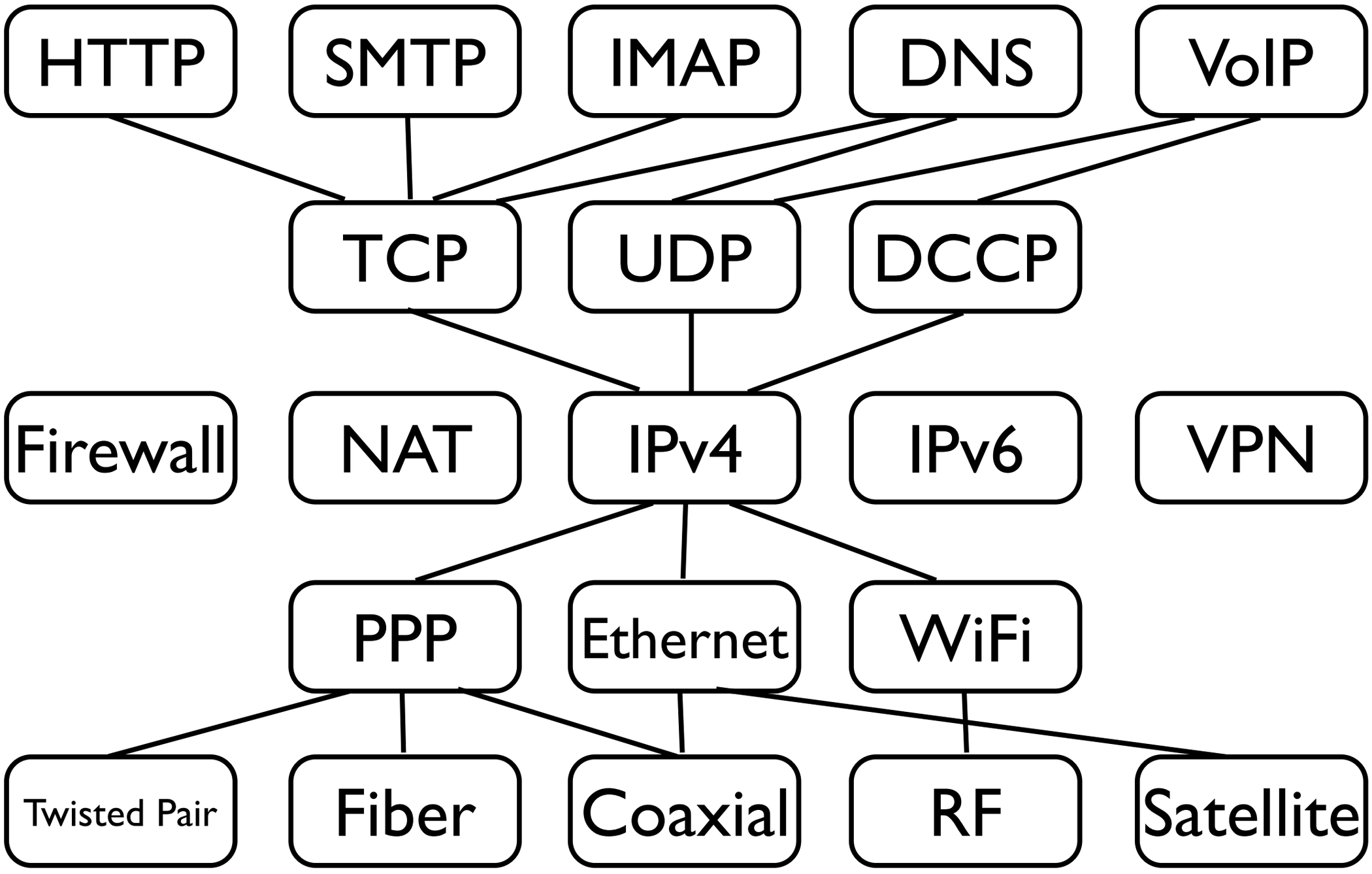
Layering, Take 3



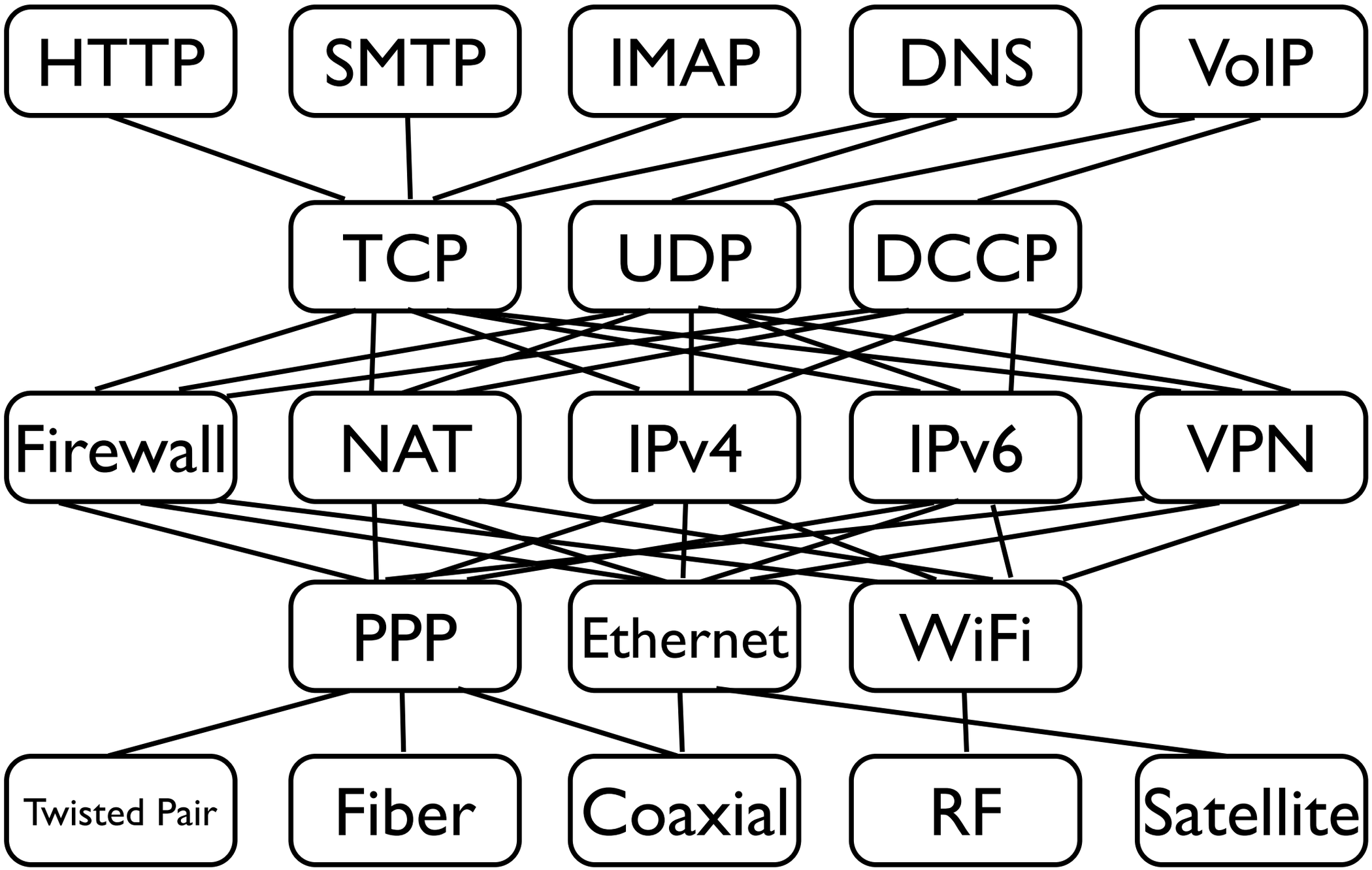
Layering, Take 3



Layering, Take 3



Layering, Take 3



Protocols

Protocols

- What is a *protocol*?

Protocols (cont.)

Protocols (cont.)

- "What time is it?"

Protocols (cont.)

- "What time is it?"
  - "2:23"

Protocols (cont.)

- "What time is it?"
  - "2:23"
  - "2:23pm"

Protocols (cont.)

- "What time is it?"
  - "2:23"
  - "2:23pm"
  - "14:23"

Protocols (cont.)

- "What time is it?"
  - "2:23"
  - "2:23pm"
  - "14:23"
  - "2:23pm EDT"

Protocols (cont.)

- "What time is it?"
  - "2:23"
  - "2:23pm"
  - "14:23"
  - "2:23pm EDT"
  - 1314383281

Protocols (cont.)

- "What time is it?"
  - "2:23"
  - "2:23pm"
  - "14:23"
  - "2:23pm EDT"
  - 1314383281
  - "tea time"

Protocols (cont.)

- "What time is it?"
  - "2:23"
  - "2:23pm"
  - "14:23"
  - "2:23pm EDT"
  - 1314383281
  - "tea time"

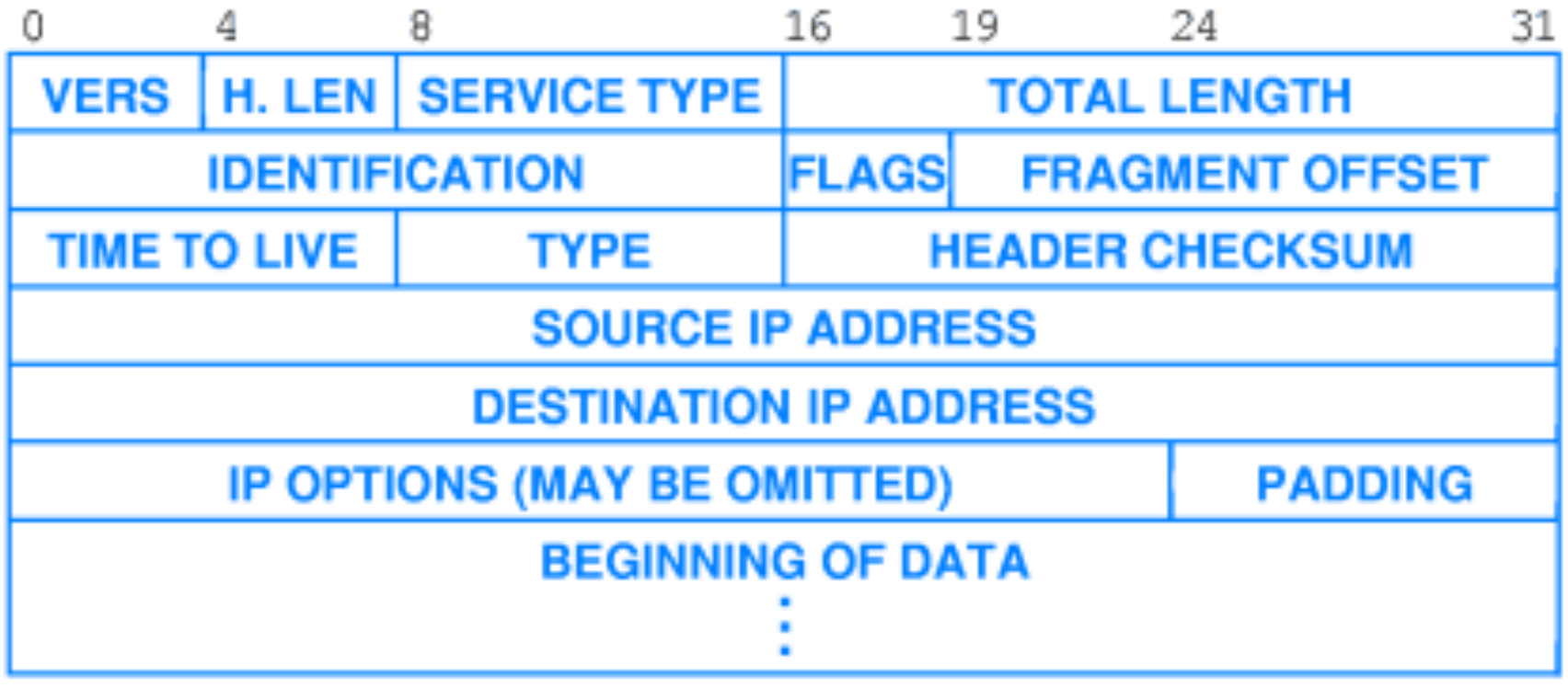
- "¿Qué hora es"
  - "dos veintitrés"
  - "Son las cinco en alguna parte"

Protocols (cont.)

- A protocol is simply a set of rules for communicating in a given fashion, including:
  - the syntax of the information exchange
  - the semantics of the information
  - the methods of arriving at the informations

Protocols (cont.)

- Example: IP



Protocols (cont.)

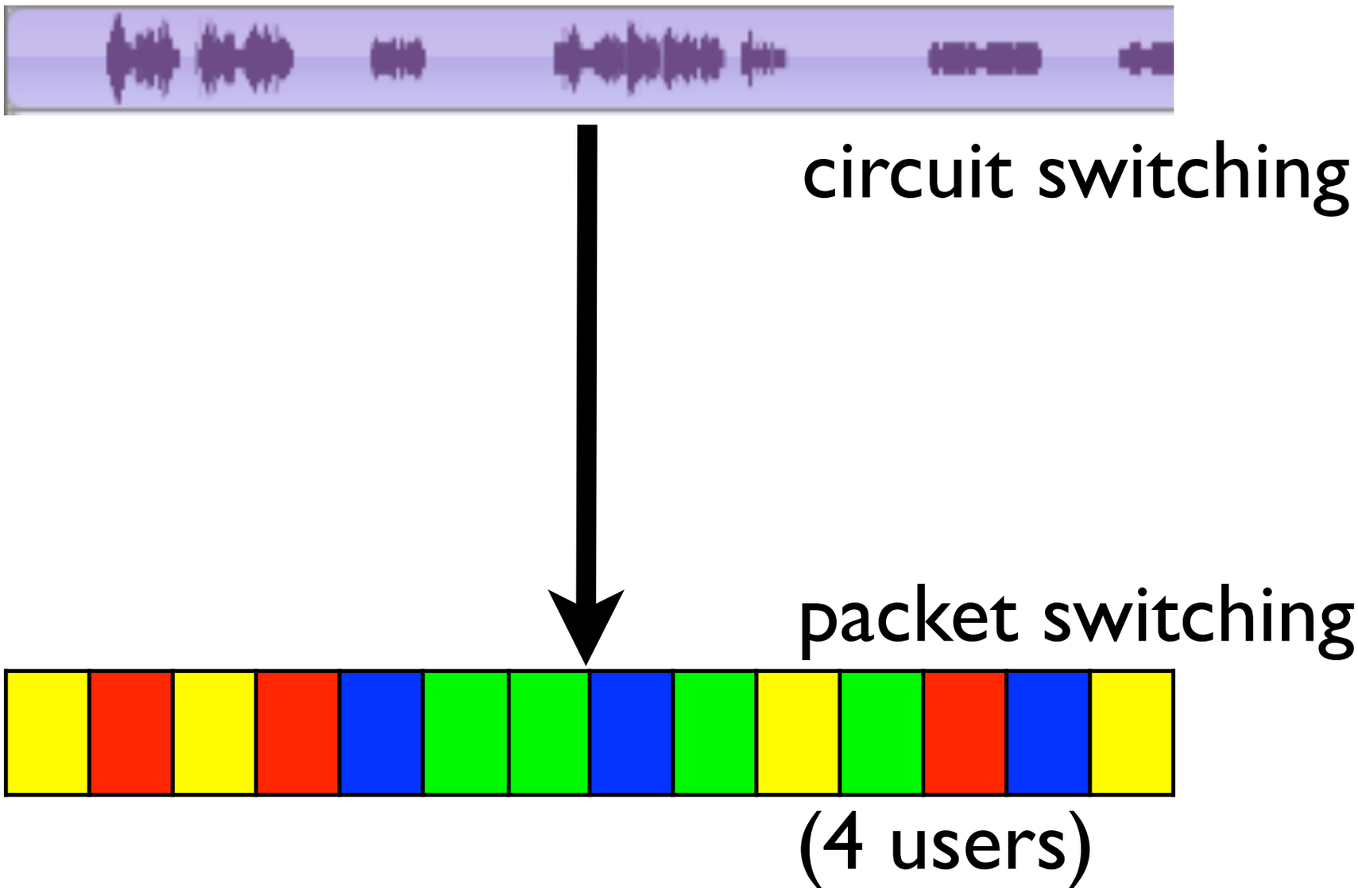
- Example: HTTP

```
> GET http://www.icir.org:80/ HTTP/1.1
> Connection: Keep-Alive
> User-Agent: hprobe 1.0 (mallman@lerc.nasa.gov)
> Host: www.icir.org:80
>
< HTTP/1.1 200 OK
< Date: Fri, 26 Aug 2011 15:36:38 GMT
< Server: Apache/2.2.15 (Fedora)
< Last-Modified: Wed, 23 Feb 2011 23:32:02 GMT
< ETag: "14401e-ee4-49cfb8102a080"
< Accept-Ranges: bytes
< Content-Length: 3812
< Connection: close
< Content-Type: text/html; charset=UTF-8
<
< [PAGE CONTENT]
```

The Big Picture

- Putting it all together:
  - Baran flipped our view from circuits to *packets*
  - We then organized functionality into *layers*
  - We have *protocols* that instantiate layered functionality

Circuits vs. Packets



Building Packets

```
HTTP/1.1 200 OK
Date: Fri, 26 Aug 2011 15:36:38 GMT
Server: Apache/2.2.15 (Fedora)
Last-Modified: Wed, 23 Feb 2011 23:32:02 GMT
ETag: "14401e-ee4-49cfb8102a080"
Accept-Ranges: bytes
Content-Length: 3812
Connection: close
Content-Type: text/html; charset=UTF-8

[3,812 BYTES OF PAGE CONTENT]
```

Building Packets

```
HTTP/1.1 200 OK
Date: Fri, 26 Aug 2011 15:36:38 GMT
Server: Apache/2.2.15 (Fedora)
Last-Modified: Wed, 23 Feb 2011 23:32:02 GMT
ETag: "14401e-ee4-49cfb8102a080"
Accept-Ranges: bytes
Content-Length: 3812
Connection: close
Content-Type: text/html; charset=UTF-8

[3,812 BYTES OF PAGE CONTENT]
```

Building Packets

```
HTTP/1.1 200 OK
Date: Fri, 26 Aug 2011 15:36:38 GMT
Server: Apache/2.2.15 (Fedora)
Last-Modified: Wed, 23 Feb 2011 23:32:02 GMT
ETag: "14401e-ee4-49cfb8102a080"
Accept-Ranges: bytes
Content-Length: 3812
Connection: close
Content-Type: text/html; charset=UTF-8

[3,812 BYTES OF PAGE CONTENT]
```

HTTP response header is 269 bytes

Building Packets

```
HTTP/1.1 200 OK
Date: Fri, 26 Aug 2011 15:36:38 GMT
Server: Apache/2.2.15 (Fedora)
Last-Modified: Wed, 23 Feb 2011 23:32:02 GMT
ETag: "14401e-ee4-49cfb8102a080"
Accept-Ranges: bytes
Content-Length: 3812
Connection: close
Content-Type: text/html; charset=UTF-8

[3,812 BYTES OF PAGE CONTENT]
```

Building Packets

```
HTTP/1.1 200 OK
Date: Fri, 26 Aug 2011 15:36:38 GMT
Server: Apache/2.2.15 (Fedora)
Last-Modified: Wed, 23 Feb 2011 23:32:02 GMT
ETag: "14401e-ee4-49cfb8102a080"
Accept-Ranges: bytes
Content-Length: 3812
Connection: close
Content-Type: text/html; charset=UTF-8

[3,812 BYTES OF PAGE CONTENT]
```

Building Packets

```
HTTP/1.1 200 OK
Date: Fri, 26 Aug 2011 15:36:38 GMT
Server: Apache/2.2.15 (Fedora)
Last-Modified: Wed, 23 Feb 2011 23:32:02 GMT
ETag: "14401e-ee4-49cfb8102a080"
Accept-Ranges: bytes
Content-Length: 3812
Connection: close
Content-Type: text/html; charset=UTF-8

[3,812 BYTES OF PAGE CONTENT]
```

Building Packets

```
HTTP/1.1 200 OK
Date: Fri, 26 Aug 2011 15:36:38 GMT
Server: Apache/2.2.15 (Fedora)
Last-Modified: Wed, 23 Feb 2011 23:32:02 GMT
ETag: "14401e-ee4-49cfb8102a080"
Accept-Ranges: bytes
Content-Length: 3812
Connection: close
Content-Type: text/html; charset=UTF-8

[3,812 BYTES OF PAGE CONTENT]
```

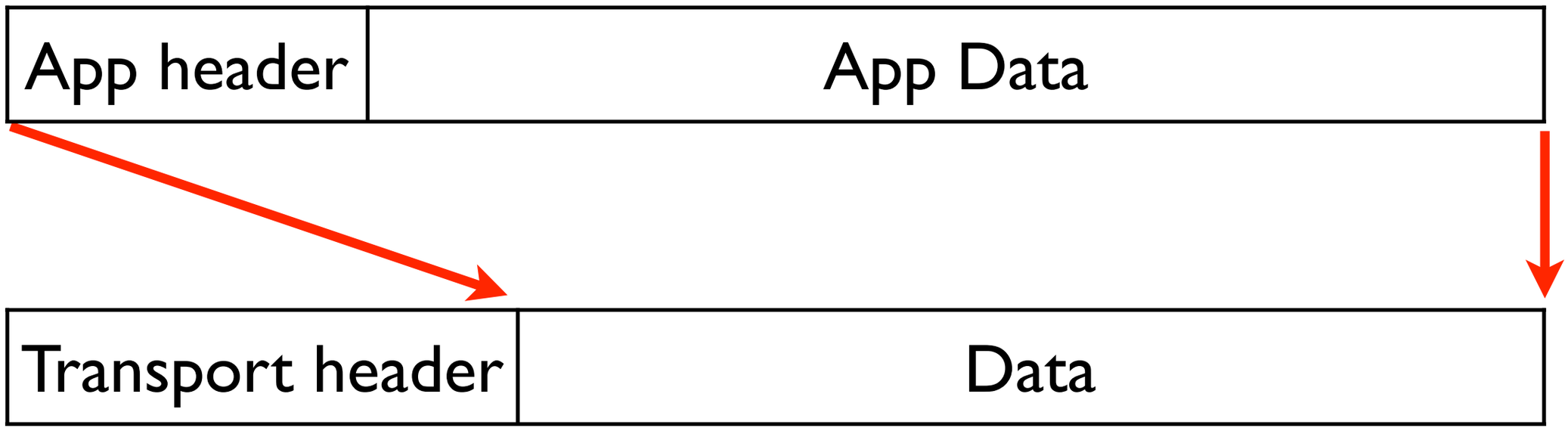
Page content is 3,812 bytes

A Packet

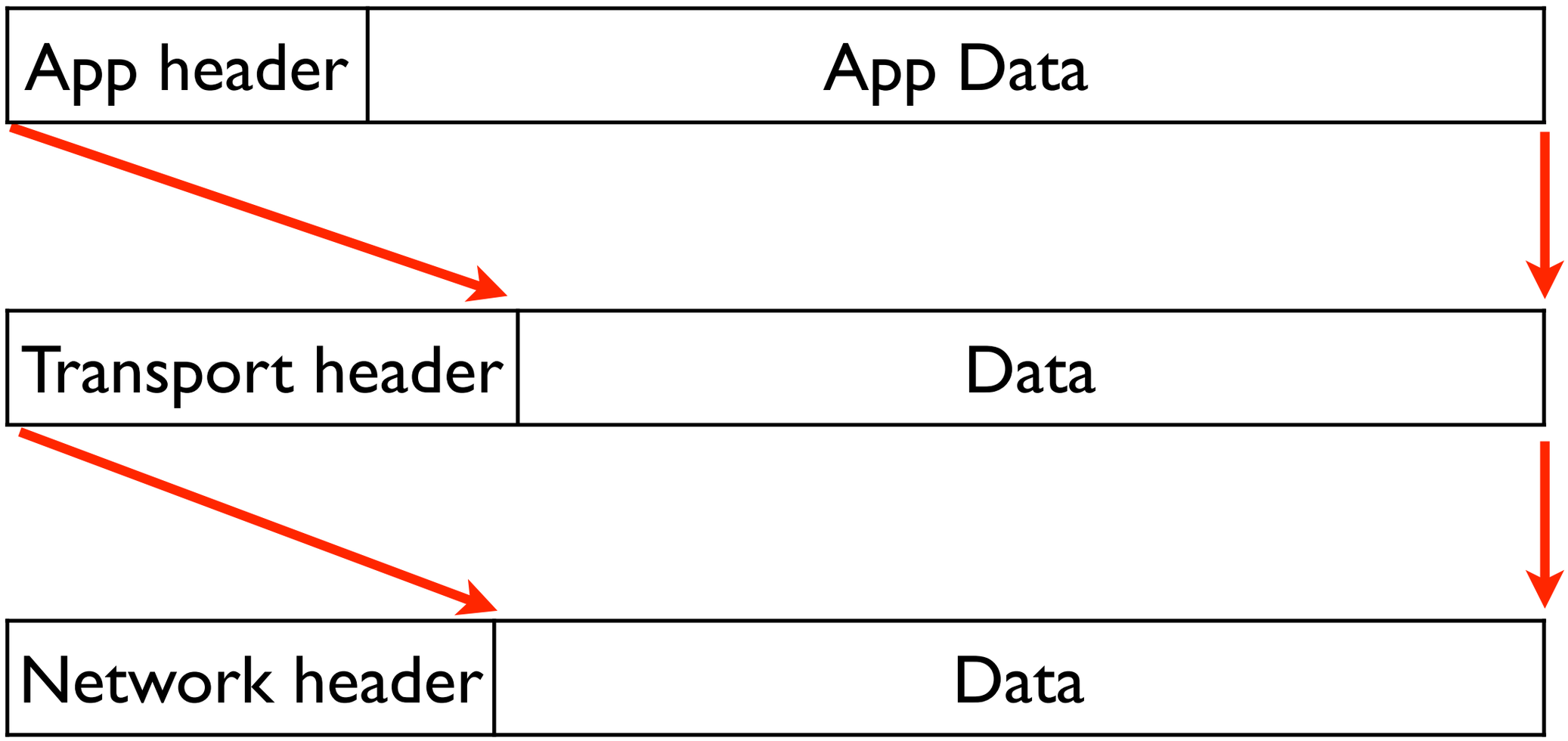
A Packet

A Packet

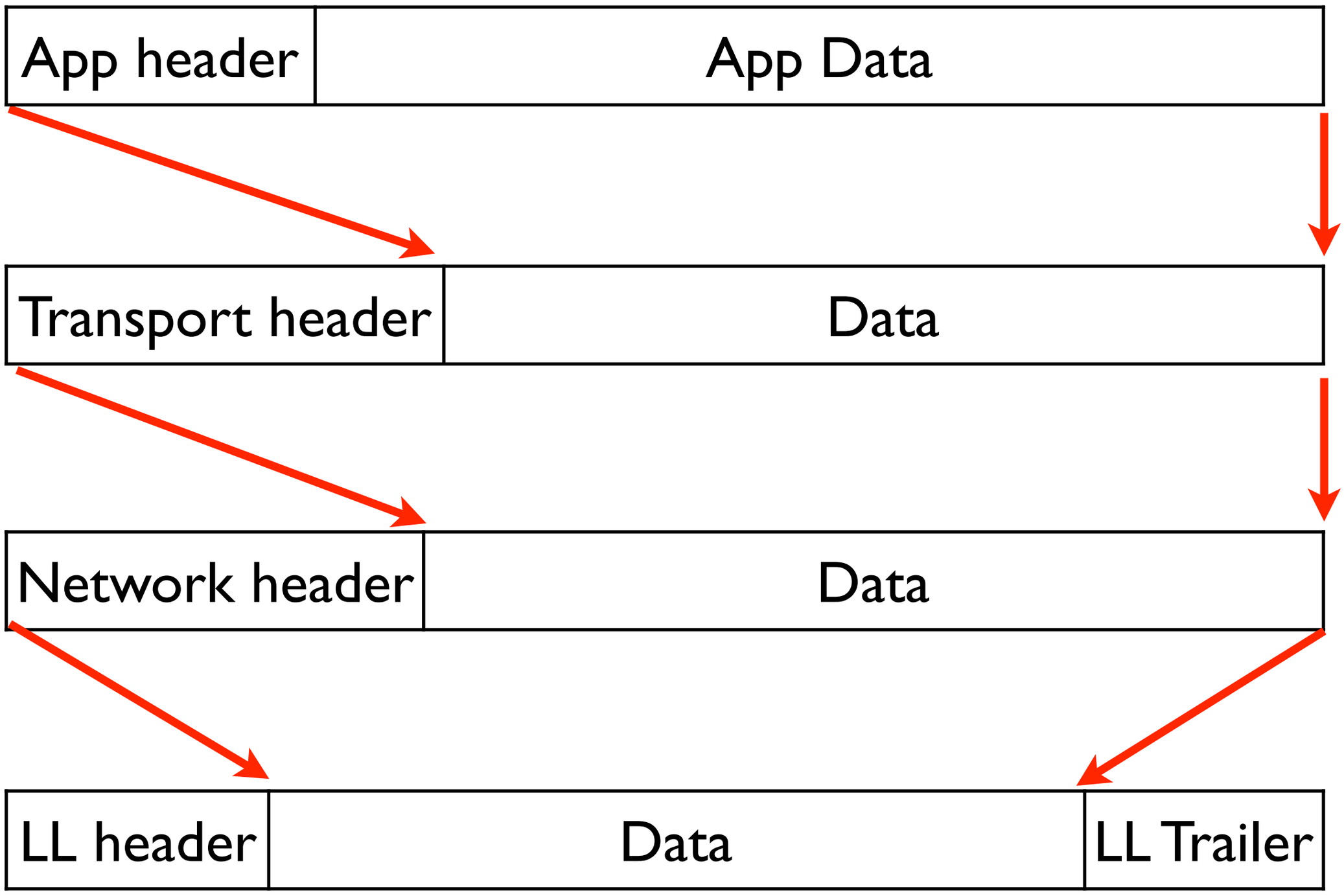
A Packet



A Packet

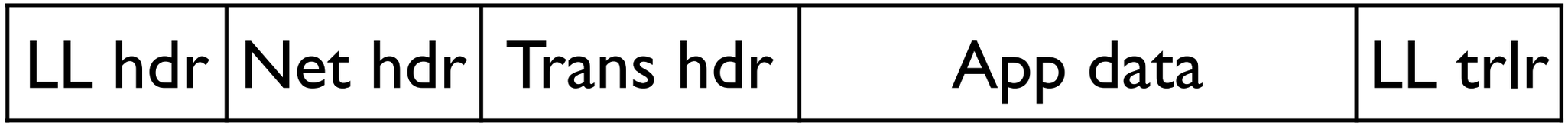

App header
App Data
Transport header
Data
Network header
Data

A Packet


App header
App Data
Transport header
Data
Network header
Data
LL header
Data
LL Trailer

A Packet (cont.)

A Packet (cont.)

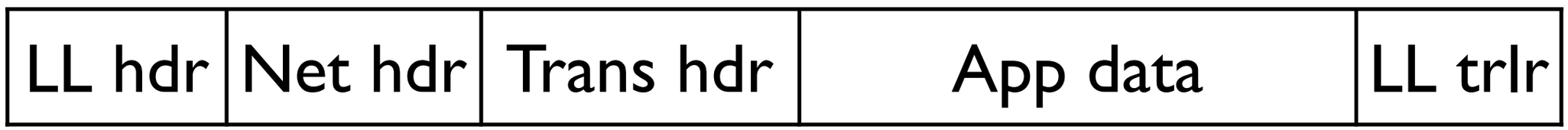


A Packet (cont.)

LL hdr	Net hdr	Trans hdr	App data	LL trlr

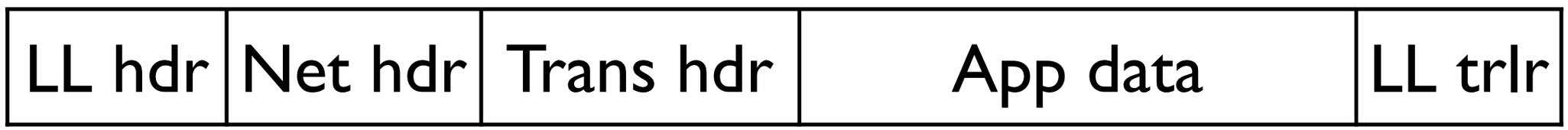
Eth	IP	TCP	HTTP hdr	HTTP data	Eth trlr

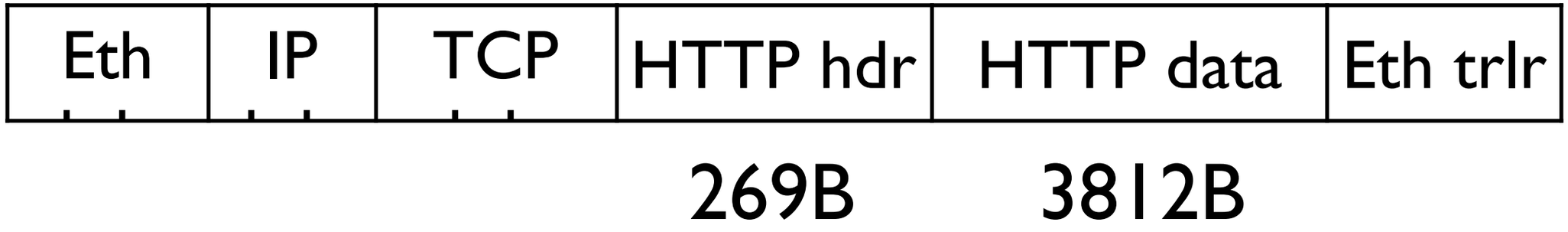
A Packet (cont.)



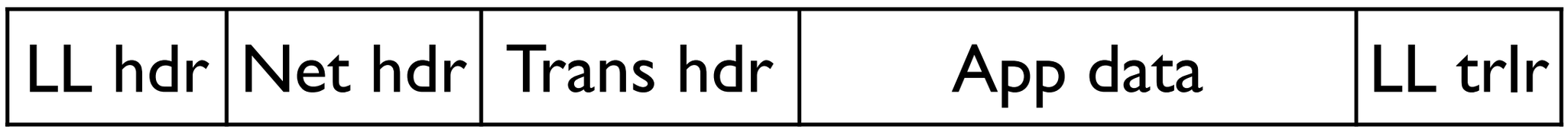
3812B

A Packet (cont.)

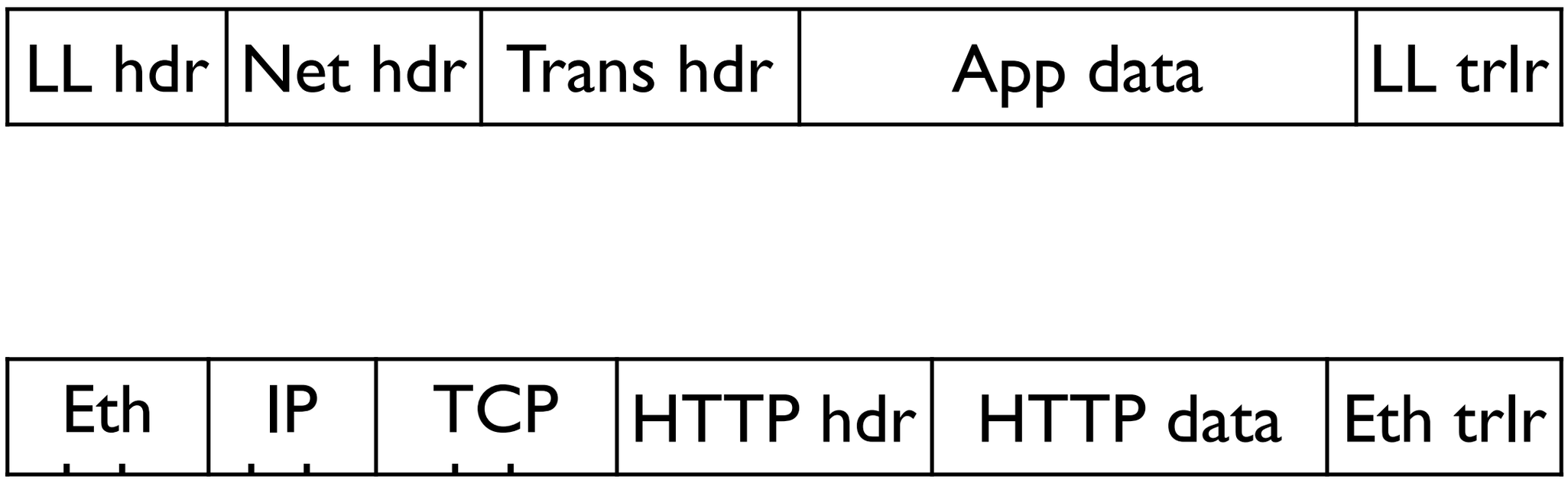




A Packet (cont.)



A Packet (cont.)



Total: 4,117 bytes
Non-content: 335 bytes (8%)
Content: 3,812 bytes (92%)

A Packet (cont.)

A Packet (cont.)

- Problem: max ethernet frame size (including everything):
  - 1518 bytes

A Packet (cont.)

- Problem: max ethernet frame size (including everything):
  - 1518 bytes
  - so, 4,081 bytes of HTTP response header (269B) and content (3,812B) will not fit in a single packet

A Packet (cont.)

A Packet (cont.)

Per-packet costs
(22 + 20 + 20 + 4 = 66 bytes)

A Packet (cont.)

Per-packet costs
(22 + 20 + 20 + 4 = 66 bytes)

Per transaction costs
(269 + 3812 = 4081 bytes)

A Packet (cont.)

A Packet (cont.)

- Packet 1:
  - Available for app:1518B - 66B == 1452B

A Packet (cont.)

- Packet 1:
  - Available for app:1518B - 66B == 1452B
  - Some needed by HTTP response header:

A Packet (cont.)

- Packet 1:
  - Available for app:1518B - 66B == 1452B
  - Some needed by HTTP response header:
    - send HTTP response header

A Packet (cont.)

- Packet 1:
  - Available for app:1518B - 66B == 1452B
  - Some needed by HTTP response header:
    - send HTTP response header
    - packet still has 1452B - 269B == 1183B of space left

A Packet (cont.)

- Packet 1:
  - Available for app:1518B - 66B == 1452B
  - Some needed by HTTP response header:
    - send HTTP response header
      - packet still has 1452B - 269B == 1183B of space left
  - 1183B < needed 3812B

A Packet (cont.)

- Packet 1:
  - Available for app:1518B - 66B == 1452B
  - Some needed by HTTP response header:
    - send HTTP response header
      - packet still has 1452B - 269B == 1183B of space left
  - 1183B < needed 3812B
  - send the first 1183B of content

A Packet (cont.)

- Packet 1:
  - Available for app:1518B - 66B == 1452B
  - Some needed by HTTP response header:
    - send HTTP response header
    - packet still has 1452B - 269B == 1183B of space left
  - 1183B < needed 3812B
  - send the first 1183B of content
  - 3812 - 1183 == 2629B left to send

A Packet (cont.)

A Packet (cont.)

- Packet 2:
  - has 1518 - 66 == 1452B of space

A Packet (cont.)

- Packet 2:
  - has 1518 - 66 == 1452B of space
  - send next 1452B of content

A Packet (cont.)

- Packet 2:
  - has 1518 - 66 == 1452B of space
  - send next 1452B of content
  - 2629 - 1452 == 1177B of content still unsent

A Packet (cont.)

- Packet 2:
  - has 1518 - 66 == 1452B of space
  - send next 1452B of content
  - 2629 - 1452 == 1177B of content still unsent
- Packet 3:

A Packet (cont.)

- Packet 2:
  - has 1518 - 66 == 1452B of space
  - send next 1452B of content
  - 2629 - 1452 == 1177B of content still unsent
- Packet 3:
  - has 1518 - 66 == 1452B of space

A Packet (cont.)

- Packet 2:
  - has 1518 - 66 == 1452B of space
  - send next 1452B of content
  - 2629 - 1452 == 1177B of content still unsent
- Packet 3:
  - has 1518 - 66 == 1452B of space
  - 1452B > 1177B remaining to send

A Packet (cont.)

- Packet 2:
  - has 1518 - 66 == 1452B of space
  - send next 1452B of content
  - 2629 - 1452 == 1177B of content still unsent
- Packet 3:
  - has 1518 - 66 == 1452B of space
  - 1452B > 1177B remaining to send
  - so, send all remaining bytes

A Packet (cont.)

	Overhead Bytes	Content Bytes	Overhead Percent
Packet 1	335	1183	22%
Packet 2	66	1452	4%
Packet 3	66	1177	5%
Total	**467**	**3812**	**11%**

A Packet (cont.)

	Overhead Bytes	Content Bytes	Overhead Percent
Packet 1	335	1183	22%
Packet 2	66	1452	4%
Packet 3	66	1177	5%
Total	**467**	**3812**	**11%**

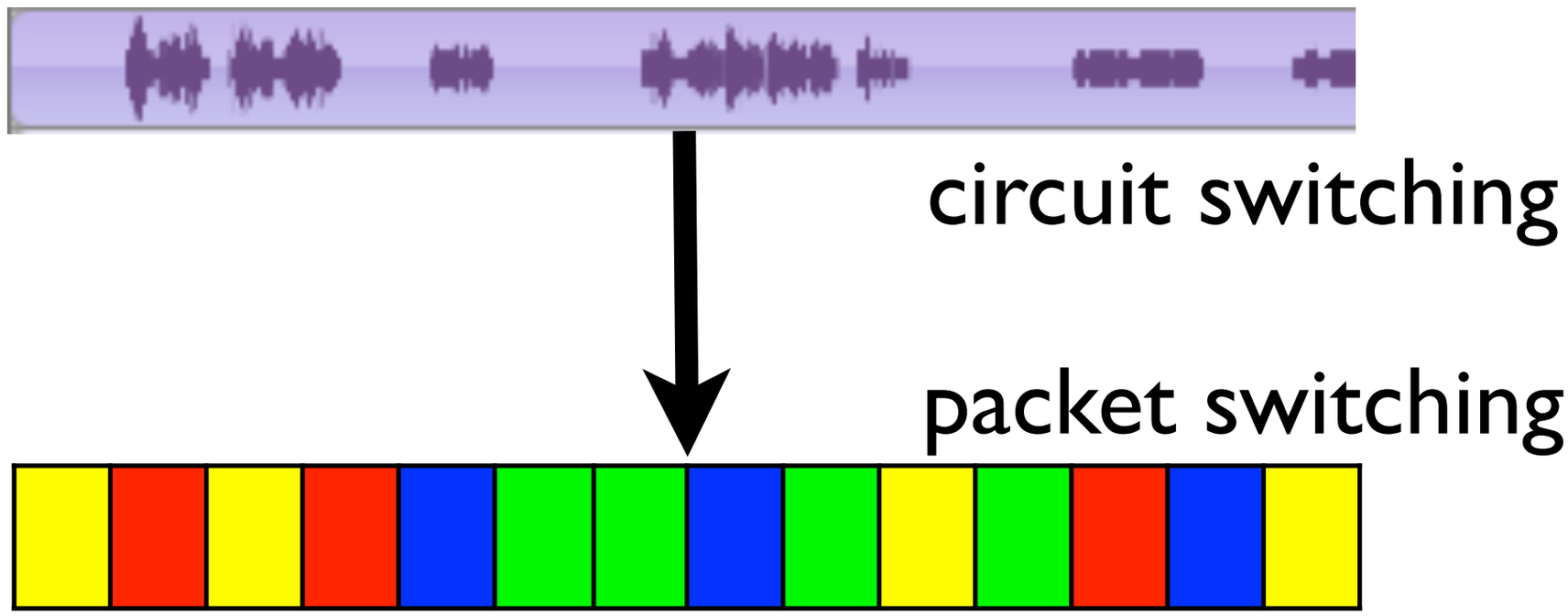
The cost of packets

A Packet (cont.)

A Packet (cont.)

circuit switching

A Packet (cont.)



A Packet (cont.)