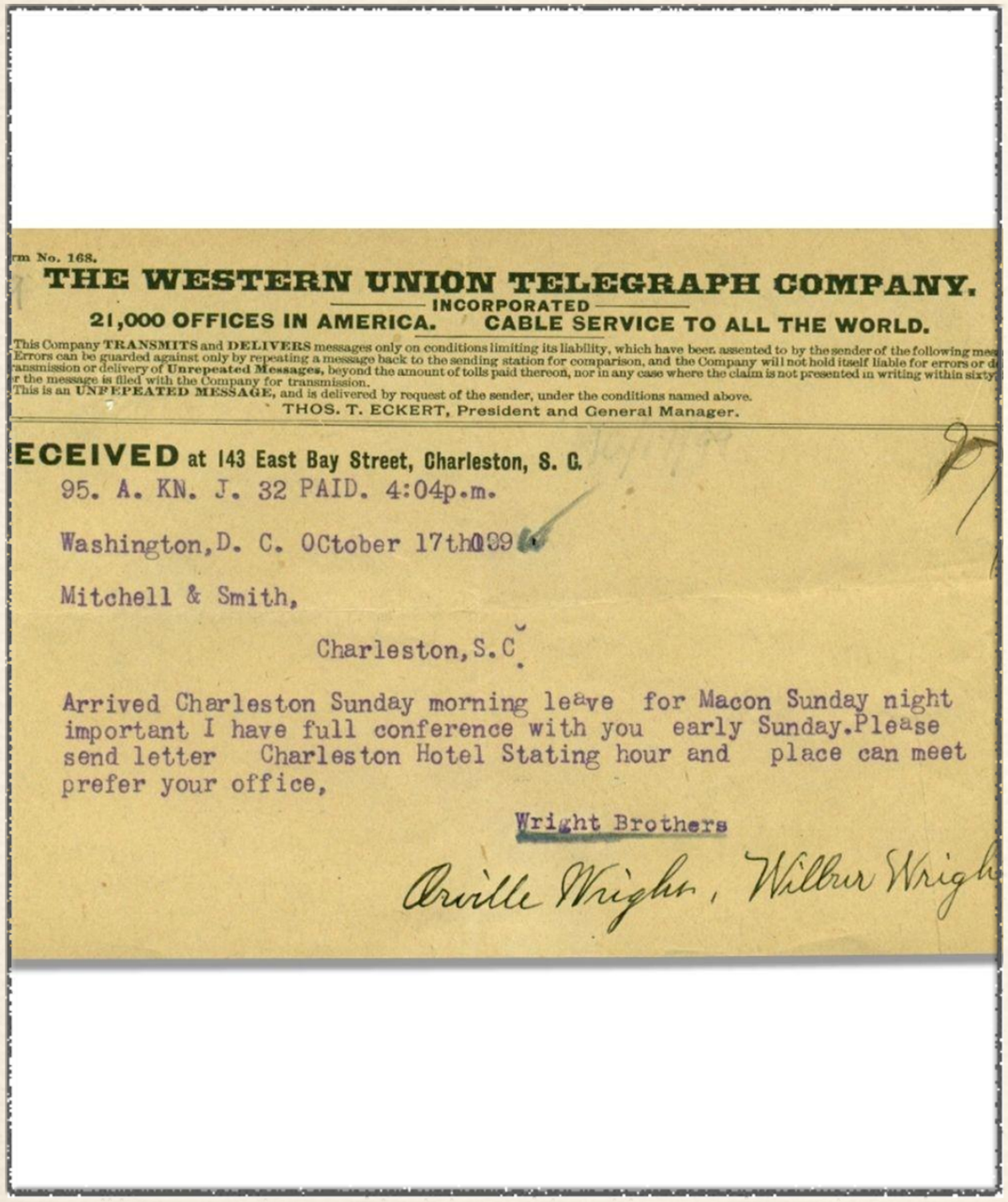
Internet Datagrams

The Millennial Telegraph

The Telegram



rm No. 168.

THE WESTERN UNION TELEGRAPH COMPANY.

INCORPORATED

21,000 OFFICES IN AMERICA. CABLE SERVICE TO ALL THE WORLD.

This Company **TRANSMITS** and **DELIVERS** messages only on conditions limiting its liability, which have been assented to by the sender of the following mes
Errors can be guarded against only by repeating a message back to the sending station for comparison, and the Company will not hold itself liable for errors or de
ansmission or delivery of **Unrepeated Messages,** beyond the amount of tolls paid thereon, nor in any case where the claim is not presented in writing within sixty
r the message is filed with the Company for transmission.
This is an **UNREPEATED MESSAGE,** and is delivered by request of the sender, under the conditions named above.

THOS. T. ECKERT, President and General Manager.

ECEIVED at 143 East Bay Street, Charleston, S. C.

95. A. KN. J. 32 PAID. 4:04p.m.

Washington, D. C. OCtober 17th 99

Mitchell & Smith,

Charleston, S.C.

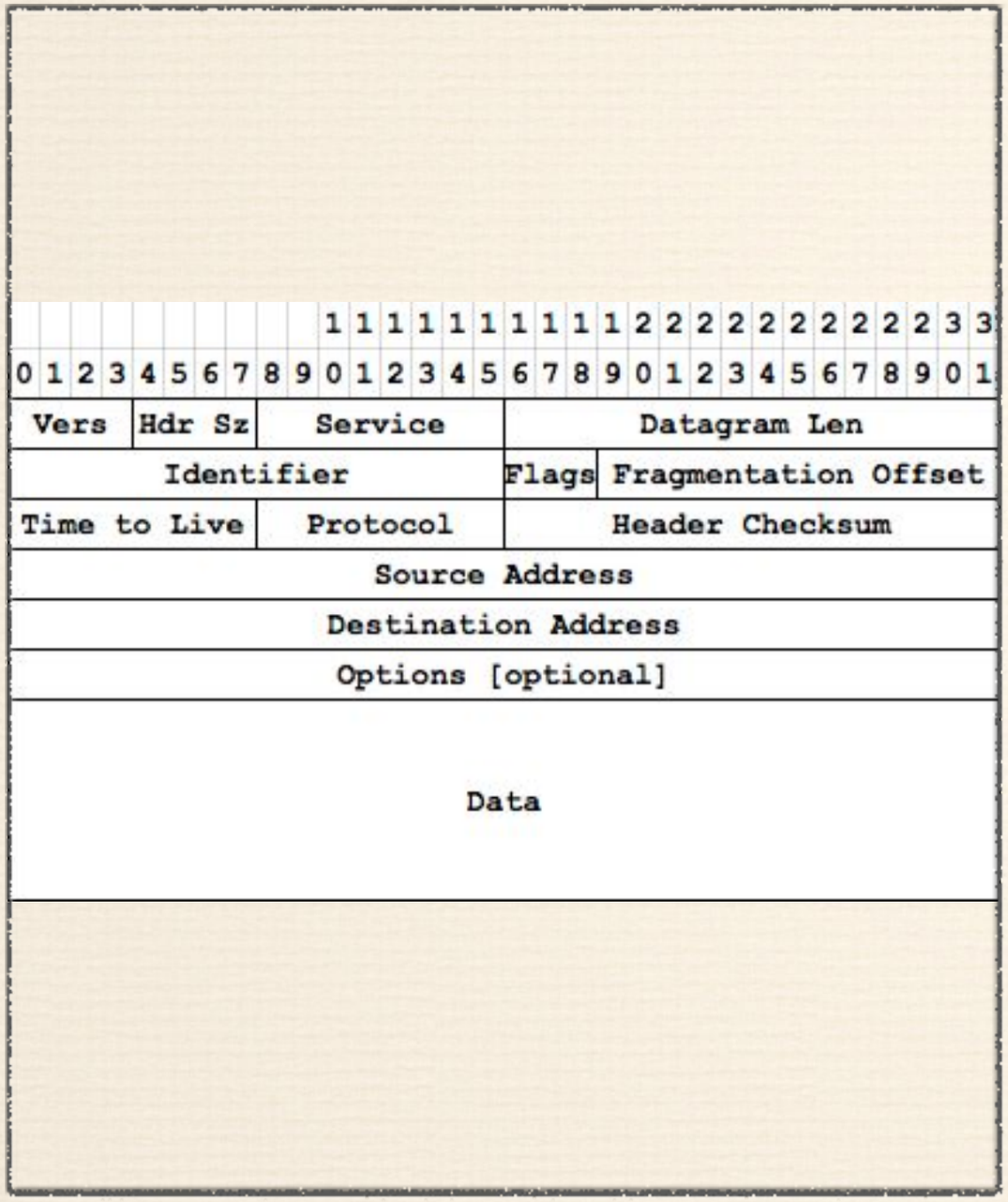
Arrived Charleston Sunday morning leave for Macon Sunday night important I have full conference with you early Sunday. Please send letter Charleston Hotel Stating hour and place can meet prefer your office,

Wright Brothers

Orville Wright. Wilbur Wright

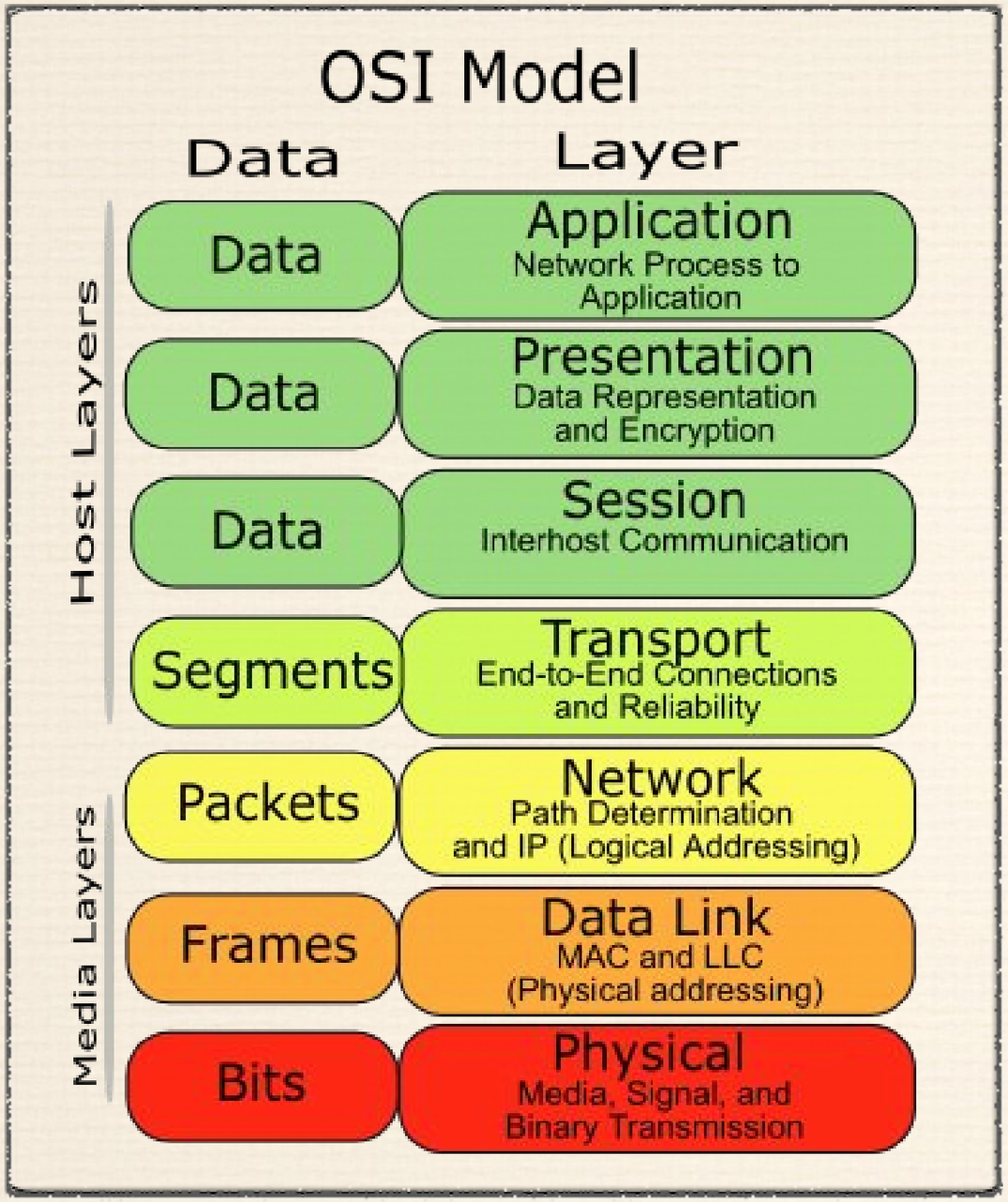
- Self-contained
- Conveys data
- Independent
- Ignorant of its medium

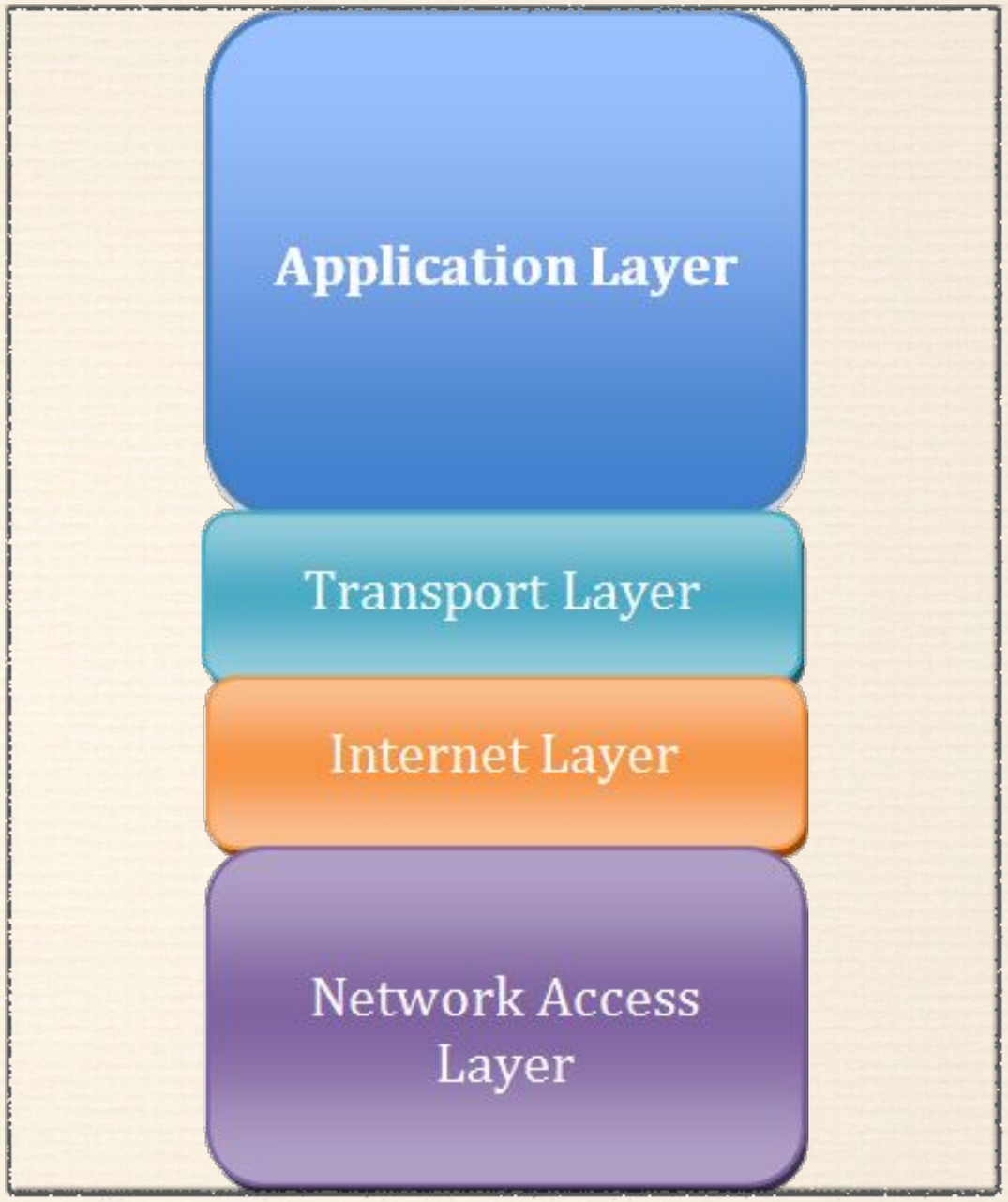
The Datagram



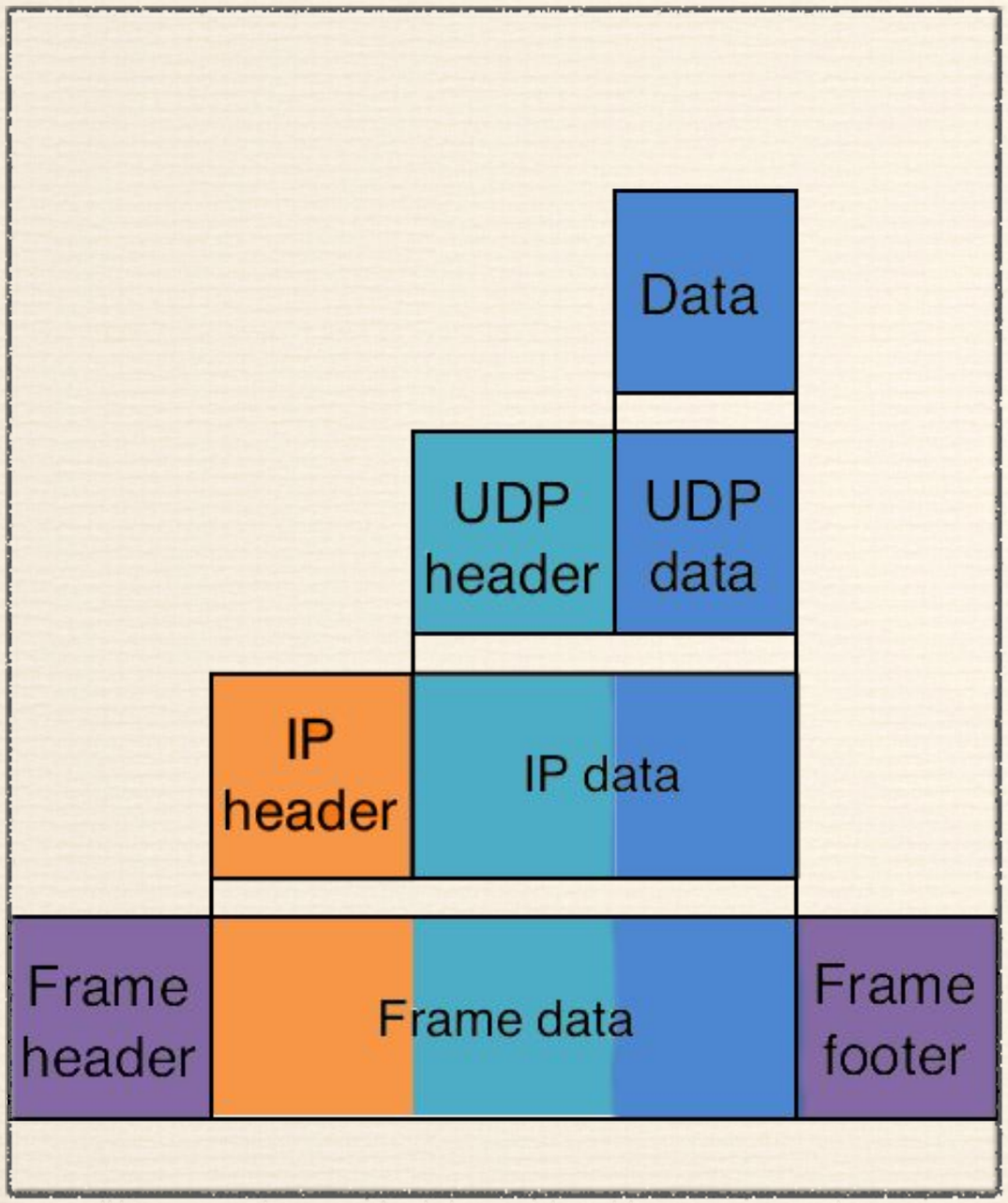
"A self-contained, independent entity of data carrying sufficient information to be routed from the source to the destination computer without reliance on earlier exchanges between this source and destination computer and the transporting network."

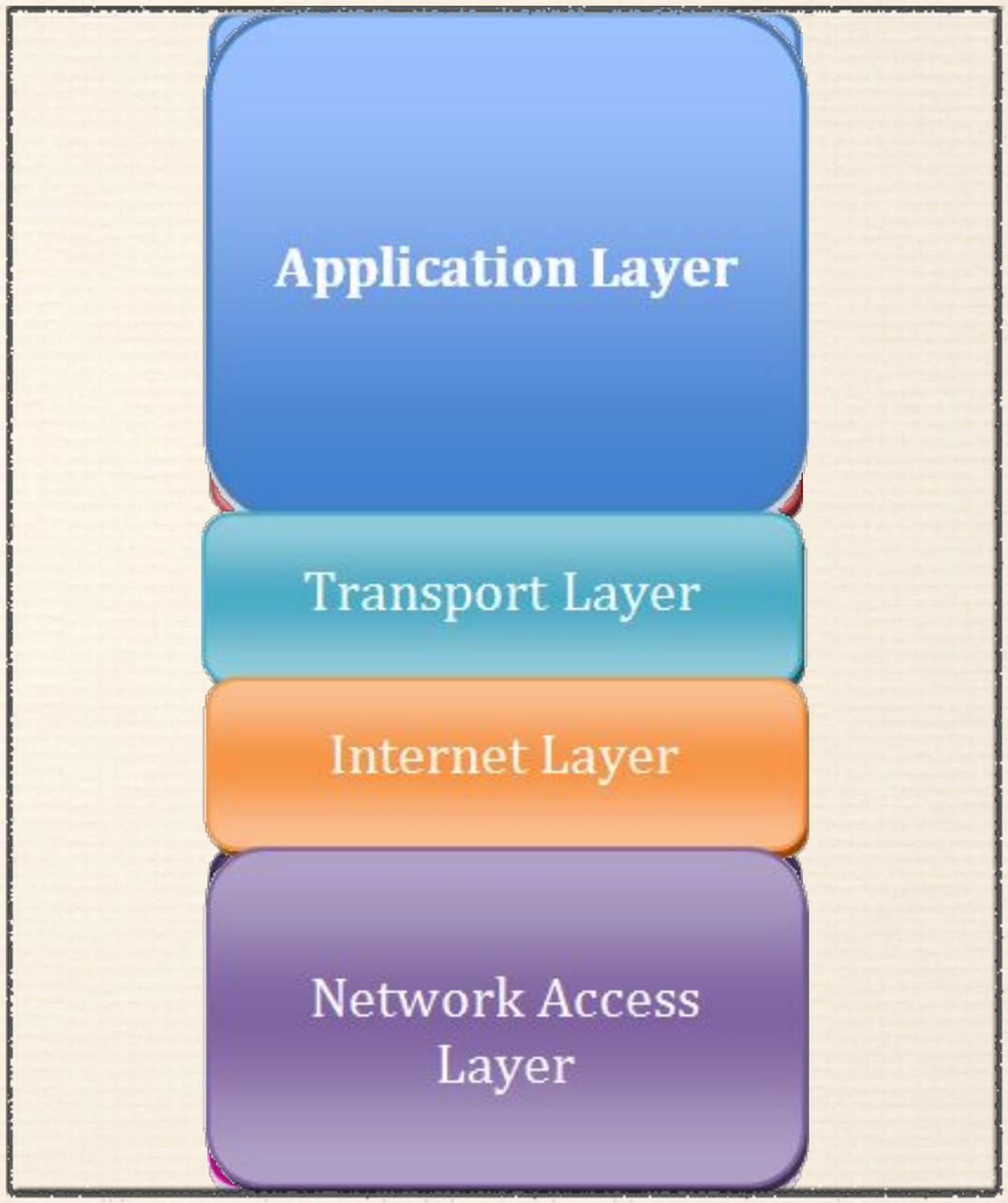
—RFC1594


OSI Model
Data
Layer
Host Layers
Media Layers
Data
Application
Network Process to Application
Data
Presentation
Data Representation and Encryption
Data
Session
Interhost Communication
Segments
Transport
End-to-End Connections and Reliability
Packets
Network
Path Determination and IP (Logical Addressing)
Frames
Data Link
MAC and LLC (Physical addressing)
Bits
Physical
Media, Signal, and Binary Transmission


Application Layer
Transport Layer
Internet Layer
Network Access Layer

The TCP/IP Model

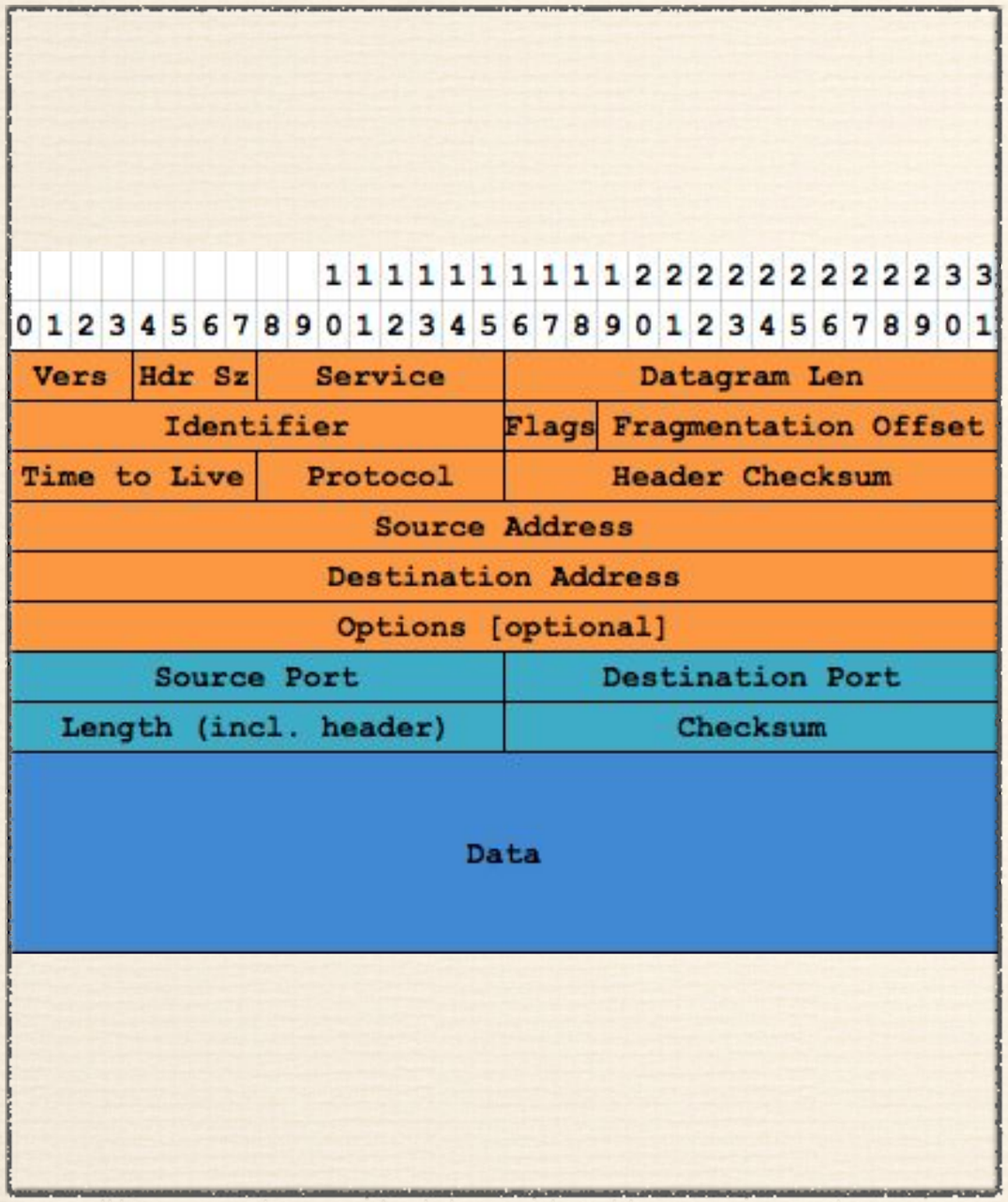




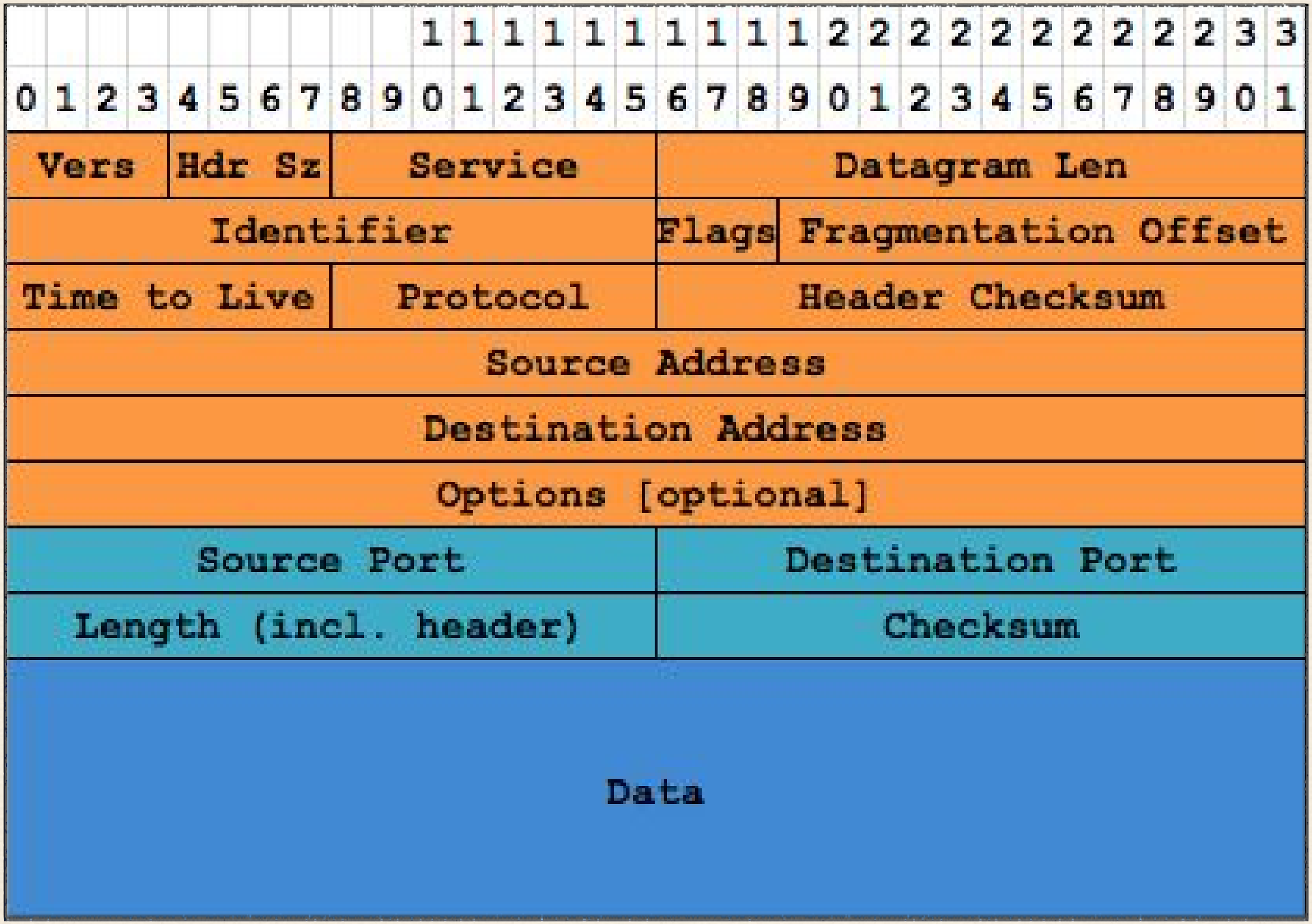
Handling Abstraction

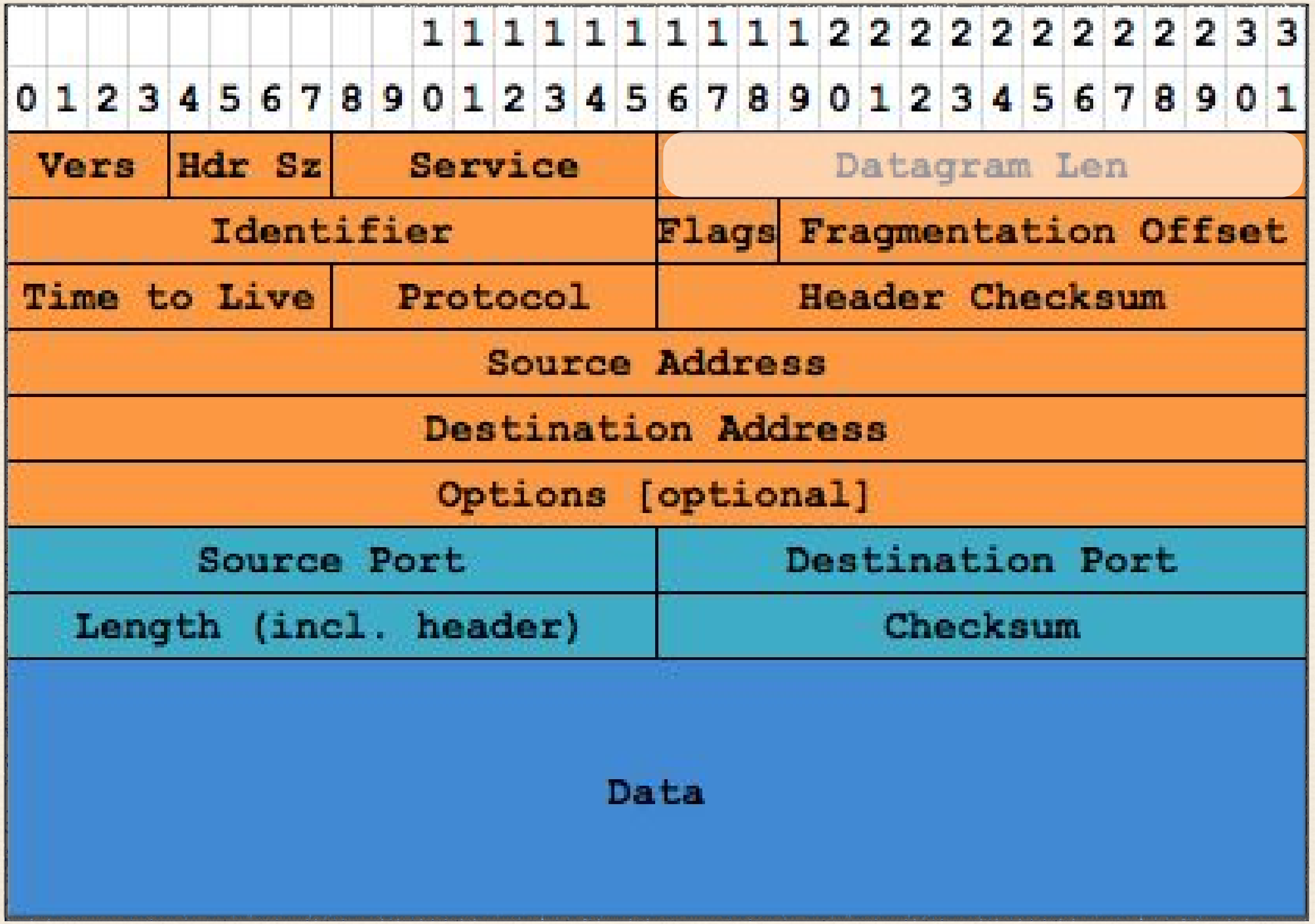
- Operate at your level
- Do what's necessary for one level down
- Be aware of the other levels

UDP



- Thinnest possible layer atop Internet Protocol (IP)
- “Fire and forget” services like DNS, RIP, DHCP
- Streaming audio and video
- Games


0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Vers
Hdr Sz
Service
Datagram Len
Identifier
Flags
Fragmentation Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options [optional]
Source Port
Destination Port
Length (incl. header)
Checksum
Data

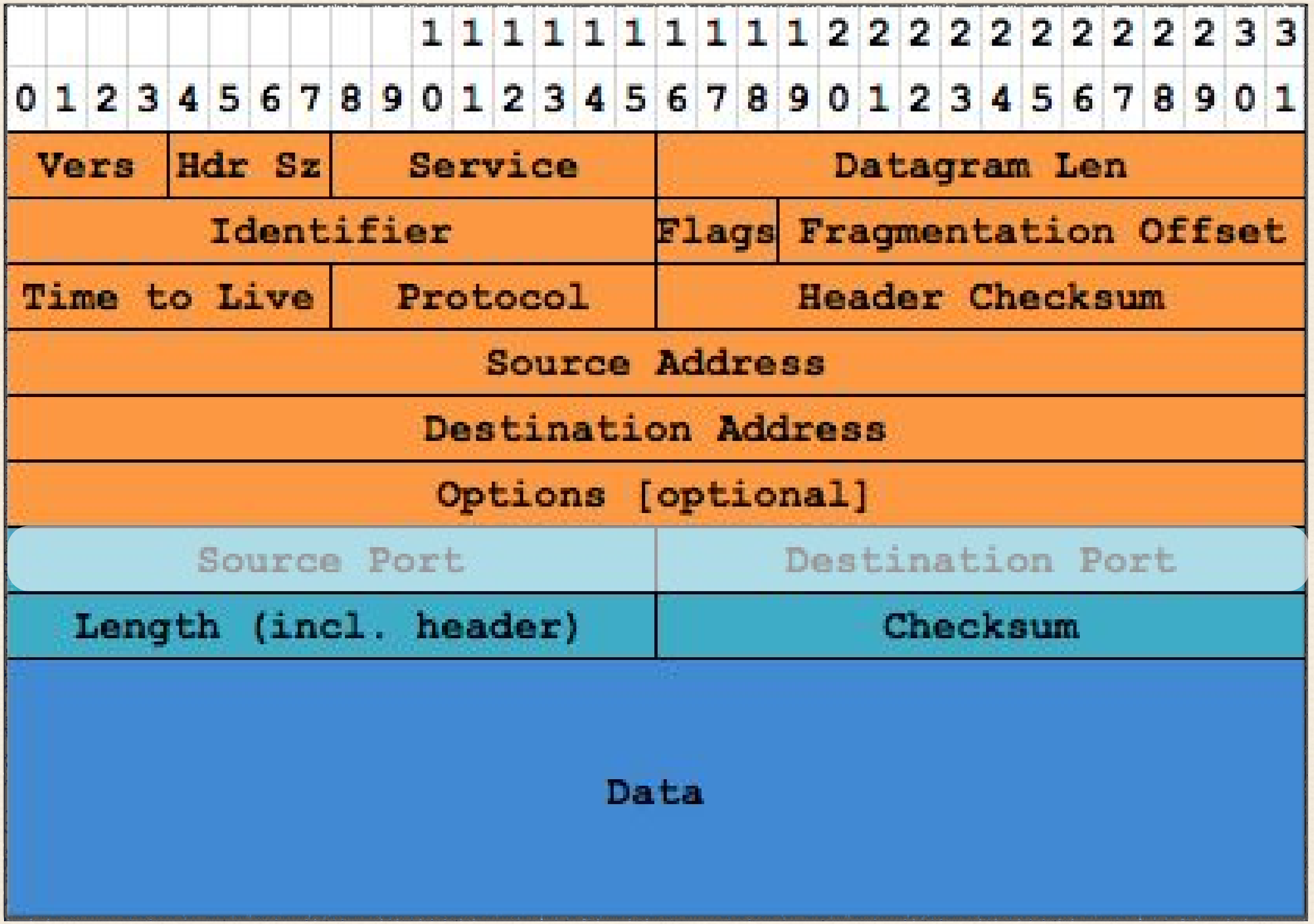

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Vers
Hdr Sz
Service
Datagram Len
Identifier
Flags
Fragmentation Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options [optional]
Source Port
Destination Port
Length (incl. header)
Checksum
Data

										1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	3	3
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Vers				Hdr Sz						Service										Datagram Len										
Identifier																				Flags		Fragmentation Offset								
Time to Live										Protocol										Header Checksum										
Source Address																														
Destination Address																														
Options [optional]																														
Source Port																				Destination Port										
Length (incl. header)																				Checksum										
Data																														

										1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	3	3
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Vers				Hdr Sz				Service												Datagram Len										
Identifier																				Flags			Fragmentation Offset							
Time to Live										Protocol										Header Checksum										
Source Address																														
Destination Address																														
Options [optional]																														
Source Port																				Destination Port										
Length (incl. header)																				Checksum										
Data																														

0 1 2 3 4 5 6 7 8 9	1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
Vers / Hdr Sz / Service	Datagram Len
Identifier	Flags / Fragmentation Offset
Time to Live / Protocol	Header Checksum
Source Address	
Destination Address	
Options [optional]	
Source Port	Destination Port
Length (incl. header)	Checksum
Data	

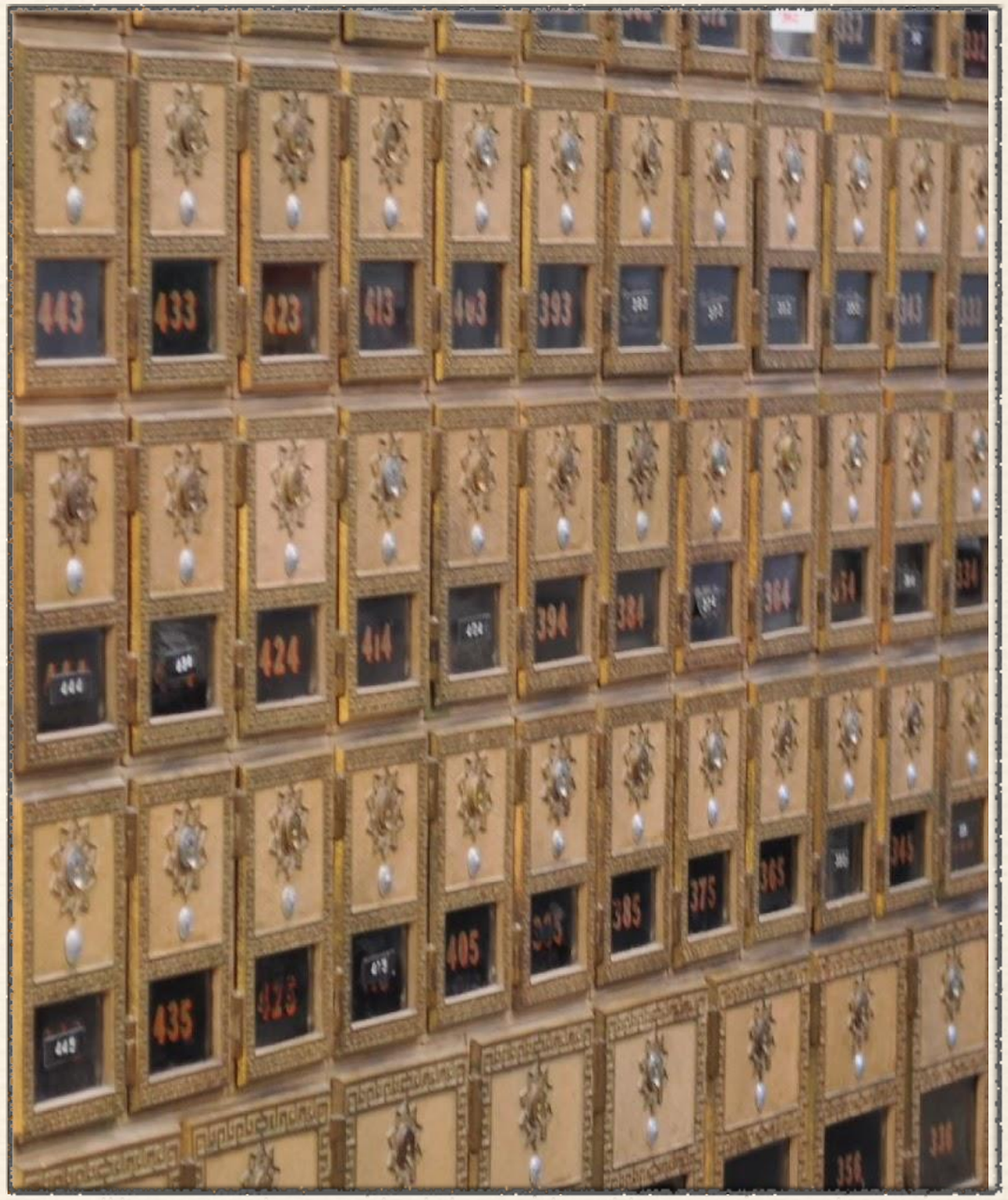
0 1 2 3 4 5 6 7 8 9	1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 3 3 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
Vers / Hdr Sz / Service	Datagram Len
Identifier	Flags / Fragmentation Offset
Time to Live / Protocol	Header Checksum
Source Address	
Destination Address	
Options [optional]	
Source Port	Destination Port
Length (incl. header)	Checksum
Data	

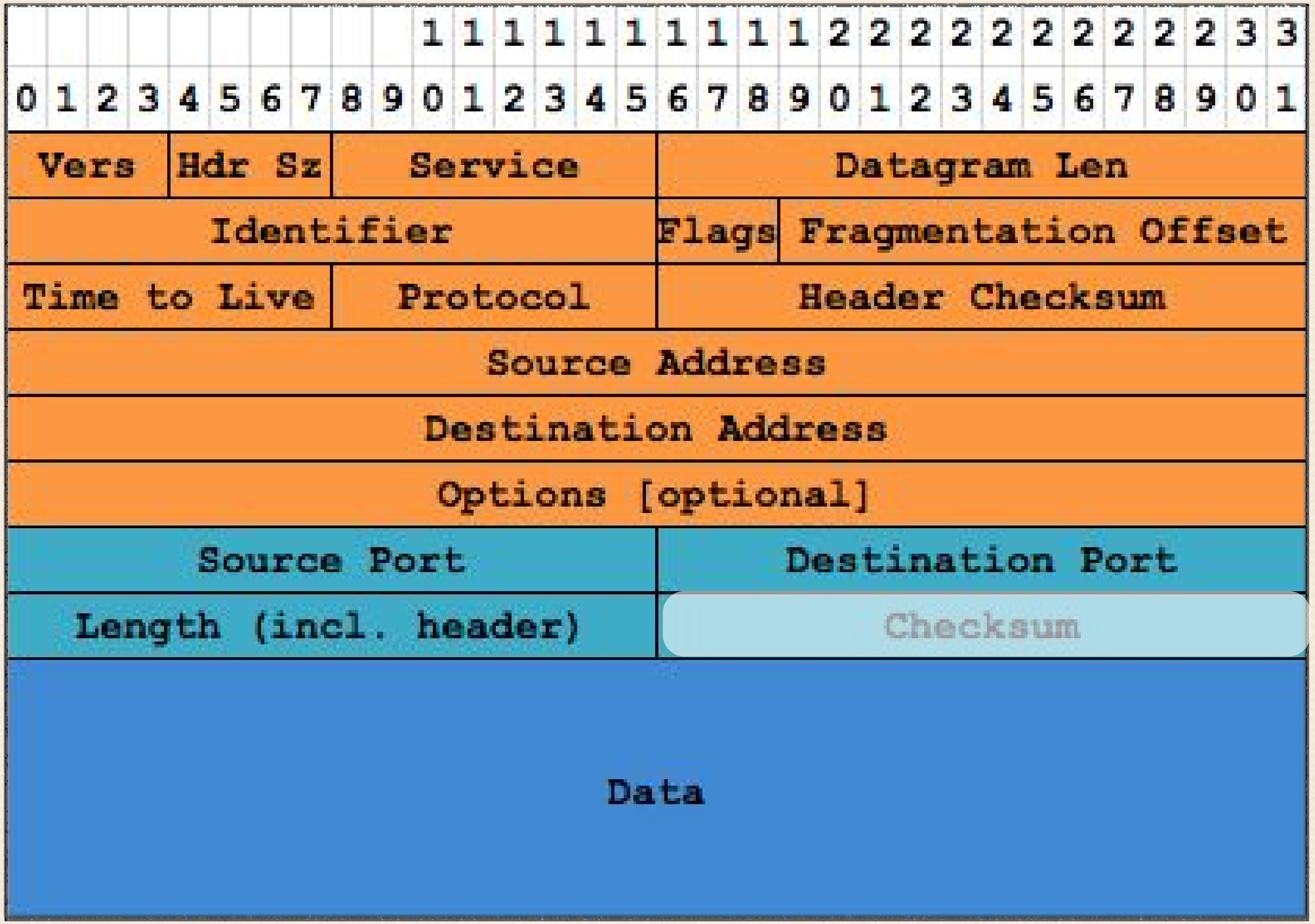

0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Vers
Hdr Sz
Service
Datagram Len
Identifier
Flags
Fragmentation Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options [optional]
Source Port
Destination Port
Length (incl. header)
Checksum
Data

Sockets and Ports

- A *socket* is a mailbox for datagrams

- A *port* is the ID number of a socket




0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Vers
Hdr Sz
Service
Datagram Len
Identifier
Flags
Fragmentation Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options [optional]
Source Port
Destination Port
Length (incl. header)
Checksum
Data

										1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	3	3
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1
Vers				Hdr Sz				Service												Datagram Len										
Identifier																				Flags		Fragmentation Offset								
Time to Live										Protocol										Header Checksum										
Source Address																														
Destination Address																														
Options [optional]																														
Source Port																				Destination Port										
Length (incl. header)																				Checksum										
Data																														

548

Maximum Packet Size?

Maximum Packet Size

1400

548

Safest Packet Size

UDP Packets

- Completely self-contained
- Minimal overhead
- Unreliable
- Sockets & Ports
- 1400 bytes *should* be safe, 548 bytes *will* be safe