

TCP Connections

The High-Level View

UDP Problems

- · Packets have to be resent
- · Packets arrive duplicated or out of order
- · Have to split data across packets
- Can overwhelm connection



...there is a better way

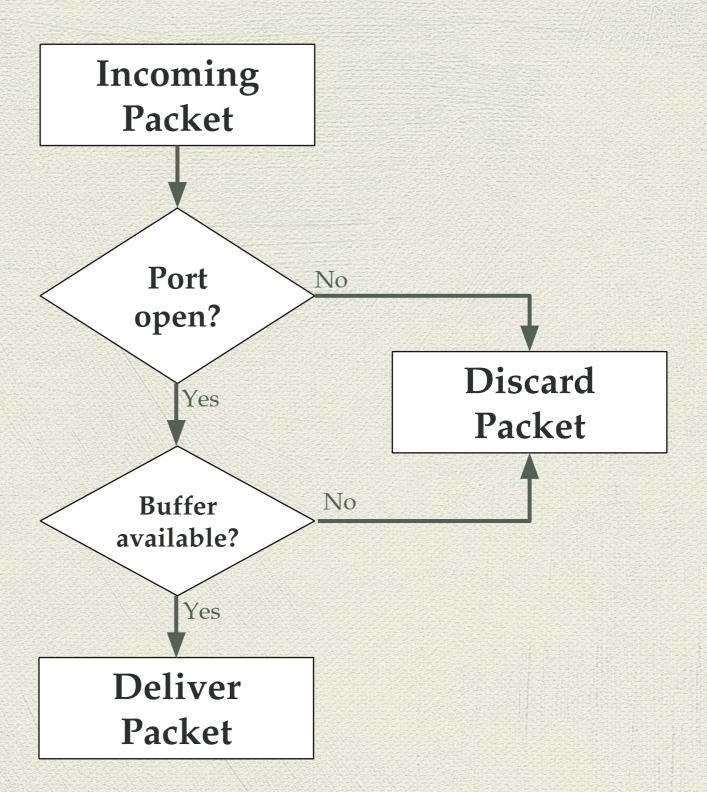
TCP Solutions

- · Packets are promised to arrive
- · Duplicate/out-of-order packets dropped
- · Data stream split into packets automatically
- · Packets are throttled to not overload network

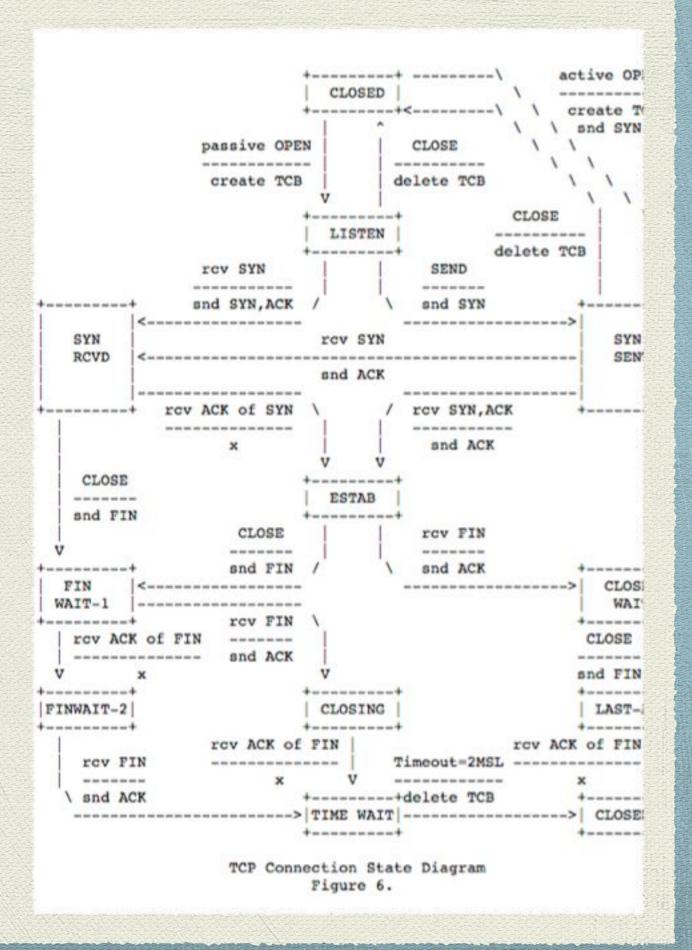
Why Not?

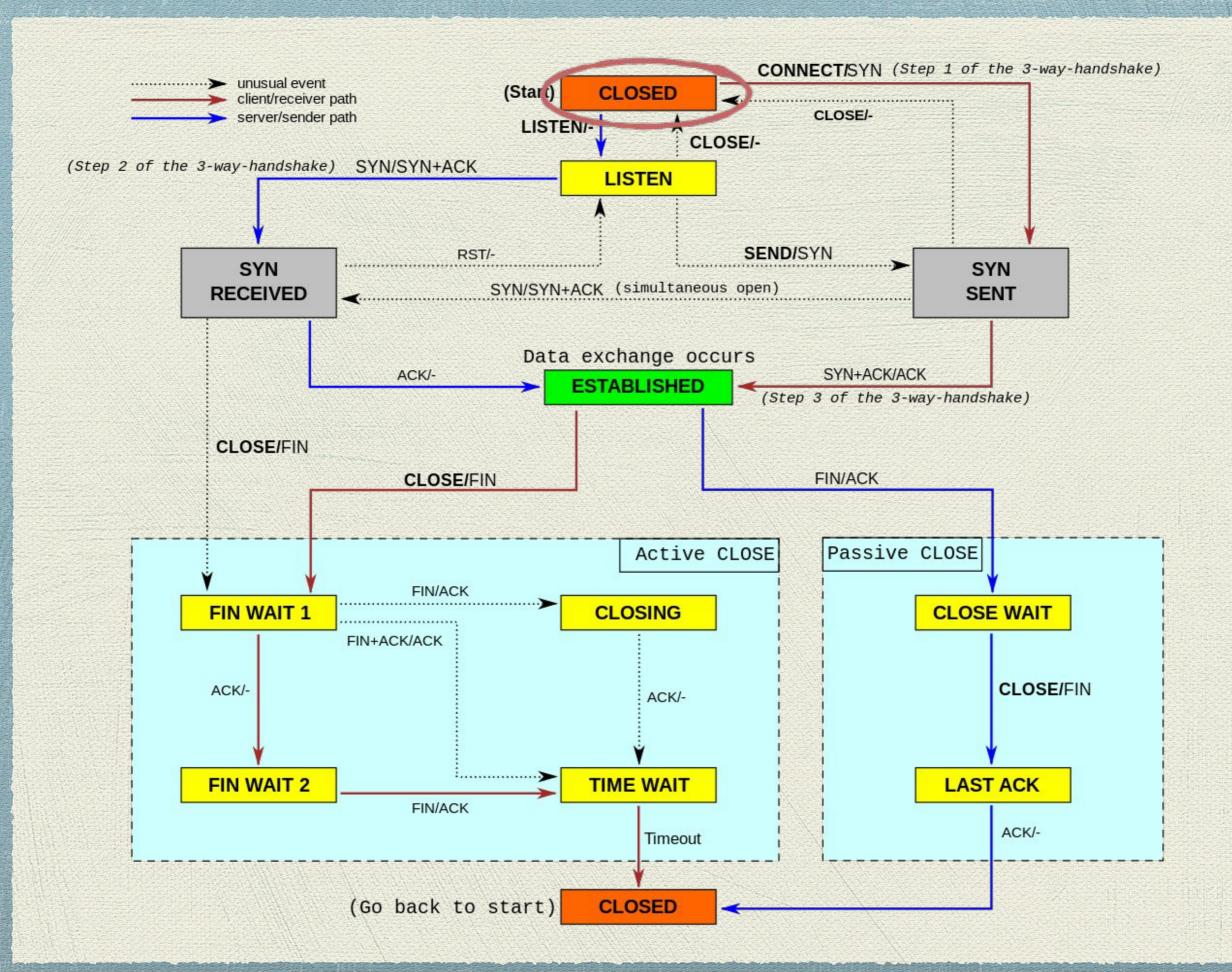
- · Not all packets equally valuable
- · Duplicate packets are a warning sign
- · Your data may be pre-segmented
- Games can afford to be bad citizens

UDP State Machine



TCP State Machine





Server Send message & start timer

Client

Server
Send message & start timer

Client

Send message & start timer

ACK received! Send message & start timer

Client



Send message & start timer

ACK received! Send message & start timer

Timer expired

Client



Send message & start timer

ACK received! Send message & start timer

Timer expired Resend message & restart timer

Client



Send message & start timer

ACK received! Send message & start timer

Timer expired
Resend message & restart timer

Client

Receive message Send ACK



Receive re-sent message Send ACK

Send message & start timer

ACK received! Send message & start timer

Timer expired
Resend message & restart timer

ACK received! Send message & start timer

Client

Receive message Send ACK



Receive re-sent message Send ACK

Send message & start timer

ACK received! Send message & start timer

Timer expired
Resend message & restart timer

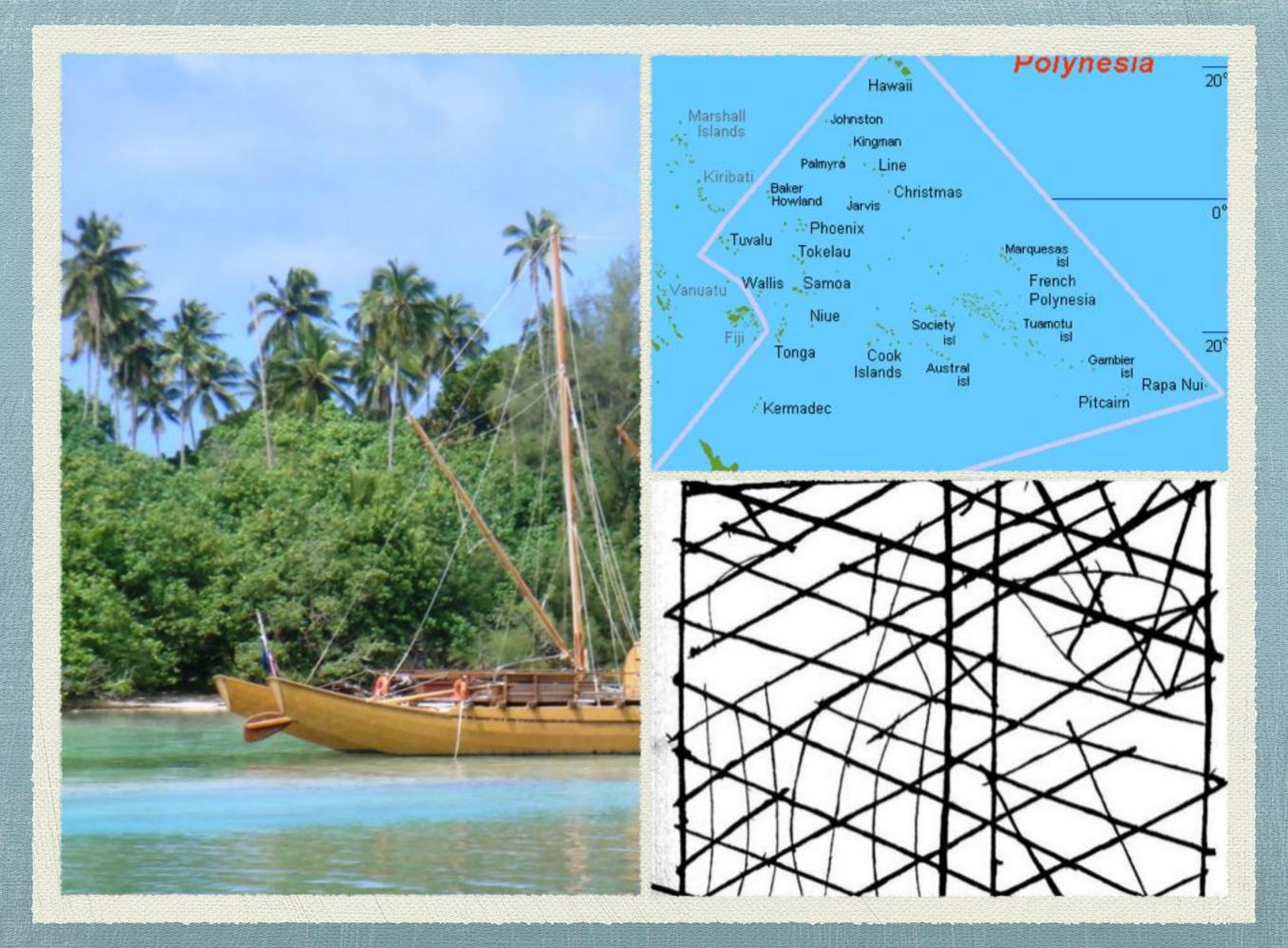
ACK received! Send message & start timer

Client

Receive message Send ACK



Receive re-sent message Send ACK



Little's Law (John Little, MIT, 1954)

- Occupancy = Latency x Throughput
- count = count/time x time
- $O = L \times T \longleftrightarrow L = O / T \longleftrightarrow T = O / L$

Server Send message 1

Client

Server Send message 1 Send message 2

Client

Send message 1 Send message 2

Client

Receive & ACK 1

Send message 1 Send message 2



Receive & ACK 1 Receive & ACK 2

Send message 1 Send message 2

1 ACKed; send 3

Client

Receive & ACK 1 Receive & ACK 2

Send message 1 Send message 2

1 ACKed; send 3

2 TIME; resend 2

Client

Receive & ACK 1 Receive & ACK 2

Send message 1 Send message 2

1 ACKed; send 3

2 TIME; resend 2

Client

Receive & ACK 1 Receive & ACK 2

Receive & ACK 3

Send message 1 Send message 2

1 ACKed; send 3

2 TIME; resend 2

Client

Receive & ACK 1 Receive & ACK 2

Receive & ACK 3

ACK 3 (&2); LIMIT=1

<u>Server</u>

Send message 1 Send message 2

1 ACKed; send 3

2 TIME; resend 2

3 & 2 ACKed; send 4 Send 5

Client

Receive & ACK 1 Receive & ACK 2

Receive & ACK 3

ACK 3 (&2); LIMIT=1

Send message 1 Send message 2

1 ACKed; send 3

2 TIME; resend 2

3 & 2 ACKed; send 4 Send 5 Set new LIMIT=1

Client

Receive & ACK 1 Receive & ACK 2

Receive & ACK 3

ACK 3 (&2); LIMIT=1

Send message 1 Send message 2

1 ACKed; send 3

2 TIME; resend 2

3 & 2 ACKed; send 4 Send 5 Set new LIMIT=1

Client

Receive & ACK 1 Receive & ACK 2

Receive & ACK 3

ACK 3 (&2); LIMIT=1

Receive & ACK 4

Send message 1 Send message 2

1 ACKed; send 3

2 TIME; resend 2

3 & 2 ACKed; send 4 Send 5 Set new LIMIT=1

4 ACKed; limit reached; wait

Client

Receive & ACK 1 Receive & ACK 2

Receive & ACK 3

ACK 3 (&2); LIMIT=1

Receive & ACK 4

· Packets sent and ACKed

· Packets not yet sent and invalid to send

1111111111222222222233 01234567890123456789012345678901 Hdr Sz Service Datagram Len Vers Flags Fragmentation Offset Identifier Protocol Header Checksum Time to Live Source Address Destination Address Options [optional] Destination Port Source Port Sequence Number ACK Number NCEUAPRSF D.Off. Window Checksum Urgent Pointer Options [optional]

Data

Sequence Numbers

Application Layer

Transport Layer

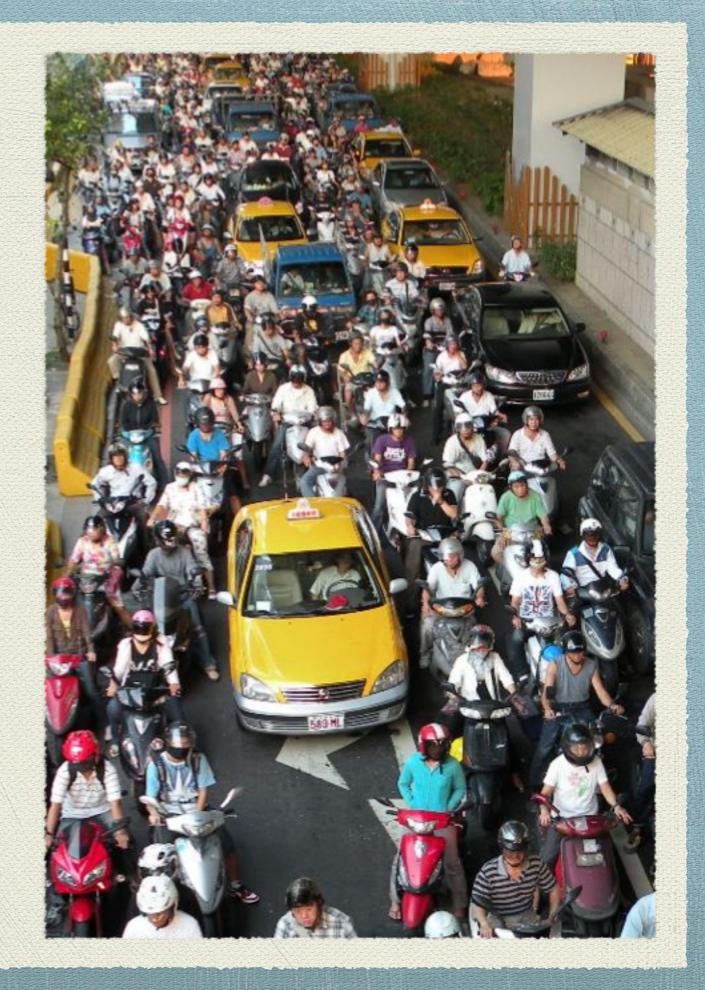
Internet Layer

Network Access Layer

	272																														
										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 Fla															Flags Fragmentation Offset																
T	Time to Live Protocol											Header Checksum																			
Source Address																															
Destination Address															100																
Options [optional]																															
	Source Port														Destination Port																
Sequence Number																															
ACK Number																															
D	D.Off. O NCEUAPRSF													Window																	
	Checksum														Urgent Pointer																
The second second	Options [optional]																														
																															I
														1	Ja	ta															
-																															

Sliding Windows

Congestion Control



This page intentionally left blank.

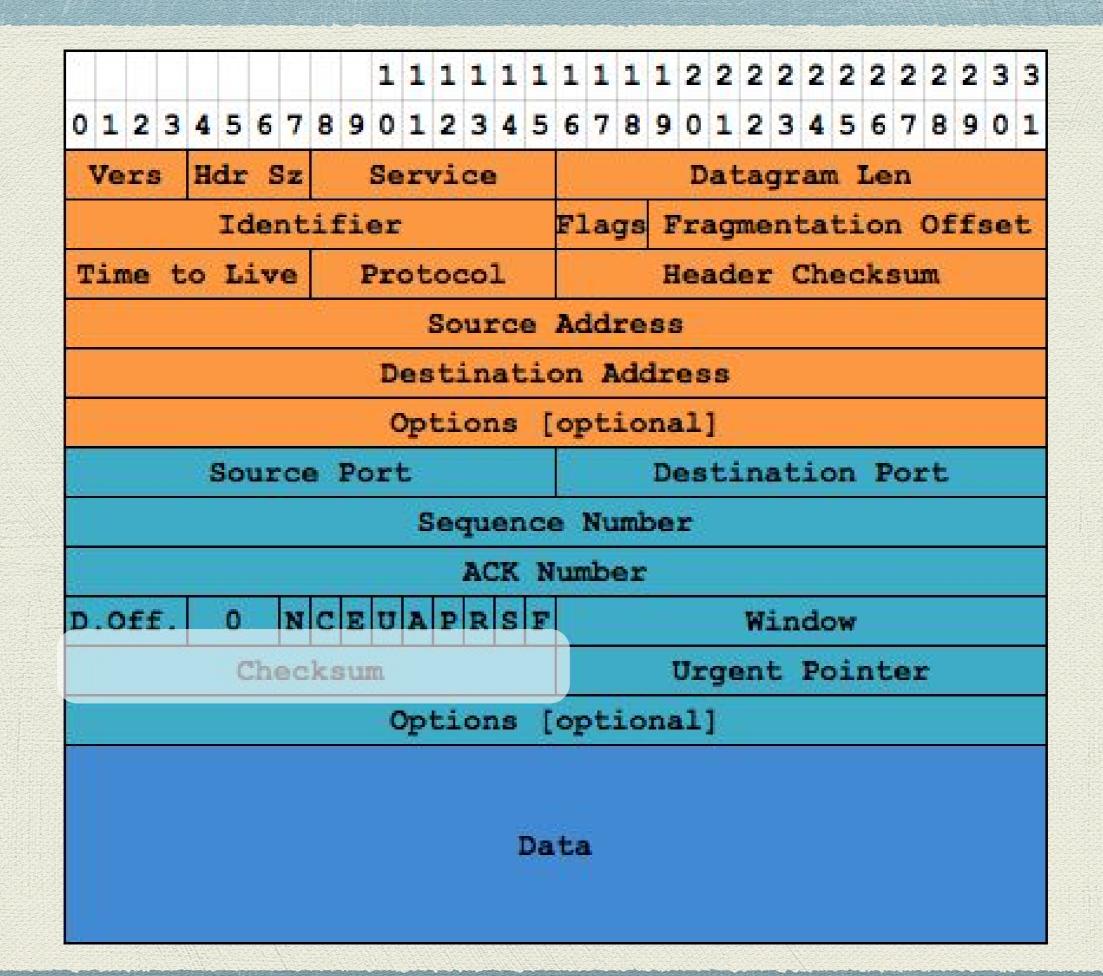


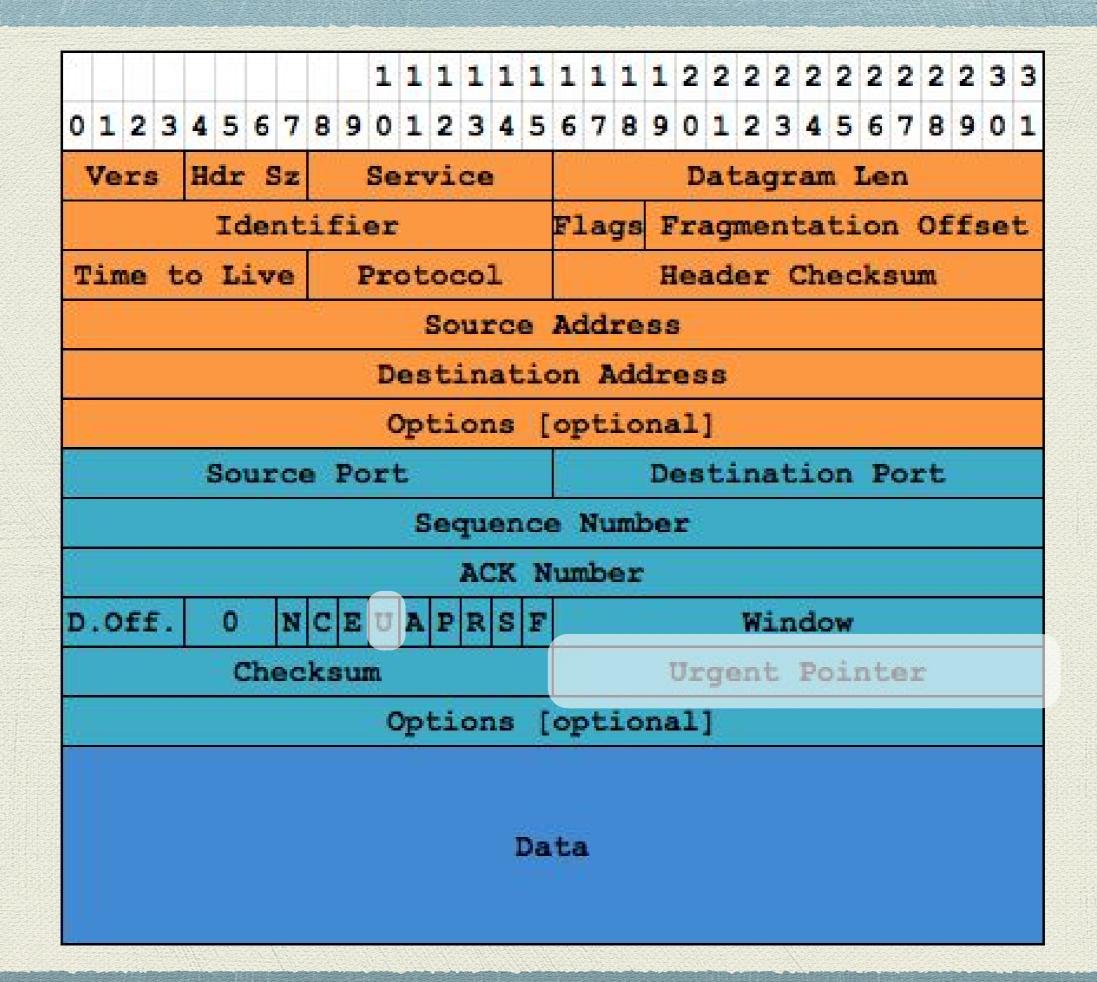
TCP Connections

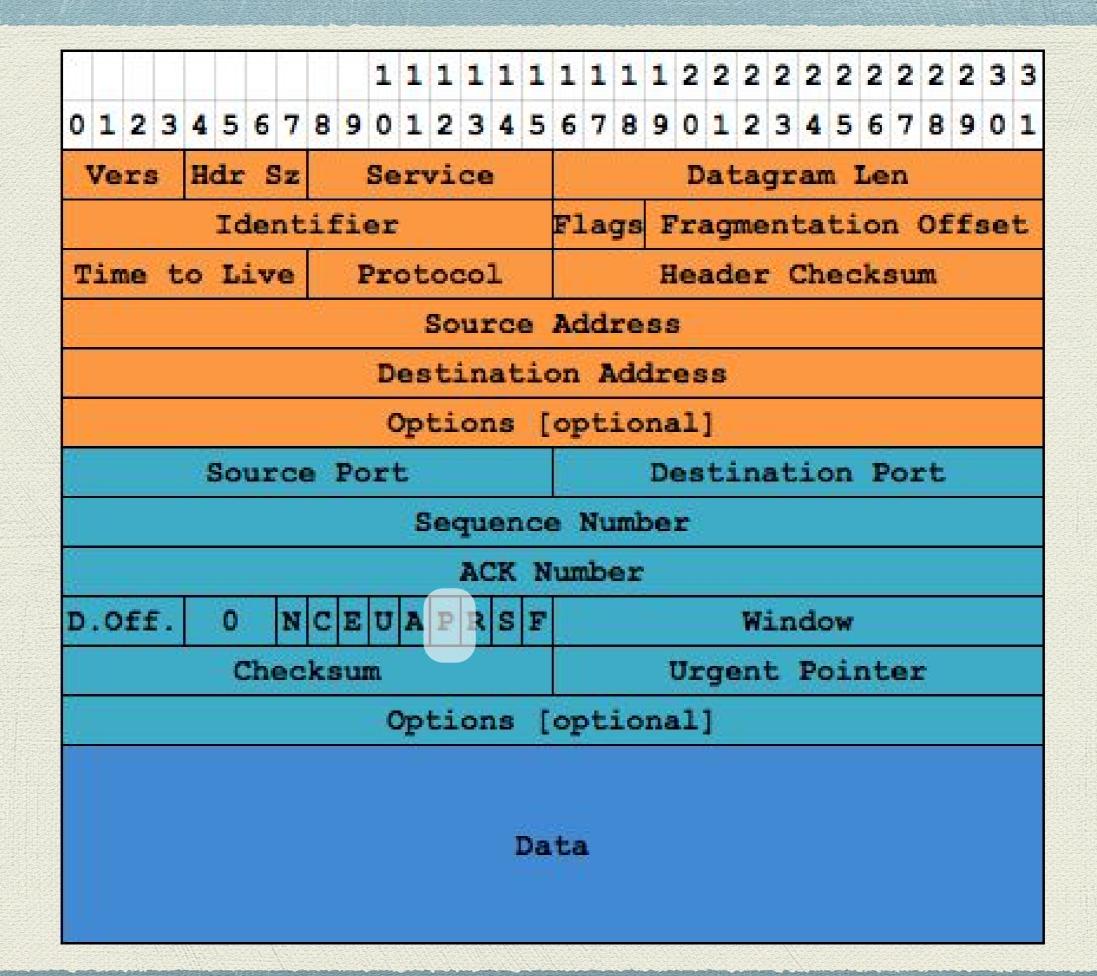
The Deep Dive

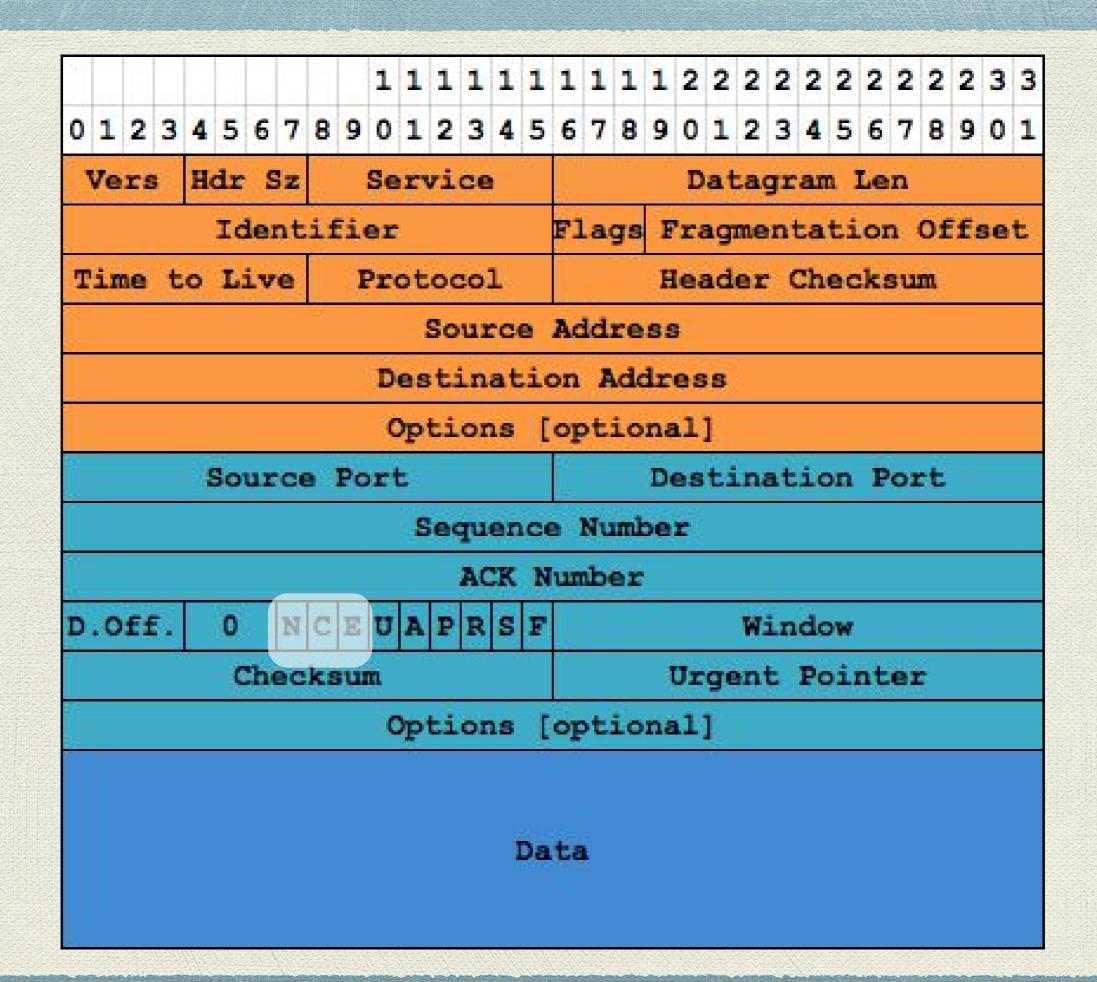
										1	1	1	1	1	1	1	1	1	1	2	2	2	2	2 2	2	2	2	2	2	3
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	2	3 4	5	6	7	8	9	0
	Ve	rs	3	H	dr	5	Sz	10011		Se	r	vi	ce					nt e	- 10	D	at	a	g	rai	m.	Le	n		71110	
					Ic	le	nt	if	i	er						FJ	la	ys.	F	re	ag	me	n	ta	ti	.01	2	Of	£	et
T	in	ne	t	0	L	Ĺν	e		I	r	ot	00	:0:	L					Н	ea	ad	er		Ch	ec	k	su	m		
								8			3	So	ur	c	B	Ac	id:	re	ss											
										D	es	ti	n	at	ic	on	A	de	ire	3 5	s									
										C	þ	ti	or	ıs	I	oľ	ot.	io	na	1	1									
				4	So	uz	ce	•	Po	rt	3								De	s	ti	n	a	ti	on	P	01	et		
											S	ec	Įu	en	C		Nu	mk)e)	•										
													AC	K	N	un	abo	er												
D	.0	£			0		N	C	E	U	A	P	R	s	F							W	i	nd	ow					
					C	h	ec	ks	u	n									τ	Jr	ge	n	t	P	oi	nt	eı			
										C)p	ti	or	s	1	or)t	io	na	1	1									
														1)a	ta														

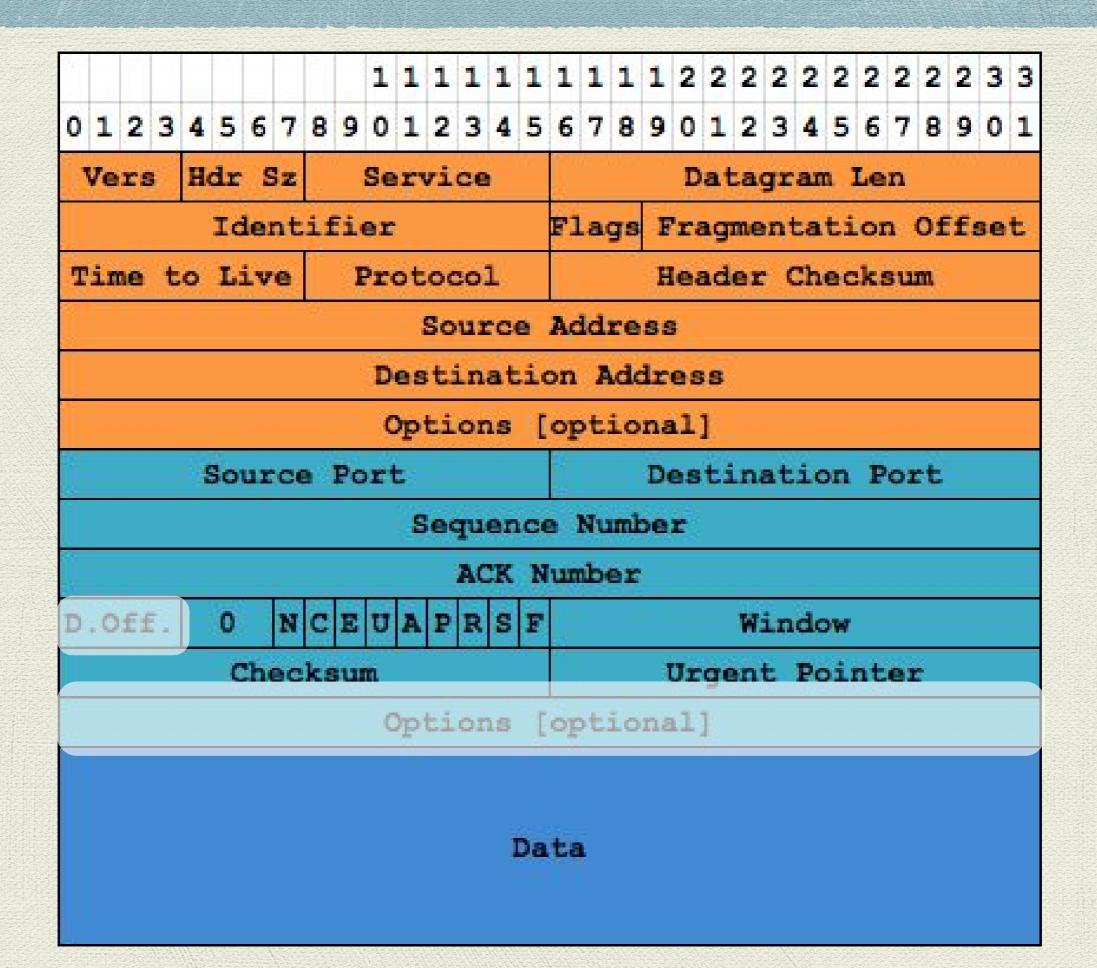
27		100120		e e e						1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	3
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0
	Ve	r	s	Н	dr		Sz	TO THE		Se	r	vi	ce			ecen				D	at	a	gr	ar	n :	Le	n		****	
					I	ie	nt	ií	Eie	er		<u> </u>	2000			Fl	a	gs	F	r	ag	me	n	ta	ti	.01	a ·	Of	£	set
T	in	ae	t	0	L	iv	e		I	r	ot	00	:0:	1					H	e	ad	er		Ch	ec	k	su	m.	Ī	
								8			3	So	ur	c	B	Ad	d	re	ss	,										
										D	es	ti	n	at	ic	on	A	de	lr	es	s									
										C	g(ti	or	ıs	I	or	ŧ.	io	na	1	1									
				13	So	uz	200	9	Po	ri	-					1			De	es	ti	n	at	iic	on	P	oz	et		
											S	ec	Įu	en	C		Vu	mì)e	r										
													AC	K	N	un	b	er												
D	.0	£	Ē.		0		N	C	E	U	A	P	R	s	F							W	ir	ıdı	w					
					(Ch	ec	ks	u	n.									1	Jr	ge	n	t	Po	oi	nt	eı	ě		
										C	ď,	ti	or	ıs	E	op	t	io	na	1	1									
														1)a	ta														

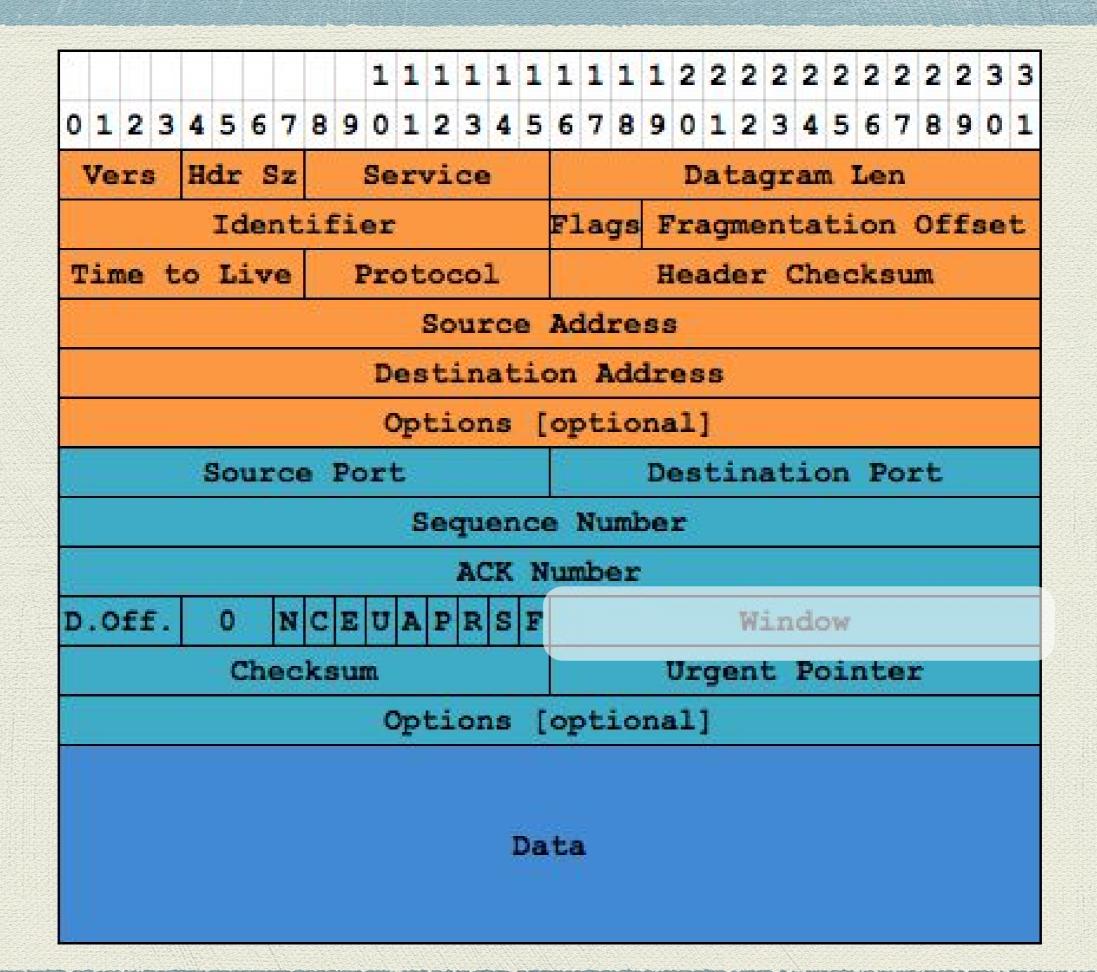


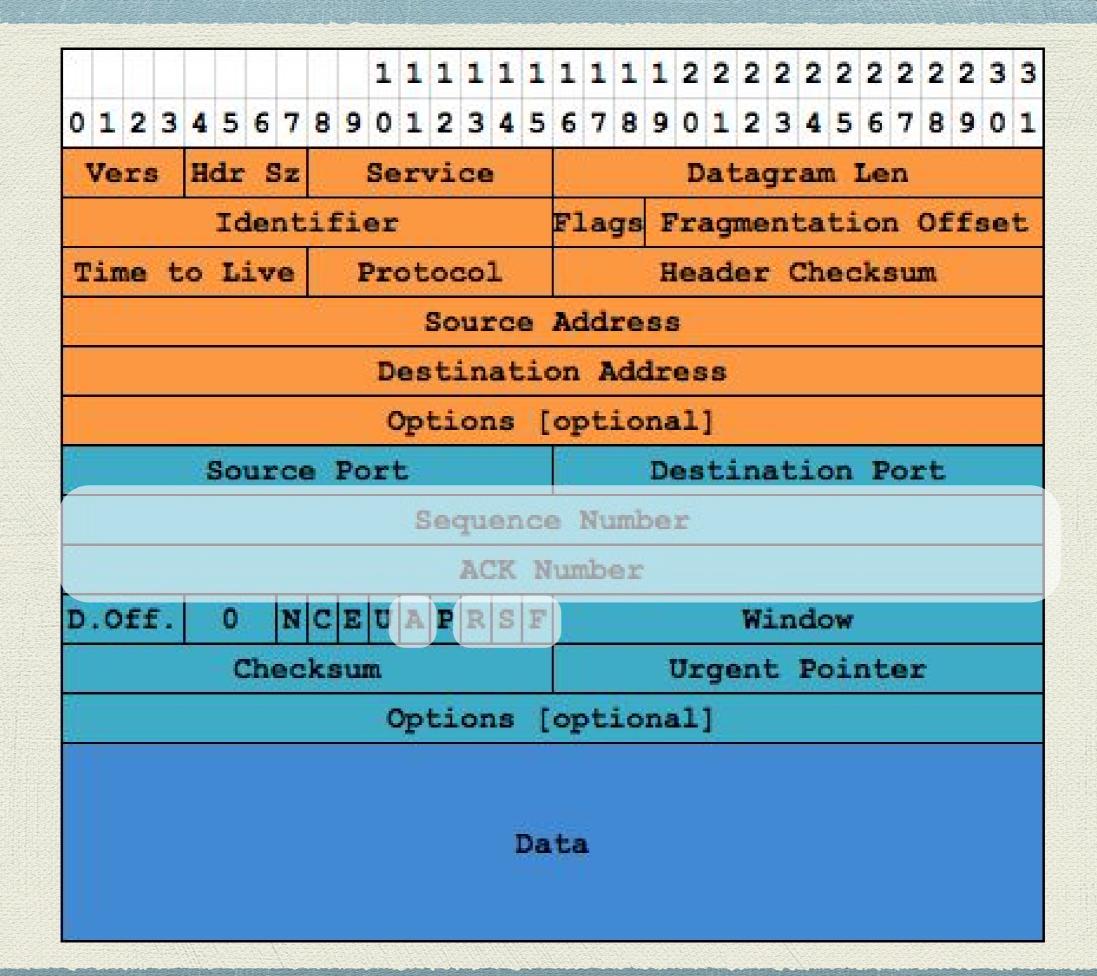


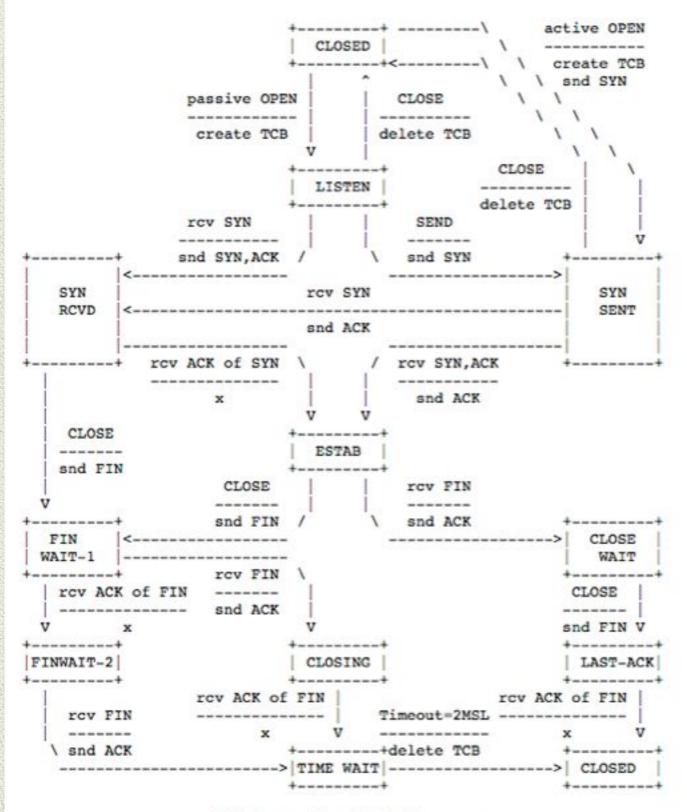




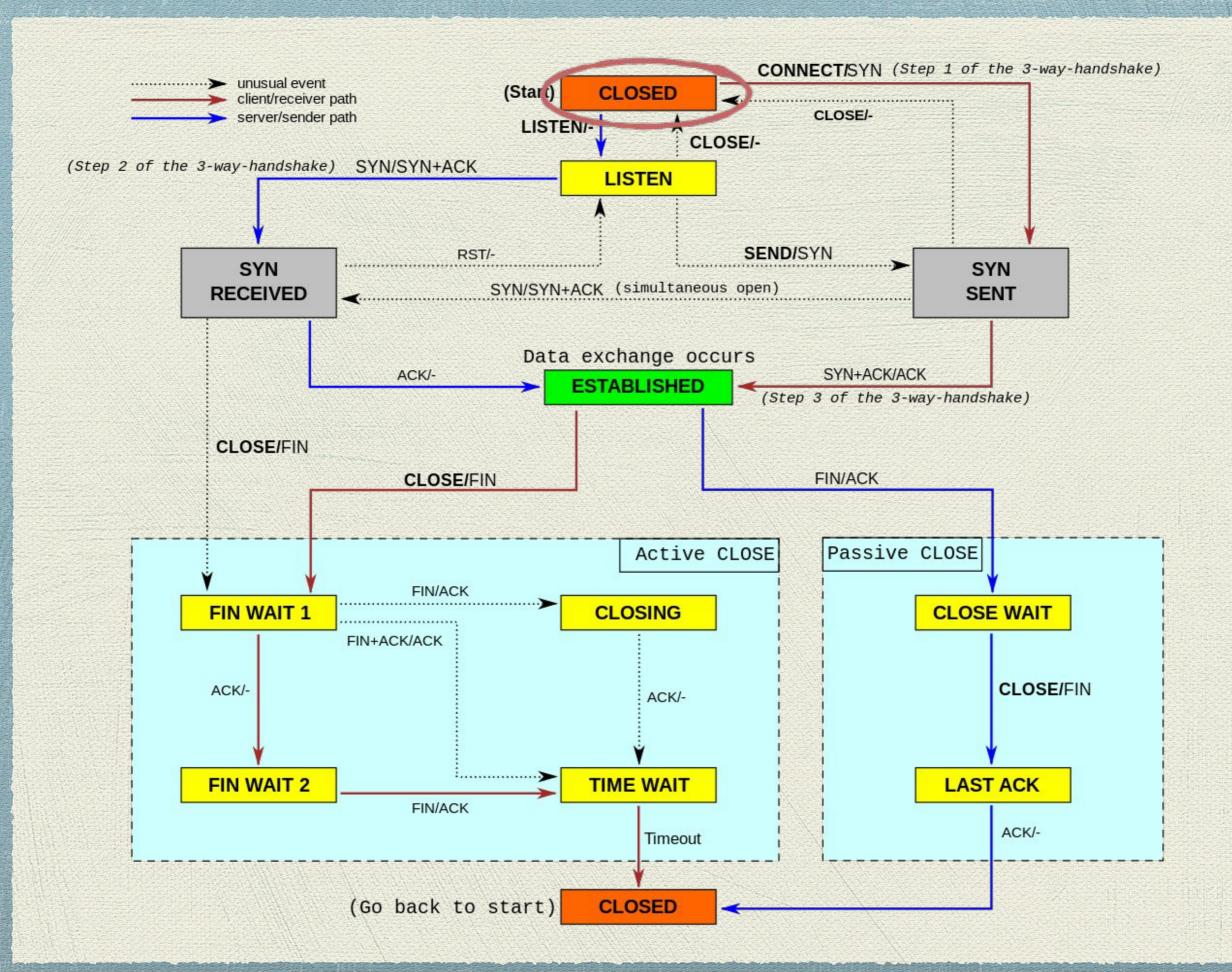


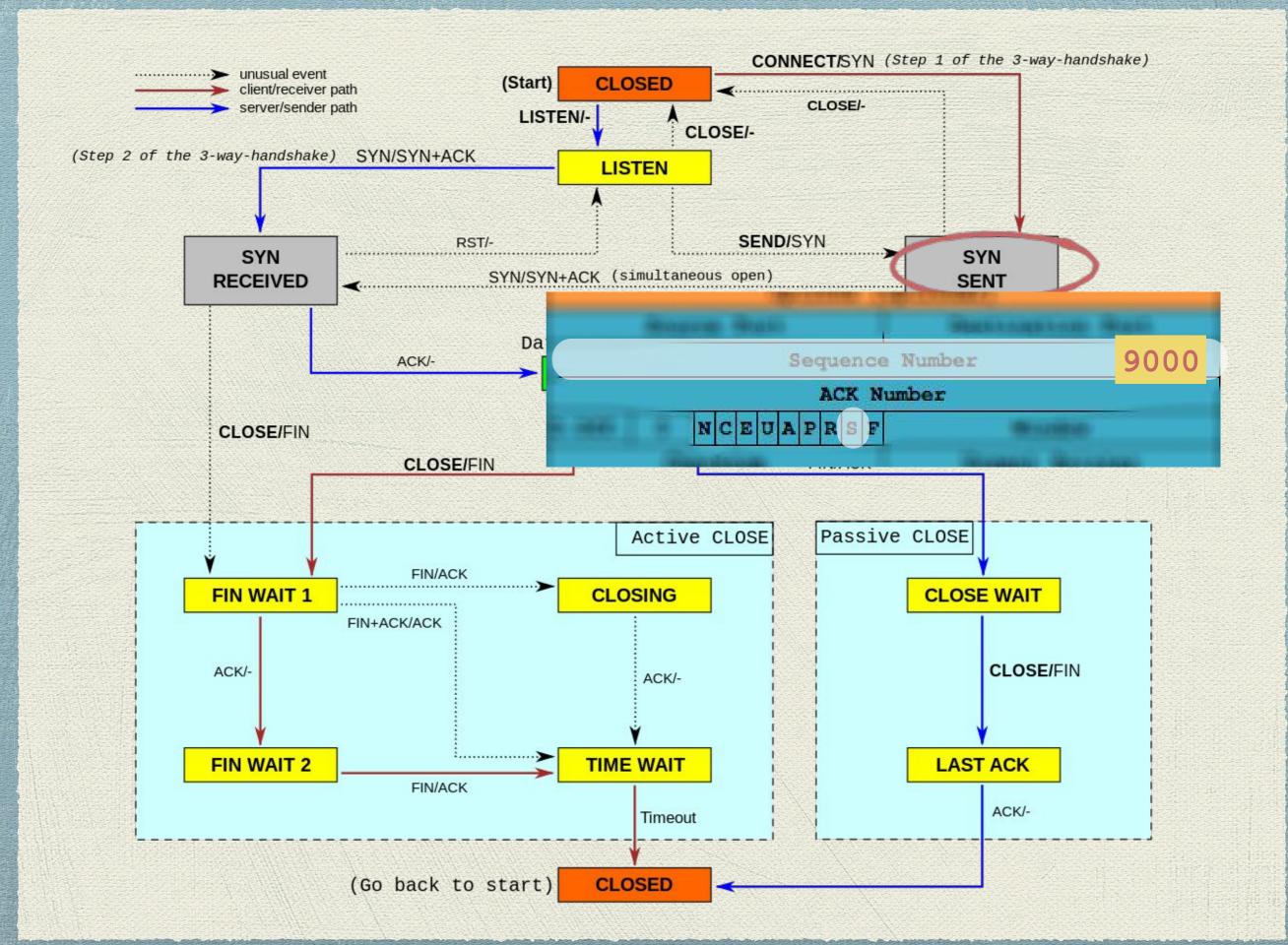


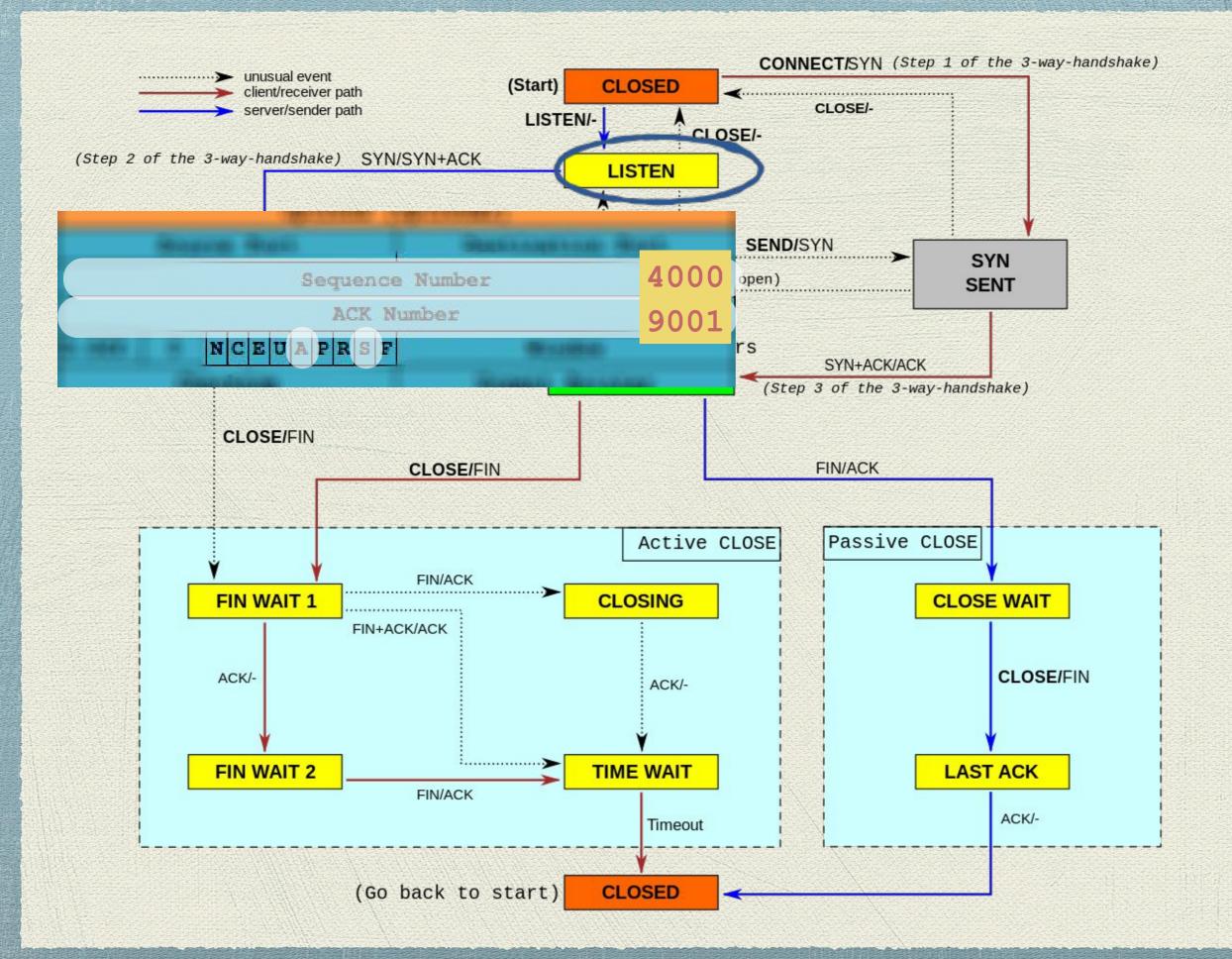


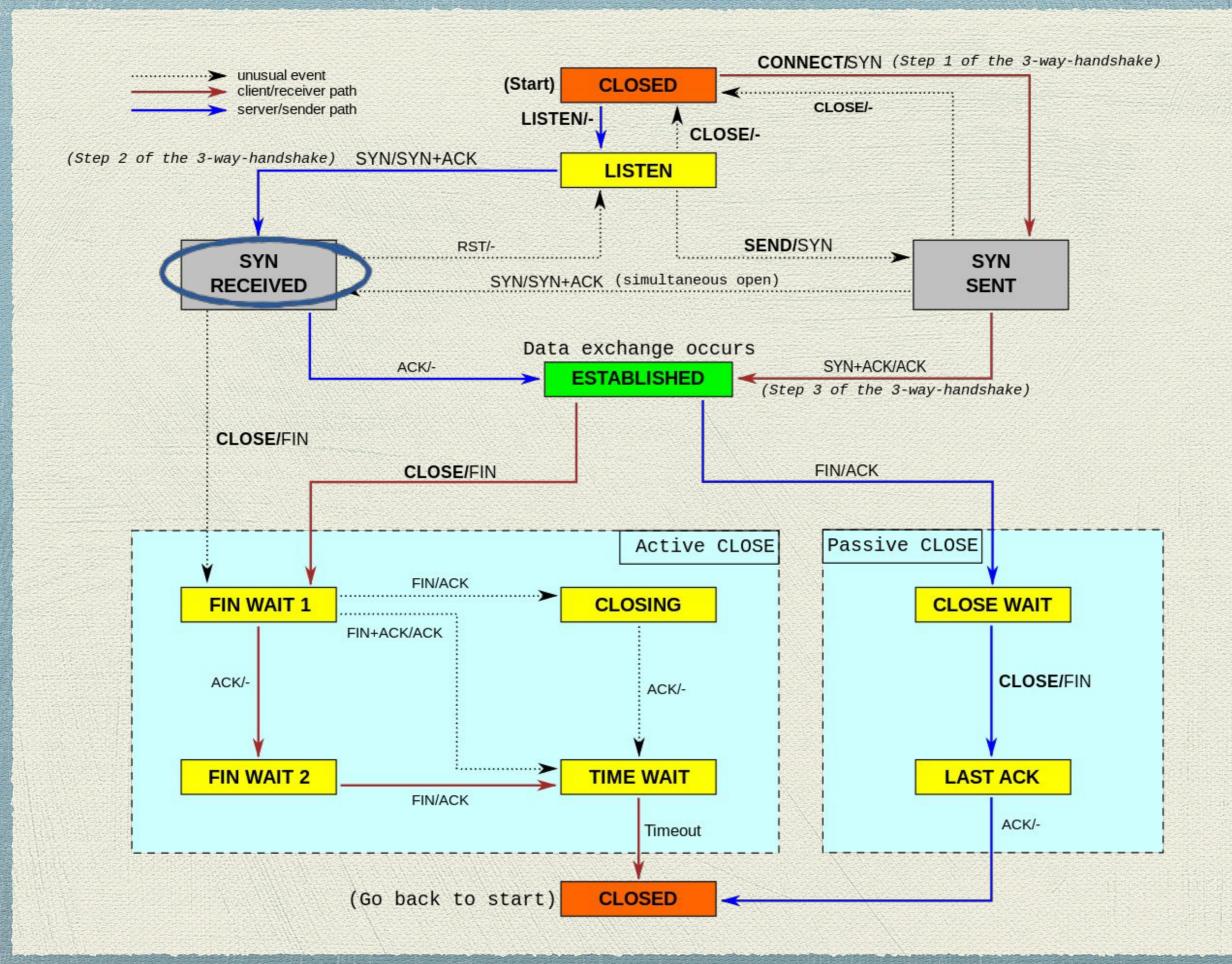


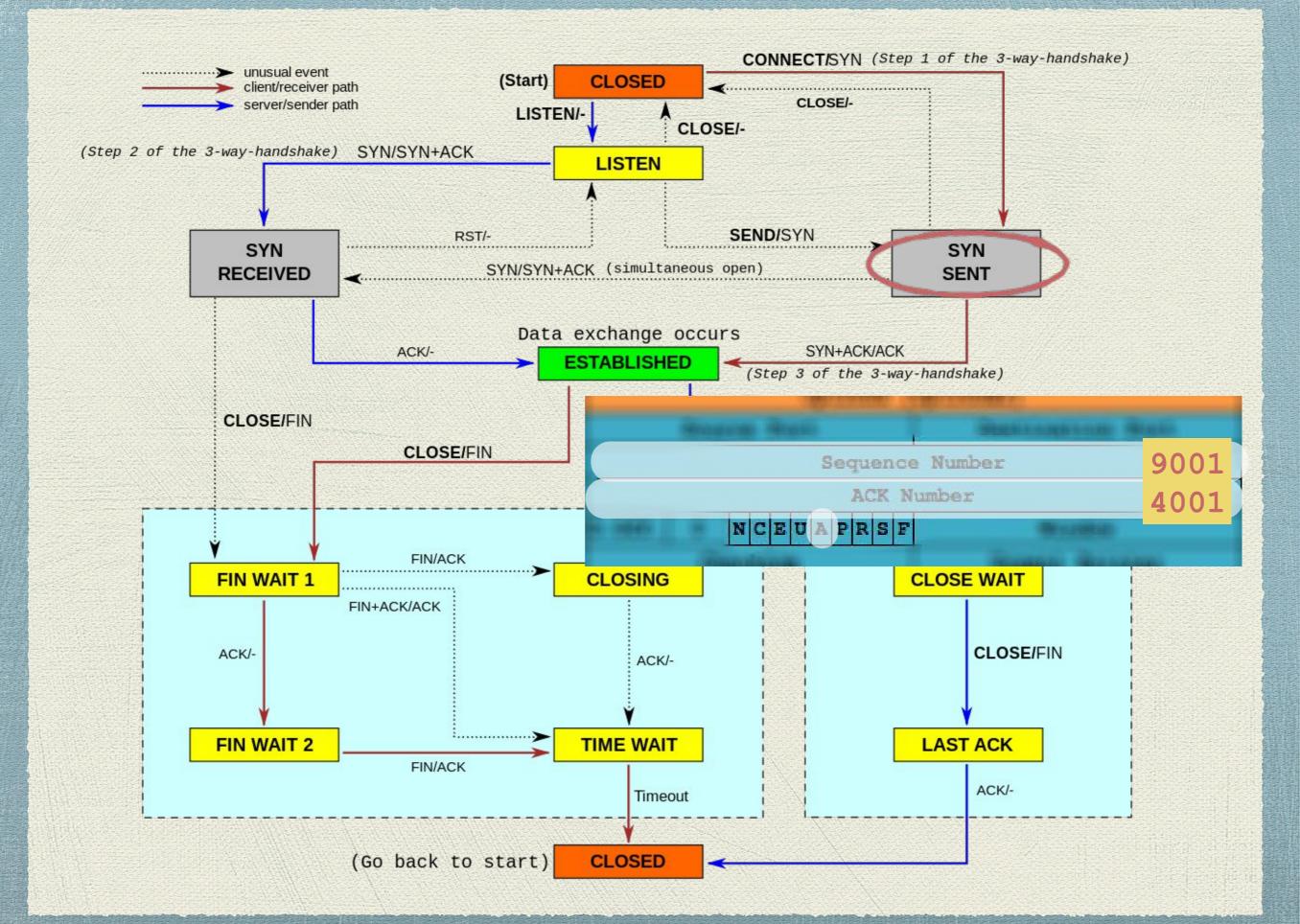
TCP Connection State Diagram Figure 6.

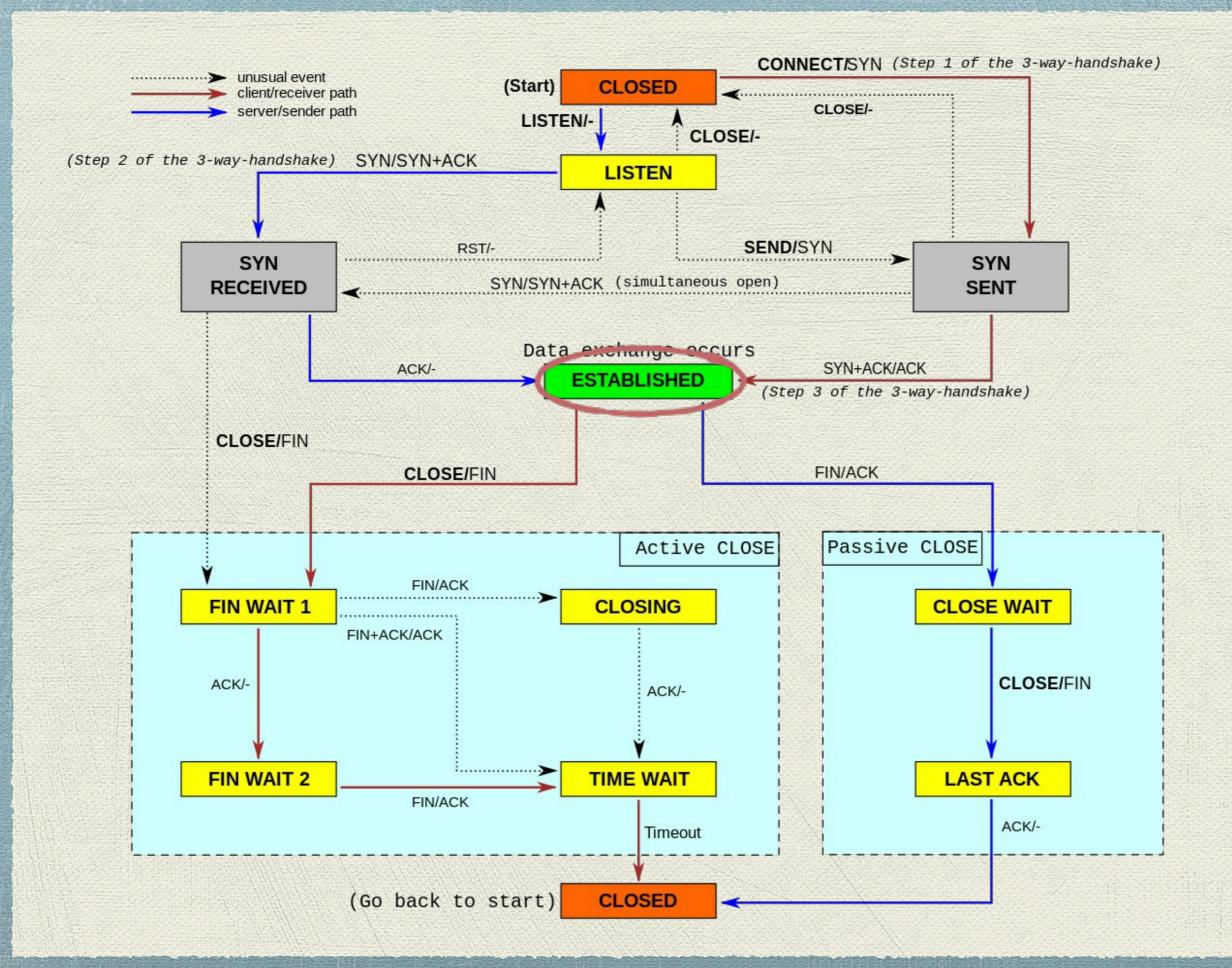


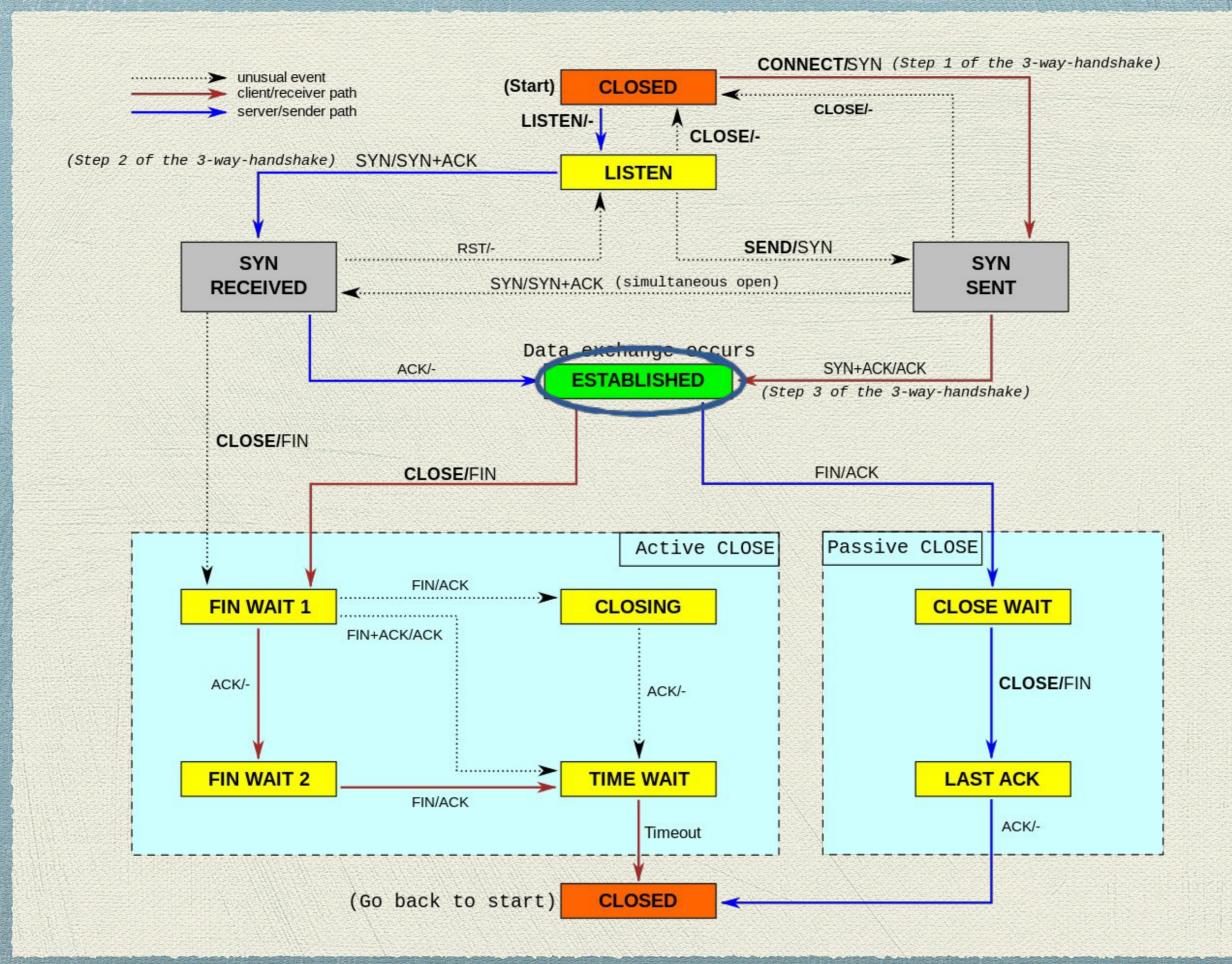












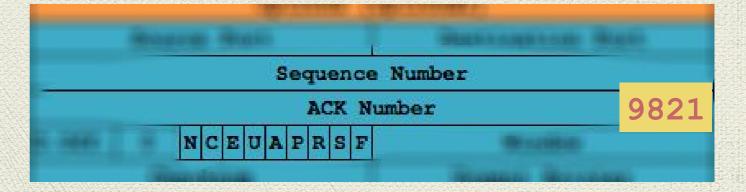
Sequence Number

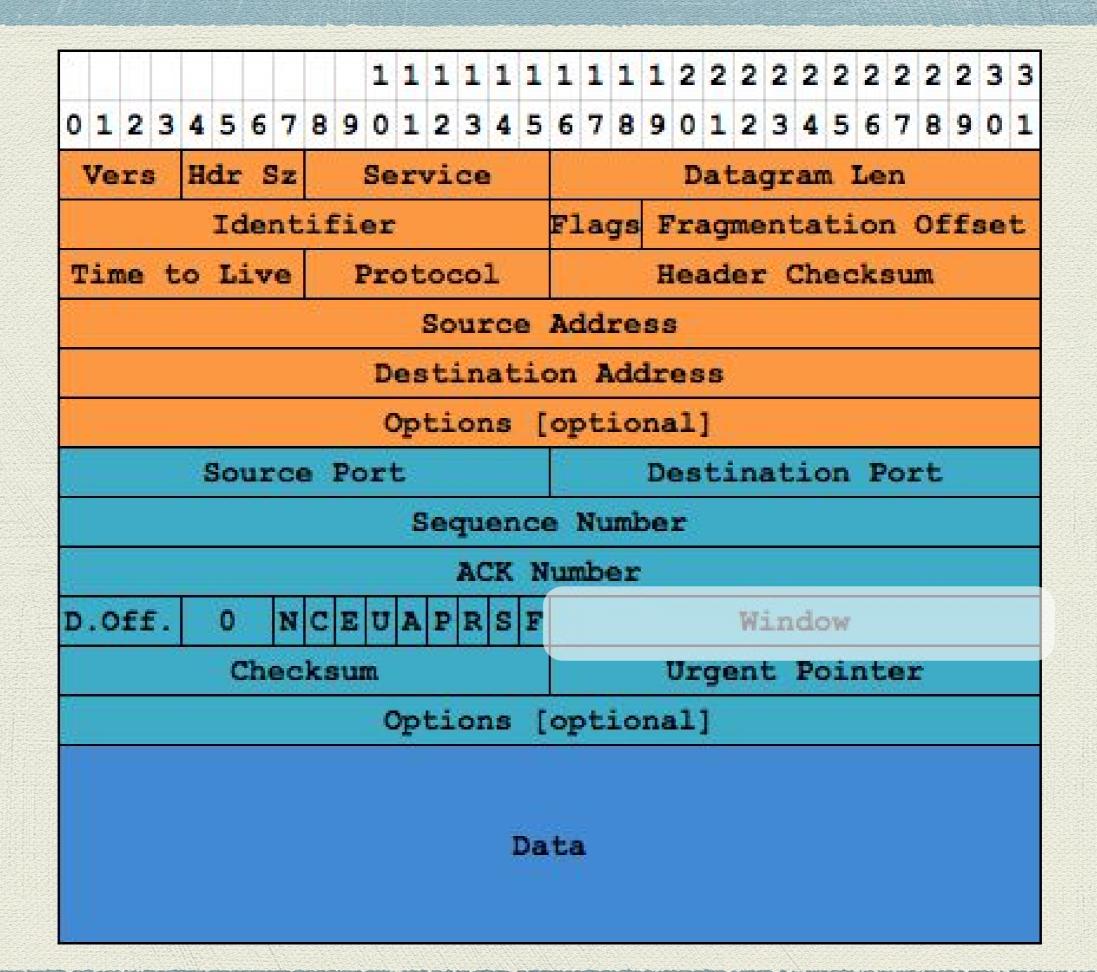
ACK Number

NCEUAPRSF

[500 bytes of data]

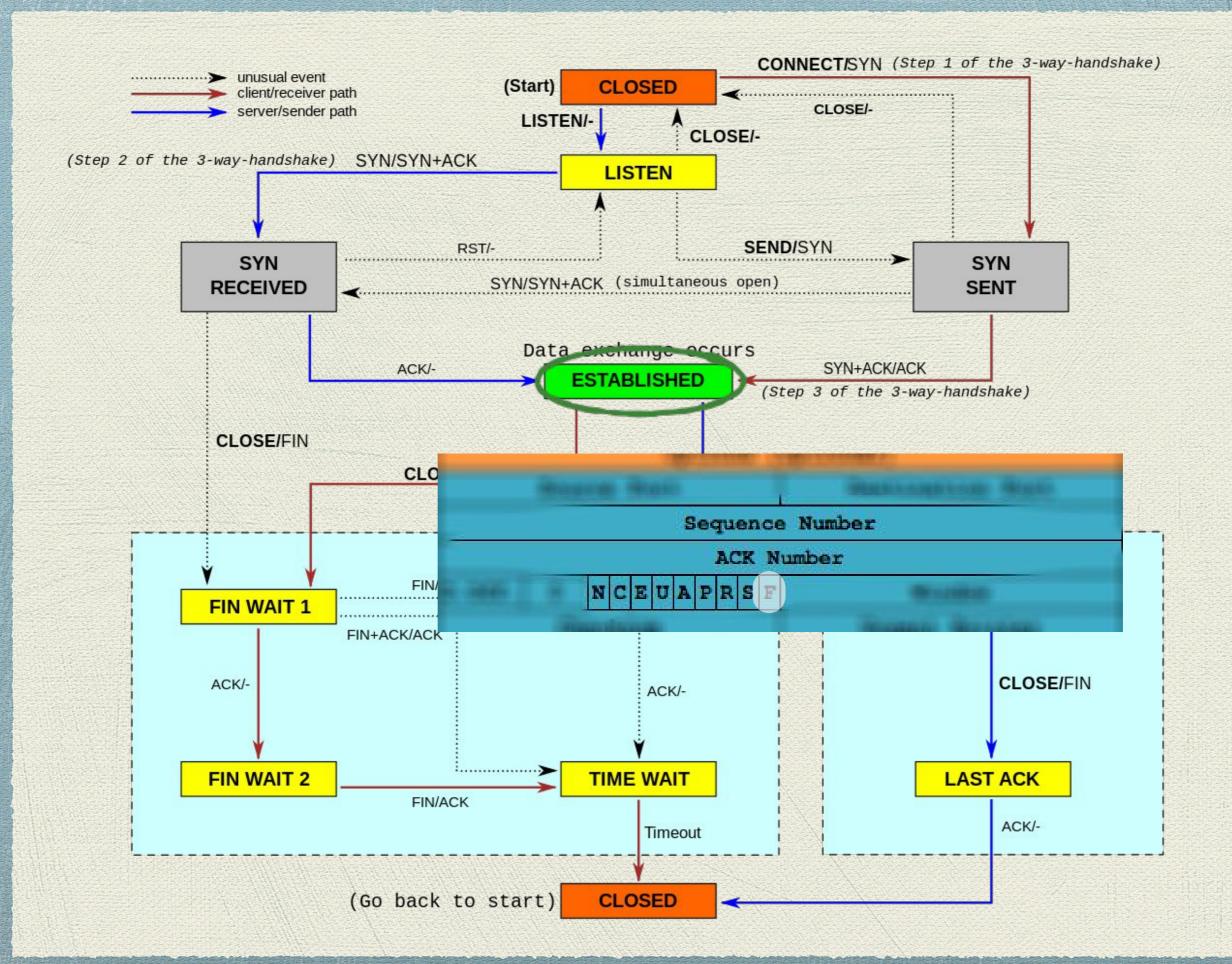
Packets Bytes

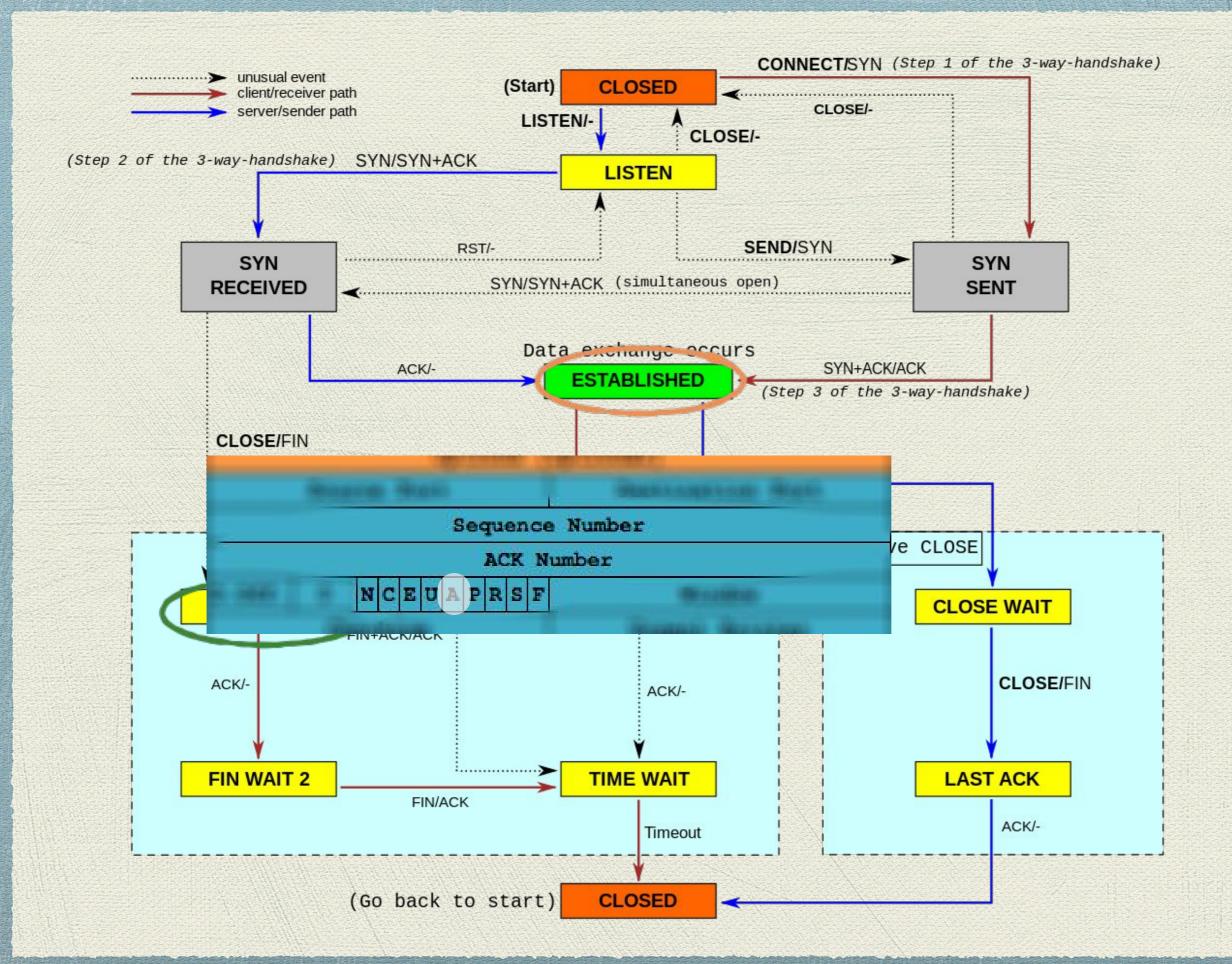


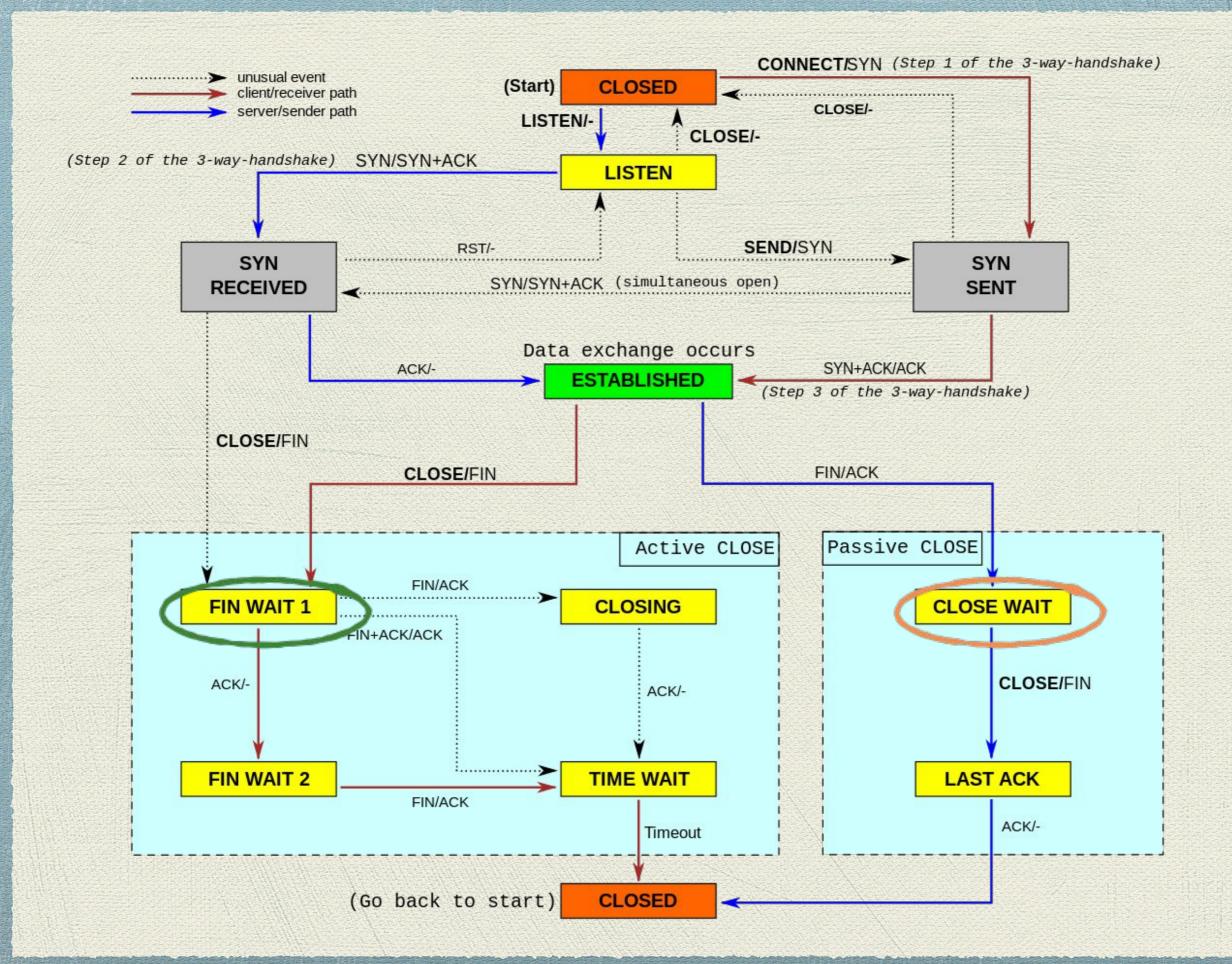


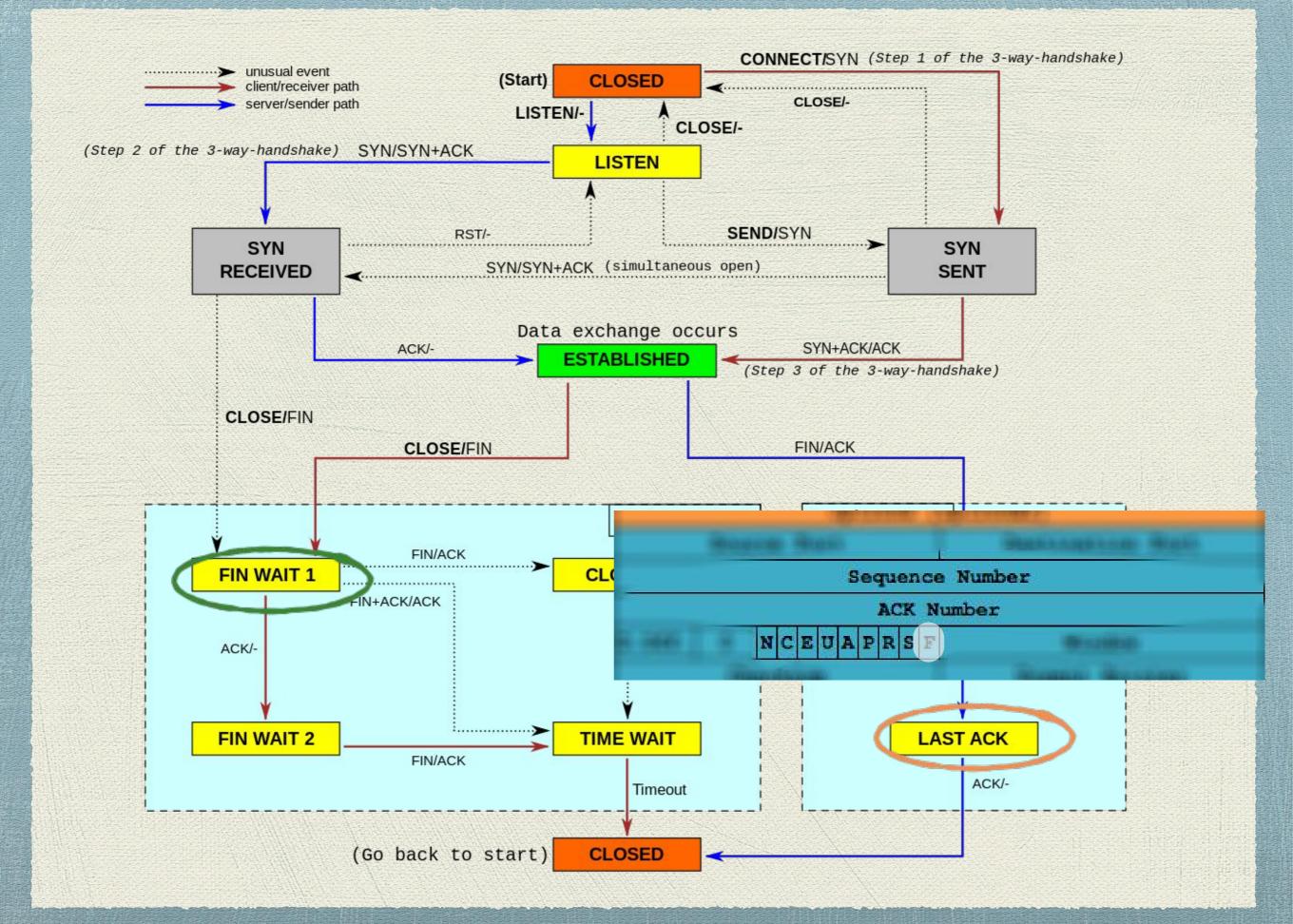
Flow Control Tricks

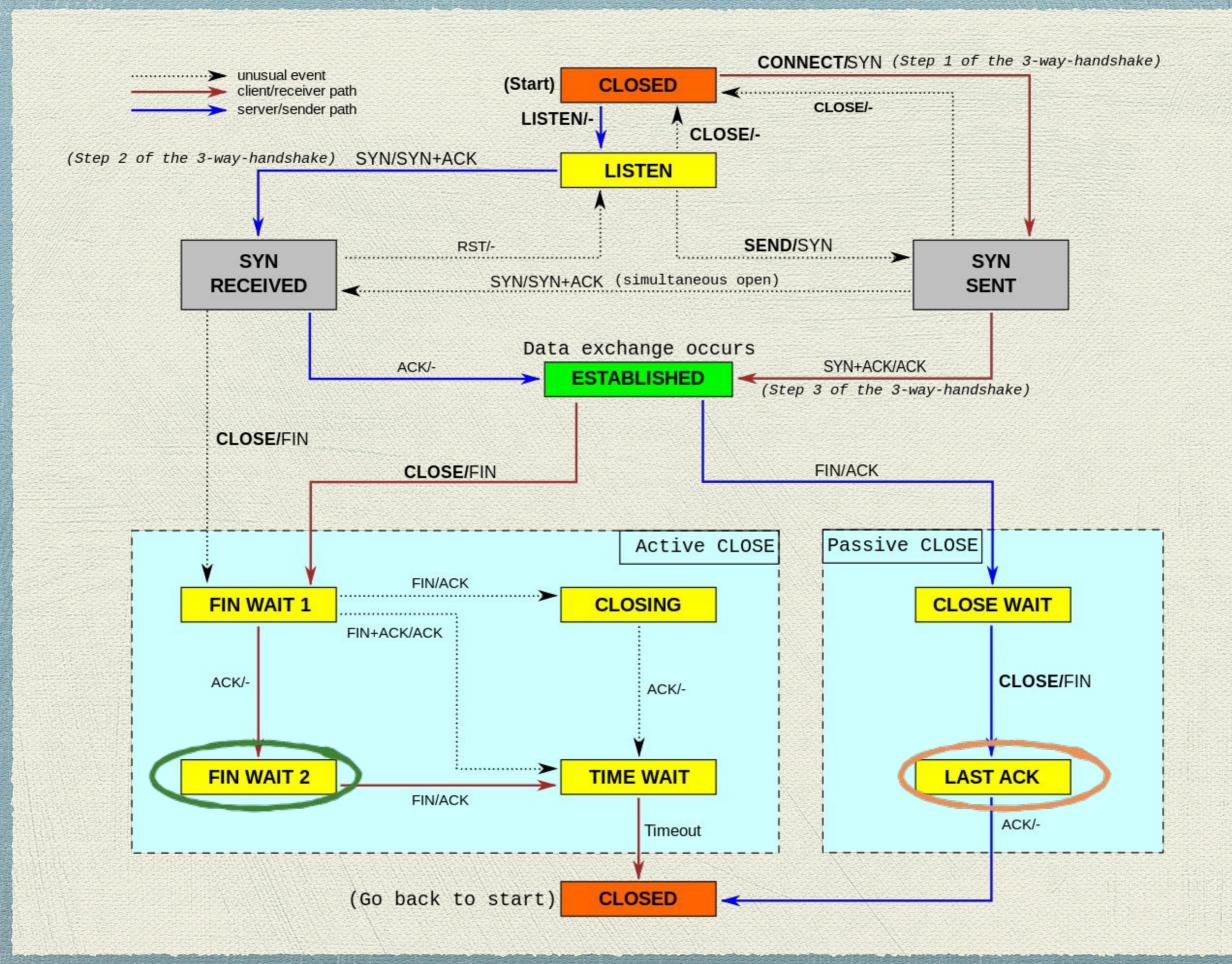
- Selective Acknowledgement [RFC1072 & RFC2018]
- Karn's Algorithm[RFC2988 & RFC6298]
- Slow Start, Congestion Avoidance, Fast Retransmit, Fast Recovery [RFC2581]
- Nagle's Algorithm [RFC896]
- Explicit Congestion Notification [RFC3168 and RFC3540]

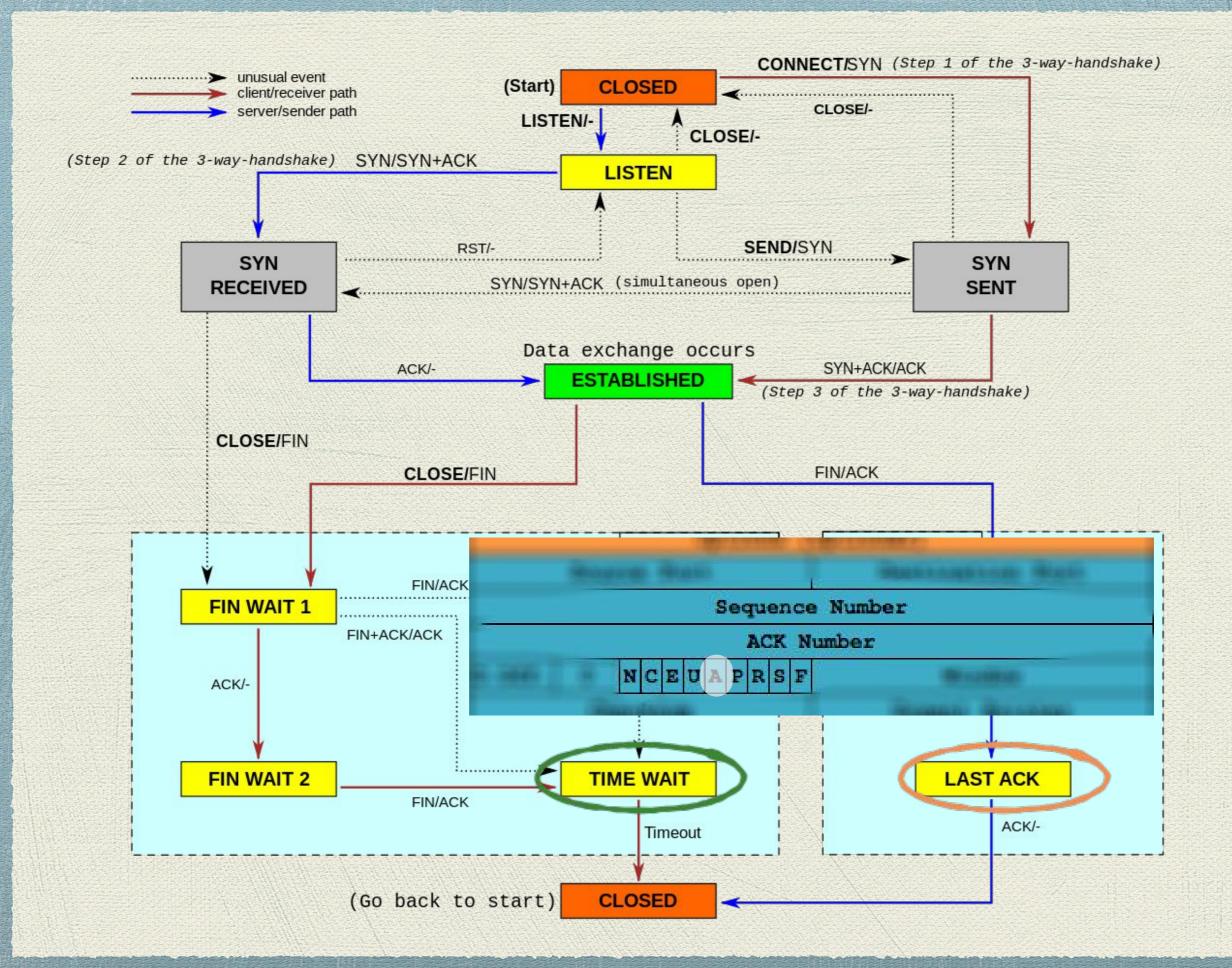


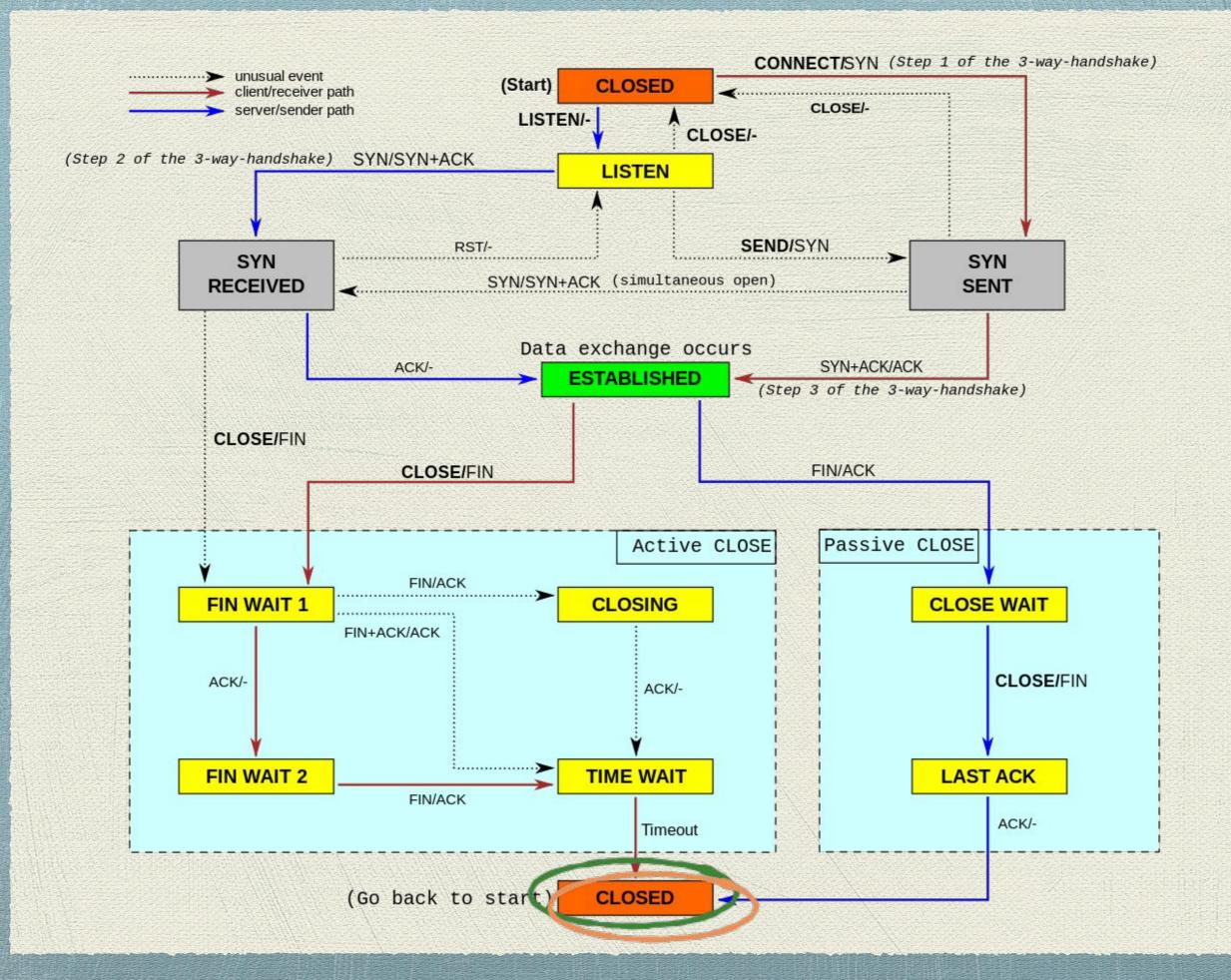












										1	1	1	1	1	1	1	1	1	1	2	2	2	2	2	2	2	2	2	2	3
0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0	1	2	3	4	5	6	7	8	9	0
	Ve	r	;	H	dr	5	Sz	ecen Se		Se	r	vi	ce							D	at	a	gr	an	a :	Le	n			
					Id	le	nt	if	i	er						F]	la	gs	F	r	ag	me	nt	a	ti	OI	n	Of	f	set
T	in	ne	t	0	L	iv	e		I	r	ot	oc	0	L					H	e	ad	er	. (h	ec	k	su	m		
								20			3	So	ur	c	9	Ac	id	re	88	;										
										D	es	ti	n	at	ic	on	1	de	ir	es	s									
										C	þ	ti	on	ıs	[or	ot.	io	na	11	1									
				3	So	ur	C	•	Pc	ri									De	es	ti	n	at	ic	n	P	or	t		
											S	eç	Įu.	en	C	•	Nu	mì	oe.	r										
													AC	K	N	ur	nb	er												
D	.0	£			0		N	C	E	U	A	P	R	3	F							W	in	do	w					
					C	h	ec	ks	ıuı	n									1	Jr	ge	n	t	Po	ii	nt	eı			
										C)p	ti	on	s	1	or	ot.	io	na	1	1									
)a	ta														

