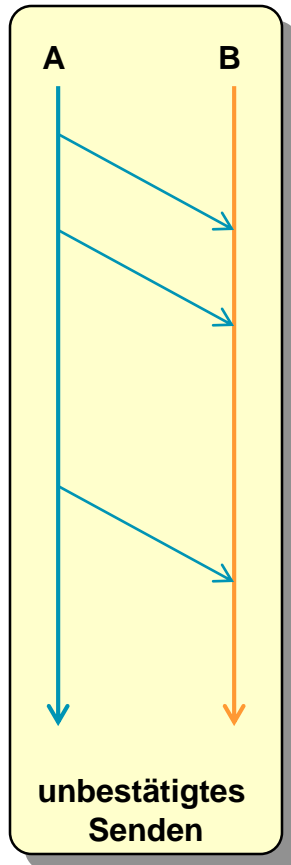
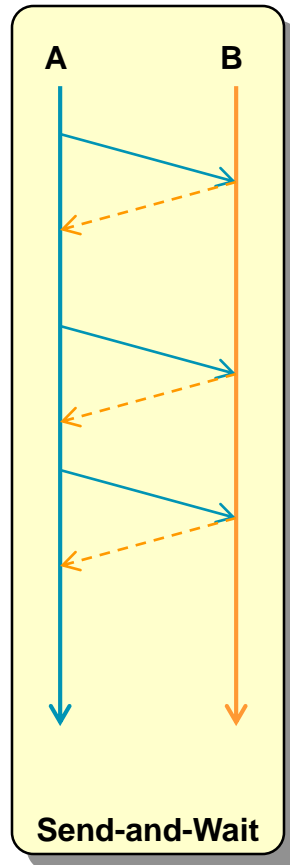




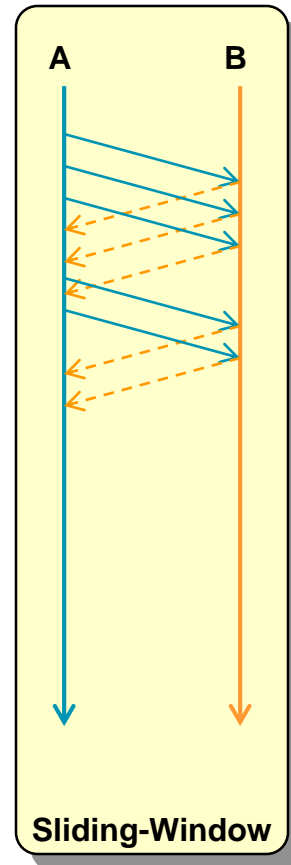
Prinzip des Sliding-Window: Zuverlässigkeit + Effizienz



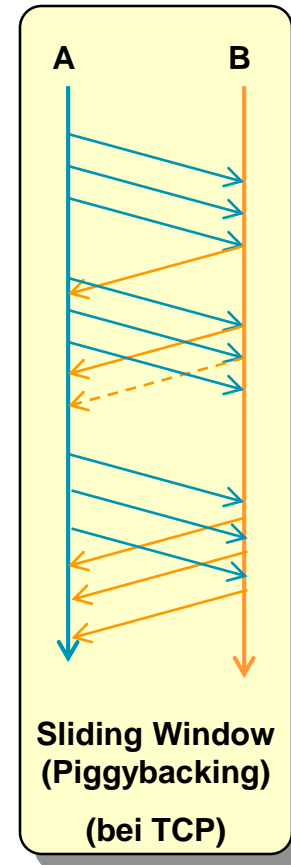
unzuverlässig
effizient



zuverlässig
ineffizient



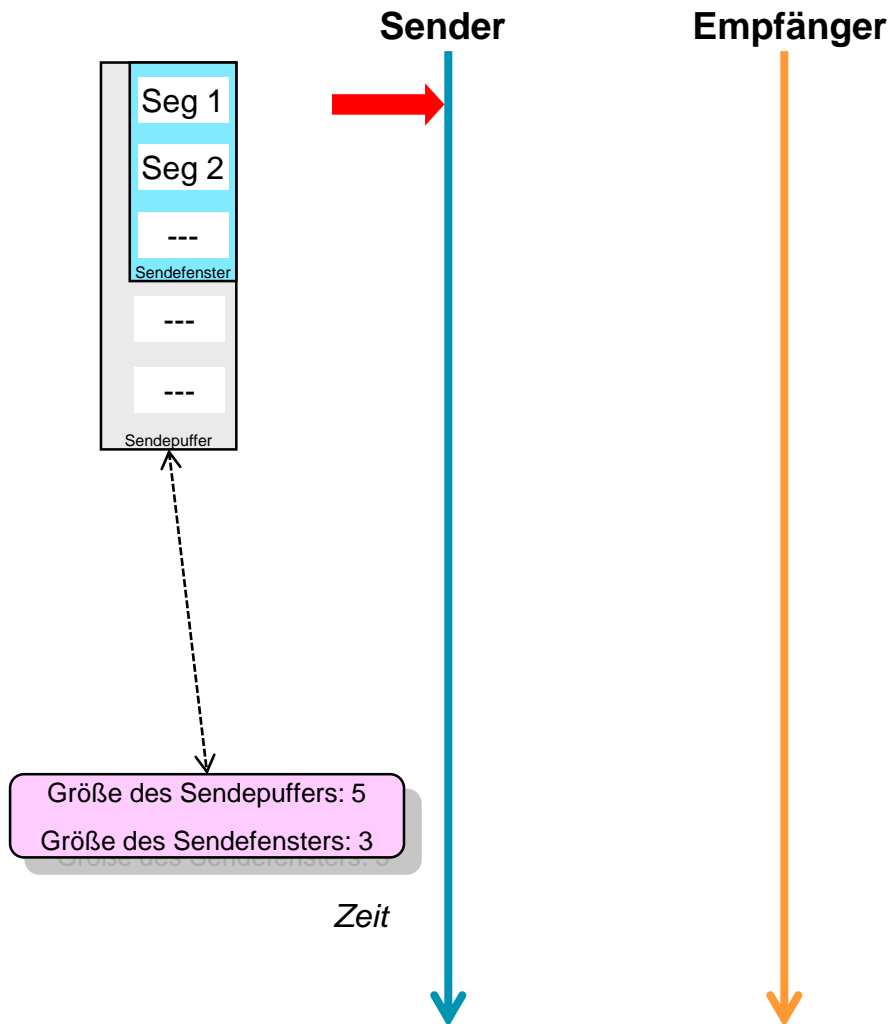
zuverlässig
effizient



zuverlässig
effizient



Sliding-Window - detaillierter



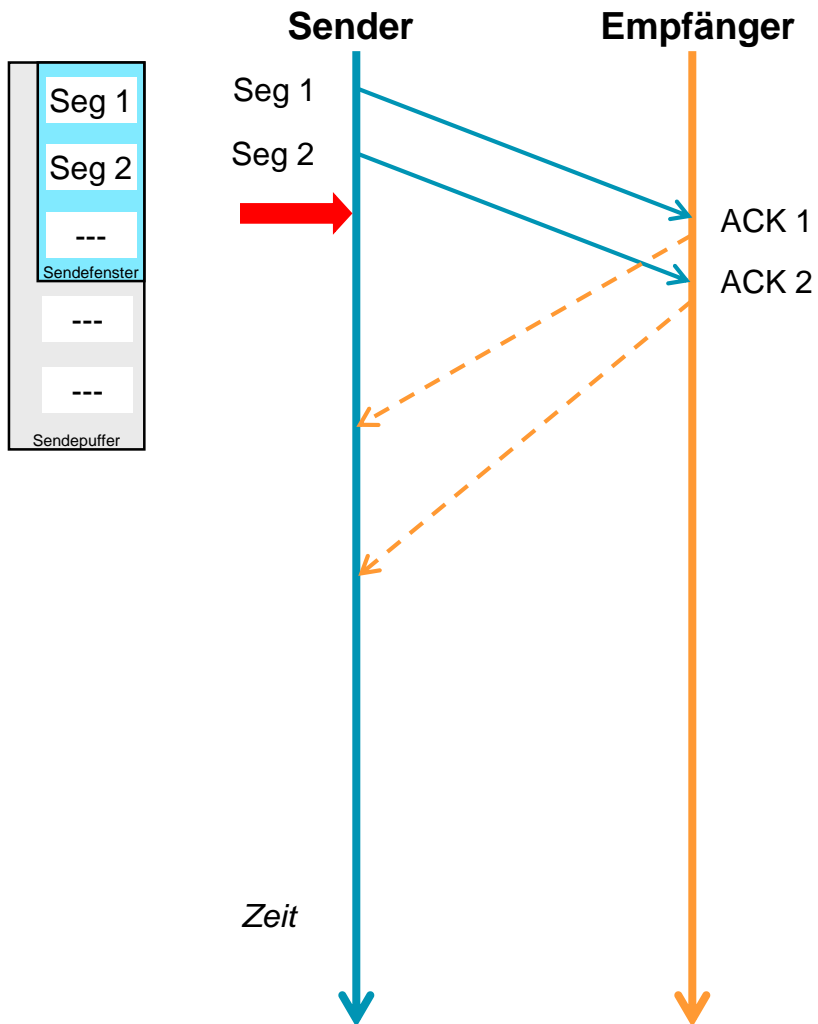
- Anwendung produziert 2 Segmente
- Diese Segmente liegen im Sendepuffer und im Sendefenster

Was ist der Unterschied zwischen Sendepuffer und Sendefenster?

Was ist die Aufgabe von Sendepuffer und Sendefenster?



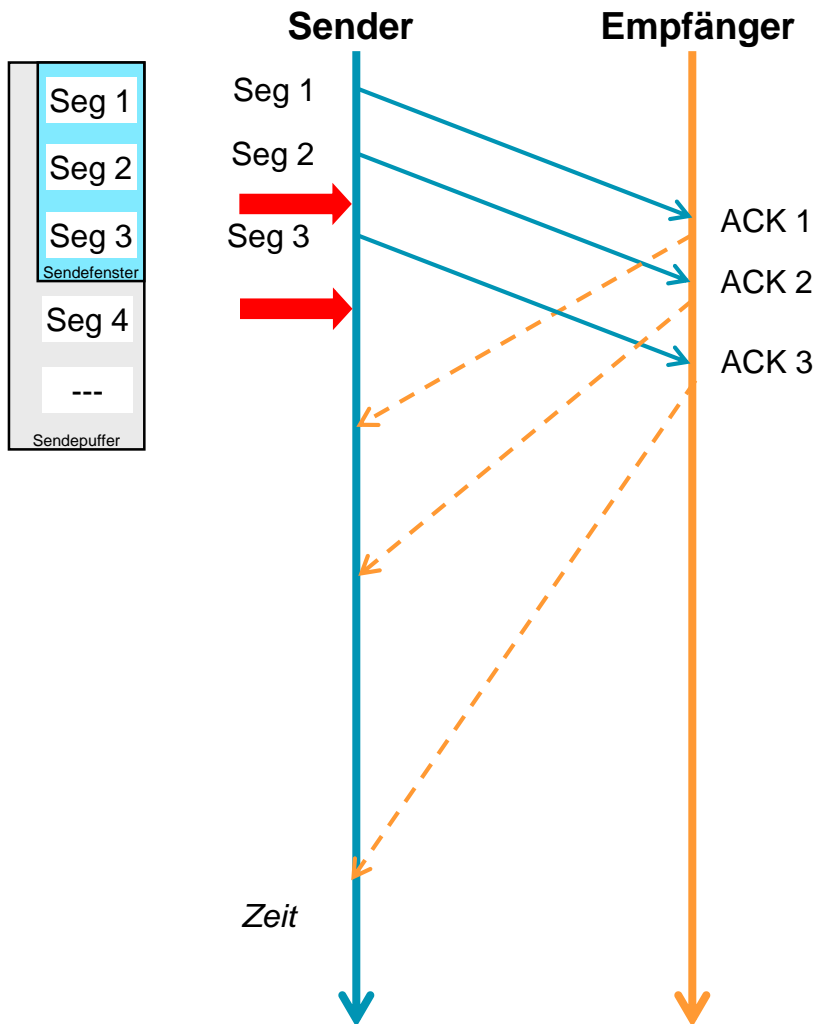
Sliding-Window - detaillierter



- Was passiert jetzt??
- Beide Segmente können gesendet werden
- Auswirkungen auf den Sendepuffer und das Sendefenster?



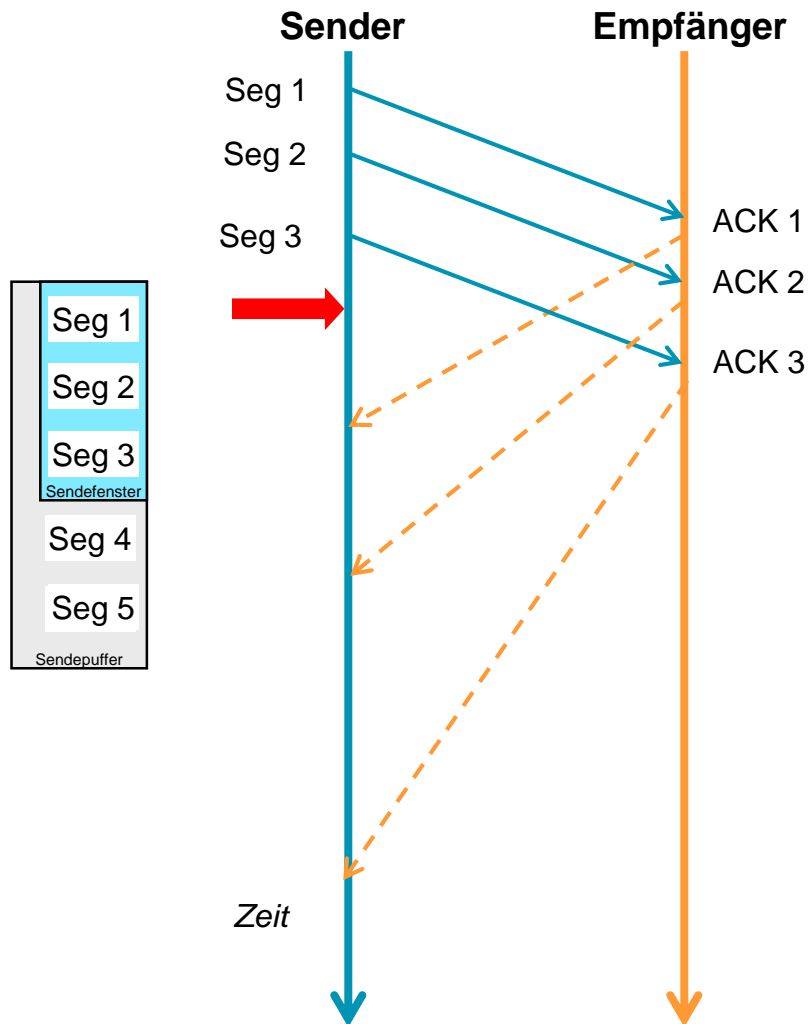
Sliding-Window - detaillierter



- Anwendung produziert zwei weitere Segmente
- Was macht der Sender?



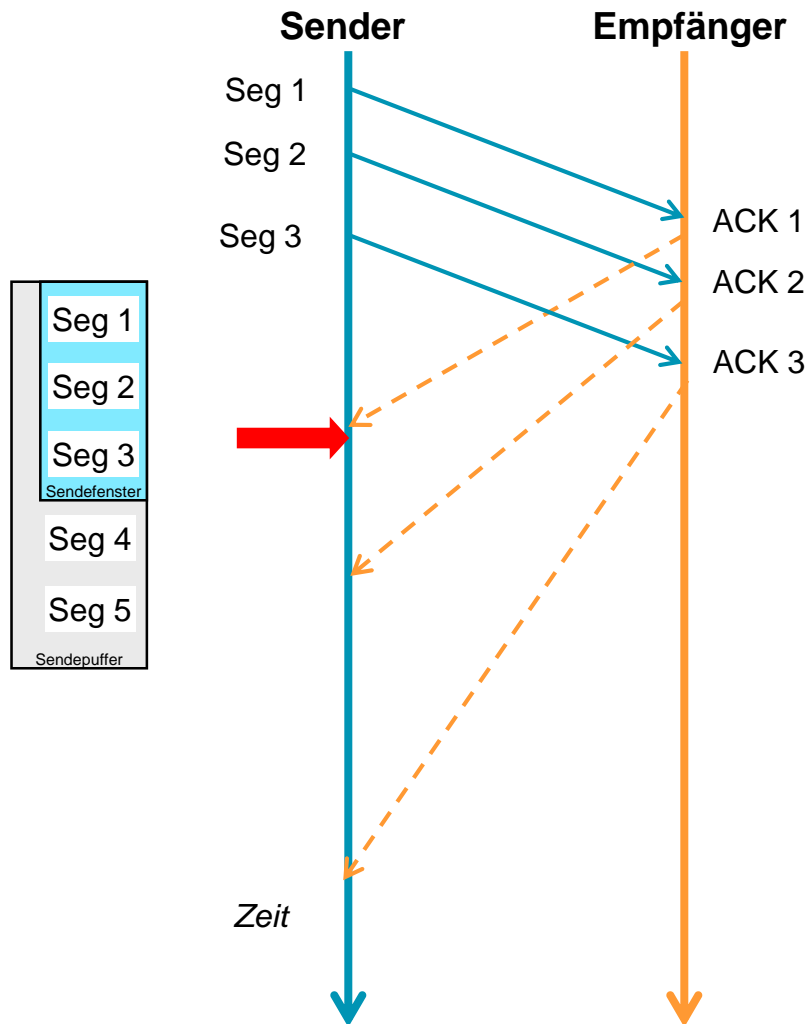
Sliding-Window - detaillierter



- Anwendung produziert ein weiteres Segment
- Was passiert?
- Was wäre, wenn sie zwei weitere Segmente produziert?



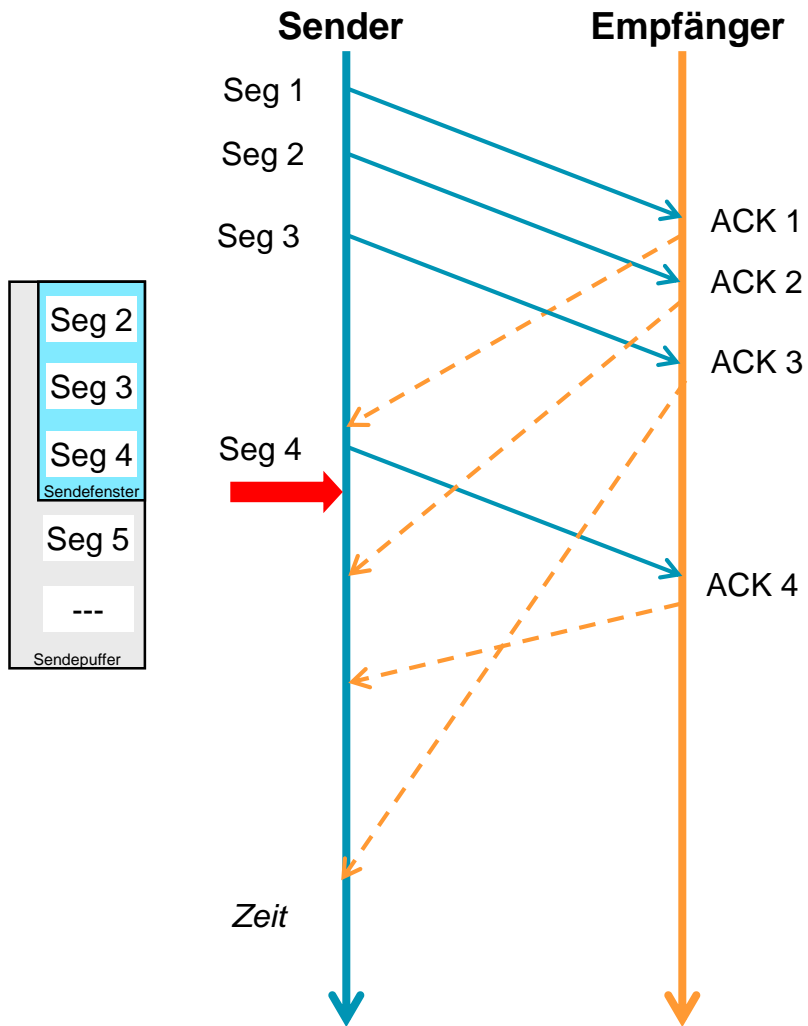
Sliding-Window - detaillierter



- ACK 1 kommt nun beim Sender an
- Was passiert?



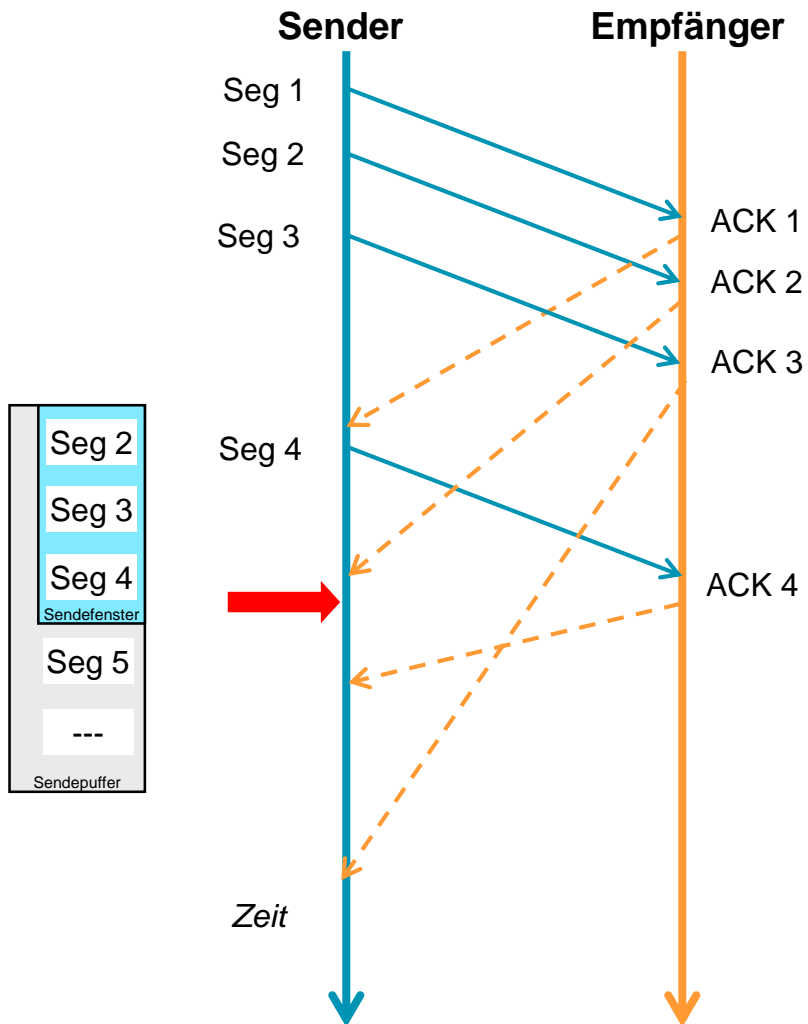
Sliding-Window - detaillierter



- **Schieben des Fensters**
- **Versenden von Segment 4**



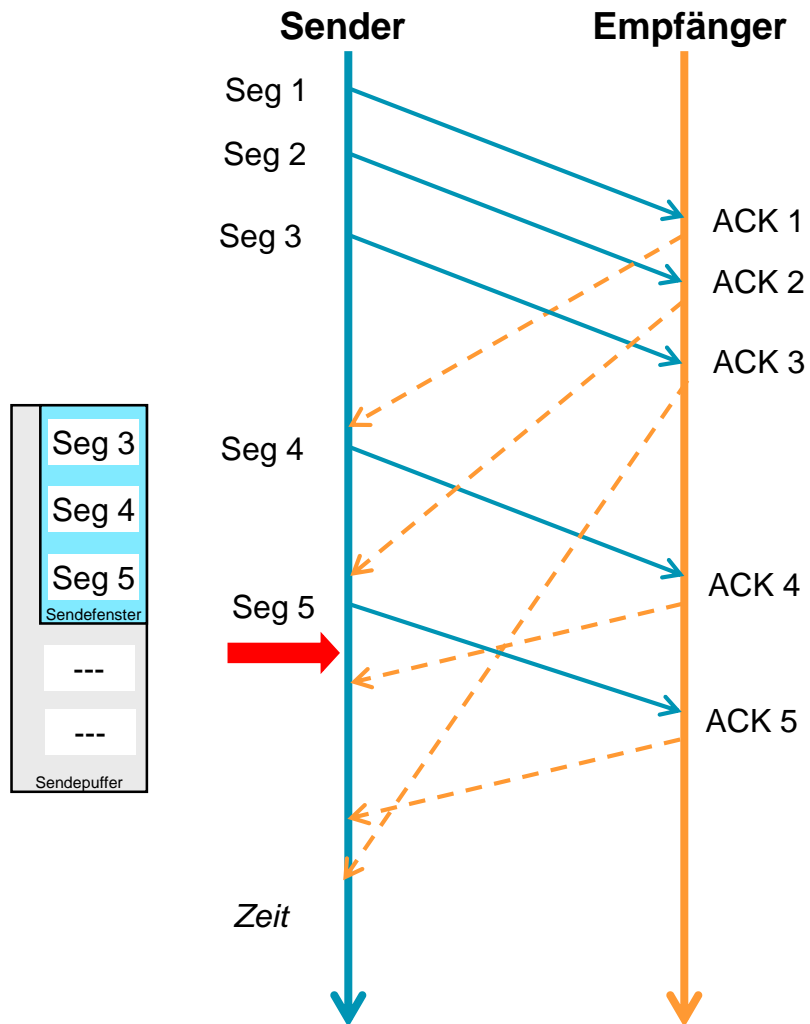
Sliding-Window - detaillierter



- Nun kommt ACK 2
- Was passiert?



Sliding-Window - detaillierter



- Schieben des Fensters
- Versenden von Seg 5

Für Fortgeschrittene:

- Was passiert, wenn ACK 4 und ACK 5 vor ACK 3 kommen?

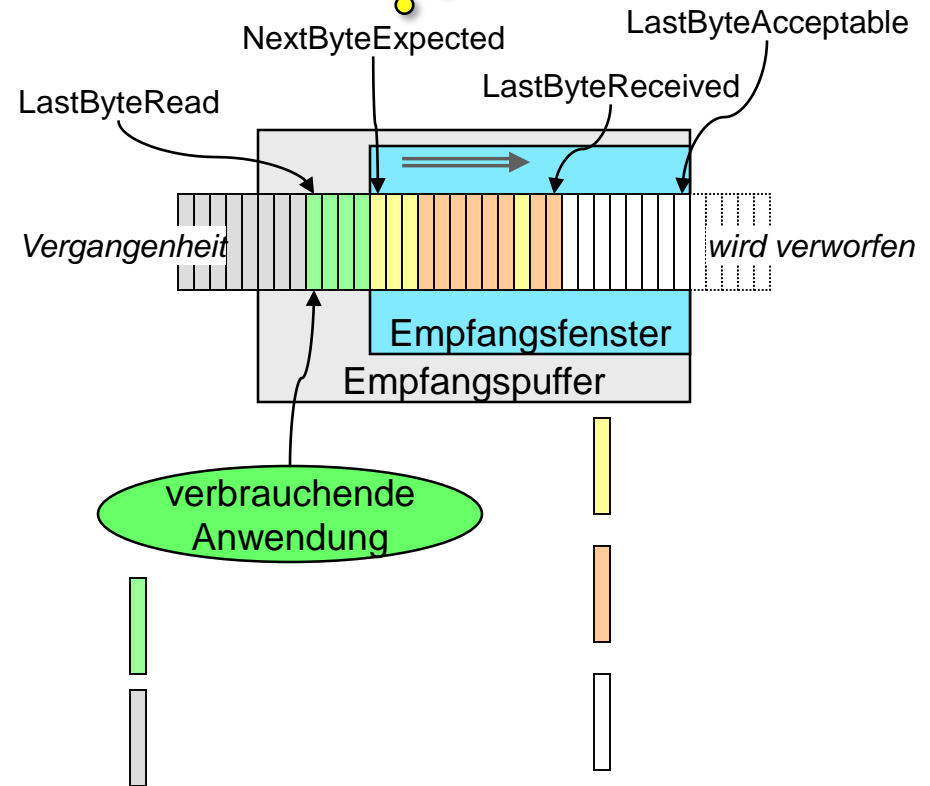
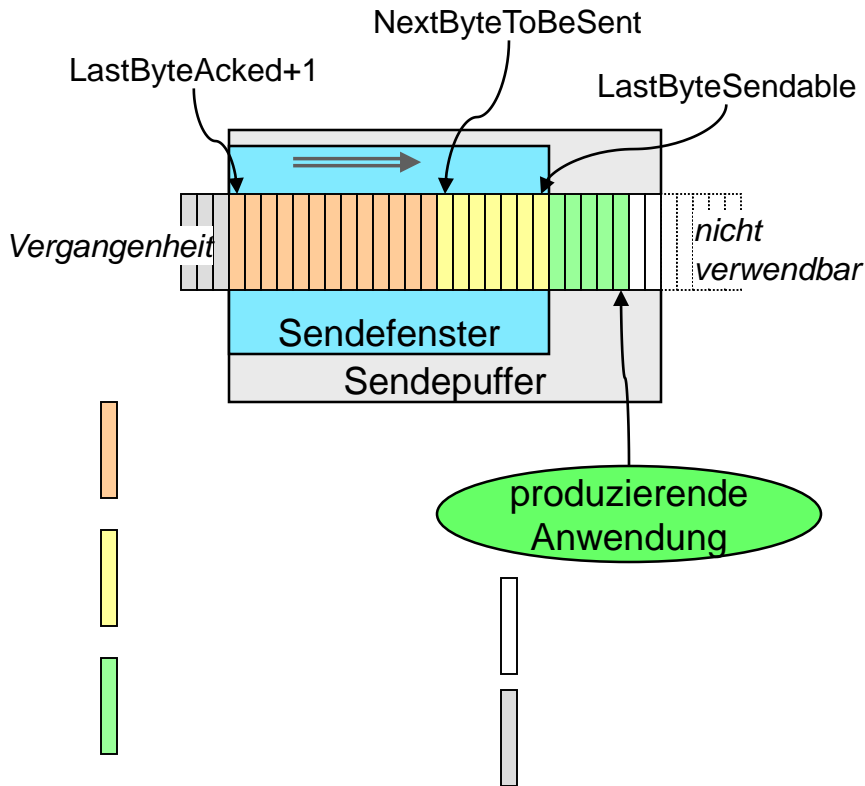
Frage an alle:

- Wie groß soll das Sendefenster sein?
- Wer legt das Sendefenster fest?

TCP-Sliding-Window – Sende- und Empfangspuffer

Spezielle Variante des Sliding-Window-Grundalgorithmus

- dynamische Fluss-Steuerung mittels "Advertised Window"
- Einbeziehung der verarbeitenden Anwendungen





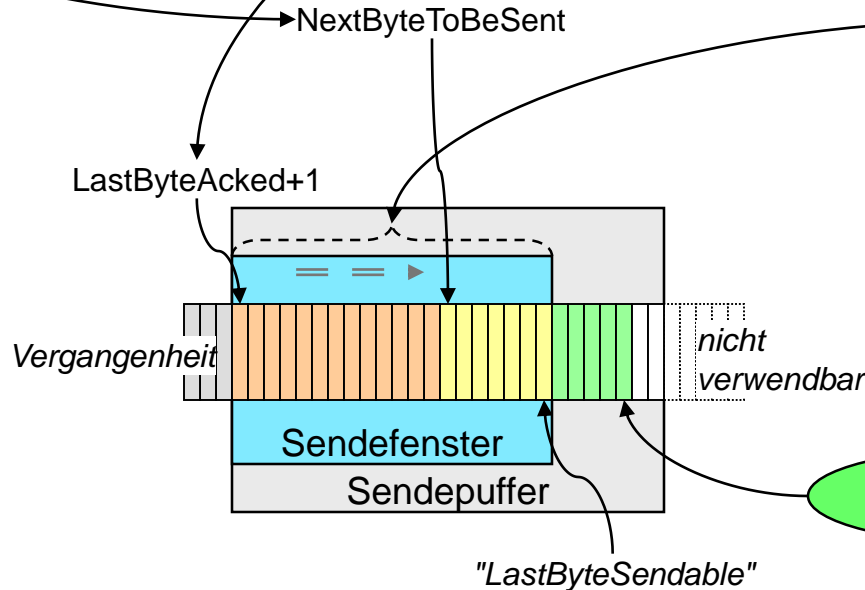
TCP-Flusskontrolle – Segmente auf Senderseite

zu sendendes Segment

empfangenes Segment

Source Port		Destination Port						
Sequence Number								
Acknowledgement Number								
Offset	Reserved	URG	ACK	PSH	RST	SYN	FIN	Advertised Window Size
Checksum		Urgent Pointer						
Options (variable Länge)			Padding					

Source Port		Destination Port						
Sequence Number								
Acknowledgement Number								
Offset	Reserved	URG	ACK	PSH	RST	SYN	FIN	Advertised Window Size
Checksum		Urgent Pointer						
Options (variable Länge)			Padding					



Advertised Window Size:

- Anzahl der Bytes, die der Empfänger bereit ist, zu empfangen
- Formel:

$$\text{AdvertisedWinSize} \leq \text{EmpfPuffer} - (\text{LastByteReceived} - \text{LastByteRead})$$

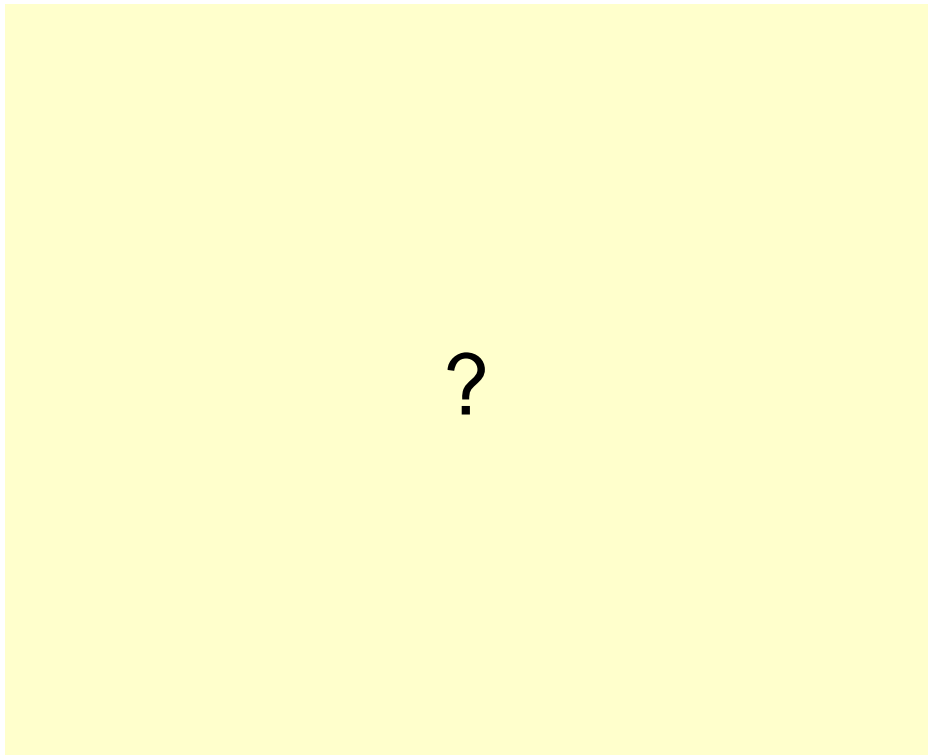
produzierende Anwendung

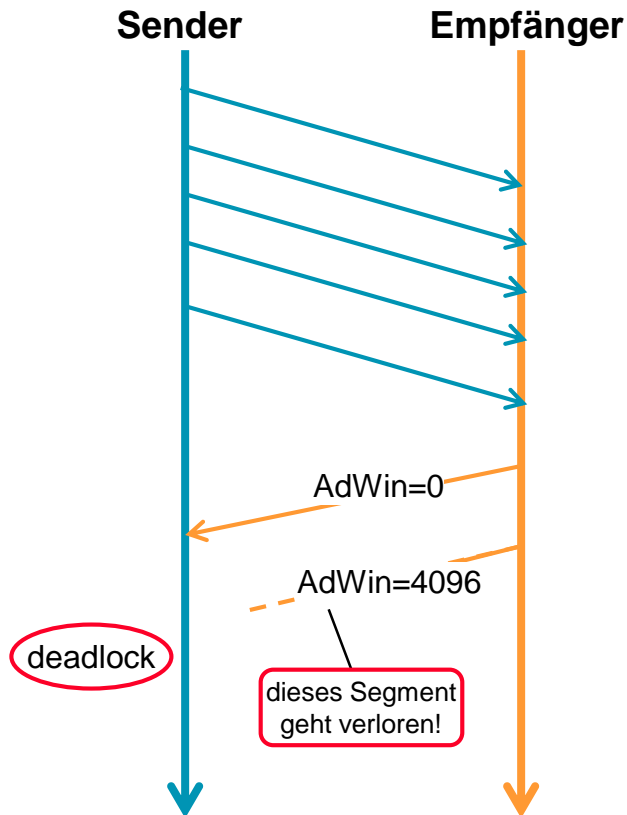
TCP-Flusskontrolle – "Zero advertised window size"

Dargestelltes Problem:

- Wie kommt es zum Deadlock?

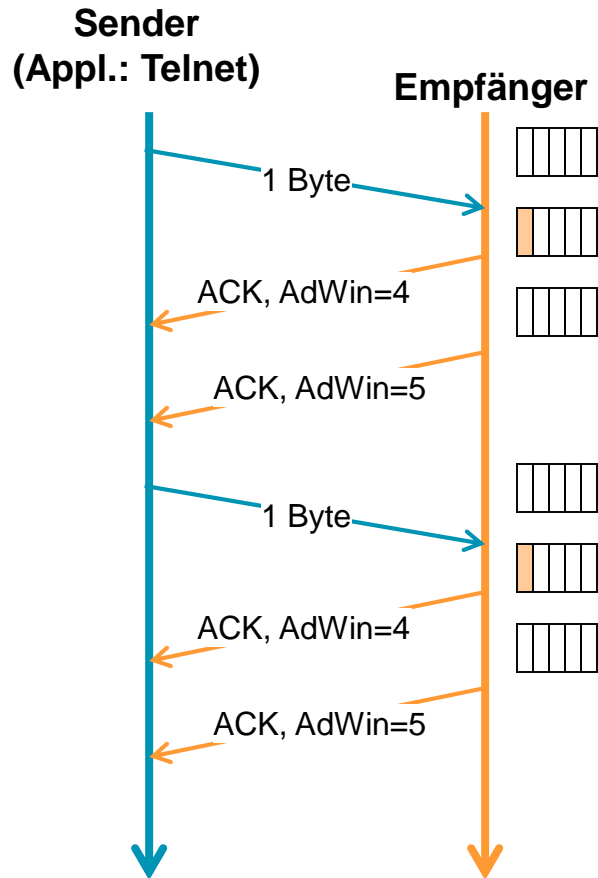
Lösung des Problems:

- 
- ?





TCP-Flusskontrolle – "Small Packet Problem"



dargestelltes Problem:

?

Lösungen:

- Empfängerseitig: delayed Acknowledgement:

?

- Senderseitig: Nagle's Algorithm (RFC 896, 1984):

?

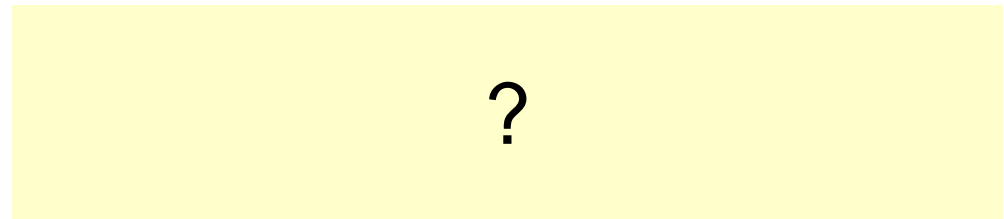
?

Abg. über sockets abschaltbar.



TCP-Flusskontrolle – "Silly Window Syndrome (SWS) RFC 813"

dargestelltes Problem:

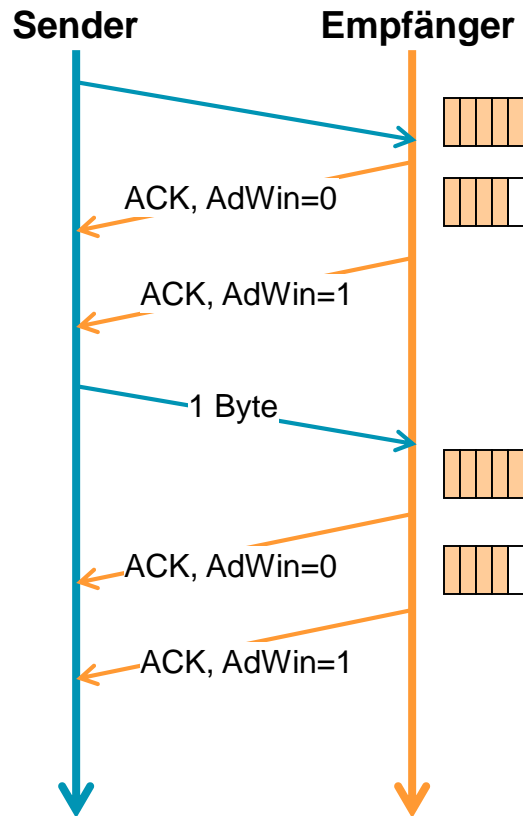
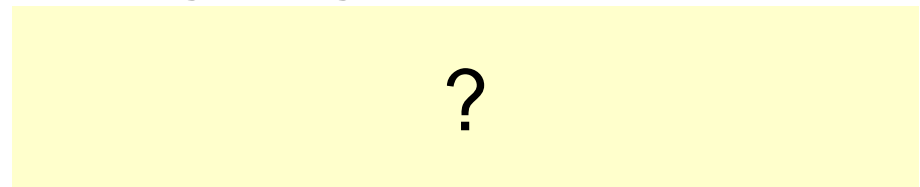


Lösungen:

- **Senderseitig:**



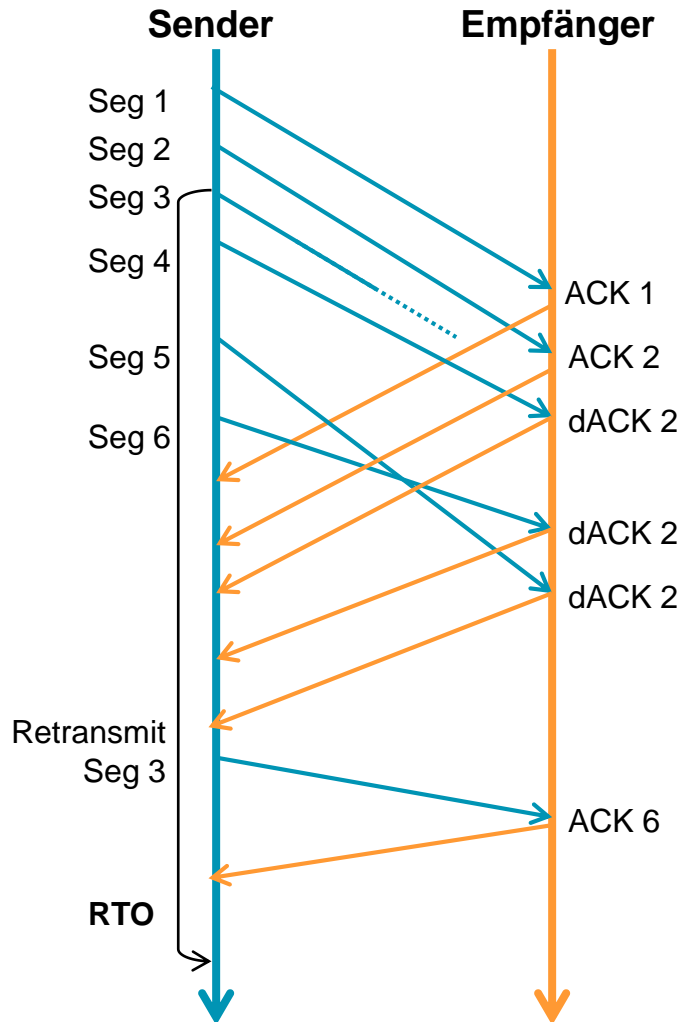
- **Empfängerseitig:**



1 Byte



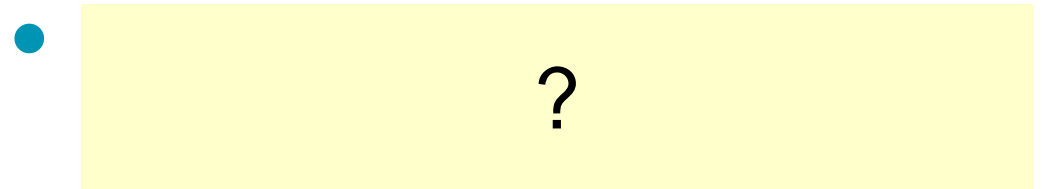
Fast Retransmit (Grundprinzip)



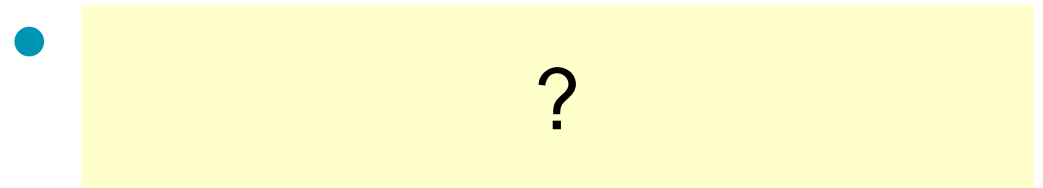
dargestelltes Problem:



Lösung:



Neues Problem:



-> Themenstellung Überlastkontrolle

Anmerkung:

Fast Recovery: Algorithmus, um nach Fast Retransmit Datenfluss zu erhalten

SACK (Selective Acknowledgement, RFC2018): Bestätigung der tatsächlich erhaltenen Segemente



TCP-Acknowledgements (gemäß RFC 1122, 10/89)

Ereignis	Reaktion TCP-Empfänger
Ankunft eines direkt nachfolgenden Segments, keine Lücke, alle vorhergehenden Segmente sind bereits bestätigt.	delayed Acknowledgement: Warte bis zu 200 ms, ob neues Segment kommt, wenn bis dahin keines kommt, muss ein ACK gesendet werden.
Ankunft von direkt nachfolgenden Segmenten, keine Lücke	Senden eines kumulativen ACKs: Bestätigen von mehreren Segmenten mit einem Acknowledgement
Ankunft eines Segments, das ganz oder teilweise eine Lücke füllt (so dass ein Teil des Bytestroms vervollständigt wird).	Immediate Acknowledgement: Sofortiges Senden eines ACKs.
Ankunft eines out-of-order Segments größer als NextByteExpected (es entsteht Lücke).	Senden eines duplicate ACKs: wiederholtes Senden des letzten ACKs (=Beginn der Lücke)