



Motivierende Fragen

- Was versteht man unter Lastkontrolle ?
- Wo können Engpässe in der Kommunikation entstehen ?
- Wie funktioniert die Flusssteuerung ?
- Was passiert bei einer Netzüberlastung ?
- Was versteht man unter Verkehrs-, Stau- und Ratenkontrolle ?
- Was versteht man unter der Dienstgüte (QoS)?



Übersicht

- | | |
|---|--|
| <ol style="list-style-type: none"> 1. Einführung und Motivation <ul style="list-style-type: none"> ▪ Bedeutung, Beispiele 2. Begriffswelt und Standards <ul style="list-style-type: none"> ▪ Dienst, Protokoll, Standardisierung 3. Direktverbindungsnetze <ul style="list-style-type: none"> ▪ Fehlererkennung, Protokolle ▪ Ethernet 4. Vermittlung <ul style="list-style-type: none"> ▪ Vermittlungsprinzipien ▪ Wegwahlverfahren 5. Internet-Protokolle <ul style="list-style-type: none"> ▪ IP, ARP, DHCP, ICMP ▪ Routing-Protokolle 6. Transportprotokolle <ul style="list-style-type: none"> ▪ UDP, TCP 7. Verkehrssteuerung <ul style="list-style-type: none"> ▪ Kriterien, Mechanismen ▪ Verkehrssteuerung im Internet | <ol style="list-style-type: none"> 8. Anwendungsorientierte Protokolle und Mechanismen <ul style="list-style-type: none"> ▪ Netzmanagement ▪ DNS, SMTP, HTTP 9. Verteilte Systeme <ul style="list-style-type: none"> ▪ Middleware ▪ RPC, RMI ▪ Web Services 10. Netzsicherheit <ul style="list-style-type: none"> ▪ Kryptographische Mechanismen und Dienste ▪ Protokolle mit sicheren Diensten: IPSec etc. ▪ Firewalls, Intrusion Detection 11. Nachrichtentechnik <ul style="list-style-type: none"> ▪ Daten, Signal, Medien, Physik 12. Bitübertragungsschicht <ul style="list-style-type: none"> ▪ Codierung ▪ Modems |
|---|--|



Grundlagen: Rechnernetze und Verteilte Systeme

Kapitel 7: Verkehrssteuerung Kriterien, Mechanismen, Verfahren

Prof. Dr.-Ing. Georg Carle
 Lehrstuhl für Netzarchitekturen und Netzdienste
 Technische Universität München
 carle@net.in.tum.de
 http://www.net.in.tum.de



Ziele

- In diesem Kapitel wollen wir vermitteln
 - Lastkontrolle
 - Flusssteuerung
 - Überlastung im Netzzinnern
 - Verkehrskontrolle
 - Staukontrolle
 - Ratenkontrolle
 - Dienstgüte

Kapitelgliederung

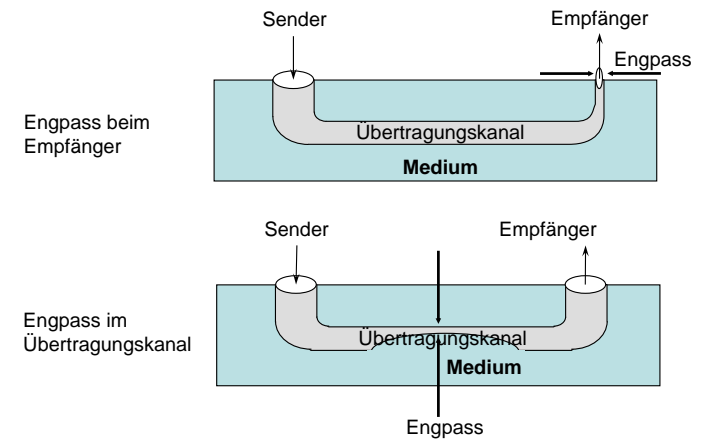
7.1. Lastkontrolle

- 7.1.1. Engpässe in Kommunikation
- 7.1.2. Flusssteuerung
 - 7.1.2.1. Datagramm versus Verbindung
 - 7.1.2.2. Arten von Flusssteuerung
- 7.1.3. Überlastung im Netzinnern
 - 7.1.3.1. Stau- / Verkehrskontrolle
 - 7.1.3.2. Anforderungen
 - 7.1.3.3. Verkehrs- /Staukontrollverfahren
 - 7.1.3.4. TCP: Flusssteuerung / Staukontrolle
 - 7.1.3.5. TCP: Fast Retransmit, Fast Recovery
- 7.1.4. Ratenkontrolle

7.2. Dienstgüte (QoS)

- 7.2.1. Dienstgüteparameter
- 7.2.2. Dienstklassen
- 7.2.3. Dienstgütemechanismen
- 7.2.4. QoS-Architekturen

7.1.1. Engpässe in Kommunikationsnetzen

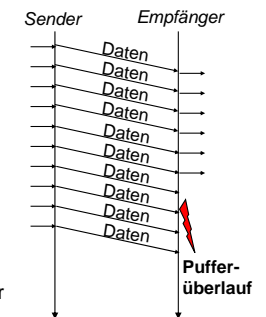


7.1. Lastkontrolle

- An einem Kommunikationsvorgang sind üblicherweise mehrere Systeme beteiligt:
 - das den Vorgang initierende System;
 - das gerufene System;
 - das Netzwerk mit den entsprechenden Zwischensystemen.
- Die auszutauschende Datenmenge muss dabei an aktuelle Eigenschaften der beteiligten Systeme (Netzknotten, Endsysteme) angepasst werden, da es sonst zu unterschiedlichen Problemen kommen kann (siehe die folgenden Folien).
- In der Vorlesung soll als Überbegriff für die Reglementierung der zu sendenden Datenmenge der Begriff **Lastkontrolle** stehen.
- Es sei hier jedoch darauf hingewiesen, dass dieser Begriff (wie auch die weiteren in diesem Thema eingeführten) in der Literatur leider nicht eindeutig definiert und benutzt wird!

Engpass im Empfänger

- **Annahme:**
 - Netz ist kein Engpass, sondern hält mit Sender(n) mit.
- **Ursachen für Engpass im Empfänger:**
 - In einem Rechnernetz kommunizieren Rechner unterschiedlicher Leistungsfähigkeit (langsamer Empfänger).
 - (oder) Empfänger bekommt Pakete von vielen Sendern
- **Problem:**
 - Leistungsfähiger Rechner sendet mit hoher Rate - oder viele Rechner senden an einen Empfänger
 - Empfänger kann diese nicht in entsprechender Geschwindigkeit verarbeiten.
 - Empfangspuffer läuft über. Daten gehen verloren.





Bsp.: Pufferüberlauf bei Punkt-zu-Punkt-Verbindung



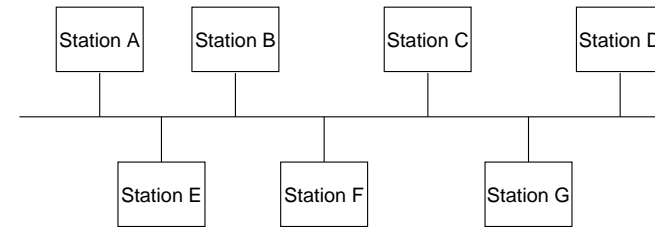
Leistungsfähiger Sender

Leistungsschwacher Empfänger

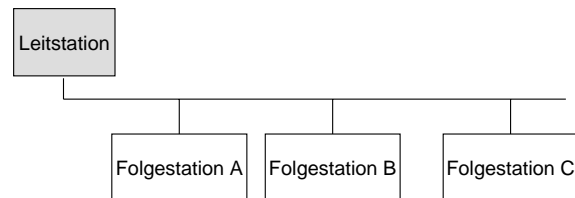


Engpass im Übertragungskanal

- **Beispiel:** Überlastung des geteilten Mediums
- **Annahme:** alle Stationen könnten die eintreffenden Daten verarbeiten
- **Aber:** Die Kapazität des geteilten Mediums reicht nicht aus für die zahlreichen Übertragungswünsche der Stationen
- **Folge:** Daten gehen schon beim Zugriff auf das Medium verloren und kommen erst gar nicht bei den Empfängern an.



Bsp.: Perfekte Lastkontrolle durch Polling-Betrieb



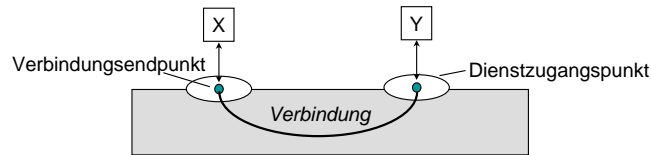
7.1.2. Flusssteuerung

- **Synonyme Begriffe**
 - Flusssteuerung
 - Flussregulierung
 - Flusskontrolle
 - Flow Control
- **Aufgabe**
 - Auf Netzebene ist der **Datenpaketempfänger** vor einem zu großen Zufluss von Paketen eines Paketsenders zu schützen.
- **Ort der Durchführung**
 - Schicht 2 (Sicherheitsschicht): Überlastungsschutz von Übermittlungsabschnitten
 - Schicht 3
 - Schicht 4



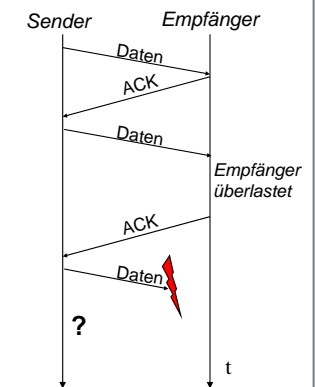
7.1.2.1. Datagramm versus Verbindung

- **Datagramm (verbindungsloser Transportdienst)**
 - Datagramme werden unabhängig voneinander transportiert und laufen ggf. auch auf unterschiedlichen Wegen durchs Netz
 - Keine Kontextinformation innerhalb des Netzes
 - Flusssteuerung muss auf höheren Schichten erfolgen
- **Verbindungen (verbindungsorientierter Transportdienst)**
 - Kontext, etabliert durch Verbindungsaufbau
 - Kontextinformation beinhaltet Adressinformation und zusätzliche Verbindungsidentifikation (z.B. Portnummer bei TCP/UDP), wenn mehreren Verbindungen vom selben Dienstzugangspunkt ausgehen
 - Flusssteuerung durch Anpassung der Sendegeschwindigkeit



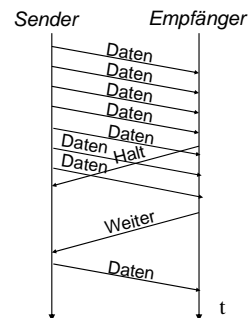
Implizite Flusssteuerung

- **Funktionsweise:**
 - Durch Zurückhalten der Quittung (z.B. ACK/NAK) kann der Sender gebremst werden.
 - Das bedeutet, dass ein Verfahren zur Fehlererkennung für die Flusssteuerung mitbenutzt wird.
- **Problem:**
 - Der Sender kann nicht mehr unterscheiden,
 - ob sein Paket völlig verloren ging, oder
 - ob der Empfänger die Quittung wegen Überlast zurückgehalten hat.
 - ineffizient



7.1.2.2. Arten von Flusssteuerung Flusssteuerung mit Halt-/Weiter-Meldungen

- **Einfachste Methode:**
 - Sender-Empfänger-Flusssteuerung
 - Meldungen:
 - Halt
 - Weiter
 - Kann der Empfänger nicht mehr Schritt halten, schickt er dem Sender eine **Halt**-Meldung.
 - Ist ein Empfang wieder möglich, gibt der Empfänger die **Weiter**-Meldung.
- **Beispiel: Protokoll XON/XOFF**
 - Mit ISO 7-Bit-Alphabetzeichen.
 - XON ist DC₁ (Device Control 1).
 - XOFF ist DC₃ (Device Control 3).
 - Nur auf Vollduplex-Leitungen verwendbar.

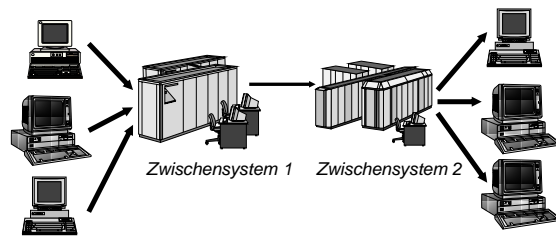


Kreditbasierte Flusssteuerung

- **Funktionsweise:**
 - Der Empfänger räumt dem Sender einen mehrere Transfereinheiten umfassenden Sendekredit ein.
 - Ist der Kredit (ohne neue Kreditgewährung) erschöpft, stellt der Sender den Transfer ein.
 - Dazu ist aber eine verstärkte Fehlerkontrolle, z.B. für den Verlust der neuen Kreditgewährung, erforderlich.
- **Realisierungsmöglichkeiten:**
 - Explizite Kreditgewährung:
 - Empfänger teilt dem Sender explizit den aktuellen Kredit mit.
 - Kreditfenster („Sliding Window“):
 - Mit jedem quittierten Paket wird das Kreditfenster verschoben.

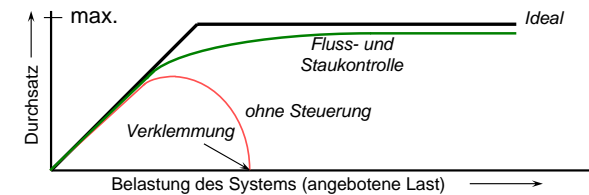
7.1.3. Überlastung im Netzinnern

- **Ausgangspunkt:**
 - Mehrere Kommunikationsvorgänge werden im Netzinneren über dieselben Zwischensysteme abgewickelt.
- **Problem:**
 - Obwohl die Sender für sich keine sehr große Last produzieren, können die Ressourcenanforderungen der einzelnen Kommunikationsvorgänge in einem Zwischensystem zu einer Überlastsituation führen



7.1.3.1. Stau-/Verkehrskontrolle

- **Begriffe**
 - Staukontrolle (Congestion Control)
 - (Über-)Lastkontrolle
 - Verkehrskontrolle (Traffic Control)
- **Problem und Ziel**
 - Im Überlast-/Staubereich sinkt im ungesteuerten Fall der Durchsatz bis zum totalen Netzstillstand, z.B. durch Verklemmungen (deadlock).
 - Erreichen eines „vernünftigen“ Netzverhaltens bei Hochlast-/Überlastbetrieb ist Ziel der Verfahren



Netzüberlastung

- **Synonyme Begriffe**
 - Netzüberlastung
 - Network Congestion
- **Gefahren**
 - Pufferüberläufe in Zwischensystemen bei speichervermittelten Netzen
 - Bei leitungsvermittelten Netzen werden bei Überlast Verbindungswünsche abgewiesen (Besetzt-Fall)
- **Gründe für lokale oder globale Netzüberlastung**
 - Zu starker Zufluss von Transfereinheiten (Paketen)
 - Ausfall von Speicherungsvermittlungen / Teilnetzen mit Verkehrsverlagerung auf andere Netzkomponenten
 - Zunahme transienter Störungen in Netzkomponenten

Verkehrskontrolle versus Staukontrolle

- **Verkehrskontrolle/Staukontrolle:**
 - Mechanismen des Netzwerks, um Stausituationen zu vermeiden und Auswirkungen von Staus zu begrenzen (aus Standard ITU-T I.371)
 - **Verkehrskontrolle:** präventive Mechanismen zur Stauvermeidung
 - **Staukontrolle:** reaktive Mechanismen zur Begrenzung der Auswirkungen in Stausituationen
 - Die Begriffe werden oftmals nicht eindeutig benutzt!
- **Teilaufgaben der Verkehrskontrolle/Staukontrolle:**
 - Zugangskontrolle } Präventiv (Verkehrskontrolle)
 - Nutzungskontrolle } Präventiv (Verkehrskontrolle)
 - Prioritätskontrolle } Reaktiv (Staukontrolle)
 - Reaktive Staukontrolle } Reaktiv (Staukontrolle)



7.1.3.2. Anforderungen an die Verkehrskontrolle/Staukontrolle

- **Überlastsituationen**
 - Netz ist nicht mehr in der Lage, Ressourcen zur vollständigen (garantierten) Durchführung der Dienste zur Verfügung zu stellen.
 - Verkehrskontrolle
 - Mechanismen, um Überlast möglichst zu vermeiden.
 - Staukontrolle
 - Mechanismen, um eine aufgetretene Überlast einzugrenzen und zu beheben.

- **Anforderungen an die Stau- und Verkehrskontrolle**
 - Einfachheit und Robustheit:
 - geringe zusätzliche Belastung des Netzes
 - Effektivität:
 - eine schnelle und wirkungsvolle Reaktion wird gefordert.
 - Fairness:
 - garantierte Dienste sollten eingehalten werden;
 - sich korrekt verhaltende Benutzer sollen von den Entscheidungen nicht betroffen sein;
 - jeder Benutzer soll angemessen berücksichtigt werden



Verkehrskontrolle: Pufferreservierung und Verwerfen von Paketen

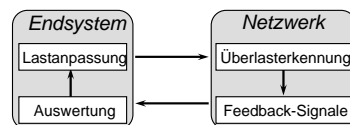
- **Pufferreservierung**
 - Geeignete Verwaltung von Puffern in den einzelnen Speichervermittlungsstellen (auch in den Endstellen), z.B. durch rechtzeitige Allokation.
 - Eine Pufferreservierung kann z.B. bei der Verbindungsaufbauprozedur für eine virtuelle Verbindung erfolgen und über die ganze Verbindungsdauer erhalten bleiben.

- **Wegwerfen von Paketen**
 - In Hochlast-Situationen werden Pakete verworfen, die danach nochmals gesendet werden müssen
 - Umsetzung im Internet:
 - Active Queue Management, RFC2309
 - RED (Random Early Detection), RFC 2481, <http://www.aciri.org/floyd/red.html>



7.1.3.3. Verkehrs-/Staukontrollverfahren

- | | |
|--|---|
| <ul style="list-style-type: none"> □ Im Endsystem <ul style="list-style-type: none"> ▪ Lokal vorliegende Informationen werden ausgewertet (wie z.B. Warteschlangenlänge, Anzahl negativer Quittungen, ...) ▪ Die momentane Situation wird bewertet und durch lokale Aktionen zu verbessern versucht. ▪ Dies betrifft <ul style="list-style-type: none"> • die weitere Verbindungsannahme, • das Verwerfen von ankommenden Paketen, • die Wegewahl, etc. ▪ Ungenauigkeiten durch rein lokale Sichtweise. | <ul style="list-style-type: none"> □ Im Netzwerk <ul style="list-style-type: none"> ▪ Im Netzwerk verteilt arbeiten Mechanismen zur Überwachung der aktuellen Situation. ▪ Bei Entdeckung einer Stausituation werden die relevanten Informationen an die entsprechende Stelle im Netz weitergegeben (Feedback). ▪ Dort muss die Arbeitsweise dann an eine mögliche Stausituation angepasst werden. ▪ Können selbst zum Stau beitragen! |
|--|---|



Verkehrskontrolle: Paketzahlbegrenzung und Rückstau

- **Begrenzung der Paketzahl im Netz**
 - Es existieren „Berechtigungspakete“.
 - Diese werden vom Sender benötigt, um ein eigenes Paket an das Netz zu geben.
 - Ein Empfänger erzeugt bei Entgegennahme eines Pakets ein neues Berechtigungspaket.

- **Rückstauemchanismen**
 - Sie regulieren den Zufluss von Paketen und halten diesen gegebenenfalls an.
 - Besonders an den Eintrittspunkten in das Netz von Bedeutung, z.B. unter Nutzung von Maßnahmen der Flusssteuerung.
 - Im Prinzip ist die Flusssteuerung auf die Abstimmung von Arbeitsgeschwindigkeiten der sendenden und empfangenden Endeinrichtungen ausgerichtet.
 - Sie ist daher eigentlich nicht für die Lösung von netzinternen Überlastproblemen gedacht.



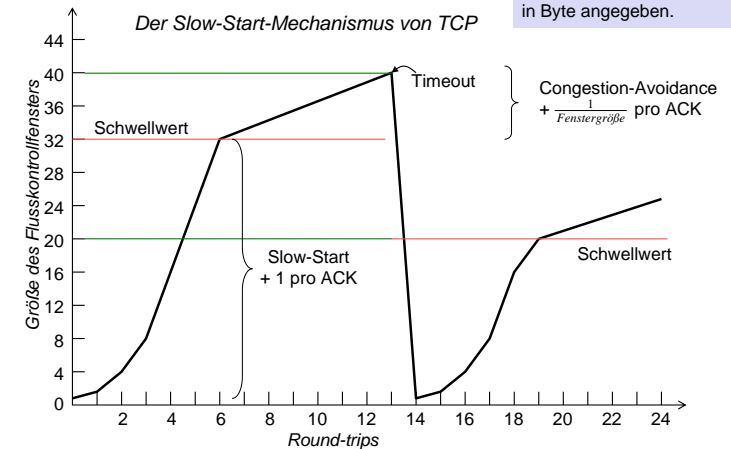
Staukontrolle: Reaktion auf Paketverlust

- **Ziel:**
 - Fehlersituationen im Netz auf Grund von Überlastung, Ausfällen etc. entgegenwirken
- **Schema:**
 - Reaktive Verfahren (z.B. bei TCP) arbeiten daher gemäß dem folgenden allgemeinen Schema:
 - Wird ein Paketverlust erkannt, setze das Flusskontrollfenster auf 1.
 - Wird z.B. durch den Ablauf eines Zeitgebers oder den Empfang eines entsprechenden Kontrollpakets des Empfängers erkannt.
 - Erhöhe das Fenster im Zuge erfolgreich übertragener Pakete allmählich, um wieder den maximalen Durchsatz erzielen zu können.
 - Mitunter werden zwei Phasen unterschieden (z.B. Slow-Start bei TCP)
 - Bis zu einem gewissen Schwellwert wird die Fenstergröße verdoppelt.
 - Ab diesem Wert wird die Fenstergröße um 1 vergrößert.
 - Der Schwellwert kann aufgrund erfahrener Paketverluste angepasst werden.
- **Allgemein:**
 - Prinzipien wie AIMD (Additive Increase Multiple Decrease) führen zu einem stabilen Verhalten. AIMD:
 - Bei erfolgreicher Übertragung Fenstergröße additiv (z.B. +1) vergrößern.
 - Bei Misserfolg Fenstergröße multiplikativ (z.B. $\cdot 0,5$) verkleinern.



Beispielablauf der Staukontrolle

Der Einfachheit halber stellen wir hier die Fenstergröße in Paketen dar. Tatsächlich wird sie bei TCP in Byte angegeben.



7.1.3.4. TCP: Flusststeuerung / Staukontrolle

- **Flusststeuerung** bei TCP:
 - regelt Datenfluss zwischen Endsystemen durch **kreditbasiertes Verfahren**
 - ACK-Feld im Paketkopf bestätigt alle niedrigeren Bytessequenznummern
 - Window-Feld gibt an, wie viele Bytes der Empfänger noch akzeptiert
- **Staukontrolle** bei TCP:
 - befasst sich mit Stausituationen in den Zwischensystemen
 - Problem: „congestion collapse“
 - Stau in Zwischensystemen führt oftmals dazu, dass Transportprotokolle nach einem Timeout Pakete wiederholen. → Die Stausituation wird verstärkt!
 - Lösung: „slow start“- und „multiplicative decrease“-Mechanismen
 - Bei Datenverlust reduziert TCP den Schwellwert, bis zu dem eine Steigerung der Senderate möglich ist, auf die Hälfte des aktuellen Fensterwerts. (multiplicative decrease).
 - Nach einer Stauperiode wird die Fenstergröße um ein Datenpaket erhöht und weiterhin nach jeder empfangenen Quittung (slow start)
 - Der „slow start“ Mechanismus verhindert, dass direkt nach einem Stau zu hoher Verkehr auftritt



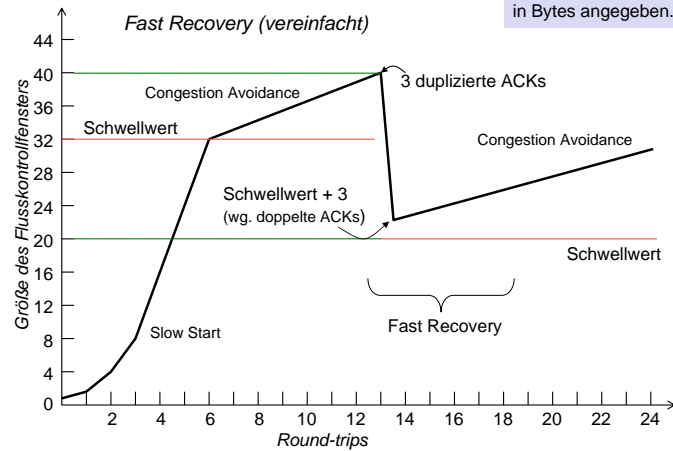
7.1.3.5. TCP: Fast Retransmit, Fast Recovery

- **Fast Retransmit**
 - 3 duplizierte ACKs (also insg. 4) für Segment n
 - Fehler erkannt ohne auf Timer zu warten
 - erneute Übertragung (Retransmit) von Segment n
 - sonst wie bei Timeout
- **Fast Recovery**
 - Fehlererkennung wie bei Fast Retransmit
 - kein Slow-Start!
 - Geht direkt über in Congestion Avoidance
 - Schwellwert = $0.5 \cdot \text{CongestionWindow}$
 - $\text{CongestionWindow} = \text{Schwellwert} + 3$
 - Bei ACK mit neuer Sequenznummer (also nicht-dupliziertes ACK)
 - $\text{CongestionWindow} = \text{Schwellwert}$
 - Übergang in Congestion Avoidance

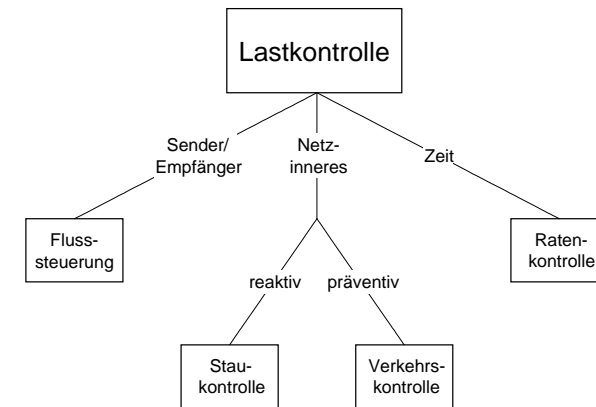


Beispielablauf der Staukontrolle

Der Einfachheit halber stellen wir hier die Fenstergröße in Paketen dar. Eigentlich wird sie bei TCP in Bytes angegeben.



Zusammenfassung



7.1.4. Ratenkontrolle

- Um die Zeit für rücklaufende Quittungen einzusparen und dennoch dem Sender nicht zu viele Freiheiten bei der Lasterzeugung zu lassen, wurde die Ratenkontrolle eingeführt:
 - Der Sender verhandelt mit dem Netz und dem Empfänger eine bestimmte Rate, d.h. eine Datenmenge pro Zeitintervall, die er schicken darf.
 - Er darf während des Zeitintervalls beliebig Daten senden, solange er die maximale Datenmenge nicht überschreitet.
 - Nach Ablauf des Zeitintervalls kann der Sender wieder über die maximale Datenmenge verfügen.



7.2. Dienstgüte - Quality of Service (QoS)

- „**Quality of Service** is the collective effect of service performance, which determine the degree of satisfaction of a user of the service“
- ITU-T Recommendation E.800
 - Von der Zufriedenheit des Benutzers hängt die Güte des Dienstes (QoS) ab - jedoch fehlt die technische Überprüfbarkeit
 - Notwendig:
 - Dienstgüteparameter: beschreiben qualitative Eigenschaften eines Dienstes
 - Dienstklassen: beschreiben den Grad der Garantien
 - Dienstverträge: legen die zu garantierenden QoS-Parameter, deren Werte und den Grad der Einhaltung (Dienstklasse) fest.
 - Dienstgütemechanismen: Maßnahmen zur Einhaltung von Dienstverträgen
 - Management der Dienste: Verwaltung und Reservierung von Ressourcen



7.2.1. Dienstgüteparameter (QoS-Parameter)

- **Leistungsorientierte Dienstgüteparameter:**
 - Verzögerung
 - Ende-zu-Ende-Verzögerung (Delay)
 - Schwankung der Ende-zu-Ende-Verzögerung (Jitter)
 - Durchsatz:
 - min./mittl./max. Durchsatz (in [bit/s] oder [Pakete/s])
 - max. Länge [Pakete] oder Dauer [s] von Lastspitzen (Bursts)
- **Zuverlässigkeitsorientierte Dienstgüteparameter:**
 - Medienabhängige Fehlerraten:
 - z.B.: Bitfehlerrate des Übertragungsmediums
 - Ressourcenabhängige Fehlerraten:
 - z.B.: Paketverlustraten in den Warteschlangen der Zwischensysteme
- **Funktionale Dienstgüteparameter:**
 - Sicherheit
 - Gruppenkommunikation (Multicast)



7.2.2. Dienstklassen (QoS-Klassen)

- **Deterministische Klasse:**
 - vorgegebene Schranken der QoS-Parameter werden exakt eingehalten
 - Ressourcen stehen einem Nutzer exklusiv zur Verfügung
 - keine Konflikte möglich, aber „Besetztfall“ (keine Ressourcen mehr übrig)
- **Statistische Klasse:**
 - vorgegebene Schranken müssen mit einer gewissen Wahrscheinlichkeit eingehalten werden
z.B.: die Ende-zu-Ende-Verzögerung muss für 95% der Pakete unter 100ms liegen.
 - Ressourcen werden bis zu einem gewissen Grad überbelegt
 - Konflikte möglich (je höher die Wahrscheinlichkeit der Garantie, desto geringer sind Ressourcenkonflikte)
- **„Best Effort“-Klasse („so gut es geht“):**
 - es werden keinerlei Garantien für Dienstgüteparameter gemacht
 - keine explizite Ressourcenreservierung für einzelne Verbindungen



Ressourcen

- **Ressourcenarten:**
 - Netzwerkressourcen
 - Übertragungskapazität (Bandbreite, Kanäle)
 - Übertragungszeit
 - (Zwischen- bzw. End-) Systemressourcen
 - Pufferspeicher
 - Rechenzeit
- **Reservierung von Ressourcen:**
 - durch Ressourcenmanagement
 - Verwaltung von Ressourcen (Belegung, Überwachung, Freigabe, ...)
 - Verteilung der Reservierungsnachrichten durch Reservierungsprotokolle
 - RSVP (Resource ReserVation Protocol)
 - NSIS (Next Steps In Signalling protocol)



7.2.3. Dienstgütemechanismen

- **Dienstgütemechanismen** dienen der Einhaltung und Überwachung von Dienstgüteparametern.
- Man unterscheidet deshalb:
 - Verkehrsmeter (Messen des Verkehrs)
 - Token Bucket
 - Average Rate Meter
 - Verkehrsformer (Beeinflussen/Formen des Verkehrs)
 - Verkehrsglätter (Traffic Shaper)
 - Token Bucket → gibt durchschnittliche Rate und Burst-Größe vor
 - Leaky Bucket → gibt maximale Rate vor
 - Verwerfer (Dropper)
 - Degradierer (Verringern der Dienstgüte oder -klasse)
 - Ändern des Warteschlangen- oder Prozess-Schedulings



7.2.4. QoS-Architekturen

- Eine **QoS-Architektur** definiert:
 - QoS-Parameter
 - Ressourcen
 - Ressourcenreservierung
 - Ressourcenmanagement
 - Dienstklassen
 - Dienstgütemechanismen

- Beispiele für QoS-Architekturen:
 - ATM
 - Integrated Services
 - Differentiated Services