

Fragen Datenkommunikation

Fragenbereich 1 (Transmission Principles, Protocol Principles)

- 1) *Was versteht man unter paralleler und serieller Übertragungstechnik? Wo wird die jeweilige Technik eingesetzt? Erklären Sie den Begriff Bitsynchronisation bei serieller Übertragungstechnik. Beschreiben Sie das Grundprinzip der asynchronen und synchronen bitseriellen Übertragungstechnik.*

Einfach gesprochen benutzt die serielle Übertragungstechnik nur eine Leitung zur Datenübertragung während die parallele Übertragungstechnik mehrere Leitungen zur Datenübertragung benutzt. Bei dieser Übertragungsmethode werden die Daten, Adressen und Kontrolldaten sowie der Takt (Clock) über mehrere Leitungen gesendet. Bei der seriellen Übertragungsmethode muss alles über eine Leitung erfolgen weshalb man sich gewisse Techniken überlegen musste um eine Datenübertragung auf serieller Basis gewährleisten zu können. Der Empfänger muss beispielsweise erkennen können welche Bits wofür stehen – außerdem muss der Empfangstakt dem Sendetakt angepasst werden. (Bit Synchronisation)

Bit Synchronisation: Man bedient sich so genannter Signalfanken – das sind Änderungen des Signals von 0 → 1 oder 1 → 0. Nur bei solchen Änderungen kann der Empfänger erkennen, dass es sich um 2 Bits handelt. Ohne Takt wäre z.B. nicht erkennbar ob es sich bei zwei 1ern hintereinander wirklich um zwei Bits oder nur um ein „langes“ Bit handelt. Dieser Takt ermöglicht es dem Empfänger nun den Bitstrom zu teilen und dann individuell zu entscheiden ob es sich um eine 0 oder 1 handelt. Die Überprüfung sollte in der Bitmitte erfolgen, da die Wahrscheinlichkeit hier am größten ist die richtige Entscheidung zu treffen.

Die Parallele Übertragungstechnik wird nur in kurzen Strecken eingesetzt (z.B. LPT-Port) in WAN's, Internet wäre dies nicht bzw. nur mit sehr hohem Kostenaufwand realisierbar und dementsprechend nicht wirtschaftlich.

Grundprinzip der asynchronen und synchronen bitseriellen Übertragungstechnik:

Asynchron: Bei der asynchronen Übertragung wird für jeden Datenblock (8 bits) neu synchronisiert. Dies erfolgt mittels 1 start bit und 2 stop bits (Technik: start-bit: Änderung von 1 auf 0, Stopp-bits: zwei 1er bits) Es muss sichergestellt werden, dass die folgenden Start-Bits wieder als solche erkannt werden. Zwischen den Datenblöcken gibt es ein variables Intervall. Diese Übertragungsmethode ist ineffizient, da für 8 bits 3 weitere bits für die Synchronisation benötigt werden.

Synchron: Bei synchronen bitseriellen Übertragungstechnik werden die Signalfanken (um den Takt zu generieren) aus Änderungen des Signallevels gewonnen. Es werden somit nur noch zu Beginn Synchronisierungs-Bits benötigt. Diese Übertragungstechnik ist heutzutage die wichtigste. Die PLL Technik wird dazu benötigt um den Empfangstakt einzufrieren wenn zwischen mehreren Bits keine Änderung des Signals auftritt (z.B. 1 1 1). Dennoch benötigt auch die PLL Technik „hin und wieder“ eine Signalfanke, was durch den simplen NRZ Code nicht gewährleistet werden kann. Aus diesem Grund haben sich zahlreiche Techniken entwickelt um fortlaufende Signalfanken zu gewährleisten (Berühmt: Manchester-Code)

- 2) *Geben Sie die Codierungsvorschrift für den Manchester-Code an. Vergleichen Sie die Eigenschaften dieses Codes bezüglich Bandbreite, Gleichanteil (DC = direct current) mit dem NRZ-Code. In welchen Netzwerken wird dieser verwendet?*

Manchester Code: Der Manchester Code arbeitet mit der prinzipiell einfachen Methode, dass jedes Bit in zwei „Halb-Bits“ unterteilt wird. Somit stellt jedes Bit für sich eine Signalfanke bereit. Das erste Halb-Bit ist das Komplementär des Datenbits und das zweite Halb-Bit ist der Datenbit selbst. → Eine Änderung von 1 nach 0 beschreibt eine logische 0 sowie umgekehrt.

Differential Manchester Code: diesem Verfahren wird die logische 0 durch Signaländerung am Beginn und in der Mitte des „Ganz-Bits“ (2 Halb-Bits) erzeugt. Die logische 1 erkennt man dadurch, dass sich nur in der Mitte des „Ganz-Bits“ das Signal ändert. (Keine Änderung des Signals → Code Violation). Der große Nachteil dieser Übertragungstechnik besteht natürlich darin, dass die doppelte Bandbreite benötigt wird. Im lokalen Netzwerk ist dies natürlich nicht von Bedeutung – Ob mein Netzwerk für 10 mbit oder 20 mbit (effektiv 10 mbit durch Manchester Codierung) ausgelegt ist spielt kostenmäßig eine untergeordnete Rolle. In WAN's oder GAN's ist dies natürlich von großer Bedeutung, da doppelte Bandbreite dort einen erheblichen Kostenaufwand bedeutet.

NRZ-Code: Der NRZ-Code verwendet folgende Methode: Die logische 0 wird durch Signaländerung definiert. Die logische 1 definiert sich durch Nicht-Änderung des Signals. Bit-Stuffing verhindert eine zu große Anzahl von 1en hintereinander.

AMI Code: Als eine Weiterentwicklung des NRZ Codes könnte man den AMI Code bezeichnen. Bei dieser Code-Variante gibt es nicht nur 2 sondern 3 Signalarten (-, 0, +). Tritt die 1 auf kommt entweder ein negatives oder positives Signal beim Empfänger an. Sind mehrere 1er hintereinander so wechselt sich die Polarität. Ein Scrambler (Zerhacker) sorgt außerdem dafür, dass eine 0-Reihen auftreten.

3) *Geben Sie die Codierungsvorschrift für den HDB3-Code an. Vergleichen Sie die Eigenschaften dieses Codes bezüglich Bandbreite, Gleichanteil (DC = direct current) mit dem NRZ-Code. In welchen Netzwerken wird dieser verwendet (LAN oder WAN)?*

HDB3-Code: Der HDB3-Code benutzt folgende Codierungsmethode: Logische 1er werden mit Hilfe eines alternierenden positiven - negativen Pulses dargestellt. Eine logische 0 generiert keinen Puls. Die Ausnahme besteht jedoch bei längerer Folge von 0en. Kommt es zu dem Fall, dass vier 0en hintereinander gesendet werden müssen, so wird anstatt der vierten 0 ein oder zwei Pulse gesendet (A- und V-Bits). Diese werden durch den Code ebenfalls als 0 interpretiert. Die Bandbreitenbeanspruchung ist gleich dem NRZ-Code. Der HDB3 Code besitzt keinen oder nur konstanten Gleichanteil.

NRZ-Code: Der NRZ-Code verwendet folgende Methode: Die logische 0 wird durch Signaländerung definiert. Die logische 1 definiert sich durch Nicht-Änderung des Signals. Bit-Stuffing verhindert eine zu große Anzahl von 1en hintereinander.

AMI-Code: Als eine Weiterentwicklung des NRZ Codes könnte man den AMI Code bezeichnen. Bei dieser Code-Variante gibt es nicht nur 2 sondern 3 Signalarten (-, 0, +). Tritt die 1 auf kommt entweder ein negatives oder positives Signal beim Empfänger an. Sind mehrere 1er hintereinander so wechselt sich die Polarität. Ein Scrambler (Zerhacker) sorgt außerdem dafür, dass keine 0-Reihen auftreten.

Beide Codes können in WAN's verwendet werden.

4) *Wozu dienen Übertragungsrahmen (Framing)? Wie schaut der prinzipielle Aufbau eines Übertragungsrahmens aus? Gehen Sie kurz auf die Bedeutung der einzelnen Felder ein. Was versteht man unter Rahmensynchronisation (Framesynchronization)? Wie erfolgt Rahmensicherung (Frame Checking) und Fehlererkennung (Error Detection) prinzipiell bei serieller Übertragungstechnik?*

Grundlagen: Informationen zwischen Systemen wird in Datenblöcken oder Informations-Frames ausgetauscht. Daher ist es wichtig zu erkennen wo ein Datenblock beginnt und wo dieser endet (Frame Synchronisation). Da es zu Übertragungsfehlern kommen kann müssen Fehler entdeckt werden (Frame Checking, Error Detection).

Ein Frame besteht aus: SYNC - Bit Synchronisation
 SD - Frame Synchronisation
 Control Information
 Data
 FCS – Checksum
 ED - Frame Trailer

Sync: Es wird eine Bitfolge gesendet um die Takt zu synchronisieren (zb 01010101...)

SD: Um das Frame als Frame kennzuzeichnen wird eine Bitfolge gesendet die im normalen Datenstrom so nicht vorkommt. (Üblicherweise 01111110). Muss sich die Applikation darum kümmern, dass solche Bitfolgen nicht gesendet werden ist der Datenstrom nicht datentransparent. Daher gibt es einige Methoden um dies zu verhindern (zb. Byte Stuffing, Bit Stuffing, Code Violations, Byte count techniques, Idle line before SD and idle line as ED)

Bit Stuffing: Als SD und ED werden 01111110 gesendet. Kommt diese Bitfolge im normalen Datenstrom vor inkludiert der Sender automatisch eine 0 nach fünf 1en. Der Empfänger entfernt diese 0 automatisch und der ursprüngliche Datenstrom ist wiederhergestellt. Somit ist die einzige Möglichkeit in der 01111110 noch vorkommt im ED und zeigt somit das richtige Ende des Frames an.

Byte Stuffing: Es werden sogenannte Control Characters des ASCII Codes als SYNC, SD, ED verwendet. Diese dürften im normalen Datenstrom nicht vorkommen. Durch Byte Stuffing ist es jedoch dennoch möglich, dass diese vorkommen. In dem Fall, dass z.B. ein „DLE“ vorkommen muss, wird es vom Sender einfach verdoppelt. Der Empfänger löscht das zweite „DLE“ aus dem Datenstrom heraus und er ursprüngliche Datenstrom ist wiederhergestellt.

Control-Field: Das Control-Feld wird für die Implementierung von Protokoll-Prozeduren benutzt. (z.B. Ack, Nack, Connect, Disconnect, Reset, Sequenc Number für die Identifikation des Frames, Addressinformationen, Framelänge, ...)

FCS: Hier wird eine Checksumme (HASH Wert) des Frames oder nur der Daten mitgeschickt anhand der Empfänger errechnen kann ob die Daten korrekt oder fehlerhaft angekommen sind. Die Checksumme wird anhand von verschiedenen Mathematischen Prozeduren ausgerechnet.

Frame Checking und Error Detection: Es wurden 2 Basisstrategien entwickelt.

1) Forward Error Control: Man schickt einfach so viele redundante Informationen mit jedem Datenblock mit, damit der Empfänger im Fall eines Verlustes diesen wiederherstellen kann. Wird für sehr große Entfernungen verwendet (Satelliten, Mars, ...)

2) Feedback Error Control: Man schickt nur soviel zusätzliche Information mit einem Datenstrom mit damit der Empfänger erkennen kann, dass in dem Datenblock ein Fehler da ist. Danach kann der Empfänger den Datenblock erneut anfordern. Dies wird mit Hilfe einer Frame Check Sequenz gemacht. Mittels mathematischer Methoden (Paritäts-Bit, Summe aller Daten modulo 2, CRC [Polynom Code]) – Eine 100%ige Sicherheit kann nicht gewährleistet werden - wird eine Checksumme mitgeschickt. Der Empfänger errechnet ebenfalls eine solche Checksumme und vergleicht diese. Stimmt sie nicht überein ist der Datenblock fehlerhaft, wird gelöscht und anschließend erneut angefordert.

5) Was versteht man unter Datentransparenz und wie wird diese erreicht? Erklären Sie das an Hand der bitorientierten (bitoriented) Methode.

Unter Datentransparenz versteht man Techniken um das Auftreten von Bit-Blöcken welche z.B. für SD oder ED (Im Allgemeinen „Nicht-Daten“) vorgesehen sind im normalen Datenstrom zu vermeiden. Die Anwendung muss sich daher nicht um das Nicht-Auftreten solcher Bitfolgen kümmern.

Daher haben sich verschiedene Techniken entwickelt.

- Byte Stuffing:** Wird bei der Zeichenorientierten Methode angewandt. (zB PPP asynchron)
- Bit Stuffing:** Wird bei der Bitorientierten Methode angewandt (PPP synchron)
- Code Violations:** z.B. Token Ring
- Byte count technique:** DDCMP
- Idle Line:** Idle Line vor SD und Idle Line nach ED (z.B. Ethernet)

Bit-orientierte Methode: (Bit Stuffing) Als SD und ED werden 01111110 gesendet. Sollte die 01111110 nun im normalen Datenstrom vorkommen würde sie das Ende des Stroms anzeigen. Um dies zu vermeiden inkludiert der Sender, falls diese Bit-Folge im Datenstrom vorkommen muss, automatisch eine 0 nach der fünften 1. Der Empfänger entfernt diese 0 automatisch und der ursprüngliche Datenstrom ist wiederhergestellt. Somit ist die einzige Möglichkeit in der 01111110 noch vorkommt im ED und zeigt somit das richtige Ende des Frames an.

6) *Was versteht man unter Datentransparenz und wie wird diese erreicht? Erklären Sie das an Hand der zeichenorientierten (character-oriented) Methode.*

Unter Datentransparenz versteht man Techniken um das Auftreten von Bit-Blöcken welche z.B. für SD oder ED (Im Allgemeinen „Nicht-Daten“) vorgesehen sind im normalen Datenstrom zu vermeiden. Die Anwendung muss sich daher nicht um das Nicht-Auftreten solcher Bitfolgen kümmern.

Daher haben sich verschiedene Techniken entwickelt.

- Byte Stuffing:** Wird bei der Zeichenorientierten Methode angewandt. (zb PPP asynchron)
- Bit Stuffing:** Wird bei der Bitorientierten Methode angewandt (PPP synchron)
- Code Violations:** z.B. Token Ring
- Byte count technique:** DDCMP
- Idle Line:** Idle Line vor SD und Idle Line nach ED (z.B. Ethernet)

Character-Oriented Methode: (Byte Stuffing): Für SYNC, SD, ED werden spezielle Zeichen (Control Characters) aus der ASCII Tabelle verwendet. Diese kommen im normalen Zeichensatz an sich nicht vor. (SOH: Start of Header, STX: Start of Text, ETX: End of Text). Diese Kontrollzeichen werden allerdings nur anerkannt wenn vor ihnen das Kontrollzeichen „DLE“ steht. Somit können diese Zeichen im normalen Datenstrom übertragen werden, da sie dementsprechend nicht als Kontroll-Zeichen interpretiert werden, da das „DLE“ fehlt. Muss dennoch einmal im normalen Datenstrom ein DLE vorkommen wird dieses durch den Sender verdoppelt. Der Empfänger erkennt dies und kann das verdoppelte Zeichen wieder entfernen.

7) *Welche physikalischen Aspekte treten bei der Übertragung von elektrischen Signalen auf? Erklären Sie diese kurz. Was bedeuten diese Aspekte für die Bitsynchronisation und für die maximal erreichbare Bitrate.*

Signale können nicht ohne Signalverluste übertragen werden. Leider werden die Signale nicht gleichmäßig verringert. Dies würde bedeutet, dass einfach die Amplitude verringert wird. Die Signale werden jedoch ungleich verringert und dadurch verzerrt. Ein weiterer Grund dieser Verzerrung ist, dass die verschiedenen Übertragungssysteme das Signal unterschiedlich verringern. Normalerweise werden Signale bis zu einer bestimmten Frequenz nahezu unvermindert gesendet, ab einer bestimmten Frequenz treten jedoch hohe Verringerungen auf. Weiters kann es zu einer Verzögerung kommen, da kein Übertragungssystem alle Komponenten mit derselben Geschwindigkeit übertragen kann. Das geht sogar soweit, dass schnelle Bits

langsame Bits überholen. Dadurch werden die Bits natürlich vertauscht. Als ein weiterer Störfaktor erweisen sich Nebengeräusche (Ungewollte Energie von Außerhalb).

Daher kommt von vornherein eine stark verminderte Signalqualität beim Empfänger an. Wenn die Bitrate weiter gesteigert wird die Bit-Synchronisation in der Mitte des Bits erschwert und schlussendlich ab einer bestimmten Bitrate unmöglich gemacht. Daher gibt es eine Beziehung zwischen Bandbreite, Leitungslänge und Maximaler Bitrate, welcher das „Nyquist's Law“ bzw. „Shannon's Law“ gewidmet sind.

8) *Was besagen die Theoreme von Nyquist und Shannon? Erläutern Sie diese kurz. Was versteht man unter „Baseband“, „Narrowband“ und „Broadband“ Übertragung im Zusammenhang mit Datenkommunikation?*

Nyquist's Law: „Nyquist's Law“ beschäftigt sich damit wie viele Bits über einen vollkommen störungsfreien Leiter übertragen werden können.

$$R = 2 * B \log_2 V$$

[R=max. Bitrate, B=Bandbreitenbereich einer limitierten Bandbreite, V=Anzahl der Signallevele {Üblicherweise 2 für binäre Übertragung}] ... Analoges Telefonnetz: ~ 3000 Hz

Shannon's Law: Shannon's Law beschäftigt sich mit der maximalen Bitanzahl über einen realen Leiter (inkl. Störungen)

$$R = B * \log_2 (1 + S / N)$$

[R=max. Bitrate, B=Bandbreitenbereich einer limitierten Bandbreite, S=Signalstärke, N=Rauschstärke]

Baseband: Die gesamte verfügbare Bandbreite wird benutzt um ein Signal zu übertragen. – Signale werden als rechteckige Impulse übertragen – Physikalischer Zustand des Mediums, Sendeleistung, Sensitivität des Empfängers und Signal/Rausch-Stärke sind die begrenzenden Faktoren der erreichbaren Bitrate – Das Signal wird encodiert um Bit Synchronisation sicherzustellen um den Gleichanteil zu verringern.

Narrowband: Die Bandbreite wird intern limitiert und die Signale müssen in Analogsignale umgewandelt werden. Dies übernimmt ein Modem (Mo(dulator)dem(odulator)). Dafür gibt es verschiedene Techniken: Amplitudenmodulation, Frequenzmodulation, Phasenmodulation, Kombinationen der 3 Verfahren → QAM

Broadband: Die verfügbare Bandbreite der Seriellen Leitung wird geteilt um eine Vielzahl von Lower Bandwidth Pfaden auf einer Leitung zu bekommen. In Analogen System ist jeder Pfad durch einen eigenen Carrier moduliert. Jede bestimmte Basis-Frequenz benutzt ein bestimmtes Frequenzband des Übertragungssystems (Beispiel: Kabelfernsehen). In digitalen Systemen meint Breitband einfach „Hohe Geschwindigkeit“

9) *Erklären Sie den Unterschied zwischen connectionless und connection-oriented Service im Zusammenhang mit Leitungsprotokollen (Line Protocols). Gehen Sie dabei auf die Begriffe Layer, Protokoll und Service ein. Wie erfolgt üblicherweise die Fehlerbehebung (Error Recovery)? In welcher Spielart ist Fehlerbehebung und Flußkontrolle (Flow Control) möglich.*

Protokolle regulieren und kontrollieren die Kommunikation zwischen 2 Stellen über Punkt zu Punkt Leitung.

Basisaufgaben:

Frame Synchronisation

Frame Protection

Error Detection → Diese Aufgaben sind im Normalfall in die Hardware integriert, sodass sich das Protokoll im Normalfall nicht darum kümmern braucht.

Optionale Aufgaben: Verbindungs- und Leitungsmanagement, Error recovery, Flow control → sind im Normalfall in der Software implementiert.

Die **Arbeitsweise** von Protokollen kann mittels des 3 Layer Modells dargestellt werden. Jeder Layer stellt eine bestimmte „Dienstleistung“ bereit. Applikation, Kommunikations-Software, Kommunikations-Hardware. Das Protokoll vermittelt nun zwischen der Kommunikations-Software des Senders und des Empfängers. Die Applikation benutzt somit die Kommunikationssoftware zur Datenübertragung, welche ihrerseits ein Protokoll benutzt um die Daten tatsächlich zu übertragen. Davon bekommt die Applikation jedoch nichts mit. Die Kommunikations-Software stellt somit der Applikation eine Dienstleistung zur Verfügung. Dieser Service kann connectinless (CL) oder connection-oriented (CO) sein.

CL: Die Kommunikations-Software unterstützt nur die Basisaufgaben (siehe oben). Bei Übertragungsfehlern müssen die fehlerhaften Blöcke weggeworfen werden und die Wiederbeschaffung der Daten fällt auf die Applikation zurück. Für diese Übertragungsmethode werden keine speziellen Rahmentypen benötigt.

CO: Die Kommunikations-Software muss vorher eine Verbindung mit dem anderen Device starten. (logische Verbindung). Übertragungsfehler werden durch die Kommunikations-Software entdeckt und mittels feedback error control korrigiert (Neuanforderung der Daten, ARQ). Die Applikation muss sich nicht mehr um die Error recovery kümmern. Allerdings sind spezielle Rahmentypen wie connect, disconnect notwendig. Vor und am Ende der Übertragung gibt es somit einen Connection request gefolgt von einem connection acknowledgement, vice versa.

Flusskontrolle: Wenn die Daten schneller ankommen als sie die Applikation verarbeiten kann wird wenn möglich Flusskontrolle eingesetzt. Ansonsten müssten wenn der Pufferspeicher des Empfängers voll ist, korrekt empfangene Daten gelöscht werden. (Diese würden über Error recovery erneut angefordert werden müssen). Anhand von Stop and Go Messages wird dem Sender mitgeteilt wann er den Sendevorgang zu stoppen bzw. wann wieder aufzunehmen hat.

10) Erklären Sie das Grundprinzip von ARQ. Gehen Sie im Detail auf die ARQ-Variante Idle-RQ ein. Verwenden Sie zur Erklärung Protokollablaufdiagramme und erläutern Sie die benötigten Betriebsmittel wie verwendete Rahmentypen, Rahmen-Identifizierer, Retransmission List und Receive List und sowie deren Zusammenspiel. Wozu wird der Timeout-Mechanismus bei diesen Verfahren benötigt?

Grundprinzip: ARQ → Automatic Repeat Request. Der korrekte Empfang eines Datenpakets wird vom Empfänger bestätigt (ACK – Message). Der Sender speichert die Datenpakete so lange bis er die Bestätigung erhalten hat. Wenn er keine Bestätigung erhält wartet dieser ein Timeout ab und sendet das Datenpaket anschließend erneut. Bei diesem Verfahren müssen die einzelnen Pakete gekennzeichnet werden um genau bestimmen zu können welche Pakete der Reihe nach kommen.

Idle RQ: Das Idle RQ Prinzip schaut so aus, dass immer ein Datenframe gesendet wird, danach wartet der Sender die Empfangsbestätigung ab und sendet erst danach das nächste Frame. Diese Methode kann durch eine NACK-Message verbessert werden, da der Sender im Falle eines Datenverlustes kein Timeout abwarten muss sondern das Frame sofort neu senden kann. Es werden nur 2 Identifizierer benötigt (0,1 → Altes Frame, Neues Frame). Diese Technik wird beim Halb Duplex Protokoll eingesetzt. Voll Duplex Leitungen werden so extrem uneffizient ausgenutzt.

Der Timeout Mechanismus wird benötigt falls die Übertragung Sender → Empfänger oder die Empfangsbestätigung Empfänger → Sender nicht beim anderen ankommt. Sobald das

Timeout abgelaufen ist und keine Bestätigung angekommen ist, wird der Rahmen neu gesendet, egal ob der Rahmen selbst oder nur die ACK-Message nicht angekommen ist.

Diagramme Seite 02 – 9

11) Erklären Sie die ARQ-Variante Continuos-RQ mit „GoBackN“ im Detail. Verwenden Sie zur Erklärung Protokollablaufdiagramme und erläutern Sie die benötigten Betriebsmittel wie verwendete Rahmentypen, Rahmen-Identifizier, Retransmission List und Receive List und sowie deren Zusammenspiel. Wozu wird der Timeout-Mechanismus bei diesen Verfahren benötigt?

GoBackN: Im Fall eines Fehlers werden alle Rahmen seit dem Fehler richtig angekommen sind erneut angefordert. Alle nachfolgenden Frames werden so lange nicht angenommen bis das fehlende Frame gesendet wird. Dies hat den Vorteil, dass das Protokoll die Rahmen nicht ordnen muss. Sie kommen in jedem Fall in der richtigen Reihenfolge an. Es besteht die Möglichkeit mittels eines ACK's alle vorangegangenen Rahmen zu bestätigen. (Multi Acknowledgement). Jeder Rahmen startet selbst einen neuen Timeout Prozess. Dieser wird erst beendet wenn das ACK für diesen Rahmen erhalten wird. Der Timeout Mechanismus wird in diesem Fall nur benötigt wenn beispielsweise der letzte Rahmen nicht angekommen ist. Dadurch wird kein ACK gesendet und es gibt auch keine nachfolgenden Rahmen mehr wegen multiple ACK. Aus diesem Grund greift das Timeout und der Rahmen wird nach gewisser Zeit erneut gesendet um dann ein gültiges ACK zu erhalten. (inkl. NACK)

Diagramme Seite 02 – 17

12) Erklären Sie die ARQ-Variante Continuos-RQ mit „Selective Acknowledgement“ im Detail. Verwenden Sie zur Erklärung Protokollablaufdiagramme und erläutern Sie die benötigten Betriebsmittel wie verwendete Rahmentypen, Rahmen-Identifizier, Retransmission List und Receive List und sowie deren Zusammenspiel. Wozu wird der Timeout-Mechanismus bei diesen Verfahren benötigt?

Selective Acknowledgement: Jeder Datenrahmen muss explizit bestätigt werden. Wird die Bestätigung nicht empfangen wird das Datenpaket erneut gesendet. Fall 1: Das fehlende Frame ist an der falschen Stelle und muss neu geordnet werden. Fall 2: Es ist nur das ACK verloren gegangen → somit muss das doppelte Frame gelöscht werden. Jeder Rahmen startet selbst einen neuen Timeout Prozess. Dieser wird erst beendet wenn das ACK für diesen Rahmen erhalten wird.

Diagramme Seite 02 – 15

13) Erklären Sie die ARQ-Variante Continuos-RQ mit „Positive Acknowledgement“ im Detail. Verwenden Sie zur Erklärung Protokollablaufdiagramme und erläutern Sie die benötigten Betriebsmittel wie verwendete Rahmentypen, Rahmen-Identifizier, Retransmission List und Receive List und sowie deren Zusammenspiel. Wozu wird der Timeout-Mechanismus bei diesen Verfahren benötigt?

Continuos-RQ: Datenrahmen werden so lange bestätigt wie sie in der richtigen Reihenfolge ankommen (Multiple ACK's können benutzt werden. → Für den Fall dass nur ein ACK fehlerhaft ankommt übernimmt das nächste auch die Rolle für das voran gegangene) Sobald ein Frame nicht ankommt wird die Bestätigung gestoppt. Trotzdem werden die nachkommenden Rahmen gespeichert. Da jedes Frame einen Timer startet werden die Fehlenden Frames automatisch nach Ablauf des Timeouts nachgesendet. Der Empfänger kann dann mittels Multiple ACK alle empfangen Rahmen bis zu dem wo er sich im Moment befindet bestätigen. Sobald ein Timeout auftritt werden die nachfolgenden Timer erhöht, damit

korrekt empfangene Rahmen nicht nochmals gesendet werden müssen, weil die Bestätigung zu lange dauern würde. Der nachgesendete Rahmen befindet sich natürlich nicht in der richtigen Reihenfolge und muss daher vom Empfänger bzw. durch das Protokoll umsortiert werden um wieder in der richtigen Reihenfolge zu sein.

Diagramm 02 – 20

14) Erklären Sie die ARQ-Variante Continuous-RQ mit „Selective Reject“ im Detail. Verwenden Sie zur Erklärung Protokollablaufdiagramme und erläutern Sie die benötigten Betriebsmittel wie verwendete Rahmentypen, Rahmen-Identifizier, Retransmission List und Receive List und sowie deren Zusammenspiel. Wozu wird der Timeout-Mechanismus bei diesen Verfahren benötigt?

Selective Reject: Datenrahmen werden so lange bestätigt wie sie in der richtigen Reihenfolge ankommen (Multiple ACK's können benutzt werden. → Für den Fall dass nur ein ACK fehlerhaft ankommt übernimmt das nächste auch die Rolle für das voran gegangene) Im Falle eines Fehlers wird nur der fehlerhafte Rahmen mittels SREJ(N) neu angefordert. Dadurch befinden sich die nachgesendeten Rahmen natürlich nicht in der richtigen Reihenfolge und müssen vom Empfänger bzw. durch das Protokoll umsortiert werden um wieder in der richtigen Reihenfolge zu sein. Jedes Frame startet einen eigenen Timer welcher resettet wird sobald die Bestätigung empfangen wurde. Kommt einmal keine Bestätigung an und tritt somit das Timeout in Kraft wird der Rahmen einfach neu gesendet. Hat ihn der Empfänger schon richtig empfangen muss er das Duplikat erkennen und entfernen.

Diagramm 02 – 22

15) Was sind Sequenznummern? Welche Typen gibt es? Wie wird deren Handhabung mittels Registervariablen realisiert? Wie arbeiten diese Elemente zusammen? Was versteht man unter piggy-backed Acknowledgement? Wozu dienen Keepalive Messages?

Um Datenrahmen eindeutig identifizieren zu können werden Sequenznummern benötigt. Sie werden benötigt, da ansonsten einerseits fehlende Rahmen nicht erkannt werden würden und auf der anderen Seite Duplikate nicht erkannt werden würden. Im Übrigen werden Sie für die Sortierung im Selective Reject, Positive Acknowledgement und Selective Acknowledgement benötigt. Die Nummer wird im I-Frame mitgesendet. Außerdem wird sie in ACK/NACK/SREJ angegeben. Im Übrigen müssen Registervariablen angegeben werden $V(S)$, $V(R)$. Diese müssen alle beim Verbindungsaufbau auf 0 gesetzt werden.

Beispiel für GoBackN: $V(S)$ gibt die Sequenznummer vom nächsten zu sendenden Frame an – $V(R)$ gibt die Sequenznummer vom nächsten zu empfangenen Frame an. Dieser Wert wird auch in $N(R)$ gesehen. Beim Senden des Rahmens wird $N(S)$ auf den Wert von $V(S)$ gesetzt. Der Empfänger akzeptiert den Rahmen nur wenn $N(S)$ gleich $V(R)$ ist. Vor der Bestätigung wird $V(R)$ dann auf den nächst höheren Wert gesetzt. Und dieser Wert dann mit $N(R)$ als Bestätigung gesendet.

Piggy-backed Acknowledgement: Die Bestätigung jedes Datenrahmens ist nur bei einseitigem Datenfluss wirklich notwendig. Acknowledge Frames produzieren unnötigen Overhead bei Vollduplex Leitungen. Daher hat mit diesen Acknowledgements in einem Datenframe welches in die andere Richtung fließt untergebracht. Gibt es keine Daten zurückzusenden wird wieder ein normales ACK Frame gesendet.

Keepalive Messages: Nach dem Verbindungsaufbau kommt es immer wieder zu Zeiten in denen keine Daten gesendet werden müssen. Damit der Sender allerdings weiß ob der Empfänger noch immer da kann er eine so genannte keepalive Message lossenden. Wird diese vom Empfänger bestätigt weiß dieser, dass er noch immer da ist.

Beispiel für eine Keepalive Technologie: Der Sender Senden „Hello“ und der Empfänger antwortet mit einem ACK (X) Message. Das X steht für die Sequence Number des Datenrahmens, der als nächstes erwartet wird.

16) Was versteht man unter Windowing? Wozu wird es benötigt? Was ist ein Sendefenster? Was sind die zusätzlichen Auswirkungen bezüglich Anzahl der Identifier?

Ohne Beschränkung der Anzahl von unbestätigten Daten-Frames würde Continues-RQ eine unbegrenzte Anzahl an Sequenz Numbers und Pufferspeicher benötigen. Aus diesem Grund wurde die Anzahl limitiert. Wenn dieses Limit erreicht ist wird der Versand von Datenrahmen gestoppt bis die Bestätigung für einen anderen Rahmen erhalten worden ist.

Windowing reduziert den Pufferspeicher der „Retransmission List“ und der „Receive List“. Auf $W * \text{maximum frame size}$. Weiters reduziert Windowing die Anzahl der Identifier. Es werden $W+1$ Identifier benötigt (es wird bei ACK immer das nächste angefordert)

Warum $W+1$?: (Annahme: $W=3$) Der Sender sendet Frames mit den Identifiern 0,1,2. Falls für diese jetzt noch keine Bestätigung angekommen ist, muss beim letzten ACK jedoch ACK (3) gesendet werden, da sonst der Identifier 0 doppelt vorkommen würde, würde man jetzt wieder mit ACK(0) weitermachen.

Größe des Sendefensters: Ist Abhängig von: Antwortzeit, Puffergröße des Senders/Empfängers, Bandbreite. Das Window muss groß genug sein um die volle Bandbreite der Leitung ausnutzen zu können.

Windows Size: $RTT \times BW$

$RTT = 2 \times \text{Propagation Delay} + \text{Serialization Delay}$

Propagation Delay: Distance in m / Velocity m / sec (üblicherweise 200.000 km/s)

Serialization Delay: Number of Bytes * 8 / Bitrate in sec. * 1000

Optimale Window-Größe: ACK's treffen genau zum richtigen Zeitpunkt ein um das Window offen zu halten. Ist das Window zu klein muss die Übertragung immer wieder gestoppt werden bis das ACK da ist. (Schlechtester Fall Idle-RQ). Window ist zu groß: Im Fall von Fehlern müssen viele Frames neu übertragen werden.

Diagramm Seite 02 – 35

17) Was versteht man unter Flußkontrolle (Flow Control) und wie kann sie realisiert werden? Warum reicht Windowing dafür alleine nicht aus? Was versteht man unter „adaptive Windowing“?

Flow-Control: Flow-Control wird dort eingesetzt wenn der Sender mehr Daten sendet als der Empfänger verarbeiten kann. Dieser besitzt keinen unerschöpflichen Buffer (Zwischen-Speicher) und sonst müsste er korrekt empfangene Frames löschen. Aus diesem Grund signalisiert der Empfänger dem Sender mittels so genannter Stop und Go Messages wann er die Übertragung abbrechen bzw. fortsetzen soll. Dies kann mit der ACK Message verbunden werden wie z.B. im HDLC Protokoll (Stop RNR (3), Go: RR(3))

Windowing könnte ebenfalls für die Flusskontrolle benutzt werden. Der Empfänger würde schlichtweg keine Bestätigungen senden im Fall von „Datenstau“ bzw. die Übertragung würde gestoppt werden, wenn das Sendefenster geschlossen wird.

Problem: Wenn der Empfänger keine Bestätigung sendet → Timeout → Das entsprechende Frame würde nochmals gesendet werden und der Sender würde nach mehreren Versuchen den Sendevorgang abbrechen.

Im Fall von Volldublex Datenkommunikation: Stop&Go Messages werden für die Flusskontrolle in beiden Richtungen verwendet.

Weiters können Stop&Go für Keepalive Prozeduren verwendet werden.

Adaptive Windowing: Ein Windows kann konstant oder dynamisch sein:

Statisch: Gleichbleibendes Sendefenster

Dynamisch: Beim Verbindungsaufbau wird ein Startwert vereinbart. Dieser kann im Betrieb variabel verändert werden. Er kann auch für Stop & Go verwendet werden (z.B. Window = 0 → Stop, Window >0 = Go) – wird z.B. von TCP benutzt.

Fragenbereich 2 (HDLC, TDM, Network Principles)

18) Welche Stationstypen, Leitungskonfigurationen (Line Configuration) und Betriebsarten (Modes of Operation) sind in HDLC vorgesehen? Was kennzeichnen Command und Response? Welche Verwendung findet die HDLC Adresse und das P/F-Bit? Charakterisieren Sie kurz das HDLC Protokoll im verbindungsorientierten Modus (ARQ Type, Transmission Methode, etc).

Allgemeine Begriffserklärung

Halb-Duplex Leitung: Es kann immer nur in eine Richtung gesendet werden. Das Senderecht wird von einem Partner zum anderen weitergegeben (Token). Nur derjenige, der den Token besitzt, darf Frames senden. Durch ein Master-Slave System wird sichergestellt, wer den Token als erstes besitzt. (P/F Funktion)

Modem: Ein Modem wandelt digitale Signale so um, damit sie über die analoge Telefonleitung transportiert werden können. Das wird über verschiedene Modulationstechniken erreicht (FM, AM, Phasenmodulation, QAM, Trellis-Code, ...)

P/F Prozedur bei einer Halb-Duplex Leitung: Master besitzt RTS=1 → Sendet Frames zum Slave. Sobald der Master das letzte Frame gesendet hat, entfernt er das RTS-Signal und das Modem beendet den Sendevorgang, das Slavemodem stoppt den Empfangsvorgang. Muss der Slave etwas senden, wird dieser durch das RTS-Signal aktiviert → sein Modem startet den Sendevorgang bzw. das Master-Modem startet den Empfangsvorgang. Nach dem letzten Frame entfernt das Slave-Modem das RTS-Signal und stoppt das Senden, während das Master-Modem das empfangen einstellt.

Um Kosten zu sparen, wurden Multipoint-Leitungen eingeführt. Diese Leitung teilen sich mehrere Stationen. Solche Leitungen können jedoch höchstens von 2 Stationen gleichzeitig benutzt werden (im Fall von Vollduplexleitungen) → Praktische Anwendung: Central Station ↔ viele Remote Stations.

P/F Prozedur bei Multipoint: Central Station = Master, Remote Stations = Slaves (können nicht untereinander kommunizieren), der Master kontrolliert die Leitung. P/F → P sagt dem Slave, er muss dem Master Daten senden, F markiert das letzte Frame, welches der Slave sendet. Die Identifikation des Slave wird mittels einer Adresse durchgeführt. Wird etwas gesendet, hören alle Slaves mit, obwohl die Frames nur für einen Slave bestimmt sind.

Das HDLC-Protokoll basiert auf synchroner Übertragung, bit-orientiertem Protokoll mit Bitstuffing, Continuous RQ (GoBackN), P/F-Prozeduren, Halb/Voll-Duplex-Übertragungen, Punkt-zu-Punkt-Übertragung, Multipoint-Übertragung, Switched oder Non-Switched Kanäle.

Stationstypen: *Primary Station:* Master, Überträgt Kommandoframes, Empfängt Antwortframes, Hält eine eigene Sitzung mit jeder Station einer Multipoint-Leitung.
Secondary Station: Slave, Empfängt Kommandoframes, Sendet Antwortframes, Können untereinander nicht kommunizieren

Leitungskonfigurationen:

„**Unbalanced Mode**“: Eine Primary und eine oder mehrere Secondary Stationen, Primary kontrolliert die anderen Stationen, Primary stellt die Verbindung her und ist für Error-Recovery zuständig. Kann für Point to Point oder Multipoint Leitungen benutzt werden.
Addressing: Secondary haben eine Adresse, Kommandoframes beinhalten die Adresse der Secondary Station, Antwortframes enthalten ebenfalls diese Adresse.

„**Balanced Mode**“: Nur Point-to-Point Leitungen. Stationen sind gleichberechtigt und haben die selbe Verantwortung für Error Recovery und Leitungsmanagement. Es werden „Combined Stations“ benötigt.

Combined Stations: Enthalten Protokollkomponenten für die Primary und Secondary Station. Übertragen Kommandos und Antworten, Empfangen Kommandos und Antworten. Benutzen Adressen um zwischen Kommandos und Antwortframes unterscheiden zu können.

Modes of Operation: Unbalanced: NRM (Normal Response Mode)
ARM (Asynchronous Response Mode)
Balanced: ABM (Asynchronous Balanced Mode)

NRM: Die Secondary Station muss explizite Berechtigung von der Primary Station haben um zu senden. Nach der Berechtigung übermittelt die Secondary Station die Antwort-Frames welche Daten beinhalten können. Nach dem letzten Antwort-Frage übergibt die Secondary Station das Senderecht wieder der Primary Station. Die Secondary Station muss nun wieder auf Sendeberechtigung waren. Diese Übertragungsmethode eignet sich am besten für Halb-Duplex Leitungen.

ARM: Erlaubt der Secondary Station eine Übertragung ohne Berechtigung der Primary Station so starten. Eine Voll-Duplex Leitung ist erforderlich. Reduzierter Overhead. Primary Station ist nach wie vor für Error recovery und Line-Management verantwortlich. Bei Multipoint Leitungen darf es nur eine Secondary Station geben. ARM wird heutzutage wenig benutzt.

ABM: Benutzt Combined Stations, Die Station kann eine Übertragung einleitung ohne vorher eine Berechtigung der Anderen Stationen abzuwarten. Beide Stationen sind für Error recovery verantwortlich und können die Verbindung starten und wieder beenden. Beste Lösung für Punkt-zu-Punkt Leitungen.

19) Erläutern Sie den Aufbau eines HDLC Rahmens und geben Sie die Formate des Control Feldes an. Wofür werden die einzelnen Formaten bzw. Rahmentypen prinzipiell verwendet. Welche Dienste lassen sich damit realisieren?

Der **HDLC Rahmen** besteht aus: Flag field (8 bits)
Address Field (8/16 bits)
Control Field (8/16 bits)
Information field (variable bits)
Frame check sequence (16/32 bits)

Spezielle Sequenzen: Flag: 01111110
Abort: mind. 7 0en, weniger als 15 0en
Idle: mehr als 15 0en.

HDLC ist Code Transparent (Bit Stuffing wenn Flag Sequenz im Frame vorkommen würde)

Control Field:

I(nformation) Format → Werden für die Datenübertragung genutzt – Benötigen eine bestehende Verbindung – Senden Sequence Number N(S), empfangen Sequence Number N(R)
Bereich der Sequence Numbers 3 bit (Window: 7) oder 7 bit (Window: 127)

S(upervisory) Format → Kontrollfunktionen z.B. RR (Bestätigung falls kein I-Frame zum Senden vorhanden ist) oder Flow-Control: Go, RNR (Teilt mit, dass der Empfänger nicht bereit ist: Flow Control: Stop), Diese beiden Methoden können auch für keepalive benutzt werden. Weiters gibts es: SREJ, REJ

U(nnumbered) Format → z.B. Verbindungsaufbau, Verbindungsabbau, Reset der Verbindung, Testzwecke, ID-Austausch, Connectionless Informationsübertragung

HDLC als Connection Oriented Service: U-Frame baut die Verbindung auf, I und S Frames werden für die Datenübertragung genutzt, das U-Frame beendet die Verbindung wieder.

HDLC als Connectionless Service: Es werden nur U-Frames benutzt (UI für Datentransport)

20) Welche Fehlerbehebungsstrategien sind beim HDLC-Protokoll vorgesehen? Gehen Sie dabei auf Checkpointing und auf die HDLC-Optionen REJ, SREJ im Detail ein.

Beim HDLC Protokoll unterscheidet man zwischen 2 Fehlerbehebungsstrategien

Im NRM Modus wird Checkpointing eingesetzt, im ARM/ABM Modus wird REJ/SREJ eingesetzt.

Checkpointing: Primary sendet ein P=1 um Information über den aktuellen Stand der Sequenz-Number zu erhalten. Im Falle einer Fehlerhaften Übertragung erkennt er dies nun an der falschen Sequenznummer und sendet die Entsprechenden Frame(s) nochmals im GoBackN Verfahren. Die Übertragung der Sequenznummer erfolgt mittels RR oder RNR. Es wird daher kein richtiges NACK Frame gesendet

REJ/SREJ: REJ wird benutzt um eine Übertragung sofort so beginnen wenn ein Fehler entdeckt wird. REJ ist in diesem Fall ein richtiges NACK Frame. Bei dieser Methode ist es nicht notwendig auf das Checkpointing des Primarys zu warten. Auch in diesem Fall wird GoBackN angewandt. SREJ kann benutzt werden um ein spezielles Frame neu anzufordern.

21) Was versteht man unter Multiplexen im Allgemeinen und Zeitmultiplexen (TDM) im speziellen? Gehen Sie im Detail auf das synchrone Zeitmultiplexen ein. Welche Eigenschaften hat dieses Verfahren hinsichtlich Übertragungszeit eines Bytes, Echtzeitfähigkeit, Handhabung von Übertragungspausen der Endgeräte, Protokolltransparenz? Wäre Flow Control erforderlich bzw. wünschenswert in diesem Zusammenhang?

Allgemein: Leitungsprotokolltechniken wurden für die Verbindung von A nach B entwickelt. Die Bandbreite wird exklusiv von diesen beiden benutzt. Aus diesem Grund würden bei mehreren Verbindungen mehr Leitungen benötigt werden → sehr kostenintensiv. Aus diesem Grund wurden multiplex Techniken entwickelt um mehrere Verbindungen auf einer Leitung zu ermöglichen.

Ein Multiplexer nimmt mehrere Eingangsleitungen und multiplext sie auf eine „große“ Ausgangsleitung zusammen.

Zeitmultiplexen: Der Zeitmultiplexer erlaubt jedem Eingang eine bestimmte Zeitspanne (Zeitslot) zu benutzen. Die Einzelnen Leitungen werden somit zusammengesetzt und auf einer einzelnen Hochgeschwindigkeitsleitung übertragen. Die gesamte Kapazität wird Zeitlich auf die

einzelnen Kanäle verteilt. Am Bestimmungsort rekonstruiert ein Demultiplexer die einzelnen Kanäle wieder.

Man unterscheidet: Synchrones TDM (Zeitschlitz haben konstante Länge)
Asynchrones TDM (Zeitschlitz haben eine variable Länge)

Synchrones TDM: In einem periodisch generierten Frame befindet sich eine gleich bleibende Anzahl von Zeitschlitz mit gleicher Länge. Diese werden durch Nummern gekennzeichnet. Somit besitzt jeder Eingangskanal einen eigenen Zeitschlitz.

Die Geschwindigkeit errechnet sich aus Benutzergeschwindigkeit x Anzahl der Zeitschlitz

Vorteile: Es gibt nur extrem kurze Wartezeiten (multiplexen, demultiplexen)
Es kann jedes Protokoll verwendet werden, da diese Methode Protokolltransparent arbeitet.
Die Endsysteme bekommen vom Multiplexen „nichts mit“ – Sie denken, es besteht eine direkte Verbindung.

Nachteile: Die Trunkleitung benötigt eine hohe Bitrate (kostenintensiv)
Wenn keine Daten gesendet werden müssen Idles in dem Zeitschlitz gesendet werden
→ Bandbreitenverschwendung
Asynchrones TDM vermeidet diese beiden Fehler → benutzt Statistiken um die benötigte Bandbreite zu errechnen und dementsprechend bereitzustellen.

Flow Control hat keinen Sinn, da ohnehin genau 64 kbit für jede Leitung zur Verfügung stehen und daher nur kurzzeitig zwischengespeichert werden muss.

22) Wie erfolgt heutzutage die Übertragung von Sprache? Was versteht man unter PCM? Was stellt der DS0 Kanal dar? Was versteht man unter Multiplexer Hierarchien, welche gibt es und wozu dienen diese?

Die digitale Sprachübertragung basiert auf Shannon's Theorem. Analoge Signale werden mittels Pulse-Code-Modulation (PCM) in einen 64 kbit/s digitalen Datenstrom verwandelt. Diese Übertragung wird in den heutigen Telefonnetzen verwendet. (Synchrones TDM)

Das Analoge Signal wird dabei Quantisiert um daraus ein digitales Signal zu erhalten. Dabei entstehen jedoch so genannte „Quantization Errors“

Um die Signalqualität zu verbessern werden niedrigere Amplituden feiner aufgelöst (Logarithmische Quantization). → bessere Signalqualität für leisere Sprachteile. (Europa: A-Law (ITU))

PCM: PCM teilt den Datenstrom in ein 8-bit Muster ein (Polarität, Segment, Segment, Segment, Step, Step, Step, Step).

Da z.B. Mobiltelefone keine 64 kbit/s erreichen musste man sich überlegen wie man Sprachdaten wirkungsvoll komprimieren kann.

Adaptive Differential Pulse Code Modulation (ADPCM) (Es wird nur die Differenz zwischen 2 Pulsen übertragen, es werden weniger Bits für die Encodierung benutzt) (16,24,32,40 kbps)

Low Delay Code Excited Linear Predictor (16 kbps)

Conjugate Structure Algebraic Code Excited Linear Predictor (8 kbps)

DS0: Der DS0 Kanal stellt 1 Timeslot in Multiplexing Frames dar. Er ist die Basis für die hierarchische Datenkommunikation. Jedes Byte muss innerhalb von 125 µs ankommen. (8000 bytes / Sekunde).

Warum eine Hierarchie ? → Nur eine standardisierte Hierarchie kann Millionen von Benutzer auf der Welt verbinden.

Man unterscheidet **2 Hauptarchitekturen**: PDH, SDH

PDH: Synchroner Übertragung: Zeitdifferenzen werden mit Bitstuffing ausgeglichen. Wird für Niedriggeschwindigkeitsleitungen benutzt. Kann für höhere Geschwindigkeiten nicht benutzt werden, da der Overhead dabei drastisch ansteigt

SDH: Übergeht Nachteile von PDH. (Nämlich: Steigender Overhead, Verschiedene Multiplexing Strukturen, Wechsel von Kanälen erfordert demultiplexen). Weiteres forderte man ein echtes Synchrones Netzwerk. Es wurden add-drop MUXes und Ring Topologien gefordert.

23) Was versteht man unter Multiplexen im allgemeinen und Zeitmultiplexen (TDM) im speziellen? Gehen Sie im Detail auf das statistische (asynchrone) Zeitmultiplexen ein. Welche Eigenschaften hat dieses Verfahren hinsichtlich Übertragungszeit eines Bytes/Rahmens, Echtzeitfähigkeit, Handhabung von Übertragungspausen der Endgeräte, Protokolltransparenz, etc.? Wäre Flow Control erforderlich bzw. wünschenswert in diesem Zusammenhang?

Allgemein: Leitungsprotokolltechniken wurden für die Verbindung von A nach B entwickelt. Die Bandbreite wird exklusiv von diesen beiden benutzt. Aus diesem Grund würden bei mehreren Verbindungen mehr Leitungen benötigt werden → sehr kostenintensiv. Aus diesem Grund wurden multiplex Techniken entwickelt um mehrere Verbindungen auf einer Leitung zu ermöglichen.

Ein Multiplexer nimmt mehrere Eingangsleitungen und multiplext sie auf eine „große“ Ausgangsleitung zusammen.

Zeitmultiplexen: Der Zeitmultiplexer erlaubt jedem Eingang eine bestimmte Zeitspanne (Zeitslot) zu benutzen. Die Einzelnen Leitungen werden somit zusammengesetzt und auf einer einzelnen Hochgeschwindigkeitsleitung übertragen. Die gesamte Kapazität wird zeitlich auf die einzelnen Kanäle verteilt. Am Bestimmungsort rekonstruiert ein Demultiplexer die einzelnen Kanäle wieder.

Man unterscheidet: Synchrones TDM (Zeitschlitze haben konstante Länge)
Asynchrones TDM (Zeitschlitze haben eine variable Länge)

Asynchrones TDM: Üblicherweise übertragen die Devices in statistischen Mengen. Weiters müssen nicht immer Daten gesendet werden. Aus diesem Grund werden bei diesem Multiplex-Verfahren die notwendige Bitrate auf der Trunk-Leitung kalkuliert. Wenn mehrere Devices gleichzeitig übertragen wollen kann jedoch nur eines auf der Trunkleitung senden. Die Daten der anderen Teilnehmer muss zwischengespeichert werden. Außerdem muss gewährleistet werden, dass die Trunkleitung nicht ausschließlich von einem Teilnehmer benutzt wird.

Beim Asynchronen Zeitmultiplexen müssen die Daten eindeutig gekennzeichnet sein (Addressing), da keine Beziehung zwischen Timeslot und Portnummer wie beim synchronen TDM existiert. Ein Port Identifier wird benutzt um die Quelle zu adressieren und über die Trunk-Leitung mit gesendet.

24) Erklären Sie das Prinzip der Leitungsvermittlung (circuit switching) im Detail. Welches Zeitmultiplexverfahren liegt zu Grunde? Welche gängige Netzwerktechnologie beruht auf diesem Verfahren?

Wenn man viele Stationen miteinander verbinden will stellt sich die Frage wie man das anstellen bzw. organisieren kann. Eine einfache Lösung wäre jede Station mit jeder zu verbinden. Das wäre jedoch nicht machbar, da man eine unzählige Anzahl von Leitungen, Modems Repeatern ect. benötigen würde. Aus diesem Grund hat sich Circuit Switching aus Synchronem TDM entwickelt.

Erklärung: Physikalische Kommunikationsports sind logisch mit einem Synchronen TDM Switch verbunden. Die Trunk-Leitungen zwischen den Switches benutzen synchrones TDM. Jedem Port wird ein Timeslot an der ausgehenden Leitung zugeteilt. Die Switches mappen nun eingehende Trunks entweder zu einem eigenen Port oder verbinden sie weiter auf den ausgehenden Port. Die Mapping Information wird im Switch gespeichert.

Vorteile: Minimale Wartezeiten, Hohe Bitrate auf Trunk-Leitungen, Idle-Signal wenn keine Daten transportiert werden, Protokoll Transparenz.

Der Pfad der Daten durch das Netzwerk wird mittels Einträgen in die switching Tabellen markiert. Die Anzahl Physikalischer Ports dann weiter reduziert werden durch Benutzung von Synchronen TDM zwischen Device und dem lokalen Switch. Somit kann ein physikalisches Device viele logische Kanäle beinhalten.

Die Switching Tabelle kann statisch, dynamisch (fail-safe) oder dynamisch (on demand) sein. Der Netzbetreiber kann auch permanente Übertragungsleitungen einrichten. Es erfolgt somit ein permanenter Eintrag in die Switching Tabelle der Circuit Switches bzw. automatischer Weiterleitung im Falle eines Ausfalles einer Trunk-Leitung.

Wichtiges Anwendungsgebiet: ISDN (bietet Sprachübertragung, Videoübertragung, Datenübertragung) – Standardisiertes User-to-Network Interface (BRI, PRI)

BRI: Basic Rate Interface: 2 B-Kanäle mit 64 kbit/s, 1 Datenkanal mit 16 kbit/s, Diese 3 Kanäle werden TDM Multiplexed auf einer physikalischen Leitung

PRI: Primary Rate Interface: 30 B-Kanäle mit je 64 kbit/s, 1 Datenkanal mit 64 kbit/s → werden über eine physikalische Leitung TDM multiplexed.

25) Erklären Sie allgemein das Prinzip der Paketvermittlung (packet switching). Welches Zeitmultiplexverfahren liegt zu Grunde? Was sind Routingtabellen und wie können diese erstellt werden? Welche Funktion spielt die Adressierung in diesem Zusammenhang? Was versteht man unter „routable/routed protocols“ und „routing Protocols“

Packet Switching: Packet Switching ist die konsequente Umsetzung des Prinzips des statistischen Zeitmultiplexens in einer Netzwerkumgebung ausgehend von $N \cdot (N-1) / 2$ Problem bei der Vernetzung von N Lokationen. Damit treten die Basiseigenschaften des statistischen Zeitmultiplexens wie variables Delay durch Pufferung, Adressierung und keine fixe Zuordnung von Bandbreite (Timeslots) zu Kommunikationsbeziehungen nun auch in der Netzwerkumgebung auf (Anmerkung: Das Grundprinzip des statistischen Multiplexens soll hier an dieser Stelle nicht erklärt werden, weil das Inhalt der Frage 23 ist).

Der statistische Zeitmultiplexer, der nun i.a. mehr als ein Trunk-Port aufweist, heißt Packet Switch. Die Endsysteme werden über Access-Ports jeweils an den nächstgelegenen Packet Switch angeschlossen. Möchte ein Endsystem Daten senden, werden diese in ein Paket verpackt, mit Adresse versehen und an den lokalen Packet Switch übertragen. Der speichert das Paket zunächst zwischen, wertet die Adressinformation mittels Routing- oder Switching-Tabelle aus, stellt das Paket in die Warteschlange der abgehenden Leitung und sendet das Paket – wenn es an die Reihe kommt - schlussendlich Richtung Ziel weiter (Store and Forward Prinzip). Nachdem die Leitungen nach statischen Werten dimensioniert sind, ist die Wartezeit abhängig von der momentanen Statistik des Verkehrsaufkommens. Redundante Leitungen können für alternative Pfade bei Fehlern bzw. für Lastaufteilung Verwendung finden. Nachdem das Endsystem mit dem Packet Switch bezüglich Adressierung und Flow Control zusammenarbeiten muss, ist das Verfahren nicht mehr protokolltransparent. Endsystem und Switch müssen dieselbe Sprache sprechen (bspw. X.25, IP, etc.).

In der Routingtabelle/Switchingtabelle ist im Prinzip abgespeichert, welche Zieladresse über welches Port erreichbar ist. Man kann das mit einer Wegweisertechnik vergleichen, wobei in jedem Packet Switch Wegweiser für ein bestimmtes Ziel derartig aufgestellt sind, dass ein Paket entlang der Wegweiser zum gewünschten Ziel weitergeleitet werden kann.

Alle auf Packet Switching basierenden Netzwerke, die „routable protocols“ verwenden, benötigen für Endsysteme eindeutige und strukturierte Adressen (OSI Layer 3). Strukturiert heißt, dass sich darin die Topologie in irgendeiner Form widerspiegelt (Beispiele dafür sind Net-ID/Host-ID bei IP oder Ländercode (+43 für A) bei Telefonnetzen). (Anmerkung: Transparent Bridging (Ethernet Switching) ist auch Packet Switching (allerdings auf OSI Layer 2), verwendet aber unstrukturierte Adressen (MAC-Adressen), die keine Topologieinformation enthalten und daher auch nicht zusammengefasst werden können.

Die Routingtabellen (die Wegweiser) können entweder statisch von Netzwerkadministrator konfiguriert werden oder dynamisch unter Verwendung von Routingprozeßen und Routingprotokollen erstellt werden. Routing ist dabei der Prozess der Wegefindung, wenn mehr als ein Weg zum Ziel vorhanden ist. Routing Protokolle sind dabei die Kommunikationsmitteln der Packet Switches untereinander, um die Netzwerktopologie herauszufinden, alle Wege zu allen Zielen zu berechnen und den jeweils besten Weg zu einem Ziel in die Routingtabelle einzutragen. Statisch heißt, dass bei einer Änderung der Topologie der Netzwerkadminstrator händisch eingreifen muß. Dynamisch heißt, dass die Routingprotokolle Änderungen mitteilen und das entsprechend neue Wege automatisch ermittelt werden.

26) Erklären Sie den Datagramm-Dienst im Detail. Welche Dienstart (service) liegt zu Grunde? Wie werden dabei Pakete weitergeleitet? Geben Sie Vor- und Nachteile dieser Methode an. Welche Netzwerktechnologien beruhen auf diesem Verfahren?

Datagramm Dienst ist Packet Switching (OSI Layer 3 mit struktuierten Adressen) im connectionless Service Mode (Anmerkung: Prinzip von Packet Switching ist hier an dieser Stelle nicht zu beantworten, weil das Thema der Frage 25 ist). Endgeräte, die kommunizieren möchten, können das, ohne vorab eine Verbindung aufzubauen. Die Weiterleitung der Pakete (in diesem Mode als Datagramme bezeichnet) erfolgt nur mittels Routingtabellen anhand des momentanen Zustands der Routingtabellen.

Die Routingtabellen können entweder statisch von Netzwerkadministrator konfiguriert werden oder dynamisch unter Verwendung von Routingprozeßen und Routingprotokollen erstellt werden. Routing ist dabei der Prozeß der Wegefindung, wenn mehr als ein Weg zum Ziel vorhanden ist. Routing Protokolle sind dabei die Kommunikationsmitteln der Packet Switches untereinander, um die Netzwerktopologie herauszufinden, alle Wege zu allen Zielen zu berechnen und den jeweils besten Weg zu einem Ziel in die Routingtabelle einzutragen.

Inkonsistente Routingtabellen (Wegweiser, die im Kreis zeigen) können dazu führen, dass Datagramme endlos kreisen. Um die Puffer nicht damit sukzessive zu verstopfen, muß ein „Kill“-Mechanismus vorgesehen sein (in IP wird dazu das TTL Feld verwendet).

Jedes Datagramm enthält die vollständige Adresse. Jedes Datagramm wird unabhängig von anderen Datagrammen (die vielleicht ebenfalls zum selben Ziel transportiert werden sollen) vom Packet Switch behandelt. Damit können Datagramme einander überholen wenn bspw. durch Ausfall eines Knotens Rerouting über einen anderen Weg erfolgt. Es ist also nicht sichergestellt, dass die Datagramme in der gleichen Reihenfolge wie beim Absenden beim Ziel ankommen. Datagramme können - bedingt durch Stau oder durch Bitfehler - von Packet Switches verworfen werden (Packet Switches agieren ja connectionless). Eine Fehlerbehebung spielt sich außerhalb des Packet Switching Netzes ab (die Endsysteme müssen durch geeignete höhere Protokolle dafür sorgen). Man nennt das Datagramm Service daher auch Best-Effort Service.

Vorteile: Einfach in Packet Switches zu implementieren (wenig Programmieraufwand); schnellste Art zu kommunizieren (weil kein Verbindungsaufbau) wenn alles in Ordnung ist; Protokolle sind nicht so komplex

Nachteile: Best-Effort Service; Fehler müssen von Endsystemen selbst beseitigt werden; Flow Control sehr mühsam oder gar nicht durchsetzbar; Ressourcenreservierung (Bandbreite, max. Delay) gar nicht oder nur mit zusätzlichem Aufwand erreichbar

Beispiele: IP, IPX (Novell), Appletalk, Banyan Vines, XNS, OSI CNLS, Apollo Domain, Decnet Phase IV

27) Erklären Sie den Virtual Call-Dienst im Detail. Welche Dienstart (service) liegt zu Grunde? Wie werden dabei Pakete weitergeleitet? Welche Aufgaben haben dabei Routingtabellen? Was sind Switching Tabellen, wie werden diese erstellt und wozu dienen sie? Geben Sie Vor- und Nachteile dieser Methode an. Welche Netzwerktechnologien beruhen auf diesem Verfahren?

Virtual Call Dienst ist Packet Switching (OSI Layer 3 mit strukturierten Adressen) im connectionoriented Service Mode. (Anmerkung: Prinzip von Packet Switching ist hier an dieser Stelle nicht zu beantworten, weil das Thema der Frage 25 ist). Endgeräte, die kommunizieren möchten, müssen zunächst eine logische Verbindung aufbauen, bevor Datenpakete übertragen werden können. Der Verbindungsaufbau erfolgt durch Weiterleitung von speziellen Call-Setup Paketen mittels der auch in diesem Mode vorhandenen Routingtabellen (Wegweiser für strukturierte Zieladressen). Beim Weiterleiten der Call-Setup Pakete werden allerdings zusätzlich noch Switchingtabellen aufgebaut, die dann das Weiterleiten der Datenpakete nach erfolgreichem Verbindungsaufbau bewerkstelligen.

Call-Setup Paketen enthalten die vollständige, strukturierte Adresse und zusätzlich noch einen lokalen Connection-Identifizierer (Kurzwahlkennzeichen). Dieser ist jeweils nur auf einer Teilstrecke eindeutig und dient zur Unterscheidung der über diese Teilstrecke gelegten Verbindungen. Beim Weiterleiten wird ein Mapping des ankommenden Connection-Identifizierers zum angehenden Connection-Identifizierer in der Switching-Tabelle festgehalten. Datenpakete enthalten nur noch den lokalen Connection-Identifizierer (aber keine komplette strukturierte Adresse), wobei sich der Connection-Identifizierer von Teilstrecke zu Teilstrecke gemäß der Switchingtabellen ändert. Durch den Verbindungsaufbau wird quasi die Spur markiert, welche die Set-Up Pakete genommen haben. Die Spur ist in den Switchingtabellen festgehalten. Die Verbindung kommt zustande, wenn der gewünschte Teilnehmer den Verbindungsaufbau akzeptiert und bestätigt. Datenpakete folgen dieser Spur. Damit ist auch sichergestellt, dass die Pakete in der gleichen Reihenfolge wie beim Absenden beim Ziel ankommen (man nennt das Sequencing).

Endsysteme glauben nach erfolgreichem Verbindungsaufbau einen „point-to-point circuit“ (physikalischen Link) zu sehen, der durch die Connection-Identifizierer am Anfang und Ende gegeben ist. Sie sehen quasi eine - durch diese Connection-Identifizierer adressierbare - Transport-Röhre (pipe). Man spricht daher vom virtual (scheinbaren) circuit, weil natürlich tatsächlich Pakete von Hop-to-Hop über klassisches Store and Forward weitergeleitet werden. Durch Aufbau mehrerer virtual circuits kann ein Endsystem natürlich auch mehrere Verbindungen gleichzeitig unterhalten. Nach Fließen der Datenpakete werden Verbindungen üblicherweise wieder abgebaut.

Kommt es in einem solchen Netzwerk zu einem Ausfall eines Trunks oder Packet Switches, so werden die davon betroffenen Verbindungen unterbrochen und die Endsysteme über spezielle Disconnect Pakete vom Bruch der Verbindungen informiert. Die Endsysteme müssen daraufhin die Verbindungen erneut mittels Call-Setup Paketen anfordern und es muß eine neue Spur durch das Netzwerk gezogen werden (Annahmen: Redundanz ist im Netzwerk vorhanden und dynamisches Routing hat neue Situation in Routingtabellen bereits eingetragen bevor Verbindungsaufbau initiiert wird).

Ein Provider kann basierend auf dieser Technik entweder ein SVC (Switched Virtual Circuit) Service - mit Circuits on Demand wie oben geschildert – anbieten oder ein PVC (Permanent Virtual Circuit) quasi als Standleitungersatz. Beim PVC fällt der Verbindungsaufbau und Abbau weg, weil die Switching Tabellen permanent vom Provider eingerichtet sind und diese Verbindungen permanent vorhanden sind. Daten-Pakete werden wie gewohnt mittels Switching Tabellen anhand des lokalen Connection Identifizierers weitergeleitet.

Vorteile: Durch die Existenz einer Verbindung ist Ressourcenreservierung (Bandbreite, max. Delay, Quality of Service QoS) für eine Verbindung möglich; Flow Control zur Verhinderung von Stau im Netzwerk ist möglich; Ablehnung einer Verbindung bei Nicht-Vorhandensein der gewünschten QoS ist möglich; Error Recovery ist möglich

Nachteile: Verbindungsaufbau kostet Zeit; Großer Implementierungsaufwand, komplexe Protokolle

Beispiele für connection-oriented Packet Switching:

X.25: Local Connection Identifier = LCN (Logical Channel Number); „dichte“ Transportröhre durch Error Recovery auf jeder Teilstrecke (durch Bitfehler oder Stau verloren gegangene Pakete werden netzintern durch ARQ Techniken wiederholt); Flow Control durchsetzbar; inband signaling (Call-Setup Pakete fließen in der selben Röhre wie Datenpakete); Pakete variabler Länge

Frame Relay: Local Connection Identifier = DLCI (Data Link Connection Identifier); „undichte“ Transportröhre durch Weglassen von Error Recovery (durch Bitfehler oder Stau können Pakete verloren gehen); Flow Control durch Congestion Indication ersetzt und daher andere Maßnahmen zur Sicherung des Netzes erforderlich (Traffic Contract, Traffic Policing, Traffic Shapping); outband signaling (Call-Setup Pakete fließen in einer separaten Röhre); Pakete variabler Länge

ATM (Asynchronous Transfer Mode): Local Connection Identifier = VPI/VCI (Virtual Path Identifier / Virtual Channel Identifier); „undichte“ Transportröhre durch Weglassen von Error Recovery (durch Bitfehler oder Stau können Pakete verloren gehen); Flow Control durch Congestion Indication ersetzt und daher andere Maßnahmen zur Sicherung des Netzes erforderlich (Traffic Contract, Traffic Policing, Traffic Shapping); outband signaling (Call-Setup Pakete fließen in einer separaten Röhre); Pakete konstanter Länge -> Zellen genannt (cell = 53Byte)

28) Erklären Sie das Grundprinzip des OSI-7-Schichten (Layer) Modells (Service, Protocol, Encapsulation/Decapsulation, Relay Systems). Benennen Sie die einzelnen Schichten und charakterisieren Sie deren Aufgabe kurz.

Das OSI-Referenzmodell wurde entwickelt um einheitliche Netzwerkstandards zu gewährleisten. Da die Kommunikation zwischen Systeme extrem komplex ist, wurden die einzelnen Tasks in Untergruppen, so genannte Layer eingeteilt. Diese Layer sind hierarchisch angeordnet (Layer bietet Service für übergeordneten Layer an und erhält Service vom untergeordneten Layer.) Jeder Layer kann dabei auf seine eigenen vom System gegebenen Ressourcen zugreifen.

Durch das Layer-Modell wird die Übersichtlichkeit gewährleistet, allerdings steigt dadurch auch die Komplexität.

Das OSI-Referenzmodell gliedert sich in 7 Schichten auf

Wenn Daten vom obersten Layer (Application Layer) in die unteren Schichten „wandern“, werden die Daten eingekapselt bis zum Schluss der gesamte Rahmen über die physikalische Leitung gesendet werden kann. Im Endsystem werden diese Rahmen dann wiederum durch die einzelnen Layer entkapselt und jeder Layer kann sich die für ihn relevanten Informationen nehmen und den um damit verringerten Rahmen in den nächst-höheren Layer weiterleiten.

In den Endsystemen müssen alle 7 Layer implementiert sein damit die Datenkommunikation funktionieren kann. Wenn 2 Systeme nicht direkt miteinander verbunden sind müssen diese mittels Relay oder Intermediate-Systemen (Packet Switches, Repeater [Nur Layer 1-3 sind nötig]) verbunden sein.

Layer 1: Physical Layer

Zugang zum physikalischen Medium

Übertragung der Bitströme

Spezifikationen der Elektronischen und Mechanischen Schnittstellen

Signal Encodierung, Clock Synchronization
 Aktivierung und Deaktivierung der Verbindung zwischen Systemen

Standards: zb LANs: 802.3 (Ethernet, 10Base5, 10Base2, 10BaseT)
 802.5 (Token Ring)
 WANs (zb. V.24, V.28, W.35, V.36, RS 449)

Devices: Repeater, Hub

Layer 2: Data Link Layer

Transport der Rahmen über eine physikalische Verbindung des Netzwerks
 Rahmensynchronisation, Frame checking, Addressing (MAC Adressen, HDLC Adressen)
 Media Access Control (LAN)
 Error Recovery und Flow-Control (im Verbindungsorientierten Modus)

Devices: Bridge

Layer 3: Network Layer

Transportiert die Pakete über das Netzwerk
 Adressierung um ein Endsystem eindeutig identifizieren zu können
 Routing/Relaying/Switching falls die Systeme nicht direkt mit einem physikalischen Medium verbunden sind.
 Fragmentierung, Multiplexing

Devices: Packet Switch, Router, WAN-Switch

Layer 4: Transport Layer

Trennt das Netzwerk von der Applikation
 Adressiert die Prozesse im Endsystem (TSAP, Transport Service Access Point)
 Implementiert QoS gefordert von übergeordneten Layern
 Vertrauenswürdige Verbindung zwischen Endsystemen
 Flow Control zwischen Endsystemen

Layer 5: Session Layer

Koordiniert und kontrolliert die Verbindungen (den Dialog) zwischen verschiedenen Endsystemen

Layer 6: Presentation Layer

Ist für die gemeinsame Sprache zwischen Endsystemen verantwortlich (Übersetzung, Adaption von Daten)

Layer 7: Application Layer

Unterstützt den Benutzer mit gängigen Netzwerkanwendungen und Basis-Netzwerkfunktionen.

Fragenbereich 3 (Kapitel: LAN Principles, Legacy Ethernet, Transparent Bridging, Internetwork Basics, Ethernet Switching, VLAN and High Speed Ethernet):

29) Was sind die grundlegenden Charakteristiken von LANs? Welche OSI Schichten sind für eine Kommunikation innerhalb eines LANs notwendig? Warum ist bei LANs eine Aufteilung der OSI-Schicht 2 in zwei Subschichten LLC und MAC notwendig? Erklären Sie kurz die Funktion von LLC (prinzipielle Dienstarten, Funktion DSAP, SSAP). Was sind MAC-Adressen und wie erfolgt die Handhabung von Rahmen beim Empfang?

Geschichtliches: Local Area Networks wurden ursprünglich als gemeinsames Übertragungsmedium geschaffen. Es sind heutzutage Geschwindigkeiten bis zu 1 Gbit/s möglich, allerdings bei beschränkter Distanz. Auf Grund der hohen Geschwindigkeiten gab es keine Netzwerk-

elemente mit „Store and Forward“ und kein „Routing“. Es wurden nur einfache Topologien verwendet (Bus, Ring, Stern). Alle Netzwerkstationen sind am selben Medium und besitzen dieselben Rechte. Jede Station kann mit jeder anderen direkt kommunizieren.

Da jeder Netzwerkadapter eindeutig identifizierbar sein muss, ist eine weltweit eindeutige Adresse vonnöten. Diese nennt sich MAC (Media Access Control). Wenn eine Nachricht ausgesendet wird, wird sie von jedem angeschlossenen Teilnehmer empfangen. (Besteht aus 6 Byte)

OSI Layer 1 und 2 sind ausreichend um die LAN Aspekte abzudecken.

Empfang von Rahmen: Jeder Lan Adapter empfängt jedes Frame welches ausgesendet wird auf Grund des Broadcast Verhaltens eines LAN's. Jetzt entscheidet die Netzwerkkarte anhand der Adresse und seiner eigenen (Burned in Adress) ob das Frame zur nächsten Schicht (Layer 3) weitergeleitet werden soll. Danach wird der NIC die CPU unterbrechen wenn er das Frame weiterleiten muss. Höhere Layer sehen das Frame nur wenn MAC-Adresse stimmt, Broadcast, Multicast Adressen. Broadcasts sollten nur bei der Initialisierung der Netzwerkkarte verwendet werden.

2 Sublayer: Der OSI-Data Link Layer (Layer 2) wurde ursprünglich für Punkt zu Punkt Leitungen konzipiert. Da das LAN eine Multipoint Leitung ist wurde der Layer 2 in 2 Sublayer (Media Access Control, Logical Link Control) aufgeteilt.

MAC: Kümmert sich um: Framing, Addressing, Error Detection, Vermeidung von Kollisionen, Fairness, Priority Frames.

LLC: LLC (Logical Link Control) umfasst die Adressierung der Endsysteme so wie die Fehlerprüfung. Der LLC Header stellt das Bindeglied zwischen dem Sublayer MAC und der Schicht 3 dar. Er besteht aus DSAP (Destination Service Access Point), SSAP (Source Service Access Point) und Control.

LLC spezifiziert 4 Dienstmethoden und zugehörige Protokolle:

Class 1 verbindungsloser ungesicherter (unacknowledged) Datagrammdienst

Class 2 verbindungsorientierter Betrieb

Class 3 Class 1 plus Bestätigung (Acknowledgement)

Class 4 Class 2 mit Bestätigung (Tunnel, P2P).

DSAP/SSAP dienen als Kennzeichnung der höheren Protokollprozesse der Ziel- und Absendersysteme. (von 128 individuellen Werten für DSAP/SSAP sind 63 für IEEE-Protokolle reserviert bzw. weitere 63 für herstellerspezifische Protokolle/für Applikationen).

30) Erläutern Sie die ursprünglichen Aspekte des IEEE 802.3 LAN (Ethernet) wie Topologie, Zugriffsverfahren (Media Access Control), (Collision Window / Slot Time) und physikalische Reichweite, minimal und maximale Rahmengröße, anhand der 10Base5 und 10Base2 Techniken. Wie ist ein Ethernet-Rahmen, wie sind MAC Adressen aufgebaut. Was wird durch AUI, PLS und PMA realisiert? Was versteht man unter Transceiver (extern/intern)? Was ist ein Repeater bzw. was besagt die Repeater Regel?

Ursprünge des IEEE 802.3: Bus Topologie basierend auf Koaxialem Kabel

Terminatoren am Ende der Leitung

Definierte Übertragungs-Power limitiert maximale Kabellänge

2 Typen mit Baseband Übertragung (Manchester Codierung),

10Base2, 10Base5

1 Typ mit Modulation (Breitband Übertragung) (10Broad36)

CSMA/CD: Die Netzwerkstationen schauen zuerst ob gerade eine Kommunikation am Bus stattfindet bevor Sie eine Übertragung beginnen. Sie können die Übertragung entdecken und

senden erst wenn die die aktuelle Übertragung zu Ende ist. Dennoch kommt es auf Grund von simultanen Übertragungen zu Kollisionen. Kollisionen werden durch Überwachung des Gleichanteils auf der Leitung entdeckt.

Konfliktbewältigung: Alle Stationen brechen die Übertragung ab. Es wird ein JAM-Signal gesendet um sicherzustellen, dass alle Stationen die Kollision bemerkt haben. Es wird ein zufallsgesteuerter Timeout - Mechanismus gestartet, nach Ablauf des Timeouts beginnt die Station zu senden. Dieser Vorgang wird 16x wiederholt, danach wird dem nächsten Layer ein Error mitgeteilt, und es wird beim nächsten Frame weitergemacht.

Collision Windows / Slot Time:

Schlimmster Fall: Die Stationen müssen die doppelte Signaldauer für die Kollisionserkennung abwarten. (Ansonsten könnte eine auftretende Kollision übersehen werden)
Die maximal erlaubte Zeit wird „Collision Window“ oder „Slot Time“ genannt.
Ethernet definiert hierfür 51,2 Mikrosekunden (Minimale Framelänge von 64 byte). Die maximale Framelänge wird auf Grund der Fairness ebenso begrenzt.

Es besteht ein Zusammenhang zwischen Slot Time, Datenrate, Kabellänge und Minimaler Framelänge: Wählt man einen Wert, so lassen sich die anderen daraus ableiten.

Um eine akzeptable Kollisionserkennung gewährleisten zu können wurde die max. Kabellänge im 10 mbit Ethernet auf 2500-3000 Meter beschränkt.

Ethernet Adresse: Preamble (Clock-Synchronisation)
DA (Destination MAC Adresse)
SA (Source MAC Adresse)
Length (Länge des IEEE 802.3 Frames)
Daten
FCS (Checksumme)

Daraus resultiert eine minimale Länge eines Frames von 64 Byte und eine maximale von 1518 Byte. Standardmäßig ist zwischen zwei Frames ein Gap von 9.6 Mikrosekunden.

MAC Adresse: I/G (Individual, Group-Broadcast)
U/L (Universal, Local)
+ 48 bits Source / Destination Adresse

Die Layer 1 (Physical Layer) unterteilt sich in mehrere Sublayer:

PLS: Physical Layer Signaling fungiert als abstrakter Layer zwischen MAC und PHY
Bietet: Daten encodierung, decodierung, Übersetzung zwischen MAC und PHY
AUI: Attachment Unit Interface: Wird zur Verbindung mit PMA benötigt.
PMA: Ineface zwischen PLS und MDI
MDI: Spezifiziert die verschiedenen Anschlüsse.

Transceiver: (Transmitter / Receiver): Offeriert Senden und Empfangen von Signalen, Kollisionserkennung, Heartbeat Funktion, Jabber Control

Transceiver Typen: 10Base5, 10Base2, FOIRL, 10BaseT, 10BaseF

External transceiver: AUI Interface verbindet das Endsystem und den Transceiver

Integrated Transceiver: Der Transceiver ist in die Netzwerkkarte integriert. Die Karte stellt BNC, RJ45, ST-Anschluss zur Verfügung

10Base5: Maximale Kabellänge: 500 m
Minimale Länge zwischen 2 Transceivern: 2,5 m
Maximum: 100 Stationen

10Base2: Maximale Kabellänge: 185 m
 Minimale Länge zwischen 2 Transceivern: 0,5 m
 Maximum: 30 Stationen

Repeater: Ein Repeater erhöht die maximale Kabellänge eines Ethernets. Er regeneriert das Signal welches er empfängt und sendet es erneut aus. Es wird nicht gespeichert und gibt daher nur ein kurzes Delay welches im Kollisionsfenster miteinbezogen werden muss. Es können Kollisionen erkannt werden und alle anderen Ports werden mittels JAM-Signal benachrichtigt. Lokale Repeater verbinden direkt 2 Coax-Kabel. Remote Repeater verbinden 2 Repeater mittels eines so genannten Link-Segment.

Multiport Repeater haben noch immer nur eine Kollisions-Domäne

Repeater-Regeln: Die Kollisionsdomäne eines Ethernet LANS ist begrenzt (51,2 microsekunden)
Topologie: Maximal 5 Segmente (4 Repeater) → 2 Segmente müssen link-Segmente sein. (Die Länge des Link-Segments darf 500 m nicht überschreiten) (Max. Länge von 2500 m)
 Bei 4 Segmenten mit 3 Repeatern darf das Link-Segment 1000 m nicht überschreiten (Max. Länge von 3000 m)

31) Warum und wie hat sich die Ethernet-Technik von der ursprünglichen Koax-Bus-Technologie weiterentwickelt. Gehen Sie dabei auf Linksegmente, 10BaseT, 10BaseF und Hub ein. Welche Aspekte waren bei der Entwicklung von Fast- und Gigabit-Ethernet zu beachten? Gehen Sie auf die neuen Aspekte wie PCS, MII, GMII, 4B5B-Coding, 8B10B-Coding ein. Erläutern Sie die neuen Möglichkeiten wie Fullduplex Operation, Autonegotiation, Flow Control und Trunking. Wodurch wird Gigabit-Ethernet zu einer Quasi-WAN Technologie?

Anfangs ging man von einer reinen Bus-Topologie aus, welche man auf Grund der Begrenzung mittels eines Repeaters verlängern konnte. Der nächste Schritt war Multiport Repeater einzusetzen. Damit hatte man die Möglichkeit von einem zentralen Punkt aus sternförmig zu verkabeln. (10BaseT) – Diese Multiport Repeater werden „Hub“ genannt. Das Netzwerk hat jedoch noch immer eine zentrale Kollisionsdomäne und es kann nur HalbDublex gesendet werden. (CSMA/CD in da box). Der nächste logische Schritt war demnach das Netzwerk in unterschiedliche Kollisionsdomänen zu unterteilen. Und die transparente Brücke (heutzutage Switch) war geboren. Eine transparente Brücke ist ein „Store & forward“ device (packet switching) (→ Layer 2 Switches)

Aufgrund der Einschränkungen in der Dimension der Netze und den Übertragungsraten versuchte man relativ bald, diese Netzwerkarchitektur weiter zu verbessern. Die Resultate daraus waren 10Base-T und 10Base-F, die mit neuen Verkabelungsstandards (TP und Glasfaser) die bisherigen Schranken nach oben hoben. Beide Verkabelungsarten erlauben nicht nur eine höhere Übertragungsrate und sind weniger störungsanfällig als Koaxialkabel (zwei Drähte werden zwar auch gestört, die Relation der Signale auf beiden Leitungen bleibt jedoch gleich!), sie erlauben auch eine wesentliche Erweiterung der Netze im physikalischen Sinne. TP-Netze werden generell in Sterntopologie aufgebaut (analog zu den Telefonnetzen). Obwohl bei 10BaseT ein Hub verwendet wird, ist das Übertragungsverfahren immer noch halbduplex. Der Hub simuliert „Collision in a Box“. Durch die Verwendung von aktiven Netzwerkkomponenten wie Hubs, Switches oder Bridges bleibt eine Kollision ein weitgehend lokales Phänomen.

Ein andere was das VLAN: Durch dieses konnten mehrere Netzwerke mit ein und denselben physikalischen Leitungen realisiert werden.

High Speed Ethernet: Die letzte Version des IEEE 802.3 spezifiziert 10 – 1000 mbit. Und der tbit-Ethernet steht in den Startlöchern, Full Dublex Ethernet, Auto-Negotiation, Flusskontrolle. Das Ethernet ist dennoch abwärtskompatibel zum alten CSMA/CD HalbDublex Ethernet.

Full-Duplex: Durch den aktuellen Standard ist nun auch Vollduplex möglich. Die Netzwerkstation kann sofort eine Transmission starten und die Leitung ausnutzen, und kann zur gleichen Zeit Daten auf der anderen Leitung empfangen.

Flow-Control: Die Geschwindigkeitsanforderungen für Switches sind sehr hoch (speziell im Vollduplex Modus. Daher können auch leistungsfähige Switches an Ihre Leistungsgrenzen gelangen. Layer 4 Flowcontrol ist nicht leistungsfähig genug → MAC-basierendes Layer 2 Flow Control (Pause Kommando → Die Station stoppt das senden von Frames für eine im MAC-Controll-Parameter-Field angegebene Zeit [vielfaches der Slot-Zeit])

Bedarf nach höheren Geschwindigkeiten benötigten bessere Codierung:

- 10 mbit/s: Manchester Codierung
- Fast Ethernet (100 mbit/s): 4B/5B Block Code
- Gigabit Ethernet (1000 mbit/s): 8B/10B Block Code

PLS: fungiert als abstrakter Layer zwischen MAC Layer und Physikalischen Layer

PCS: Heutzutage wird das Coding mittels eines Medienabhängigen Physical Coding Sublayers unter dem MII durchgeführt. Encapsuliert das MAC-Frame, Encodet jeweils 4B/5B oder 8B/10B

MII: Media Independent Interface: MII ist ein Interface zwischen MAC Layer und den Physikalischen Layer: Versteckt Coding Angelegenheiten vor dem MAC layer, Oft eine mechanische Verbindung zum Physikalischen medium, Bringt eine einheitliche Spezifikation für alle physikalischen Medien, Unterstützt mehrere Datenraten (10, 100, 1000 mbit/s)

Siehe: [9-22/24](#)

Trunking: Auf Trunkleitungen zwischen Layer 2 Switches ist FullDuplex möglich (zb. 200 mbit/s mit Fast Ethernet)

Autonegotiation: Autonegotiation erlaubt zwei 100BaseT Devices Informationen über ihre Kapazitäten auszutauschen (Signalrate, CSMA/CD oder Full-Duplex). Sie erhalten dieses Signal durch die Link-Integrity-Test-Pulse-Sequence. (Eine normale Link-Pulse Technik ist bei 10BaseT bereits verfügbar) Somit können 100Base T Stationen 10 BaseT Stationen erkennen. Werden nur beim Verbindungsaufbau gesendet. 100BaseT Stationen können sich anhand der Message die größte Anzahl an Features ausmachen.

4B/5B: Code-Effizienz: 4/5 → 80% (Manchester Code: 50%) (Es werden 4bit in 5bit codiert. Dadurch wird es leicht Fehler zu erkennen. Auch die Synchronisation beim Empfänger wird leichter, da leicht definiert werden kann, wie viele Nullen hintereinander gesendet werden.)

8B/10B: Code-Effizienz: 8/10 → 80% (Manchester Code: 50%) (Es werden 8bit in 10bit codiert. Zu den Vorteilen wie bei 4B5B kommt noch dazu das jeweils 8bit invertiert und nicht invertiert codiert werden können. Dadurch wird der Gleichanteil der Übertragung vollständig eliminiert.)

32) Erklären Sie die prinzipielle Methode des Bridgings und des Routings. Auf welchem Layer des OSI Modells, mit welchen Adressen arbeitet Bridging? Auf welchem Layer des OSI Modells, mit welchen Adressen arbeitet Routing? Erläutern Sie die Funktionsweise einer transparenten Brücke im Detail (3 Entscheidungen bezüglich Weiterleiten von Rahmen, Aging, ohne Betrachtung der Spanning Tree Methode). Geben Sie Vor- und Nachteile von Bridging bzw. Routing an.

Bridge: Packet Switch auf OSI Layer 2 implementiert: Benutzt zur Weiterleitung MAC Adressen

Router. Packet Switch auf OSI Layer 3 implementiert: Benutzt zur Weiterleitung L3 Adressen.

Transparent Bridging:

Die Brücke ist unsichtbar für die Endsysteme. Diese glauben weiterhin, dass Sie sich auf einem Datenbus befinden. Die Brücke benutzt Layer 2 MAC Adressen um zu entscheiden ob ein Frame weitergeleitet wird oder nicht. Die gesamten MAC Adressen werden in einer Bridging-Tabelle registriert. (statisch oder dynamisch)

Heutige Switches sind schnelle transparente Brücken.

Im Fall von **dynamischen Bridging** sorgt ein Aging Mechanismus dafür, dass nicht mehr vorhandene MAC Adressen gelöscht werden und an einer anderen Stelle im Lan wieder eingebunden werden können. (5 min).

Auf Grund der **Transparenz** muss die Brücke jedes Frame empfangen und verarbeiten. Weiters ist Flow Control daher ebenfalls nicht möglich.

Die Bridge splittet ein Netzwerk in **mehrere Segmente**, die alle eine eigene Kollisionsdomäne besitzen. Eine Kollision an einem Segment wird daher von den anderen nicht bemerkt. Es bleibt jedoch **eine Broadcast Domäne**.

3 Entscheidungen der Weiterleitung:

Filterung: wenn die Destination im selben Lan-Segment ist wo das Frame auch empfangen wurde wird es gefiltert („Es kommt auch ohne Weiterleitung an“)

Forwarding: Die Destination befindet sich nicht im selben Lan-Segment wo das Frame herkommt („Es muss weitergeleitet werden“)

Flooding: Während dem Lernprozess ist der Bridge die Empfangsadresse unbekannt → somit muss das Frame in jedem Fall auf allen Ports weitergeleitet werden.

Frames mit Multicast / Broadcast Adresse werden ebenfalls in jedem Fall weitergeleitet.

Lernprozess: Allgemein gesagt lernt die Brücke durch den Datenverkehr bzw. Absenderadressen an welchem Port sich welche MAC-Adresse befindet → sobald Sie dies weiß kann sie die Frames korrekt weiterleiten.

Bei Broadcaststorms kann es bei gewissen Anordnungen (parallele Leitungen) der Bridges zu einem Endlosen Kreis kommen, welcher früher oder später das Netzwerk lahm legen wird. Um dies zu Vermeiden hat man das Spanning Tree Protocol entwickelt.

[Siehe Skizze 09-8](#)

Router:

Router basieren auf Layer 3 Adressen und Protokollen
Diese sind strukturierter als Layer 2 Adressen (Host, Subnet)
Hardware unabhängig
Identifizieren ein bestimmtes Endsystem in einem Subnet

Erfordernisse

Die Endsysteme müssen den Router kennen
Bei Ortswechsel muss das Endsystem die Adresse anpassen
Um die Routing Tabellen korrekt zu halten muss der Router mittels Routing Protokollen Informationen austauschen.

Fakten:

Endsysteme leiten die Datenpakete direkt zum Router mittels dessen MAC-Adresse. Der Router muss daher nur diese Pakete verarbeiten.
Der Transport in einem Netzwerksegment erfolgt weiterhin über Layer 2
Flow Control ist möglich
Broadcast / Multicast Pakete werden durch den Router geblockt
Unabhängig von Layer 2

Router können wiederkehrende oder parallele Pfade benutzen

Siehe Skizze 09- 35-39

Vorteile/Nachteile

Bridging

- + Benutzen nur MAC Adresse
- + Unsichtbar für Endsysteme
- Müssen jedes Frame verarbeiten
- Anzahl der Einträge → Anzahl der Devices im Netzwerk
- Spanning Tree – Eliminiert redundante Leitungen
- Keine Flusskontrolle
- Broadcast Domain -keine LAN/WAN Kopplung
- STP hat evtl. nicht optimalen Weg
- + Schneller als Routing
- + Ortswechsel leicht möglich

Routing

- Brauchen strukturierte Layer 3 Adresse
- Endsystem müssen den Router kennen
- + Verabreiten nur Frames die an sie adressiert sind.
- + Anzahl der Einträge → Anzahl der Subnets
- + Redundante Leitungen möglich
- + Flusskontrolle möglich
- + Behindert LAN nicht
- + Router hat immer den richtigen Weg
- Langsamer als Bridging
- Ortswechsel benötigt Änderung der L3 Adres.

33) Was versteht man unter Broadcast Storm im Zusammenhang mit „Transparent Bridging“? Wie wird er verhindert? Erklären Sie Aufgabe und Funktionsweise der Spanning Tree Methode im Detail.

Ein Broadcast ist eine Rundsendung von einer Station an alle anderen. Broadcaststürme entstehen am häufigsten auf der Sicherungsschicht, da hier viele Pakete der unterschiedlichen Netzwerkprotokolle nebeneinander existieren. In der Regel entstehen Broadcaststürme durch Konfigurations- oder Softwarefehler, wenn viele der angeschlossenen Stationen auf einen Broadcast antworten. Nachdem die Brücken ja für alle Schichten über 2b nicht sichtbar sind, kann auch eine andere Schicht einen solchen Sturm bemerken und ihm entgegenwirken. Der Broadcast Storm tritt auf wenn in einem vorhandenen Netzwerk mehr als eine Leitung zwischen 2 LAN Segmenten vorliegt. Frames mit unbekannter Destination kreisen somit „ewig“ zwischen den Stationen hin und her und stopfen kontinuierlich den Bufferspeicher der Bridge mit Daten, bis dieser schließlich voll ist. Um dies zu verhindern hat man das Spanning Tree Protocol entwickelt.

Spanning Tree Protocol: Spanning-Tree ist ein Verfahren zur Schleifenunterdrückung in brückengekoppelten Netzwerken. Bei diesem Verfahren werden physikalisch redundante Netzwerkstrukturen ermittelt und in einer zyklensfreien Struktur abgebildet. Diese Maßnahme reduziert die aktiven Verbindungswege einer beliebig vermaschten Netzwerkstruktur zu einer Baumtopologie (daher die Bezeichnung Spanning Tree, SPT). Mathematisch betrachtet ist eine Baumstruktur so geartet, dass alle vernetzten Punkte nur durch einen Weg miteinander verbunden sind. Bei einem Ausfall ist diese Verbindung dann unterbrochen. Mit einem bestimmten Algorithmus können aber zusätzliche Verbindungen geschaffen werden, die nur dann für den Verkehr freigegeben werden, wenn eine Station mit sonst keiner Verbindung erreicht werden kann. Das heißt, dass die Eindeutigkeit der Wege erhalten bleibt. Außerdem sind alle vernetzten Punkte von allen anderen vernetzten Punkten aus erreichbar, zudem gibt es zwischen zwei beliebigen vernetzten Punkten keine Zyklen.

Parameter für das STP

Bridge ID: Kombination von MAC-Adresse und Prioritätsnummer (normal: niedrigste MAC → höchste Priorität) – kann allerdings vom Admin verändert werden.

Port Kosten: Abstrakte Kosten um das Lokale Interface zu benutzen
Indirekt proportional zur Übertragungsrate (Default: 1000 /Übertragung in Mbit/s)
Kann vom Admin verändert werden,

Port Identifier: Kombination von Port MAC Adresse und Prioritätsnummer, Vom Administrator konfiguriert.

Algorithmus

Die Root-Bridge wird ausgewählt

Nach dem Start werden alle Ports in den Blockierzustand versetzt und jede Brücke versucht die Root-Brücke zu werden indem Sie die entsprechende BPDU Message sendet.

Blockieren bedeutet: Die Frames der Entstationen werden nicht empfangen und nicht weitergeleitet, BPDU Frames werden jedoch empfangen und verarbeitet.

Mittels der BPDU teilt die Bridge mit welche Bridge als Rootbridge angesehen wird, die Kosten zu seiner Root Bridge und die eigene Bridge ID und Port ID.

Die Bridge mit der niedrigsten ID wird zur Root Bridge.

Nach der Auswahl der Root Bridge steht es nur dieser zu die Configurations BPDU Message zu senden.

Strategie: Empfängt die Bridge eine BPDU Message mit höhere ID, so sendet sie die eingene ID und die andere Brücke sollte aufhören, Empfängt die Bridge eine BPDU Message mit niedrigerer ID so hat sie ihrerseits aufzuhören.

Die Route-Ports werden ausgewählt

Jetzt legen die Bridges fest welche Ports die niedrigen Kosten zur Root Bridge haben → Root Port

Es wird für jedes LAN Segment eine festgelegt Bridge ausgewählt

Gleichwertig zur Root Bridge wird eine Designated Bridge für jedes LAN Segment ausgewählt

Die festgelegten und Root Ports werden auf Weiterleiten geschaltet

Alle anderen Ports werden geblockt.

Es wird eine festgelegte Leitung zu jedem Lan-„Ende“ (Zweig) gelegt.

Error Detection: Alle 1-10 Sekunden wird eine Hello Message gesendet. Kommt nach 2 Versuchen keine Antwort gibt es entweder einen Rood Bridge Fehler oder einen Designated Bridge Fehler → In beiden Fällen übernimmt eine andere Bridge diese Funktion.

34) Was versteht man unter Collison Domain und Broadcast Domain im Zusammenhang mit Ethernet und Transparent Bridging? Was ist L2 Switching (Ethernet Switching)? Was sind VLANs? Was ist Tagging? Zeigen Sie kurz den Wandel auf, den gebridgte LANs durch die neuen Technologien Fast Ethernet, Gigabit Ethernet genommen haben.

Eine Kollisionsdomäne bedeutet, dass dass alle Geräte auf einem Bus liegen. Versuchen nun 2 Devices gleichzeitig zu senden kommt es zur Kollision. Da dieser Vorfall mit der Größe eines Netzwerk direkt proportional ansteigt, müssen LAN's in mehrere Collision-Domains unterteilt werden. Die Kollisionen in einem Segment werden so vom anderen Segment nicht wahrgenommen.

Ein Ethernet mit Bridges bleibt jedoch eine Broadcast Domäne. Broadcast Frames werden immer weitergeleitet, da die Endsysteme das LAN noch immer als eine einzige Domäne sehen und demnach mit einem Broadcast alle Stationen erreichen wollen.

Layer 2 Swichting → Transparent Bridging:

Die Brücke ist unsichtbar für die Endsysteme. Diese glauben weiterhin, dass Sie sich auf einem Datenbus befinden. Die Brücke benutzt Layer 2 MAC Adressen um zu entscheiden ob ein Frame weitergeleitet wird oder nicht. Die gesamten MAC Adressen werden in einer Bridging-Tabelle registriert. (statisch oder dynamisch)

VLAN's: Heutzutage arbeiten Arbeitsgruppen über weitere Ebenen auseinander, haben jedoch ein Bedürfnis nach einem „eigenen Netzwerk“ und sollten Daher von den anderen Gruppen getrennt werden. (Security). Auch Broadcasts sollen nur von der eigenen Arbeitsgruppe sichtbar sein. Weiters muss das VLAN flexibel in Bezug auf die Lokalisation des Benutzers sein.

Idee des VLAN: Multiplexing von verschiedenen LANs auf der selben Infrastruktur.
 Separate Bridging/Switching Tabelle für jedes VLAN
 Separates Broadcast Handling für jedes VLAN
 Separates Spanning Tree Protokoll für jedes VLAN

VLAN Zuordnungen

Port-Based: jeder Port eines Switches wird fix einem VLAN zugeordnet (unflexibel bei Ortswechsel)

MAC-Based: jede MAC-Adresse wird einem VLAN zugeordnet: Erlaubt Integration von älteren Komponenten und ist flexibel bei Ortswechsel

Protocol-based: Eine Station könnte auch Benutzer in verschiedenen VLAN's sein

Tagging: Switches müssen via VLAN-Trunks verbunden sein und jedes Frame mit einem Identifier, der angibt in welches VLAN das Frame gehört.

Kommunikation zwischen VLAN's: Switches dürfen keine Frames zwischen verschiedenen VLANs hin und herschicken. Aus diesem Grund müssen Endsysteme auf Router zurückgreifen. Diese können entweder Bestandteil mehrerer VLAN's sein oder in der Lage sein Tags der Frames zu ändern.

Entwicklung des LAN's: Anfangs ging man von einer reinen Bus-Topologie aus, welche man auf Grund der Begrenzung mittels eines Repeaters verlängern konnte. Der nächste Schritt war Multiport Repeater einzusetzen. Damit hatte man die Möglichkeit von einem zentralen Punkt aus sternförmig zu verkabeln. (10BaseT) – Diese Multiport Repeater werden „Hub“ genannt. Das Netzwerk hat jedoch noch immer eine zentrale Kollisionsdomäne und es kann nur Halbduplex gesendet werden. (CSMA/CD in da box). Der nächste logische Schritt war demnach das Netzwerk in unterschiedliche Kollisionsdomänen zu unterteilen. Und die transparente Brücke (heutzutage Switch) war geboren.

High Speed Ethernet: Die letzte Version des IEEE 802.3 spezifiziert 10 – 1000 mbit. Und der tbit-Ethernet steht in den Startlöchern, Full Duplex Ethernet, Auto-Negotiation, Flusskontrolle. Das Ethernet ist dennoch abwärtskompatibel zum alten CSMA/CD Halbduplex Ethernet.

Switches müssen nun in der Lage sein Ethernet's mit 10 Mbit/s, 100 Mbit/s oder 1000 Mbit/s miteinander zu verbinden.

Bedarf nach höheren Geschwindigkeiten benötigten bessere Codierung:

- 10 mbit/s: Manchester Codierung
- Fast Ethernet (100 mbit/s): 4B/5B Block Code
- Gigabit Ethernet (1000 mbit/s): 8B/10B Block Code

35) Was passiert, wenn zwei Endgeräte (Host A auf LAN 1 mit MAC A und Host B auf LAN 2 mit MAC B) erstmalig miteinander kommunizieren. LAN 1 und LAN 2 sind über eine Bridge B (Port 1 führt zu LAN 1 und Port 2 zu LAN 2) verbunden. Sie können davon ausgehen, dass Spanning Tree bereits eingeschwungen ist und die Bridging Tabelle zu Beginn leer ist. Schildern Sie die zeitlichen Abläufe an Hand der ersten drei Rahmen, die zwischen Host A und Host B ausgetauscht werden (also A->B, B->A, A->B).

1) MAC A sendet ein Frame mit SA: A und DA: B an den Host B

- 2) Die Bridge lernt dadurch, dass sich auf Port 1 die MAC-Adresse A befindet, da er den Host B jedoch nicht kennt muss er die Message flooden.
- 3) Host B Empfängt das Frame und sendet beispielsweise eine ACK Message mit SA: B und DA: A zurück. Von diesem Frame erfährt die Bridge nun, dass sich Host B mit der MAC Adresse B auf Port 2 befindet, da er vorher gelernt hat, dass sich MAC A auf Port 1 befindet leitet er das Frame dorthin weiter

Nach diesen 2 Schritten ist in diesem Fall die Bridging Tabelle bereits vollständig, dh. beim versenden aller weiterer Rahmen weiß die Bridge genau was sie zu tun hat.

Fragenbereich 4 (Kapitel: IP Technology Basics, IP Details, TCP):

36) Charakterisieren Sie kurz die wesentlichen Eigenschaften des IP-Protokolls (Network Type (Packet oder Circuit Switching) / Service Type (CO oder CL), beteiligte Komponenten (IP Host, Router), Forwarding Prinzip). Wozu ist eine Begrenzung der Lebensdauer eines IP-Datagramms notwendig, wie wird sie realisiert? Wann wird IP Fragmentierung vorgenommen, wie wird sie realisiert? Was versteht man unter TOS und wie kann dieses umgesetzt werden. Über welchen Bereich erstreckt sich die Checksum?

Fakten:

Die IP Technologie basiert auf Packet Switching und ist Connectionless (Best effort Service).
Das Endsystem wird IP Host genannt
Die IP Adresse ist eine strukturierte Layer 3 Adresse.

IP Protocol: OSI Layer 3 Protokoll mit Datagramm Service
Die Pakete werden vom Sender über ein oder mehrere Netzwerke zum Empfänger gesendet.
Der Empfang in der richtigen Reihenfolge kann nicht garantiert werden.
IP Datagramme werden im Layer 2 enkapsuliert. (Key-Feature des von TCP/IP → Unabhängigkeit vom physikalischen Netzwerk)

Funktionen des IP Protokolls:

Mechanismus für Packet Forwarding basierend auf einer Netzwerkadresse,
Error Detection,
Fragmentation und Zusammenführung von Datagrammen,
Mechanismus um die Lebenszeit eines Datagramms zu begrenzen

Weiters ist der Mechanismus um die Lebenszeit eines Datagramms zu verhindern notwendig, um die Endlose Zirkulation eines Frames aufgrund falscher Routingtabellen zu verhindern.

Umsetzung: mittels des TTL Feldes im IP-Header

Limitiert die Lebenszeit eines Datagramms (Bereich 0-255 Sekunden)
Wird vom Sender auf einen Startwert von 32 oder 64 (üblich) gesetzt
Jeder Router, welchen das IP Datagramm durchläuft verringert das TTL Feld um die Wartezeit. (Wartezeit < 1 → 1)
Wenn das TTL Feld bei 0 ankommt, wird das Datagramm gelöscht

Die Fragmentierung ist notwendig wenn ein Datagramm ein Netzwerk mit kleiner Framelänge passieren muss.

Umsetzung: Fragment Offset

Gibt die Position des Fragments in Bezug auf den Beginn des Original-Datagramms an.
Offset ist immer ein vielfaches von 8
Das erste Fragment bzw. ein unfragmentiertes IP Datagramm bekommen das Offset 0

Fragmente mit derselben Empfangsadresse, Protokoll, Identifikation werden gemäß ihrer Offsets zum Original-IP-Datagramm zusammengesetzt.

Zusammensetzen:

Wird vom Empfänger vorgenommen, da Fragmente unterschiedliche Pfade durchlaufen können.

Der Pufferspeicher muss somit ebenfalls vom Empfänger bereitgestellt werden.

Einige Fragmente könnten evtl. nicht ankommen (Natur von IP)

Außerdem müssen Maßnahmen getroffen werden um rechtzeitig Pufferspeicher freizumachen wenn ein IP-Datagramm nicht rekonstruiert werden kann. →

Das erste Offset starten einen Timer → Innerhalb dieser Zeit können die restlichen Fragmente ankommen → Tun sie es nicht, werden die IP-Datagramm-Teile verworfen. → Der Timer limitiert somit die Reassemblierungszeit.

Forwarding Prinzip: Es wird ein Pfad gewählt, über den IP Datagramme gesendet werden. Für die Wahl des besten Weges sind sowohl IP Host (Default Router) als auch die Router (Direktes weiterleiten) verantwortlich. Router leiten die Datagramme gemäß ihrer Routing-Tabellen weiter

TOS: Alte Bedeutung: Priorität des Datagramms, Bevorzugte Netzwerk Charakteristik
In IPv4 wurde TOS recycled zum „Differentiated Service CodePoint (DSCP). Dieses wird nun benutzt um die Verkehrsklasse einer gegebenen Kommunikation zwischen 2 IP-Hosts zu bestimmen. → Unerlässlich für IP QoS (Quality of Service) zb IP Telefonie

Checksum: Die Checksumme erstreckt sich nur über den IP-Header.

37) Welche Aufgaben und welche Struktur haben IP-Adressen? Was bringt Subnetzadressierung und wie wird sie bewerkstelligt? Was muß in den Endgeräten konfiguriert werden, um IP Kommunikation zu ermöglichen? Welche Sichtweise (lokal oder global) haben die Endgeräte (IP Hosts) dadurch? Was muß in den Routern konfiguriert werden, um IP Kommunikation zu ermöglichen. Welche Sichtweise (lokal oder global) haben die Router (IP Gateways oder IP Router) dadurch? Worauf muß bei der Adressierung in Falle von Classful Routing achten? Welche Möglichkeiten der Adressierung hat man im Falle von Classless Routing (Stichwort VLSM und Supernetting)?

Definition: Eine IP-Adresse identifiziert einen Rechner in einem Netzwerk auf eindeutige Art und Weise. Nachdem es aber in einem Netzwerk viele verschiedene Arten von Architekturen und Rechnersystemen gibt, benötigt man ein Adressierschema, das von den zugrunde liegenden Hardwareadressen unabhängig ist. Außerdem sind 32-Bit Adressen meistens nicht so gut im Kopf zu behalten, worauf man eine alphanumerische Vereinfachung einführt (www.xxx.at,...). Durch diese einheitliche Adressierung wird die Illusion eines großen nahtlosen Netzwerkes geschaffen.

Je nach Einsatzgebiet gibt es 5 unterschiedliche IP-Adressklassen:

- 1) Die Klassen A, B und C unterscheiden sich durch eine unterschiedliche Länge der Netz- und Nutzeridentifikationsfelder. Bei Klasse A-Adressen wird der erste Quad (das erste Byte) vorgegeben, die restlichen drei Quads sind frei wählbar (um die 16 Millionen verschiedene Adressen). Das most-significant-Bit ist dabei immer Null.
- 2) Die Klasse B-Adressen: Die ersten zwei Quads sind vorgegeben, die anderen können frei gewählt werden. Die zwei höchsten Bits sind dabei Eins und Null. Dadurch erhält man maximal 65.536 Adressen. Die HostID (lokale Adresse) wird nach unten hin immer kleiner, während die NetID immer größer werden.
- 3) Bei Klasse-C-Adressen werden die ersten drei Quads vorgegeben und nur mehr das letzte Byte kann man frei vergeben. Man erhält 256 Adressen. Die Anfangskombination ist „110“.
- 4) Klasse-D-Adressen sind für Multicast-Adressen vorbehalten.
- 5) Klasse-E-Adressen sind reservierte Adressen für zukünftige Anwendungen.

Subnetzmaske: Nachdem der Adressraum langsam zur Neige ging, viele IP-Adressen aber ungenutzt blieben, hat man eine neue zusätzliche Strukturierung eingeführt, die die Adressbereiche effizienter ausnutzt. Eine Subnetzmaske gibt an, welcher Teil der IP-Adresse das Subnetz beschreibt. Bei einer Subnetzmaske von 255.255.255.0 geben die binären Einsen der 255er an, das die ersten drei Quads das Subnetz angeben. Im Detail wird das Verhältnis von NetID zu HostID geändert (im Subnetz!), mit dem Ziel, selbstständig eine Adressvergabe durchführen zu können. Dabei ändert sich die NetID nicht, die HostID (die ja den Zielrechner angibt) wird allerdings verringert, weil zusätzlich die SubnetID in das Adressfeld eingefügt wird.

Den Endgeräten muss gesagt werden wohin sie die abgehenden Pakete schicken sollen, dh. wenn die gesuchten IP-Adressen im lokalen Netz sind, werden die Pakete direkt geschickt, bei unbekannt Adressen werden über eine default-router. Das muss allerdings konfiguriert werden, da normalerweise Pakete mit unbekannt Adressen weggeworfen werden.

Bei IP-Routern muss konfiguriert werden wohin sie ankommende Pakete weiterleiten sollen. Auch hier wird meistens konfiguriert, dass unbekannt Adressen an einen default-router weiter geschickt werden. Dadurch brauchen die Geräte nur lokale Routing-Tabellen. Es wäre auch unmöglich, dass jeder kleine Router die komplette Struktur des Internets kennt. Die benötigten Routing-Tabellen können entweder statisch programmiert werden, oder dynamisch ermittelt werden. Bei dynamischen Tabellen schicken die Router regelmäßig Information über ihre Sichtweise des Netzes an ihre Nachbarn.

Routing:

Classfull: Die Routing-Protokolle RIP oder IGRP können keine Information über Subnetz-Masken in Updates tragen. Dadurch muss die Subnetz-Maske im gesamten Gebiet konstant sein, es ist nicht möglich eine variable Länge zu verwenden. Wenn ein Update geschickt wird mit einer anderen als der Subnetz-Nummer werden nur die Klasse A, B oder C Nummern veröffentlicht.

Classless: Die Routing-Protokolle RIPv2, OSPF oder eIGRP können Information über Subnetz-Masken in Updates tragen. Dadurch wird es möglich Subnetz-Masken mit variabler Länge zu verwenden (VSLM). Route Summeraziation kann auf alle Adressen angewendet werden.

Supernetting: Die aktuelle Subnetz-Maske ist kleiner als die natürliche Subnetz-Maske der gegebenen Klasse.

38) Was passiert, wenn zwei Endgeräte (Host A mit IP Adresse 10.1.0.1/16 und MAC A auf LAN 1 (IP Subnet 10.1.0.0) und Host B mit IP Adresse 20.2.0.2/16 und MAC B auf LAN 2 (IP Subnet 20.2.0.0)) erstmalig miteinander kommunizieren. LAN 1 und LAN 2 sind über einen Router R (IP Adressen 10.1.0.254 und 20.2.0.254) verbunden (Port 1 führt zu LAN 1 und Port 2 zu LAN 2). Sie können davon ausgehen, dass die Routing Tabelle vollständig ist, aber alle ARP Caches zu Beginn leer sind. Schildern Sie die notwendigen Konfigurationen in den Endsystemen und die zeitlichen Abläufe an Hand der ersten drei Nutz-Rahmen, die zwischen Host A und Host B ausgetauscht werden (also A->B, B->A, A->B).

Host A sendet einen ARP-Request nach der MAC des Router (IP muss konfiguriert sein). Der Router antwortet ihm nun und teilt dem Host A seine MAC-Adresse mit (MAC R), welcher dieser sich speichert. Jetzt sendet Host A ein Datenpaket mit der Sender-Adresse 10.1.0.1/16, Destination-Adresse 20.2.0.2/16, MAC-Sender (MAC A) sowie die Empfänger MAC (MAC R) → Das Datenpaket geht folglich an den Router. Der Router seinerseits macht nun einen ARP Request der MAC Adresse für die IP (20.2.0.2/16 auf dem Netzwerk 2 (Port 2)). Host B antwortet dem Router, dass die MAC-Adresse von 20.2.0.2/16 MAC Adresse B ist. Jetzt kann der Router das IP Datagramm (IP Sender: 10.1.0.1/16, IP Empfänger: 20.2.0.2/16, MAC-Sender R, MAC-Empfänger: B) weiterleiten.

Nach diesem Vorgang wissen alle Beteiligten Stationen über die MAC-Adressen der jeweils anderen Station Bescheid.

39) Wozu dient das ARP-Protokoll? Beschreiben Sie es im Detail. Wann und wie verwendet ein IP Host das ARP Protokoll in normalen Situationen (d.h. kein proxy ARP)? Was ist Proxy ARP und wozu kann es verwendet werden?

ARP: ARP (Adress Resolution Protokoll) dient dazu, die MAC-Adressen in die zugehörigen IP-Adressen umzuwandeln, damit überhaupt eine Kommunikation auf der Vermittlungsschicht mittels des IP-Protokolls stattfinden kann. Das ARP-Protokoll legt zu diesem Zweck Mapping-Tabellen an, die die MAC-Adressen den Netzwerkadressen zuordnen. Vor dem Verbindungsaufbau über das Ethernet fragt IP bei ARP nach der Ethernet-Adresse der zugehörigen Ziel-Internet-Adresse an. ARP vergleicht seine Adresstabellen (auch ARP-Tabellen oder Internet-nach-Ethernet-Translation-Tabellen genannt) mit der Anfrage.

Hat ARP keinen Eintrag in seiner Tabelle, so wird über eine Anfrage an alle Netzknoten (Broadcast) die Ethernet-Adresse der zugehörigen Internet-Adresse erfragt. Nur Netzknoten mit einem Eintrag zu dieser IP-Adresse antworten auf die Anfrage. Die Antwort auf den ARP-Broadcast wird in der ARP-Adresstabelle gespeichert, die aber nach einem speziellen Abfragealgorithmus gespeichert werden müssen, weil die beiden Adressarten unterschiedliche Längen haben.

Eine vormals effiziente Methode, den Adressraum in einem Netz besser zu nutzen, ist das proxy-ARP. Es wird dazu verwendet, zwei unterschiedliche Netze über eine einzige netID anzusprechen (sei es aus Sicherheitsgründen oder zwecks der Performance). Die Stationen in den unterschiedlichen Teilen des Netzes wissen nichts über den Proxy. Wenn also ein Rechner etwas zu einem in einem anderen Teil gelegen Rechner senden will, schickt er die Pakete mit der Zieladresse zu dem Proxy (den er nicht sieht). Der Proxy tritt nun beim Zielrechner als Rechner1 auf und sendet ihm die Pakete weiter. Beide Stationen sind der Meinung, sich in einem homogenen Netz zu befinden. Diese Methode ist zwar veraltet aber dennoch für die Absicherung von ganzen Netzteilen in Verwendung (Firewalls). Die heute üblichere Methode für die effiziente Nutzung des Adressraumes eines Netzwerkes ist das Subnetting.

40) Wozu dient das ICMP-Protokoll? Wie funktioniert es? Was lässt sich damit signalisieren bzw. realisieren?

Warum ICMP (Internet Control Message Protocol)

Der Datagramm Service von IP kann keine Garantie oder Bestätigung für die Lieferung des Datagramms geben.

ICMP generiert eine Error Message um die Zuverlässigkeit zu erhöhen und um Informationen über Fehler und Packetverlust im Netzwerk bereitzustellen.

ICMP erlaubt Informationen über die Fehlerursache zu finden.

ICMP muss von jeder IP-Station unterstützt werden, es können aber verschiedene Implementierungen auftreten.

Arbeitsweise: Die IP-Station (z.B. Router), die ein Übertragungsproblem feststellt generiert die ICMP Message. Diese wird zur Ursprünglichen Station geschickt (Versender des Originalen IP Pakets). Die Messages werden als normale IP Datagramme gesendet (ICMP Code + Header im Data-Field). Die Analyse dieser Message gibt dem Administrator die Möglichkeit, die Fehlerursache zu finden.

Sollte das IP Datagramm mit der ICMP Message verloren gehen wird **keine** neue Message gebildet um keine ICMP Lawinen heraufzubeschwören.

Message Format: Type: General Message Type
Code: Detailed Specification
Checksum: Kalkuliert über ICMP Header und Daten
Extension Field: Wird nur für spezielle Messages benutzt
Internet Header + 64 bits Originales Datagramm

Types: Type 3 (z.B. Network/Host/Protocol/Port) unreachable)
 Type 5 (Wenn ein Router einen besseren Weg über das Netz weiß wird er den Sender dementsprechend informieren)

41) Charakterisieren Sie kurz die grundlegendsten Eigenschaften von TCP. Wozu dienen weitere Portnummern und Sockets beim TCP-Protokoll? Wie werden Ports in Client Server Beziehungen verwendet? Wie wird der Verbindungsaufbau und Verbindungsabbau bei TCP vorgenommen? Wie werden die Sequence- und Acknowledge Nummern verwendet? Welche Rolle haben die Flags. Wie erfolgt das Error Recovery? Welche Besonderheit gibt es bei Handhabung der Timeouts? Über welchen Bereich erstreckt sich die Checksum? Was ist UDP?

Die Aufgabe des TCP-Protokolls (verbindungsorientierter Dienst, Ende-zu-Ende-Verbindungen, Verbindungsaufbau über 3-way-handshake) ist die Adressierung der Dienste der Anwendungsschicht über Port- und Socketnummern. Zusätzlich hat es die Aufgabe, die Zuverlässigkeit von IP zu erhöhen. Dazu werden Mechanismen wie Erkennung und Korrektur von Fehlern, Flusskontrolle, Neuordnung der Segmente, falls bei der Übertragung die Reihenfolge vertauscht wurde, und Entfernung von doppelten Segmenten verwendet. Erreicht wird das durch Sequenznummern zur Kennzeichnung der Segmente, Quittierung (piggy bagged, Fullduplex), wenn Segmente in der richtigen Reihenfolge empfangen wurden, und Wiederholung von Segmenten aufgrund eines Timeouts.

Um eindeutige Verbindungen zwischen zwei Rechnern aufbauen zu können, verwendet TCP so genannte Portnummern. Wenn aber ein Rechner mehrere Dienste über denselben Port in Anspruch nehmen will, müssen diese simultanen Verbindungen zusätzlich gekennzeichnet werden. Das geschieht über Socketnummern (zusammengesetzt aus der IP-Adresse und dem jeweiligen Port). TCP fungiert also gleichzeitig als Multiplexer und Demultiplexer. TCP unterteilt die Port in zwei verschiedene Gruppen: well-known Ports (Portnummern von 0 bis 1023) und registered Ports (Portnummern ab 1024). Gemeinsam mit dem Socket-Konzept können auf einem well-known Port mehrere simultane Verbindungen aufgebaut werden. Wenn der Client eine Verbindung zu einem speziellen Dienst des Servers aufbaut, wählt der Client einen freien Source-Port und wird mit einem well-known-Port des Servers verbunden. Einige Dienste können auch dynamisch zugewiesene Portnummern verwenden (FTP, ...).

Der Verbindungsaufbau unter TCP läuft über das so genannte 3-Wege-Handshake ab, bei dem drei Nachrichten ausgetauscht werden. Ein dreifacher Austausch ist notwendig und hinreichend (Mathematiker haben das nachgewiesen!), um ungeachtet von Paketverlusten, Duplikaten und Verzögerungen eine eindeutige Vereinbarung sicherzustellen. Die Nachrichten, die ausgetauscht werden, bestehen aus einer Sequenznummer, die zufallsverteilt generiert wird, und einem Acknowledgement. Eine solche Nachricht, die für den Aufbau einer Verbindung herangezogen wird, nennt man auch Synchronisationssegment (SYN-Segment), während eine Nachricht mit dem Zweck, die Verbindung wieder abzubauen, Endsegment (FIN-Segment) genannt wird. Eine Verbindung wird also erst bei gegenseitiger Absprache auf- oder abgebaut. Die erste sendende Station generiert eine Sequenznummer, die an die zweite Station übermittelt wird. Das Acknowledgement ist noch nicht gesetzt. Beim Empfänger wird die ankommende Sequenznummer als Acknowledgement (um eins erhöht) gesetzt und eine neue Sequenznummer wird generiert. Beides wird wieder an die erste Station übermittelt, die mit einem Rücksenden des gesamten Pakets die Verbindung bestätigt. Das Resultat ist eine "synchronisierte Verbindung". Ab diesem Zeitpunkt können Daten übermittelt werden. Nachdem immer eine neue Sequenznummer generiert wird, können gleichzeitig mehrere Anwendungen eine Verbindung auf- und abbauen. In der Kommunikation werden Time-outs benutzt, um unerwünschte Verzögerungen zu vermeiden und den Datenfluss zu verbessern. Beispielsweise werden sie benutzt zur Bestimmung der maximalen Antwortzeit beim Pollen und bei der Adressierung, bevor eine Prozedur automatisch neu initiiert wird. TCP geht mit solchen Time-outs flexibler um als wir das beim Ethernet gehabt haben. Es geht zwar auch von einem fixen Startwert aus, richtet die Zeit aber dann je nach Verbindung immer neu ein. Bei einer langsameren Verbindung wird anhand der Zeit, die ein Paket benötigt, um vom Sender zum Empfänger und wieder retour zu wandern (Roundtime), die Zeit für das Time-out

festgelegt. Sollte ein Paket bei bestehender Verbindung länger brauchen, als der Timer es vorschreibt, wird es verworfen und neu gesendet. TCP optimiert dadurch den Durchsatz von selbst.

Die **Checksum** erstreckt sich im Header von Bit Null bis 15. Sie erstreckt sich über den Header, die Nutzdaten und über einen 12 Byte langen Pseudo-IP-Header, mit dem zwar fehlerlose aber falsch geleitete Pakete erkannt werden können.

Bis jetzt wurden lediglich einige grundlegende Aspekte von TCP betrachtet. Um in heutigen IP-Netzen werden allerdings weit mehr Funktionen gefordert als bisher betrachtet.

Eine sehr wichtige Funktion ist **Slow Start and Congestion Avoidance**. TCP kann damit selbstständig festlegen, wie viele Daten ins Netz geschickt werden. Dies geschieht bereits auf Senderseite – basierend auf der Größe des Fensters – und nicht erst, wenn es bei einem Empfänger zu einem Überlaufen kommt.

Fast Retransmit und Fast Recovery: Auch im Fall eines Neusendens muss die Time-out Zeit eingehalten werden. Das führt unweigerlich zu einer Verlangsamung des Netzes. Muss neu gesendet werden, wird sofort ein Time-out durchgeführt.

Delayed Acknowledgements: Wenn auf der Empfängerseite ein Paket ankommt, wird im Normalfall sofort ein ACK gesendet. Das führt auch zu unnötig viel zusätzlichem Verkehr auf dem Netz. Bei interaktiven Anwendungen herrscht meist ein reger Verkehr auf der Empfängerseite. Wenn also das ACK erst gesendet wird, wenn im Puffer des Senders ein zweiter Frame zur Übertragung bereitsteht, der das ACK piggy-backen kann, erreicht man damit eine wesentlich kleinere Netzlast. Die interaktive Applikation wartet damit jedes Mal 200ms, ob nicht vielleicht doch noch ein ACK mitreisen will, und sendet dann erst den nächsten Frame. In diesen 200ms hat die Applikation eventuell auch etwas zu senden. Wenn nicht, macht sich das ACK alleine auf den Weg.

UDP (User Datagram Protocol): ist einfacher als TCP, verbindungslos, Layer4 Service, wird in Situationen eingesetzt, wo Datagrammverlust nicht besonders kritisch ist bzw. wo die Implementierung gering sein muss. Deshalb ist es auch leichter zu implementieren. UDP verwendet die selben Portnummern wie TCP. Im UDP-Header empfinden sich Source- und Zieladresse, die Länge des UDP-datagramms und die Checksumme.

Portnummer: Allen Prozessen werden Portnummern zugeordnet, damit sie gleichzeitig ausgeführt werden können.

Socket: Ist die Kombination von IP-Adresse und Portnummer, somit einzigartig.

Verbindungsaufbau: "three way handshake": 1.request, 2.response, 3.response of response

Sequence Number: Position des 1. Oktetts von diesem Segment im Datenstrom.

Acknowledge Number: bestätigt die korrekte Ankunft von allen Oktetts bis zur Ack-Nummer minus 1 und zeigt auf die Nummer des nächsten Oktetts.

Besonderheiten d. Timeouts:

- high timeouts: daraus folgen "lange" Wartezeiten
- low timeouts: daraus folgen unnötige Übertragungswiederholungen

Checksum: beinhaltet den TCP-Header und Datenbereich und einen 12 Byte Pseudo-IP-Header (bestehend aus Source- und Destination-IP-Adresse, IP-Protocolltype und IP-Segmentlength). Der Pseudo-IP-Header erlaubt Fehlererkennung.

Sliding Window:

- rechte Ecke bewegt sich nach rechts: Empfänger bestätigt Daten und leert TCP-Buffer-Space
- linke Ecke bewegt sich nach rechts: Daten sind gesendet und bestätigt.
- rechte Ecke ! links: geht nicht, weil doppelte Acks gelöscht werden.

- linke Ecke links: darf nicht passieren (!shrinking Window)

TCP-Flow-Control wird durch dynamic windowing unter Verwendung vom Sliding Window Protokoll gemacht.

Slow Start: wenn es eine TCP Verbindung gibt, wird das Staufenster zuerst auf 1 Segment initialisiert. Bei Bestätigung verdoppelt usw. bis Stau auftritt) dann muss der Sender die Senderate verlangsamen. Slow Start reduziert die Senderate durch ! Congestion Avoidance (Sender Imposed Flow Control).

42) Wie wird die Flußkontrolle bei TCP durchgeführt? Beschreiben Sie diese im Detail.

Bei TCP ist die Fenstergröße bzw. das Window ein 16 Bit langes Feld im TCP-Header, das der Flusskontrolle zwischen Sender und Empfänger dient. Die Flusskontrolle basiert auf einer fortlaufenden Nummerierung der Datenbytes. Zu diesem Zweck teilt der Fenstermechanismus mit jedem Acknowledgement mit, wie groß der noch verfügbare Puffer im Empfangs-Knoten ist. Schickt der Receiver einen Wert gegen Null, wartet der Sender mit dem Fortsetzen bis er wieder ein Freizeichen bekommt. Somit ist gesichert, dass es auf Empfängerseite nicht zu einem Datenüberlauf (flooding) kommt!

Ja nach Größe des Fensters – der Empfänger kann in einem Acknowledgement die momentane Größe seines Puffers bekannt geben – kann der Sender die Paketgröße anpassen, um eine optimale Ausnutzung der Ressourcen zu gewährleisten. Sollte es aufgrund von zu hoher Netzauslastung zu einem Stau – und damit zu einem verspäteten Eintreffen der Pakete (... sind nicht verloren gegangen!) kommen, worauf hin alle Sender kein ACK erhalten und durch ein Timeout erneut senden, was die Situation natürlich weiter verschärft –, kann TCP durch Absenken der Datenrate zur Auflösung des Staus beitragen, wodurch die Lage nicht weiter verschärft wird.

43) Welche Rolle spielt prinzipiell das Bandwidth-Delay Produkt in ARQ Verfahren? Welche Auswirkungen auf das Sende- Fenster ergeben sich daraus? Gehen Sie auf die Performanceaspekte ein, die sich durch Slow Start und Congestion Avoidance bei TCP einstellen. Welche Rolle spielt dabei das Duplicate Ack? Schildern Sie diese Verfahren im Detail.

Die Performance jedes verbindungsorientierten Protokolls mit error-recovery (ARQ) ist von Haus aus in seiner Bandbreite begrenzt. Ein Optimum kann erreicht werden, wenn man Continuous RQ mit einer "sliding window" Technik verwendet. Hier muss aber das Fenster groß genug sein, um ein stoppen des Sendens zu verhindern. Groß genug bedeutet, dass die Serialisations- und Propagation-Verzögerungen abgedeckt werden. Allerdings kann das Sende- und Empfangsfenster auch durch den Speicherbereich limitiert sein.

Allerdings muss das Sendefenster groß genug sein, damit der Sender das komplette Übertragungsvolumen ausschöpfen kann. Diese kann durch Verzögerungen – die an Buffern entstehen –, begrenzte Übertragungsgeschwindigkeit und eine begrenzte Bandbreite erhöht werden. Somit kann das Übertragungsvolumen durch das Delay-Bandwith Product ausgedrückt werden.

Um die "**Pipe**" zwischen Sender und Empfänger mit Daten füllen zu können muss:

- die Fenstergröße –die vom Empfänger angeboten wird – größer sein als das Delay-Bandwith Produkt der Pipe
- die Fenstergröße \geq Kapazität der pipe (bits)
= bandwidth (bits/sec) · round-trip time (sec)

Eine sehr wichtige Funktion ist **Slow Start and Congestion Avoidance**. TCP kann damit selbstständig festlegen, wie viele Daten ins Netz geschickt werden. Dies geschieht bereits auf Senderseite - basierend auf der Größe des Fensters - und nicht erst, wenn es bei einem

Empfänger zu einem Überlaufen kommt. Slow Start (und Congestion Avoidance) sind in heutigen TCP-Implementationen vorgeschrieben. Slow Start erfordert von TCP eine Verwaltung eines zusätzlichen Fensters: congestion window (cwnd). Es gibt eine Regel, die stets eingehalten werden muss: Der Absender kann bis zum Minimum des Congestion-Fensters und des annoncierten Fensters übertragen.

Wird eine neue TCP-Verbindung hergestellt, so wird das Congestion Fenster mit einem Segment initialisiert. Immer wenn ein Sender ein Acknowledgment empfängt, wird das Congestion Fenster um eine Segmentgröße erhöht. Auf diese Weise wird die Rate mit der gesendet wird jede Runde verdoppelt, bis eine Ansammlung im Empfänger auftritt. Dann wird die Senderate wieder verlangsamt. Eine Ansammlung (Congestion) kann durch Timeouts und duplicate acknowledgements erkannt werden.

44) Gehen Sie auf die Performanceaspekte ein, die sich durch Fast Retransmit und Fast Recovery bei TCP ergeben. Welche Rolle spielt dabei das Duplicate Ack? Schildern Sie diese Verfahren im Detail. Wozu dienen delayed Acknowledgements und der Naggle Algorithmus?

Fast Retransmit: Ursprünglich konnte ein Packetverlust nur durch Auslaufen des Retrasmission Timers erkannt werden. Auch im Fall eines Neusendens muss die Time-out Zeit eingehalten werden. Das führt unweigerlich zu einer Verlangsamung des Netzes. Muss neu gesendet werden, wird sofort ein Time-out durchgeführt. Bei Fast Retransmit hat der Empfänger sofort ein duplicate ACK zu senden um dem Sender zu zeigen, welche Segmente von ihm erwartet werden. Allerdings sendet der Empfänger auch ein duplicate Ack wenn Segmente nur in der falschen Reihenfolge auftreten. Aus diesem Grund wartet der TCP Sender noch ein drittes ACK ab → Dieser Mechanismus wird „Fast Retransmit“ genannt.

Fast Recovery: Fast Recovery geht sozusagen Hand in Hand mit Fast Retransmit um den einfachen Packetverlust zu reparieren.

Mechanismus: sstresh wird auf die Hälfte der aktuellen Window-Größe gesetzt. Das Sendefenster selbst wird auf diesen Wert + 3 gesetzt (daher kann man davon ausgehen, dass der Empfänger schon 3 Duplicate ACK's erhalten hat. Danach wird Congestion Avoidance ausgeführt. Für jedes weitere Duplicate ACK erhöht der Sender das Sendefenster um 1. Ab dem Zeitpunkt an dem der Sender ein ACK von neuen Daten erhält setzt der das Sendefenster wieder auf den Wert von sstresh und Congestion Avoidance wird erneut ausgeführt.

Delayed Acknowledgements: Wenn auf der Empfängerseite ein Paket ankommt, wird im Normalfall sofort ein ACK gesendet. Das führt auch zu unnötig viel zusätzlichem Verkehr auf dem Netz. Bei interaktiven Anwendungen herrscht meist ein reger Verkehr auf der Empfängerseite. Wenn also das ACK erst gesendet wird, wenn im Puffer des Senders ein zweiter Frame zur Übertragung bereitsteht, der das ACK piggy-backen kann, erreicht man damit eine wesentlich kleinere Netzlast. Die interaktive Applikation wartet damit jedes Mal 200ms, ob nicht vielleicht doch noch ein ACK mitreisen will, und sendet dann erst den nächsten Frame. In diesen 200ms hat die Applikation eventuell auch etwas zu senden. Wenn nicht, macht sich das ACK alleine auf den Weg.

Der Nagle Algorithmus: Manche kleine Applikationen (z.B. telnet, rlogin) senden nur sehr kleine Segmente "tinygrams". Es können nun sehr stark frequentierte tinygrams zu einer hohen Last bei langsamen WAN-Verbindungen werden. So besagt der Nagle Algorithmus, dass (wenn eine TCP-Verbindung auf eine Bestätigung (Ack) wartet) kleine Pakete nicht gesandt werden dürfen, bis das acknowledgement angekommen ist. So kann TCP in der Zwischenzeit kleine application-Daten sammeln und diese zu einem gesamten Segment zusammenführen.

Fragenbereich 5 (Kapitel: IP Routing Overview, IP Standard Routing Protocols RIP, OSPF, Classful/Classless Routing):

45) Was versteht man unter „direct“ und „indirect delivery“ im Zusammenhang mit IP Forwarding? Was ist ein Default Gateway? Wie sind statische Routen charakterisiert? Wann werden diese bzw. können diese eingesetzt werden? Was ist Default Routing? Wie wird eine Default Route gekennzeichnet? Wo kommen Default Routes zum Einsatz? Was ist grundsätzlich dynamisches Routing? Welche Rolle spielen Routing Protokolle und die Routing Metrik dabei?

Im Falle des indirect routing muss das IP-Forwarding von Routern übernommen werden, wobei die Zieladresse dem IP-Header zu entnehmen ist. Die Zustellung des Pakets erfolgt dabei hop by hop, d.h. von Router zu Router. In diesem Fall spricht man auch von indirekter Zustellung (indirect delivery) von IP-Datagrammen. Der Host ist nur dafür zuständig, einen default-Router als nächste Station auszuwählen. Der Host selbst ist für die direkte Zustellung von Datagrammen zuständig (direct delivery). Welche Zustellungsart gewählt wird, hängt von der Destination-net-ID ab. Eine direkte Zustellung wird wohl nur dann verwendet, wenn die netID des Hosts mit der netID des Zielhosts übereinstimmt. Im umgekehrten Falle ist eine indirect delivery vorzuziehen.

Alle Datenpakete, die ein Rechner nicht im eigenen Netz versenden kann, werden an das Default Gateway gesendet. Meist ist das eine eigene Hard- und Software, die die Weiterleitung übernimmt. Mit einem Gateway werden meistens Netzwerke verbunden, die aufgrund der unterschiedlichen Protokollstruktur nicht miteinander kommunizieren können. Ein Gateway kann also alle in einem Netzwerk vorhandenen Protokollarten ineinander umwandeln.

Default Gateway: Die IP-Hosts sind bei statischen Routen selbst dafür verantwortlich einen default-router (Default Gateway) auszuwählen, über welchen der nächste Hop im Falle einer indirekten Zustellung erfolgen soll.

Statische Routen: Die Routingtabellen werden vom Netzwerkadministrator vordefiniert. Sie sind nicht responsive auf Änderungen der Topologie. Bei komplexen Netzwerken kann es dann schwer werden statische Routen zu generieren bzw. zu ändern. Allerdings entsteht bei der Verwendung statischer Routen kein zusätzlicher Overhead in der CPU und kein zusätzlicher Verkehr im Netzwerk. In manchen Technologien sind nur statische Routen möglich (z.B. X.25, ISDN). Sie werden auch manchmal wegen Security-Vorteilen verwendet.

Default Routing: Allgemein werden unbekannte Ziele vom Router verworfen. Man kann allerdings eine Default Route definieren. Jetzt werden alle unbekanntes Ziele zu einem bestimmten (default) Router geschickt. Dies kann den Vorteil haben, dass dieser die Zieladresse kennt. Es hat auch den Vorteil, dass nun nicht mehr jeder Router eine komplette Routing-Tabelle benötigt. Das Default-Network wird über die Net-ID 0.0.0.0 beschrieben. Default Routing wird verwendet, wenn man ein lokales Netzwerk mit z.B. dem Internet verbinden will. Auch können mittels Default-Routing mehrere Subnetze miteinander verbunden werden.

Dynamische Routen: Routingtabellen werden hier dynamisch upgedatet. Sie erhalten die Informationen über Routing-Protokolle von anderen Routern. Die Routing-Protokolle müssen die momentane Netzwerktopologie kennen. Weiters sollten sie den besten Weg zu jedem erreichbaren Netz kennen. Sie sollten auch die Informationen über das Erreichen den besten Weges in Routingtabellen speichern. Um den besten Weg feststellen zu können müssen aber die Metrik-Informationen des Netzes bekannt sein. Meist werden statisch vordefinierte Variablen (z.B. hop, cost, bandwidth,...) verwendet. Hier gibt es 2 grundlegende Technologien: Distance Vector und Link State

46) Erläutern Sie das dynamisches Routing-Protokoll RIP im Detail. Welche Probleme können bei der prinzipiellen Vorgehensweise von RIP ohne die in Frage 47) beschriebenen Mechanismen auftreten? Welche Verbesserungen gegenüber RIPv1 gibt es durch RIPv2

RIP (Routing Information Protocol) ist ein typisches IGP (Interior Gateway Protocol). Die Entscheidungen, die dieses Protokoll trifft, beruhen auf Abzählung der Hops (Distance Vector

Protocol, mit Hilfe eines Vektors wird die Richtung des Routers festgelegt, der die schnellste Verbindung in ein bestimmtes Netz aufbauen kann). Die Information, mit welchem Netzwerk der RIP-Router verbunden ist, wird in die Routing Tabellen eingetragen, in denen dann die netIDs der direkt mit ihm verbundenen Netze und die Entfernung (in Hops) von ihnen steht. Alle 30 Sekunden wird über einen Broadcast die eigene Routing Tabelle an benachbarte Router gesendet, die ihre Tabellen dann dementsprechend updaten können. Die upgedateten Tabellen werden schließlich weiterversendet. Nach einer bestimmten Zeit wissen alle Router über alle Netzwerkadressen im kompletten Netz bescheid. Enthalten die upgedateten Listen Router, die dieselbe netID haben, wird die Verbindung mit den wenigsten Hops in die Liste eingetragen. Haben zwei Verbindungen dieselbe Anzahl an Hops, wird automatisch die erste eingetragen. Mit diesem Protokoll werden die Netzkapazitäten optimal ausgenutzt. Sobald eine bessere (schneller oder mit weniger Hops, bessere Metrik) Verbindung zu einem Router besteht (auslesbar aus den Update-Listen), wird diese Verbindung ohne lange Nachzufragen einfach in die Liste übernommen. Sollte ein Update mit einer schlechteren Metrik als die für dieses Netz derzeit gespeicherte daher kommen, wird es nur in die Tabelle übernommen, wenn es von dem Router kommt, der in der Tabelle als Vektor für dieses bestimmte Netz eingetragen ist. Alle anderen für dieses Netz werden ignoriert.

Wenn ein Routing-Tabellen-Eintrag nicht innerhalb von 180 Sekunden upgedatet wird, wird er als veraltet erklärt. Ohne einen speziellen Mechanismus haben alle anderen Router nach mindestens 180s wieder eine vollständige Routing- Tabelle. über spezielle Network-Unreachable-Messages (werden an alle Router gesendet) dauert dieser maximal 180s. Während dieser Übergangsphase wird nach der alten Tabelle verfahren. Bei sehr großen Netzen kommt es aufgrund der 180s Phase bei Ausfällen oder Störungen zu einer sehr langsamen Konvergenz. Weil der Inhalt eines Updates der Routing-Tabellen verbindlich ist (Trusted Information Principle), kann es bei RIP sehr leicht zu Schleifenbildung kommen. Durch die beiden oben genannten Fehler können Datagramme über redundante Wege kreisen und zu einem "Count to Infinity" Problem werden. Ein weiteres Problem ist, dass Routing Tabellen immer als ganzes verschickt werden, was für große Netze nicht sehr vorteilhaft ist, weil sehr viele netIDs in der Tabelle enthalten sind. Ein ständiges periodisches Updaten der Listen fällt bei Weitverkehrsnetzen zusätzlich ins Gewicht.

Methoden, den oben erläuterten Problemen Abhilfe zu schaffen, sind Maximum Hop Count, Split Horizon, Poison Reverse, Hold Down und viele mehr.

Die in der ersten Version verwendeten Datagramme enthielten oft freie Stellen. Eine Neuerung in der zweiten Version im Vergleich zur ersten ist, dass diese freien Plätze bestimmten Funktionen zugeteilt wurden. Die Neuerungen sind Routing Domains, Übertragung von Subnetmasks und Next Hop Redirect Information, Route Advertisements über EGP-Protokolle und Authentifikation.

Bei Routing Domain wird ein Subnet in mehrere Domänen aufgeteilt. Routing Updates können schließlich den Zieldomänen übermittelt werden. Ein Router kann damit mehrere Domains gleichzeitig mit den entsprechenden Datagrammen versorgen.

Eine weitere Neuerung im RIP2-Header ist das SUBNET MASK. Die zu der jeweiligen IP-Adresse gehörige Subnetzmaske wird mitübertragen, wodurch Variable Length Subnet Mask (VLSM) möglich wird. Im Feld IP ADDRESS kann die IP-Adresse jenes Routes eingetragen werden, der als Next Hop für die Weiterleitung verwendet werden soll. Er muss im selben Teilnetz direkt erreichbar sein. Die IP-Adresse 0.0.0.0 in dem Feld bedeutet, dass der aussendende Router als Next Hop für das angegebene Netz fungiert. Nicht alle Router in einem Teilnetz müssen Routing Updates aussenden (in bestimmten Szenarios).

Für die Weiterleitung der Updates wird kein Broadcast mehr verwendet, was alle angeschlossenen Geräte beschäftigen würde, sondern ein Multicast mit Adresse 224.0.0.9, mit der sich Router, die zu dieser Gruppe gehören, identifizieren. Authentifikation wird verwendet bei Routing Updates. Momentan gibt es nur einen Typ von Authentifikation (Typ 2), der einem einfachen Passwortschutz entspricht. Ein Router, der ein Update ohne gültiges Passwort erhält, ignoriert dieses. RIPv2 ist abwärtskompatibel. Ein RIPv2-Router, der im RIPv1-Modus arbeitet, sendet nur RIPv1 Datagramme. Ein RIPv2-Router, der im RIPv1-Kompatibilitätsmodus arbeitet, sendet zwar RIPv2-

Datagramme aus, jedoch als Broadcast, weil ein RIPv1-Router die zusätzlichen Headerfelder einfach ignoriert. RIPv2-Router im RIPv2-Modus senden Datagramme als Multicast.

47) Wozu dienen bei RIP bzw. Distance Vector Protokollen i.a. Techniken wie Maximum Hop Count, Split Horizon, Poison Reverse und Hold Down. Beschreiben Sie diese im Detail.

Die maximale erlaubte Distanz zwischen zwei Subnetzen wird bei RIP auf 16 begrenzt. Ist im Abschnitt DISTANCE in der Routing Tabelle der Wert 16 gespeichert, ist das Netz nicht mehr erreichbar. IP-Datagramme mit dieser netID werden dann vom Router verworfen, der dann ein ICMP-Datagramm mit "network unreachable" generiert. Ein Nichterreichen eines Netzes kann also aktiv bekannt gegeben werden. Das 180s Timeout in den Nachbarroutern muss dann nicht abgewartet werden. Die Anzahl der Hops kann nicht größer als 15 sein.

Maximum Hop Count alleine reduziert aber nicht temporäre Routing Loops. Eine Methode gegen Routing Loops und Verringerung der Slow Convergence (Zählen bis 16) ist **Split Horizon**. Sie verhindert, dass Informationen über Netze in die Richtung geschickt werden, aus der sie gekommen sind, außer die Information enthält eine bessere Verbindung, die in die Tabelle eingetragen werden kann. Außerdem wird die Konvergenzzeit beim benachbarten Router auf die Zeit der Fehlererkennung (180s) anstatt von $16 \cdot 30s = 480s$ reduziert.

Eine alternative Methode ist **Poison Reverse**. Dabei geben Router in ihren Routing Updates die Nichterreichbarkeit (Message = Poison) von Netzen in die Richtung bekannt, aus der sie die Informationen über diese Netze erhalten haben, wobei beim benachbarten Router die Konvergenzzeit auf die Zeit der Fehlererkennung reduziert wird (180s). Das funktioniert ganz gut in einfachen Netzen.

Hold Down: In komplexeren Netzen benötigt man einen zusätzlichen Mechanismus, um Schleifen zu verhindern. Es veranlasst einen Router nach einer Nichterreichbarkeitsmeldung eines Netzes weitere Informationen über dieses Netz für eine bestimmte Zeit zu ignorieren. Die Informationen stammen nicht von dem Router, der die Nichterreichbarkeitsmeldung geschickt hat. Ein typischer Wert ist dabei 240s. Alle Router im Netz haben damit die Möglichkeit, die ausgeschickte Nichterreichbarkeitsmeldung zu erhalten. Außerdem wird dadurch ein gewisser Einschwingvorgang abgewartet (die Nachricht breitet sich ja als Welle im Netzwerk aus!), wodurch Inkonsistenzen in den Routing Tabellen und damit Schleifen vermieden werden. Durch ein Hold Down wird aber die Konvergenzzeit ein wenig erhöht, was sich in manchen Fällen ungünstig auswirken kann.

48) Was ist Classful Routing? Wie erfolgt dabei der Routing Table Lookup? Was ist Classless Routing? Wie erfolgt hier der Routing Table Lookup? Welche zusätzlichen Möglichkeiten gibt es bei der Adressierung? Was ist CIDR? Was versteht man unter Route Summarization? Warum sollte aber auch bei Classless Routing, die IP Adressierung der physikalischen Topologie folgen?

Classfull Routing: Routing Protokolle wie RIP, IGRP können keine Subnetz-Informationen über routing updates übertragen. Dies führt zu folgenden Konsequenzen:

- Wenn eine gegebene Klasse A, B oder C gegeben ist und diese in weitere Subnetze unterteilt wird, so muss die Subnetmask in allen Bereichen gleich sein) Es ist keine variable Länge der Subnetzmaske (VLSM) erlaubt.
- Wird ein Routing-Update zu einem Interface geschickt, dessen Netzwerknummer unterschiedlich dem des subnetted networks ist, so wird nur die Netzwerknummer des Klasse A, B oder C Netzwerkes verkündigt. So wird die Zusammenfassung der Routen (route summarize) nur über die Grenzklassen optimiert. Folglich muss ein Subnetz-Bereich kontingent sein.
- Classful routing

Routing Table lookup: Nehmen wir an, es wird ein IP-Datagramm mit einer gegebenen IP-Adresse vom Router empfangen. Jetzt kann die IP-Adresse als Klasse A, B oder C Netz identifiziert werden. Als nächstes wird ein lookup in der Routing-Tabelle ausgeführt. Wenn sich dort kein Eintrag befindet, wird das IP-Datagramm weggeworfen. Handelt es sich aber um einen Treffer, so wird die IP-Adresse mit jedem bekannten Subnet überprüft. Existiert kein solches Subnetz, so wird das Paket weggeworfen.

Auch wenn das übergeordnete Netzwerk beim Router bekannt ist, aber das Subnetz nicht existiert wird das Datenpaket verworfen. Das ändert auch nichts, wenn ein Default Gateway eingestellt ist. Somit muss der unterteilte Bereich der Subnetze kontingent (durchgehend) sein. Somit muss jedes Subnetz eines gegebenen Netzes nur über den Pfad mit den jeweiligen Subnet-IDs ansprechbar sein.

Classless Routing: Wird verwendet, wenn Routing-Protokolle Informationen über Subnetzmasken bei Routerupdates übertragen können. Beispiele sind: RIPv2, OSPF, eIGRP.

Das hat folgende Vorteile:

- Variable Längen der Subnetzmasken können verwendet werden (VLSM). Dadurch kann der Adressbereich effizienter aufgeteilt werden.
- Route Summarize kann auf jeder Adresse durchgeführt werden (und nicht nur auf class boundaries)
- Classless routing

Routing Table lookup: Nehmen wir an, es wird ein IP-Datagramm mit einer gegebenen IP-Adresse vom Router empfangen. Diese Adresse wird nicht als Class A, B oder C interpretiert. Es wird ein lookup in der Routing-Tabelle ausgeführt, der den besten Treffer für diese IP-Adresse zurückliefert. Dazu werden die IP-Präfixe in der Routing-Tabelle bit für bit (von links nach rechts) mit der IP-Adresse verglichen. Das IP-Datagramm wird an jenes Subnetz übermittelt, das am besten an die IP-Adresse passt. Das bietet den Vorteil, dass IP-Adressen mit beliebigem Subnetting verwendet werden können und man so unabhängig vom übergelegtem Netzwerk ist. So kann nun auch ein Sub-Subnetting betrieben werden.

CIDR (Classless Interdomain Routing): Bei diesem Verfahren werden IP-Adressen zusammengefasst, wobei ein Block von aufeinander folgenden IP-Adressen der Klasse C als ein Netzwerk behandelt werden. Eine Organisation bekommt dabei nicht mehr ein ganzes Netz zugeteilt, sondern nur mehr Subnetze mit einer jeweiligen Subnetzmaske. Die Routing Tabellen der Router werden dadurch auch ein wenig entlastet, in dem der IP-Adresse ein Präfix angehängt wird, mit dem eine große Firma oder ein großer ISP gekennzeichnet werden kann. Auch darunterliegende Netze können damit zusammengefasst werden (Supernetting). Bei IP-classless kann die Grenze zwischen Netzwerk- und Hostteil der IP-Adresse nicht nur an den Byte-Grenzen, sondern auch an beliebigen Bit-Positionen innerhalb der 32-Bit gesetzt werden. Durch die Subnetz-Maske wird angegeben, wie viele Bits der Adresse den Netzwerkteil bilden (Bsp.: 193.171.213.0/24 ... die ersten 24 Bit bilden die Netzwerk-Adresse).

Die IP-Adressierung sollte aber trotzdem ein wenig strukturiert und ans physikalische Netz angepasst sein. Es hat nicht viel Sinn, einen Host in einem Subnetz zu suchen, in dem er gar nicht ist, nur weil die Strukturierung nicht gegeben ist. Es ist nicht notwendig, einen bestimmten Host im kompletten Internet zu suchen; das Netz wird dadurch sicher nicht schneller werden.

49) Beschreiben Sie kurz das Grundprinzip einer Link-State Routing Technologie wie OSPF. Wie kommen in OSPF Nachbarschaftsbeziehungen zustande? Welche OSPF Messages werden dazu verwendet? Wie können Nachbarschaftsbeziehungen eindeutig in der Topology Database beschrieben werden? Wozu dienen OSPF Database Description Messages? Welche Netzwerktypen/Linktypen kann ein Router-LSA (Typ1) beschreiben? Wie wird tatsächlich das

LSA (die Verkehrsfunknachricht) bewerkstelligt (Stichwort "Hot Potatoe")? Welche Bedeut. haben dabei LSA Sequence-number und LSA Age? Wie werden OSPF Messages transportiert?

Bis jetzt existierte in jedem Router eine a-priori konsistente Datenbank in jedem Router. Die grundlegende Bedeutung der so genannten link states sind anlegen und behalten der Daten in der Datenbank. Ein Link-State steht für eine lokale Nachbarschaft zwischen 2 Routern. Er wird zwischen Ihnen aufgebaut. Andere Router werden über den Link- Aufbau mittels Broadcast über die neue Verbindung informiert ("traffic news"). Die Link-States werden kontinuierlich überprüft. OSPF ist ein IP Protokoll.

Die Router haben die komplette Verkehrslage des Netzwerks gespeichert. Es wird jedoch durch den Dijkstra Algorithmus ein Verkehrsnetz gebaut, welches schließlich auch benutzt wird. Der Dijkstra Algorithmus sorgt dafür, dass es zu jedem Knoten im Netzwerk nur noch einen Weg gibt (ähnlich Spanning Tree Protocol)

Wie werden Linkstates benutzt? Angrenzende Router erklären sich als Nachbarn, wenn sie ihren Link-State auf up setzen. Dieser Link-State kann mittels eines Hello-Requests überprüft werden. Jede Änderung eines Link-States wird den anderen Routern der OSPF-Domain mittels LSA (Link State Advertizements) mitgeteilt. Das ist ein Broadcast-Mechanismus. LSAs sind kürzer als normale Routing-Tabellen. Die komplette Topologie-Map vertraut LSA.

Wie kommen Nachbarschaftsbeziehungen zustande? Zuerst sendet jeder Teilnehmer einen hello-Request. Wird ein neuer Nachbar erkannt, so sendet der erste seine database description message. Er bekommt dann vom 2. einen LS request. Das bedeutet, dass er mehr über den anderen erfahren möchte. Im nächsten Schritt bekommt er mittels eines LS update die Daten des ersten. Dieser Quittiert den Empfang mittels eines LS ack. Nun möchte auch der 2. Partner seine Topologie-Datenbank dem 1. übermitteln. Dazu sendet er ihm eine Database description message. Diese wird dann wiederum mit einem LS request beantwortet. Mit dem nächsten LS update werden die Daten übertragen, die mit einem LS ack quittiert werden. Nachdem sich nun diese 2 Router synchronisiert haben, melden sie ihren ganzen anderen Nachbarn die neuen Beziehungen weiter. Da dies sofort geschieht und nicht erst nach der laut Timer nächsten LSA-Message (30 min) dauert es nur eine kurze Zeit bis jedem Router das gesamte Netzwerk bekannt ist („Hot Potatoe“)

Die Database: Jeder Router besitzt eine Topologie-Datenbank. Sie ist wie eine "Network Roadmap". Sie beschreibt somit das gesamte Netzwerk. Die Datenbank basiert auf einem Graphen. Jeder Knoten steht für einen Router. Jede Ecke steht für ein Subnetz, wobei jeder Router den Graphen als root verwendet. Anhand dieser Datenbank kann sich der Router den besten Weg in ein anderes Netz berechnen. In der Datenbank sind ja alle Wege vorhanden. So gibt es kein Warten mehr, falls ein anderer Router ein Gerücht verbreitet. Nachdem nun der kürzeste Pfad ermittelt worden ist, wird er in der Routing-Tabelle eingetragen. OSPF ist auch fähig zwischen internen und externen net-IDs zu unterscheiden.

Mit einem Router-LSA (Typ1) kann ein bestimmtes Subnet beschrieben werden.

In OSPF gibt es 3 Arten von Routing:

- **intra area routing**
Hier werden Daten innerhalb einer Area übertragen. Es existieren in dieser Area Router-Link LSA (**Typ1 → Router LSA**) und Network Link LSA (Typ2).
- **iner area routing**
Hier findet ein Datenaustausch zwischen 2 Areas über eine Backbone-Area statt. Es existiert hier ein Summary Link LSA (Typ3 oder 4). Typ3 wird zur Verbindung von Netzwerken und Typ4 zum Verbinden von IP-Adressen auf ASBRs verwendet.
- **exterior routing**
Pfade zu externen Zielen sind statisch konfiguriert oder über ASBR (Autonomous Systems

Boundary Routers) mittels EGP oder BGP importiert. Dazu dient ein AS External Summary LSA (Typ5)

LSA Age: Die LSA Message wird alle 30 Minuten gesendet. Empfängt ein Router 60 Minuten lang keine LSA Messages altert der Link aus und wird somit gelöscht.

50) Was bedeutet Broadcast Umgebung (shared media wie LAN) für das OSPF Prinzip? Welche Funktion haben Designated Router und Backup Router in einer Broadcast Umgebung? Hat das auf das Weiterleiten von IP Datagrammen einen Einfluß? Welche Abläufe bzw. welche Arten von Destination Adressen gibt es in einer Broadcastumgebung bei der Übertragung von OSPF Messages? Mit welchem LSA-Typ wird eine Broadcast Umgebung bekanntgegeben?

Broadcast Mechanismus von LSA: ist extrem wichtig, da die Konsistenz der Topologie – Datenbank davon abhängt. Jede LSA Nachricht muss explizit bestätigt werden, ansonsten greift ein Timeout und die LSA Nachricht wird erneut gesendet. Wiederholt sich ein Fehler zu oft, wird Nachbarschaft aufgelöst, da es beim entsprechenden Nachbarn offenbar zu einem Fehler gekommen ist und sonst die Konsistenz des Netzwerkes nicht mehr gewährleistet wäre. Zusätzlich werden Link States alle 30 min wiederholt bzw. altern nach 60 Minuten aus. → Grund: automatische Korrektur von Fehlern im Netzwerk. Der Einfluss auf die Weiterleitung von IP-Datagrammen ist gegeben, da sich aufgrund des OSPF Mechanismus „hello“-Message, LSA-Update doch erheblicher Traffic entsteht.

OSPF Broadcast Networks:

Wenn sich mehrere Router in einem Multi-Access Netzwerk befinden funktioniert, das Jeder-mit-Jedem Prinzip aufgrund der $N*(N-1)/2$ Problems nicht. Weiters wären die Informationen über die Nachbarn alle redundant, da jeder Router alle anderen Router als Nachbarn ansehen würde. Es würde somit ausreichen einen Zentralen „Knotenpunkt“ zu haben. → Dieser heißt in OSPF „Designated Router“

Adressen: OSPF benutzt dedicated IP Multicast Adressen für den Austausch der Routing Messages. (zb 224.0.0.5 „All OSPF Routers“; 224.0.0.6 „All Designated Routers“)

Broadcast Umgebung: Sobald mehrere Router einen Multi-Access auf ein Netzwerk-Segment haben (z.B. LAN, X.25, Frame Relay) wird ein Designated Router und ein Backup Router bestimmt. Diese werden mittels der hello-Messages ausgewählt. Der Grund für den Designated und Backup Router ist, dass sonst ein zu großer Netzwerk-Traffic auftreten würde, wenn sich die Router synchronisieren. Aus diesem Grund wird ein Router zum Designated Router. Er wirkt nun als Ansprechpartner für alle anderen Router. überprüft wird er durch den Backup Router. Fällt der Designated Router aus, so wird der Backup-Router zum Designated-Router und es wird ein anderer Backup-Router bestimmt.

Designated Router (DR): Er versorgt alle anderen Router dieses Segments mit Nachbarschafts-Verbindungen über virtuelle Punkt zu Punkt Verbindungen. Der Designate Router ist für die Abgabe von Network-LSAs verantwortlich. Der Backup Router ist einfach die Ausfallsicherung für den Designated Router, welcher dessen Aufgabe übernimmt wenn dieser ausfällt.

51) Was ist der Grund eine OSPF Domain in Areas zu unterteilen? Welche Mechanismen kommen hier zum Tragen? Was versteht man unter Backbone Area und wie erfolgt der Anschluß anderer Areas? Welche LSA-Type kommen dadurch zusätzlich zum Einsatz? Wie erfolgt die Handhabung dieser Type durch einen Area Border Router bzw.durch einen areainternen Router? Was versteht man unter Route Summarization im allgemeinen und wie kann man das bei OSPF mit Areas nützen? Worauf muß man bei Route Summarization in Zusammenhang mit OSPF aufpassen?

Jede Area hat ihre eigene Topologiedatenbank. Somit bleibt die Area-spezifische Routing-Information innerhalb der Area. Ändert sich die Topologie eine Area, bleibt der Routing-Traffic innerhalb der Area. Somit wird bei route summarization der Traffic drastisch reduziert. Jede OSPF-Area bekommt dann ihre eigene area-ID. Diese sind ähnlich den AS-Nummern. Sie ist wie eine IP-Adresse strukturiert oder nur eine einfache Nummer. Allerdings muß sie innerhalb einer OSPF-Domain eindeutig sein. Eine OSPF-Domäne beinhaltet zumindest eine einfache Area. Ein Router, der mit mehreren Areas verbunden ist, wird Area Border Router (ABR) genannt. Ein ABR kennt die Topologie-Datenbanken aller mit ihm verbundenen Areas. Prinzipielle OSPF Areas müssen über eine spezielle Area verbunden sein: der Backbone Area. Sie hat die area-ID 0.0.0.0 oder 0. Existiert in dieser Domain nur eine Area, so ist sie die Backbone Area.

Nicht backgebonte Areas dürfen nicht direkt miteinander verbunden werden. Diese Aufgabe übernimmt die Backbone-Area. Dieses Konzept erzwingt eine sternförmige Konfiguration aller Areas um die Backbone Area. Backbone Area Routers sind entweder über direkte physikalische Links oder über virtuelle Links miteinander verbunden. In speziellen Fällen kann ein virtueller Link dazu verwendet werden um den Verkehr von isolierten Areas innerhalb der backbone Area zu tunneln.

LSA-Typen

- **intra area routing**
Hier werden Daten innerhalb eine Area übertragen. Es existieren in dieser Area Router-Link LSA (Typ1) und Network Link LSA (Typ2).
- **iner area routing**
Hier findet ein Datenaustausch zwischen 2 Areas über eine Backbone-Area statt. Es existiert hier ein Summary Link LSA (Typ3 oder 4). Typ3 wird zur Verbindung von Netzwerken und Typ4 zum Verbinden von IP-Adressen auf ASBRs verwendet.
- **exterior routing**
Pfade zu externen Zielen sind statisch konfiguriert oder über ASBR (Autonomous Systems Boundary Routers) mittels EGP oder BGP importiert. Dazu dient ein AS External Summary LSA (Typ5)

Area Border Router: Der Area Border Router beinhaltet 2 Topologiekarten (Eigene Area, Backbone-Area). Er exportiert die Routen seiner eigenen Area zum Backbone Router mittels Summary LSA´s. Der Area Border Router importiert alle Routen von anderen Areas in seine eigene Area. Auch dies wird mittels Summary LSA´s gemacht.

Summary LSA´s sind „Distance Vector updates“. Sie wird von den ABR generiert um die Router in einer Area bezüglich der Kosten von „außen“ zu informieren sowie vice versa. Weiters können Summary Link LSA´s für Route Summarization benutzt werden.

Route Summarization: Kann entweder manuell oder vom Area Border Router konfiguriert werden (Minimierung der Routing-Tabellen-Einträge), Classless Routing, Summarization kann überall in der IP Adresse stattfinden. (zb. können mehrere Class C Netzwerke zu einer einzelnen Adresse zusammengefasst werden. [201.1.0.0 bis 2.01.1255.0 (Subnet Maske 255.255.255.0) wird zu 201.1.0.0 (Subnet Maske 255.255.0.0) zusammengefasst]) – Beim Zusammenfassen werden nur die niedrigsten Kosten gemeldet.

Wenn ein Router eine Summary LSA erhält fügt er die Kosten aus der Summary LSA zu den Kosten hinzu um den genannten Area Border Router zu erreichen. Wenn ein Area Border Router eine Summary LSA von einem Backbone erhält fügt er die Kosten aus der Summary LSA zu den Kosten hinzu um den genannten Area Border Router zu erreichen. Das Resultat wird in den Routing Tabellen festgehalten. Weiters wird eine Summary LSA in die anderen Areas mit den fertigen Kosten gesandt.

In OSPF können somit beispielsweise eine Vielzahl in IP´s für die Backbone Area zusammengefasst werden.

52) Was passiert, wenn zwei Teile einer OSPF Domain verbunden werden oder auseinanderfallen? Welcher Selbstheilmechanismus bei Zusammenführung von zuvor getrennten Netzwerkteilen ist zu beobachten? Welche Designregel sollte man beachten? Welche LSA-Typen werden benötigt, um externe Netze in einer OSPF Domain bekanntzumachen? Welche speziellen Router werden benötigt? Welche externe Metrik Typen gibt es? Was versteht man unter Stub Areas in Zusammenhang mit externen Netzen?

Jeder Router weiss bescheid über:

- Die genaue Topologie seiner Area und kennt die besten Pfade zu allen Netzwerken in seinem Netzwerk
- ABR seiner eigenen Area und den Kosten um andere ABRs zu erreichen. ABR's werden in einer separaten Liste gespeichert.
- Wenn ein Netzwerk aktiviert wird, wird ein korrespondierendes Summary LSA vom ABR ausgesandt. (mit den aktuellen Kosten um das Netzwerk vom gegebenen ABR aus zu erreichen).

Verbindung: Wenn 2 Teile eines OSPF Netzwerkes verbunden werden, werden sich die Router mit einer „hello“-Message „begrüßen“. Danach wird einer der beiden Router dem anderen seine LSA Message senden. Dieser wird daraufhin mit einem LSA-Request die komplette Topologiekarte des anderen verlangen, da sie ihm vermutlich unbekannt ist. Daraufhin wird der Router 1 dem Router 2 sein LSA-Update schicken. Danach läuft das ganze nochmals in umgekehrter Reihenfolge ab.

Trennung: Wenn 2 Teile eines OSPF Netzwerkes getrennt werden werden die eingetragenen Link States aufgrund des Timeouts von 60 min (da aufgrund dessen keine weiteren LSA-Messages mehr empfangen werden) ausaltern.

Exterior routing: Pfade zu externen Zielen sind statisch konfiguriert oder über ASBR (Autonomous Systems Boundary Routers) mittels EGP oder BGP importiert. Dazu dient ein AS External Summary LSA (Typ5)

Wenn ein Router ein Summary LSA erreicht, so werden die vom Summary LSA verkündeten Kosten zu den Kosten addiert, die notwendig sind, das verkündete ABR zu erreichen. Diese Kosten werden dann in der Routing Tabelle gespeichert.

Wenn ein ABR-Router ein Summary LSA vom Backbone erhält, so geht er gleich vor, wie vorher beschrieben. Weiters sendet er ein Summary LSA in die Area mit den kulminierten Kosten und setzt das ABR-ID zum aktuellen Wert.

Stub Areas: Normalerweise erhält jeder interne Router Informationen über jedes externe Ziel. OSPF erlaubt nun eine Definition von Stub Areas um Speicherbereiche der internen Router zu minimieren. Nun weiss nur der Area Border Router jeder Area über alle externen Ziele Bescheid. Jeder interne Router erhält Standardrouteinträge. Es wird nun der ganze Verkehr der nicht innerhalb der Domain bleibt an den Area Border Router weitergeleitet.

Fragenbereich 6 (Kapitel: BootP-DHCP-TFTP, Telnet-FTP, DNS, SMTP, HTTP):

53) Geben Sie einen Überblick (Grundprinzip, Funktionsweise, Protokollabläufe, Einsatz, etc). über das Protokoll BootP. Welche Konfigurations-Parameter können im Header, welche anderen wichtigen Konfigurations-Parameter können in der Vendor Specific Area transportiert des Headers werden? Was versteht man unter BootP Relay Agent, wann wird dieser benötigt und welches Feld im BootP Header ist dafür verantwortlich?

BOOTP wurde entwickelt um RARP zu ersetzen, und bietet nun Bootstrapping an. BOOTP basiert auf einem Client-Server Prinzip und benutzt UDP als Kommunikation

Bootstrapping: Erlaubt Disk-losen Clients sowie Netzwerkkomponenten ohne flüchtigen Speicher, OS-Code zu laden Parameter von einem Zentralen Server zu konfigurieren..

Vorgang:

- Der BOOTP Client sendet einen Request an den BOOTP-Server (255.255.255.255 sowie 0.0.0.0 als Source Adresse).
- Der Server benutzt die MAC Adresse des Clients um ihn in einer Datenbank zu suchen und zu verifizieren.
- Bei Erfolg: Der Server sendet die geforderte Boot Information mittels Broadcast zum Client.
- Ende der BOOT-P-Prozedur.

Boot-Info enthält: Die IP-Adresse eines IP-Hosts der die Bootimages enthält, sowie die Dateinamen dieser Bootfiles. Der Client benutzt nun diese Information um die Bootfiles via TFTP zu laden. Diese Trennung bewirkt, dass der eigentliche BOOTP Server nur eine kleine Reference Tabelle speichern muss – die (evtl. größeren) Bootimages können ausgelagert werden.

Der BOOTP-Client ist für die Error Detection verantwortlich

Aufgrund des Limited Broadcasts (255.255.255.255) wäre das ganze auf ein einfaches LAN ausgelegt. Um auch BOOTP Server von anderen Subnets zu erreichen müssen die BOOTP Server als Relay Agent arbeiten können.

Konfigurationsparameter: Operation Code, Hardware Type, Length of the Hardware Address, Hops, Transaction ID, Client IP, Your IP, Server IP, Router IP, Client MAC Address, Server Host Name, Bootfilename.

Vendor Specific Area: Kann zusätzliche Information des BOOTP-Servers enthalten (zb. Subnet Maske, Hostname, Domainname, IP-Adresse des DNS-Server

54) Geben Sie einen Überblick (Grundprinzip, Funktionsweise, Protokollabläufe, Einsatz, etc.) über das Protokoll DHCP. Welcher Zusammenhang besteht mit BootP? Schildern Sie im Detail welche Abläufe beim Leasen einer IP Adresse zu durchlaufen sind (Welche DHCP Messages und welche Optionen werden verwendet? Wie funktioniert das mit den Timern T1, T2? etc.)

Allgemein: DHCP ist ein Protokol um Host Spezifische Konfigurationen von einem Server auf einen Client zu übertragen und ist somit ein Mechanismus um Clients temporäre oder permanente Adressen zuzuweisen. Der DHCP Server empfängt die Anfrage eines Clients und sucht sich aus einem IP Pool eine Adresse heraus, und gibt sie an den Client weiter. (auf TCP/IP Basis) Der Client kann diese Adresse nun für eine bestimmte Zeit benutzen. Nach Ablauf dieser Zeit muss der Client erneut eine Adresse beantragen. Durch den automatischen Prozess verhindert DHCP einige Probleme, die sonst bei der manuellen Konfiguration auftreten könnten.

Der Client kann fragen nach: IP Adresse, Subnetz Maske, DNS Server, default TTL, max. Fragment Size, Default Gateways, ARP Cache Timeout, TCP Keepalives, Ethernet Encapsulation, ...

Verbindung zu BOOTP: DHCP benutzt den Header von BOOTP für die Übertragung der Daten (DHCP ist somit BOOTP basierend)

3 Methoden zur Adressgewinnung:

Automatisch (DHCP gibt dem Client eine permanente Adresse)

Dynamisch (DHCP gibt dem Client eine Adresse für eine bestimmte Zeit)
Manuell (Adresse wird manuell erstellt, andere Parameter übernimmt DHCP)

Leasen einer IP:

- IP Lease Request:** Wenn ein Client startet sendet er einen Broadcast an alle DHCP Server (0.0.0.0 als Source Adresse, 255.255.255.255 als Destination Adresse). Dieser Request wird in einer DHCPDISCOVER Message geschickt.
 IP Lease wird benutzt wenn TCP/IP zum ersten Mal gestartet wird, eine vom Client verlangte IP Adresse verweigert wird, der Client vorher schon eine IP Adresse geleased hat, welche jedoch abgelaufen ist.
- IP Lease Offer:** Alle DHCP Server, die den Broadcast empfangen senden dem Client eine DHCP OFFER Message welche MAC-Adresse, Offered IP Adresse, Subnet Maske, Length of Lease, Server ID enthält
- IP Lease Selection:** Wenn ein Client ein Angebot von mind. einem DHCP Server empfängt sendet er einen DHCPREQUEST ins Netzwerk um anzuzeigen, dass keine weiteren Angebote mehr akzeptiert werden. Diese Message beinhaltet die Server ID um dem einen Server anzuzeigen, dass sein Angebot akzeptiert worden ist.
- IP Lease ACK/NACK:** Im Erfolgsfall wird ein DHCPACK gesendet, welches nochmals die IP Adresse sowie andere Konfigurationsparameter enthält. Danach kann der Client TCP/IP vollständig initialisieren.

DHCP Renew: Bei der IP Vergabe wurden 2 Timer gesetzt ($T1 = 0,5 \times \text{Lease Time}$, $T2 = 0,875 \times \text{Lease Time}$). Diese werden gestartet sobald sich der Client im Bound Zustand befindet. Nach Ablauf des $T1$ wird ein Versuch gestartet eine neue IP Adresse erlangen. Schlägt dieser fehl greift $T2$ und es wird abermals nach Ablauf der Versuch gestartet eine neue IP Adresse zu erlangen. Der DHCP Server kann jetzt einfach die IP Adresser erneuern oder mittels DHCPNACK anzeigen, dass sich der Client um eine neue IP Adresse bemühen muss.

55) Geben Sie einen Überblick (Grundprinzip, Funktionsweise, Protokollabläufe, Einsatz, etc.) über das Protokoll Telnet. Was versteht man unter NVT? Was sind Telnet Commands, Optionen und Standard Functions und wozu dienen sie? Welche Portnummern werden verwendet? Was ist punkto Security zu sagen.

Telnet ist eine Methode um mit anderen Internet Hosts zu kommunizieren. Es bietet ein Standard-Interface sowie ein Terminal. Mittels Telnet kann man sich von einem lokalen Host Remote einloggen und Kommandos ausführen. Telnet bietet ein Client-Server Modell.

Basics: Telnet ist Connection-oriented und benutzt das TCP Protokoll auf **Port 23**
 Konzept des Network Virtual Terminals (NVT)
 Telnet war eine der ersten Internet Applikationen
 Telnet ist außerdem eine der populärsten Internet Applikationen da es flexibel ist, wenig Ressourcen benötigt und Telnet in jedes UNIX (sowie anderes OS) integriert ist.

Virtual Terminals: Ein Telnet Client kann das Verhalten eines realen Terminals emulieren. Intern endet jede Telnet Verbindung bei einem Network Virtual Terminal (**NVT**). Das NVT offeriert ein standardisiertes, Netzwerkweites Terminal (Printer, Keyboard, Half-Duplex Mode). Somit können verschiedene Clients die unterschiedliche Telnet's verwenden die Kommunikation auf ein gemeinsames Level übersetzen

Telnet selbst läuft allerdings im Full-Duplex Modus – aus Benutzersicht läuft Telnet jedoch nur auf Halb-Duplex (Reduzierung der Netzwerkkosten und Server-Interrupts. Der Telnet Server möchte zuerst alle Daten zum Client schicken bevor dieser weitermacht.)

Verhandlungsoptionen: Um die wenigen Möglichkeiten von NVT zu erweitern, bietet Telnet die Möglichkeit neue Optionen zu verhandeln, welche dann von den Systemen benutzt werden können.

NVT Character Set: NVT benutzt ein 8 bit Daten-Format: Trotzdem benutzt NVT den US 7 bit ASCII Code (Druckbare Zeichen + einiger Kontrollzeichen)

Interne Telnet Kommandos: Für Verhandlungs- und Signalzwecke benutzt Telnet spezielle Kommandos (8 bit lang). Die Kommandos werden mit einem Speziellen „IAC“ (Interpret as Command) prefixt. (Wenn dieser im Datenstrom vorkommt → Bytestuffing). Alle Kommunikationen werden mit diesen Kommandos geführt (Länge von 2-3 Bytes, IAC, Command, mögliches 3tes Byte). Bei weitergehenden Verhandlungen können Kommandos auch länger sein (Wird durch SB (Subnegotiation Begin) und SE (Subnegotiation End) eingeschlossen)

Standard-Funktionen: Um die Kompatibilität zu vereinfachen wurden gewisse Standard-Funktionen definiert. Jedes dieser Kommandos inintiiert eine Kontrollfunktion

Sicherheit: Telnet Clients können zu einer Vielzahl von Server-Ports Verbindung aufnehmen (Port 25: SMTP, Port 80: http,)
Telnet verschlüsselt Passwörter nicht – Sniffers !! (Daher sollte man Telnet Benutzer nie Root-Privilegien geben bzw. SSH benutzen
Einige Versionen von Telnet unterstützen „Telnet Enviroment Option“ und können angegriffen werden (Benutzer bekommen Zugriff auf das Rootverzeichnis)
Trojanische Pferde klonen Virtuelle Terminals

56) Geben Sie einen Überblick (Grundprinzip, Funktionsweise, Protokollabläufe, Einsatz, etc). über das Protokoll FTP. Was versteht man unter Virtual File und Reduktionsansatz? Wie ist die Abgrenzung von FTP zu FileServer OS? Was versteht man unter PI und DTP? Wie verläuft die Kommunikation über die Kontrollverbindung? Wie unterscheiden sich die Abläufe von normalen FTP vom passiven FTP? Welche Portnummern werden verwendet? Was ist punkto Security zu sagen? Vergleichen Sie es abschließend mit TFTP.

Grundsätzlich gibt es **2 verschiedene Methoden** Daten zum Austausch anzubieten.

- 1) Definition von virtuellen Daten welche für den Transfer in reale Daten übersetzt werden müssen. → Es müssen alle Varianten berücksichtigt werden. Die Übersetzung vom realen zum virtuellen Dateisystem muss implementiert werden (komplex). Der Vorteil in dieser Methode liegt darin, dass die virtuellen Dateisysteme leicht eine Vielzahl von realen Dateisystemen unterstützen können. (zb. ISO FTAM Protokol)
- 2) Reduktion: Extrahiert einige wenige Eigenschaften von vielen verschiedenen Formaten. (Dateitypen, Dateiorganisation, Benutzerverwaltung, Passwort, Symbolische Namen, I/O-Operationen, einige rudimentäre Betrachtungs- und veränderungsoptionen) → Bei dieser Methode ist keine Übersetzung zwischen verschiedenen Endsystemen notwendig → FTP

FTP: „Sharing by File Transfer“ – Die Dateien werden kopiert und haben danach keinen Bezug mehr zueinander. Daher kann die heruntergeladene Datei beliebig verändert werden, was den FTP-Server nicht zu kümmern braucht.

File Server OS: „Online Sharing Systems“: Erlaubt mehreren Usern eine Datei über das Netzwerk zu benutzen. Diese können die Datei direkt am Server bearbeiten (Allerdings nur immer einer !) (zb. Novell File Server, Sun NFS)

Datei-Representation durch FTP: ASCII (8 bit NVT), EBCDIC (8 bit für IBM to IBM), IMAGE (8 bit binary)

Datei-Organisation durch FTP: Dateistruktur (Strings von Bytes, Ende durch EOF)
Record-Struktur (Liste von Records, Ende durch EOR)

Transfertypen: Stream: Die Daten werden in einem kontinuierlichen Stream übertragen, EOF, EOR bezeichnen Ende der Datei

Block: Die Daten werden in Blöcke unterteilt, EOR: Ende eines Blocks, EOF: Ende der Datei. Diese Übertragungsmethode ermöglicht das Wiederaufnehmen von abgebrochenen Downloads)

Compressed: Die Daten werden komprimiert übertragen. Selbe Dateisequenz wird nur einmal übertragen, danach wird dem Client mitgeteilt wie er diese Sequenz zu wiederholen hat.

Grundlegendes: FTP basiert auf dem Client-Server Prinzip
Es wird über 2 TCP Verbindungen kommuniziert (Server-Controll-Verbindung well-known Port 21), Datenverbindung: Well-known Port 20
TCP bedeutet, dass FTP keine zusätzliche Error-Recovery benötigt
Access Protection via Username, Passwort → Wird allerdings im Klartext übertragen → Sicherheitsproblem (zb durch Sniffen)

Nach der Verbindung unterhalten sich Client und Server via Protokoll Interpreter (PI) via dem NVT Format. PI ist hierbei dafür verantwortlich, den lokalen Syntax nach NVT zu übersetzen. Der Client sendet Kommandos zum Server, dieser antwortet in die andere Richtung. Wenn ein Kommando einen Datentransfer einleitet werden ein Client DTP sowie ein Server DTP (Data Transfer Process) gestartet. Danach öffnet der Server über den Port 20 eine weitere TCP Verbindung. Wenn die Verbindung beim Startvorgang auf „Passive Mode“ eingestellt wurde öffnet der Client diese weitere TCP Verbindung (Firewall Friendly). Nach der Übertragung wird die Verbindung wieder geschlossen.

Sicherheit ist wie schon angesprochen nur relativ wenig vorhanden, da zwar Username und Passwort abgefragt werden, dies jedoch im Klartext geschieht.

Unterschied zu TFTP: TFTP ist weit weniger komplex als FTP und wird zb von BOOTP verwendet. Es wurde geschaffen um einfachste Datenübertragung zu bieten und bieten keinerlei Funktionen zum Lesen von Verzeichnissen bzw. Sicherheitsfeatures.

57) Charakterisieren Sie kurz die grundlegenden Eigenschaften von DNS. Warum erfolgt die Namensvergabe in einer baumartigen Hierarchie? Womit lässt sich diese vergleichen (Stichwort: Filesysteme)? Welche Grundregeln gelten bei der Bildung von Namen und bei der Zuordnung von Namen zu IP Hostrechner? Was versteht man unter Domain, Domain-Name und Label? Was ist ein FQDN? Welche Top-Level-Domains gibt es? Was versteht man unter IN-ADDR.ARPA? Wie ist diese aufgebaut? Wofür wird diese verwendet?

Historisches: Ursprünglich wurden alle Domainnamen unitär gespeichert. Mit dem zunehmenden Wachstum des Internets stellte sich jedoch schnell heraus, dass dies heute nicht mehr möglich ist. Daher wurde 1984 das Domain Name System (DNS) ins Leben gerufen.

DNS ersetzt eine IP Adresse eines Hosts in einen lesbaren Namen. (Hostname Resolution). Dabei geht DNS nach einer Art Baumstruktur vor. Jeder Teil der Hierarchie wird „Domain“ genannt, jeder Hierarchielevel wird „Label“ genannt → „Domain Name“. Der DNS-Baum wird durch Name Server realisiert. Jeder dieser Server nimmt sich einem Subnet des DNS-Baumes an – so genannten Zonen. (Der lokale Ort des Servers hat nichts mit dem DNS-Baum zu tun. Der DNS Baum lässt sich mit einem Filesystem (Root-Pfad) eines Computers vergleichen (zb C:\ ↔ „.“) – Es wird aber umgekehrt gelesen.

Wenn jemand einen Domainnamen in eine IP Adresse auflösen will fragt er einen DNS-Server über das DNS-Protokoll. Der Nameserver wird entweder manuell konfiguriert oder mittels DHCP ermittelt.

Grundregeln:

- Hosts mit mehreren Netzwerkadressen können über einen einzelnen Domainnamen erreicht werden.

- Hosts mit einer einzelnen IP-Adresse können über mehrere Domainnamen erreicht werden.
- Das Root-Verzeichnis des DNS-Baumes ist ein „.“
- Das Root-Verzeichnis wird durch mehrere Root-Server repräsentiert.
- Unter dem Root heißen die Domains: Top-Level-Domain, Second-Level-Domain, ... (zb. .com, .de, .at, .mil, .gov, .edu, ...)

FQDN → Fully Qualified Domain Name bedeutet alle Labels inkl. dem „.“

IN-ADDR.ARPA: Möchte man die zu einer IP-Adresse die zugehörige Domain wissen kann man auf IN-ADDR.ARPA zurückgreifen. Ohne diesen Service müsste die komplette DNS-Datenbank nach einer IP Adresse durchsucht werden. Die zu suchende IP Adresse muss allerdings verkehrt eingegeben werden. Die IN-ADDR.ARPA Datenbank ist daher nach der IP Adresse und nicht nach dem Domainnamen geordnet.

58) Wie erfolgt die Handhabung der verteilten DNS Namensdatenbank? Was versteht man dabei unter Zone-Files? Was ist SOA bzw. was kennzeichnet es? Was versteht man unter Primary und Secondary DNS Server? Was sind Master Files? Was sind Root-Hints? Wie ist Bind aufgebaut? Welche Komponenten gibt es? Wie arbeiten diese zusammen? Wie erfolgt üblicherweise die Namensauflösung, wenn der Resolver nicht unmittelbar den autoritativen DNS Server befragt (Stichwort rekursiv, iterativ)?

Die Handhabung von DNS erfolgt durch Zonen. (. dot) Ist die Hauptzone. Jeder Domaineintrag ohne einen solchen DOT ist eine relative Domain. Jeder DNS-Server kennt nur die eigenen Einträge, seine Zone Files und die Links zu den anderen Hauptdomaninservern die er in seinem Cache hat (com org ..). Kennt ein interner DNS Server eine UnterDomain nicht so leiter er die den Unterdomainserver weiter. Das **SOA (Start of Authority)** markiert die Grenzen der Zuständigkeiten der einzelnen DNS Server.

Die **Handhabung verteilter DNS** Namensdatenbanken erfolgt durch pruning (unterteilung einer domain in unterdomains mit eigenem Dns (BIND berkeley internet domain server www.foo.org > pub.foo.org)

Zone files sind eben diese unterdatenbanken

Master files enthalten die IP Einträge der für die gesuchte Domain zuständiger Nameserver

Primary DNS nur einer pro Domain er hat die Einträge in seiner Masterfile für alle Unterdomains.

Secondary DNS mehrere Server möglich sie enthalten eine Kopie des Masterfiles des Master DNS-Servers. Haben auch Autorität im Primary Bereich und werden für Redundanz und Lastausgleich verwendet. Empfohlen durch RFC 1035.

Root.hints sind Anfragen an einen Hauptdomaninserver Root NS (momentan 13)

BIND besteht aus einem Server named genannt, einer „resolver library“ und gibt Unterzonen an weitere BIND Server ab.

Aufbau: Das UserProgram macht eine Anfrage an den Resolver. Dieser fragt den ForeignNS (das äußere Netz betreffend) ab und bekommt eine Antwort die er in seinen Cache schreibt (Shared Database) sowie dem Client-Programm mitteilt. Die Shared Database wird vom internen NS durch das Masterfile refresh und umgekehrt. Der NS arbeitet andere Anfragen von anderen Resolvem von auserhalb ab und gleicht sich mit maintenance queries/responses mit anderen außerhalb liegenden NS ab.

Zusammenarbeit von **BIND u DNS** sie sprechen über Zonen miteinander **Rekursibe DNS**

Abfrage die Anfrage wird an einen Default DNS Server weitergegeben dieser forwardet, wenn sich diese Domain nicht im Zuständigkeitsbereich des Servers befindet, die Anfrage einem Root NS. zb. docs.foo.org Dieser antwortet dann mit der IP des zuständigen Hauptservers z.B. für .org. Dieser Root Server wird dann mit der Sucher nach www.foo.org befragt und gibt die IP des foo.org Servers zurück. Der www.foo.org NS wird mit der Anfrage auf docs.foo beauftragt. Schlussendlich

bekommt der Client die IP des gesuchten Servers. **Alternative DNS Abfrage** Dabei wird eine Liste von bestimmten zuständigen NS direkt an den Clienten übermittelt.

59) Charakterisieren Sie kurz die das Prinzip von Email und SMTP. Was beschreibt RFC821/822? Gehen Sie auf die Abläufe des SMTP Protokolls näher ein (SMTP Commands). Wie schaut die Struktur von Emails aus? Welche Rolle haben MUA und MTA? Welche Protokolle kommen zwischen MUA und MTA bzw. MTA-MTA zum Einsatz (Unterscheiden Sie dabei zwischen Mail-Upload und Mail-Download)? Wozu dienen POP und IMAP? Beschreiben Sie diese kurz? Welche POP Commands gibt es und wozu dienen diese? Was kann mit IMAP verbessert werden?

Email ist der am meisten genutzte Service des Internet. Jeder User kann mit jedem anderen kommunizieren. Email verwendet das Mailbox-Prinzip: Der Empfänger muss nicht online sein, um eine Email zu empfangen, sondern sie wird auf irgendeinem Server im Internet zwischengespeichert. Emailadressen werden in der Form user@domain angegeben. Das Email-Format wird in der **RFC 822** standardisiert. Email selbst wurde in seiner Basisfunktion bereits 1972 eingeführt (Achtung TCP erst 1974)

SMTP (Simple Mail Transfer Protocol) (RFC 821) dient zum Austausch elektronischer Post auf Grundlage einer TCP basierenden verbindungsorientierten Rechner zu Rechner Kommunikation. Im Nachrichtenaustausch hat SMTP verschiedene Kommandos (Client -> Server)

z.B. HELLO - Vorstellung
 MAIL - Angabe des Absenders
 RCPT - Angabe des Empfängers
 DATA - Senden der Nachrichten
 QUIT – Ende

Kommandos (Server -> Client) haben Nummern: z.B: 220 service ready.

Basiskomponenten sind der Mail User Agent (MUA, Programm zum lesen und schreiben von Emails), Sender spool-file (Jede Mail, die gesendet werden soll wird vom MUA an diese File angehängt), Mail Transfer Agent (MTA, schickt diese Mail an die Mailbox des Empfängers, z.B. mit SMTP), Mailbox (definierte File, Besitz eines Empfängers, eingehende Mails sollten hier gespeichert werden).

Zwischen **MUA** und **MTA** besteht entweder eine direkte Verbindung (Wenn sie auf demselben Rechner implementiert sind) oder es werden POP3 oder IMAP4 Protokolle zum abrufen von Mails aus der Mailbox oder SMTP zum Senden von Mails (Spoolfile) verwendet. Zwischen zwei MTAs wird in jedem Fall SMTP verwendet. MUA steht hierbei für den Mail User Agent und MTA für den Mail Transfer Agent)

POP (Post Office Protokoll) ist ein Protokoll für den Zugriff den Mail Client auf den Mail Server. POP3 setzt auf TCP auf. Seine wichtigsten Funktionen sind Login beim Mailserver, Authentifikation durch ein Passwort, Abfrage von Nachrichten auf der Mailbox des Servers sowie deren Löschung aus dem permanenten Speicher. Auf der Plattform des Mailservers muss ein SMTP und ein POP3 Server installiert sein. Pop Commands: Einloggen, Ausloggen, Nachrichten abholen, Nachrichten löschen.

IMAP (Internet Message Access Protocol) ist ähnlich dem POP, aber weiter ausgereift. Es erlaubt einem Client Zugang zu einem Mailserver zum Manipulieren von Emails und Mailboxen (inklusive erzeugen, löschen und umbenennen von Mailboxes) Nachrichten können auf dem Server gespeichert bleiben und sind dadurch ist auch der Zugriff von anderen Rechnern möglich.

60) Geben Sie einen Überblick über das Grundprinzip von WWW (Hypertext, HTML, URL, HTTP, WEB-Browser, Web-Server). Welche Methoden für Dynamic WWW auf der Browser-Seite gibt es? Schildern Sie diese kurz. Welche Methoden für Dynamic WWW auf der Server-Seite gibt es? Schildern Sie diese kurz.

WWW: Das WWW (Wor1d Wide Web) besteht aus Clients, Servern und Objekten. Die Clients werden als Web-Browser bezeichnet. Sie fordern von den Web-Servern Objekte (HTML-Dokumente) an, um diese auszuwerten. Objekte sind elektronische Dokumente und Daten jeglicher Art, insbesondere jedoch Hypertext- und Hypermedia-Dokumente. Das WWW integriert weitere Internetdienste mit Hilfe der einheitlichen Benutzerschnittstelle des Browsers.

URL: Mit dem Uniform Resource Locator (URL) gibt es einen eindeutigen Zeiger auf eine Seite im Internet. HTML (Hypertext Markup Language) ist eine Text beschreibende Sprache. Zum download von HTML Dokumenten wird das Hypertext Transfer Protocol (HTTP, Server hinter Port 80 erreichbar) verwendet.

Hypertext- und Hypermedia-Dokumente enthalten mehrere Komponenten. Hypertext ist Text, der durch Links (Verweise) ergänzt wird. Ein Link ist ein Verweis auf eine andere Textstelle oder ein anderes Dokument (Objekt). Links können insbesondere verweisen auf:

- andere (Text-)Stellen in demselben Dokument (Objekt)
- Objekte (Dateien) innerhalb eines Dateisystems oder Computers
- Objekte (Dateien, Dokumente) in einem Netz
- Hypermedia enthält zu Text und Links zusätzlich multimediale Anteile wie Grafik, Bilder (Bewegt- und Festbilder) sowie Sprache (bzw. allgemein Töne)

Um das Verhalten des Browser **dynamischer** zu machen gibt es auf **Client-Seite**

- JavaScript / JScript (oder ECMA-Script, HTML Erweiterung und Programmiersprache von NetScape, JavaScript Programme werden direkt in das HTML-Dokument geschrieben, Das Prgramm wird im Client-Browser ausgeführt und kann das Verhalten von Formen, Knöpfen und Textelementen kontrollieren, JScript ist die Microsoftversion von JavaScript),
- JavaApplets (Beruhen auf der C++ Sprache von SUN entwickelt, Programm wird in der Java Virtual Machine (JVM) ausgeführt außerhalb derer kann der Programmcode nichts ausführen),
- ActiveX (Microsoft Version von JavaApplets) oder
- Flash (Software welche zur Animation von Web-pages verwendet werden kann). Mittels swf-File wird die gesamte Animation auf den heimischen Rechner geladen und dort ausgeführt.

Auf **Serverseite** gibt es dafür

- Common Gateway Interface (CGI, erlaubt dem Server Programme auszuführen, die auf den User reagieren, läuft über Interpreter, populärster ist Pracical Extraction and Report Language, PERL)
- Hypertextr Preprocessor (PHP, Optimierung von CGI bzw. PERL)
- Active Server Pages (Microsoft Version von PHP)
- Servlets (im Prinzip JavaApplets, die vom Server ausgeführt warden)
- Server-Sides-Include (SSI, erlaubt einem Web-Server, der Java versteht, teile eines HTML in eine andere Sprache zu übersetzen (Servlet) und dem Client zu schicken) oder
- Java Server Pages (JSP)

Beschwerden, Wünsche, Anregungen bitte an michael.dallinger@gmx.net