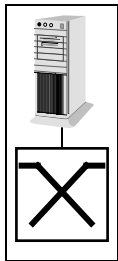


# Netze und Protokolle für das Internet



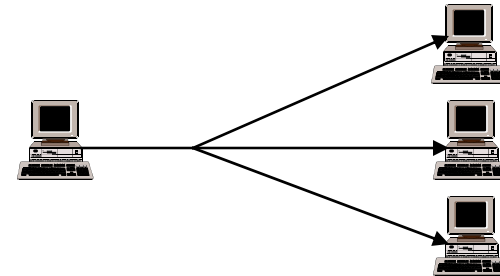
## 10. Transportprotokolle zur Gruppenkommunikation

# Inhalt

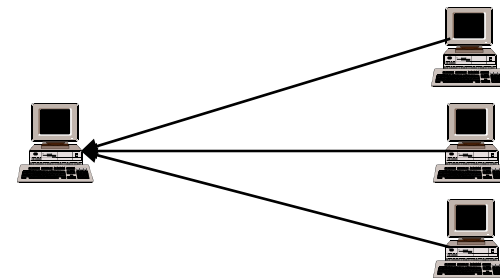
- Kommunikationsformen
- Zuverlässigkeitsklassen
- Anwendungen von zuverlässigem Multicast
- TCP-Eigenschaften
- Multicast Transport Protocol
- Reliable Multicast Protocol
- Scalable Reliable Multicast
- Baum-basierte Ansätze
  - Reliable Multicast Transport Protocol
  - Reliable Multicast proXies
  - MTCP
  - Active Reliable Multicast
- Vorwärtsfehlerbehebung
- Staukontrolle
- IETF WG Reliable Multicast Transport

# Kommunikationsformen

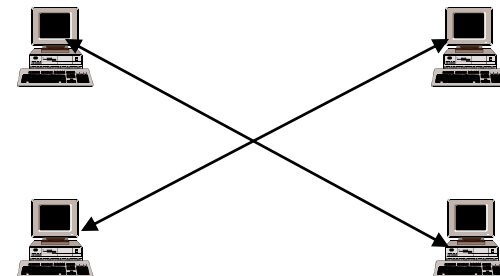
- 1:1 Unicast
- 1:n Multicast



- n:1 Concast



- n:m Multipeer



# Zuverlässigkeitsklassen

- unzuverlässig
  - keine Auslieferungsgarantien
  - ggf. Übertragung redundanter Information
- halbzuverlässig
  - statistisch zuverlässig
    - Schwellwert gibt an, wieviele Gruppenmitglieder die Daten innerhalb eines bestimmten Zeitintervalls erhalten müssen.
    - Gruppengrösse muss bekannt sein.
  - k-zuverlässig
    - garantiert den Empfang der Daten an k Gruppenmitglieder ( $k \leq n$ ,  $n =$  Gruppengrösse)

Nach Ablauf eines Zeitintervalls können Korrekturmaassnahmen ergriffen werden, z.B. Übertragungswiederholung
- zuverlässig
  - Daten werden fehlerfrei, ohne Duplikate und in der korrekten Reihenfolge bei allen Gruppenmitgliedern ausgeliefert.

# Anwendungen von zuverlässigem Multicast

- Push-Technologien
- Software-Aktualisierungen
- Cache-Aktualisierungen
- Verteiltes Rechnen
- Computer-Supported Cooperative Work (CSCW)
- Application Sharing

# Shared Whiteboard

The screenshot shows a shared whiteboard application. At the top, there are several control panels: 'Grabber Panel', 'Video Panel', 'Rate Panel', and 'Info ...'. The 'Grabber Panel' includes volume and microphone gain sliders, a 'Release audio' button, and 'Stop video', 'Push to talk', and 'Quit' buttons. The 'Video Panel' shows video statistics like '4.0 f/s', '9.2 kbps', and '0.0 % loss', along with 'Zoom x 2', 'Zoom x 1/2', 'Color', and 'Dismiss' buttons. A list of participants is visible, including 'turlletti@jerry.inria.fr', 'cdiot@pussy.inria.fr', and 'tbraun@merlot.inria.fr'. The main whiteboard area contains a text document with the following content:

SECTION Sending H.261 video over the Internet

To send H.261 video using standard UDP datagrams, a packetization scheme of H.261 video stream has been described in an Internet Draft [turlletti94] submitted to the Audio Video Transport (AVT) Working Group at the IETF. RTP [Shulzrinne94] is used over UDP to achieve multiplexing. The scheme proposed takes care of the hierarchical structure of the H.261 coding. The bitstream produced by a standard H.261 coder includes forward error correction (FEC). The FEC is over 492 bit blocks of the encoded bit stream and it bears no relation to the hierarchical structure of H.261, i.e. picture GOB, MB layers. So, to be conform to the ALF philosophy and in order to break the bitstream into units which do not have any dependencies on other parts of the bitstream, the FEC is removed. The smallest unit of data that has the less dependencies on other parts of the bitstream (ADU) is the GOB. Indeed, MBs have addresses relative to preceding MBs, and quantizer and motion compensation vector of a MB may depend on a previous MB within the same GOB. The output data flow generated by an H.261 coder is intrinsically VBR. It depends on the quality of the video camera, the type of the images being encoded which is function of the movement, the scene structure, the scene lighting, etc. Most of the time, GOB information fits in a packet. But when there are very few changes in the scene, several GOBs can be grouped inside a same packet to decrease the packet sending rate and avoid network congestion. A small header is added to each packet to encode information required to decode out of orders ADU received and to efficiently depacketize segmented GOBs received. As part of the MICE (Multimedia Integrated Conferencing for Europe) project [Kirstein93] {footnote: See URL <<http://www.cs.ucl.ac.uk/mice/mice.html>>}, this packetization scheme has been used by some commercial hardware codecs (GPT, Bitfield) allowing interoperability between them and IVS.

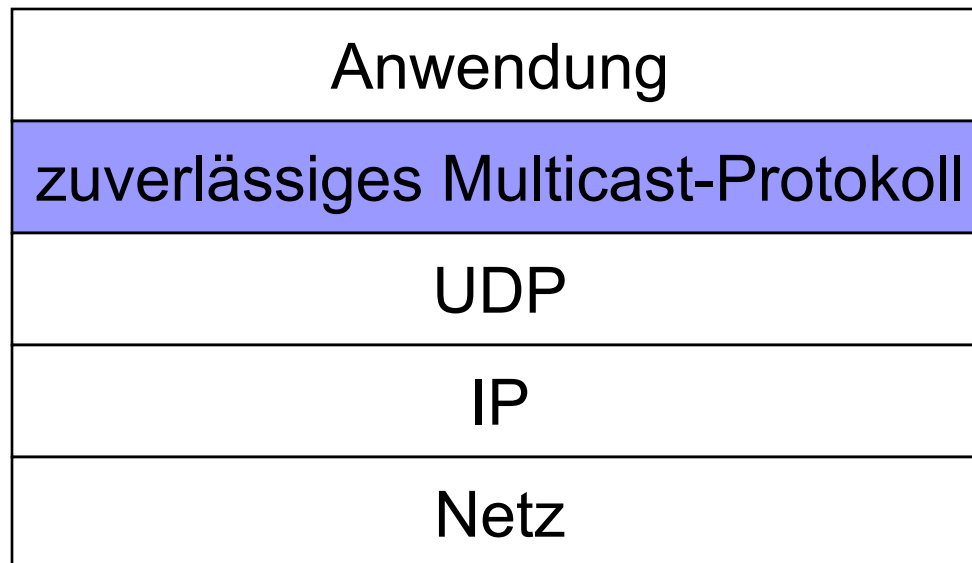
Handwritten annotations in blue and red are present:

- where can I find these papers ?  
on zenon.inria.fr:rodeo/ivs/papers (by anonymous ftp)
- A GOB is a Group of Blocks
- what size exactly ?  
Currently 4 bytes

On the right side, there are three video windows showing participants: 'turlletti@jerry.inria.fr CIF', 'cdiot@pussy.inria.fr CIF', and 'tbraun@merlot.inria.fr CIF'. A toolbar with various drawing tools and a 'Quit' button is also visible.

# Zuverlässige Multicast-Kommunikation im Internet

- TCP ist nur für 1:1-Verbindungen geeignet
- Multicast-Unterstützung im Internet durch UDP
- UDP ist unzuverlässig und besitzt keine Stau- bzw. Flusskontrolle.
- Transportprotokolle im Internet sollten TCP-Eigenschaften aufweisen, besonders hinsichtlich Staukontrolle → TCP-Freundlichkeit
- Multicast-Transportprotokolle setzen meist auf UDP auf und sind in Anwendungen integriert.



# TCP-Eigenschaften

- Flusskontrolle mit Fenstermechanismus
- Staukontrolle mit Slow Start Mechanismus
- Fehlerkontrolle
  - Folgenummern
  - Prüfsumme
  - Quittierungsnummern
  - Übertragungswiederholung
- Reihenfolgetreue

# Klassifikation zuverlässiger Multicast-Transportprotokolle

- Sender-initiierte Protokolle
  - Quelle verwaltet Informationen über alle Empfänger
  - Empfänger senden ACKs und NACKs
  - Probleme: Sender muss Empfänger kennen, ACK-Verarbeitung
  - Beispiele: XTP, NETBLT
- Empfänger-initiierte Protokolle
  - Verantwortung für Zuverlässigkeit beim Empfänger
  - Anforderung von Übertragungswiederholungen durch NACKs
  - Beispiel: SRM
- Baum-basierte Protokolle
  - Unterteilung der Empfänger in Teilgruppen, Baumstrukturen
  - Beispiel: RMTP
- Ring-basierte Protokolle
  - ausgezeichnete Station (Token-Halter) zur Generierung von ACKs, Token-Weitergabe
  - Beispiel: RMP

# Multicast Transport Protocol

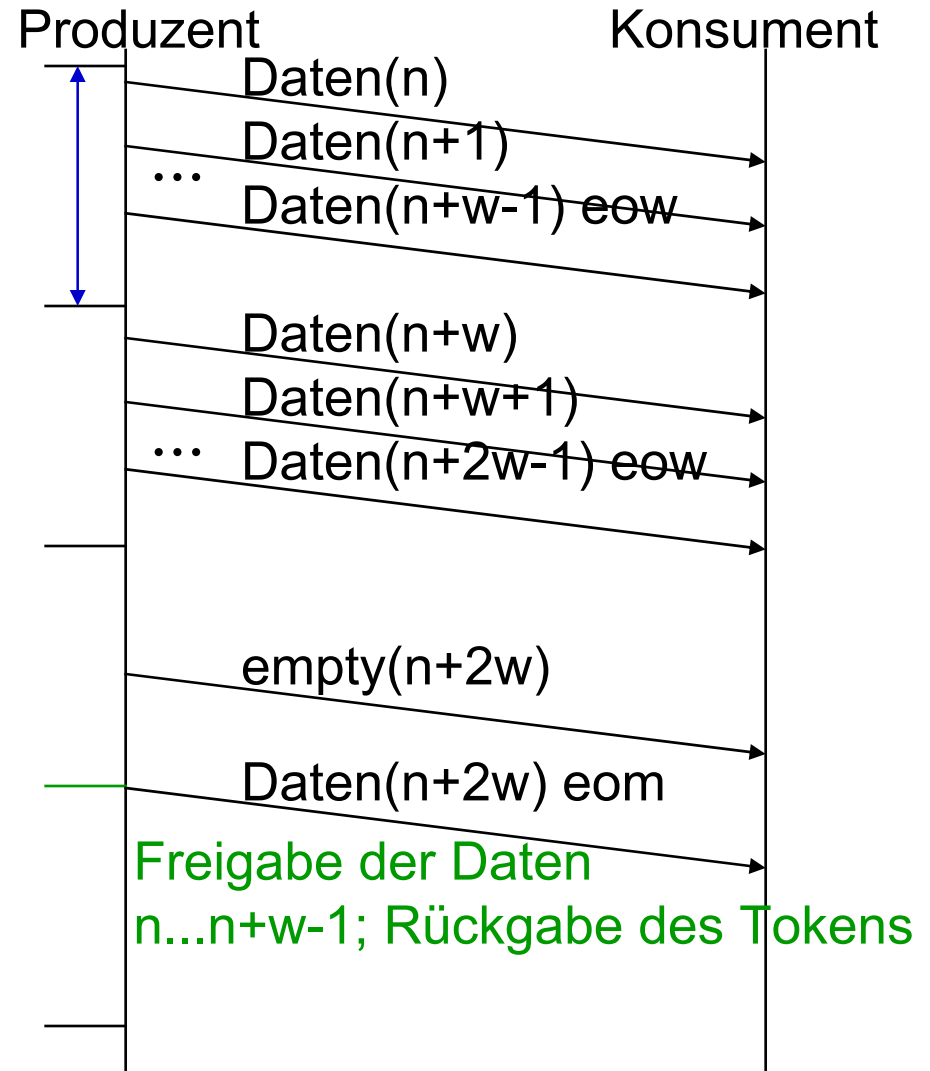
- halbzuverlässiger Multipeer-Dienst mit Ordnungserhaltung
- Rollen der Gruppenmitglieder
  - Master
    - entscheidet über Aufnahme neuer Mitglieder
    - vergibt Senderecht
    - überwacht zuverlässige Datenübertragung und Ordnungserhaltung
    - kann auch als Produzent wirken
  - Produzent (Sender + Empfänger)
  - Konsument (Empfänger)
- Kommunikationsgruppe = Web
  - Aufbau des Webs: Senden von JOIN-REQUEST an Multicast-Gruppe
  - Master antwortet per Unicast mit JOIN-CONFIRM oder JOIN-DENY
  - Bei ausbleibender Antwort kann ein Teilnehmer die Master-Rolle übernehmen.
  - Freiwilliges oder erzwungenes Verlassen des Web
  - Master löst Web mit QUIT-REQUEST auf.

# Vergabe der Senderechte bei MTP

- Produzent darf Daten nur dann versenden, wenn er im Besitz des Senderechts ist.
- Produzent muss beim Master per Unicast Senderecht anfordern.
- Master erteilt Senderecht (Token) unter Angabe einer Nachrichten-Sequenznummer.
- Pakete einer Nachricht werden mit Paket-Sequenznummer gekennzeichnet.
- Produzent gibt Token explizit durch Setzen von End-of-Message-Flag in letzter Nachricht frei.

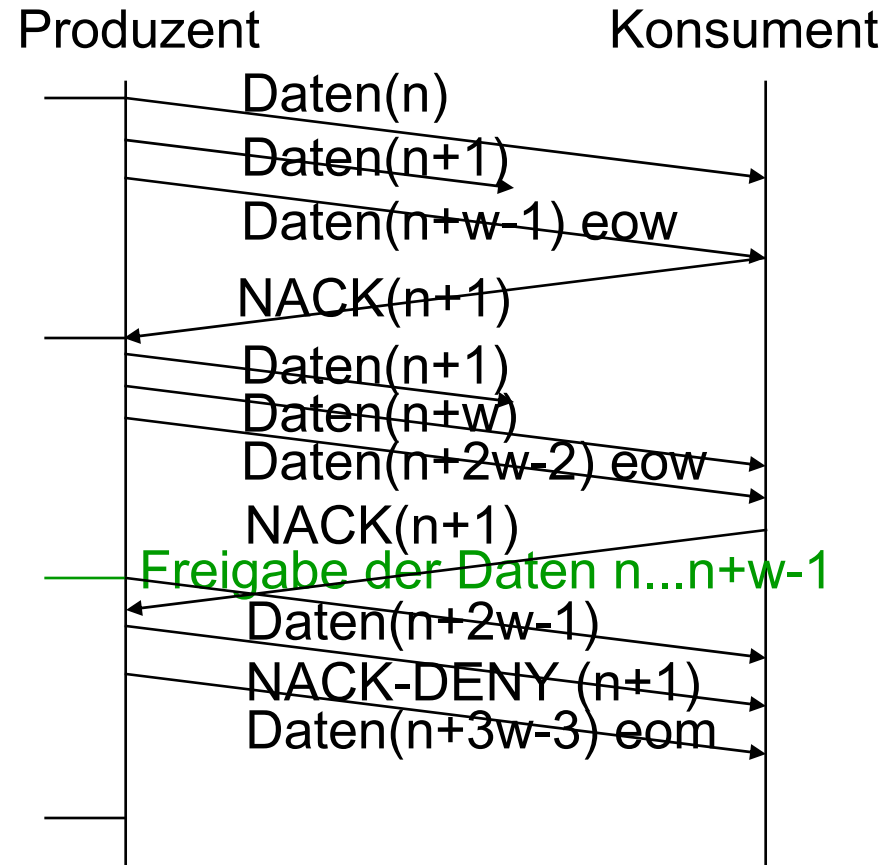
# MTP-Datentransfer

- Vereinbaren von Parametern beim Aufbau eines Webs
  - **Heartbeat**
    - mindestens ein Empty-Paket pro Zeitintervall
  - **Window**
    - Anzahl der Pakete, die in einem Heartbeat-Intervall gesendet werden dürfen
    - Beispiel:  $w=3$
  - **Retention**
    - Anzahl der Heartbeat-Intervalle, während der ein Produzent gesendete Nutzdaten für Übertragungswiederholungen bereit halten muss
    - Beispiel:  $r = 2$



# MTP-Fehlerkontrolle

- selektive Übertragungswiederholung
- Retention-Wert stellt Zuverlässigkeitsmass dar.
- Beispiel:  $r=1$
- Übertragungswiederholung (per Multicast) unterliegen der Flusskontrolle (Ratenkontrolle)
- Problem: Quittungsimplosion



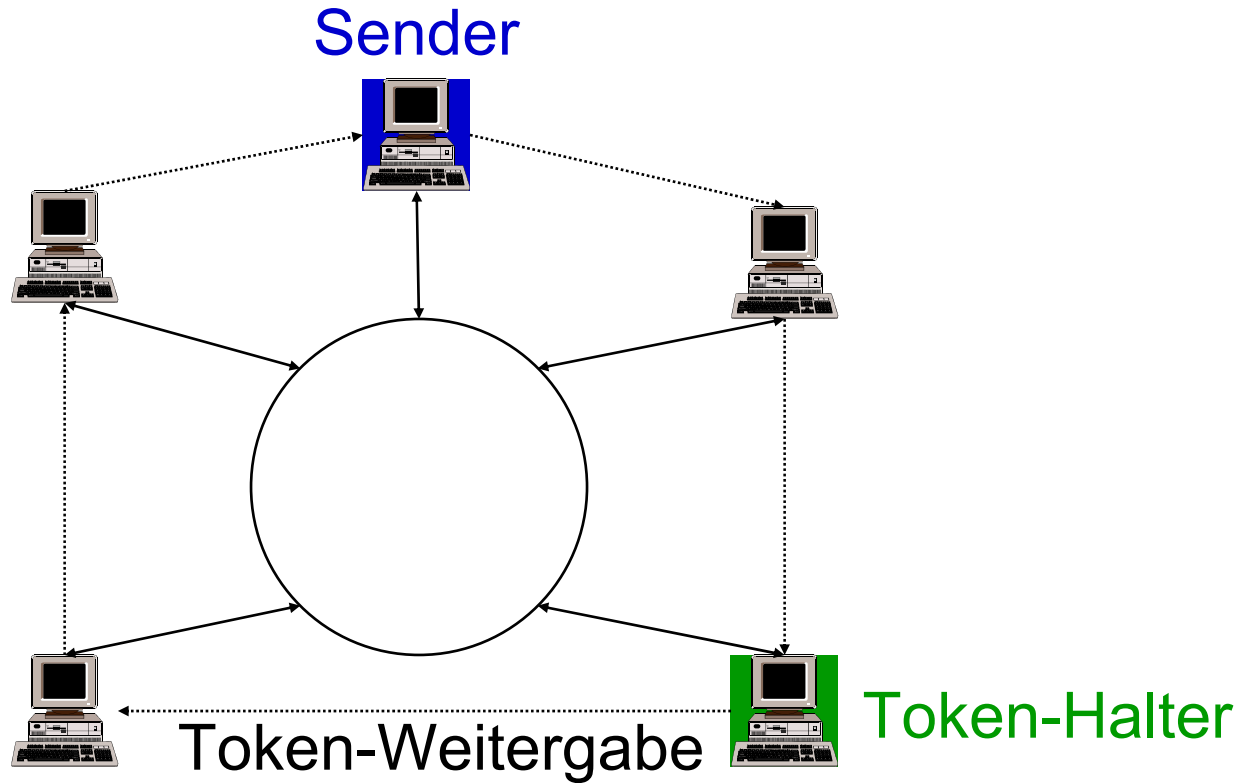
# MTP: Ordnungs- und Konsistenzzerhaltung

- Nachrichten-Sequenznummer legt Auslieferungsreihenfolge fest.
- Gruppenmitglieder sollten Daten einheitlich ausliefern  
→ Konsistenz
- Master entscheidet, ob eine Nachricht gültig ist oder nicht, d.h. nur beim Master vollständig eingetroffene Nachrichten werden als gültig erklärt.
- Problem: Master erhält Nachricht, Empfänger wegen zu kleinem Retention-Intervall aber nicht.
- Zustände der Nachrichten beim Master
  - akzeptiert
  - ausstehend
  - zurückgewiesen
- Statusvektor (Teil jeder Nachricht) enthält Zustand der letzten 12 Nachrichten.
- Neue Nachricht darf nur gesendet werden, wenn älteste Nachricht nicht mehr den Zustand „ausstehend“ hat.

# Reliable Multicast Protocol

- zuverlässige und ordnungserhaltende n:m-Kommunikation
- Anordnung von Gruppenmitgliedern in Ring
- Senden durch Nicht-Gruppenmitglieder über Proxy-Mitglied
- Senden von positiven / negativen Quittungen (ACK/NACK) per Multicast
- Ausgezeichnetes Mitglied (Token-Halter) hat Aufgabe, Daten von mehreren Sendern zu serialisieren und positive Quittungen zu erzeugen.
- Token rotiert zwischen Ring-Mitgliedern (→ Fehlertoleranz)
- dynamische Aufnahme und Löschen von Ring-Mitgliedern
- verschiedene Auslieferungsdienste

# RMP-Szenario



# RMP-Dateneinheiten und Quittungen

- Dateneinheiten enthalten Tupel
  - Sender-ID
  - Sequenznummer des Senders
  - gewünschter Auslieferungsdienst
- Quittungen enthalten
  - globale Sequenznummer (Zeitstempel)
    - Token-Halter ordnet jeder quittierten Dateneinheit eine globale Sequenznummer zu.
  - Tupel der quittierten Dateneinheiten
  - Adresse des bisherigen und nächsten Token-Halters

# RMP-Protokoloperationen

- Senden
- positive Quittierung (ACK)
  - Multicast durch Token-Halter
  - kann mehrere Pakete verschiedener Sender quittieren
  - enthält Zeitstempel
- negative Quittierung (NACK)
  - Mitglied mit fehlenden Daten sendet NACK per Multicast
- Übertragungswiederholung
  - durch Token-Halter oder anderen Knoten
- Auslieferung
  - abhängig von Auslieferungsdienst
- Token-Weitergabe
  - mit Senden eines ACK
  - Neuer Token-Halter muss alle Nachrichten erhalten haben.
- Wechseln der Mitgliedschaft
  - List Change Request
  - Token-Halter erzeugt neue Liste und bestätigt Aufnahme

# RMP-Auslieferungsdienste

- unzuverlässig
  - einmalige, mehrmalige oder keine Auslieferung, ungeordnet
- zuverlässig
  - mindestens eine Auslieferung, aber nicht geordnet
- Quellen-geordnet
  - mindestens 1 Auslieferung, in der gleichen Reihenfolge wie von einem Sender erzeugt, keine Ordnung zwischen den Sendern
- Total-geordnet
  - Auslieferungsreihenfolge von verschiedenen Sendern ist bei allen Empfängern identisch.
- k-elastisch
  - totale Ordnung, Auslieferung bei mindestens k Empfängern
- Mehrheits-elastisch
  - K-elastisch mit  $k > N+1/2$ , N: Anzahl der Empfänger
- Total-elastisch
  - K-elastisch mit  $k = N$

# RMP-Auslieferungsoperationen

- ungeordnet
  - sofortige Auslieferung nach Erhalt einer Nachricht
- Quellen-geordnet
  - Pakete einer Quelle werden ausgeliefert sobald alle Pakete mit niedrigerer Sequenznummer empfangen wurden.
- Total-geordnet
  - Pakete werden ausgeliefert, wenn alle Pakete mit kleinerem Zeitstempel ausgeliefert wurden.
- k-elastisch
  - Pakete mit kleinerem Zeitstempel wurden ausgeliefert und Token wurde k-mal weitergegeben
- Mehrheits-elastisch
  - Pakete mit kleinerem Zeitstempel wurden ausgeliefert und Token wurde  $N/2$ -mal weitergegeben
- Total-elastisch
  - Pakete mit kleinerem Zeitstempel wurden ausgeliefert und Token wanderte einmal um den Ring

# Scalable Reliable Multicast

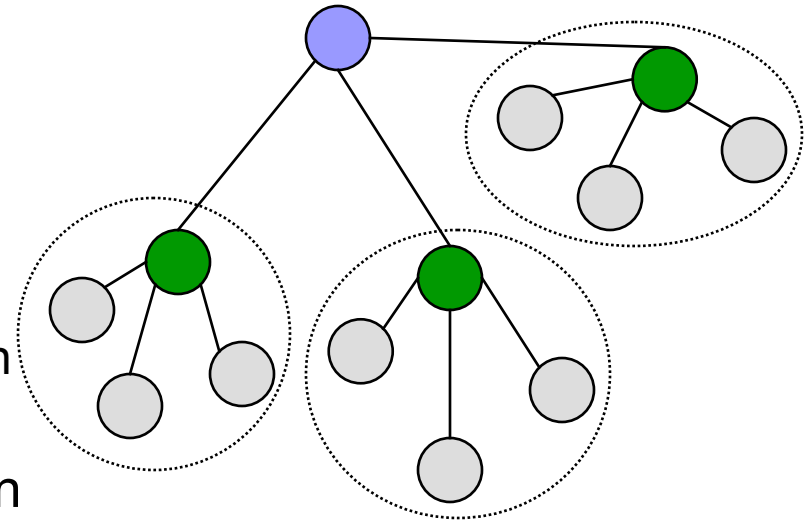
- Menge von in Anwendung zu integrierende Funktionen, z.B. shared whiteboard
- Komponenten
  - anwendungsabhängig: eindeutige Bezeichnung der Dateneinheiten
  - anwendungsunabhängig: Kontrollalgorithmen
- Anwendung muss Ordnung selbst herstellen.
- Empfänger sind für die zuverlässige Zustellung selbst verantwortlich.
- Ratenbasierte Flusskontrolle durch Sender
- Übertragungswiederholungen sollten durch nächsten benachbarten Empfänger erfolgen.
- Protokollnachrichten
  - Repair Request
  - Repair (Übertragungswiederholung)

# SRM: Fehlererkennung und -behebung

- Empfänger erkennen nicht erhaltene Pakete anhand von Lücken im empfangenen Sequenznummernbereich sowie durch periodisch gesendete Statusnachrichten (inkl. höchste empfangene Sequenznummer und Zeitstempel) der anderen Gruppenmitglieder
- Repair Requests werden nach Timeout  $[c_1 \cdot d_R, (c_1 + c_2) \cdot d_R]$  gesendet,  $d_R$  = geschätzte Einwegverzögerung zwischen Sender und Empfänger R
- Beim Empfang einer Repair-Request-Nachricht von Empfänger X wählt Empfänger Y einen Timeout  $[c_3 \cdot d_{XY}, (c_3 + c_4) \cdot d_{XY}]$ ,  
 $d_{XY}$  = geschätzte Einwegverzögerung zwischen X und Y
- Bei Ablauf des Timeout ohne Empfang einer Repair-Nachricht wird Repair per Multicast gesendet.
- Ziel: 1 Repair, 1 Repair-Request

# Reliable Multicast Transport Protocol

- Protokoll für einen Sender
- Aufbau einer Hierarchie zum
  - Reduzieren von ACK/NACK-Nachrichten
  - Reduzieren von Verzögerungen durch lokale Übertragungswiederholungen
- Empfänger werden in lokale Regionen gruppiert.
- Jede Region besitzt ausgezeichneten Empfänger (designated receiver, DR), welcher die lokale Region repräsentiert.
- mehrere Hierarchieebenen möglich
- Sender und DRs senden Kontrollnachrichten mit gleichen TTL-Werten  
→ Auswahl der DRs mit grösstem TTL-Wert

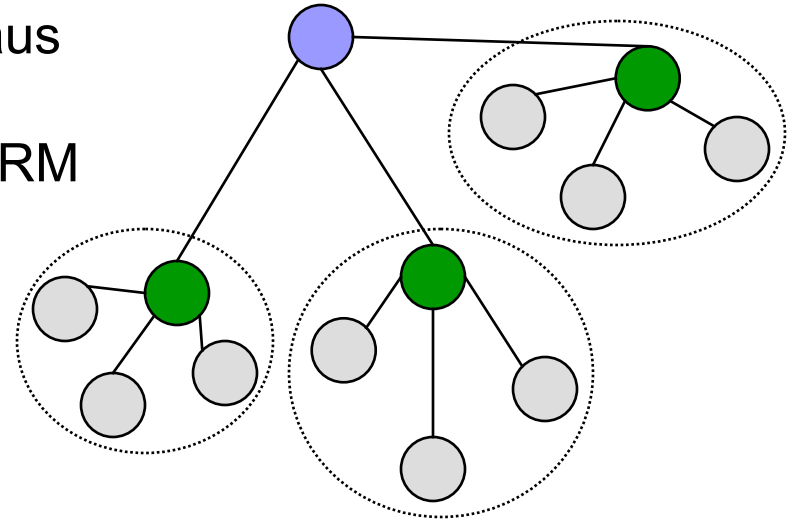


# RMTP-Protokoloperationen

- Empfänger senden periodisch Statusnachrichten zum DR
- DR sendet Statusnachrichten (Kombinationen von ACK / NACK) zum Sender
- Statusnachrichten enthalten Sequenznummer des ersten nicht erhaltenen Pakets + Bitvektor über Status der folgenden Pakete
- Rate der Statusnachrichten hängt von RTT zwischen Empfänger und DR bzw. zwischen DR und Sender ab.
- Lokale Übertragungswiederholungen nach Timeout per Unicast / Multicast abhängig von Empfänger-Anzahl
- Raten-/Fenster-basierte Flusskontrolle
- Empfänger bzw. DR kann auf bestimmte Daten verzichten, z.B. nach Beitritt oder Auflösung einer Netzpartitionierung

# Weitere Baum-basierte Ansätze

- Reliable Multicast proXies
  - RMXs bilden Spanning Tree (vgl. Bridges)
  - RMXs tauschen Daten über TCP aus  
→ Staukontrolle
  - Lokale Multicast-Verteilung über SRM
- MTCP
  - Zwischenknoten (Sender Agents) melden Staukontrollinformationen stromaufwärts
    - Congestion Window (cwnd):  
Wert basierend auf Schätzung der minimale Bandbreite
    - Anzahl noch nicht quittierter Daten von stromabwärts liegenden Knoten (twnd)
  - Aggregation:  $\min(cwnd_i)$ ,  $\max(twnd_i)$



# Active Reliable Multicast

- Ansätze
  - Active Reliable Multicast
  - Reliable Multicast Active Network Protocol
- Router entlang des Multicast-Baums
  - speichern Daten im lokalen Cache
  - aggregieren ACKs
  - unterdrücken NACKs
  - führen lokale Übertragungswiederholungen aus.
- Code: in-band oder out-of-band

# Vorwärtsfehlerbehebung

- Die meisten Multicast-Transportprotokolle basieren auf Übertragungswiederholung (Automatic Repeat Request)
- Alternative: Vorwärtsfehlerbehebung (Forward Error Control)
  - $k$  von  $n$  Paketen einer Nachricht sind zum Wiederherstellen der Nachricht notwendig
- Kombinationsmöglichkeit von FEC und ARQ
  - Empfänger empfängt  $m < k$  Pakete und fordert  $j$  Pakete erneut an.
  - Sender wiederholt mindestens  $j$  Pakete
  - Übertragungswiederholungen können verschiedene fehlerhafte Pakete reparieren.
- Digitale Fontänen
  - Permanentes Senden von (verschiedenen) redundanten Paketen
- Asynchronous Layered Coding
  - Senden von redundanten Paketen über verschiedene Kanäle
  - Auswahl der Kanäle durch Empfänger anhängig von Stausituation

# Staukontrolle

- TCP-Staukontrolle kann mit einer mathematischen Beziehung approximiert werden.
  - Datenrate wird berechnet als Funktion der
    - Anzahl quittierter Pakete
    - Paketumlaufzeit
    - Retransmission Timeout

→ TCP Friendly Multicast Congestion Control (TFMCC)
- Kombination mit ALC
  - Auswahl von Kanälen durch Empfänger um gleiches Verhalten wie TCP-Staukontrolle zu erreichen.
- Router Paketfilterung
  - Empfänger senden Stauberichte stromaufwärts
  - Router filtern ausgehende Pakete

# IETF WG Reliable Multicast Transport

- Ein einziges zuverlässiges Transport Protokoll für alle möglichen Anwendungsszenarien erscheint nicht sinnvoll.
- daher: Definition von flexibel verwendbaren Komponenten („Building Blocks“)
- spezifizierte Komponenten für
  - Forward Error Control
  - Layered Coding Transport
  - Wave and Equation Based Rate Control
- NACK-Oriented Reliable Multicast Protocol