



# **IPTV Using Peer-to-Peer Application Layer Multicast or *IPTV - Internet killed the TV Star***

Tobias Hoßfeld, Rastin Pries  
[hossfeld,pries]@informatik.uni-wuerzburg.de  
*www3.informatik.uni-wuerzburg.de*

# Motivation

---

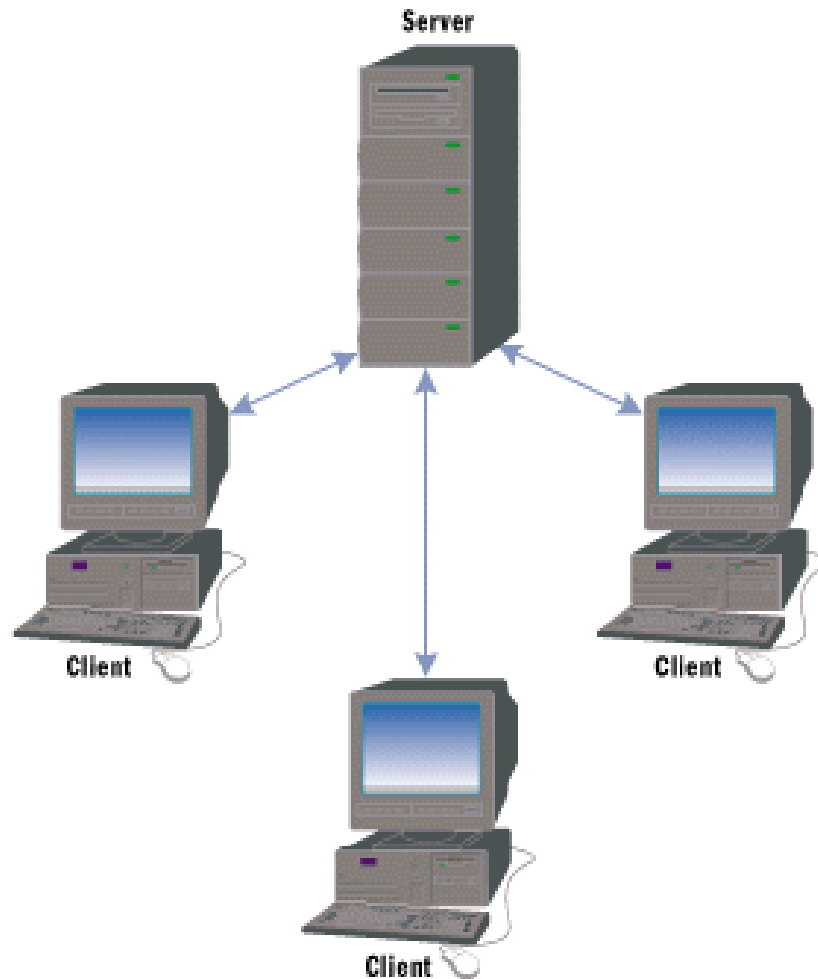
- ▶ P2P Technologie erfolgreich im Internet eingesetzt
  - BitTorrent, eDonkey: effizienter Austausch großer Dateien möglich in Flash Crowd Szenarien, **Skalierbarkeit**
  - Skype VoIP: **intelligente**, adaptive Mechanismen (Codec, Bandbreite, Application Layer Routing), Quality of Experience
  - Distributed Hash Tables: skalierbare, deterministische Suche, **Selbstorganisation** der Peers in logischen Strukturen
- ▶ Warum?
  - Popularität durch kostenlose Applikationen, viele Nutzer
  - Ressourcen der Nutzer werden in System eingebunden (Bandbreite, Speicherkapazitäten)
  - Wachsender Breitbandzugang (DSL), Tarifmodelle (Flat Rate)
- ▶ **Wo kann man P2P in IPTV einsetzen? Theorie? Praxis?**

# Agenda

---

- ▶ Client/Server und P2P
- ▶ Übersicht P2P basiertes IPTV
- ▶ Beispiel:
  - Application Layer Multicast für Realtime Streaming
  - ZIGZAG Protokoll
- ▶ (P2P-) IPTV in der Praxis
- ▶ Zusammenfassung

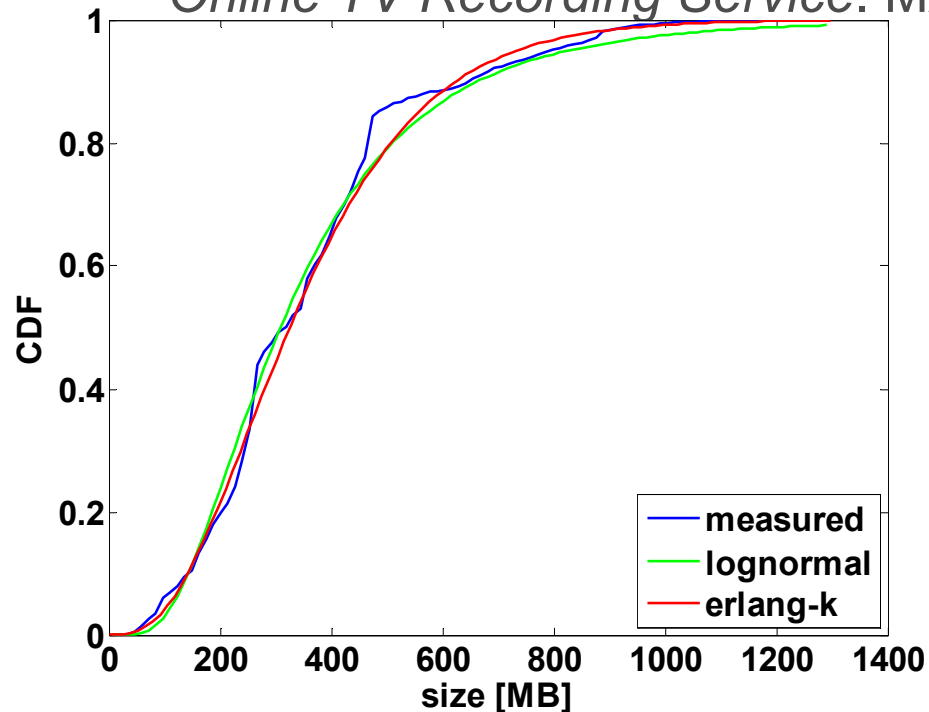
# Probleme bei Client-Server-Modell



- ▶ Skalierbarkeit
  - Mehr Dienste: Multimedia, Video
  - Mehr Teilnehmer
  - Mehr Ressourcenbedarf: Bandbreite, Speicher, CPU
- ▶ QoS und QoE
  - Jitter bei Videostreaming
  - Überlast (bei Flash Crowds)
- ▶ Flexible, adaptive, kostengünstige Netzelemente
- ▶ **Peer-to-Peer**

# Beispiel: Festplattenkapazität und Bandbreite

- ▶ Tobias Hoßfeld and Kenji Leibnitz. *Modeling and Evaluation of an Online TV Recording Service*. MAMA2007 @ SIGMETRICS '07

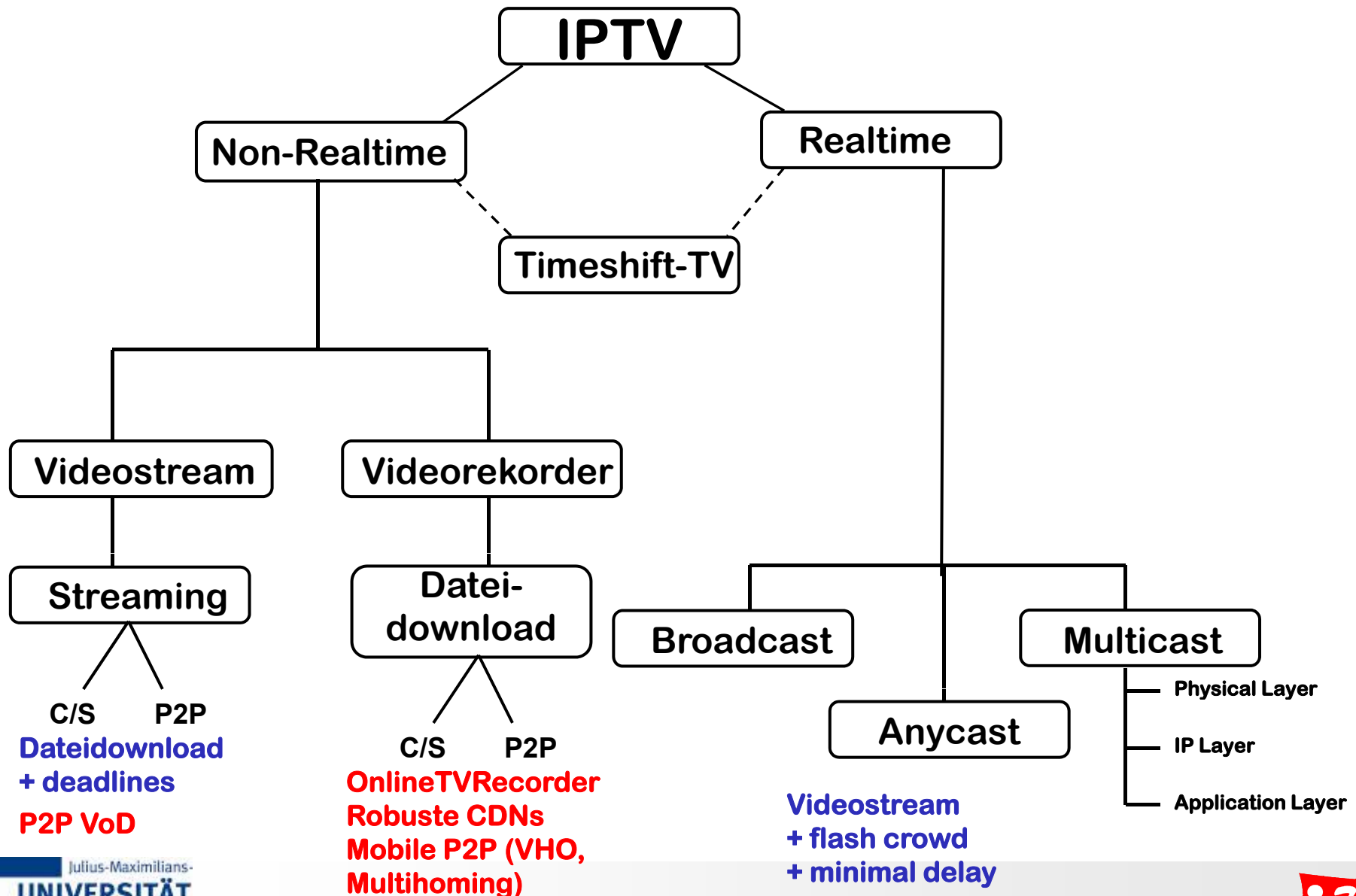


- ▶ gemessen: 8.8 MB / Minute
- ▶ 20 Sender, 1 Woche
- ▶ Kapazität: 1.7748 TB
- ▶ Pro Peer: 0.7395 MB
  
- ▶ Bandbreite: 2.8160 TByte/s (alle schauen gleichzeitig)

- ▶ Goldmedia:

- ▶ Bisher haben 2,4 Mio. Menschen in Europa IPTV abonniert.
- ▶ Bis 2010 in Deutschland mehr als 1,3 Mio. Haushalte mit IPTV; weltweit bis zu 60 Mio. Menschen.

# Übersicht über IPTV

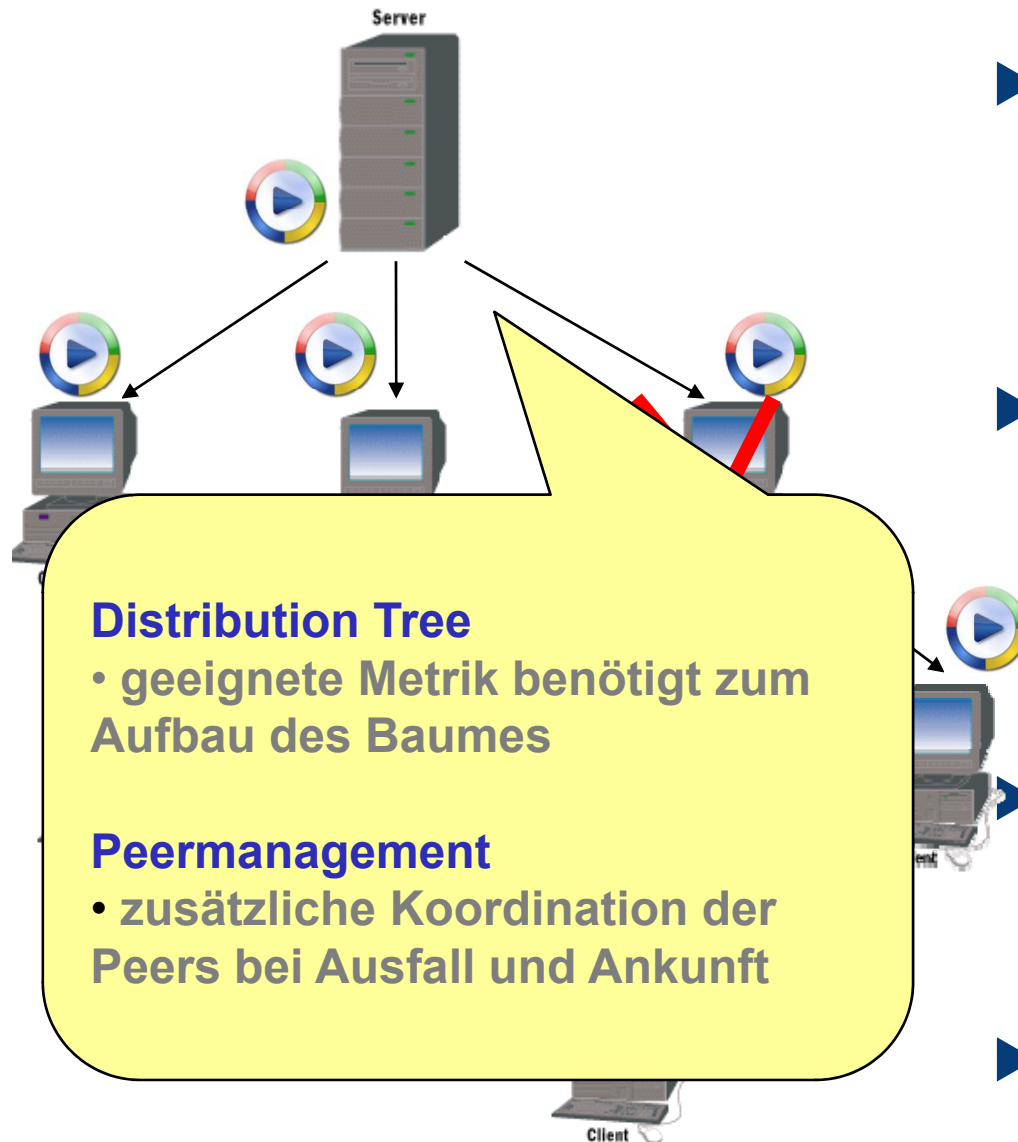


# (Near) Realtime IPTV mit P2P

---

- ▶ Warum über Internet? Laut iSuppli: „Die Fernsehunterhaltung wird um Interaktivität, Personalisierung und die Integration von Sprach- und Datendiensten bereichert.“
- ▶ Warum P2P? Skalierbarkeit, Flexibilität, Selbstorganisation,...
- ▶ Multicast auf unteren Schichten erreicht nicht alle Teilnehmer
- ▶ IP-Multicast: nicht unterstützt; Änderung der Router
- ▶ Application Layer Multicast:
  - realisiere Multicast auf Layer 7 (unabh. von darunter-liegendem physikalischem Netz)
  - Peers stellen Ressourcen wie Speicher, Bandbreite, CPU zur Verfügung
  - **Challenge:** robuste, effiziente Koordination der Peers

# P2P-basiertes IPTV

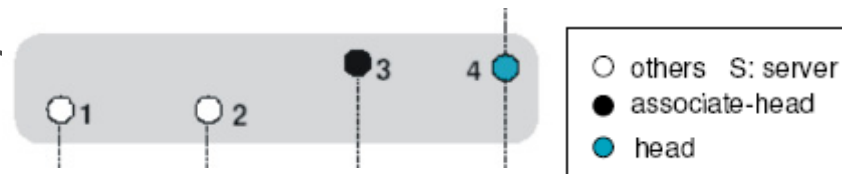


- ▶ Zeitliche Verzögerung
  - kleine Baumhöhe benötigt
  - Berücksichtigung von Delays zwischen Peers
- ▶ Bandbreite:
  - Peers mit hoher Bandbreite näher an Wurzel
  - mobile Peers: wechselnde Kapazitäten durch VHO
- ▶ Zuverlässigkeit:
  - Zuverlässige näher an Wurzel
  - mobile Peers und Churn
- ▶ Ausfall einen Knoten bringt Probleme: **Peermanagement**



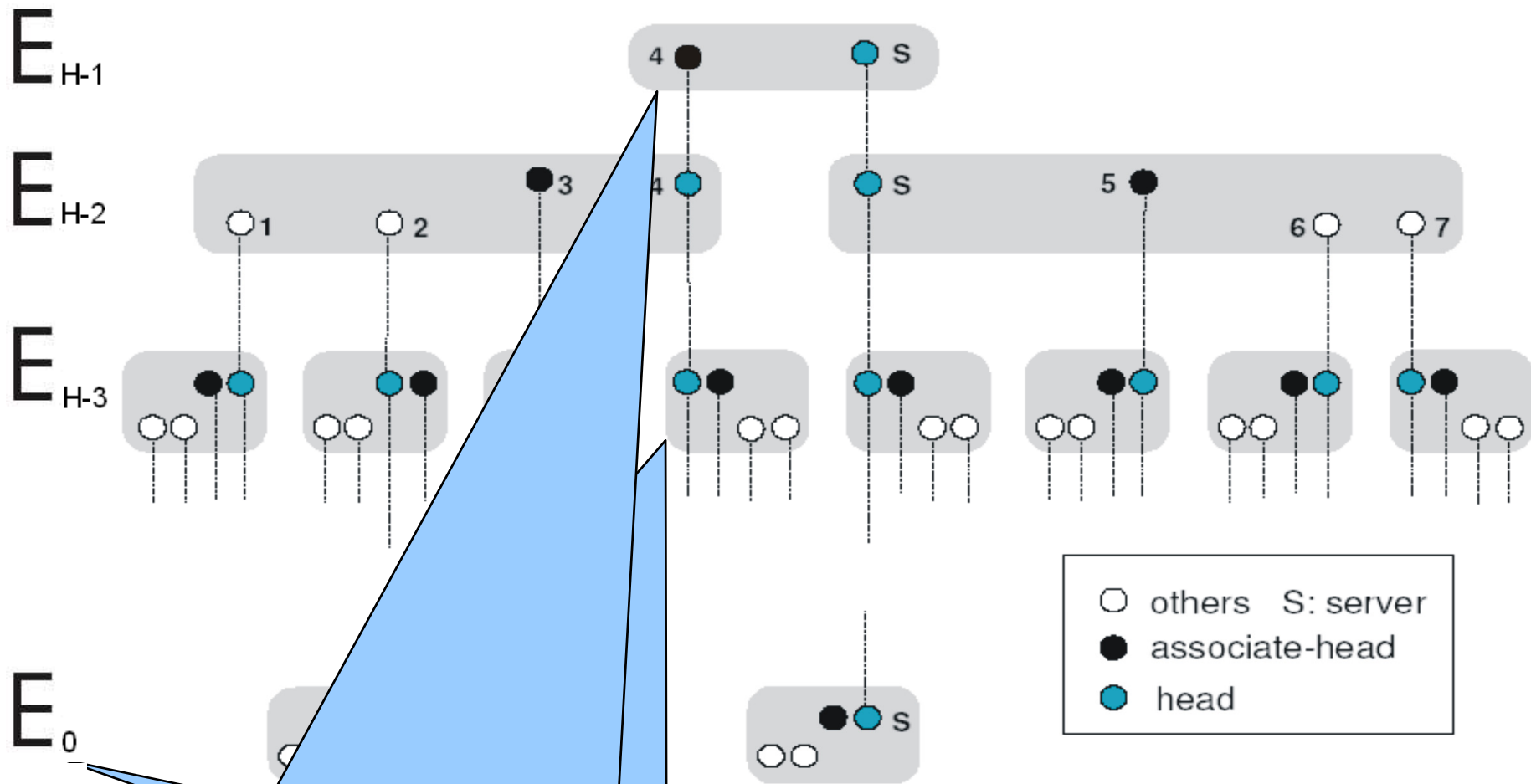
# ZIGZAG – P2P Based Application Layer Multicast

- ▶ Entwickelt von Tran, Hua, Do (University of Florida, Orlando, 2003)
- ▶ Weiterentwicklung von NICE (*NICE is Internet Cooperative Environment*)
- ▶ Topologie: hier



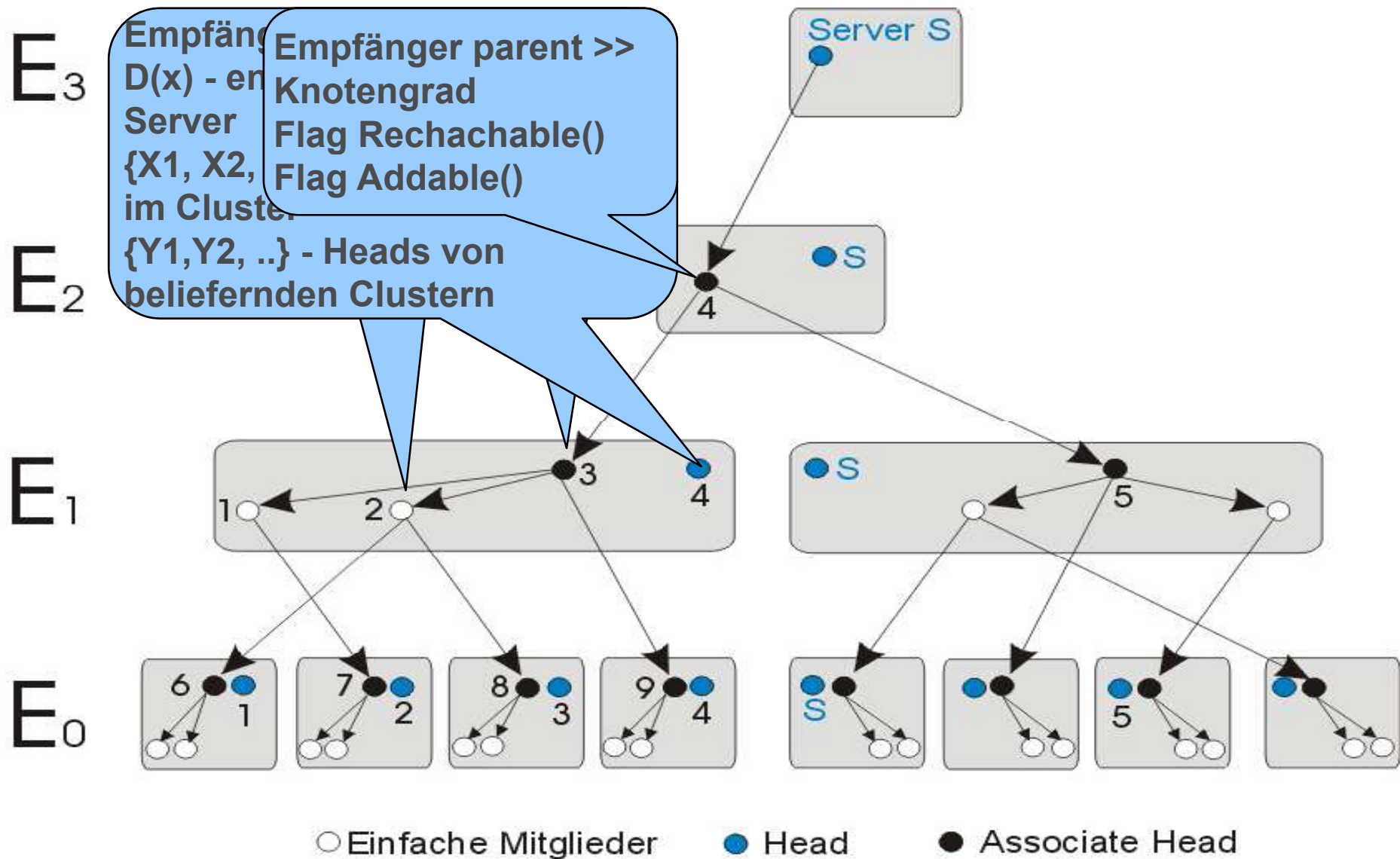
- ▶ Robustheit durch Trennung von Peermanagement und Tree
  - Head des Clusters zuständig für Administration des Netzwerks
  - Associate-Head des Clusters verantwortlich für Nutzdatenverteilung
- ▶ Delays und geringe Baumtiefe werden berücksichtigt
- ▶ Datenverbreitung durch ZIGZAG zwischen Clustern

# ZIGZAG Administrative Organisation von Peers

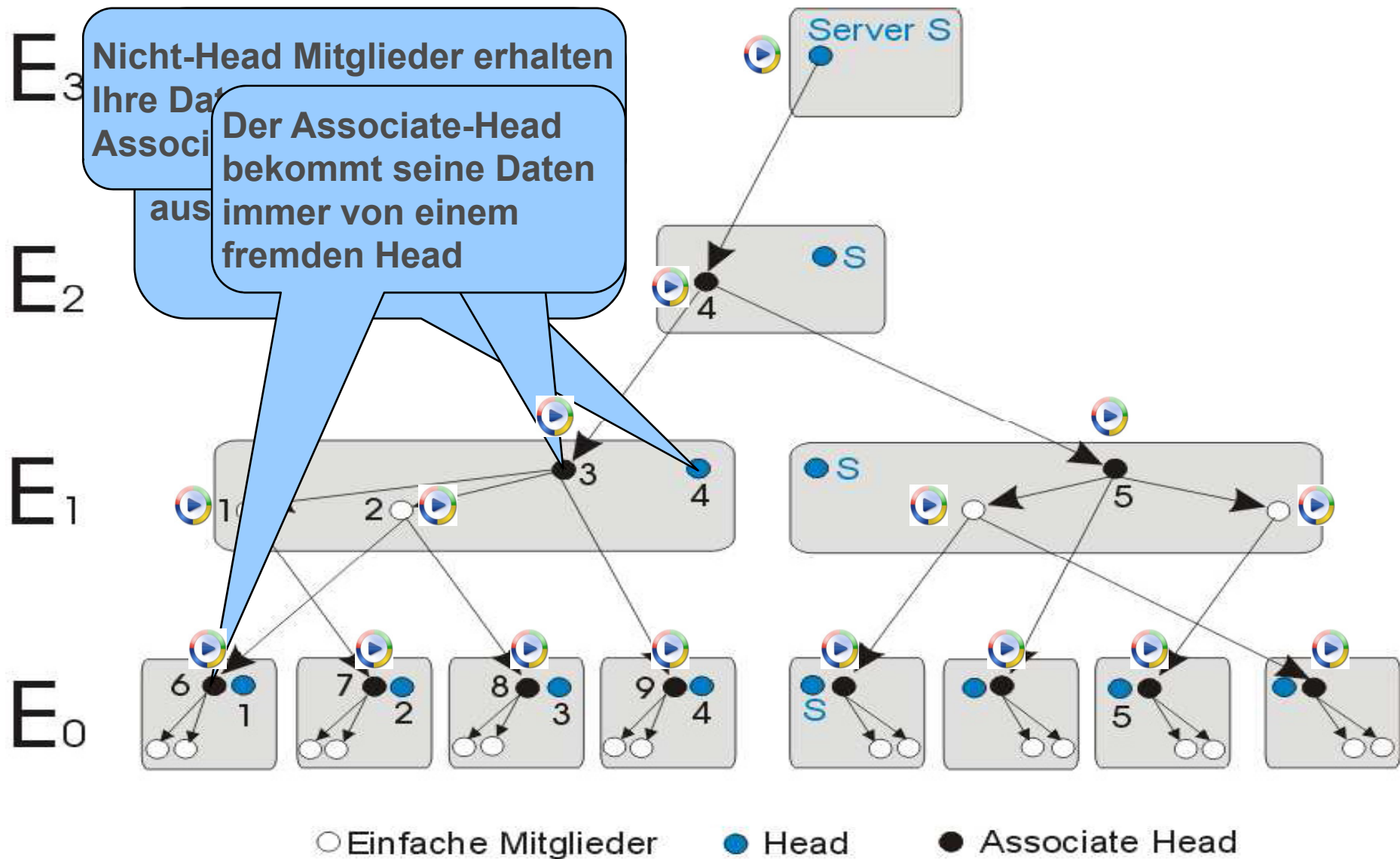


Ebene H-1 hat nur einen Cluster, dessen Größe  $[2, 3k]$ .

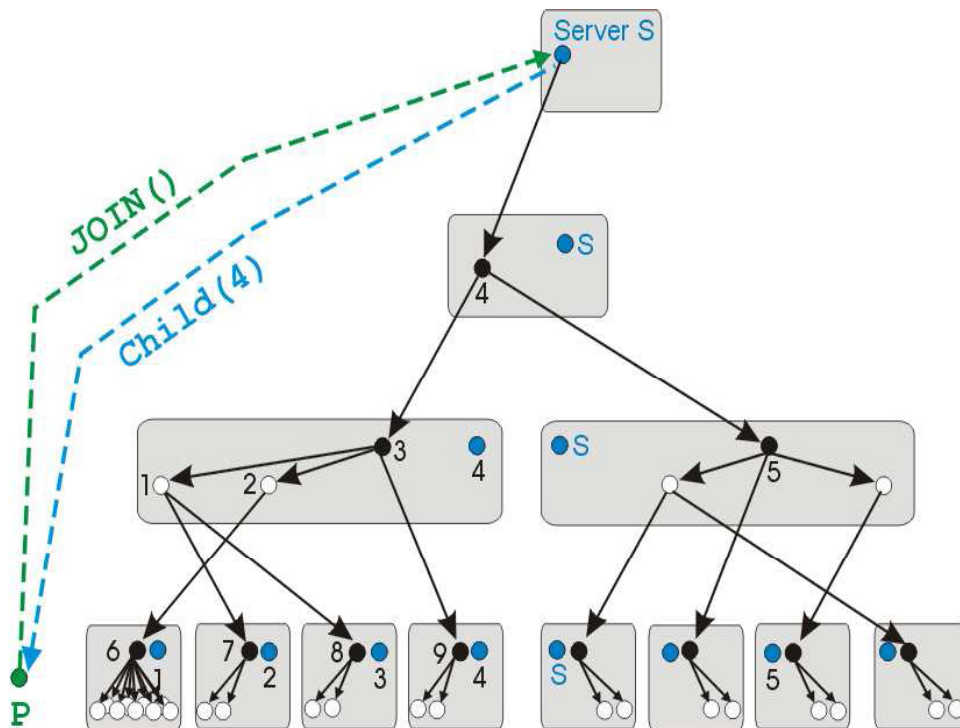
# ZIGZAG Peermanagement-Protokoll



# ZIGZAG Multicast Baum



# ZIGZAG - Rekursiver Join-Algorithmus



JOIN-Algorithmus:  
If  $X$  is layer-0 ass. head

Else  
If  $Addable(X)$   
Select a child  $Y$  :  
 $Addable(Y)$  and  
 $D(Y)+d(Y,P)$  is min  
Forward join request to  $Y$

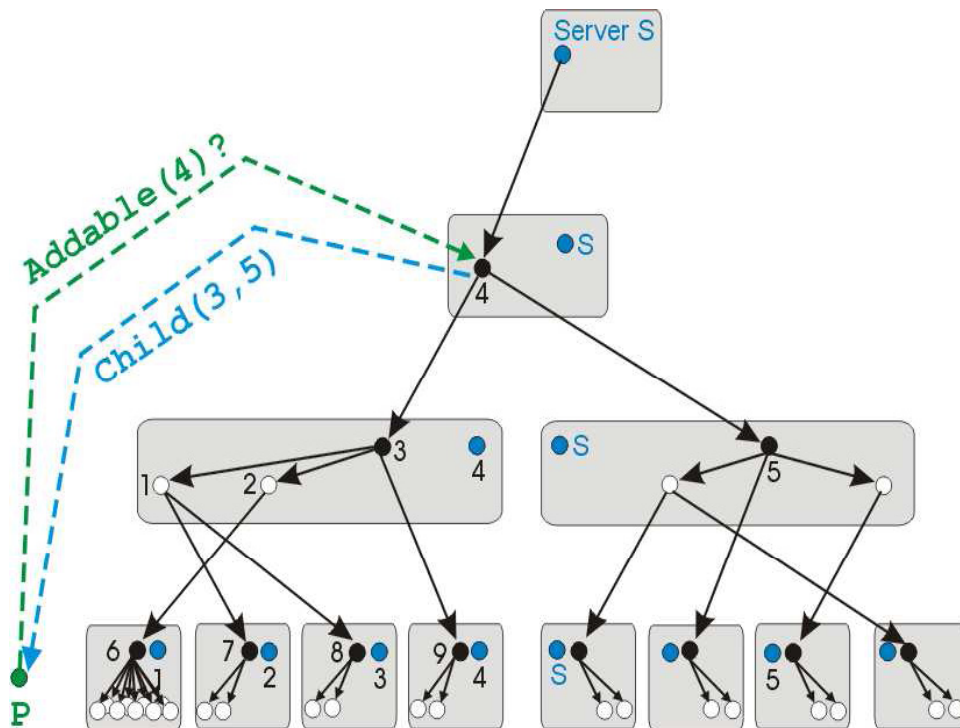
$Addable(X)$ -true, if  $Reachable(X) = true$  and  $X$  in cluster  $[k;3k-1]$

$Reachable(X)$ -true, if there exists a path in the multicast tree from  $X$  to a layer-0

$d(Y, P)$  – Verzögerung von  $Y$  nach  $P$

$D(Y)$  – Verzögerung von  $Y$  zum Server von  $Y$

# ZIGZAG - Rekursiver Join-Algorithmus



JOIN-Algorithmus:  
If  $X$  is layer-0 ass. head

Else  
If  $Addable(X)$   
Select a child  $Y$  :  
 $Addable(Y)$  and  
 $D(Y)+d(Y,P)$  is min  
Forward join request to  $Y$

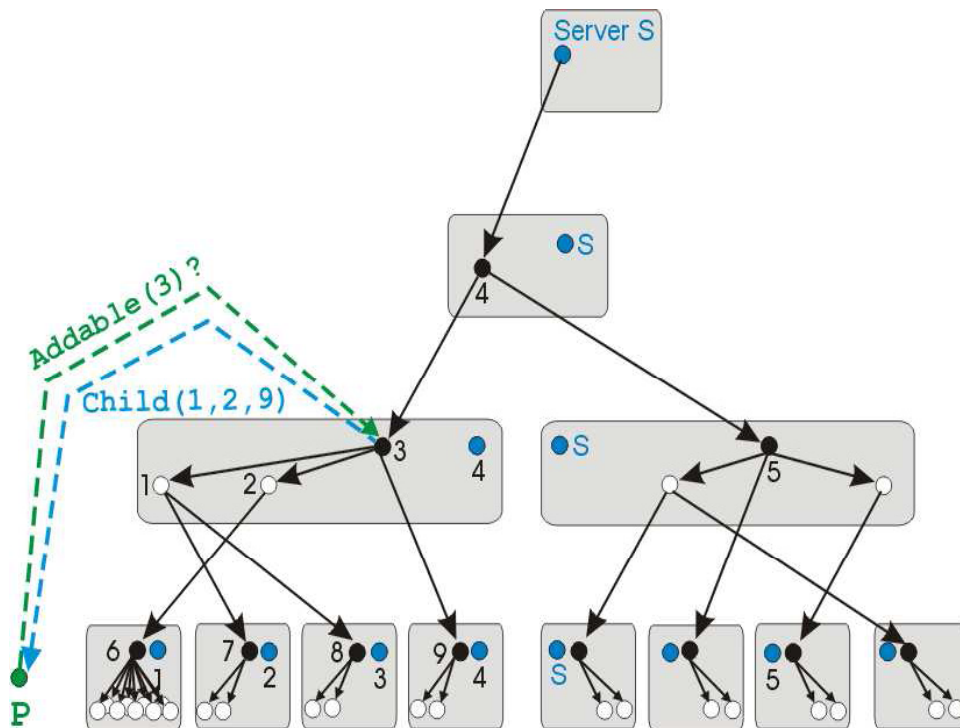
$Addable(X)$ -true, if  $Reachable(X) = true$  and  $X$  in cluster  $[k;3k-1]$

$Reachable(X)$ -true, if there exists a path in the multicast tree from  $X$  to a layer-0

$d(Y, P)$  – Verzögerung von  $Y$  nach  $P$

$D(Y)$  – Verzögerung von  $Y$  zum Server von  $Y$

# ZIGZAG - Rekursiver Join-Algorithmus



JOIN-Algorithmus:  
If  $X$  is layer-0 ass. head

Else  
If  $Addable(X)$   
Select a child  $Y$  :  
 $Addable(Y)$  and  
 $D(Y)+d(Y,P)$  is min  
Forward join request to  $Y$

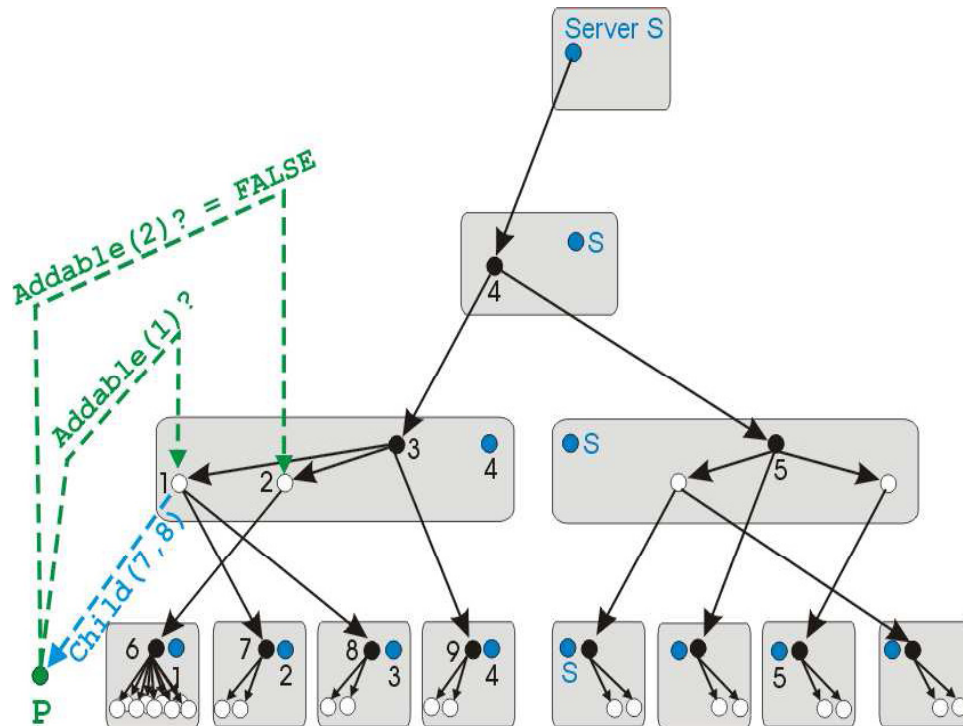
$Addable(X)$ -true, if  $Reachable(X) = true$  and  $X$  in cluster  $[k;3k-1]$

$Reachable(X)$ -true, if there exists a path in the multicast tree from  $X$  to a layer-0

$d(Y, P)$  – Verzögerung von  $Y$  nach  $P$

$D(Y)$  – Verzögerung von  $Y$  zum Server von  $Y$

# ZIGZAG - Rekursiver Join-Algorithmus



JOIN-Algorithmus:  
If  $X$  is layer-0 ass. head

Else  
If  $Addable(X)$   
Select a child  $Y$  :  
 $Addable(Y)$  and  
 $D(Y)+d(Y,P)$  is min  
Forward join request to  $Y$

$Addable(X)$ -true, if  $Reachable(X) = true$  and  $X$  in cluster  $[k;3k-1]$

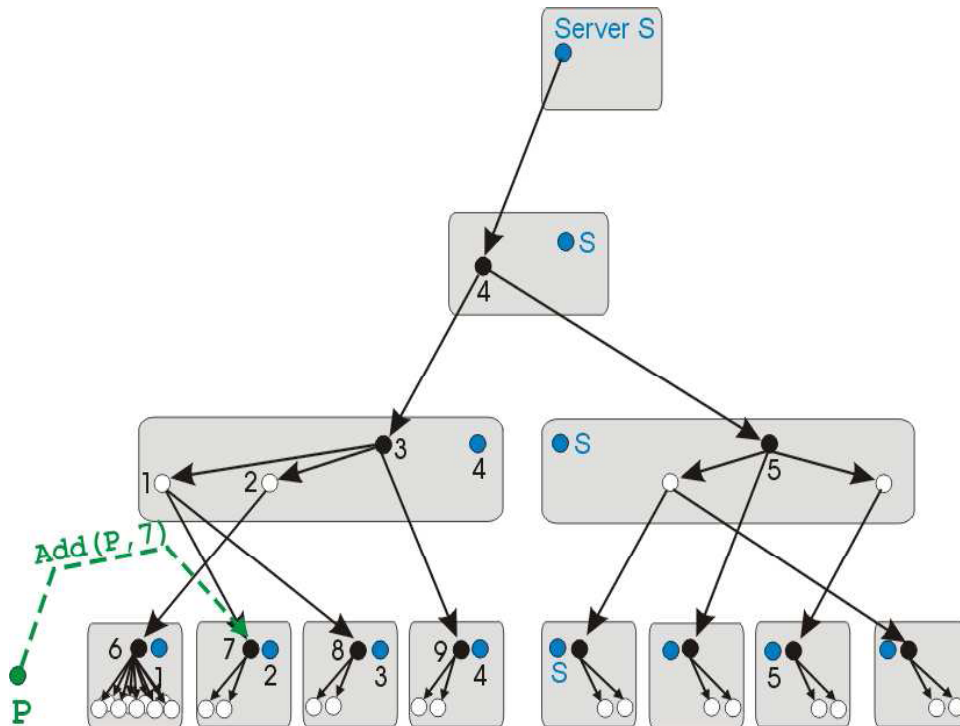
$Reachable(X)$ -true, if there exists a path in the multicast tree from  $X$  to a layer-0

$d(Y, P)$  – Verzögerung von  $Y$  nach  $P$

$D(Y)$  – Verzögerung von  $Y$  zum Server von  $Y$



# ZIGZAG - Rekursiver Join-Algorithmus



JOIN-Algorithmus:

If  $X$  is layer-0 ass. Head

Add  $P$  to only cluster of  $X$

Make  $P$  new child of  $X$

Else

If  $Addable(X)$

Select a child  $Y$  :

$Addable(Y)$  and

$D(Y)+d(Y,P)$  is min

Forward join request to  $Y$

Else

Select a child  $Y$  :

$Reachable(Y)$  and

$D(Y)+d(Y,P)$  is min

Forward join request to  $Y$

$Addable(X)$ -true, if  $Reachable(X) = true$  and  $X$  in cluster  $[k;3k-1]$

$Reachable(X)$ -true, if there exists a path in the multicast tree from  $X$  to a layer-0

$d(Y, P)$  – Verzögerung von  $Y$  nach  $P$

$D(Y)$  – Verzögerung von  $Y$  zum Server von  $Y$

# Zusammenfassung

---

## ▶ P2P basiertes IPTV

- Flexibel und umgeht heutige Internetbeschränkungen (wie IP-Multicast)
- Unabhängig von der darunter liegenden Topologie
- Keine Veränderungen an Router notwendig
- Resistenter gegen Angriffe/ Ausfälle
- Nutzt vorhandene Ressourcen und Infrastruktur aus

## ▶ Nachteil:

- Benötigt komplexe Koordination der Peers und Overhead

## ▶ Herausforderungen bei Datenverteilung:

- Verteilung der Last, Berücksichtigung von physikalischem Netz, geeignete Metriken, Quality of Experience und QoS



# Praxis

*[www3.informatik.uni-wuerzburg.de](http://www3.informatik.uni-wuerzburg.de)*



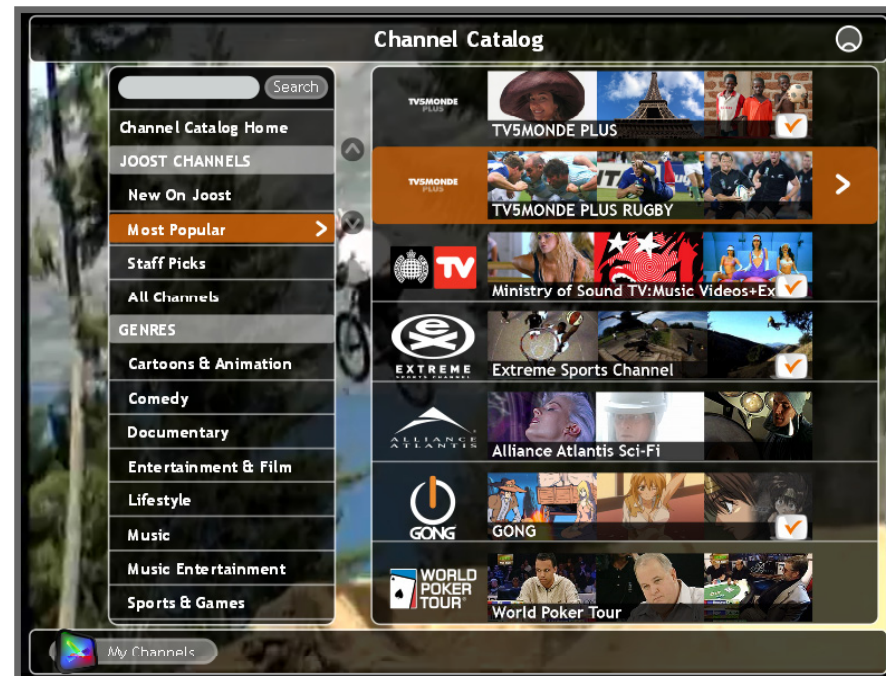
- ▶ Joost ist ein Video on Demand Dienst der im Oktober 2006 von den Skype und Kazaa Gründer Niklas Zennström und Janus Friis unter dem Namen Venice Project gegründet wurde.
- ▶ 16. Januar 2007: Offizieller Name des Projektes nun endgültig Joost
- ▶ 27. August 2007: Beta-Version 0.12.0 verfügbar - Start der offenen Beta

▶ Systemvoraussetzungen:

- Windows XP, Vista
- Mac OS X
- Linux
- Mind. 500 MHz CPU
- Mind. 48 MB Grafikkarte



- ▶ Basiert ähnlich wie KaZaA und Skype auf dem FastTrack-P2P-Prinzip
- ▶ Besteht hauptsächlich aus Open-Source-Komponenten (Mozilla, Apache HTTPD, Redland, Postgres, OpenSSL, RDF, SVG, XUL)
- ▶ Verwendet den H.264 Codec
- ▶ Im Gegensatz zu YouTube wird auf lizenzierte Qualität gesetzt
- ▶ In Deutschland über 50 Kanäle zur Auswahl
- ▶ Finanzierung läuft über personalisierte Werbeclips
- ▶ Programm kann gestoppt, gespult und noch einmal angeschaut werden





**Zattoo is TV to go! What made us couch potatoes was that we had to watch it there.**

- ▶ Zattoo wurde 2005 in den USA von Sugih Jamin, Professor der Informatik an der University of Michigan und Beat Knecht, Marketingfachmann für Softwareprodukte und früherer McKinsey Berater in Silicon Valley, gegründet.
- ▶ Zattoo, jap. für „Menschenmenge“
- ▶ Bereits über 850.000 Nutzer
- ▶ Verfügbar in der Schweiz, Deutschland (seit 14.09.2007), Dänemark, Spanien
- ▶ Nächsten Länder: Belgien, Österreich, Frankreich
- ▶ Systemvoraussetzungen:
  - Windows XP, Vista, Mac OS X, Linux
  - Mind. 1.5 GHz, 512 MB RAM



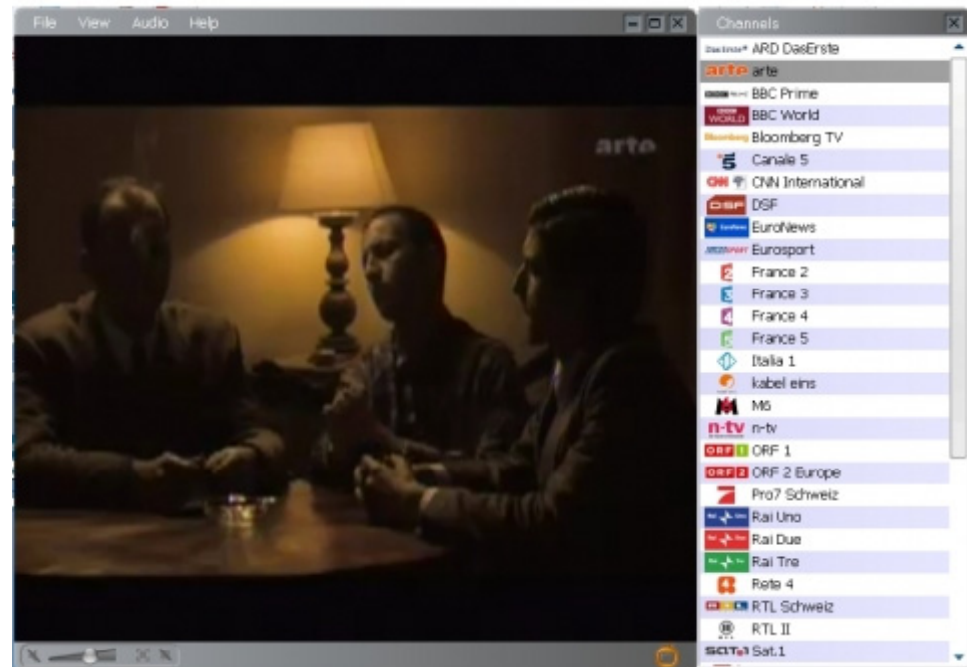
- ▶ Aufnahmen und Pausen sind aus rechtlichen Gründen nicht möglich
- ▶ Kanalwechsel in weniger als 10 Sekunden, davon 2 Sekunden Zwischenspeicherung
- ▶ Gesamtverzögerung ca. 20 Sekunden
- ▶ Auflösung von 350x288 Pixeln





- ▶ Suchfunktionen und Chat-Möglichkeiten sollen ähnlich wie bei Joost eingebaut werden
- ▶ In Zukunft neben den kostenlosen Sender Pay-TV
- ▶ Finanzierung von Zattoo soll über Werbeeinblendungen beim Kanalwechsel erfolgen.
- ▶ Übernahme von Videotext ist nicht geplant
- ▶ Mehr als 50 Kanäle geplant


**Nicht als Ersatz für den normalen Fernseher, sondern für mobile Nutzer gedacht**



# IPTV

---

Ausserdem gibt es noch:

- ▶ Babelgum 
- ▶ PPLive
- ▶ TVUPlayer
- ▶ ChooseAndWatch & FreeTube

Das Internet ist voll mit verschiedenen IPTV-Software-Angeboten:

Feidian, PPMate, ab365 Live, QQ Live, MaxTV Online, VGO Live, Streamer One, SopCast, PPStream, PeerCast, MySee, TVAnts, TV Koo!, Coolstreaming Mediacenter, usw.

# Was ist IPTV?

---

**Vielen Dank für Ihre Aufmerksamkeit!**

Fragen?