Are Information-Centric Networks Video-Ready?

Christos Tsilopoulos, George Xylomenos and George C. Polyzos

Mobile Multimedia Laboratory, Athens University of Economics and Business

DEPARTMENT OF INFORMATICS

OPERATIONAL PROGRAMME EDUCATION AND LIFELONG LEARNING
investing in knowledge society
MINISTRY OF EDUCATION, LIFELONG LEARNING AND RELIGIOUS AFFAIRS
MANAGING AUTHORITY
European Union
European Social Fund
Co-financed by Greece and the European Union
NSRF 2007-2013
EUROPEAN SOCIAL FUND

Presentation

- Discussion
  - No specific solution/system presented
  - Highlight good and not so good features of ICN w.r.t. video transport
  - Point issues that need attention
- Why is this discussion important?
  - Video applications attracted ICN researchers
  - Prototype implementations focus on message passing using ICN primitives
  - Critical aspects w.r.t. performance and scalability left for future work

Our experience with ICN

- Participated in

2008 - 2010 | 2010 - 2012 | 2011 - 2013

- Publish-Subscribe Internetworking (PSI)
- Implemented video applications in prototypes [1]
  - Appealing demos
  - Promising application
- Message passing but not deep study of application behavior
  - Many core pieces of the network architecture still missing

[1] Parisis et al., "Demonstrating Usage Diversity Over an Information-Centric Network," demo in IEEE INFOCOM 2013.

Can we finalize some aspects on ICN and move on?

- Many ICN proposals
  - Content-Centric Networking
  - NetInf
  - Publish-Subscribe Internetworking
  - …
- With similarities
  - Goal: Primary focus to content distribution
  - Self-identified information items
  - Universal caching, anycast, multicast
- And differences
  - Diverse approaches in core functions
    - Item lookup, routing, forwarding
  - CCN: pull-based, distributed control plane, hop by hop routing/forwarding
  - PSI: push-based, centralized control plane, explicit-routing

Internet Video Transfer

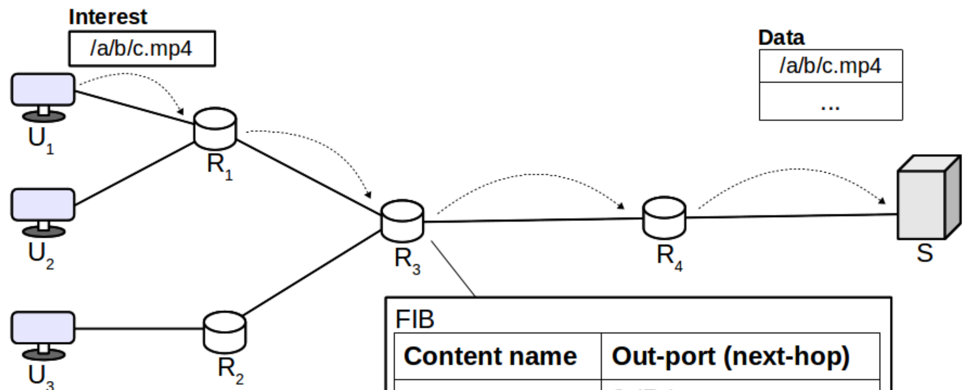
- Internet video applications operate on top of <u>well defined architecture</u>
  - End-to-end system design
  - Network layer: best effort, IP host addresses
  - Transport layer: UDP, TCP
  - Application layer: RTP, HTTP
- Applications choose protocols based on application context and protocol behavior
  - Video on Demand vs Live Streaming
  - Stream adaptation
- Can we port existing video applications to ICN as is?
  - ICN API looks similar to application layer protocols…
- <u>Not that simple</u>

Rest of presentation

- Two diverse ICN architectures
  - Content-Centric Networking
  - Publish-Subscibe Internet
- Two kinds of video applications with different transport requirements
  - Video on Demand: reliable transfer
  - Live Streaming: real-time delivery
- Which features of ICN facilitate video transfer
- What seems problematic

Content-Centric Networking (CCN)

- Named content packets
  - Hierarchical names
  - Interest – Data packets
  - No host addresses
- Pull-based operation
  - One Interest per Data
- Packet caches in routers
- Native multicast and anycast
  - Strategy layer in routers
- Receiver-driven transport
  - Error control performed by receiver
  - Congestion control under research



FIB

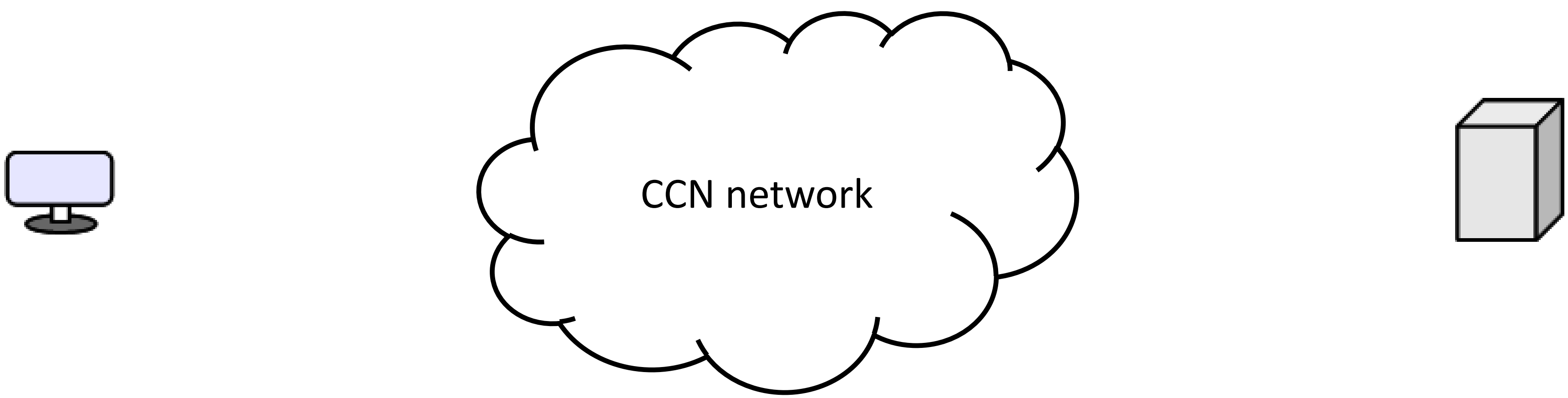
Content name	Out-port (next-hop)
/a/b/	3 (R_4)

PIT

Content name	Requested from port
/a/b/c.mp4	1 (R_1)

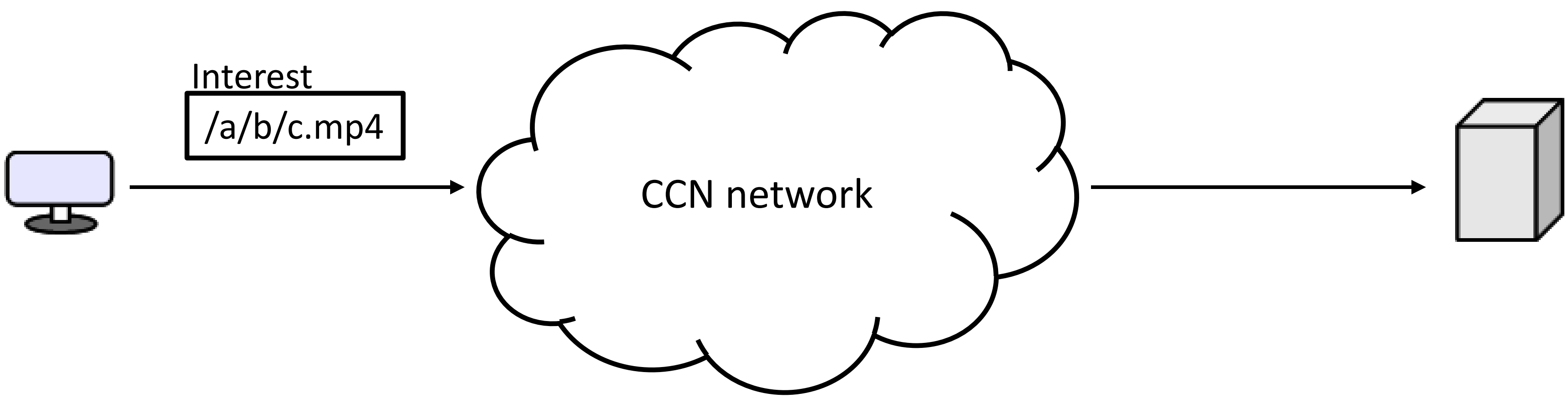
Video on Demand over CCN

- Request each video packet
  - Similar to HTTP streaming
  - Difference: request network packets, not chunks
- Receiver-driven stream adaptation *looks* straightforward

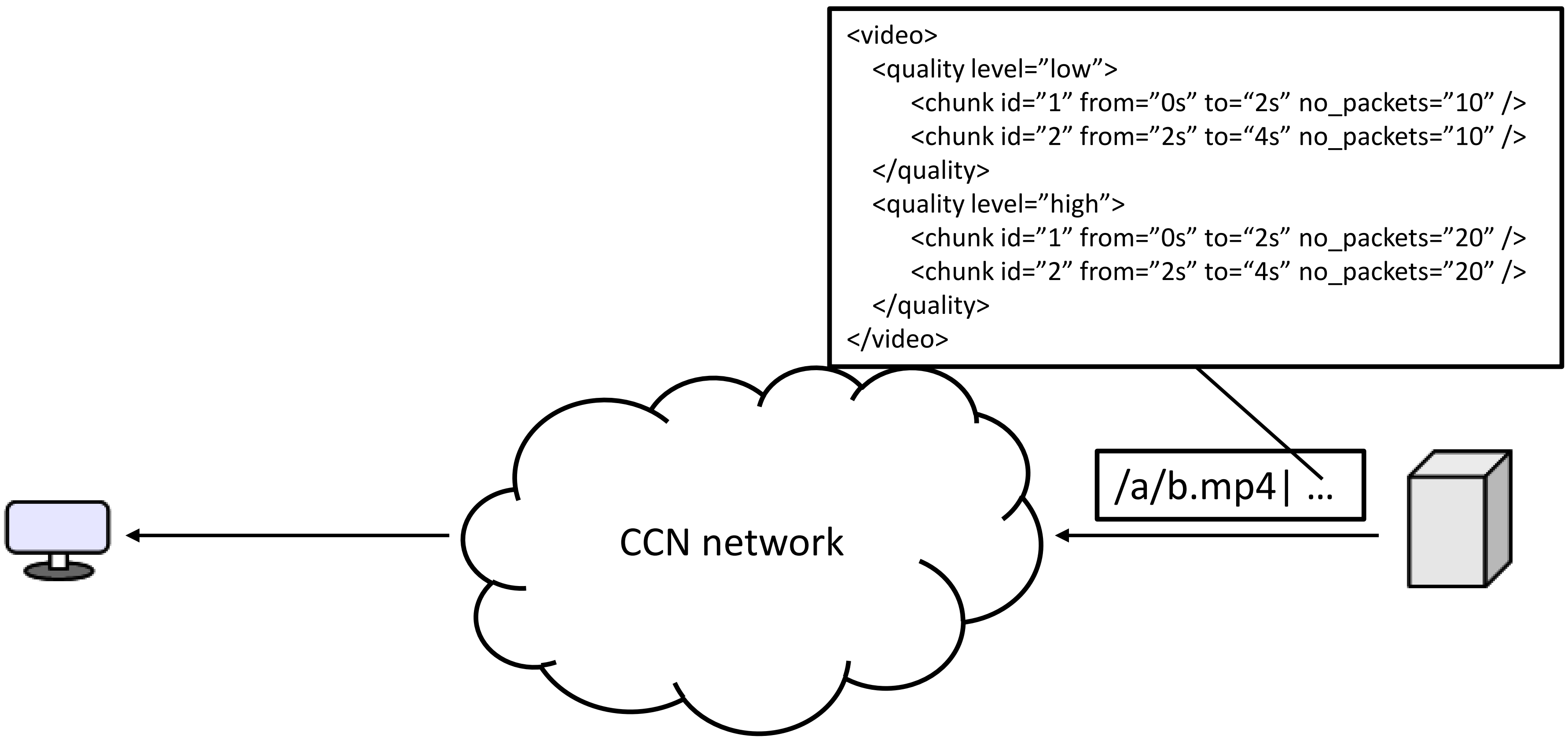


Video on Demand over CCN

- Request video */a/b/c.mp4*



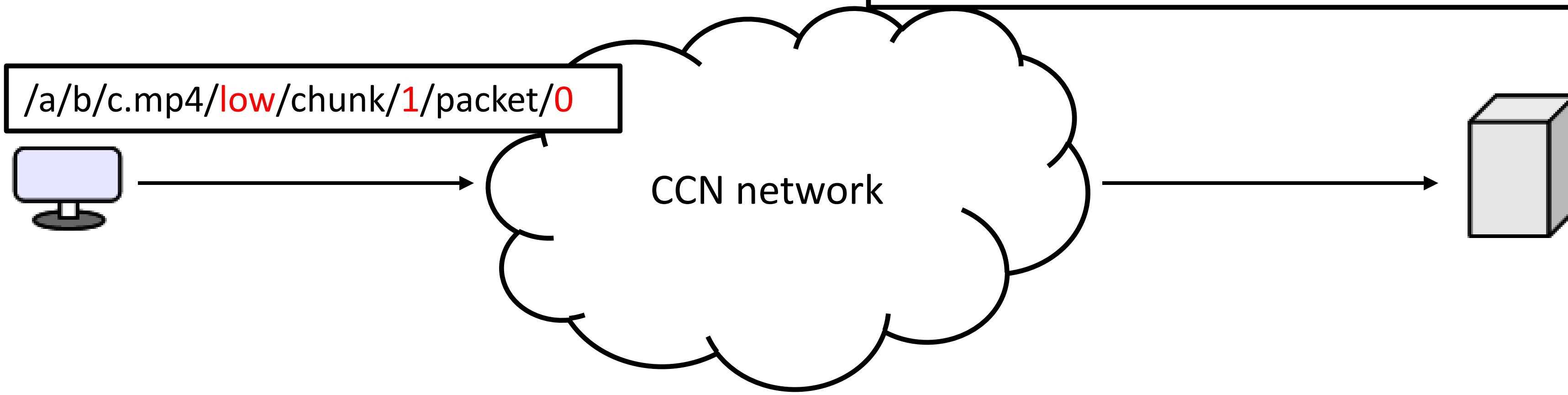
Video on Demand over CCN



Video on Demand over CCN

- Interests forwarded to content source
  - Longest-prefix match

```
<video>
  <quality level=”low”>
    <chunk id=”1” from=”0s” to=“2s” no_packets=”10” />
    <chunk id=”2” from=”2s” to=“4s” no_packets=”10” />
  </quality>
  <quality level=”high”>
    <chunk id=”1” from=”0s” to=“2s” no_packets=”20” />
    <chunk id=”2” from=”2s” to=“4s” no_packets=”20” />
  </quality>
</video>
```



Receiver-driven Stream Adaptation

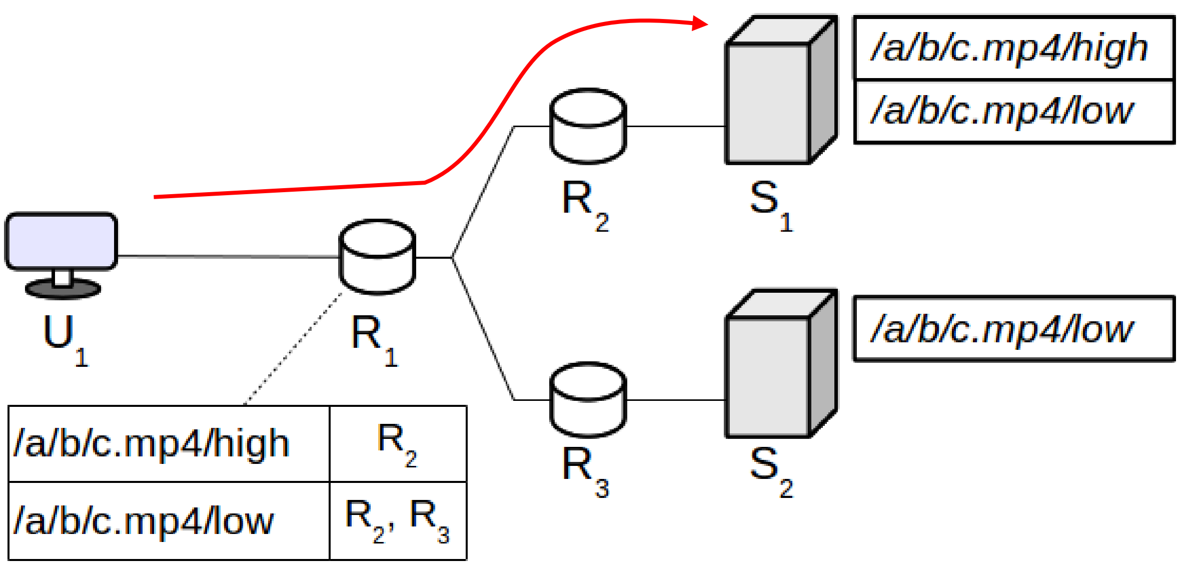
- Current rationale: adapt stream quality based on *end-to-end bandwidth estimation*
  - Packets arrive quickly? Increase quality
  - Packets arrive late? Decrease quality

Receiver-driven Stream Adaptation

- Current rationale: adapt stream quality based on *end-to-end bandwidth estimation*
  - Packets arrive quickly? Increase quality
  - Packets arrive late? Decrease quality
- Hard to estimate *end-to-end bandwidth* in CCN
  - Content source is unknown to receiver

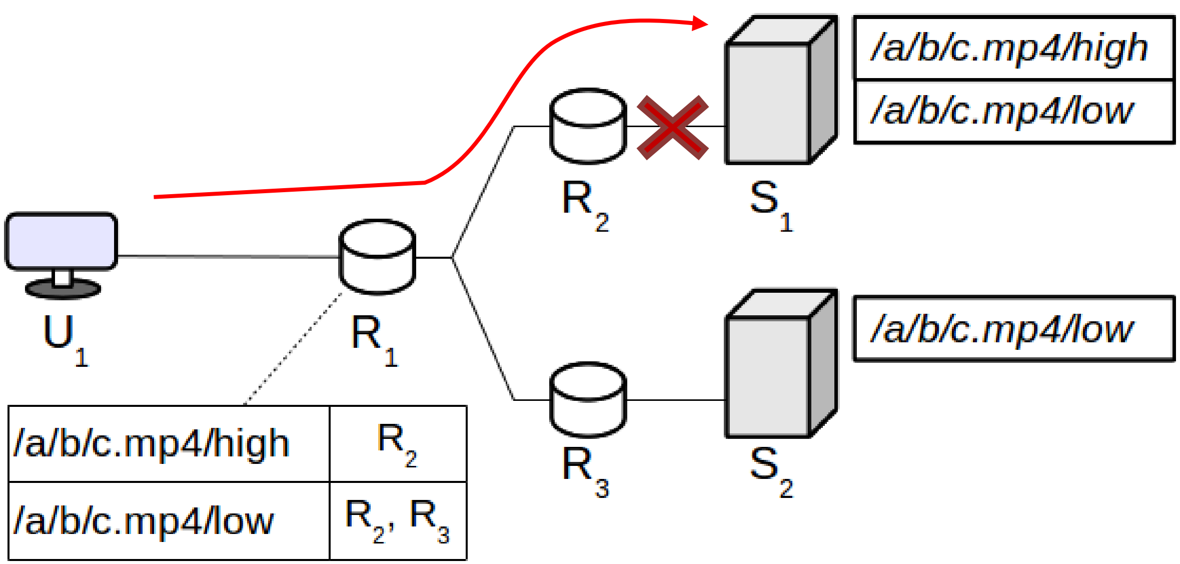
Receiver-driven Stream Adaptation

1. U_1 starts with high quality



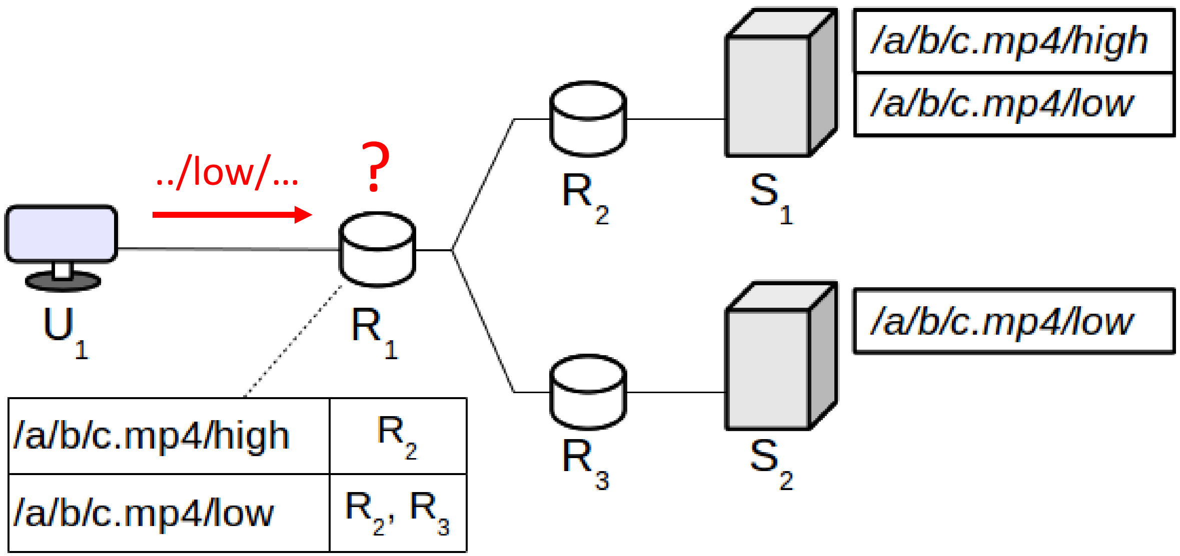
Receiver-driven Stream Adaptation

1. U_1 starts with high quality
2. Congestion in $R_2 – S_1$
   – U_1 switches to low quality



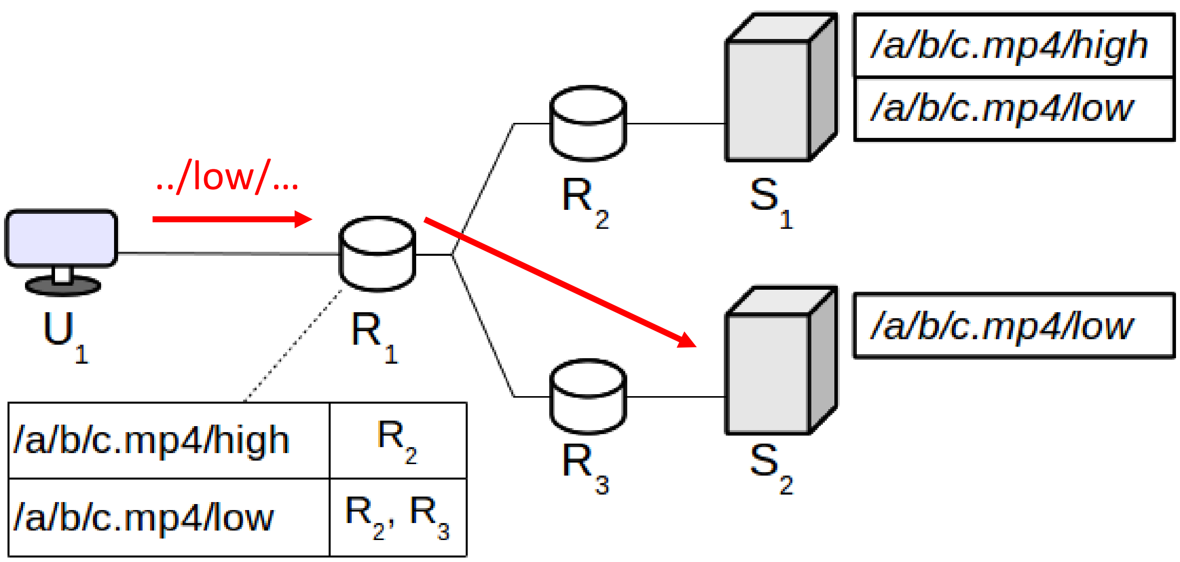
Receiver-driven Stream Adaptation

- What does R_1 do?



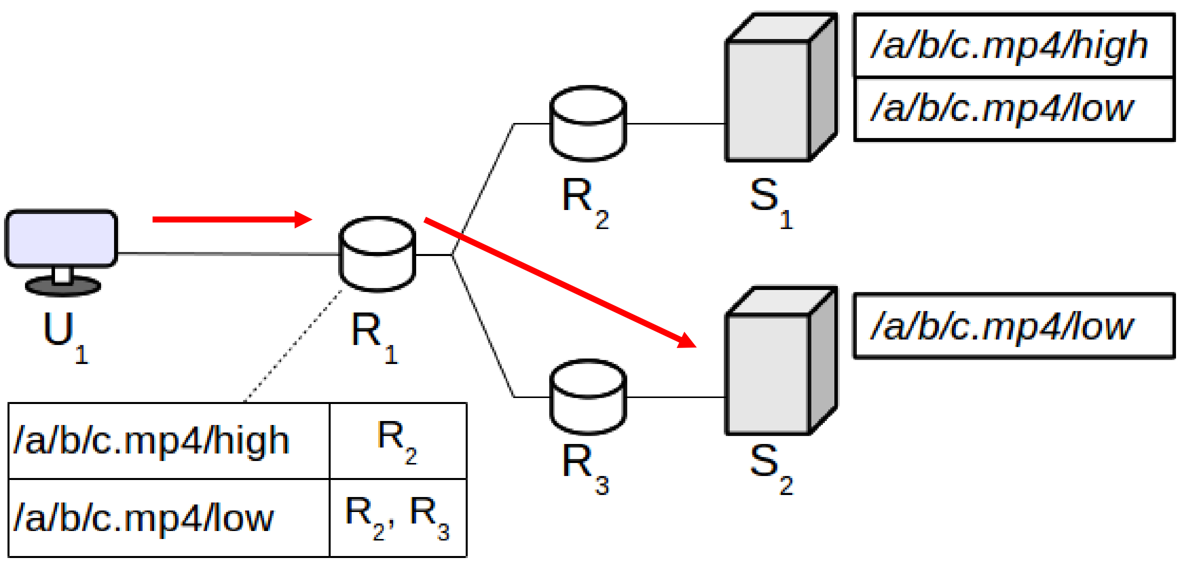
Receiver-driven Stream Adaptation

- What does R_1 do? Forward Interest to R_3



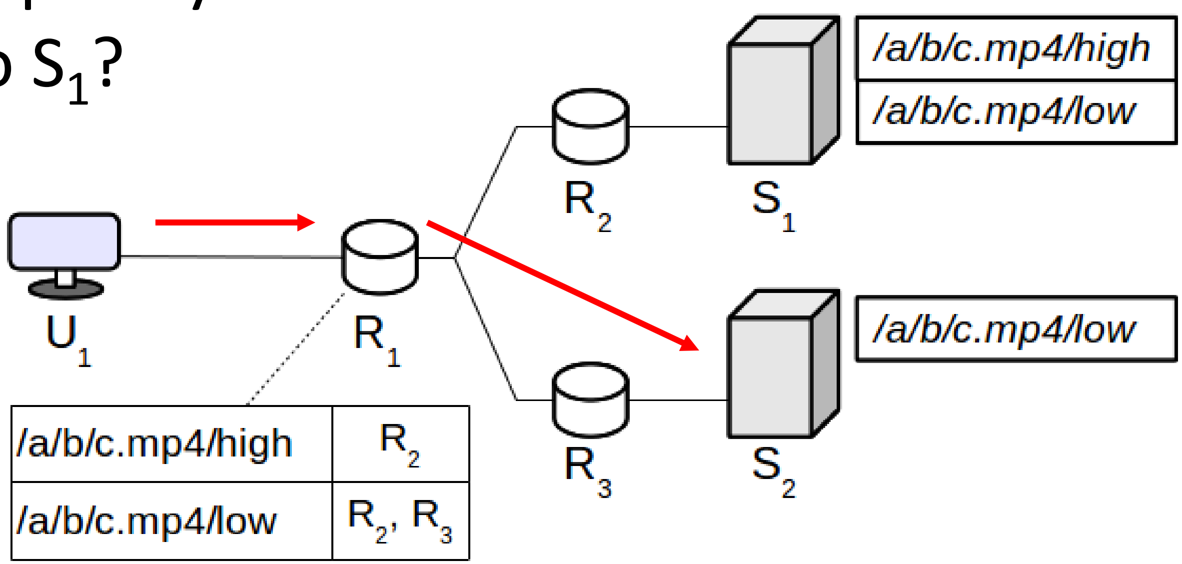
Receiver-driven Stream Adaptation

- What does R_1 do? Forward Interest to R_3
- What if R_1-S_2 even worse than R_1-S_1?



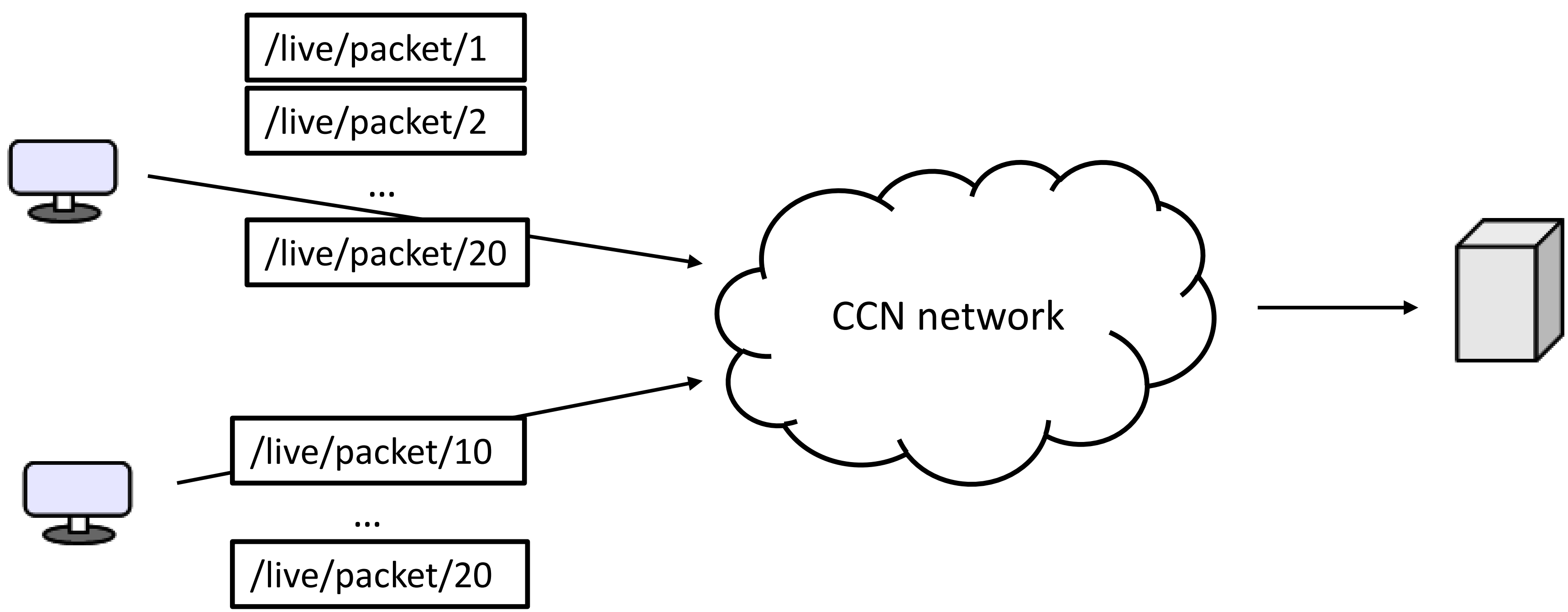
Receiver-driven Stream Adaptation

- What does R_1 do? Forward Interest to R_3
- What if R_1-S_2 even worse than R_1-S_1?
  - Client switches back to *high*?
  - Explicitly send Interest to S_1?



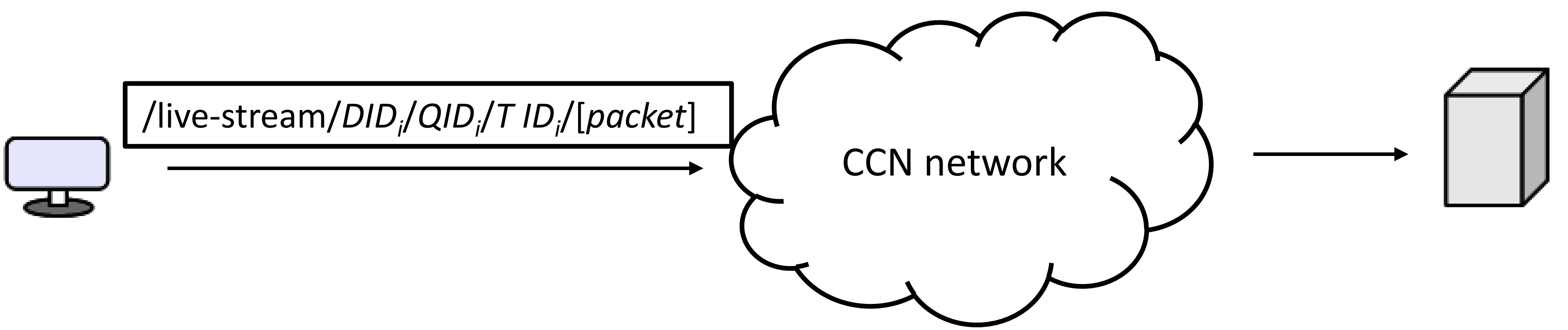
Live Streaming in CCN

- Real-time delivery
  - Proactively transmit Interests for upcoming packets
- Native multicast support



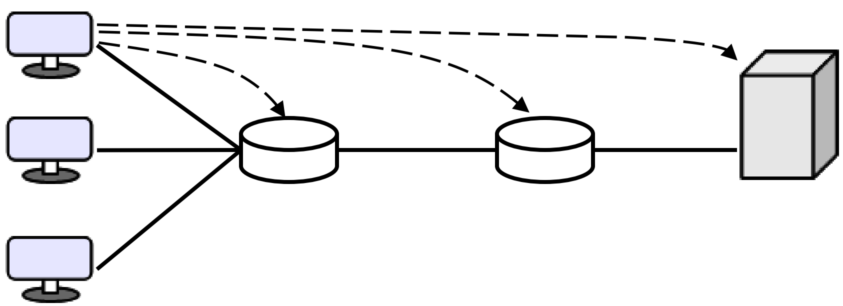
Live Streaming in CCN

- Receiver-driven layered multicast
- Case study: H264 Scalable Video Coding
  - Dependency ID (DID)
  - Quality ID (QID)
  - Temporal ID (TID)
  - Interest: */live-stream/DID_i/QID_i/$T\ ID_i$/[packet]*
- Simple network operation
  - No specific Media Aware Network Elements
  - No multicast JOIN-LEAVE messages



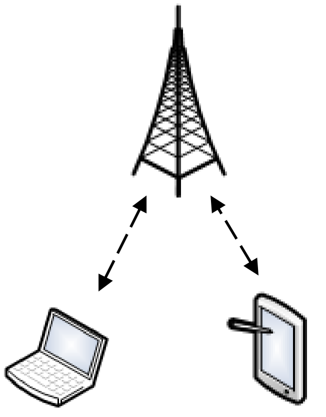
Live Streaming in CCN

- Packet caches in routers
- Assist in error recovery
  - Cache replacement policy according to packet content
  - I frames > P frames > B frames
  - Video packetization
- Complicates end-to-end bandwidth estimation [3]



[3] Grandl, Su and Westphal, "On the Interaction of Adaptive Video Steaming with Content-Centric Networking,"
in Packet Video Workshop 2013.

Live Streaming in CCN



- Overhead caused by Interests
  - One Interest per Data
- Asymmetric /congested uplinks?
- Interest Aggregation [4]
  - Single Interest requests multiple Data packets
  - Additional complexity in routers
  - What if lost?
- Persistent Interests [5]
  - One Interest for all streaming Data packets
  - Similar to IP multicast (channel mode)
  - Longer lifetime than plain Interest
  - PIT size?

[4] Byun, Lee and Jang, "Adaptive Flow Control via Interest Aggregation in CCN,", in IEEE ICC 2013.

[5] Tsilopoulos and Xylomenos, "Supporting Diverse Traffic Types in Information Centric networks," in ACM SIGCOMM ICN workshop 2011.

CCN Summary

<table>
<tr><th></th><th>Improved</th><th>Unclear</th><th>Problematic</th></tr>
<tr><td>Video on Demand</td><td>Native anycast support.

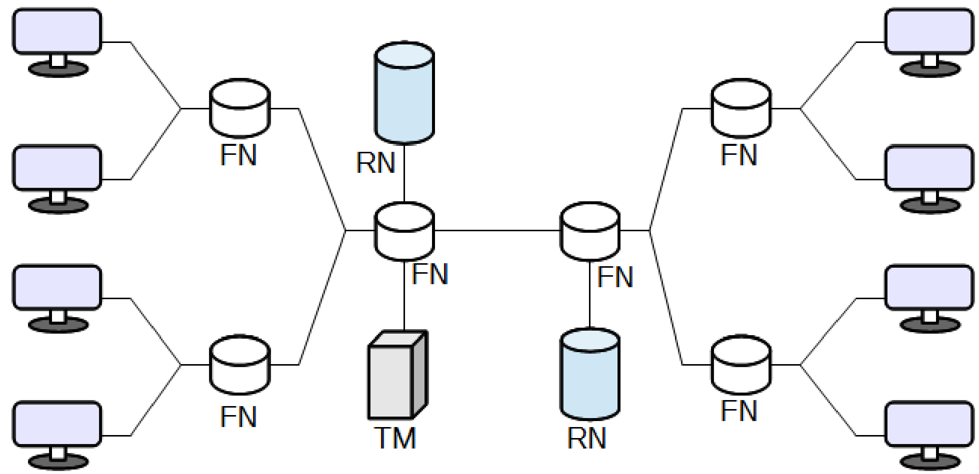
Enhanced retransmission-based error control with in-network packet-level caching.</td><td>End-to-end throughput estimation for stream adaptation.</td><td rowspan="2">Network overhead for explicitly requesting individual Data packets</td></tr>
<tr><td>Live Streaming</td><td>Enhanced retransmission-based error control with in-network packet-level caching.

Packet distinction in caching policies.</td><td>Service degradation in asymmetric links.

Lost Interests upstream result in missing Data on the downstream.</td></tr>
</table>

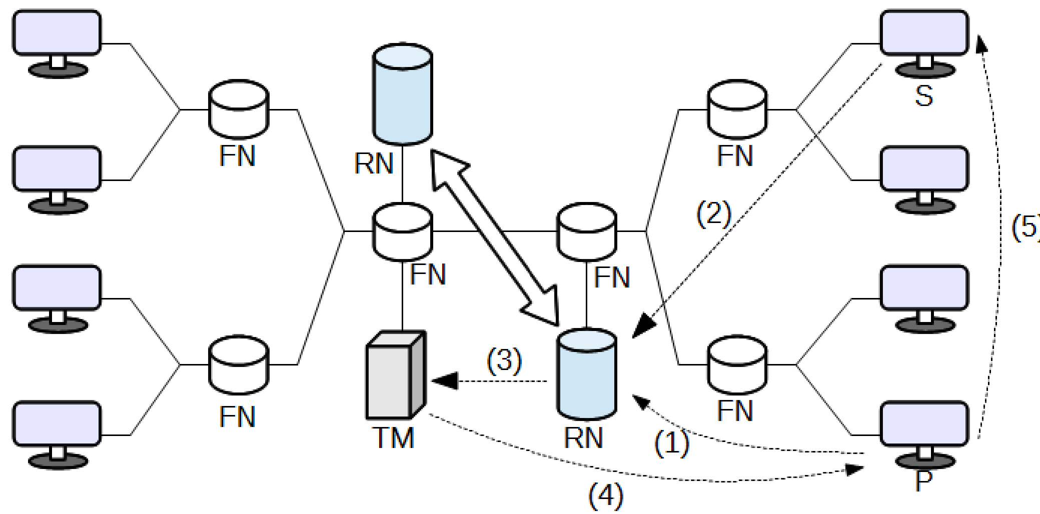
Publish-Subscribe Internetworking (PSI)

- 3 distinct network functions
  - Rendezvous
  - Topology Management & Path Formation
  - Forwarding



- Decouple routing from forwarding
  - Centralized route selection
  - Explicit-routing, Bloom filter-based
- Pub/sub API
- Abstract notion of content item
  - Not strictly a network packet
  - Could be a larger data unit: chunk or entire file, media stream
- Push-based

Publish-Subscribe Internetworking (PSI)

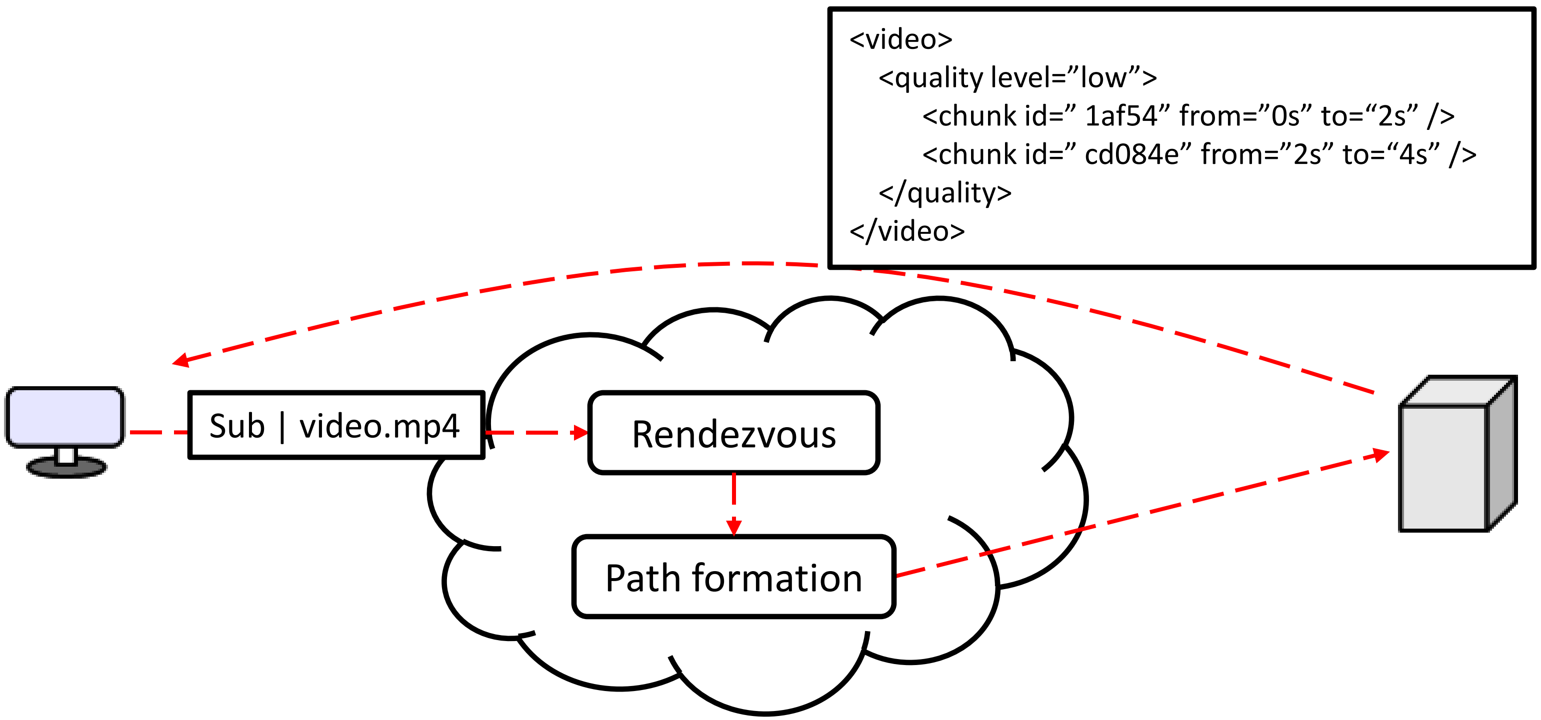


Operation

1. Producer publishes item (announcement)
2. Consumer subscribes to item
3. Network locates item
4. Computes publisher → subscriber path
   - Source route
   - Hands it to publisher
5. Publisher transmits data over specified path
   - Sender-driven or receiver-driver transport

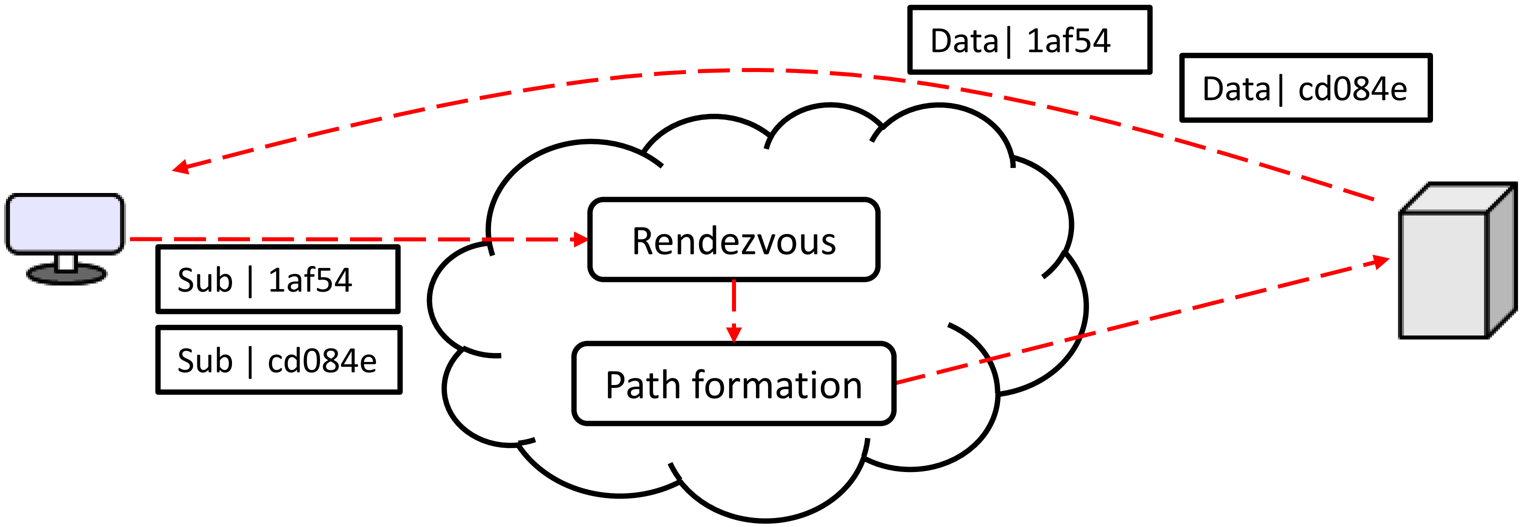
Video on Demand over PSI

1. Subscribe to video
   – Obtain metadata



Video on Demand over PSI

1. Subscribe to video
   - Obtain metadata
2. Subscribe to each piece

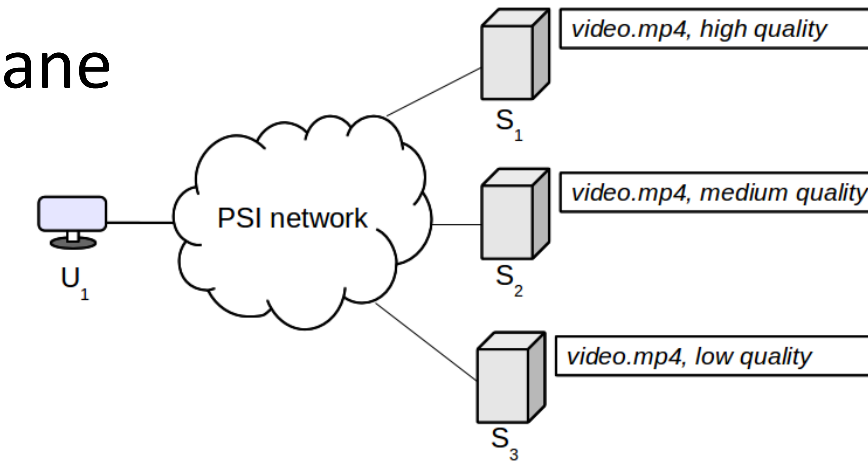


Video on Demand over PSI

- Granularity of video pieces
- Small pieces
  - \+ Receiver-driven stream adaptation
  - – Scalability: number of announcements to Rendezvous
  - – Amount of subscriptions: delay for resolution-path formation
- Large pieces
  - – Coarse-grained stream adaptation
  - \+ Less announcements
  - \+ Fewer subscriptions

Video on Demand over PSI

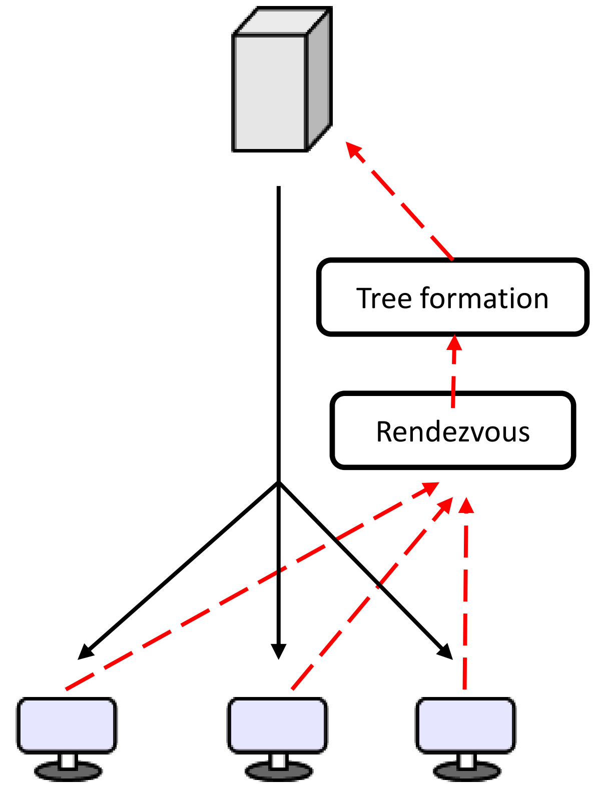
- What we have not seen *yet*:
- Utilize centralized control plane
- Network selects video source and quality on behalf of users
  - QoS parameters



- Need to enrich pub/sub primitives
  - Network must understand data
- Tradeoff general purpose with app specific semantics

Live Streaming in PSI

- Name the stream, not each packet
  - Channel mode, similar to IP multicast
  - \+ One subscription only
  - – No packet caches in routers
- Centralized multicast tree computation
  - \+ Optimization benefits, e.g. Steiner trees [6, 7]
  - – Increased delays



[6] Li et al., "ESM: Efficient and scalable data center multicast routing," IEEE/ACM Transactions on Networking 2012.
[7] Tsilopoulos et al., "Efficient real-time information delivery in future internet publish-subscribe networks," ICNC 2013.

PSI Summary

	Improved	Unclear	Problematic
Video on Demand	Native anycast support. Optimal path selection through centralized route control.	End-to-end throughput estimation. Optimal path selection requires extensions to pub/sub primitives.	Delays for resolution of subscriptions and unsubscriptions.
Live Streaming	Optimal multicast delivery through centralized route control.	Scalability of centralized multicast tree construction (with dynamic user behavior).	

Thank you

Questions?