

www.openboxware.net

OPENBOXWARE®

Gestione avanzata
di flussi multimediali

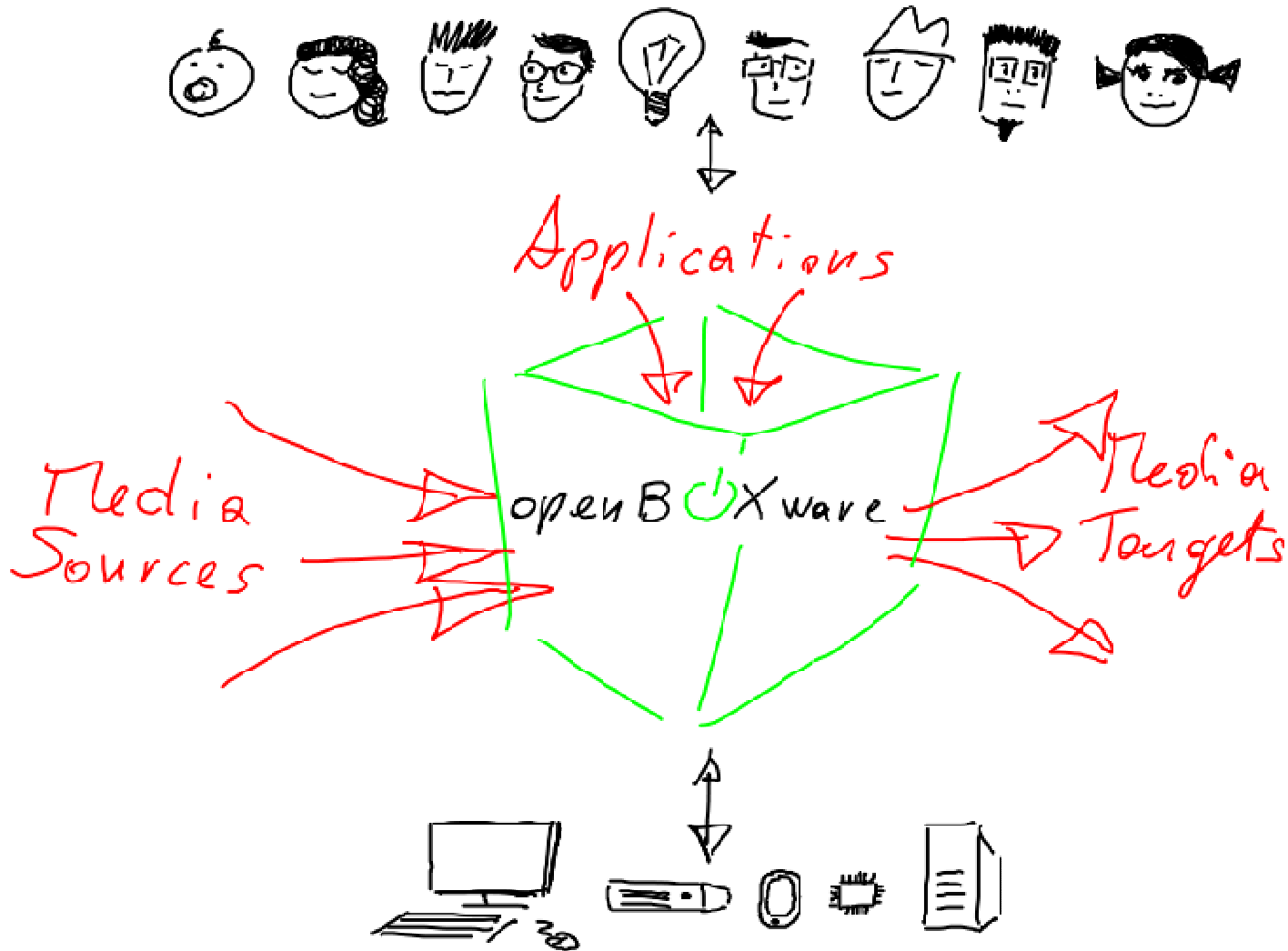
Corso di Laurea in Informatica Applicata
Università degli Studi di Urbino
21 dicembre 2010

Alessandro Bogliolo Lorenz Klopfenstein Saverio Delpriori



openBOXware framework

Concept

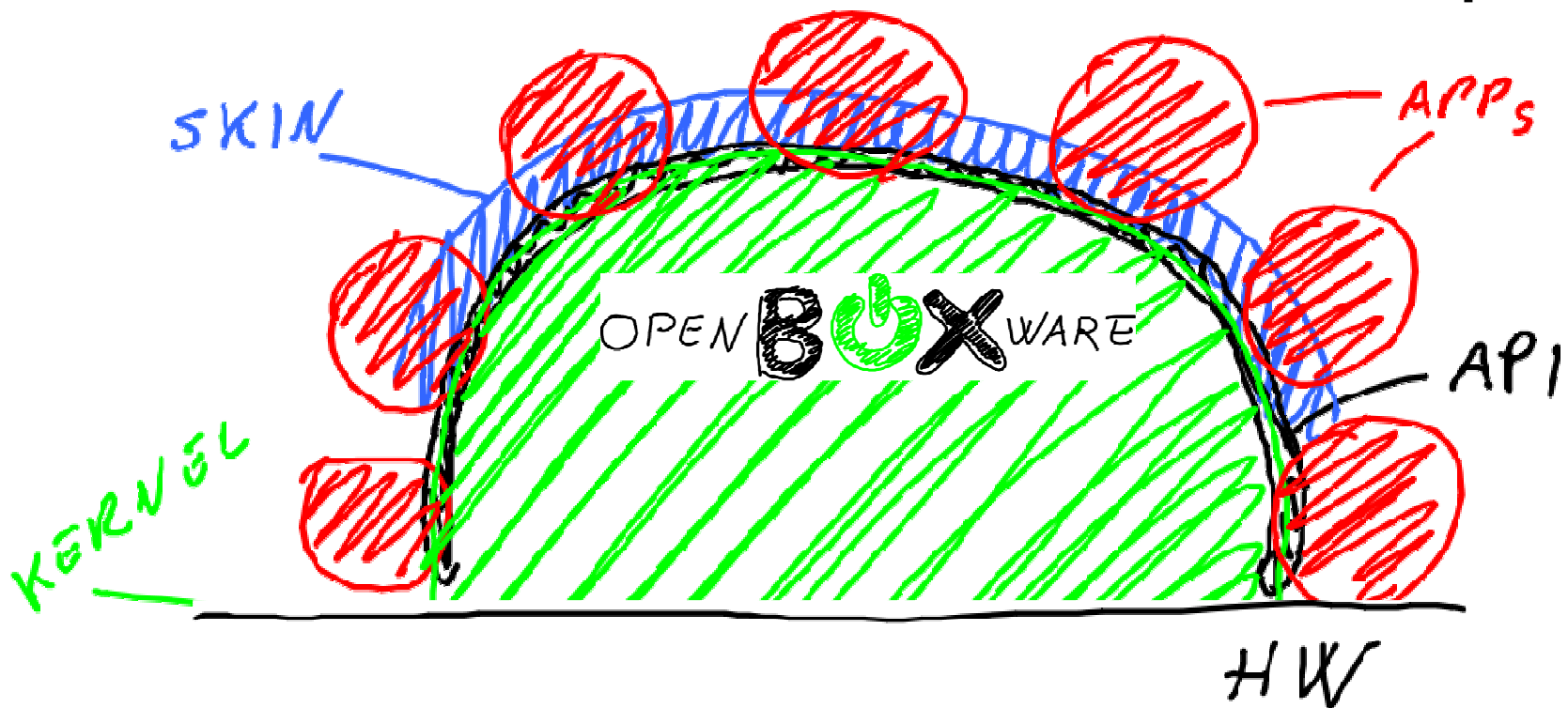


Concept

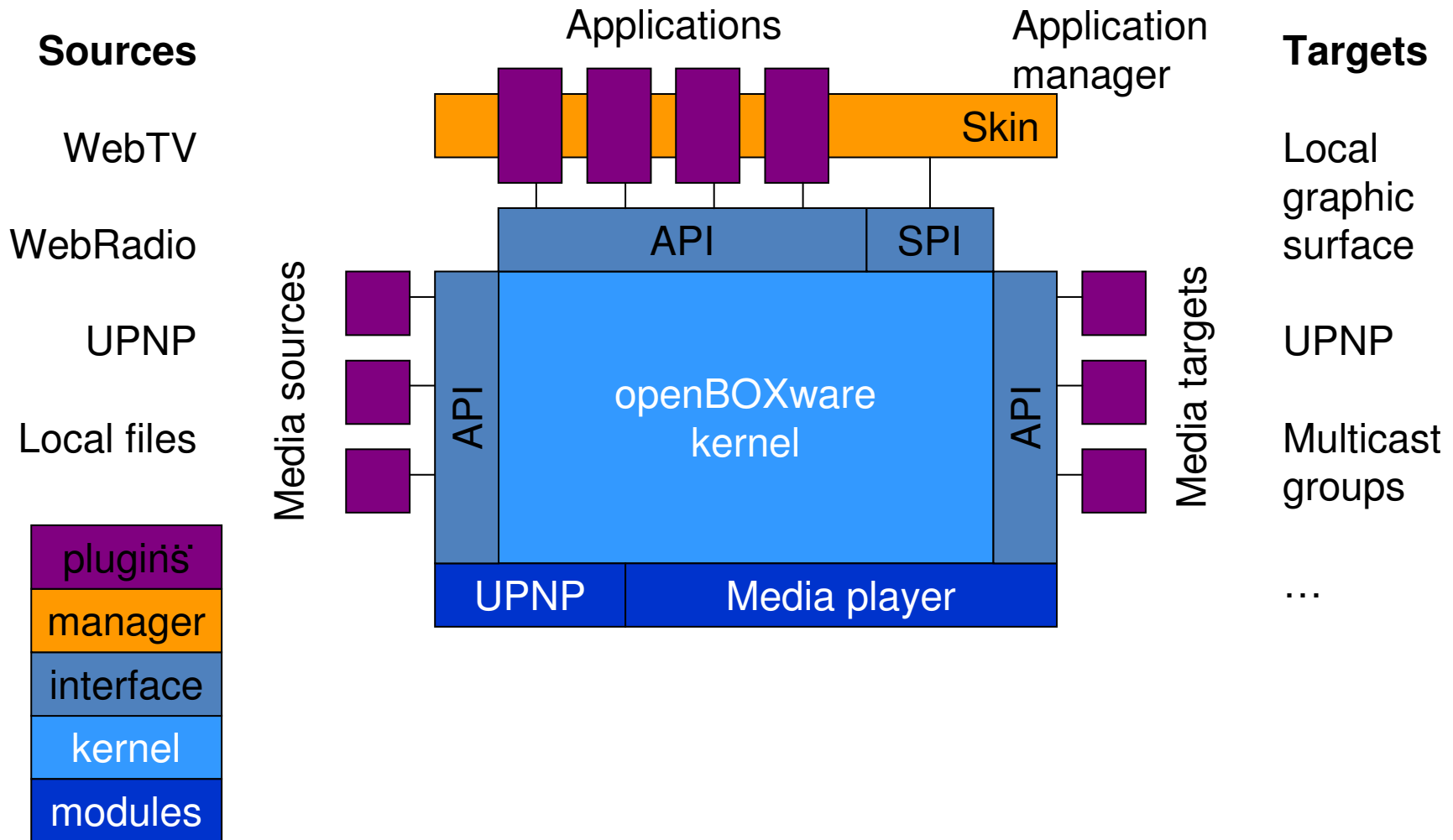


users

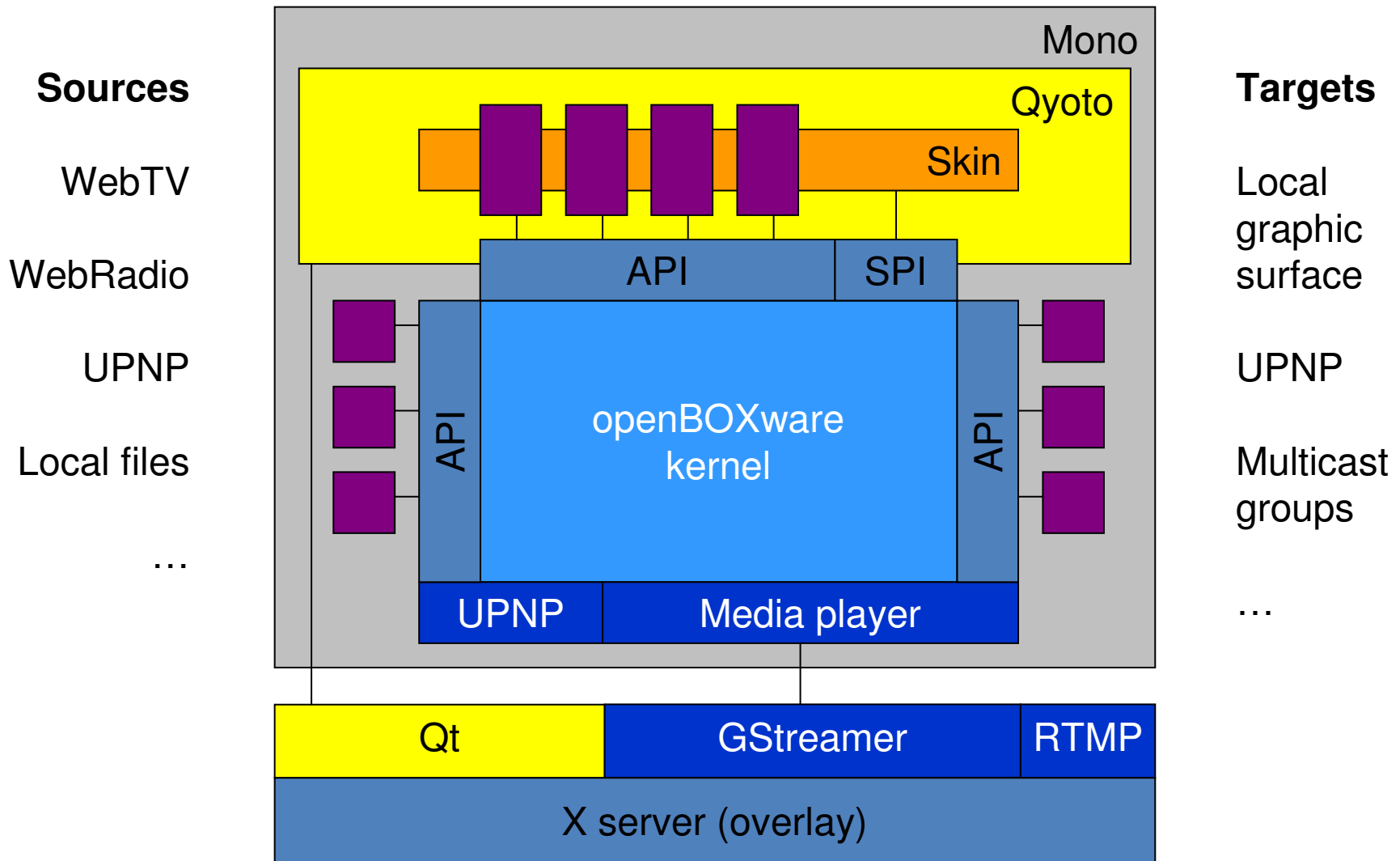
developers



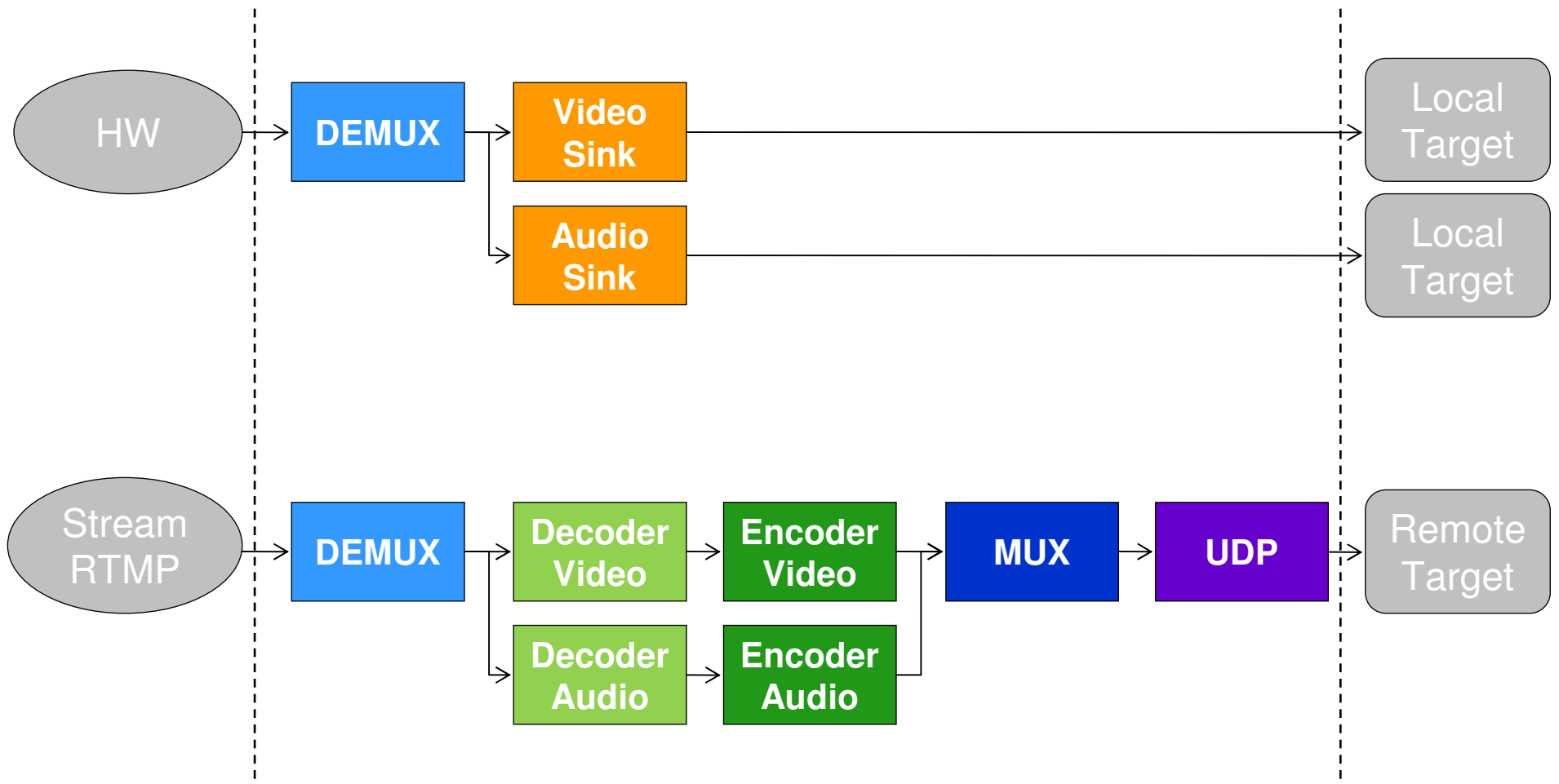
Framework



Environment



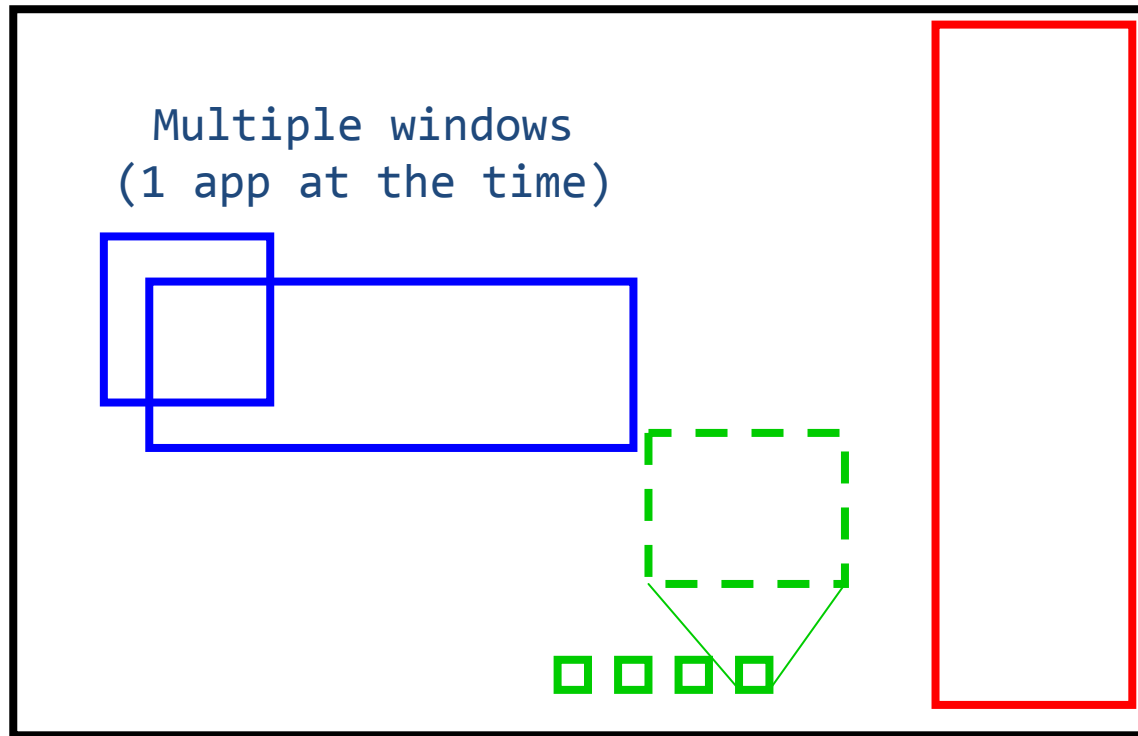
Multimedia pipelines



Execution modes



Full screen (1 app at the time)



Side bar
(1 app
at the time)

All other apps
in background

Notification of events and dialog boxes
(FIFO queue)

Personalization

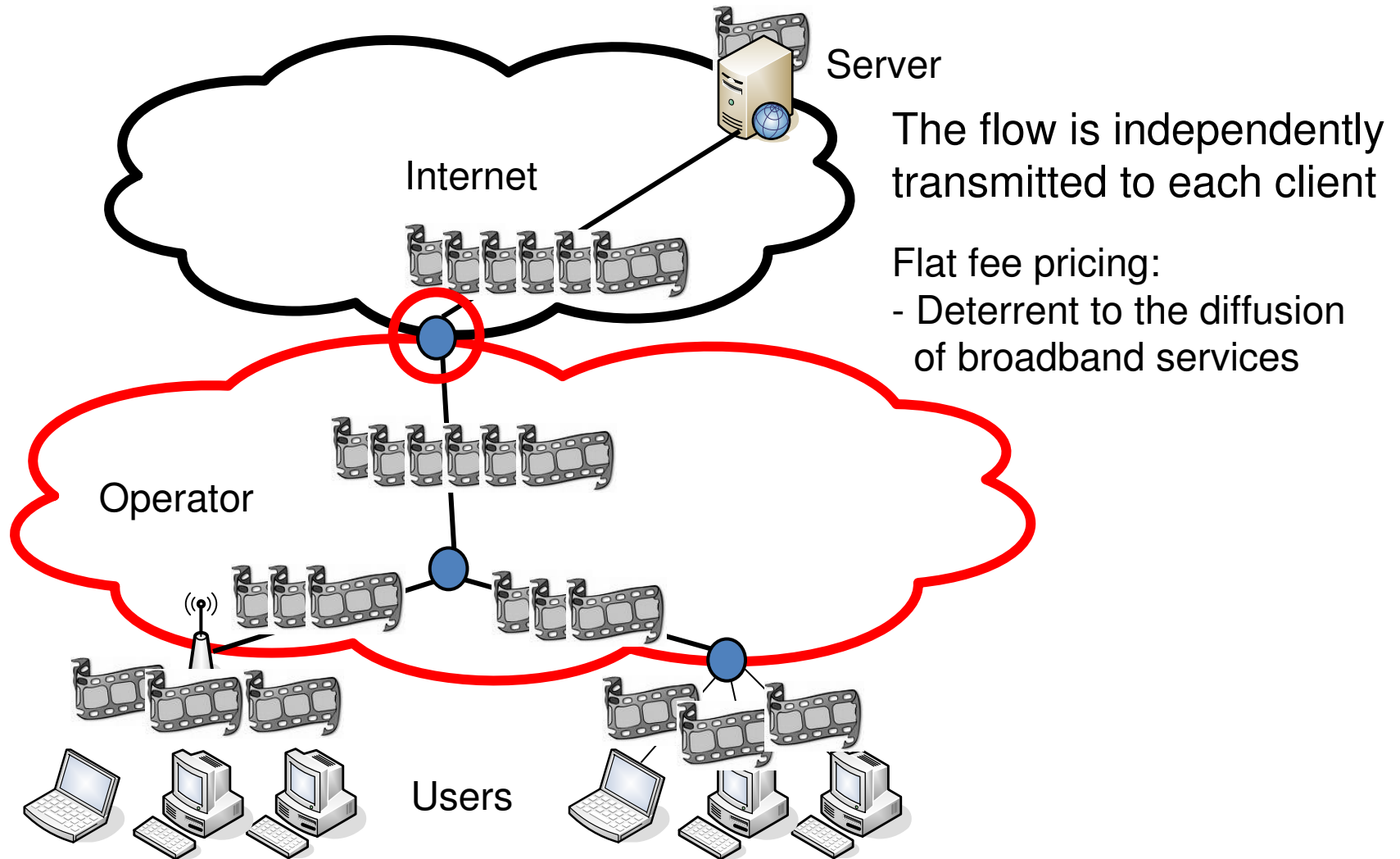


- Vendor's perspective
 - Porting of the kernel on different HW platforms
 - Development of brand-specific skins
 - Development of product-specific addins
 - General/specific application stores
- User's perspective
 - Choice of the skin
 - Choice of addins
 - Development of new addins

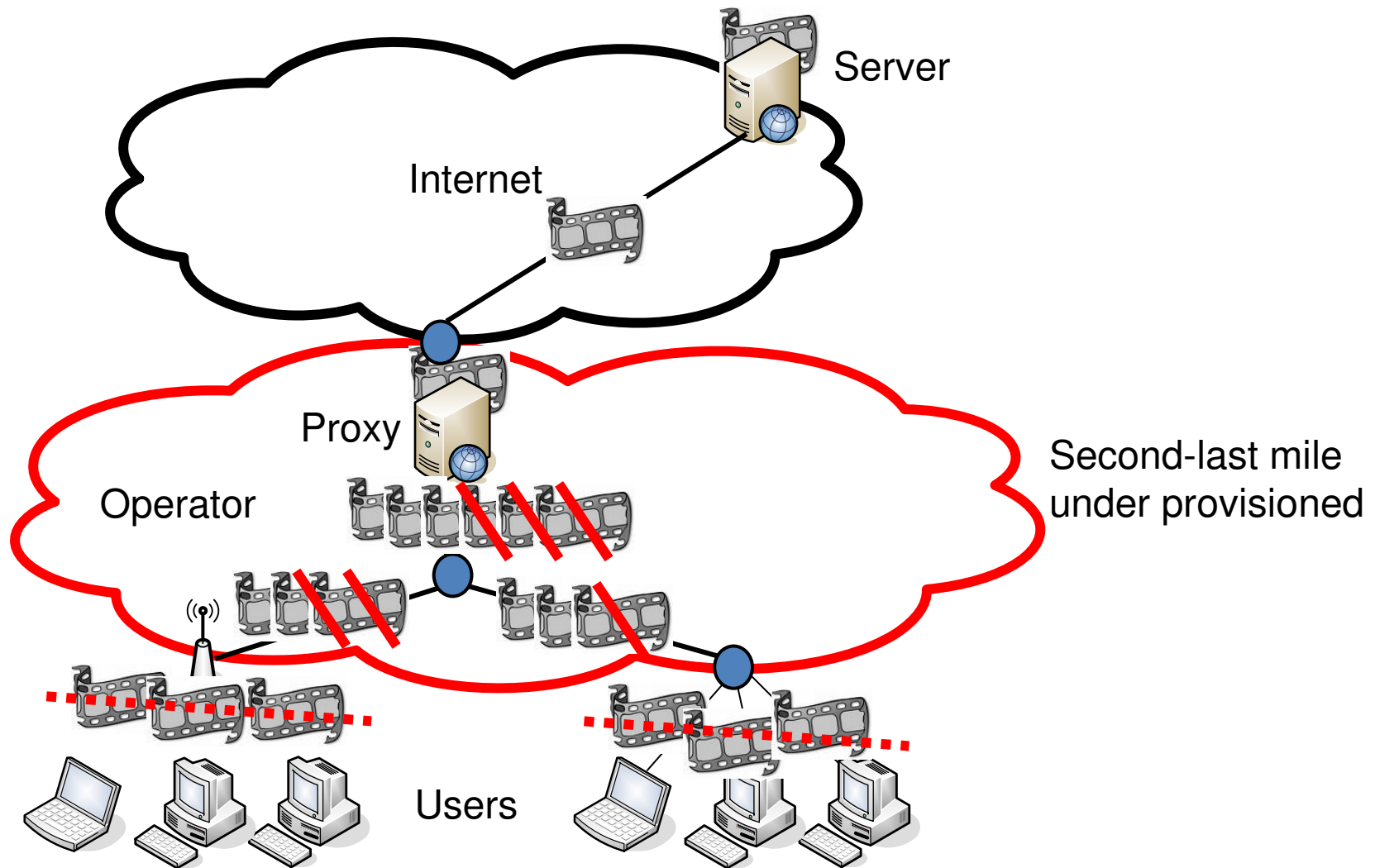


Networking issues

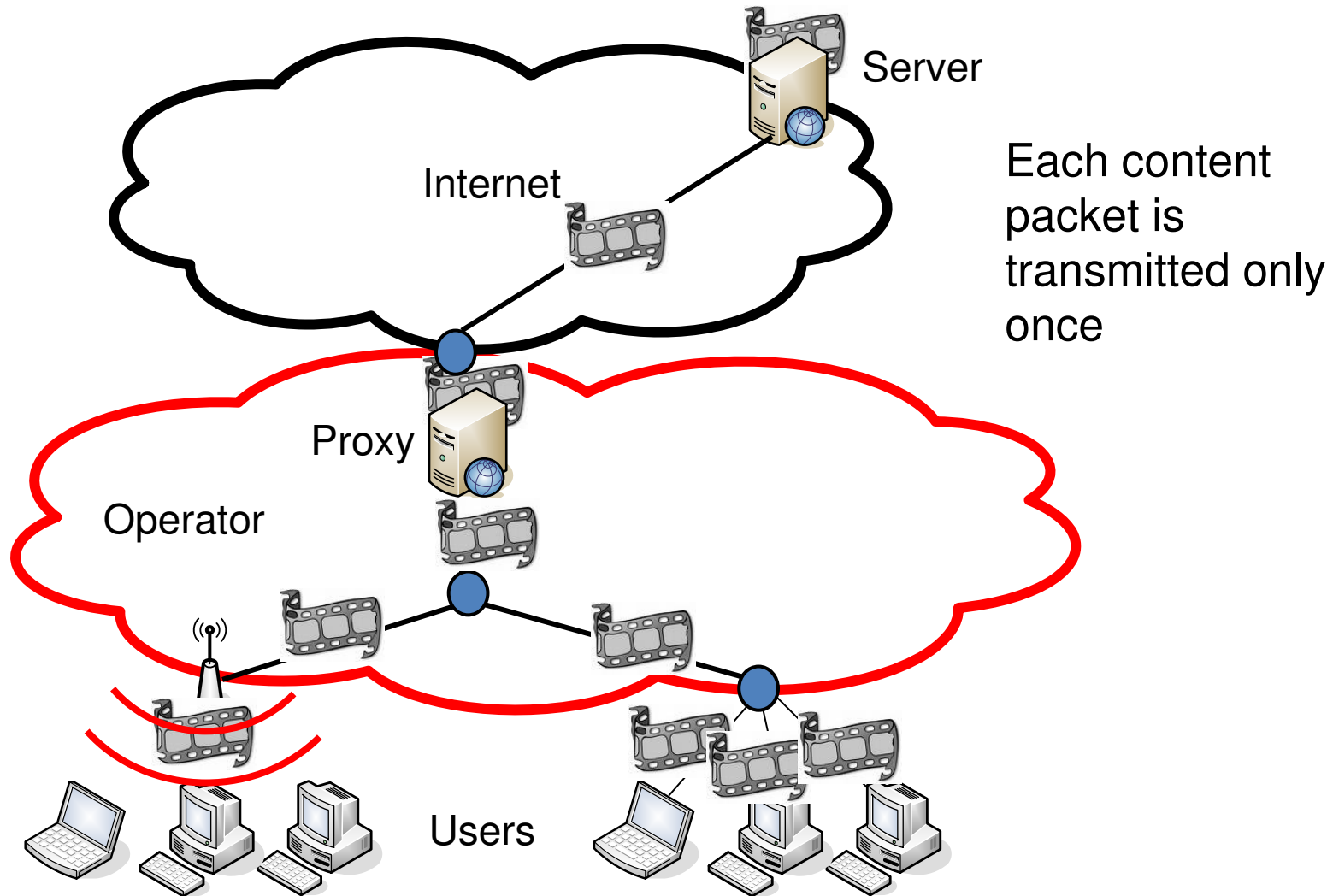
UNIcast



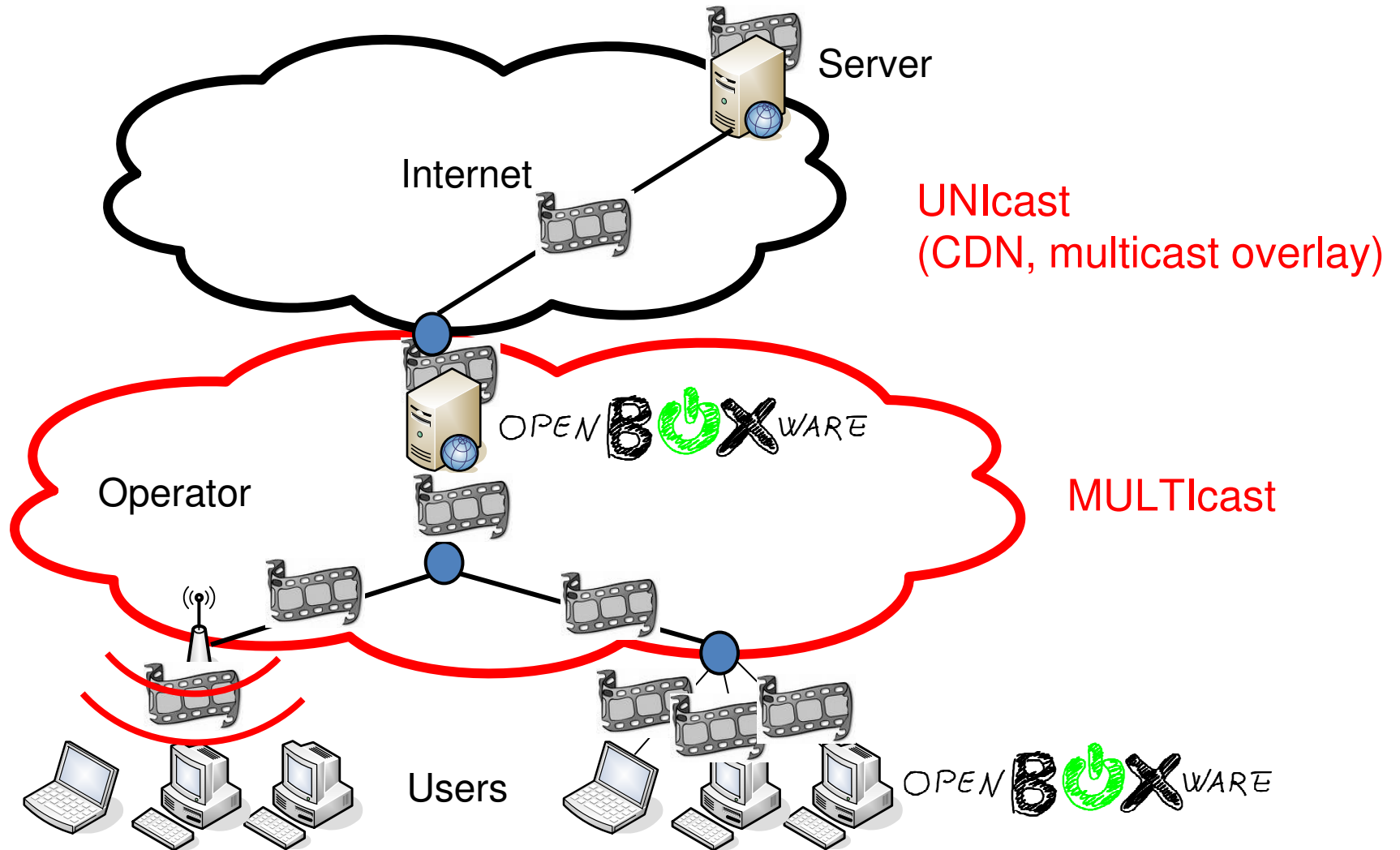
Content Delivery Networks



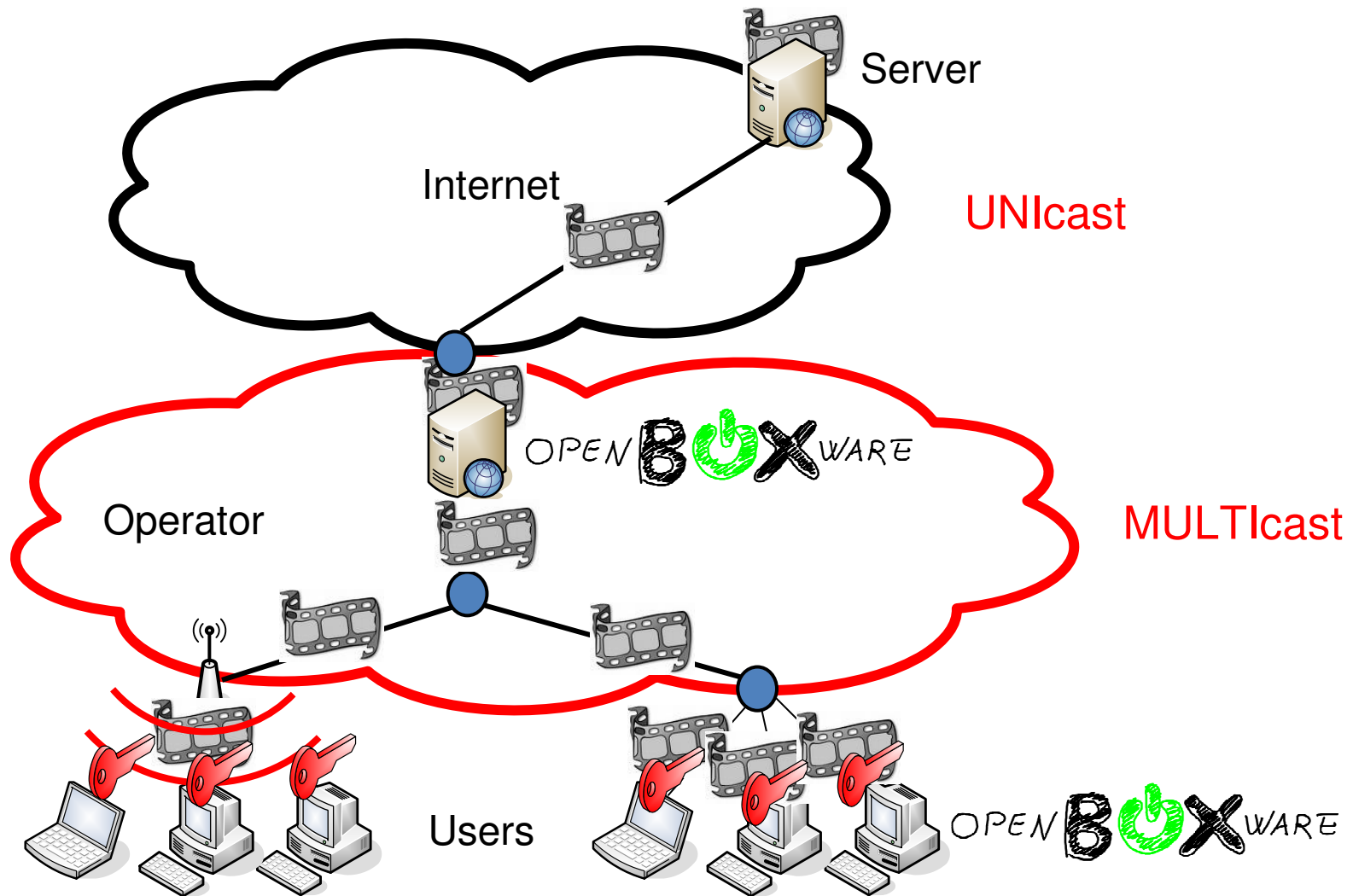
MULTIcast: in principle



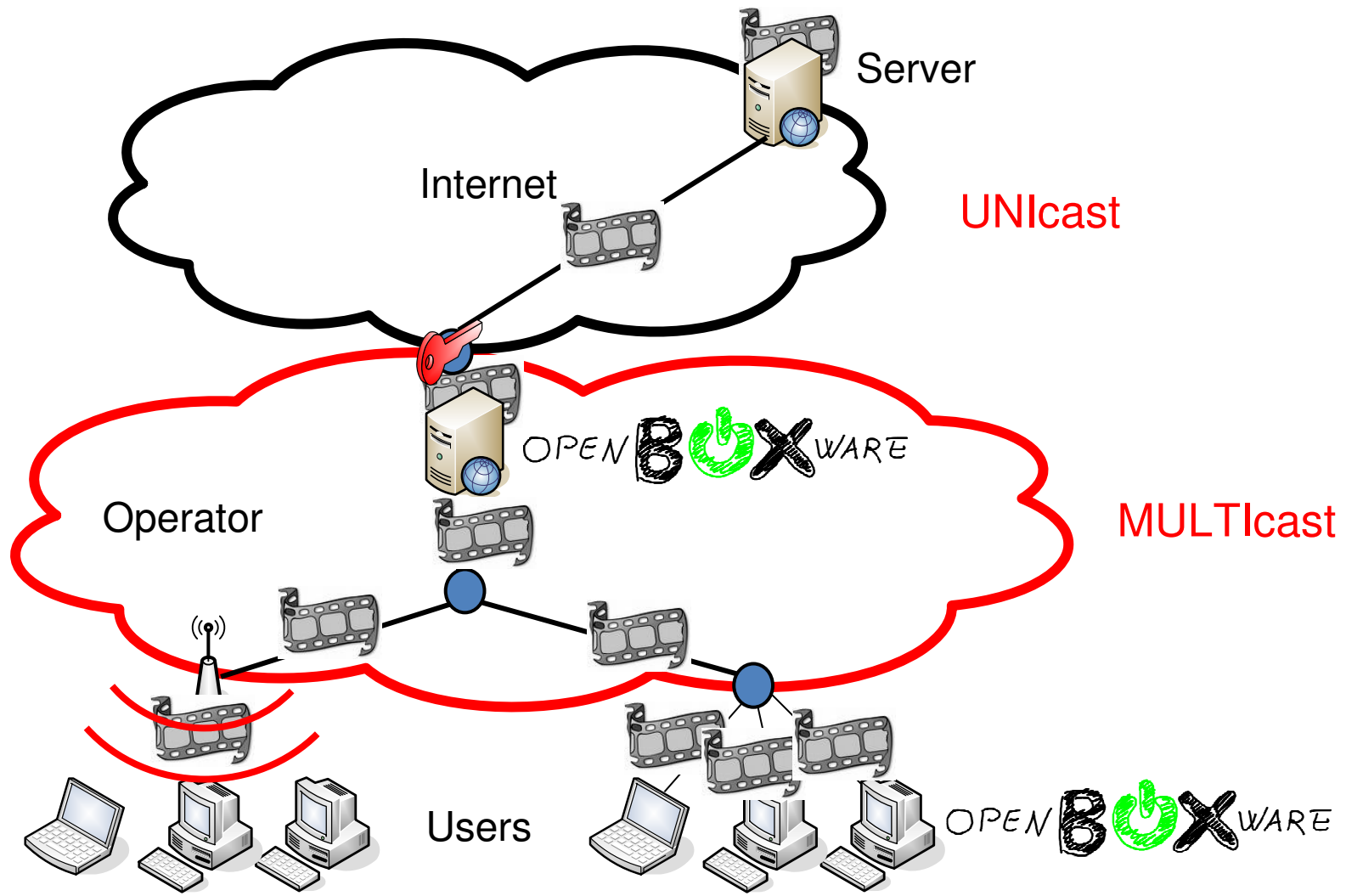
MULTIcast: in practice



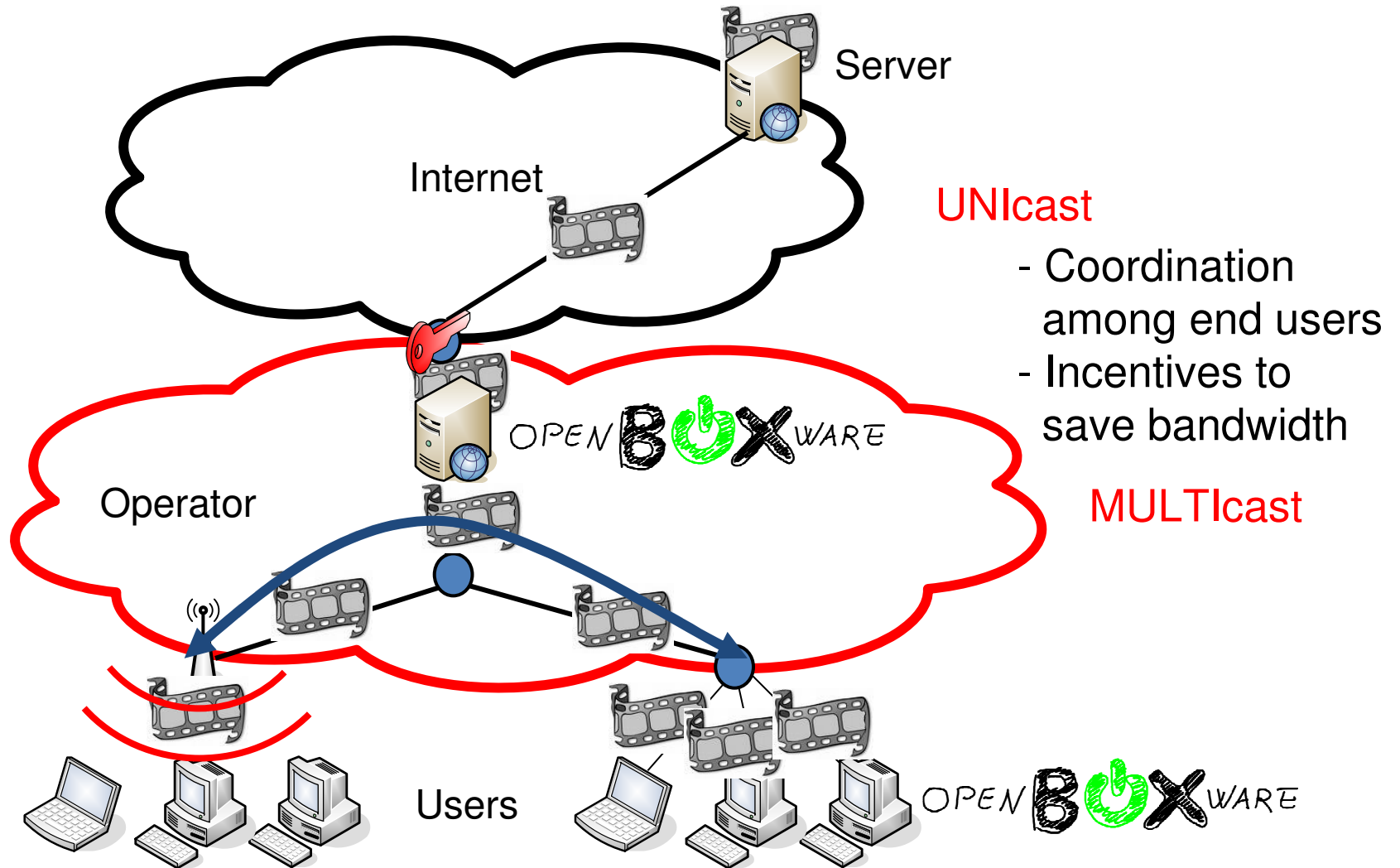
Closed network



Open network: free on air IPTV



Bandwidth sharing



New scenarios

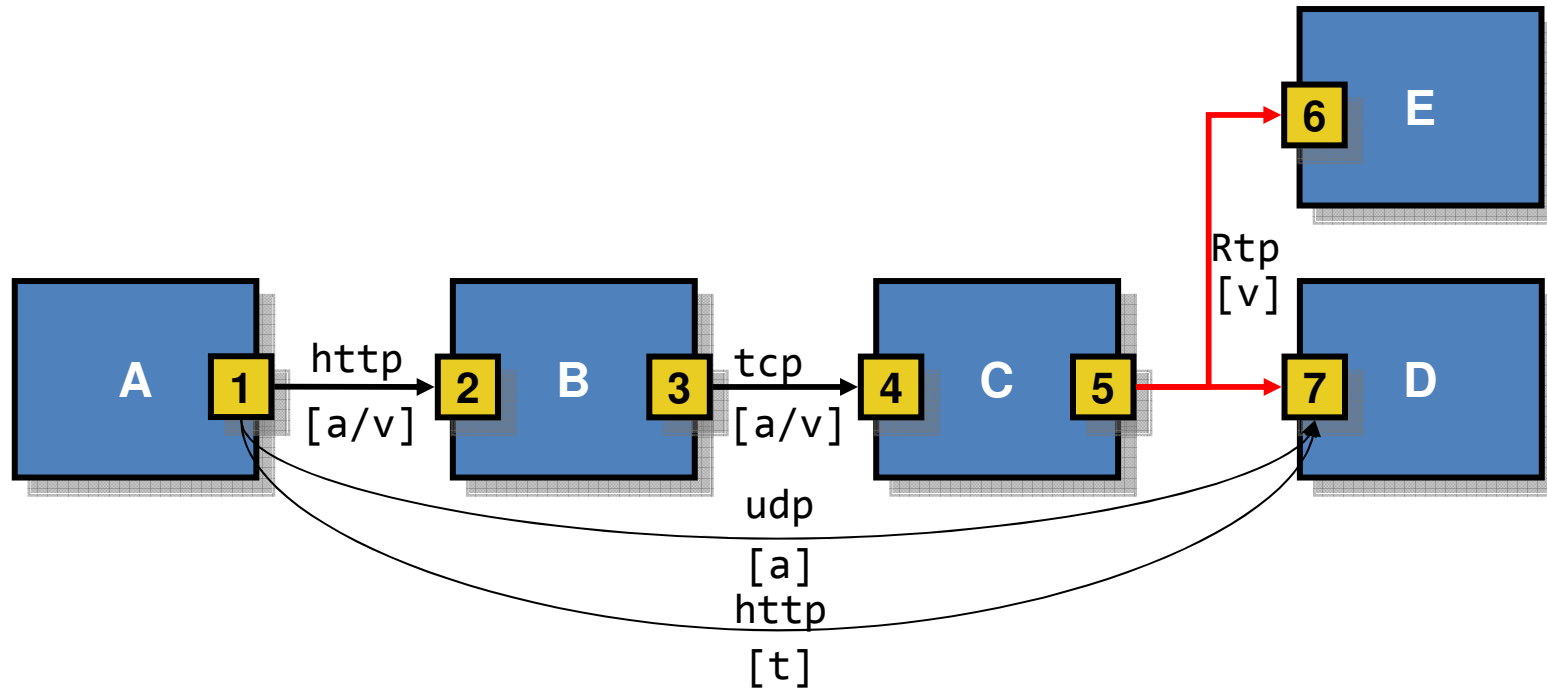


- Fixed cost for the diffusion of multimedia contents independently of the number of users
- Free on air TV channels as added values for access network infrastructures
- Incentives to broadband penetration
- Contribution to bridge cultural and infrastructural digital divide
- Economic sustainability of access networks

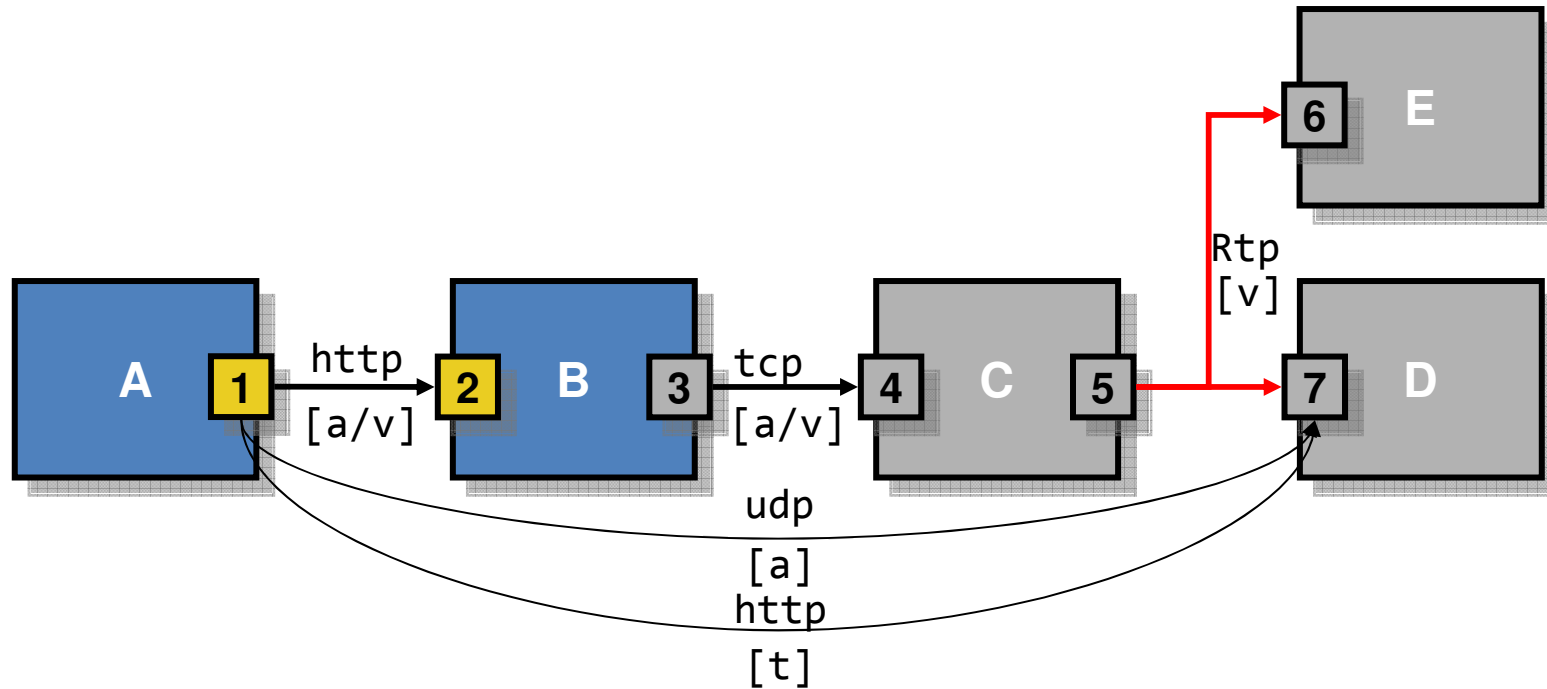


Advanced multimedia flows

Demo



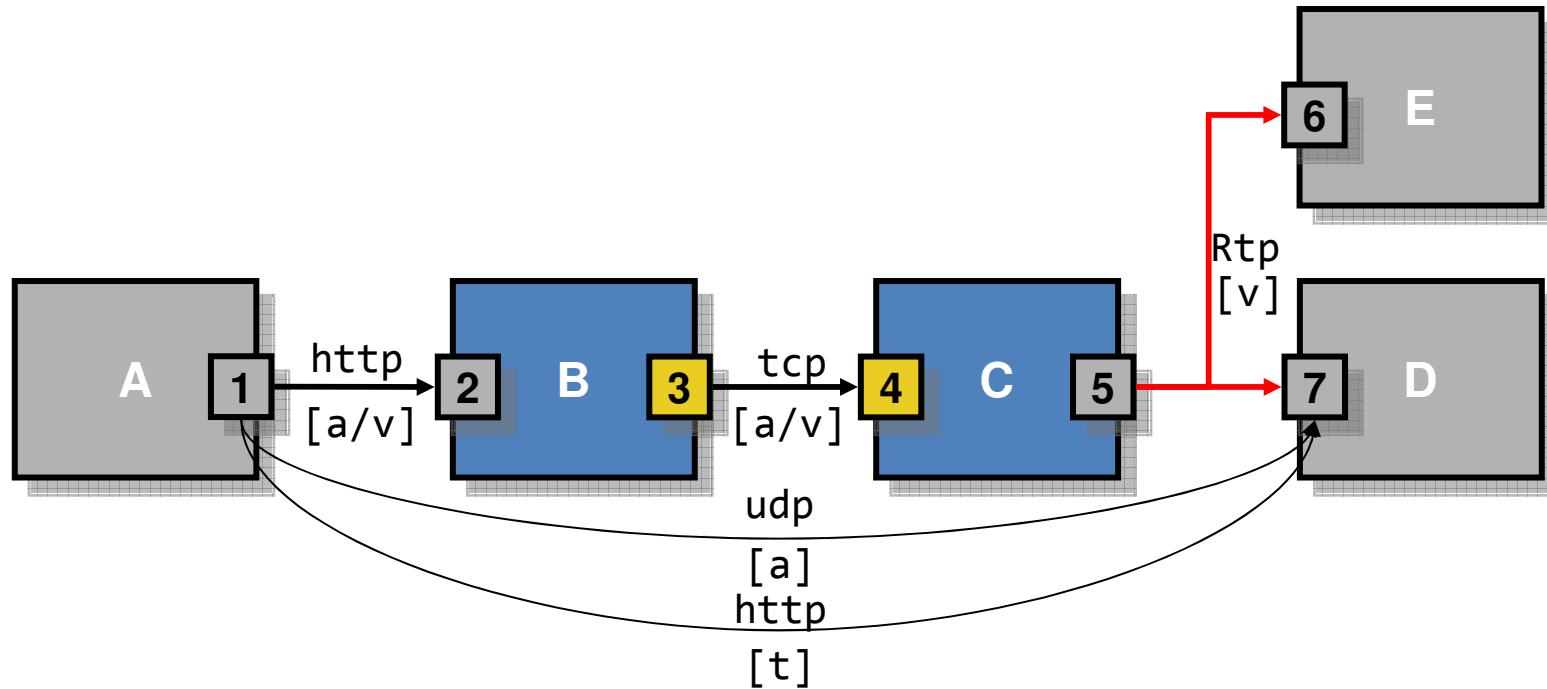
Demo [1st hop: HTTP]



1. openBOXware HTTP server
2. Media Source HTTP (client)

MP4 container (H.264 + AAC)

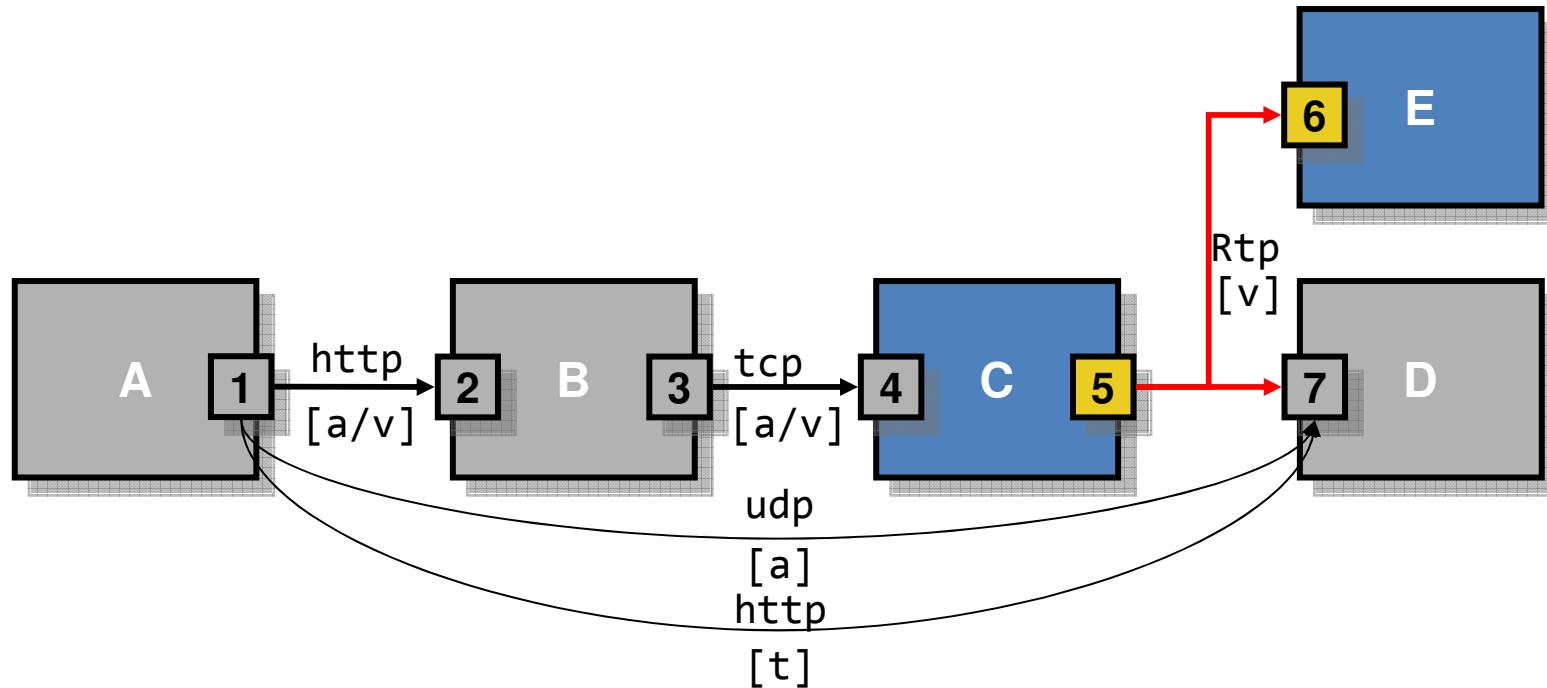
Demo [2nd hop: TCP]



- 3. Media target TCP (client)
- 4. Media source TCP (server)

MP4 container without any transcoding
PUSH mode streaming

Demo [3rd hop: multicast]



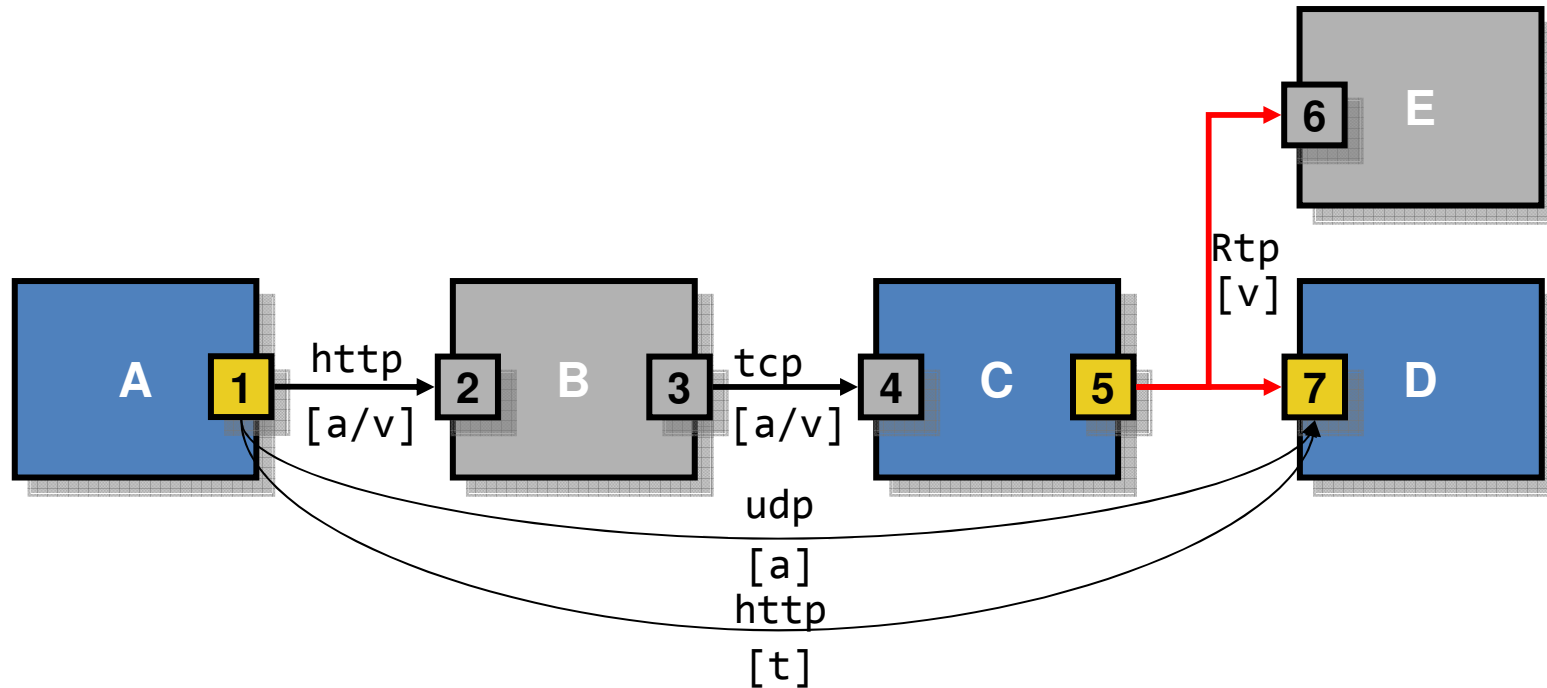
5. Media target multicast RTP over UDP

6. Media source SDP (client)

SDP descriptor provided by A.1

RTP container (H.264 encoding with headers - corrupted audio)

Demo [enhanced multicast]



1. Unicast Audio UDP (AAC with ADTS header) + subtitles (srt)
5. Media target multicast RTP over UDP
7. Application including
 - media source SDP (video)
 - custom pipeline UDP (audio)
 - subtitle widget