



A Cross Layer Architecture for Real-time Applications

M. Boutabia, L. Rojas, H. Afifi

*RS2M Dept., Institut National des
Telecommunications (INT)*

91011 Evry France

May 2007



Outline

- Introduction
- The architecture
- Implementation
- Performance Evaluation
- Media adaptation
- Conclusion



Introduction

- The majority of future communications will be wireless & many access technologies will coexist
- All communications devices have to support kinds of mobility (terminal mobility, session mobility.. etc) it will be a necessity
- For the supporting of those new services on heterogeneous technologies, there are important works such as SIP, Mobile IP, etc.
- Nevertheless, for time constrained applications, the service provided by those infrastructures present some limitations.



Introduction

Mobile IP → network level

- triangular routing
- encapsulation overhead
- the need of a fixed ip adress

SIP → application level

- heavy (exchange of several messages)
- important handoff delay (more than 2 s)



Introduction

- These limitations can be avoided if the number of exchanged messages is reduced .
- A simplified and cross-layer approach can provide a more suitable seamless service for time constrained data flows.



Introduction

The challenge in providing a suitable seamless services for time constrained applications is :

- A minimal handover delay
- Media adaptation taking into account devices capacities and network changes



Introduction

RTSP (real time streaming protocol) → starts and controls the media session

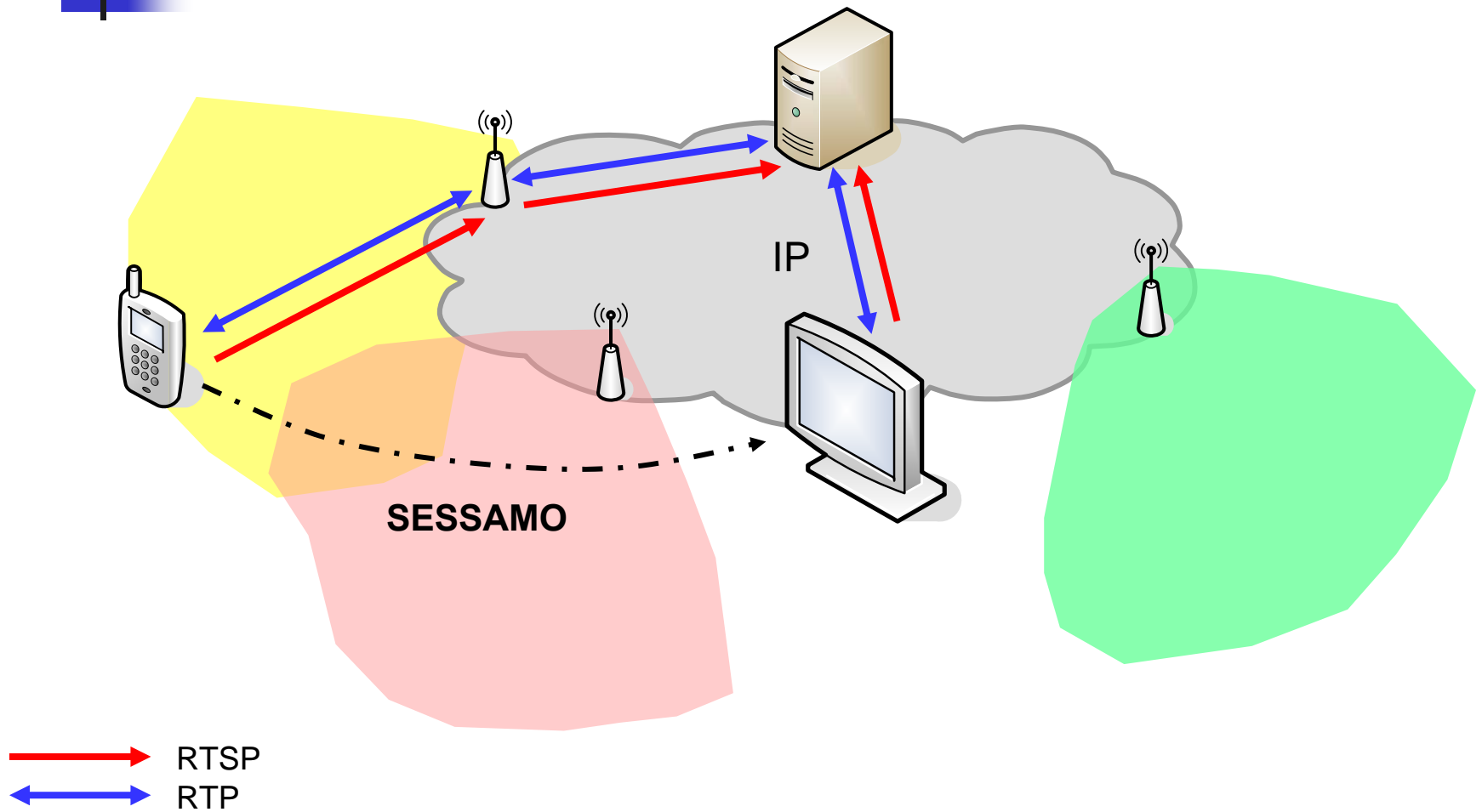
- connectionless
- extensible
- easy to analyse
- secure
- outband and transport independent



The architecture

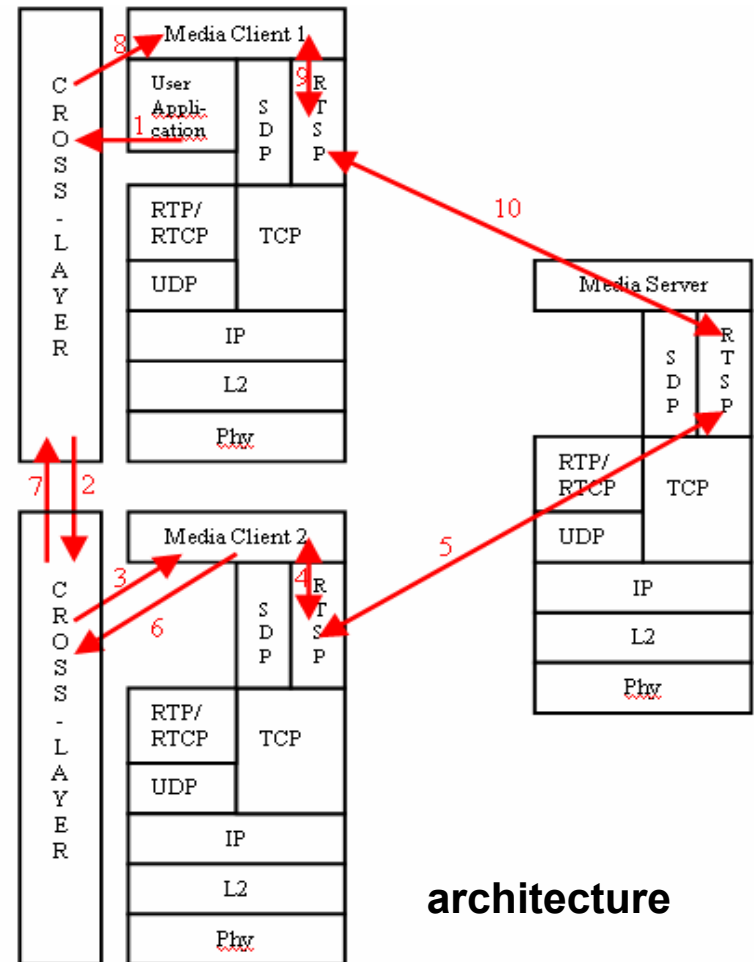
- It is a cross-layer architecture
 - Informations about lower layers can be known by upper layers
 - Upper layers can take decisions about the technology interface to be used
- It simplify the procedures to handle session mobility
 - It reduces the number of control messages required to manage mobility

The architecture



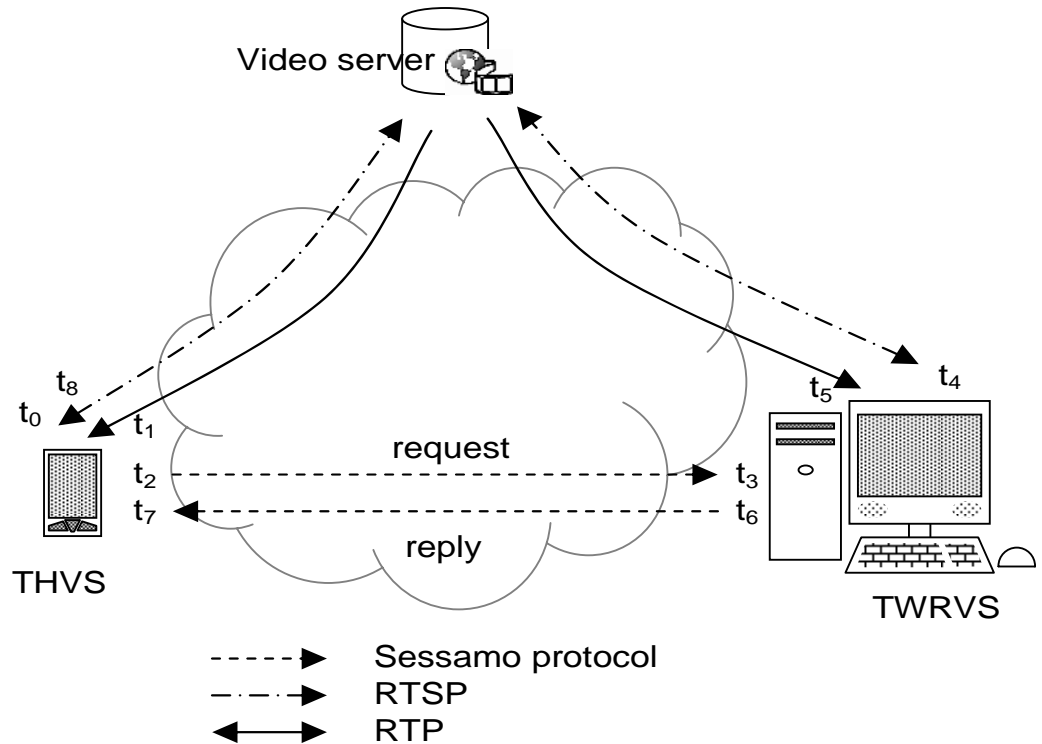
The architecture

- SESSAMO is a client to client protocol light and easy to analyse
- send only one message
- all required parameters are included in that message (instant, URL ...)
- the server is not affected (no change)



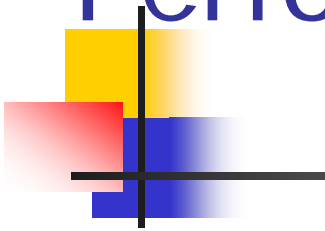
Implementation

- video server LIVE555
- multimedia player Mplayer
- OS Linux Redhat 9



Test framework

Performances evaluation



RTT	2.329 ms
$t_{\text{shd}} = (t_5 - t_3) + (t_3 - t_2)^\dagger$	248.859 ms
$t_{\text{os}} = (t_8 - t_7) + (t_6 - t_7)^\ddagger$	158.803 ms
$t_{\text{stp}} = t_4 - t_3$	164.522 ms

t_{shd} = session handover delay

t_{os} = overlap session time

t_{stp} = starting time of the RTSP player

Important : t_{stp} present 66% of the total session handover delay

$\dagger t_3 - t_2 \approx \text{RTT}/2$

$\ddagger t_6 - t_7 \approx \text{RTT}/2$



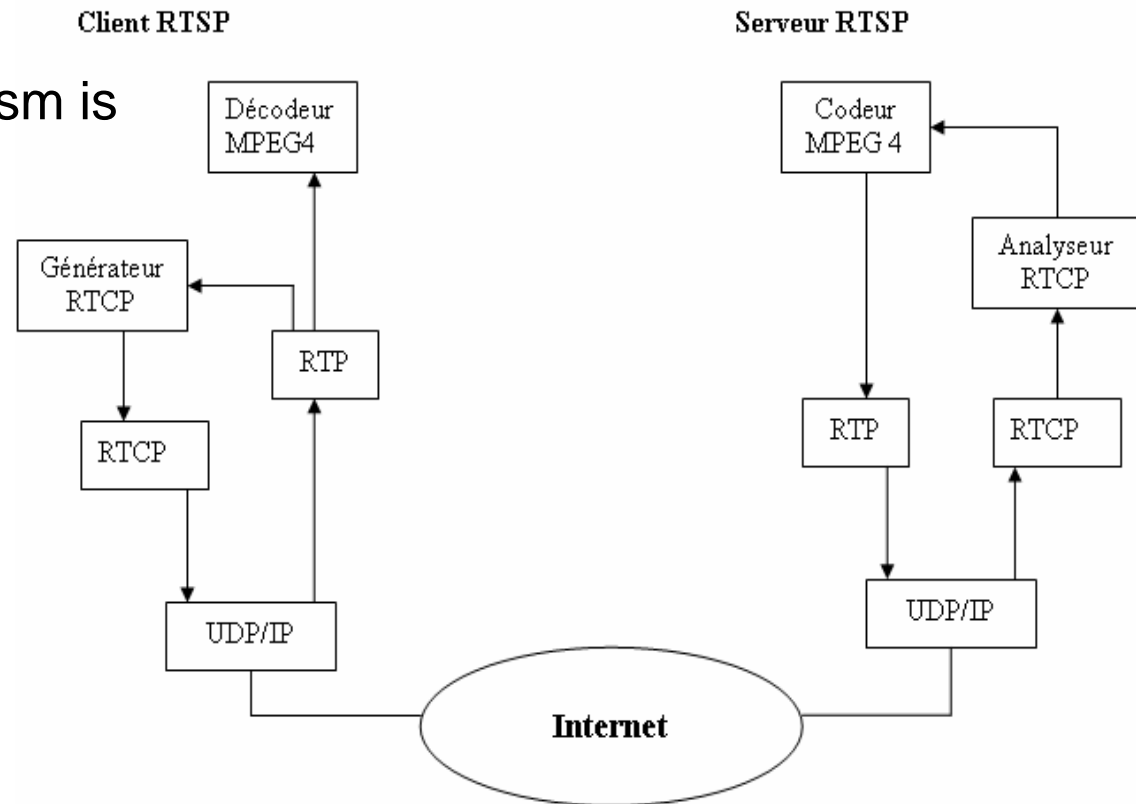
Media adaptation

- variable rate according to the activity in the video
- no guarantee in IP network (best effort)
- variable network state
- Different bandwidth from a network to another
- use of UDP in multimedia transport

Media adaptation

The proposed adaptation mechanism is based on :

- layer coding of MPEG4
- RTCP error reports





Conclusion

We designed and implemented a cross-layer architecture for session mobility

We measure the impact of the session mobility on the handover process

The results show that our architecture is the most adapted for multimedia and real time applications

Our results that mobility can be managed without modifying standard protocols such as RTSP



THE END

Thank you!!