Design and Implementation of Service Migration

(服務遷徙之設計與實現)

Chin-Shiuh Shieh (謝欽旭)

Department of Electronic Engineering
National Kaohsiung University of Science and Technology, Taiwan

Outline

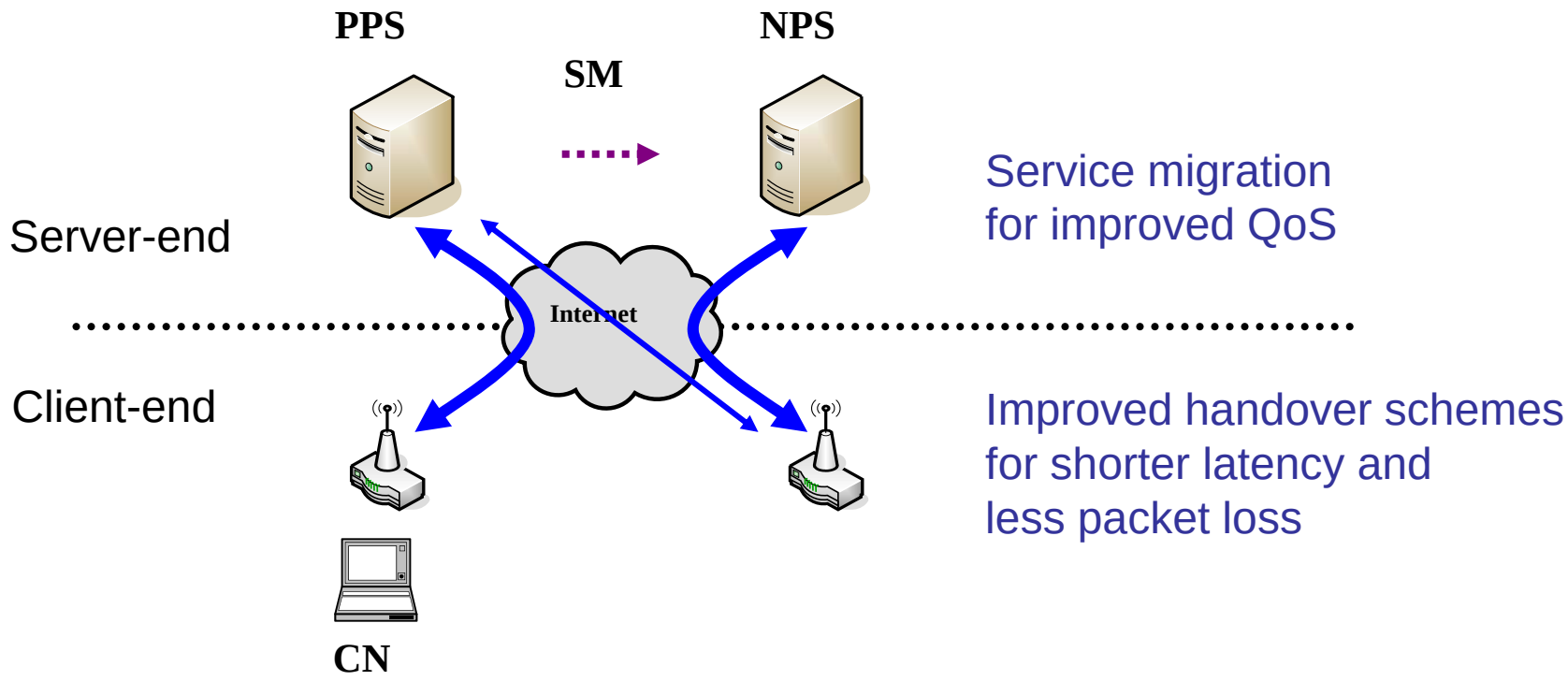
- Abstract
- Introduction
- Related Techniques
- Service Migration
- TCP-based Service Migration
- UDP-based Service Migration
- SCTP-based Service Migration
- SDN-based Service Migration
- Conclusions

Abstract

- QoS provisioning in mobile wireless networks from complementary perspectives:

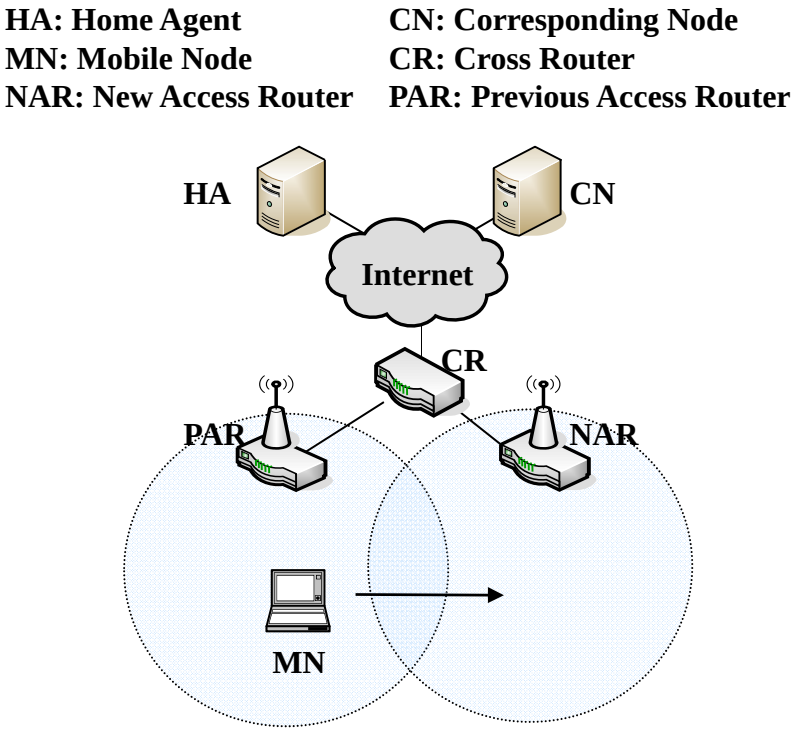
PPS: Previous Proximity Server
CN: Client Node

NPS: New Proximity Server
SM: Service Migration



Introduction

- Handover
  - The procedure that needs to be taken as a Mobile Node (MN) moving across different network domains to retain the continuation of ongoing services.
  - Challenges: Latency, packet loss, …



Introduction (cont)

- Mobility supports
  - Mobile Internet Protocol Version 4 (MIPv4)
  - Mobile Internet Protocol Version 6 (MIPv6)
  - Hierarchical Mobile IPv6 (HMIPv6)
  - Fast Handover Mobile IPv6 (FMIPv6)
  - …
  - Anycast-supported FMIPv6 (AFMIPv6)
  - Multicast-supported FMIPv6 (MFMIPv6)

Introduction (cont)

- Internet had become a primary media for
  - Communication
  - Distribution of digitized data
- Content distribution services
  - world-wide-web, file transferring, multi-media casting, …
- Client-server architecture

  → Bottleneck

Introduction (cont)

- Content distribution systems
  - Multiple servers
  - Same contents and same services
- Service migration
  - Mobility results in topology and proximity change.
  - Migrate an ongoing service from distant server to a nearer one.

Related Techniques

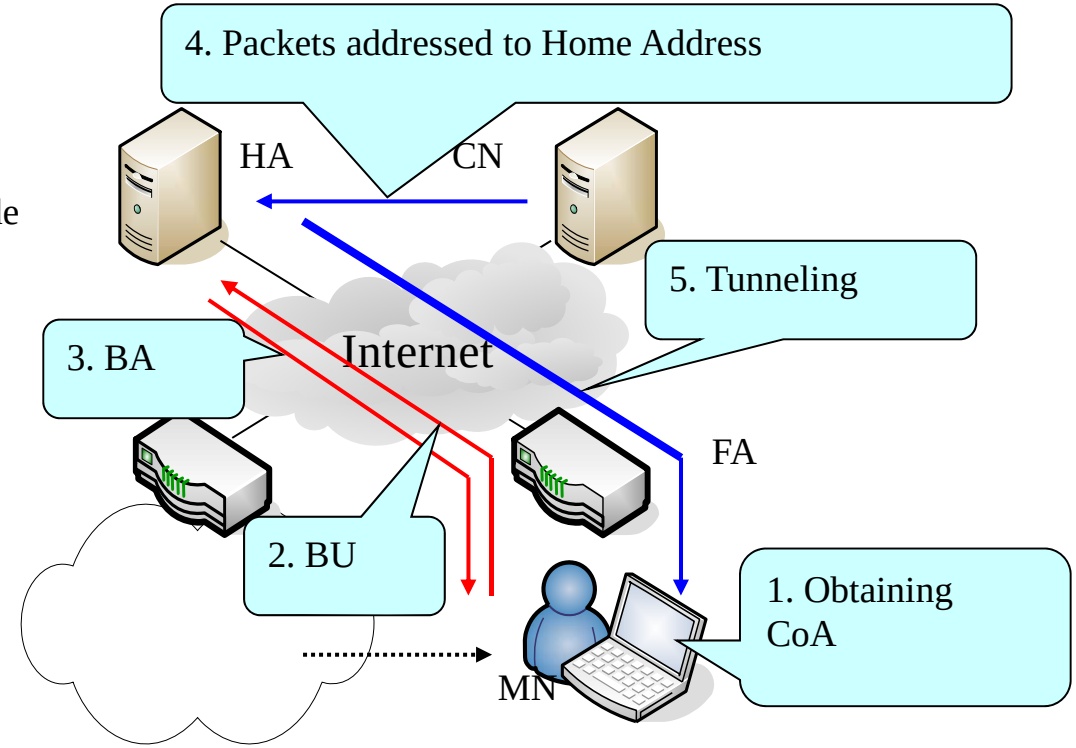
- Mobility supports in mobile wireless networks
- Content distribution systems
- Process migration

Mobility Supports in Mobile Wireless Networks

- Handover at different layers
  - Application-layer
    - VoIP (Voice over IP) using SIP (Session Initiation Protocol)
  - Transport-layer
    - Dual-homing in SCTP (Stream Control Transmission Protocol)
  - Network-layer
    - MIPv4, MIPv6, …

Mobility Supports in Mobile Wireless Networks - Mobile IPv4

HA: Home Agent
FA: Foreign Agent
MN: Mobile Node
CN: Correspondent Node
BU: Binding Update
BA: Binding ACK
CoA: Care-of Address



Mobility Supports in Mobile Wireless Networks - Mobile IPv6

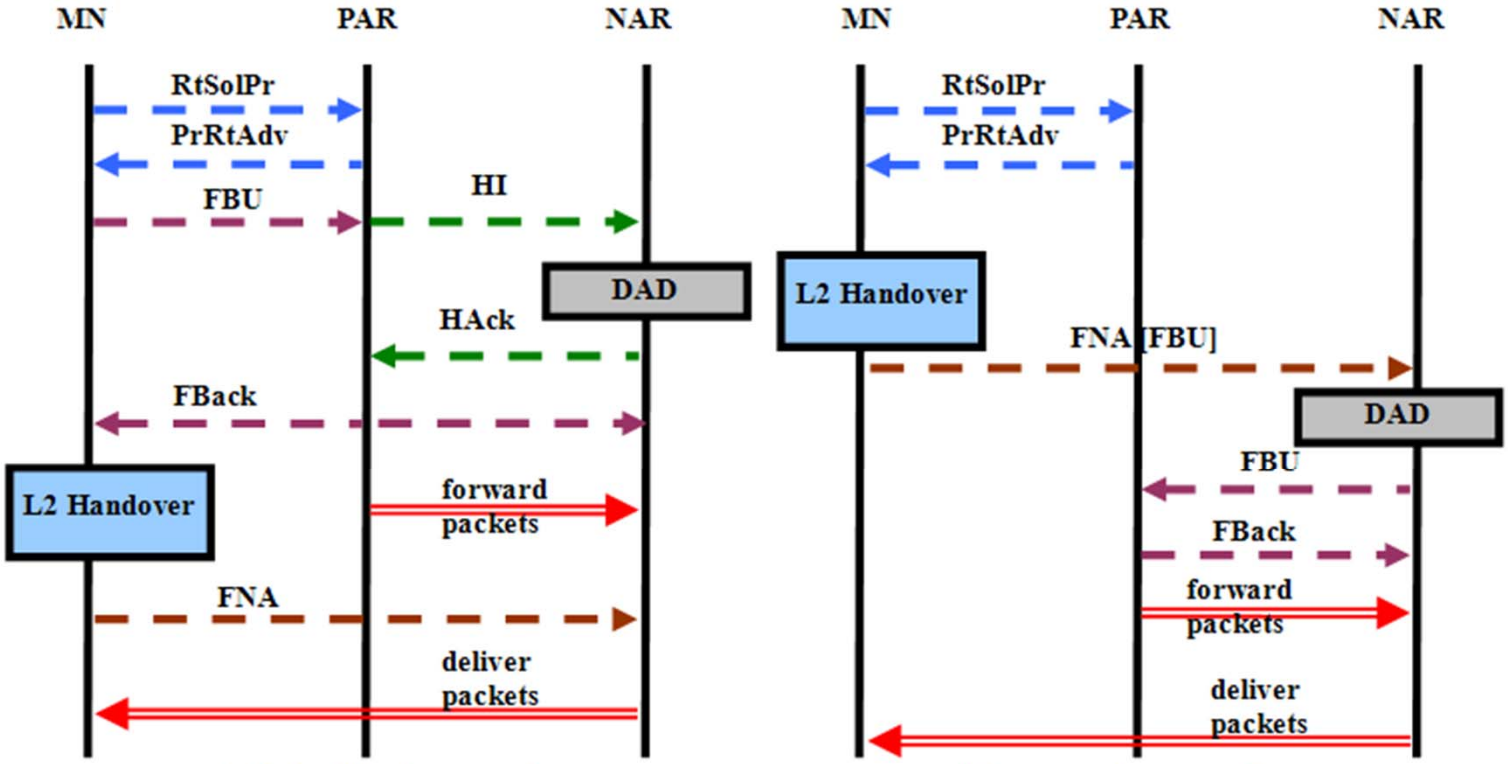
- Built-in mobility supports in IPv6
- Route optimization
- No need for Foreign Agent
- Address auto-configuration

Mobility Supports in Mobile Wireless Networks – Fast Handover MIP

- MN acquires new prospective CoA from NAR prior to handover.
- Advance time-consuming DAD operation.
- PAR buffer and forward packets to NAR.

Mobility Supports in Mobile Wireless Networks – Fast Handover MIP (cont)

RtSolPr: Router Solicitation for Proxy Advertisement
PrRtAdv: Proxy Router Advertisement
FBU: Fast Binding Update
FBack: Fast Binding Acknowledgement
HI: Handover Initiate
HAck: Handover Acknowledge
FNA: Fast Neighbor Advertisement



(a) Predictive mode (b) Reactive mode

Content Distribution Systems

- Multiple servers with the same contents
- Server selection
  - Portal node
  - Domain Name System (DNS) / Dynamic DNS
- An initially best server might be no longer the best as CN moves

  ➔ Service migration

Process Migration

- Migration of a running process from one system to another for load balancing
- Operations
  - Suspension of a running process
  - Transferring of its state to another system
  - Resuming of its execution on the target system

Process Migration (cont)

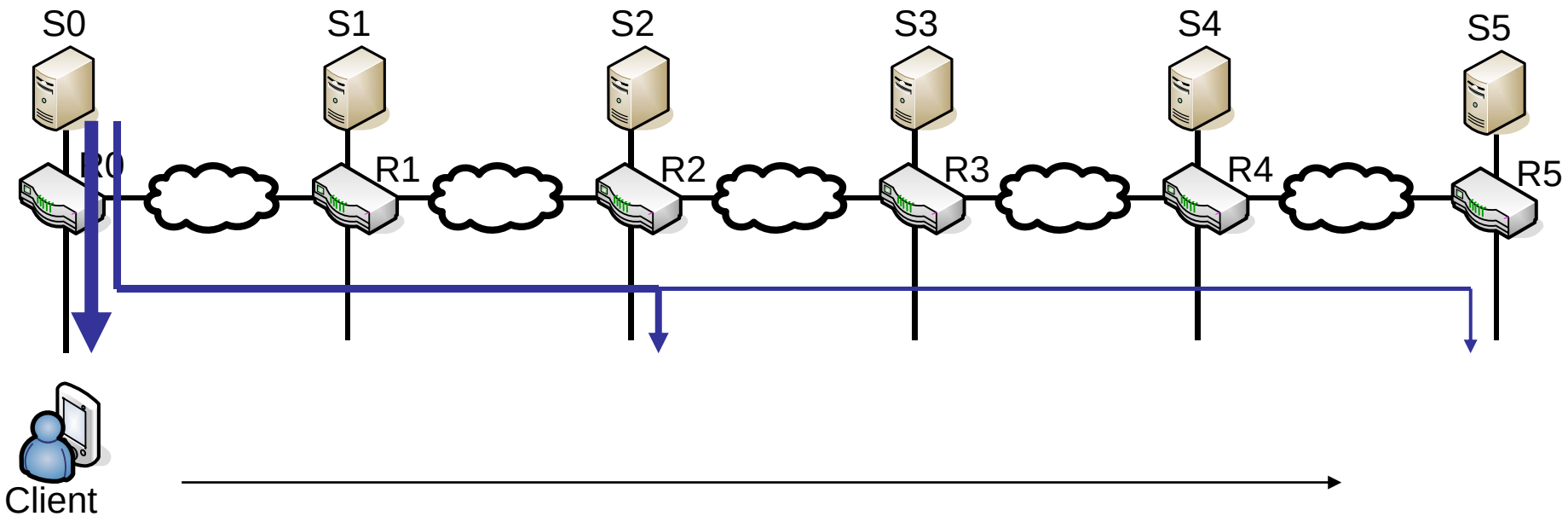
- Available implementations connection migration
  - TCP Connection Endpoint Migration (Linux kernel)
  - Socket Shifting (Linux, loadable kernel module)
  - MIGSOCK (Linux TCP/IP Stack)
  - Mockets (Java, wrapper)
  - TCP Connection Passing (Linux kernel)

Service Migration

- QoS Provisioning in Mobile Environments by Service Migration
- A New Service Framework Featuring Service Migration

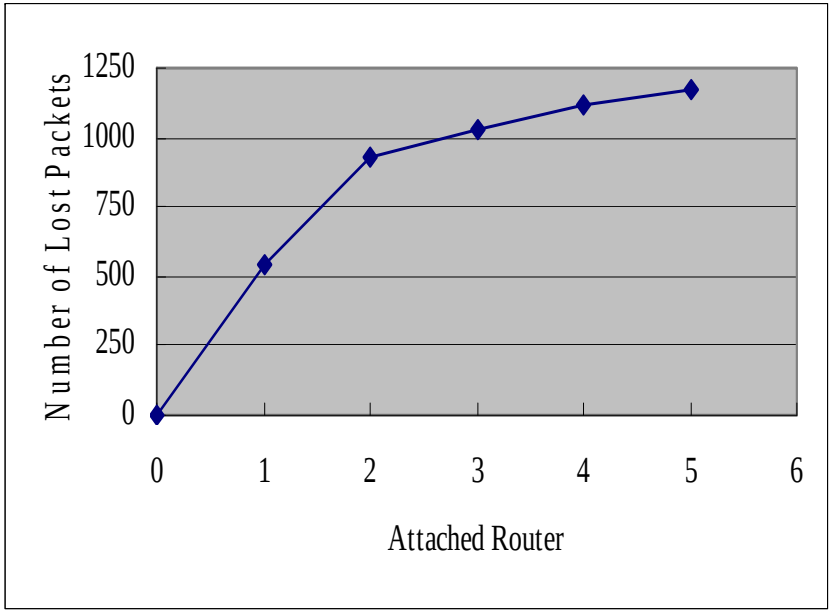
Service Migration (cont)

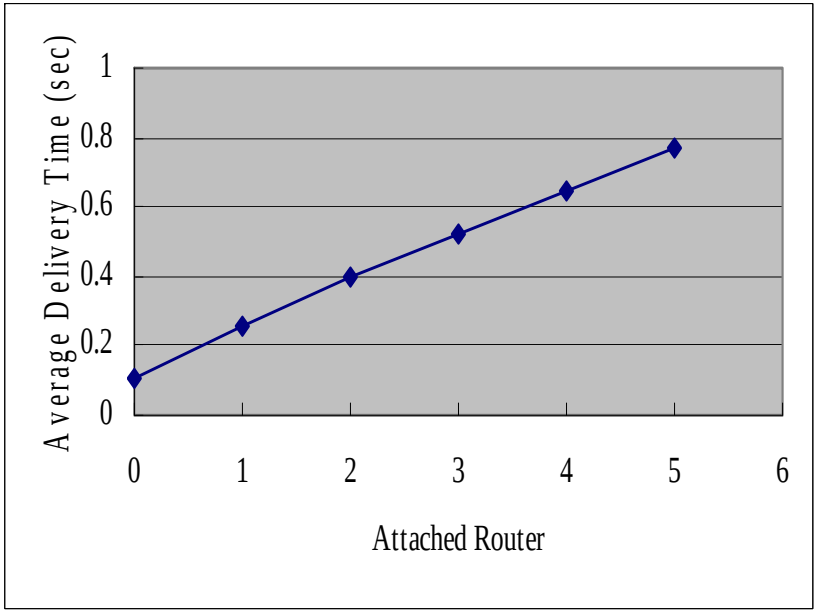
- Without service migration



Service Migration (cont)

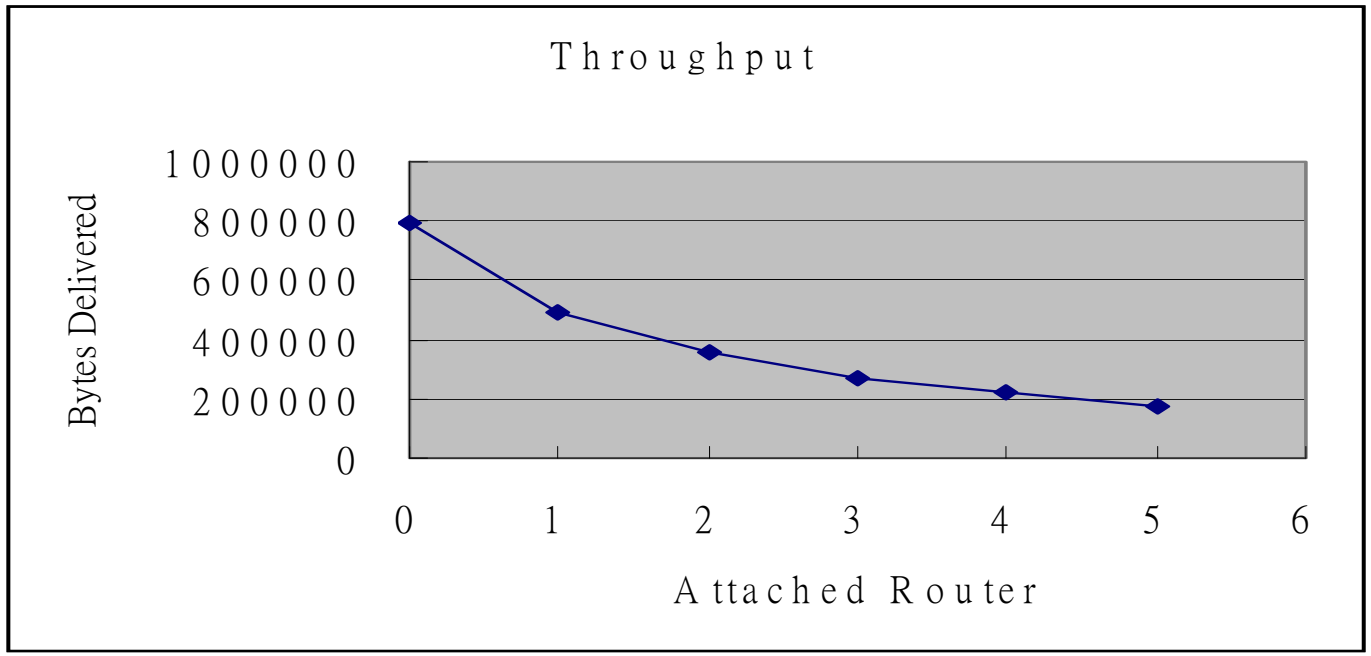
– For UDP traffics, more delay and packet loss as hop count get increased.





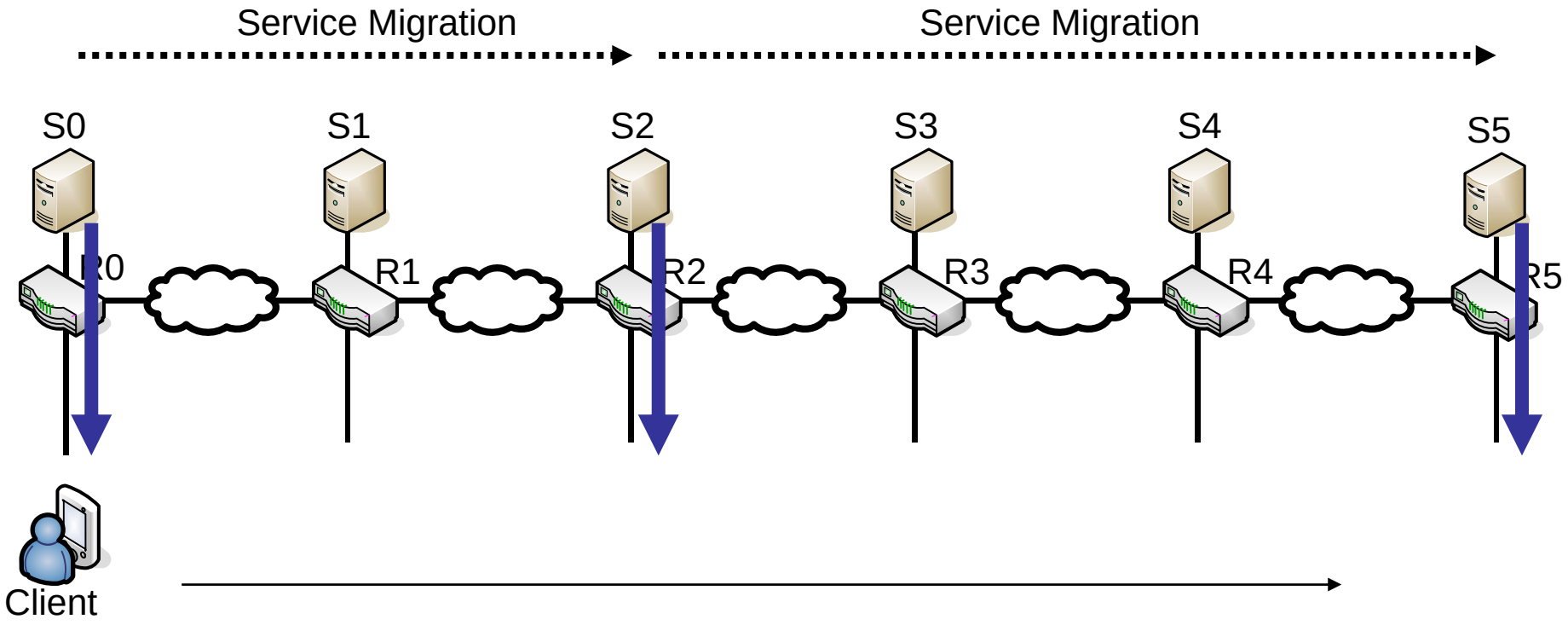
Service Migration (cont)

– For TCP traffics, available bandwidth decreases as hop count get increased.



Service Migration (cont)

- With service migration

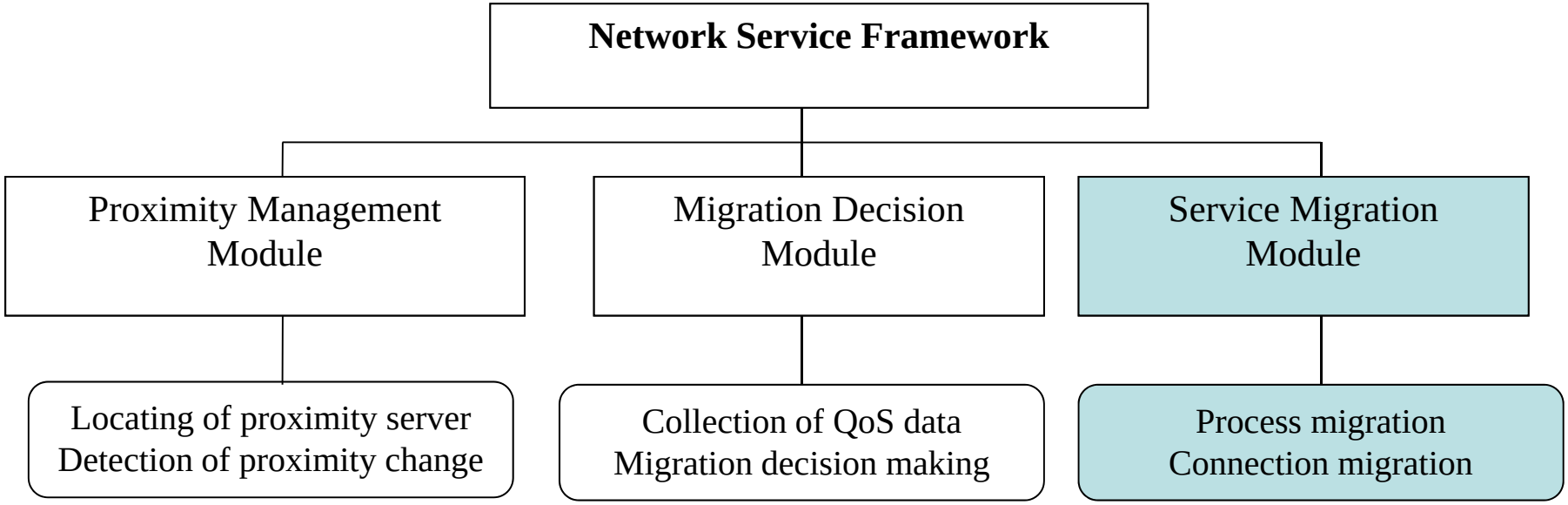


Service Migration (cont)

- Service migration
  - Migrate an ongoing service to new nearest server on topology change or node mobility.
  - Individual user can receive better service.
  - Global network may have better utilization.

A New Service Framework Featuring Service Migration

- System Architecture



A New Service Framework Featuring Service Migration (cont)

- Proximity Management Module (PMM)
  - Detection of proximity change
    - Change in Round-Trip Time (RTT), Hop count / Time To Live (TTL)
    - Mobile IP stack
  - Locating of proximity server
    - Anycast
    - Portal node

A New Service Framework Featuring Service Migration (cont)

- Migration Decision Module (MDM)
  - Responsible for the collecting and monitoring of service quality, and decision-making on service migration
  - Weighted network distance
    - Bandwidth, delay, server capacity, service type, service remaining time, connection fee, …
    - Associate each of above factors with distinct weighting coefficients.

A New Service Framework Featuring Service Migration (cont)

- Service Migration Module (SMM)
  - Performing the actual process and connection migration.
  - Specialized, dedicated, light-weighted service migration scheme to replace those general-purpose, heavy-weighted process migration techniques.

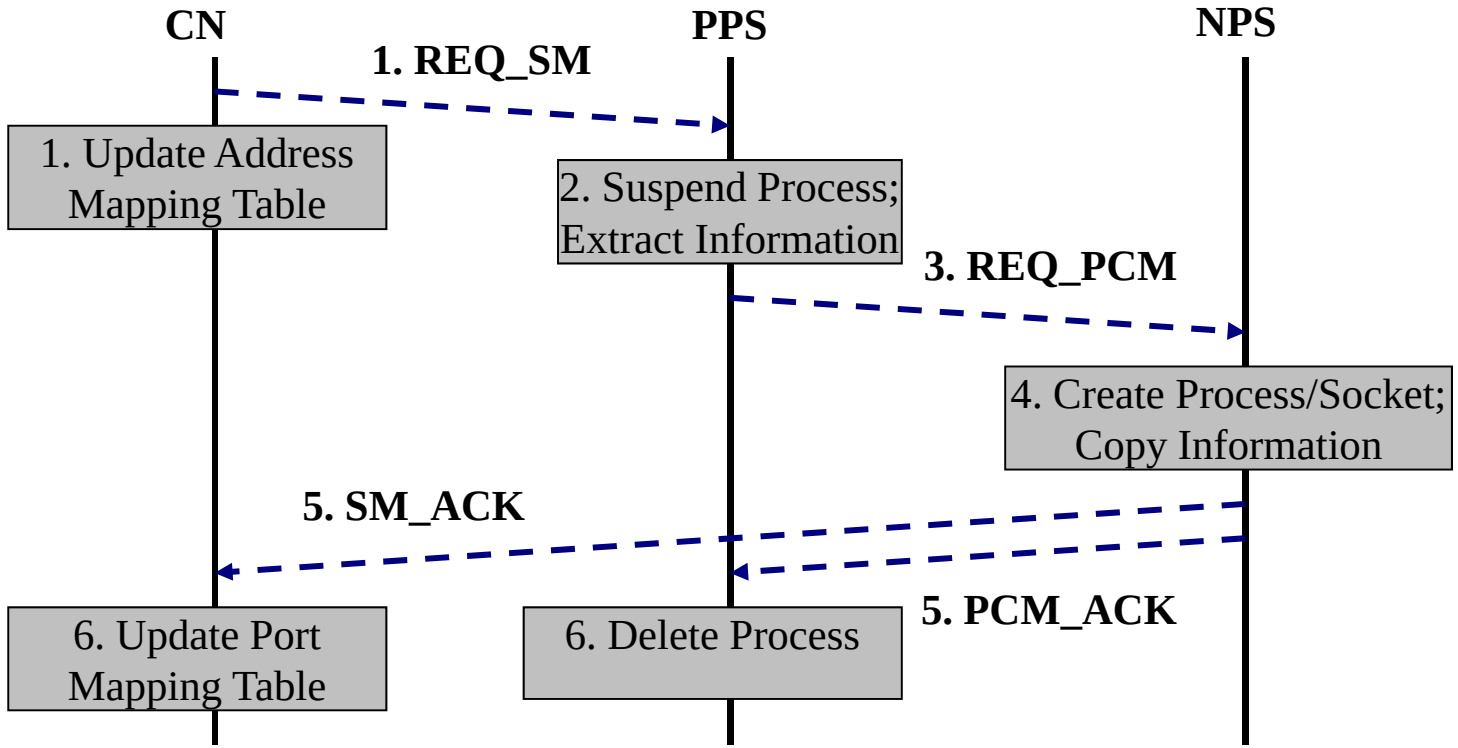
A New Service Framework Featuring Service Migration (cont)

- Connection migration at client-end
  - Packets from NPS are illegal to CN, since their IP address-port number pairs are incorrect.
  - IP address-port number mapping
    - Done at server-end with egress router support
    - Done at client-end with Network Address Translation (NAT)

Light-Weighted Service Migration for TCP-based Services

- Operation

CN: Client Node PPS: Previous Proximity Server NPS: New Proximity Serve
REQ_SM: Request for Service Migration REQ_PCM: Request for Process/Connection Migration
SM_ACK: Acknowledge for Service Migration PCM_ACK: Acknowledge for Service Migration



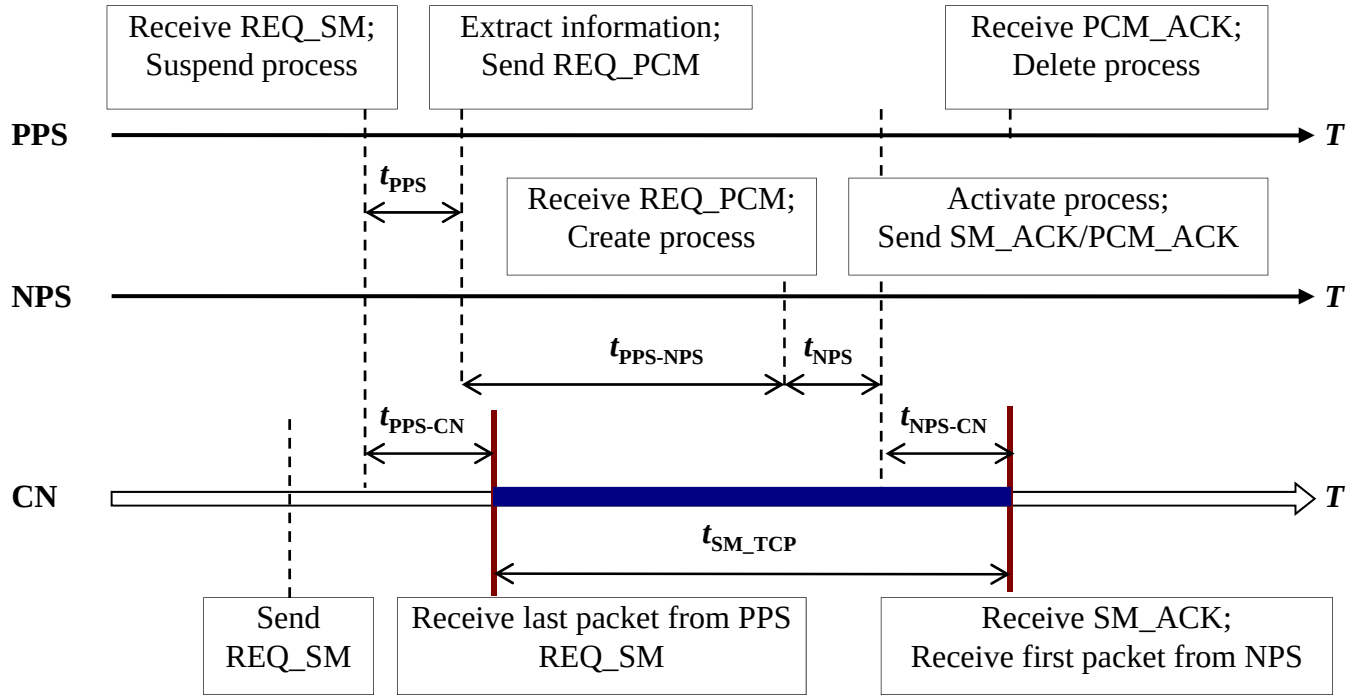
Light-Weighted Service Migration for TCP-based Services (cont)

- Let
  - $t_{PPS\text{-}CN}$: The link delay from PPS to CN
  - $t_{NPS\text{-}CN}$: The link delay from NPS to CN
  - t_{PPS}: The processing time at PPS for process suspension and information extraction.
  - t_{NPS}: The processing time at NPS for process creation and information copy.
  - $t_{PPS\text{-}NPS}$: The transfer time for socket information from PPS to NPS.

Light-Weighted Service Migration for TCP-based Services (cont)

- Performance analysis

CN: Client Node PPS: Previous Proximity Server NPS: New Proximity Serve
REQ_SM: Request for Service Migration REQ_PCM: Request for Process/Connection Migration
SM_ACK: Acknowledge for Service Migration PCM_ACK: Acknowledge for Service Migration



Light-Weighted Service Migration for TCP-based Services (cont)

- The service disruption period experienced by the CN, t_{SM_TCP}

$$t_{SM\_TCP} = \max(0, t_{PPS} + t_{PPS\text{-}NPS} + t_{NPS} + t_{NPS\text{-}CN} - t_{PPS\text{-}CN})$$

- MDM will suggest migration if $t_{NPS\text{-}CN} < t_{PPS\text{-}CN}$.
- Zero service disruption period is possible if $t_{NPS\text{-}CN} << t_{PPS\text{-}CN}$, and PPS and NPS are powerful enough and connected by high capacity backbone.

Light-Weighted Service Migration for TCP-based Services (cont)

- Assume B_{PPS} / B_{NPS} is the available bandwidth from PPS to CN / from NPS to CN, and L is the amount of data remained to be delivered.

- Remaining service time without migration,

$$T = L / B_{PPS}$$

- Remaining service time with migration,

$$T_{SM\_TCP} = L / B_{NPS} + t_{SM\_TCP}$$

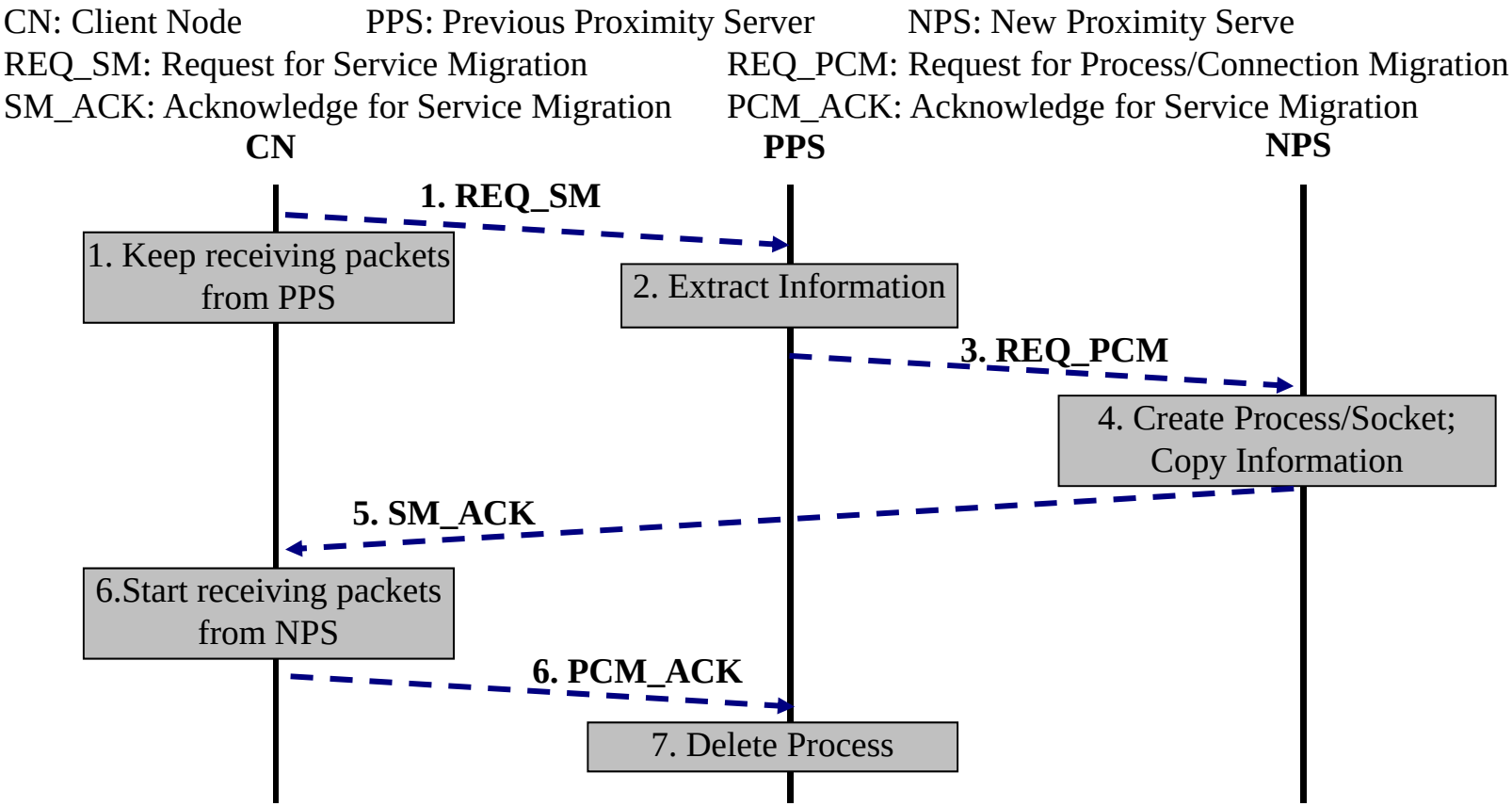
Light-Weighted Service Migration for TCP-based Services (cont)

$$T - T_{\text{SM\_TCP}} = L\left(\frac{1}{B_{\text{PPS}}} - \frac{1}{B_{\text{NPS}}}\right) - t_{\text{SM\_TCP}}$$

implies that a service migration is profitable if the remaining data and the bandwidth difference are large enough to cover the cost of service migration.

Seamless Service Migration for UDP-based Services

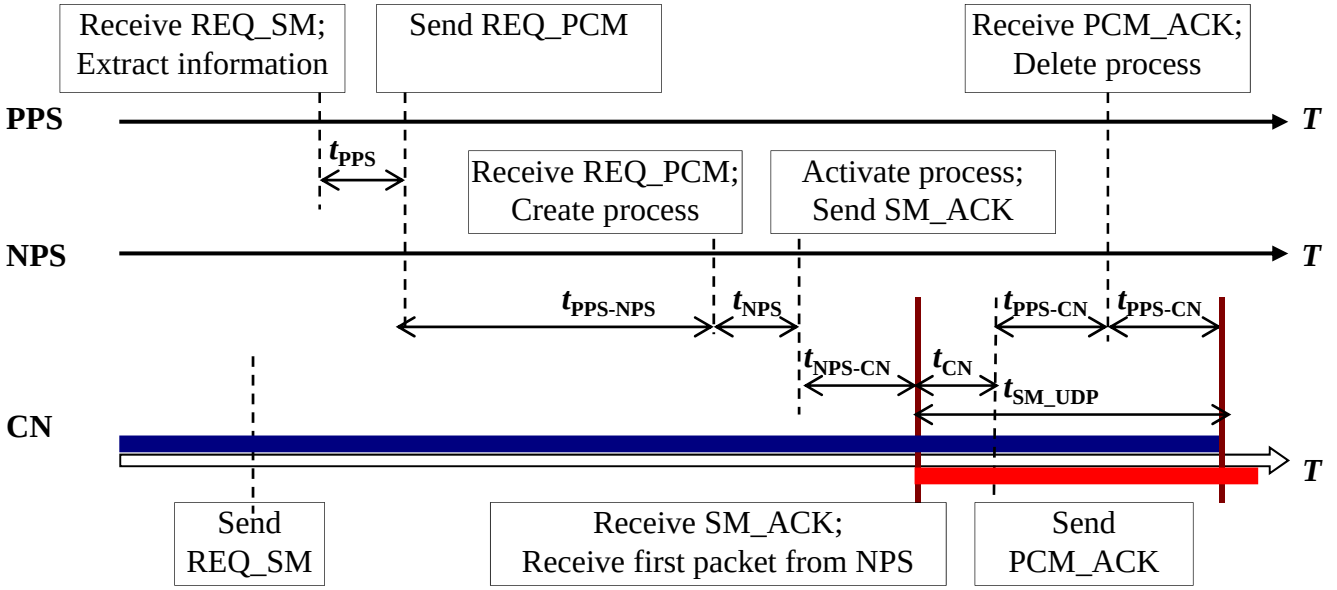
- Operation



Seamless Service Migration for UDP-based Services (cont)

- Performance analysis

CN: Client Node PPS: Previous Proximity Server NPS: New Proximity Serve
REQ_SM: Request for Service Migration REQ_PCM: Request for Process/Connection Migration
SM_ACK: Acknowledge for Service Migration PCM_ACK: Acknowledge for Service Migration
: Receiving packets from PPS : Receiving packets from NPS



Seamless Service Migration for UDP-based Services (cont)

- Let
  - $t_{\text{PPS-CN}}$: The link delay from PPS to CN or from CN to PPS.
  - t_{CN}: The processing time at CN for the buffering of packet from NPS.
- Period of duplicated packet streams

$$t_{\text{SM\_UDP}} = t_{\text{CN}} + t_{\text{PPS-CN}} + t_{\text{PPS-CN}}$$

- The term t_{CN} is to count for the buffering time required at CN for multimedia streaming applications.

Simulation and Implementation

- Implementation of Lighted-weighted Service Migration for TCP-based Services
- Implementation of Seamless Service Migration for UDP-based Services

Implementation of Lighted-weighted Service Migration for TCP-based Services

- TCP is stateful.
- Connection status must be retained.
- Implementation configuration
  - Server end: CentOS 4.5 (Linux Kernel 2.6.11), TCP Connection Passing v11, and gcc 3.4.6
  - Client end: Microsoft Windows XP2, WinpkFilter v3.0 and Borland C++ Builder 6.0

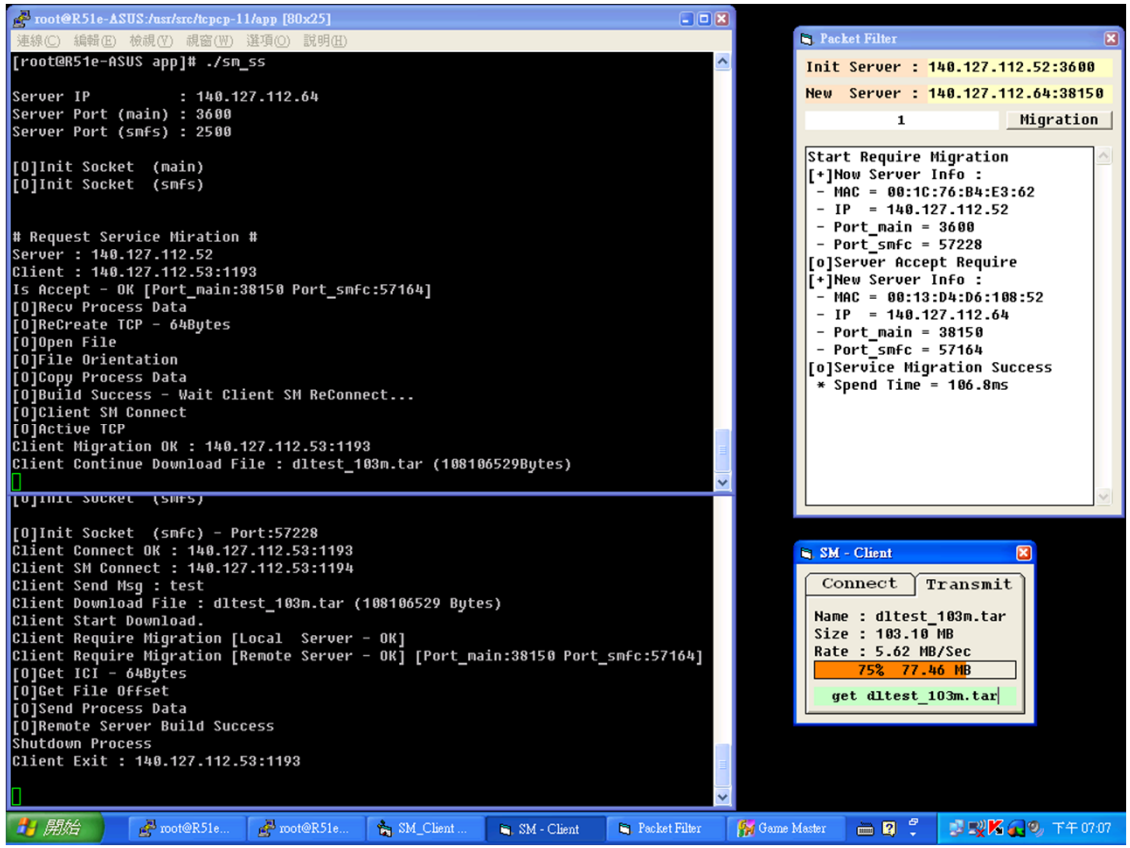
Implementation of Lighted-weighted Service Migration for TCP-based Services (cont)

- TCP Connection Passing (TCPCP) v11
  - A kernel patch designed for load-balancing and failover applications
  - APIs for
    - Extraction of essential information regarding an ongoing TCP connect, called Internal Connection Information
    - Copying ICI to a newly created socket on a new host

Implementation of Lighted-weighted Service Migration for TCP-based Services (cont)

- WinpkFilter v3.0
  - A toolkit by NT Kernel Resources
  - After installation, it resides between the network adaptor and the TCP/IP stack
  - APIs in its run-time library can be used to
    - Intercept, modify, filter, or generate raw packets passing through it.
  - For IP address – port number mapping at client-end

Implementation of Lighted-weighted Service Migration for TCP-based Services (cont)

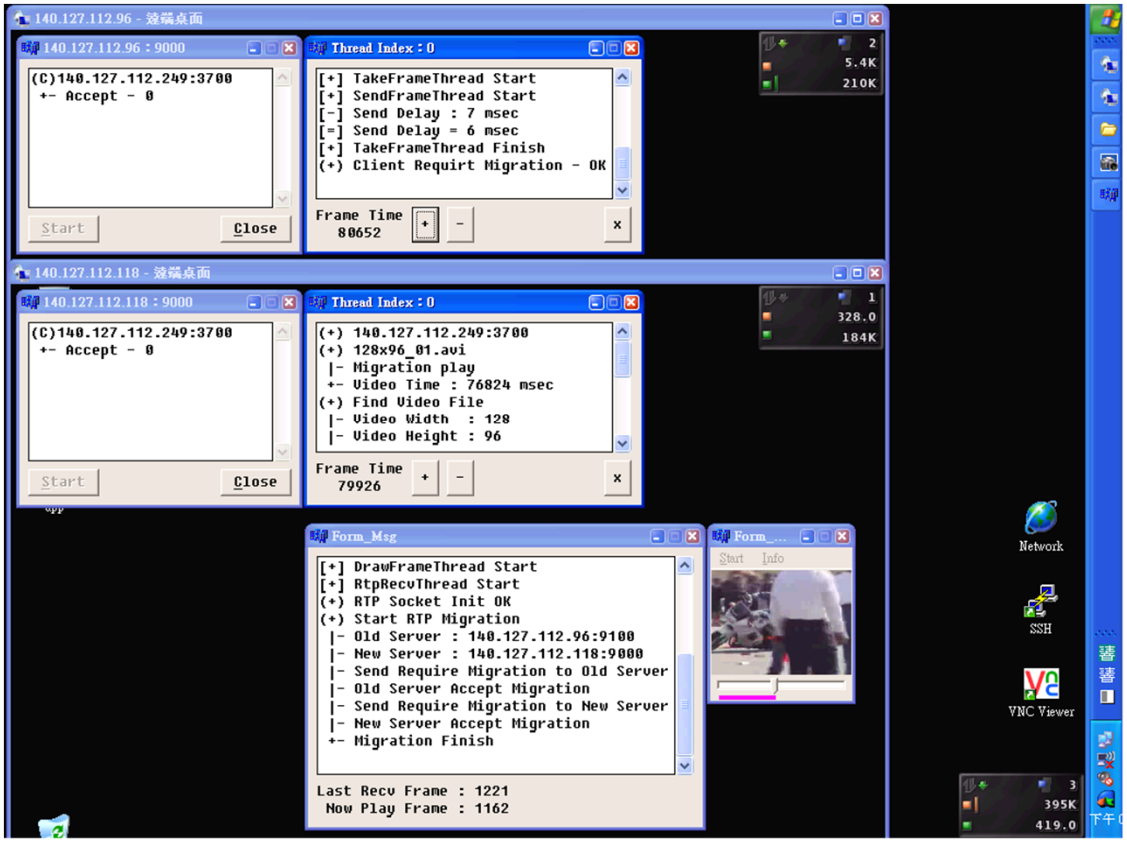


- A file transfer service in progress can be successfully migrated from one server to another.

Implementation of Seamless Service Migration for UDP-based Services

- UDP is stateless.
- Seamless service migration is possible.
  - Tear-down stream to PPS after stream to NPS is established.
- Implementation configuration
  - Microsoft Windows XP SP2
  - Borland C++ Builder 6 Fix 4
  - Winsock API and Indy UDP component
  - Real-time Transport Protocol (RTP)

Implementation of Seamless Service Migration for UDP-based Services (cont)

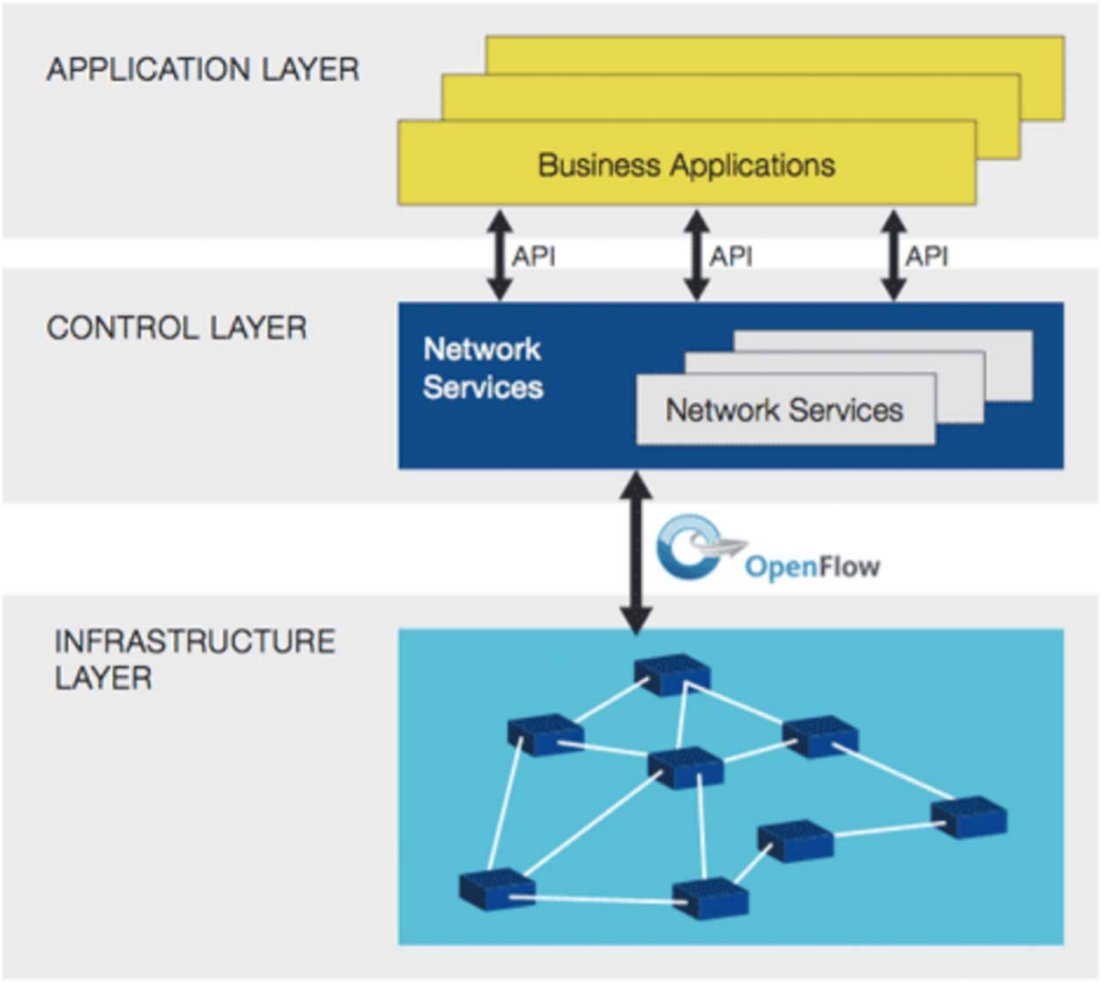


- The client of an ongoing video stream can seamlessly switch its sending source from one server to another.

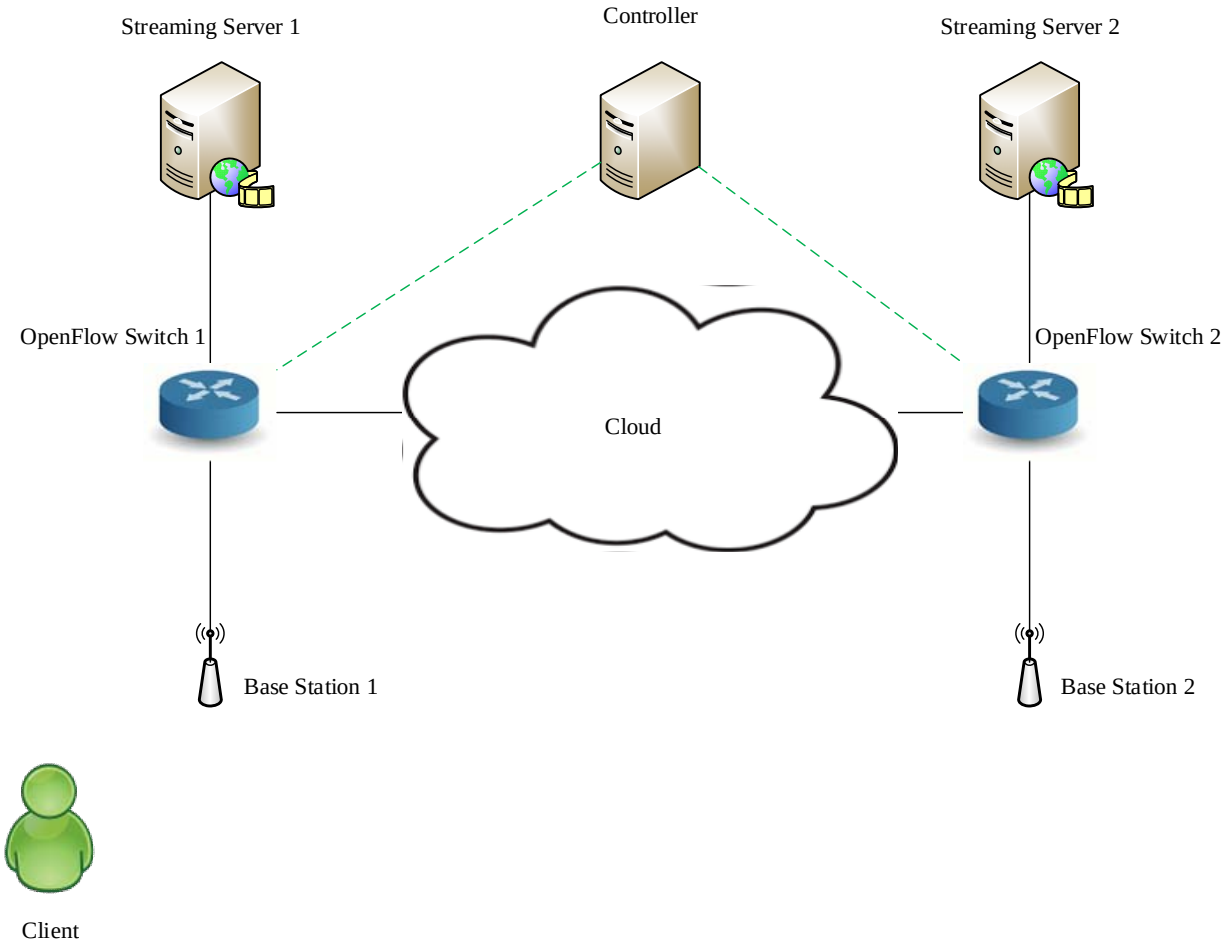
SDN-based Service Migration

- Software-Defined Network (SDN)
- Separation of control and data plans
- Programmable controller controls the routing/forwarding of switches
- Dynamic configuration of network topology

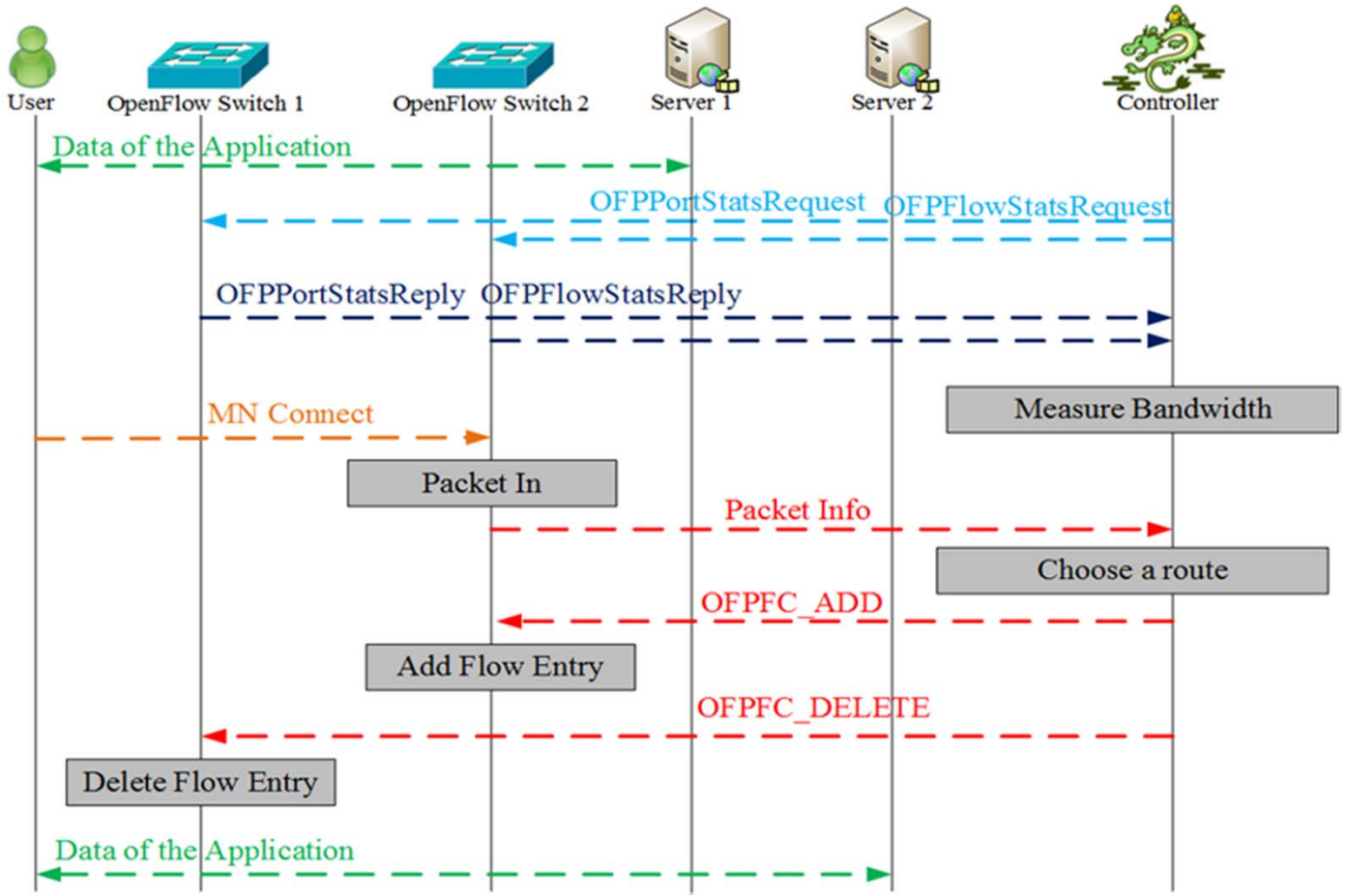
Software-Defined Network



Scenario


Streaming Server 1
Controller
Streaming Server 2
OpenFlow Switch 1
OpenFlow Switch 2
Cloud
Base Station 1
Base Station 2
Client

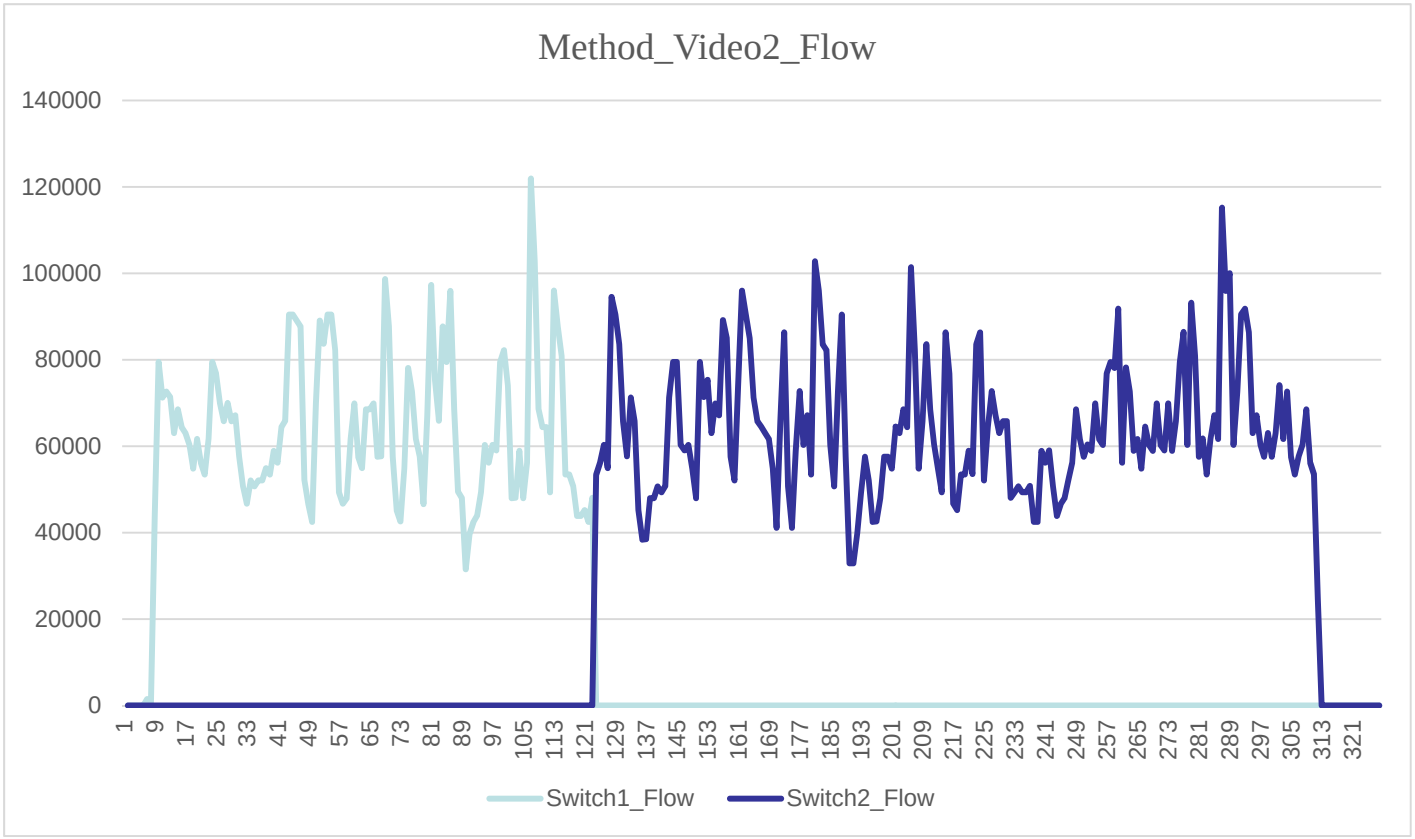
Signaling Diagram



System Configuration

項目	作業系統	軟/硬體名稱	數量
OpenFlow 控制器	Ubuntu 14.04	Ryu 3.19	1
OpenFlow 交換器	Ubuntu 14.04	OpenvSwitch 2.0.0	2
802.11 設備	Linux	D-link 300	2
串流伺服器	Ubuntu 14.04	VLC	2
可連網行動設備	Windows 7	Asus x450v	1
測試影片		Avi	2

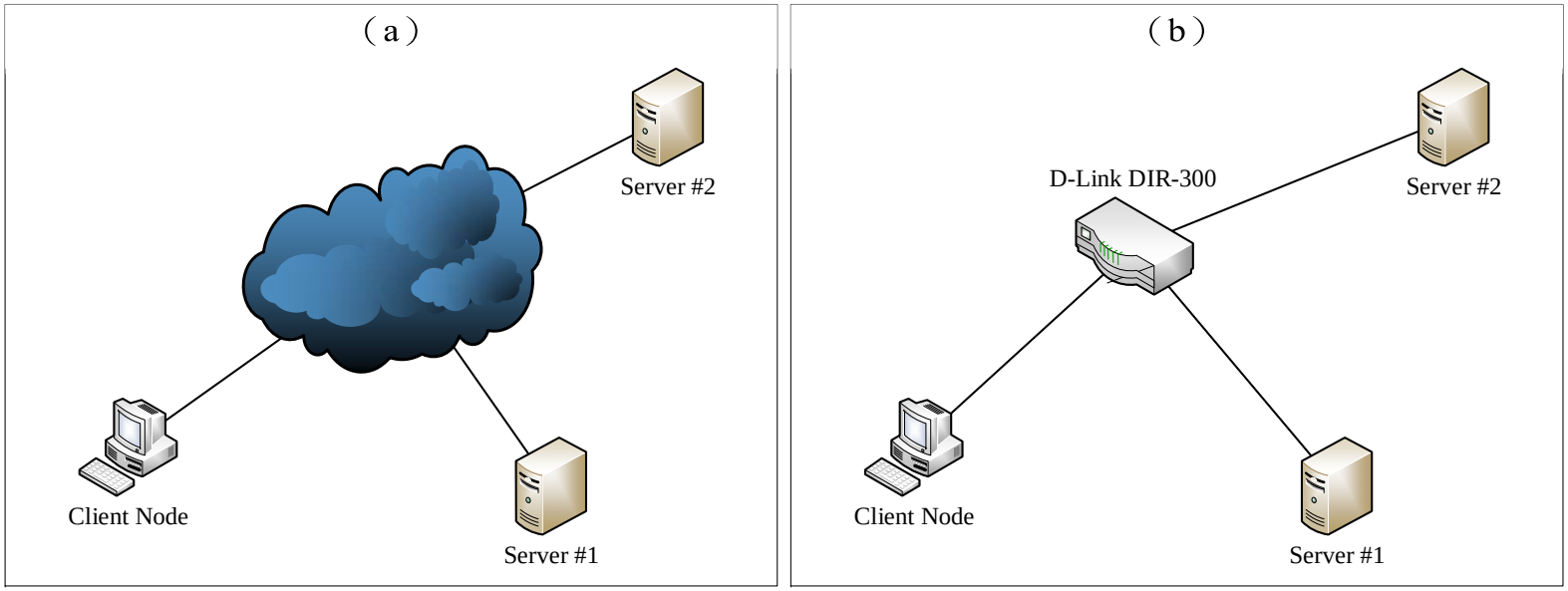
Service Migration



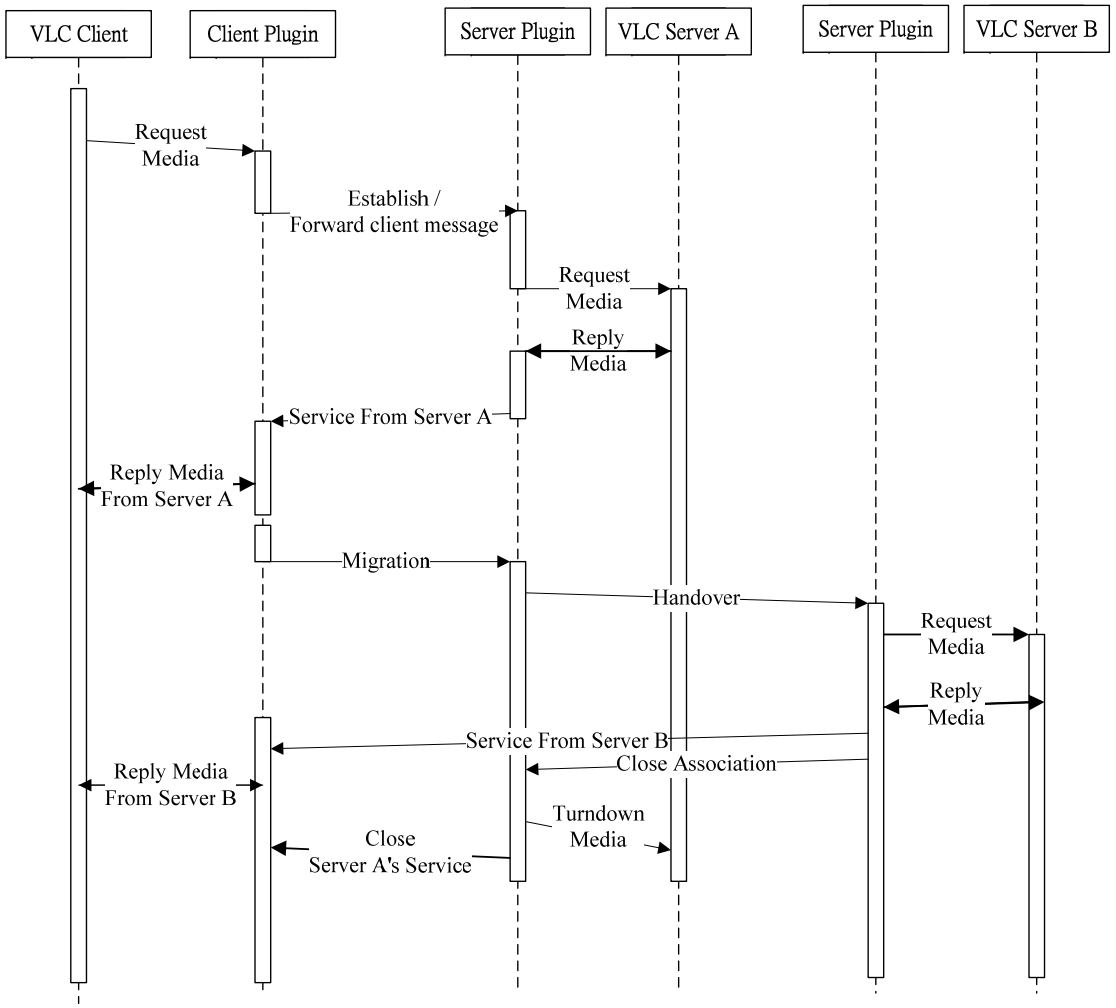
SCTP-based Service Migration

- Stream Control Transmission Protocol
- SCTP’s multi-streaming for seamless VoD service migration

Scenario


(a)
Server #2
Client Node
Server #1
(b)
D-Link DIR-300
Server #2
Client Node
Server #1

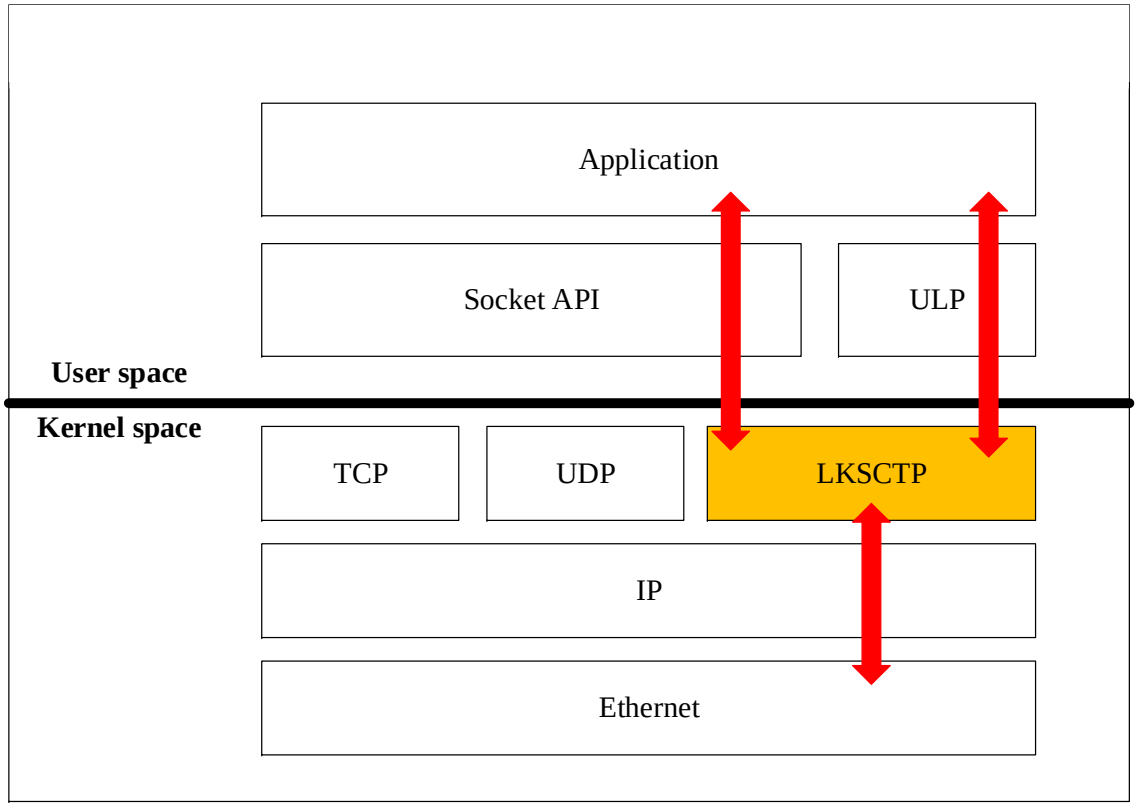
Signaling Diagram



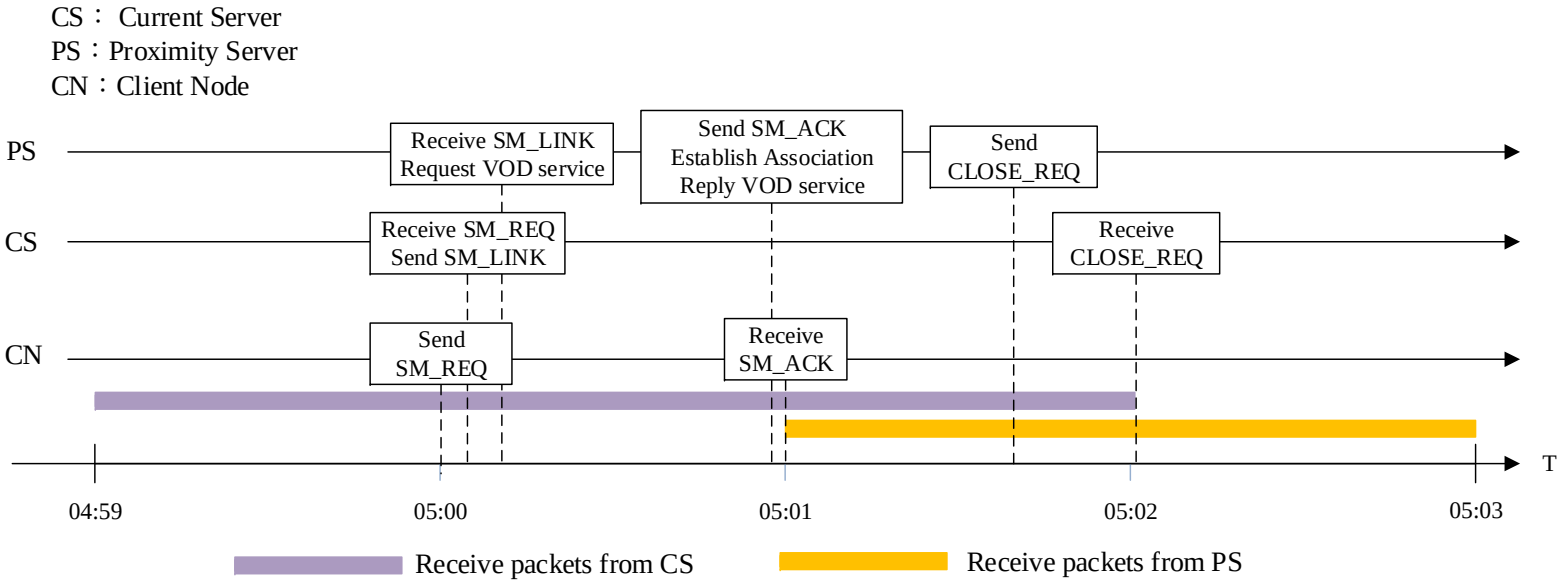
System Configuration

		Client Node	Server #1	Server #2
System	Operating System	Ubuntu 14.04.1	Ubuntu 14.04.1	Ubuntu 14.04.1
	Kernel version	3.16.0-30-generic	3.16.0-30-generic	3.16.0-30-generic
Hardware	CPU	Intel Core i3-5010U	Intel Core i3-4010U	Intel Core i3-5010U
	RAM	4G	8G	4G
	Hard Disk	ATA 128G	ATA 128G Kingston 240G	ATA 128G
	Ethernet interface	Realtek 1Gbps	Realtek 1Gbps	Realtek 1Gbps

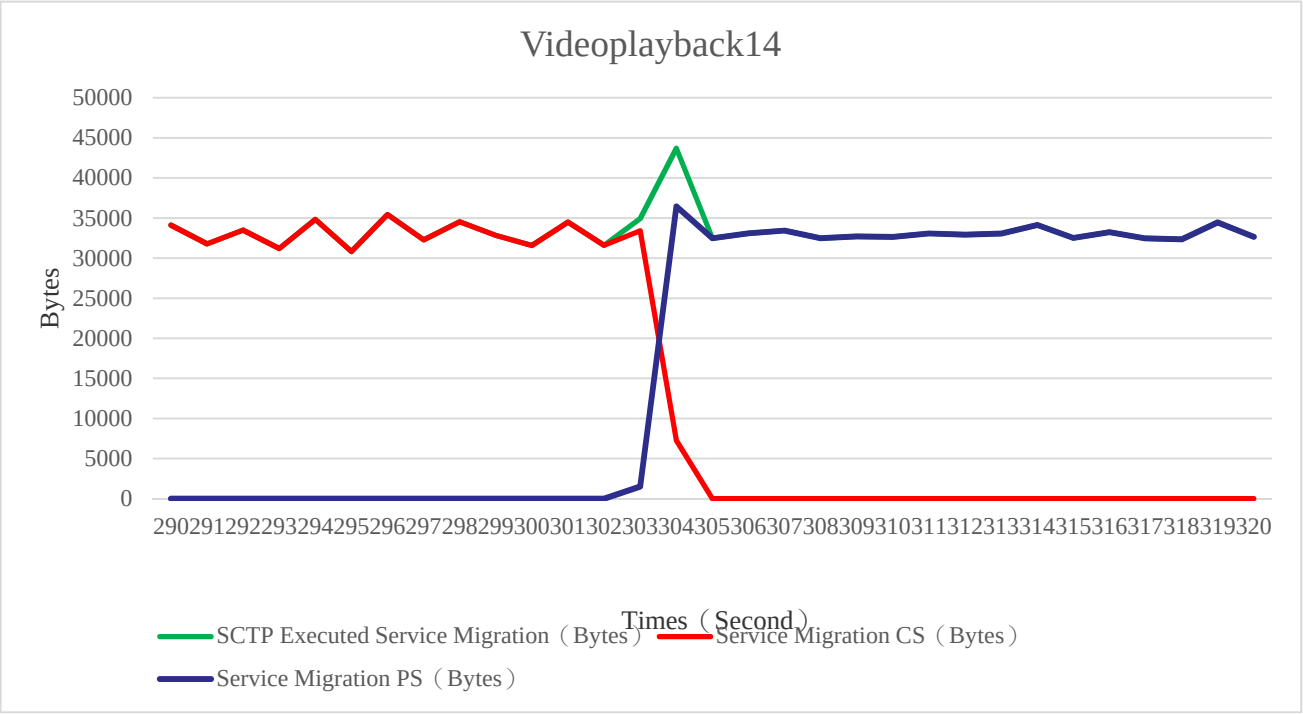
SCTP Module


Application
Socket API
ULP
User space
Kernel space
TCP
UDP
LKSCTP
IP
Ethernet

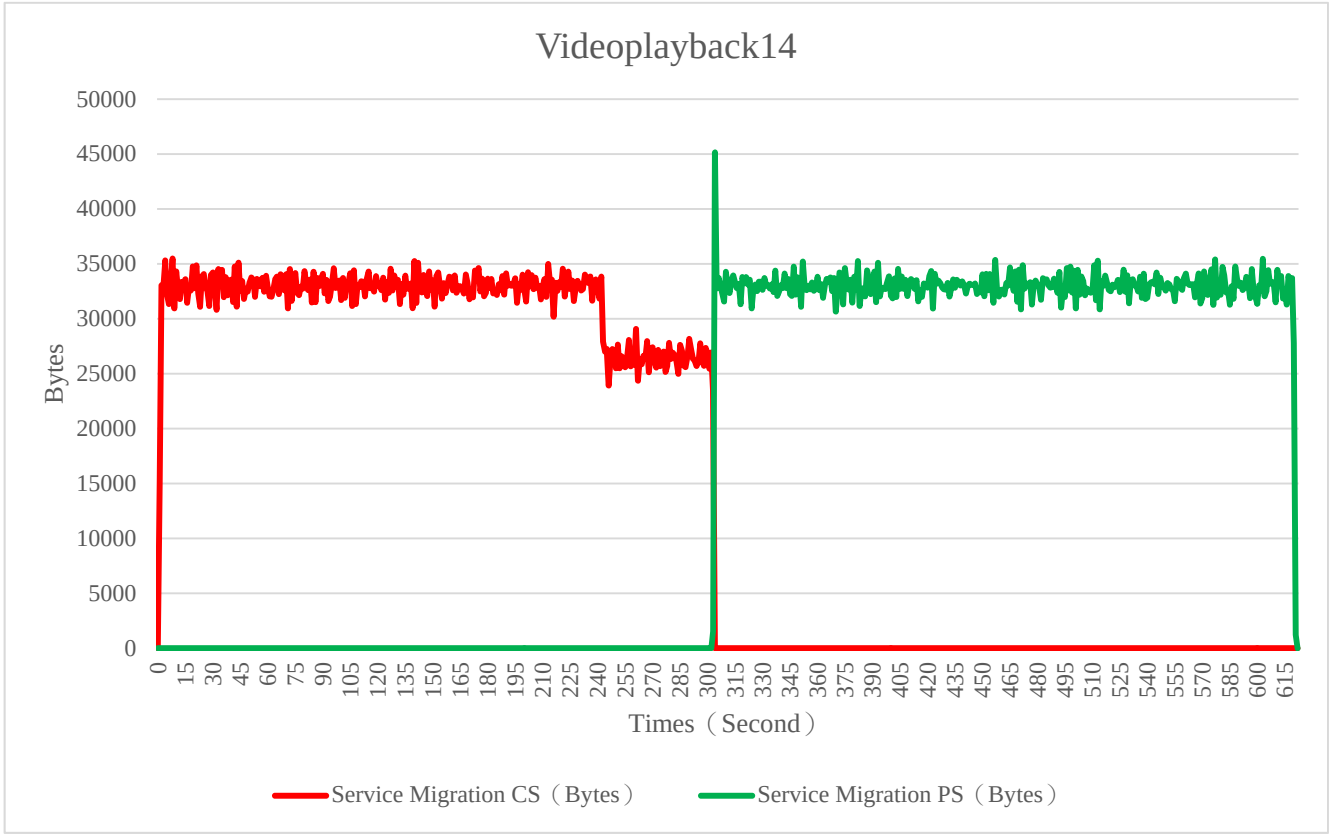
Timing Chart



Service Migration



Service Migration



Conclusions

- Conclusions
  - Service migration at server end for improved QoS and better network utilization
    - Migrates an ongoing service to its new nearest server
    - Individual connection receives improved service quality
    - Infrastructure has better utilization
    - Benefit of service migration is more eminent in case of vertical handover involving different access technologies

Thank you for your listening!

Comments and suggestions are sincerely welcomed!