Deployability of Content-Centric Networking

Group 3 :

Mani Prashanth Varma Manthena (4243846)
Michal Golinski (4253590)

Dr. ir. Fernando Kuipers
Niels van Adrichem
(Network Architectures and Services Group)

Introduction

• TCP/IP (host based communication model) vs. CCN (content-centric networking)

• Problem description :

➢ **Deployability** considerations of CCN like content to cache, cache size, caching under **changing link conditions**, etc.

➢ To **compare CCN** performance **with** present **TCP/IP** architecture under various simulated real world scenarios.

Outline

- Related work

- The Project CCNx

- Measurement scenario

- Measurement results

- Conclusion and future work

Related work

• Cache size implementation and memory management of CCN based on various aspects like memory blocks price, efficiency of memory storage and buffering, amount of traffic in the network, etc.

• Performance comparison of CCN with TCP/IP e.g. **data transfer efficiency** comparison of CCN, HTTP and HTTPS – *Van Jacobson (2009).*

• Performance comparison of CCN with TCP/IP for various **real time services** and applications like live streaming, audio and video distribution, etc.

The Project CCNx

- Introduction

- Comparison of IP and CCN protocol stack

figure: Van Jacobson (2009)
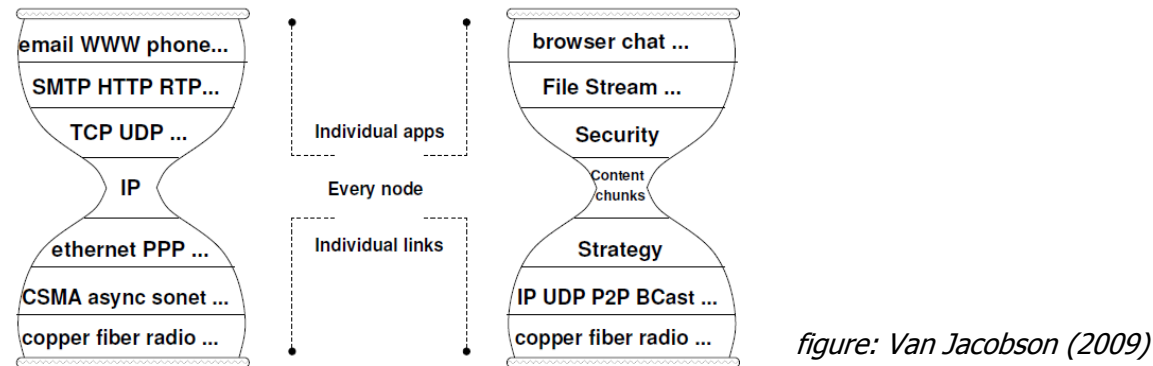
- CCN packet types – interest and data

- CCN content naming – hierarchical structure

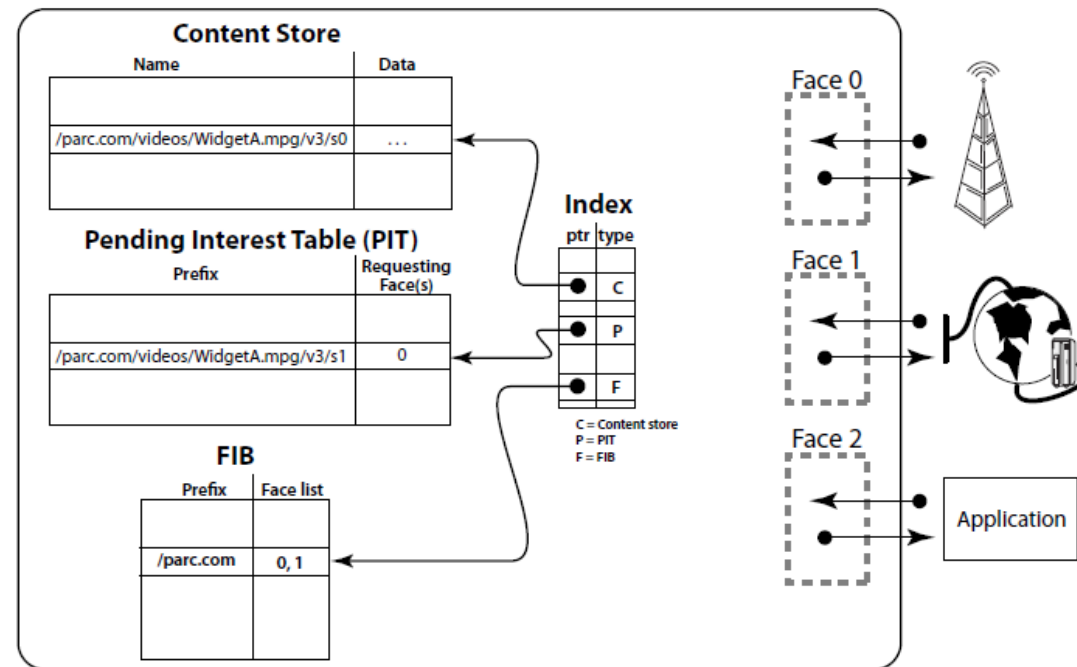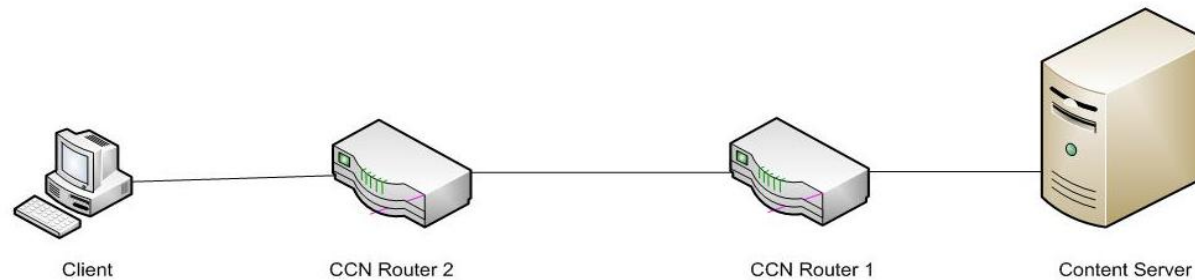- CCN forwarding engine model
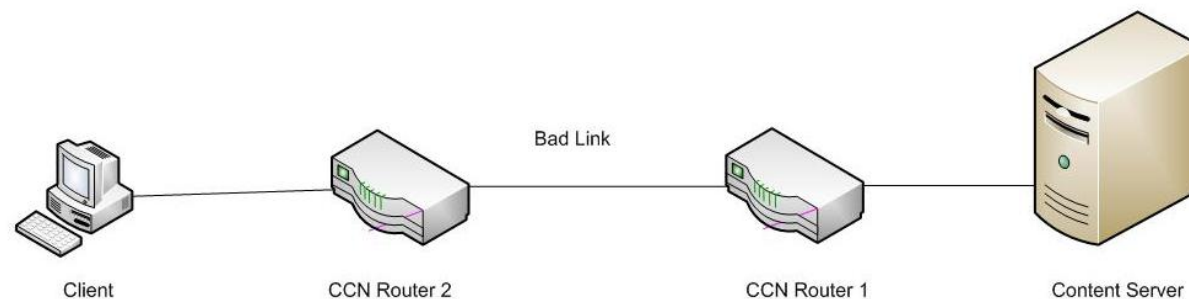

figure: Van Jacobson (2009)

- Project CCNx allows users to change several CCN protocol parameters like cache size, MTU size, etc. in the protocol open source configure file.
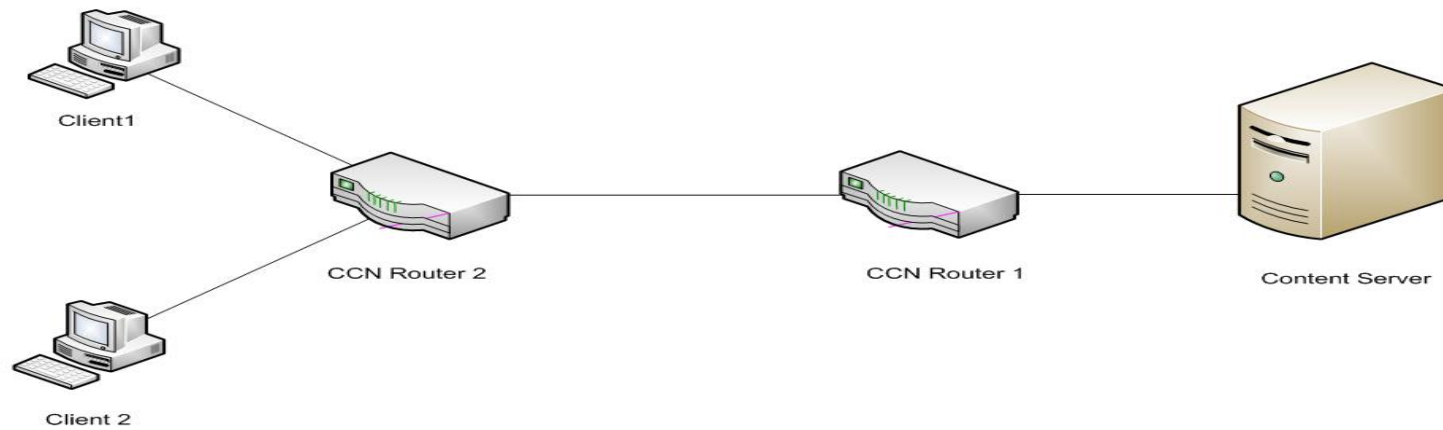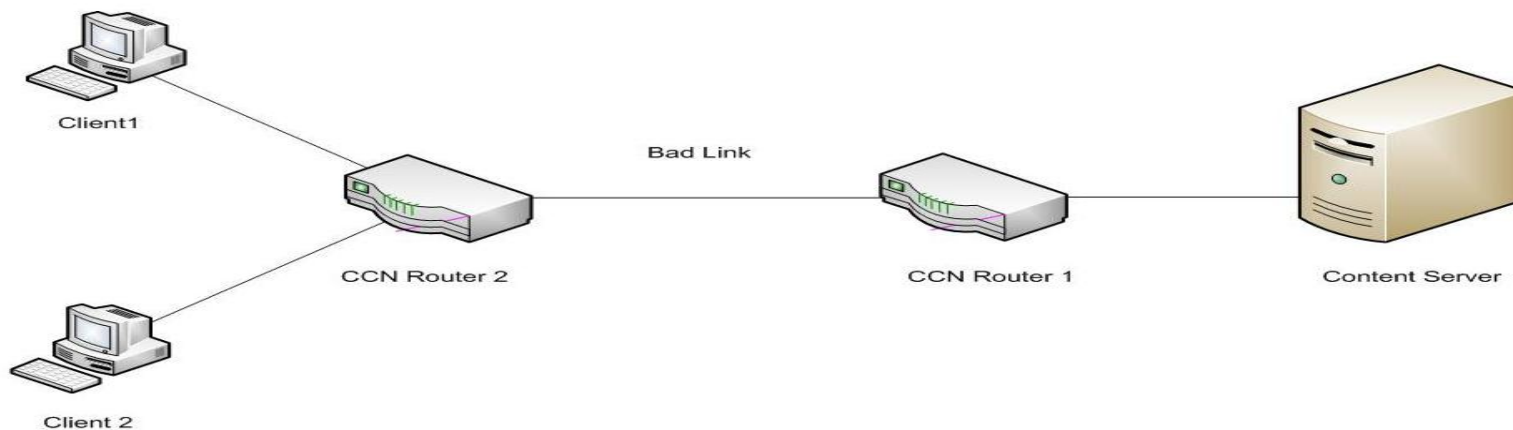
Measurement scenario

- Measurement setup – A

- Measurement setup - B

- Measurement setup – C


- Measurement setup - D

• Problems encountered – delay measurements, hardware configuration, MTU not responding

• Expected results

➢ Larger cache sizes would decrease overall delay, jitter, etc. in measurement setups C and D.

➢ CCN would out perform HTTP in measurement setups C and D and would approximately match the performance of HTTP in other setups and scenarios.

➢ MTU size could fix the problem of decrease of bandwidth.

➢ CCN could handle packet losses like TCP/IP.

Measurement results

- Influence of cache size on delay

➢ Mean packet delays
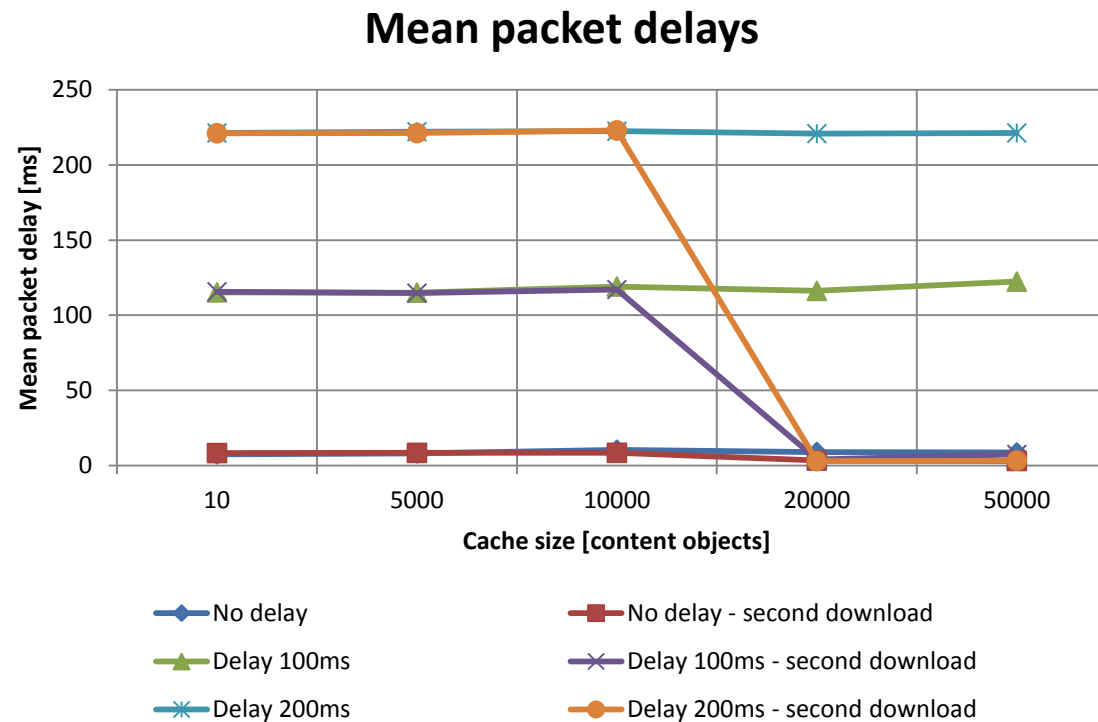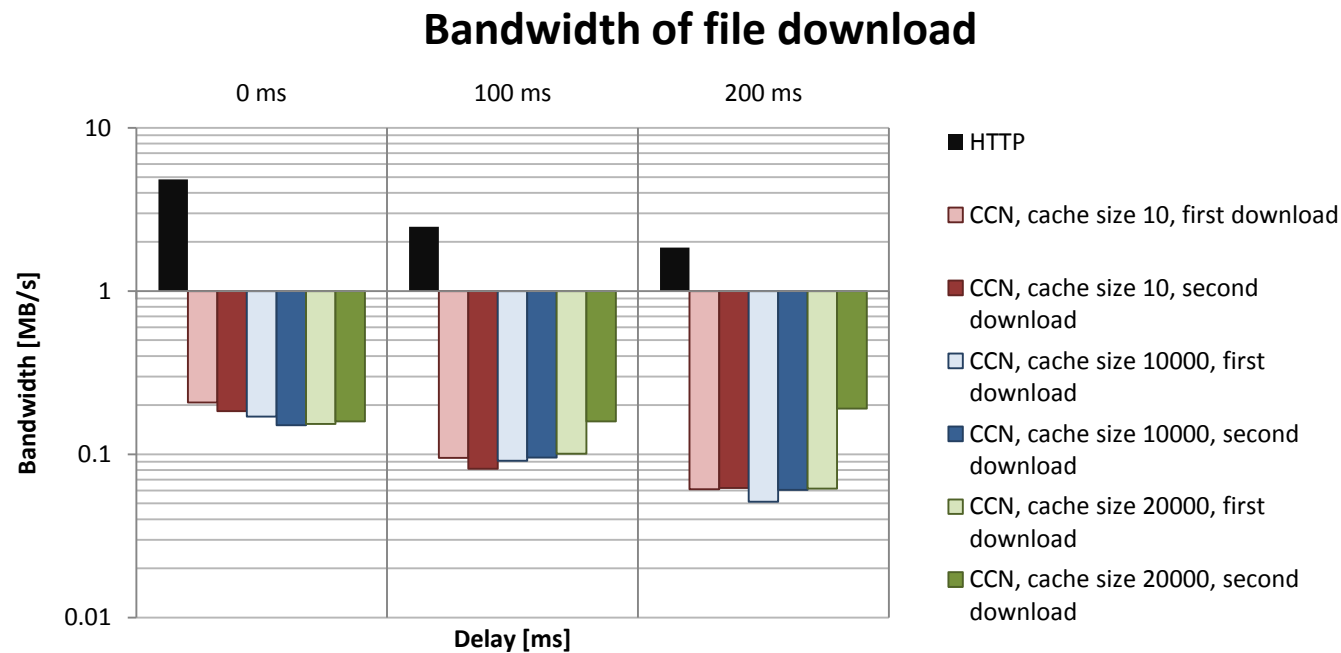

Mean packet delays

➢ Standard deviation of packet delays (jitter)

Standard deviation of packet delays
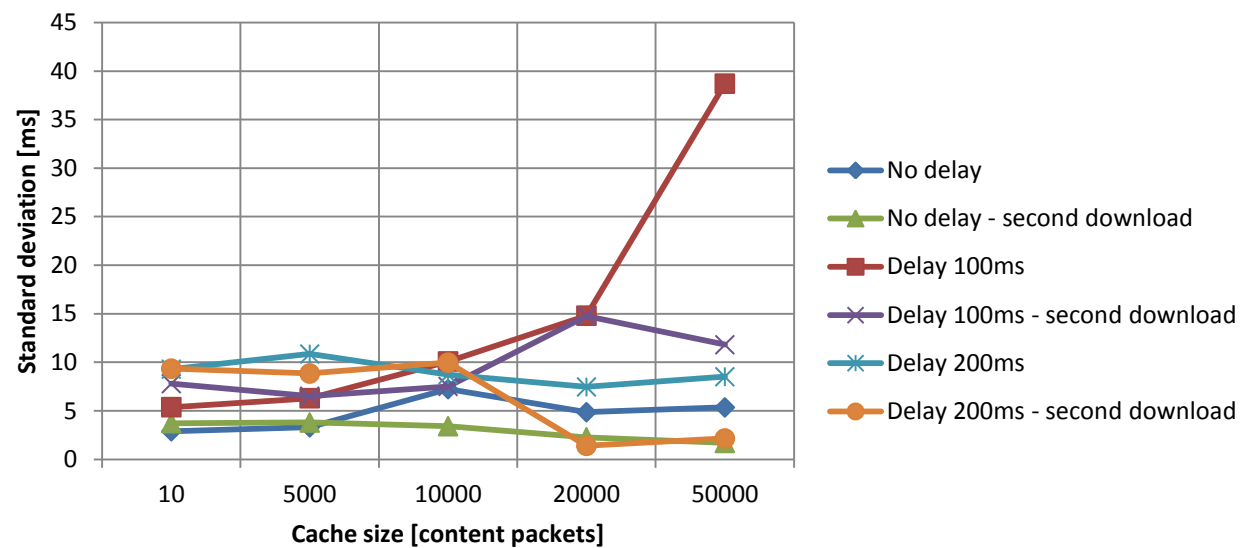
- Influence of delay on a bandwidth

➢ Bandwidth of file download (logarithmic scale)

Bandwidth of file download

➢ # Relative bandwidth loss

Relative bandwidth loss

Legend:
- HTTP
- CCN, cache size 10, first download
- CCN, cache size 5000, first download
- CCN, cache size 10000, first download
- CCN, cache size 20000, first download
- CCN, cache size 50000, first download
- CCN, cache size 10, second download
- CCN, cache size 5000, second download
- CCN, cache size 10000, second download
- CCN, cache size 20000, second download
- CCN, cache size 50000, second download

➤ Achieved CCN bandwidths

Achieved CCN bandwidths

➢ MTU (Maximum Transmission Unit) size and download time
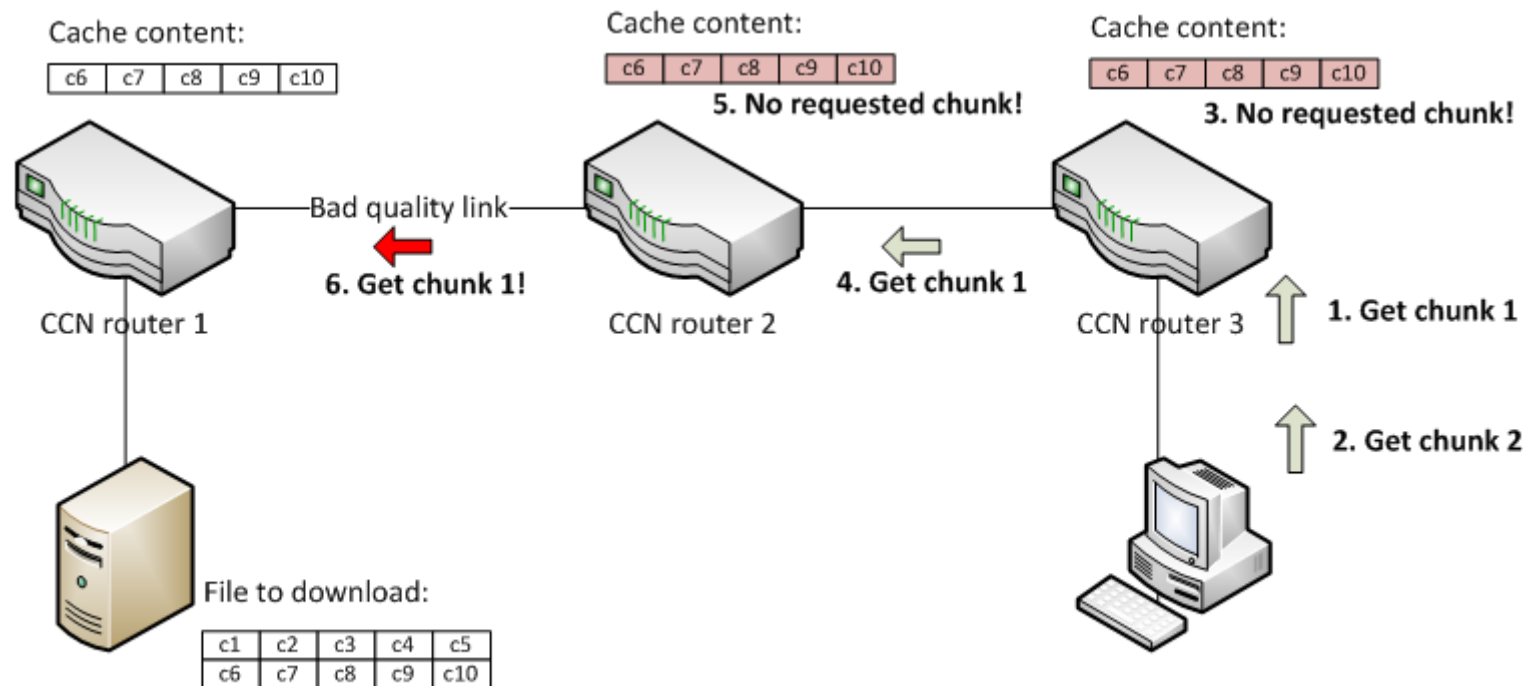
Download time vs MTU
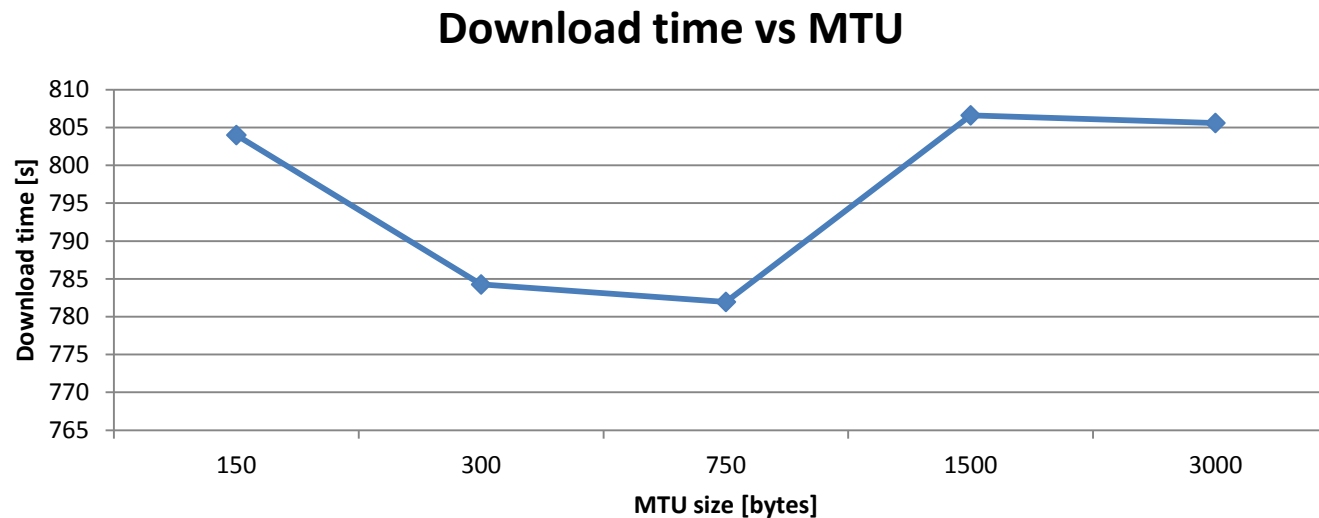
➢ Smart caching strategies
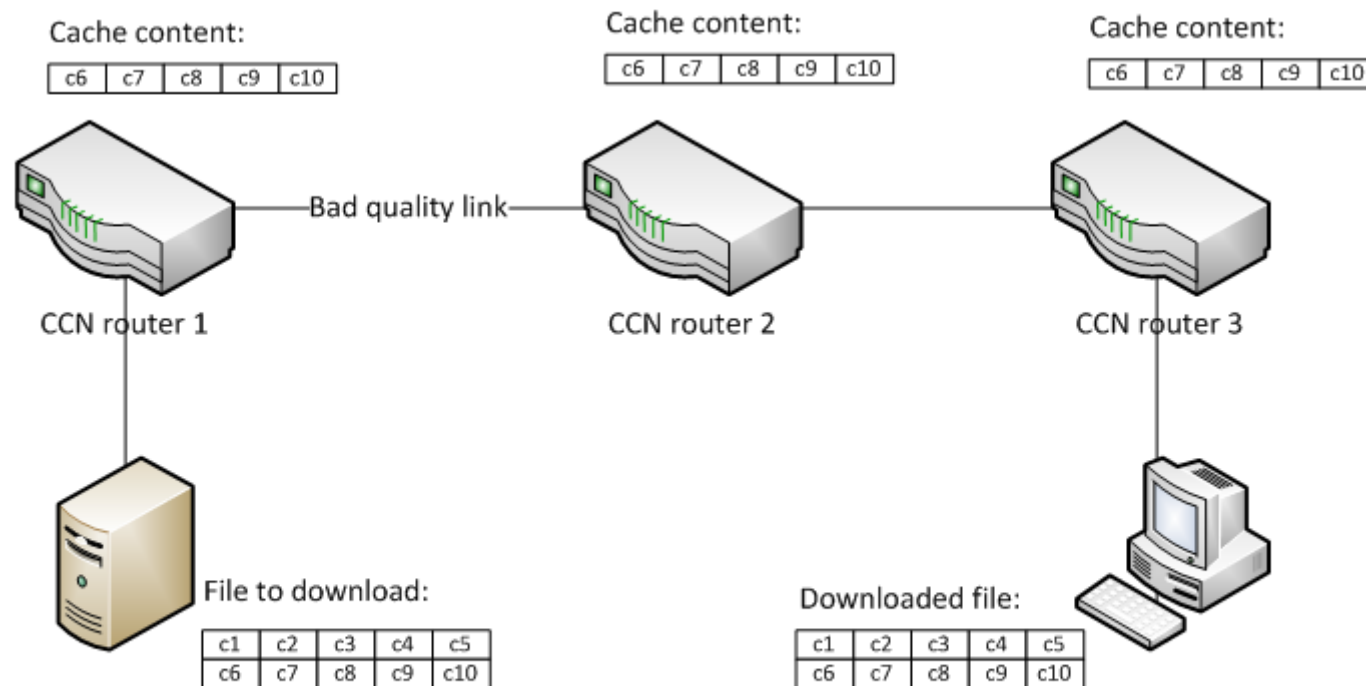
a) State of the network after file download

➤ Smart caching strategies

b) Start of second download – measured case

➢ Smart caching strategies

Cache content:
| c6 | c7 | c8 | c9 | c10 |

Cache content:
| c6 | c7 | c8 | c9 | c10 |

Cache content:
| c6 | c7 | c8 | c9 | c10 |

Some chunks found!

—Bad quality link—

5. Get chunk 1

4. Get chunk 1

1. Get all the chunks stored locally

CCN router 1

CCN router 2

CCN router 3

2. Send chunks 6-10

3. Get remaining chunks starting from lowest one (Get chunk 1)

File to download:
| c1 | c2 | c3 | c4 | c5 |
| c6 | c7 | c8 | c9 | c10 |
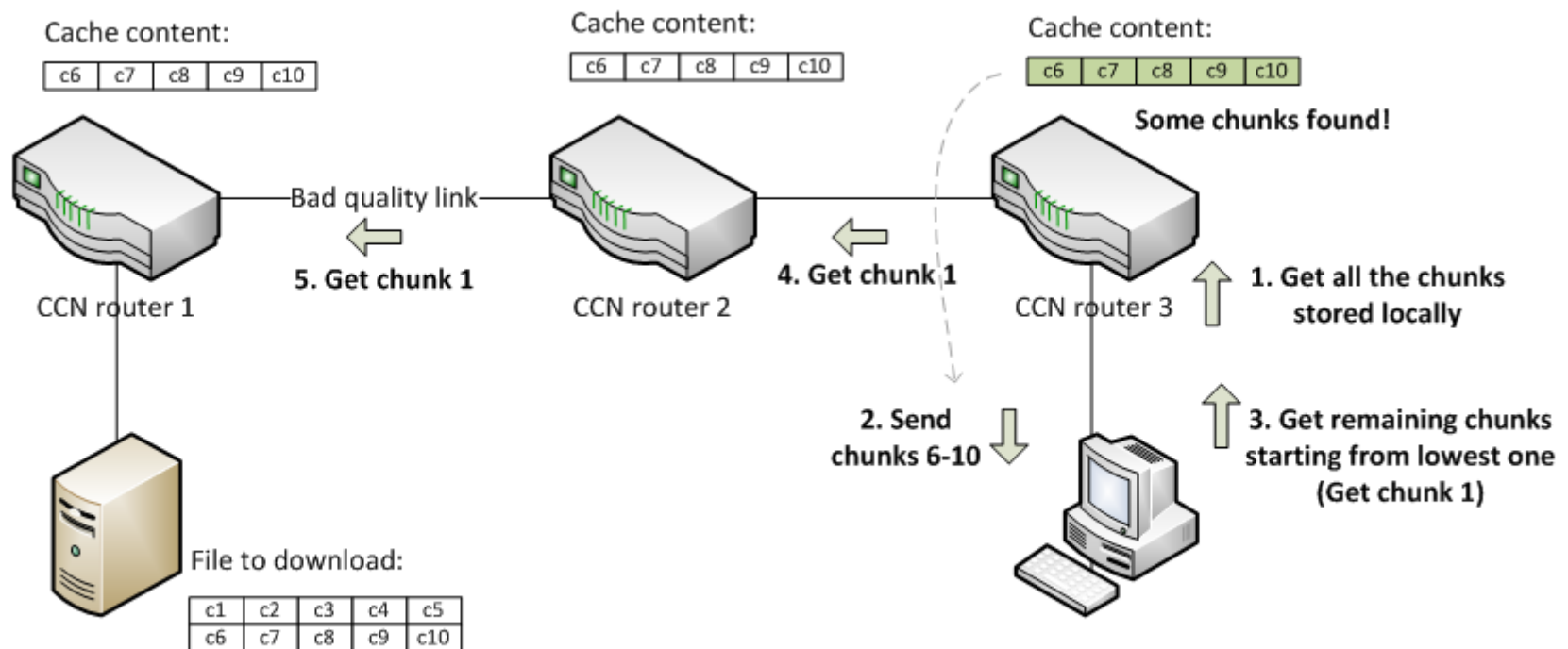
c) Start of second download – expected case

➢ Packet delays in a scenario of smart caching

PDFs of packet delays for a scenario with bad quality links

Number of packets with a given delay

Increase with the increase of cache size

Decrease with the increase of cache size

Delay
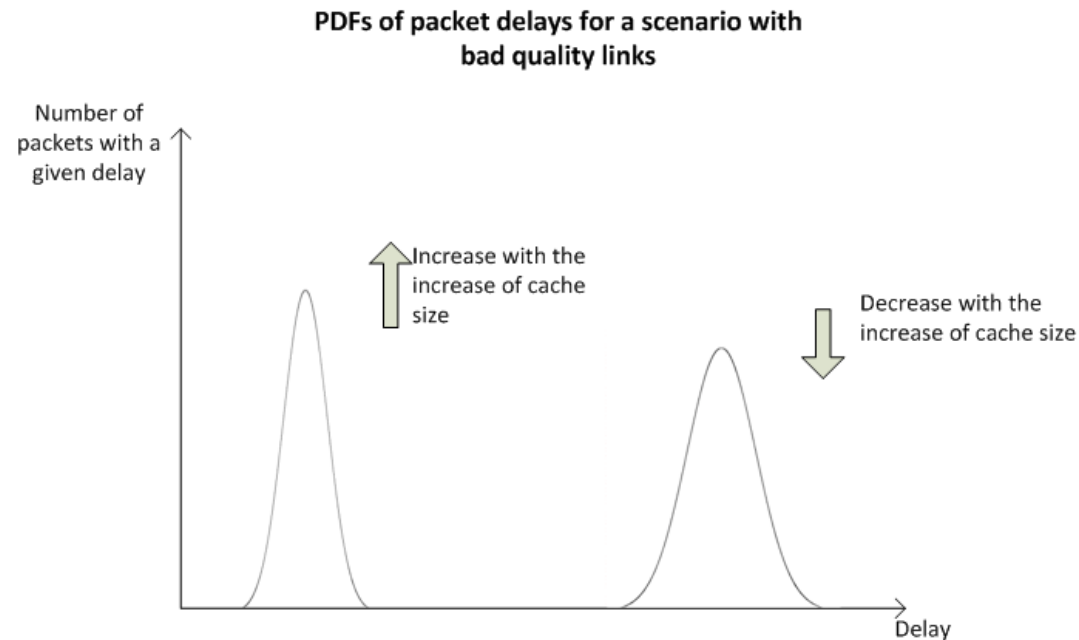
• Links with losses

➢ CCN network was unable to handle the file download

➢ Strategy layer of CCNx does not implement handling of packet losses

Conclusion and future work

- Our results showed some **loopholes** in the present implementation of CCNx like poor **caching strategy**, unable to handle packet losses and **implementation flaws** in MTU size.

- For most of the real world scenarios, **HTTP** still has **better performance** compared **to CCN**, but CCN proved to outperform HTTP in the case of large cache sizes and multiple users/hosts requesting the same content data.

- We also proposed **smart caching** strategies and other implementation changes required in the present CCNx.

- Our measurements were performed by **simulating wide area network** on a small scale network.

- Such a simulation **can not truly represent** a wide area network like Internet where network traffic parameters change with time and space.

- So, we would like to **extend** our measurements and test on more of a **real world**, wide area network (Internet) by using something like research test beds as in **PlanetLab**, as our future work.