Performance analysis of FTP versus CCNx

Reza Tourani, Travis Mick,
Frank Natividad, Satyajayant Misra

Computer Science Department
New Mexico State University

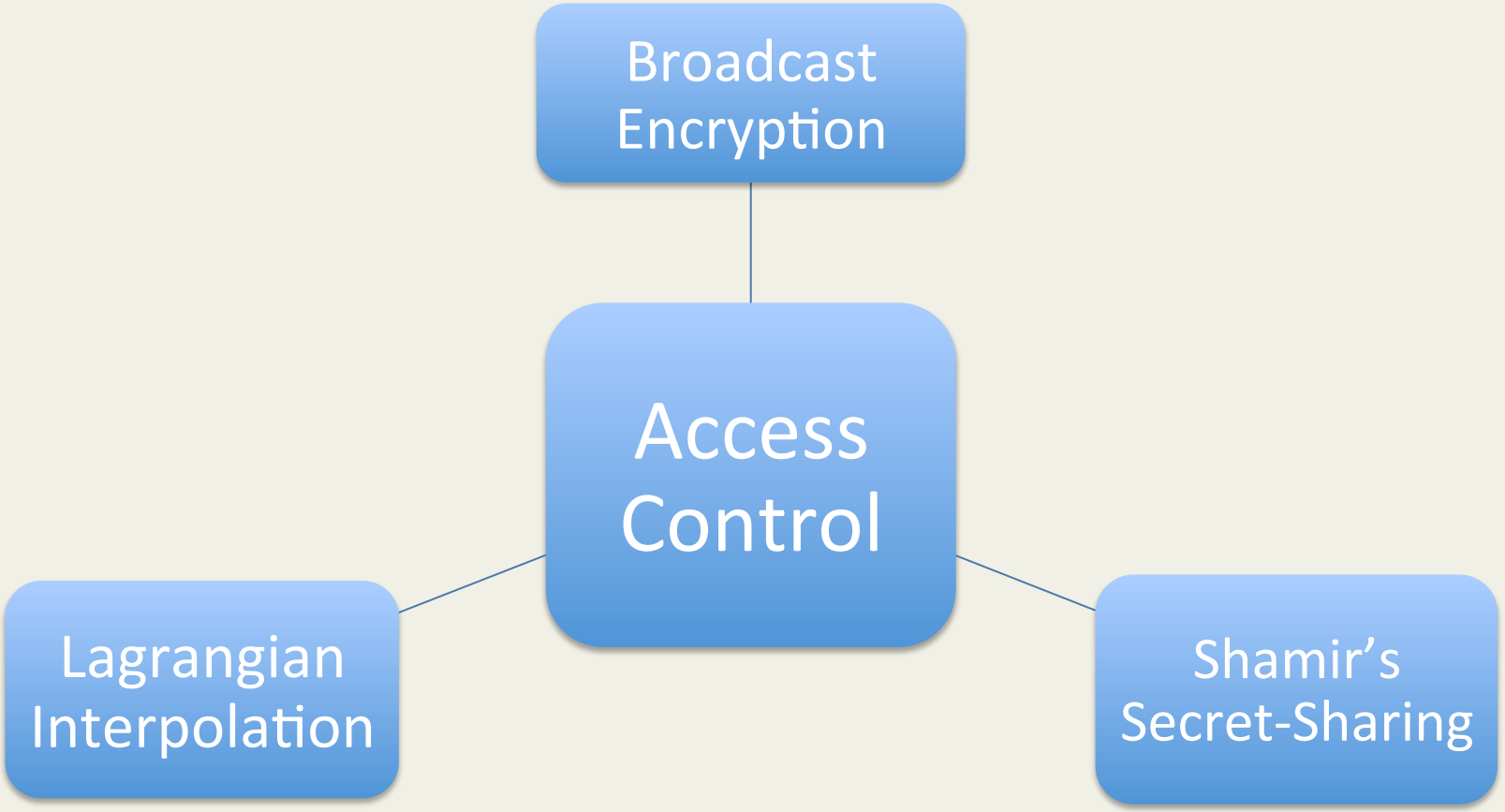
Outline

- Introduction and Motivation
- Problem Definition
- Testbed Setup
- Numerical Results
- Conclusion

NM STATE UNIVERSITY

Secure Content Delivery in ICN*

(Access control)



*S. Misra, R. Tourani, and N. E. Majd. Secure Content Delivery in Information-Centric Networks: Design, Implementation, and Analyses. In Proceedings of the 3rd ACM SIGCOMM Workshop on Information Centric Networking (ICN 2013).

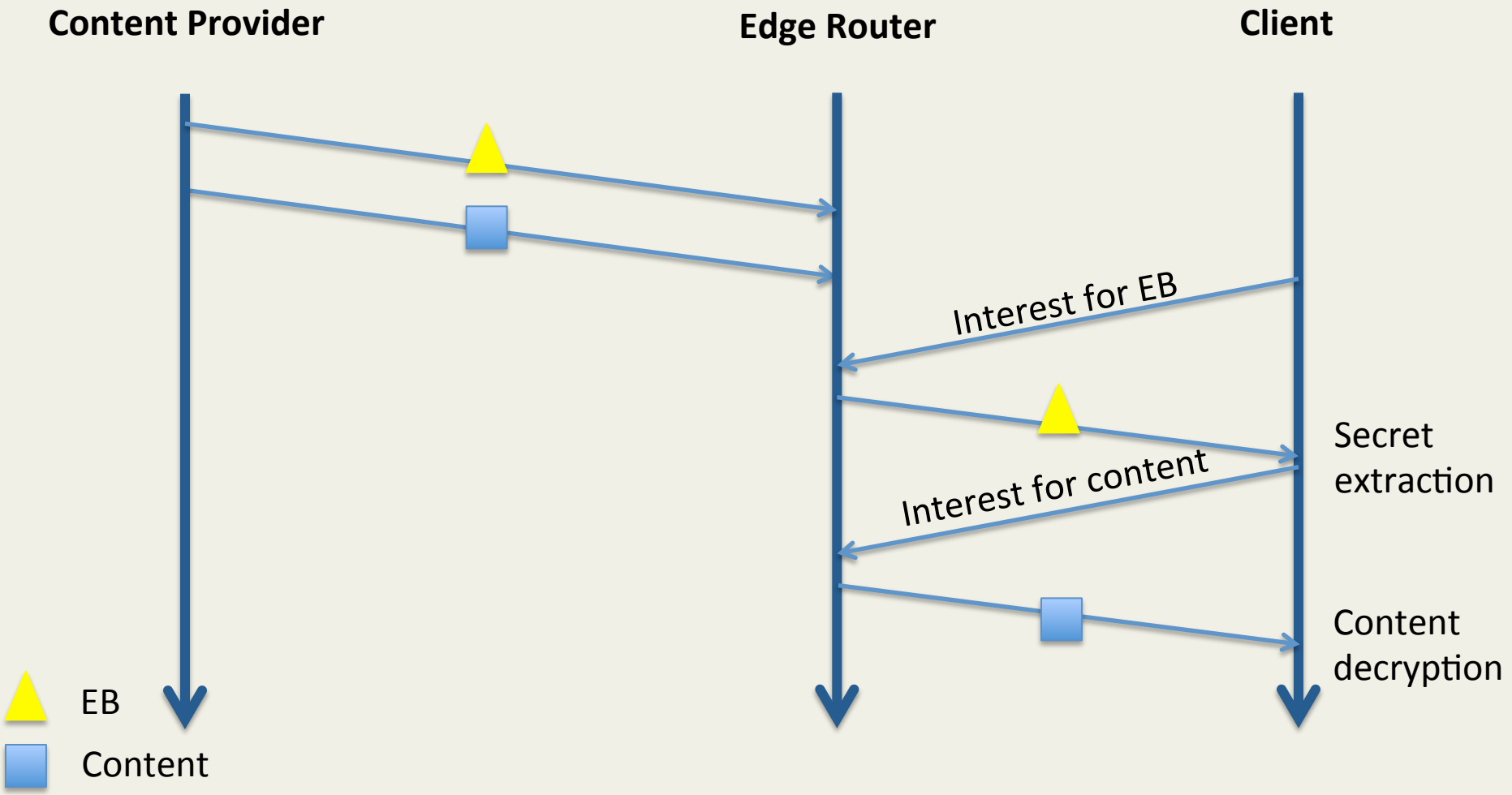
Secure Content Delivery in ICN

- **Server**
  - Polynomial and shares generation
    (generation of ‘’n+t’’ shares, one share per client).
  - *Enabling block (EB)* generation
    (key encryption with ‘’t’’ shares).

- **Client**
  - Secret extraction at the client
    (legitimate client uses its share and the ‘’t’’ shares in EB to extract the key).

NM STATE UNIVERSITY

Secure Content Delivery in ICN



Secure Content Delivery in ICN

Server's pre-computation reduces the computation at the client side.

t (in million)	0.1	0.3	0.5	0.7	1
Laptop Extraction (s)	0.14	0.46	0.71	1.03	1.34
Smart phone Extraction (s)	1.16	3.86	5.92	7.44	10.65
Enabling block size (MB)	1.2	3.6	6	8.4	12

NM STATE UNIVERSITY

Problem: Performance of CCNx vs. FTP

Comparing the performance of CCNx-0.7 and FTP as the baseline communication.

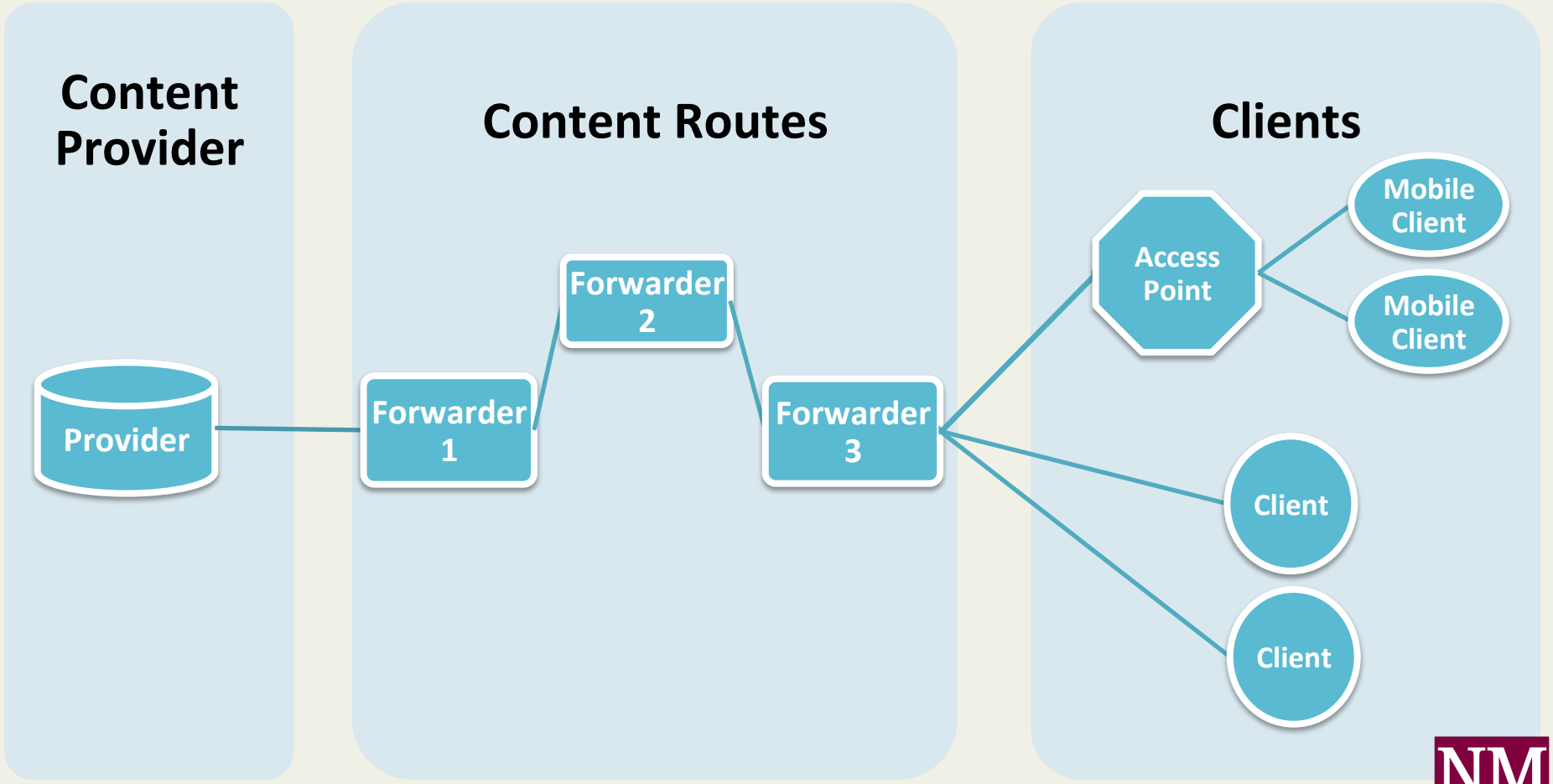
Our criterion is defined as the content download time.

Testbed Setup

- **Content providers**
  - 3 nodes: 2.4 GHz Intel Core i7, 8 GB RAM.
- **Content forwarder**
  - 4 nodes: 2.5 GHz Intel Core 2 Quad, 3.8 GB RAM.
- **Clients**
  - 6 nodes: 1.66 GHz Intel Centrino Duo, 2.5 GB RAM. (Stationary)
  - 1 node: 3 GHz Intel Xeon Quad Core, 2 GB RAM. (Stationary)
  - 3 Nexus 4: 1.5 GHz Quad core, 2GB RAM. (Mobile)
  - 1 Nexus 5: 2.3 GHz Quad core, 2GB RAM. (Mobile)
- **Access point**
  - 802.11 n
- **Switches**
  - 100 Mb/s switches

NM STATE UNIVERSITY

Testbed Setup

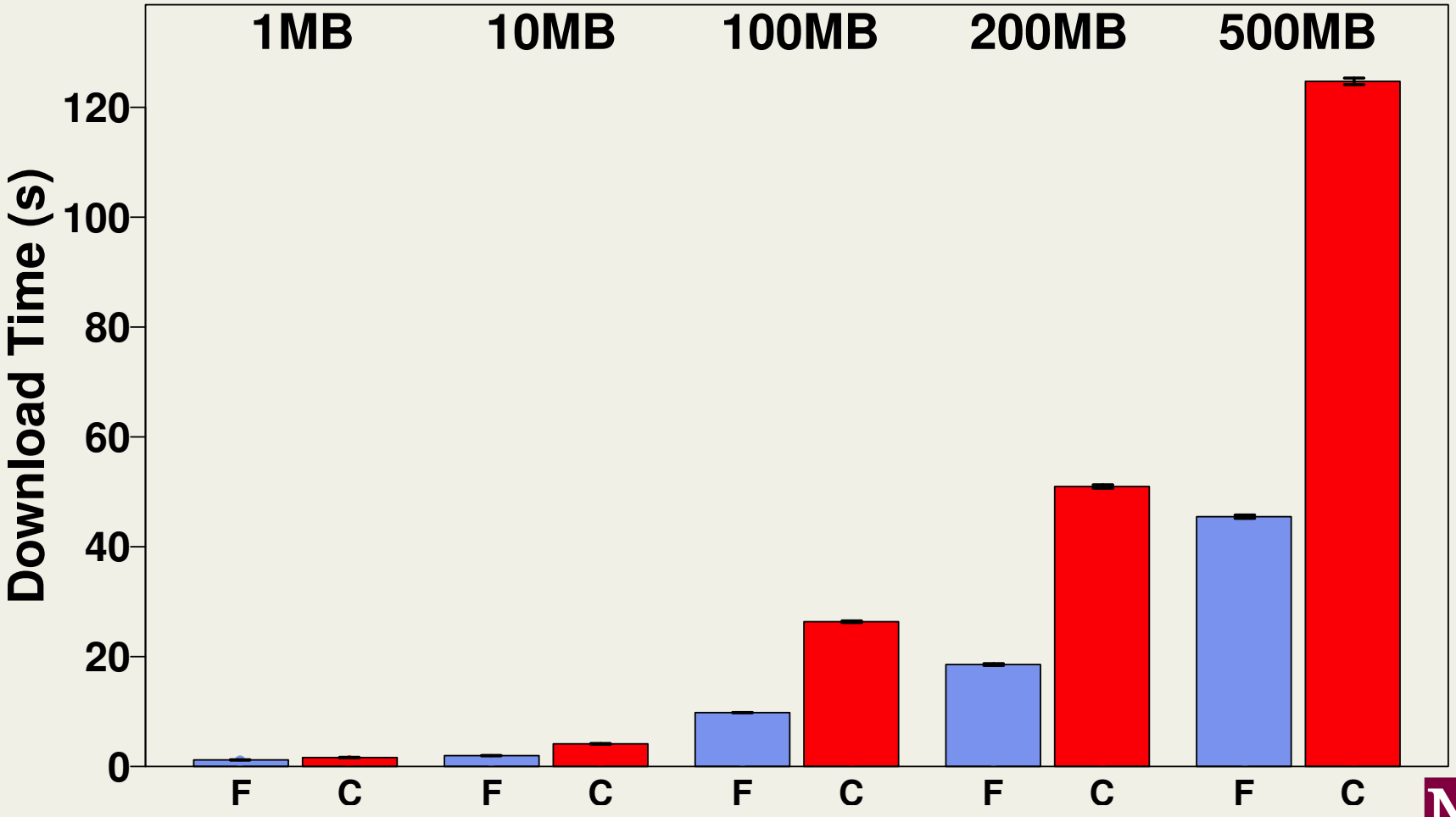


Test Setup

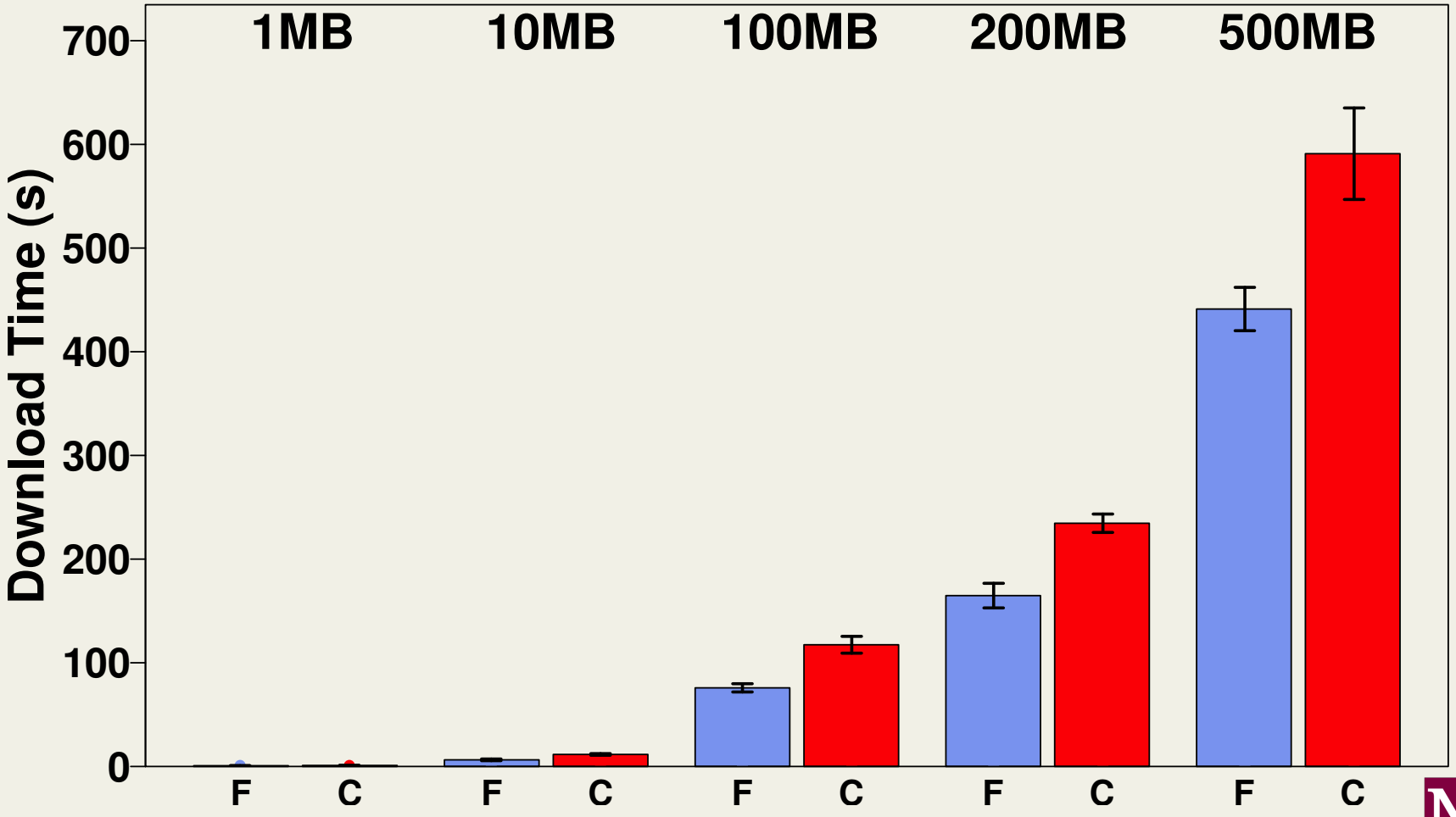
- Using the CCNx-0.7 *ccnputfile* and *ccngetfile* tools to store/retrieve contents to/from the content provider.
- One client requests the content from the provider.
- Caching was enabled on all the routers.
- Various content objects (1 MB, 10 MB, 100 MB, 200 MB, and 500MB).
- For the FTP client and server, we used the GNU Wget package.

NM STATE UNIVERSITY

Stationary Client



Mobile Client



Effect of Cache Distance on the Stationary Client Download Time CCNx 0.7

	CCNx			FTP
Content Size	Local Cache	1-hop	4-hop	4-hop
1 MB	1.54	1.62	2.21	1.19
10 MB	3.21	4.11	9.13	1.97
100 MB	12.92	26.34	70.47	9.81
200 MB	24.87	50.93	155.54	18.54
500 MB	57.20	124.73	326.49	45.46

NM STATE UNIVERSITY

Conclusion

- FTP is supported by the TCP's sophisticated AIMD congestion control algorithm and pipelining versus static pipelining of CCNx.
- Content store operations (the same delay was observed in *ccnrm* tool).
- The need for a sophisticated congestion control and flow control algorithms for CCN.

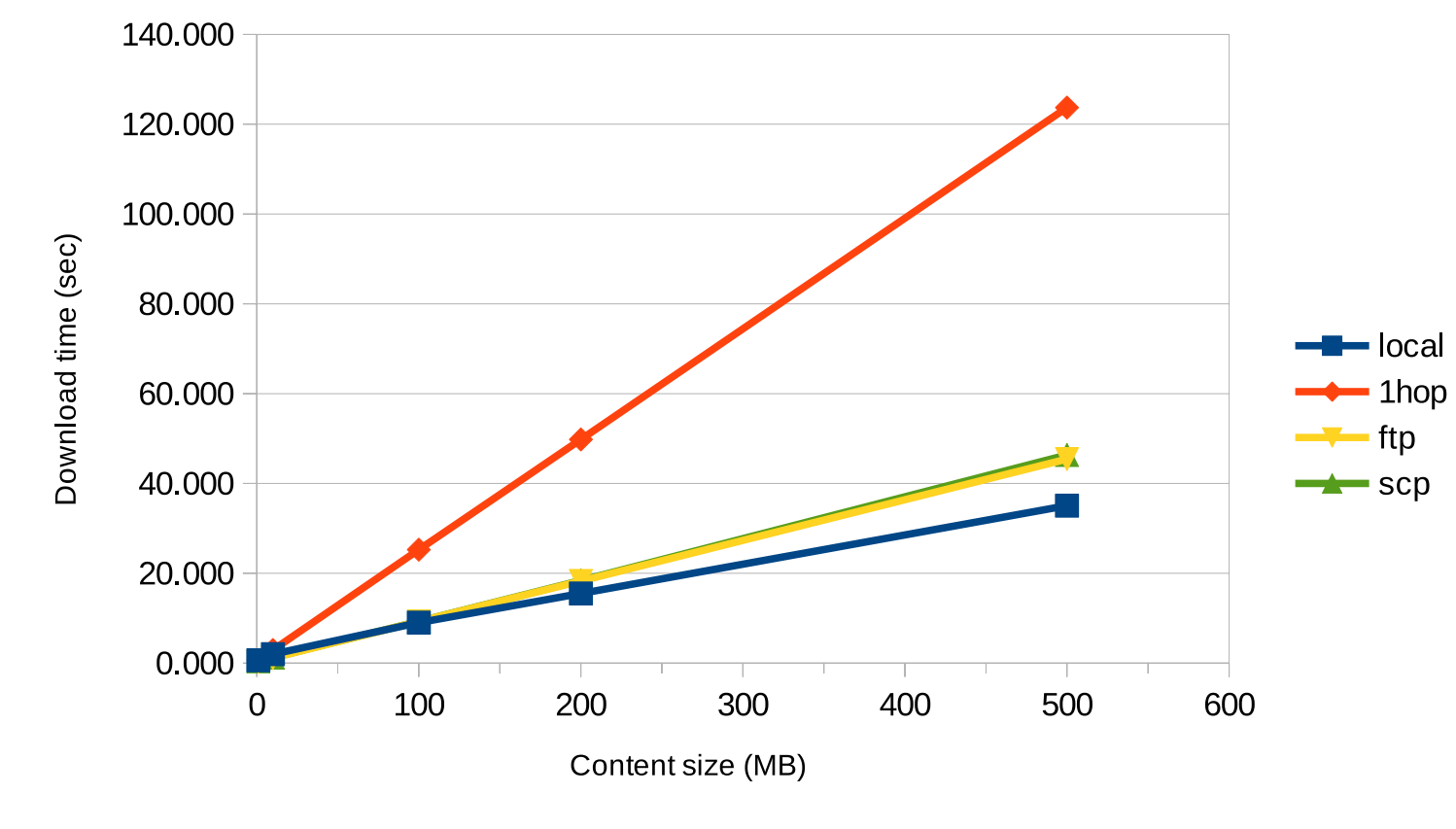
NM STATE UNIVERSITY

Effect of Cache Distance on the Stationary Client Download Time CCNx 0.8

	CCNx		FTP
Content Size	Local Cache	1-hop	3-hop
1 MB	0.61	0.65	0.30
10 MB	1.98	2.91	1.11
100 MB	9.01	25.25	9.27
200 MB	15.49	49.79	18.27
500 MB	35.04	123.70	45.41

NM STATE UNIVERSITY

Effect of Cache Distance on the Stationary Client Download Time CCNx 0.8



Thank you!