A Network Evolution Story:

from Communication, to Content Distribution, to Real-Time Computation

Antonia Tulino

Università degli Studi di Napoli Federico II & Tandon School of Engineering NYU

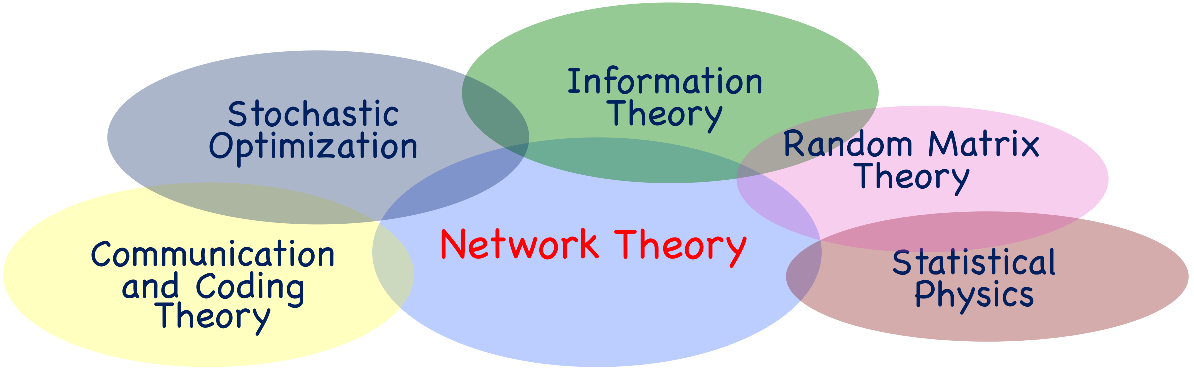
Outline

- Communication

- Content Distribution

  Efficient Content Storage and Delivery

  - Cache-aided coded multicast
  - Distributed network compression
  - Dynamic Data



- Real-time Computation

  Efficient Service Configuration (Storage/Computation/Delivery)

  - Network Slicing (NFV/SDN)
  - Mobile Edge Computing (MEC)
  - Real-time Stream processing

Acknowledgements

NOKIA Bell Labs

- Jaime Llorca, Marc Roelands, Alessandra Sala, Narayan Raman, Nakjung Choi, Danny Raz (now Technion).

NYU NEW YORK UNIVERSITY

- Elza Erkip, Parisa Hassanzadeh.

USC UNIVERSITY OF SOUTHERN CALIFORNIA

- Giuseppe Caire (now TUB), Andreas Molisch, Mingue Ji (now Utah), Hao Feng.

The University of Texas at Austin

- Alex Dimakis, Karthikeyan Shanmugam.

MIT Massachusetts Institute of Technology

- Jianan Zhang, Abhishek Sinha, Eytan Modiano.

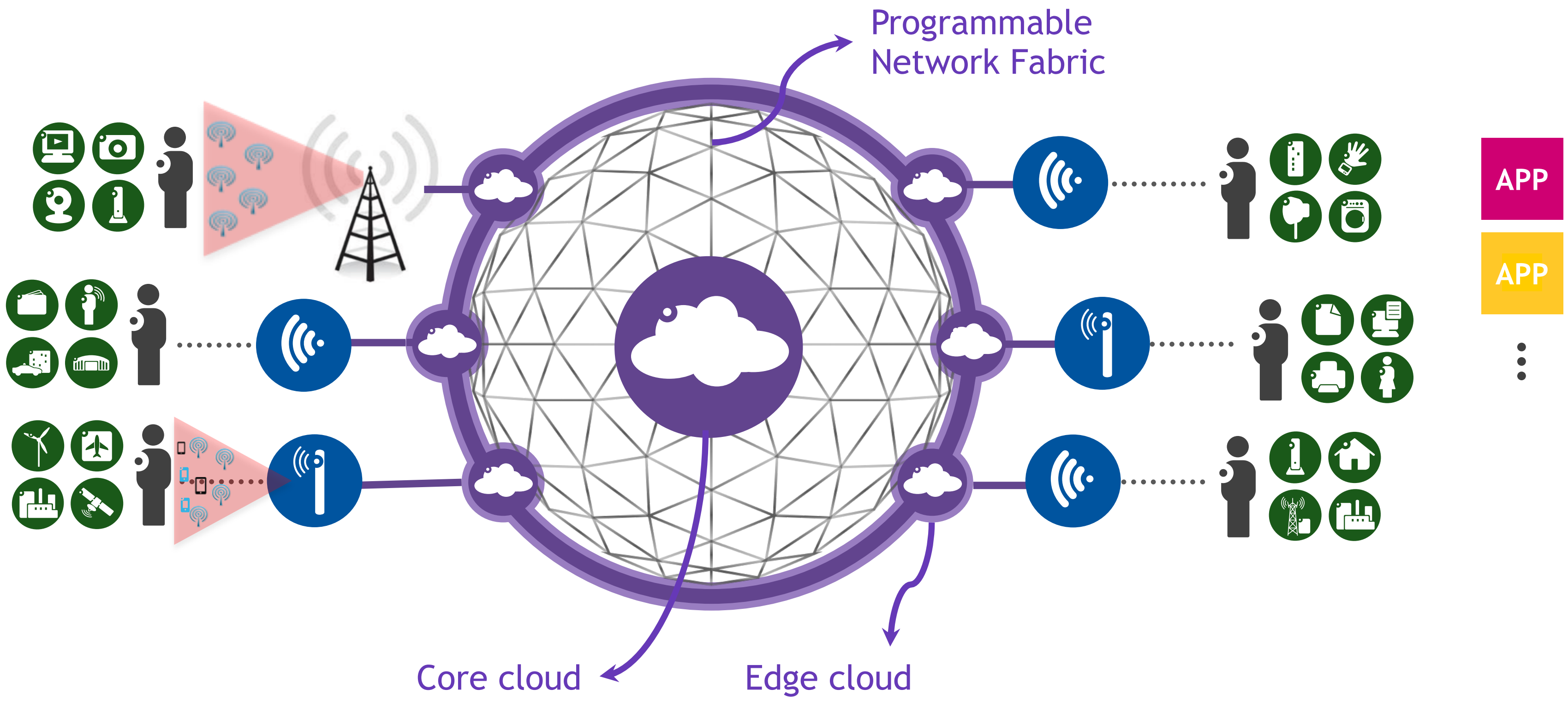
Yale University

- Konstantinos Poularakis, Leandros Tassiula.

UAB Universitat Autònoma de Barcelona

- Marc Barcelo, Jose Vicario, Antoni Morell

CLOUD-INTEGRATED NETWORKS AS UNIVERSAL COMPUTE PLATFORMS



(5G & beyond) cloud-integrated networks will become universal general-purpose compute platforms, where a large variety of services and applications will be deployed in the form of slices within a common physical infrastructure taking advantage of the cloud network's reach, elasticity, and flexibility.

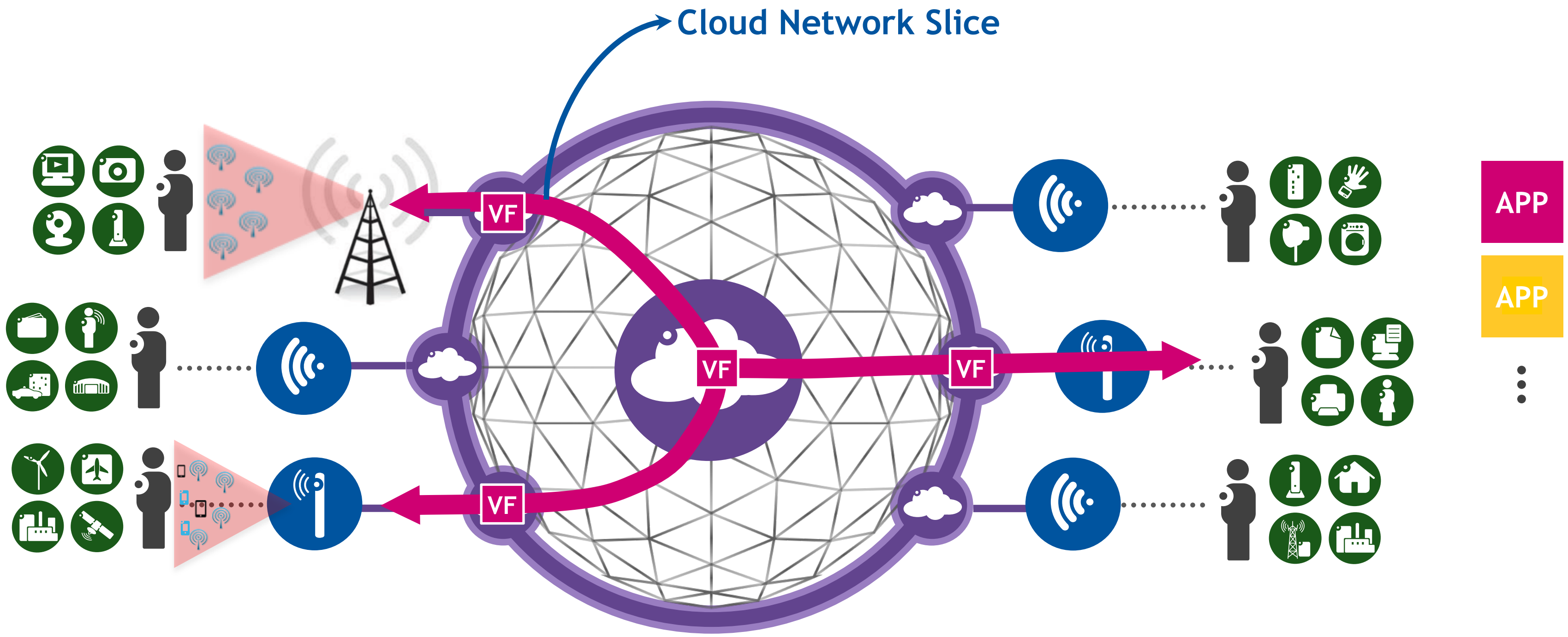
M. Weldon, "The Future X Network: A Bell Labs Perspective," CRC PRESS, October 2015.

CLOUD-INTEGRATED NETWORKS AS UNIVERSAL COMPUTE PLATFORMS

- Ideal for next generation services

  1) Network services

     - 5G slices



CLOUD-INTEGRATED NETWORKS AS UNIVERSAL COMPUTE PLATFORMS

- Ideal for next generation services

  1) Network services
     - 5G slices

  2) Automation services

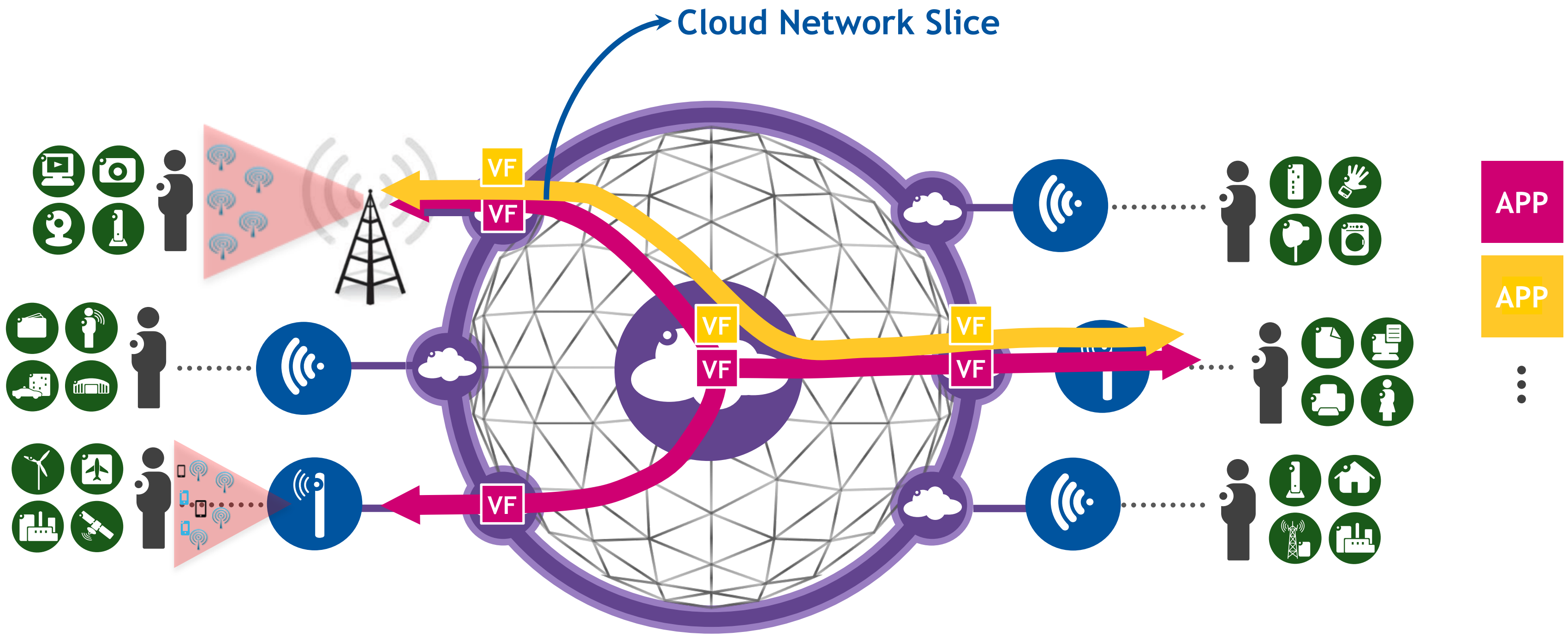
     Smart X, IoT

  3) Augmented experience services

     Virtual X, Augmented X (e.g. reality/cognition)

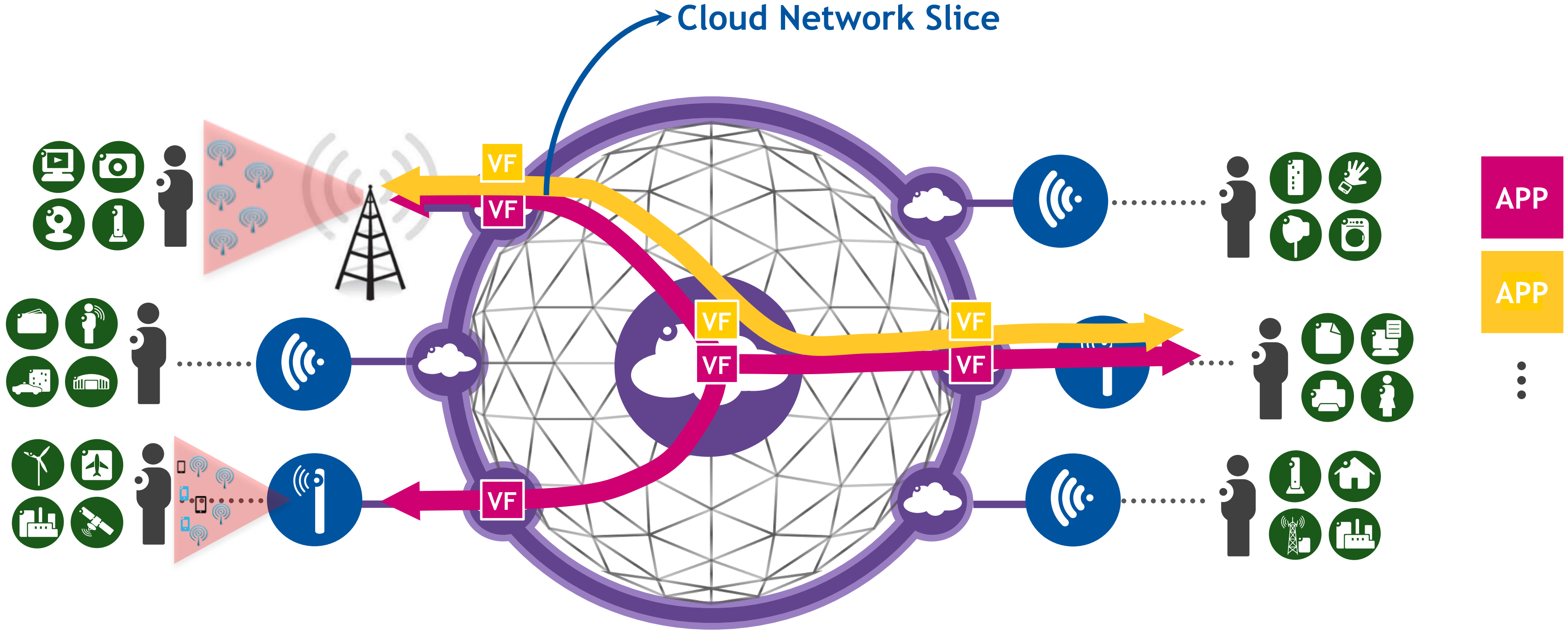
     Immersive video

     Real-time computer vision/scene analysis



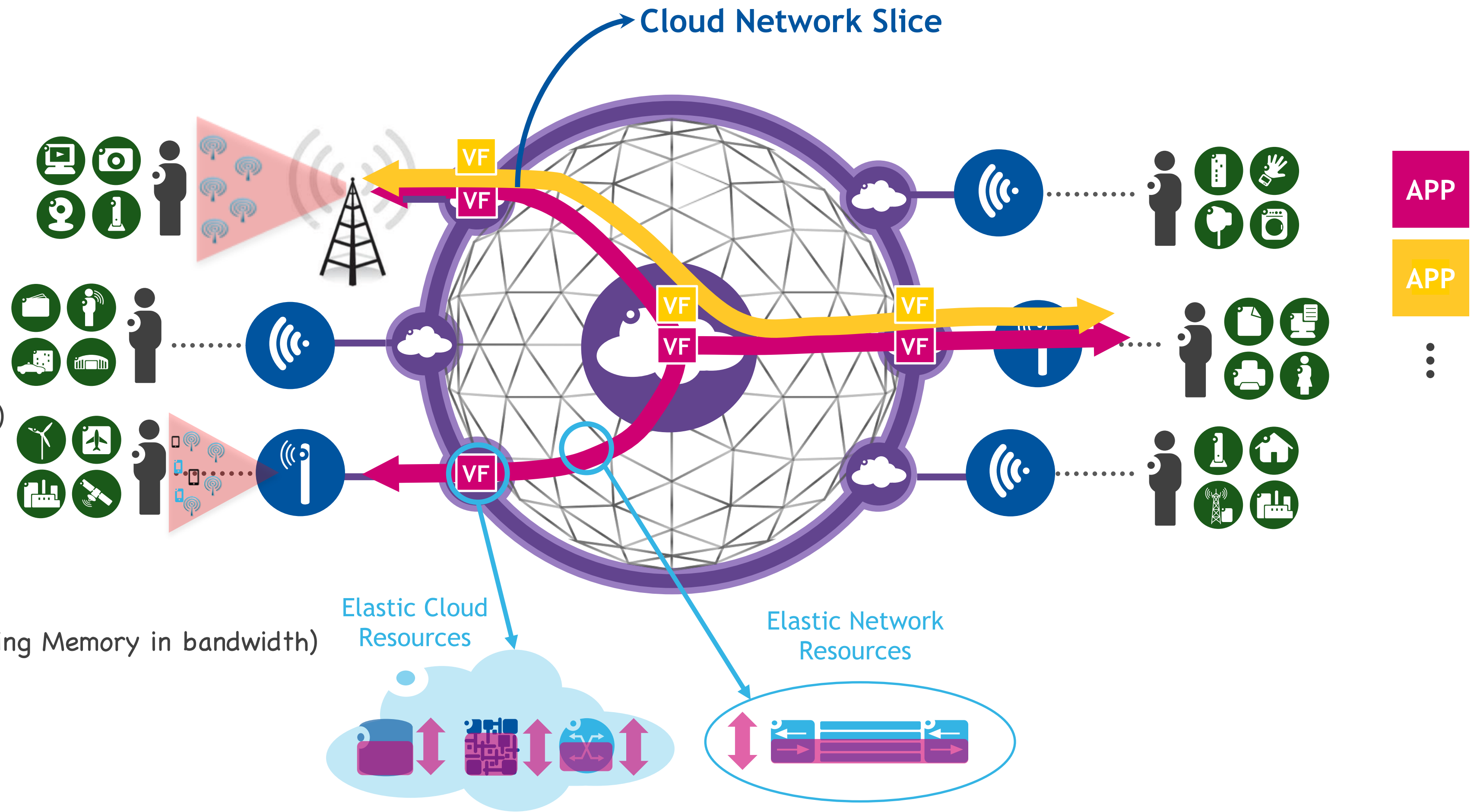
CLOUD-INTEGRATED NETWORKS AS UNIVERSAL COMPUTE PLATFORMS

- Opportunities
  - Users can consume resource- and <u>interaction</u>-intensive applications from resource-limited devices
  - Operators can reduce costs and create new value-added services
  - Overall sustainability



CLOUD-INTEGRATED NETWORKS AS UNIVERSAL COMPUTE PLATFORMS

- Key enablers
  - Network function virtualization (NFV)
  - Software defined networking (SDN)
  - Network Slicing
  - Advance RAT (Turning space in bandwidth)
    - Network densification,
    - Massive MIMO & mmW
    - D2D communications
  - Cooperative information sharing (Turning Memory in bandwidth)
    - Cooperative (edge) caching,
    - Network coding,
    - multicast transport
    - Network Compression



Objectives

- Understand the fundamental efficiency limits of the future networked cloud
- Develop practical solutions that push the network closer to its limits

- **NFV**: move hardware appliances into software functions deployed at multiple cloud locations and elastically scaled computing resources.

- **SDN**: program the network in between and steer network flows through the appropriate set of functions.

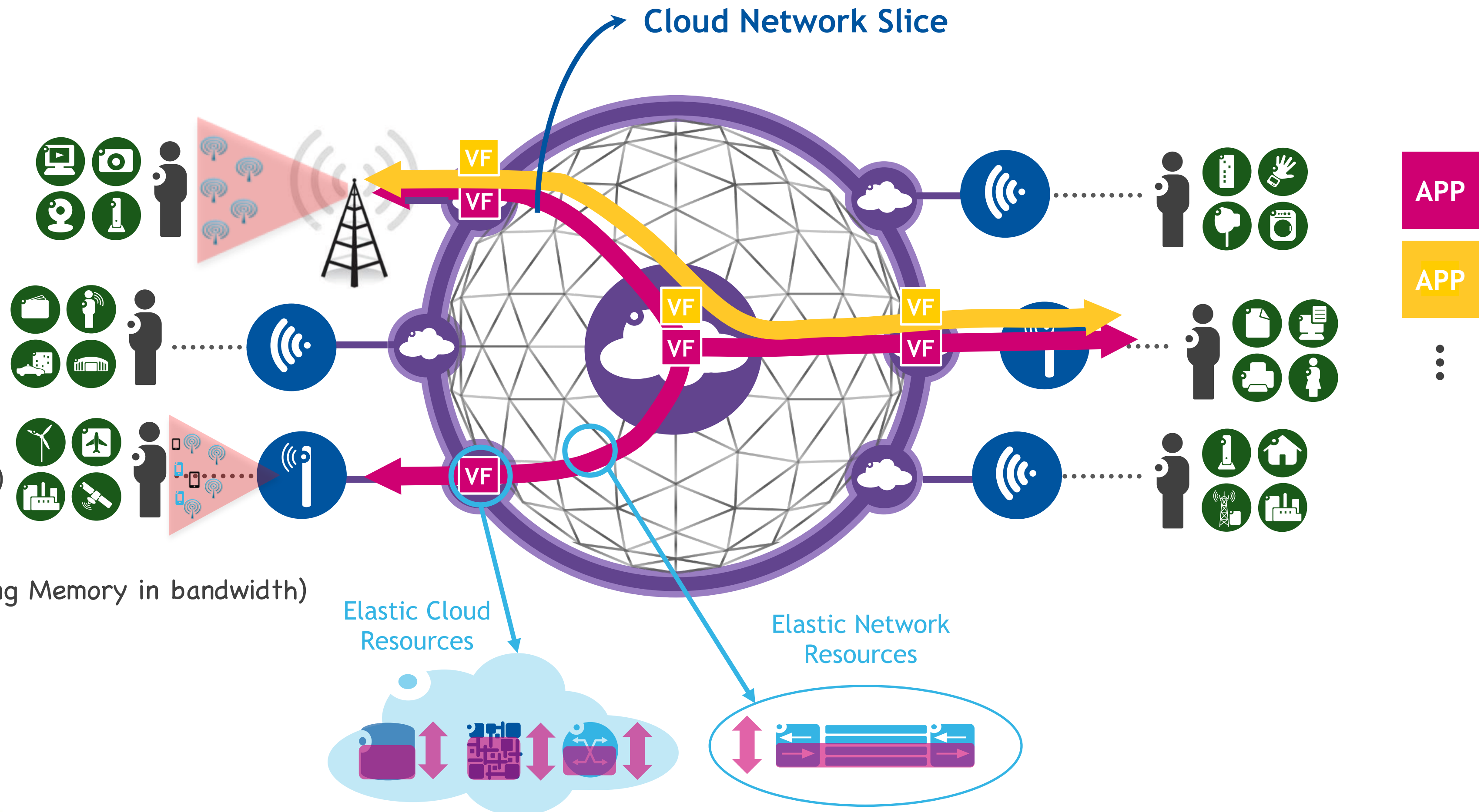
- **Network slicing**: create cloud network slices which are hence elastic and programmable.

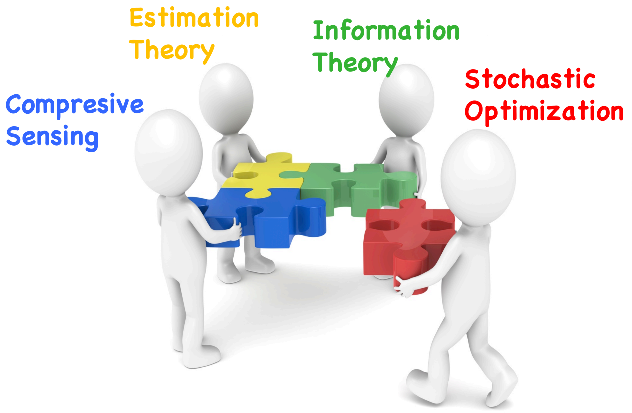
Elastically allocate both cloud (storage and computing) and network resources according to changing demands, in order to meet service requirements while minimizing the use of the physical infrastructure.

- **Network Densification**
- **Massive MIMO**
- **Millimeter wave (mmW)**
- **D2D communications**

CLOUD-INTEGRATED NETWORKS AS UNIVERSAL COMPUTE PLATFORMS

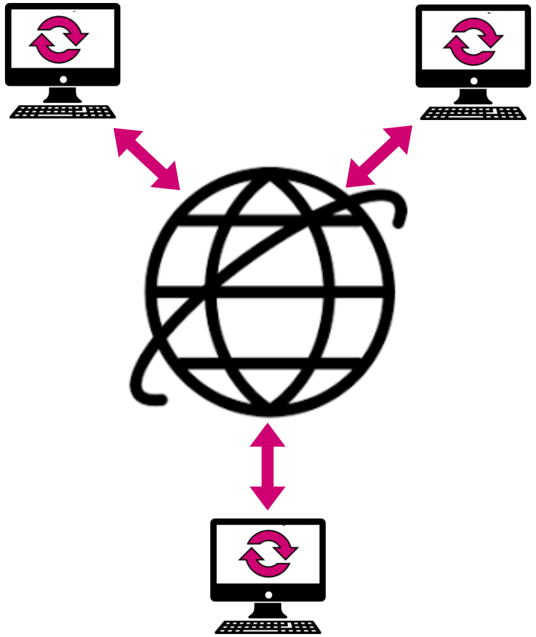
- Key enablers
  - Network function virtualization (NFV)
  - Software defined networking (SDN)
  - Network Slicing
  - Advance RAT (Turning space in bandwidth)
  - Cooperative information sharing (Turning Memory in bandwidth)





TOWARDS REAL-TIME AUGMENTED COGNITION

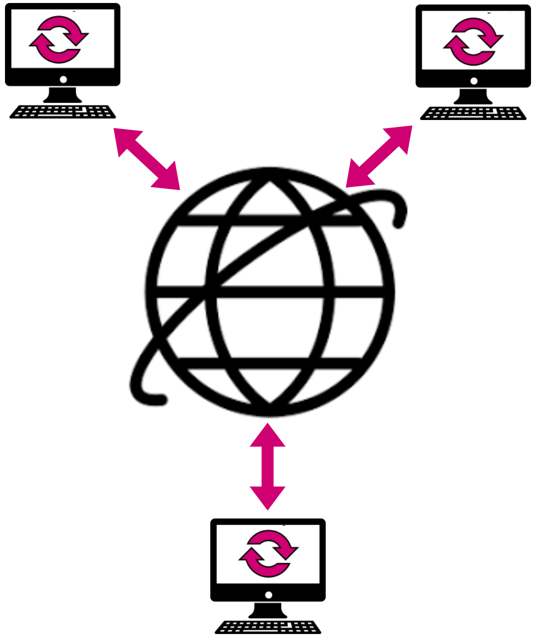
Communication



- Resource limited
- Interaction limited

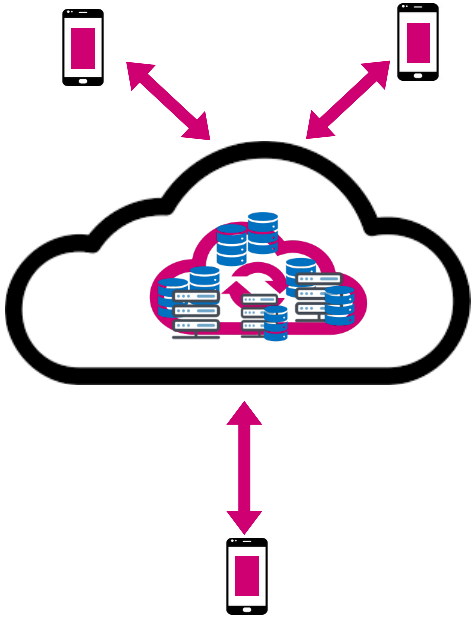
TOWARDS REAL-TIME AUGMENTED COGNITION

Communication



- Resource limited
- Interaction limited

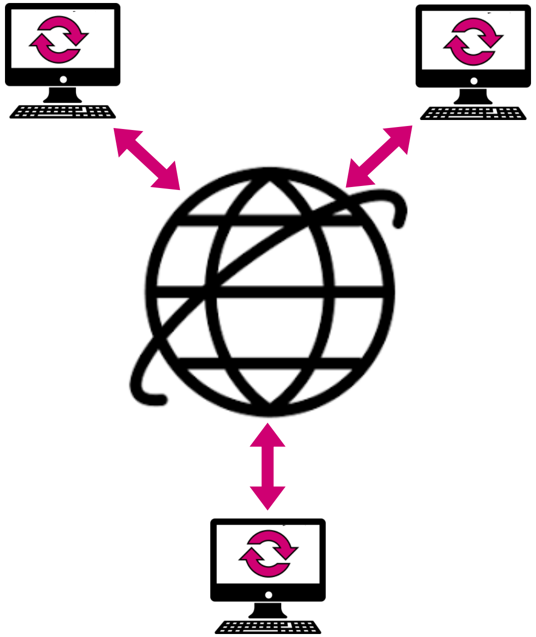
Content Distribution



- Resource intensive
- Interaction limited

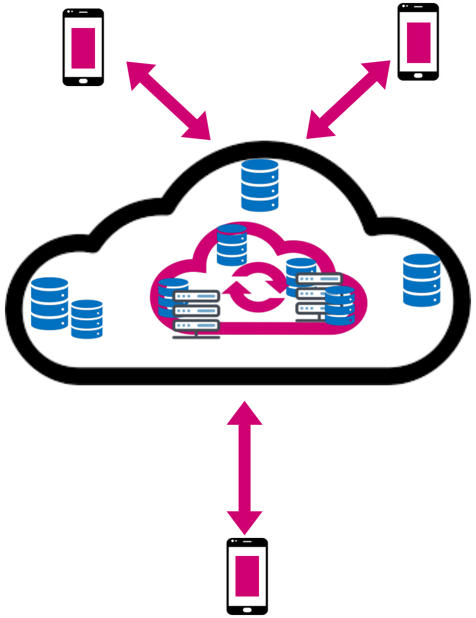
TOWARDS REAL-TIME AUGMENTED COGNITION

Communication



- Resource limited
- Interaction limited

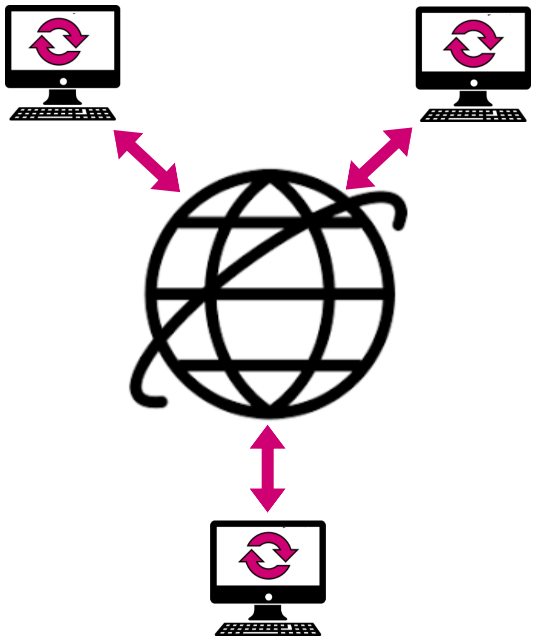
Content Distribution



- Resource intensive
- Interaction limited

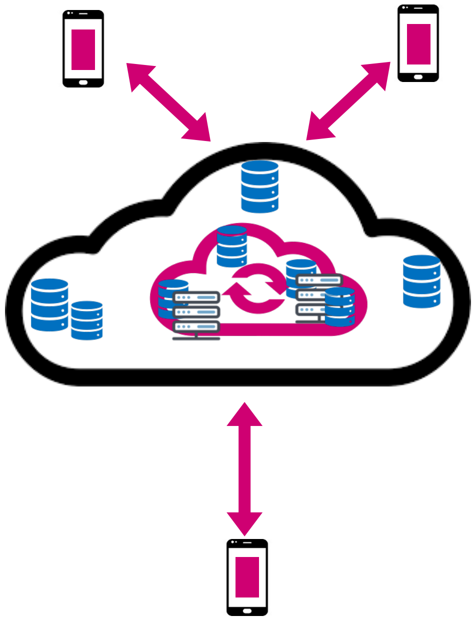
TOWARDS REAL-TIME AUGMENTED COGNITION

Communication



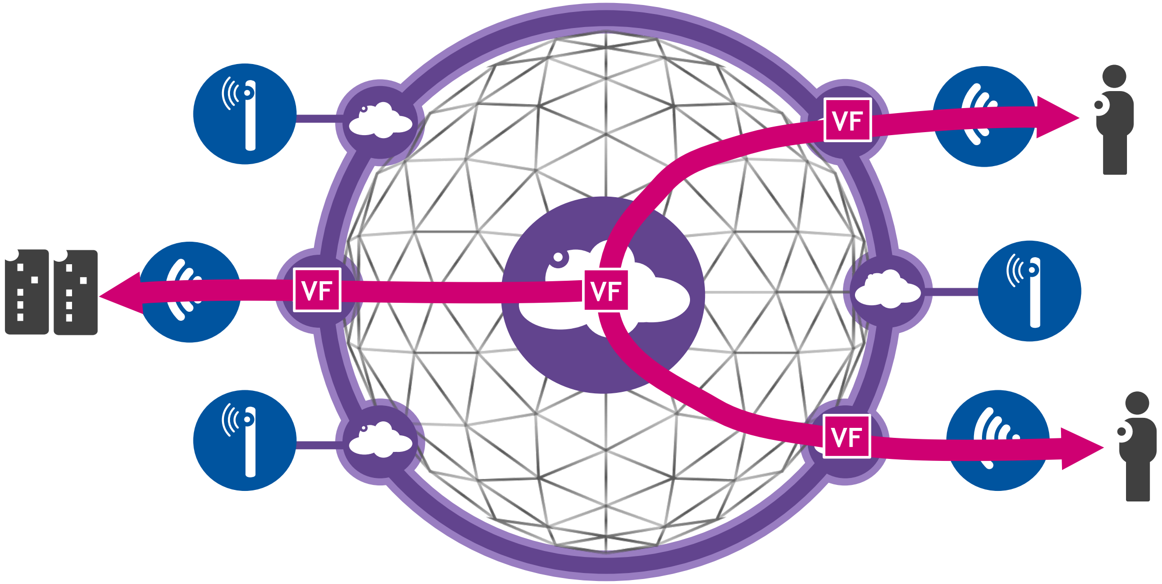
- Resource limited
- Interaction limited

Content Distribution



- Resource intensive
- Interaction limited

Real-time Computation



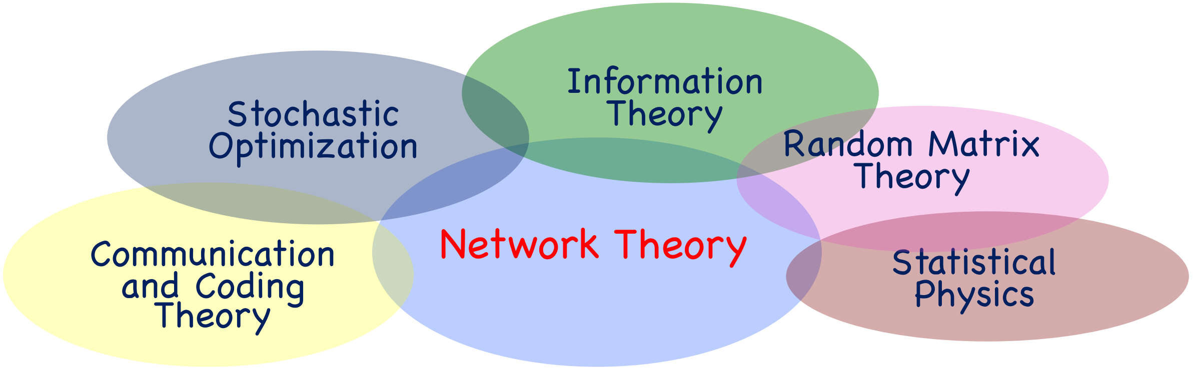
- Resource intensive
- Real-time interaction

Bridging the time-scale gap between information capture/sensing, analysis/processing, and delivery/consumption

Outline

- Communication
- Content Distribution

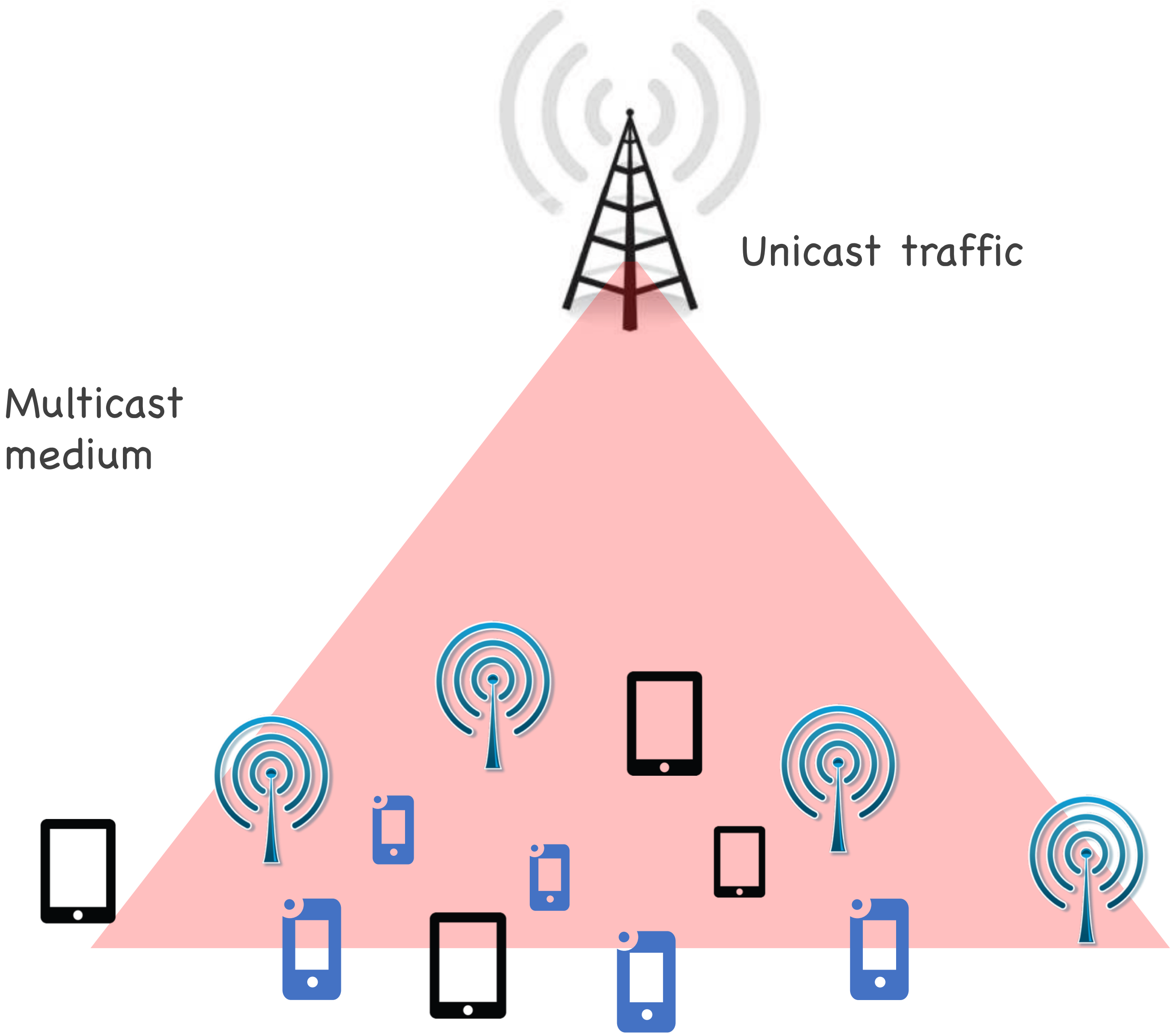
  Efficient Content Storage and Delivery
  - Cache-aided coded multicast
  - Distributed network compression
  - Dynamic Data



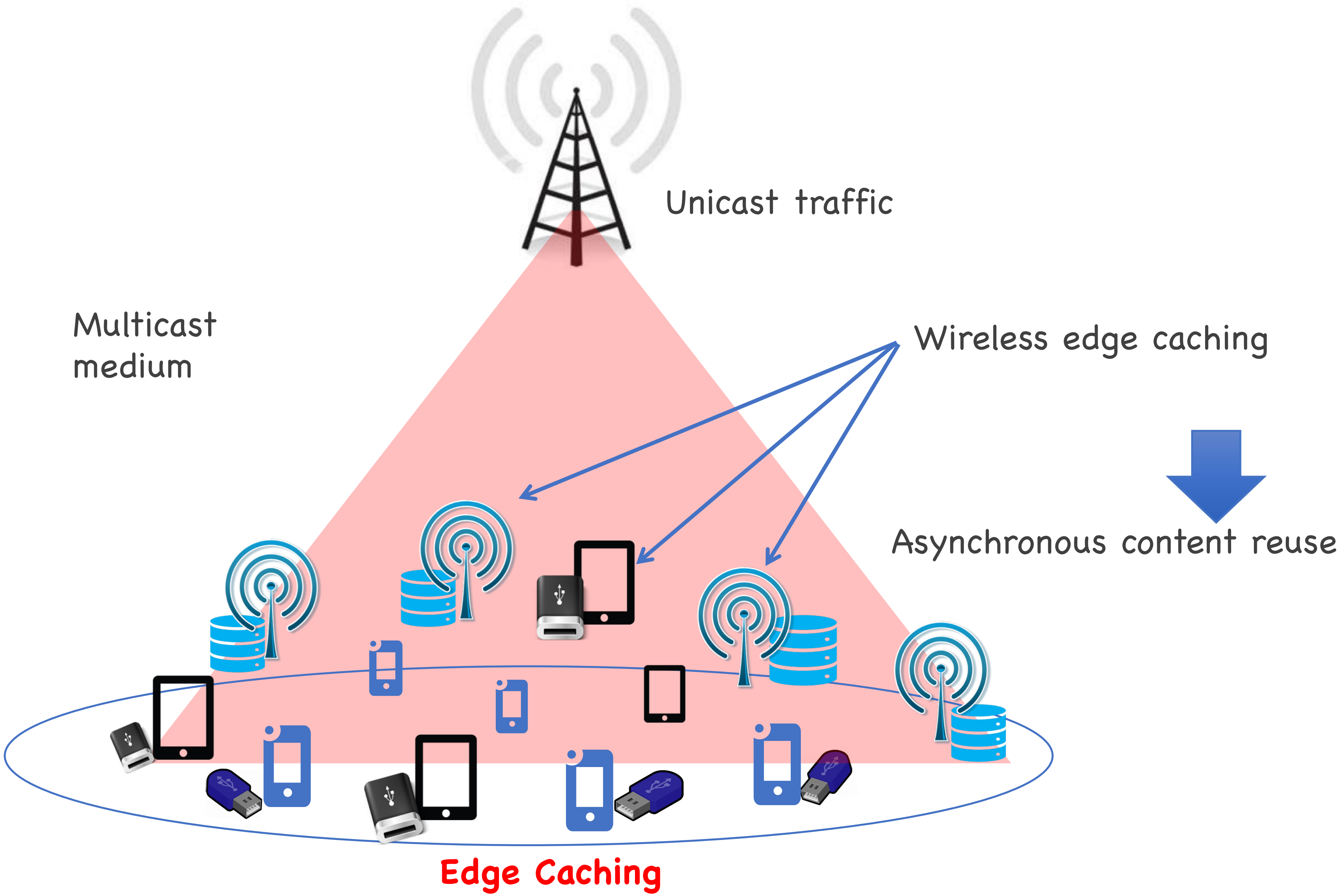
- Real-time Computation

  Efficient Service Configuration (Storage/Computation/Delivery)
  - Network Slicing (NFV/SDN)
  - Mobile Edge Computing (MEC)
  - Real-time Stream processing

The Wireless Bottleneck



The Wireless Bottleneck



The Wireless Bottleneck

Approaches

FemtoCaching: Caching at the infrastructure side (SBS, Helpers)

M: Memory at femtocaching

N: number of files

$$\mathcal{L}oad \simeq \Theta\left(\frac{N}{M}\right)$$

Rate ≈ $\mathcal{L}oad$ ≈ Delay

$$\mathcal{L}oad = \frac{\text{average number of transmissions}}{\text{File size}}$$

The Wireless Bottleneck

Approaches

FemtoCaching: Caching at the infrastructure side (SBS, Helpers)

M: Memory at femtocaching

N: number of files

$$\mathcal{L}oad \simeq \Theta\left(\frac{N}{M}\right)$$

Rate ≈ $\mathcal{L}oad$ ≈ Delay

$$\mathcal{L}oad = \frac{\text{average number of transmissions}}{\text{File size}}$$

Requires infrastructure nodes to grow linearly with the users.

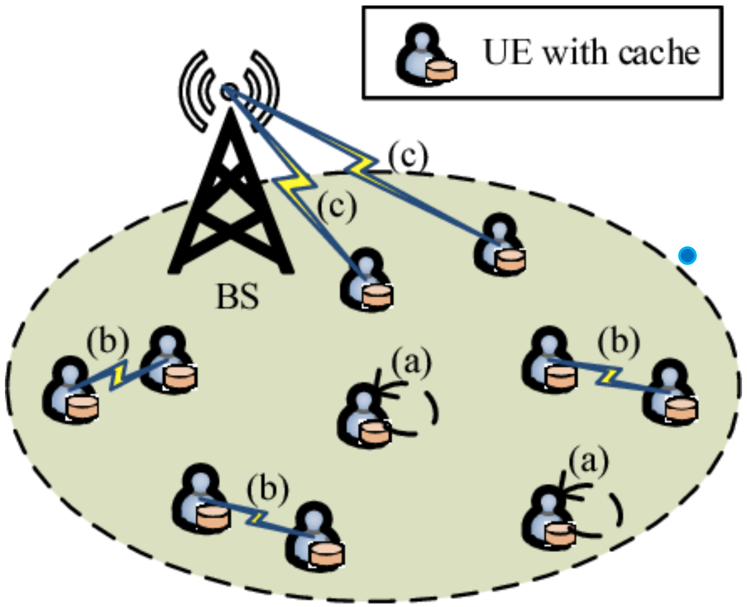
The Wireless Bottleneck

Approaches

D2D Caching: content replication and multi-hop.

M: Memory at user device

N: number of files



$$\mathcal{L}oad \simeq \Theta\left(\frac{N}{M}\right)$$

Rate ≈ $\mathcal{L}oad$ ≈ Delay

$$\mathcal{L}oad = \frac{\text{average number of transmissions}}{\text{File size}}$$

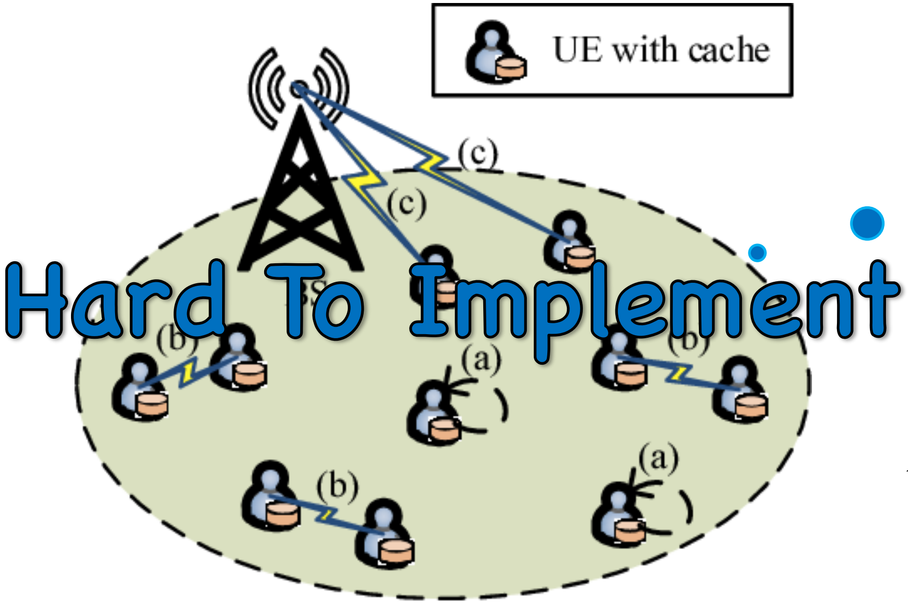
The Wireless Bottleneck

Approaches

D2D Caching: content replication and multi-hop.

M: Memory at user device

N: number of files



$$\mathcal{L}oad \simeq \Theta\left(\frac{N}{M}\right)$$

Rate ≈ $\mathcal{L}oad$ ≈ Delay

$$\mathcal{L}oad = \frac{\text{average number of transmissions}}{\text{File size}}$$

Requires no infrastructure but very hard to implement

- no good D2D standard in place,
- coordination across a large network

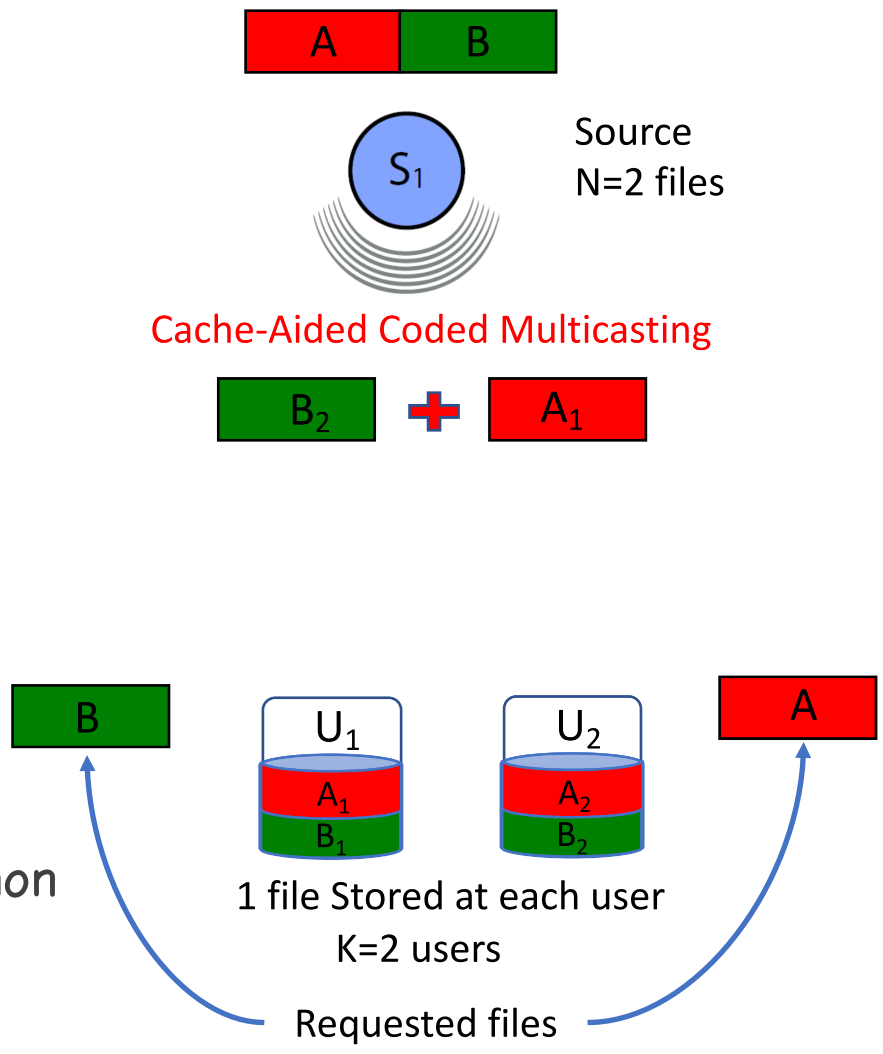
The Wireless Bottleneck

Question:

Can we achieve scalability with finite infrastructure and no D2D communication?

Yes we can!

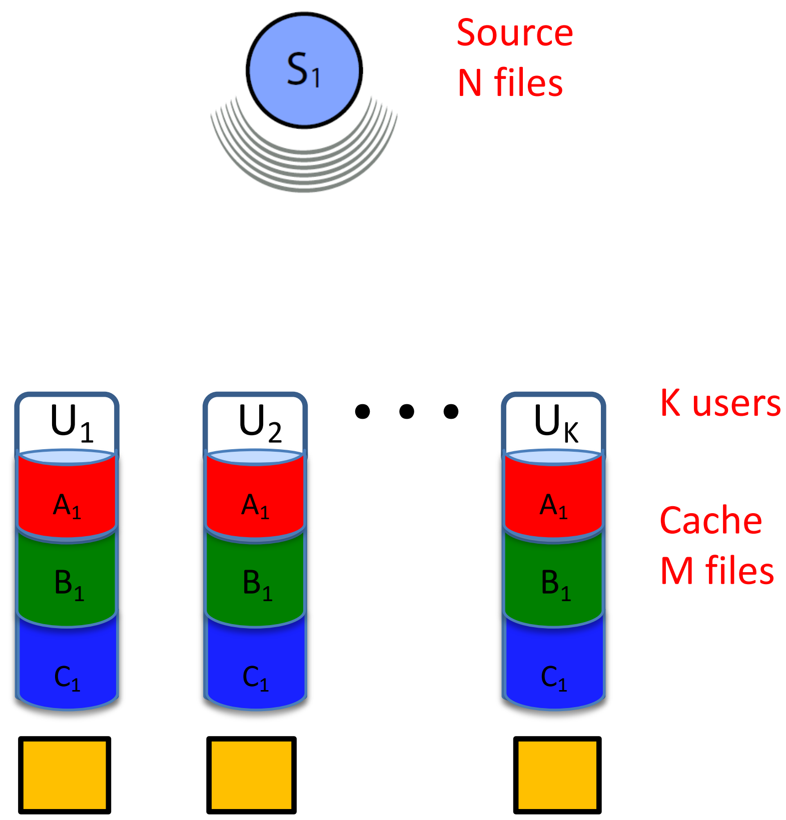
Cache-Aided Coded Multicast (CCM):



Main Idea:

- leverages side information at wireless edge caches to efficiently serve jointly multiple unicast demands via common multicast transmissions,
- leads to load reductions that are proportional to the aggregate cache size.

Cache-Aided Coded Multicast



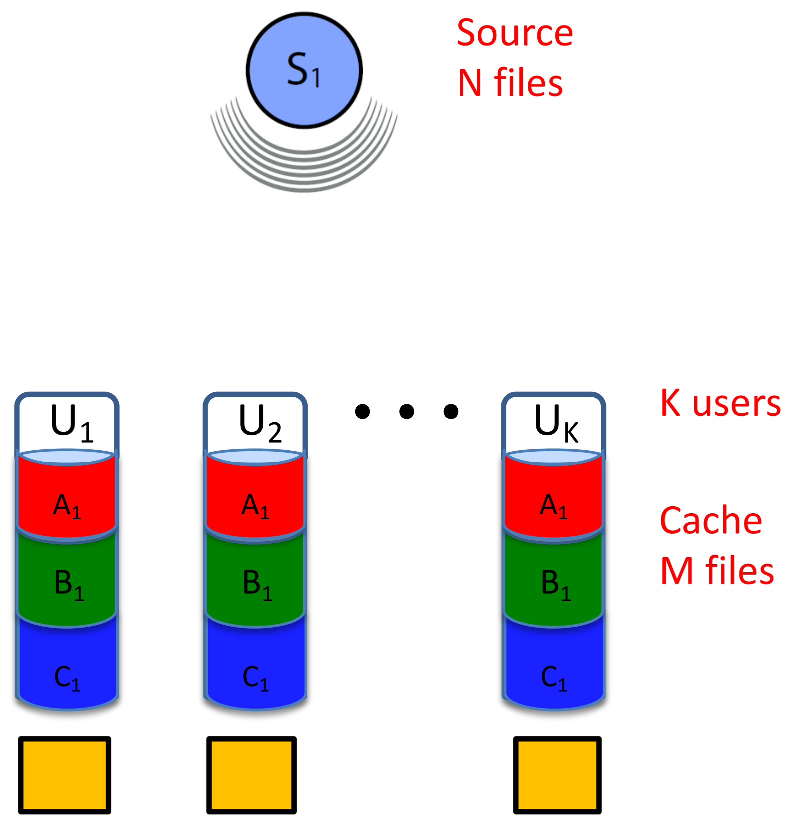
Fractional Cooperative Caching (Cache Encoder)
- Split files into F packets and store them strategically

Coded Multicast
- Coded multicast transmission simultaneously serve multiple distinct requests via index coding

Delivery Phase

Caching Phase

Cache-Aided Coded Multicast



Normalized user's cache size

Think of $\mu = \frac{M}{N} = \frac{\text{cache size}}{\text{num. of files}}$ as a constant

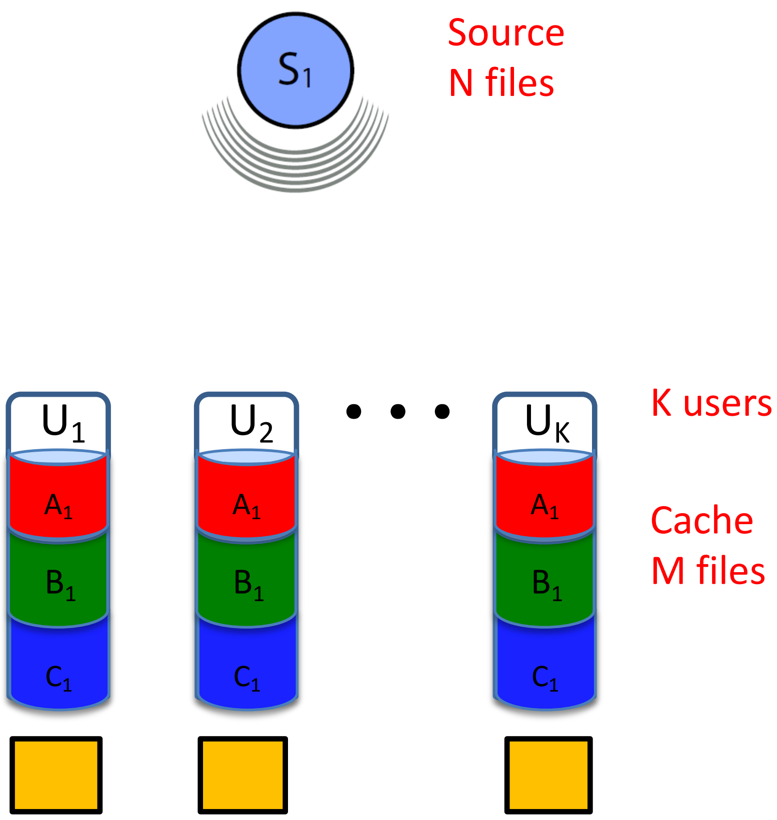
Fractional Cooperative Caching (Cache Encoder)

- Split files into F packets and store them strategically

Coded Multicast

- Coded multicast transmission simultaneously serve multiple distinct requests via index coding

Cache-Aided Coded Multicast



Normalized user's cache size

Think of $\mu = \frac{M}{N} = \frac{\text{cache size}}{\text{num. of files}}$ as a constant

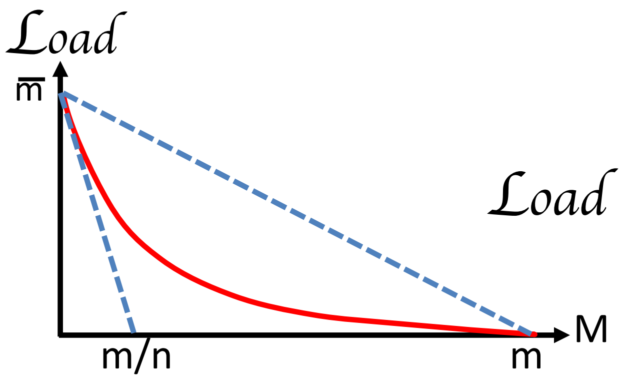
Fractional Cooperative Caching (Cache Encoder)

- Split files into F packets and store them strategically

Coded Multicast

- Coded multicast transmission simultaneously serve multiple distinct requests via index coding

In the relevant regime of KM ≫ N (i.e. Kμ ≫ 1)

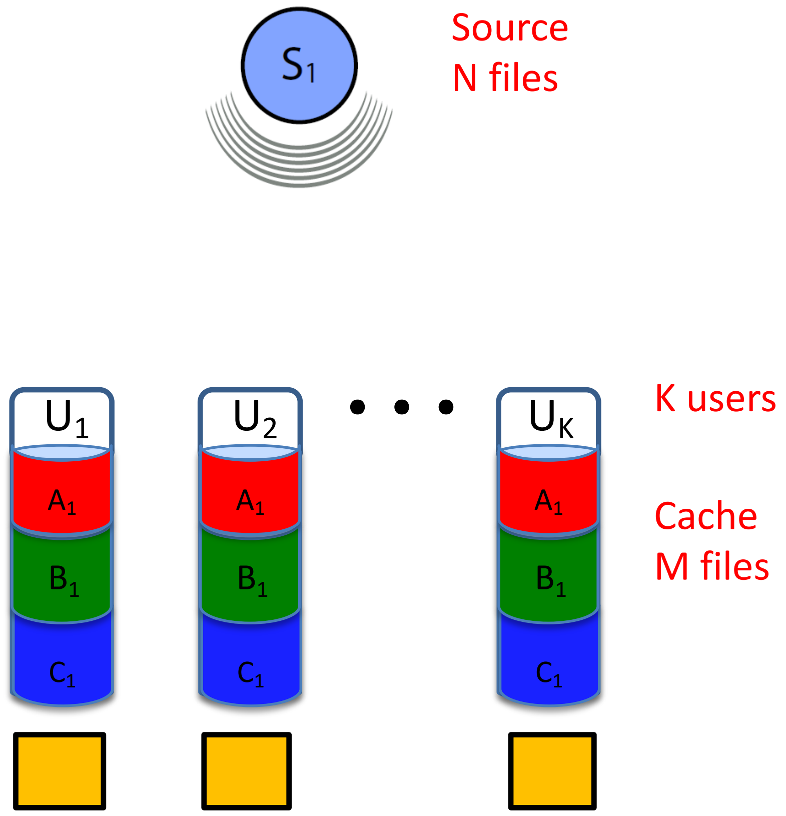


$$\mathcal{L}oad \simeq \frac{K(1-\mu)}{1+K\mu} \simeq \Theta(1/\mu) \simeq O(1)$$

Local caching gain

Global caching gain

Cache-Aided Coded Multicast



Normalized user's cache size

Think of $\mu = \frac{M}{N} = \frac{\text{cache size}}{\text{num. of files}}$ as a constant

Fractional Cooperative Caching (Cache Encoder)

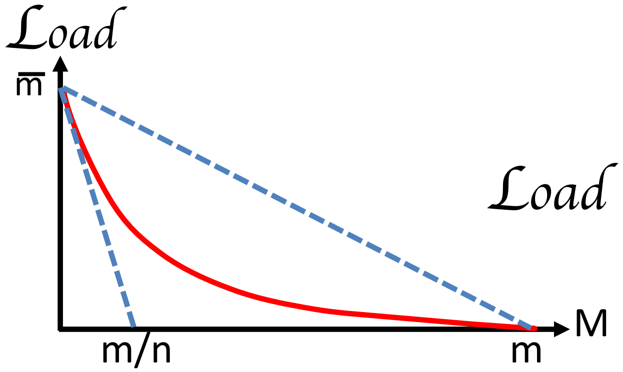
- Split files into F p... them strategically

Index Coding with a twist

Coded Multicast

- Coded multicast ...aneously serve multiple distinct requ... dex coding

In the relevant regime of KM ≫ N (i.e. Kμ ≫ 1)



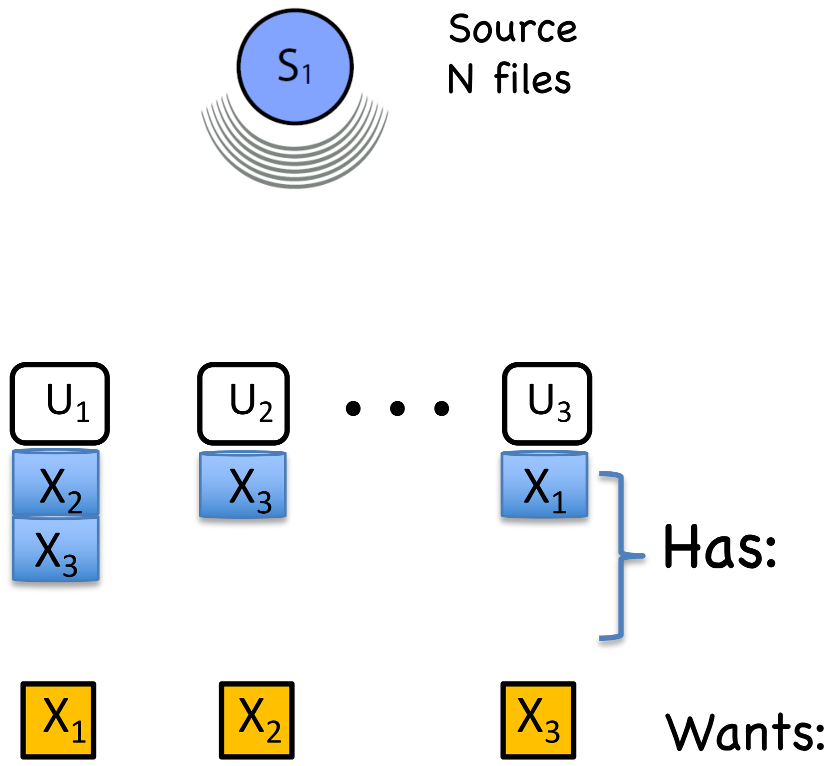
$$\mathcal{L}oad \simeq \frac{K(1-\mu)}{1+K\mu} \simeq \Theta(1/\mu) \simeq O(1)$$

Local caching gain

Global caching gain

(Index) Coding turns unicast traffic into multicast traffic

Index Coding

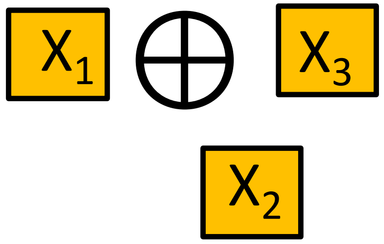


Source: Broadcasts to all users.

Each transmission is 1 file.

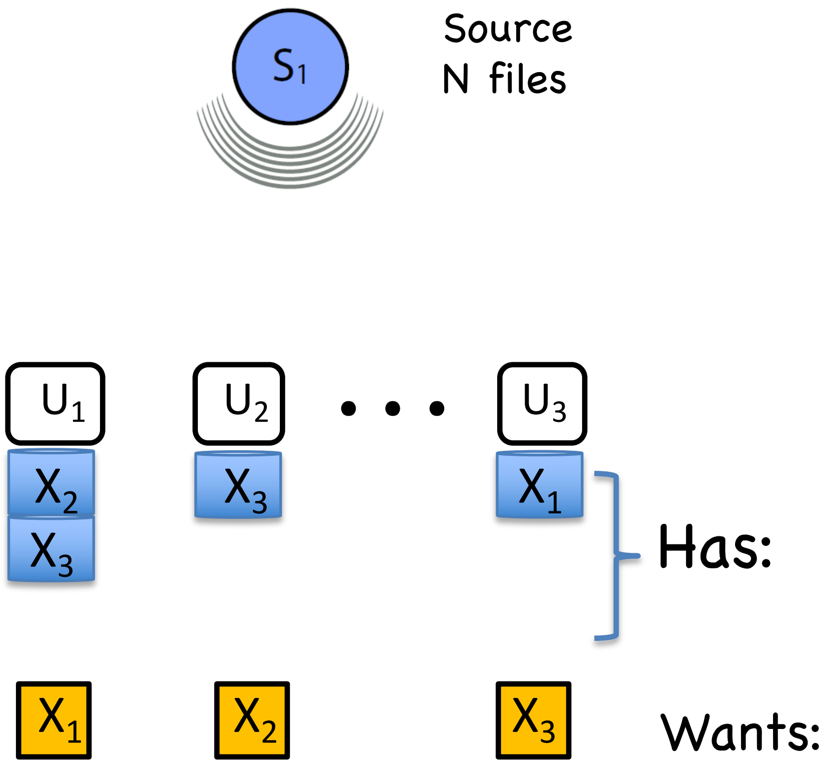
Side information allows savings

Minimum number of transmissions?



Graph Coloring solution

Index Coding



Source: Broadcasts to all users.

Each transmission is 1 file.

Side information allows savings

Minimum number of transmissions?

IC is a fundamental and challenging problem
(Birk & Kol'98; Bar-Yossef et al.; Alon et al.; El Rouayheb et al.; Effros et al.; Maleki et al.)

At the beginning...

- Maddah-Ali, and Niesen, 2012. "Fundamental limits of caching", ArXiv.
- J. Llorca, A.M. Tulino K. Guan, and D. Kilper, 2013 Network-coded caching-aided multicast for efficient content delivery", ICC.
- M. Ji, A. M. Tulino, J. Llorca, and G. Caire, 2014 "On the average performance of caching and coded multicasting with random demands." SWCS.

Over the years…

Several optimality results

- M. Maddah-Ali, and U. Niesen, TIT 2014]: order optimal under uncoded placement.
- K. Wan, D. Tuninetti, P. Piantanida, ITW 2016]: optimality under distinct demands $K \leq N$ and uncoded placement.
- M. Ji, A. M. Tulino, J. Llorca, and G. Caire, TIT 2017]: order optimal for arbitrary popularity distribution
- Q. Yu, M. A. Maddah-Ali, S. Avestimehr, TIT 2018]: optimal for uncoded placement.
- Q. Yu, M. A. Maddah-Ali, S. Avestimehr, TIT 2019]: optimal within a factor of 2 (no restriction on placement).

Over the years…

Several optimality results

- M. Maddah-Ali, and U. Niesen, TIT 2014]: order optimal under uncoded placement.
- K. Wan, D. Tuninetti, P. Piantanida, ITW 2016]: optimality under distinct demands K ≤ N and uncoded placement.
- M. Ji, A. M. Tulino, J. Llorca, and G. Caire, TIT 2017]: order optimal for arbitrary popularity distribution
- Q. Yu, M. A. Maddah-Ali, S. Avestimehr, TIT 2018]: optimal for uncoded placement.
- Q. Yu, M. A. Maddah-Ali, S. Avestimehr, TIT 2019]: optimal within a factor of 2 (no restriction on placement).

Gains of CCM unbounded for uniform distribution, M/m=1/10, n=1000 users, only 10 transmissions!

$$\mathcal{L}oad \simeq \frac{K(1-\mu)}{1+K\mu}$$

Think of $\mu = \frac{M}{N} = \frac{\text{cache size}}{\text{num. of files}}$ as a constant

Normalized per user cache size

BUT Still very far from achieving these gains because of two main technical barriers

Technical Barriers

Think of $\mu = \frac{M}{N} = \frac{\text{cache size}}{\text{num. of files}}$ as a constant

Gains of CCM theoretical unbounded

$$\mathcal{L}oad \simeq \frac{K(1-\mu)}{1+K\mu}$$

BUT Still very far from achieving these gains because of two main technical barriers

- **Coding Complexity**
  - Number of packets grows exponentially with number of caches.
  - How should F scale as a function of M,m,n to get these gains?

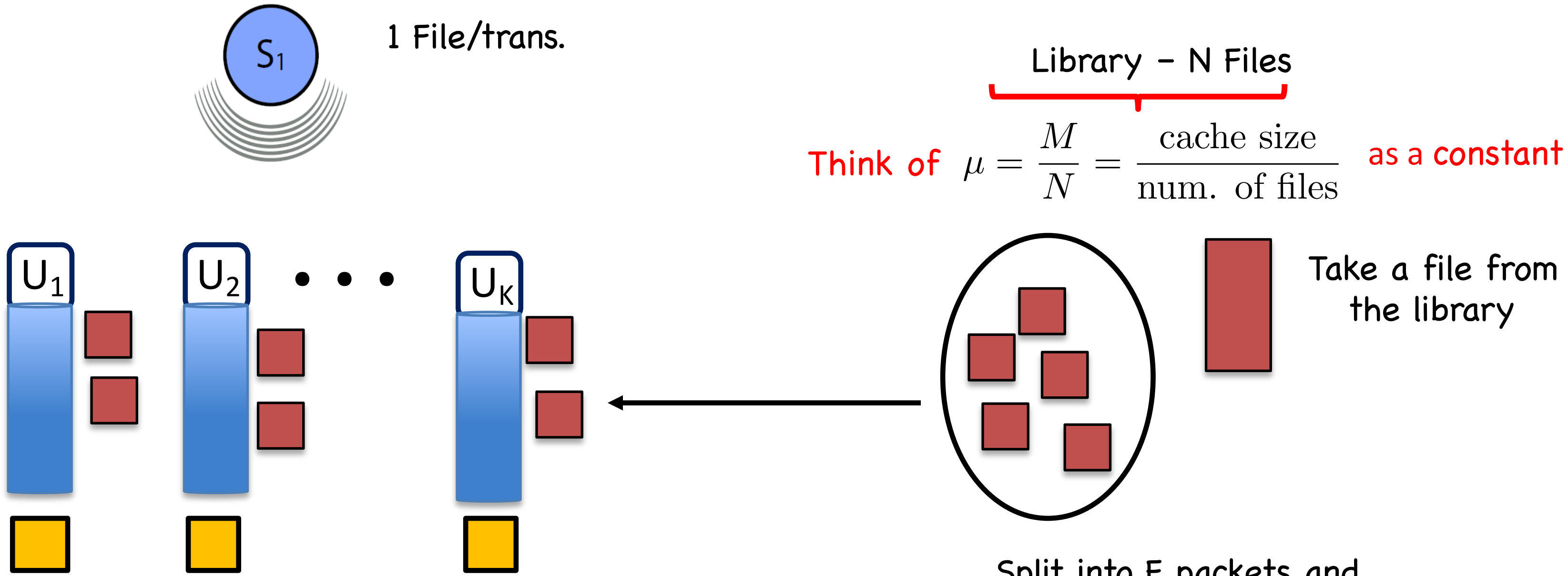
- **Heterogeneous Channels**
  - Different caches have different channels: worst cache channel dictates the overall performance
  - How to include channel coding in order to maintains the gains.

Technical Barriers

- Coding Complexity

Think of $\mu = \frac{M}{N} = \frac{\text{cache size}}{\text{num. of files}}$ as a constant

  - How should F scale as a function of M,K,N to get these gains?



Think of $\mu = \frac{M}{N} = \frac{\text{cache size}}{\text{num. of files}}$ as a constant

Split into F packets and place strategically + XOR packets during delivery.

Key Question: How large F needs to be ?

$$F = \exp\left(K f(\mu)\right) = \exp\left(\Theta(K)\right)$$

all original schemes number of packets grows exponentially with number of caches

Coding Complexity

Think of $\mu = \frac{M}{N} = \frac{\text{cache size}}{\text{num. of files}}$ as a constant

Centralized

Level of cache coordination

$F = \exp\left(K f(\mu)\right) = \exp\left(\Theta(K)\right)$ [all schemes up to 2016]

$F = \exp\left(K f''(\mu)\right) = \exp\left(\Theta(K)\right)$ [Tang-Ramamoorthy '17, Yan *et al* '16]

Very practical schemes Exponentially smaller !!

$F = \exp\left(\sqrt{K} f''(\mu)\right) = \exp\left(\Theta(\sqrt{K})\right)$ [Yan *et al* '16, Shangguan *et al* '16]

$F = K, \mu \geq K^{-\delta(\epsilon)}$ [Shanmugam, Tulino, Dimakis 2017]

Load = O(1) , then F = K is impossible !!

Caching gain = K

Load $= O\left(\frac{K}{K\mu}\right) = O(1)$

Caching gain = $K^{1-\varepsilon}$

Load $\leq K^{\epsilon}$

All these results are about constructions of RUZSA-SZEMÉREDI bipartite graphs

$F = \exp\left(g f'(\mu)\right)$ [Hachem *et al* '17], [Lampiris *et al* '18], [Parrinello *et al* '18] [Jin, Cui, Liu, and Caire. TC, 2019]

PHY: leveraging spatial multiplexing

users are grouped

Load $= O\left(\frac{K}{K\mu}\right) = O(1)$

Load $= O\left(\frac{K}{g}\right)$

[Shanmugam, Ji, Tulino, Llorca Dimakis 2016]

$F = \exp\left(K f(\mu)\right) = \exp\left(\Theta(K)\right)$

Caching gain = K

$F = \exp\left(g f'(\mu)\right) = O(\mu^{g})$

Caching gain = g

Distributed

Technical Barriers

Think of $\mu = \frac{M}{N} = \frac{\text{cache size}}{\text{num. of files}}$ as a constant

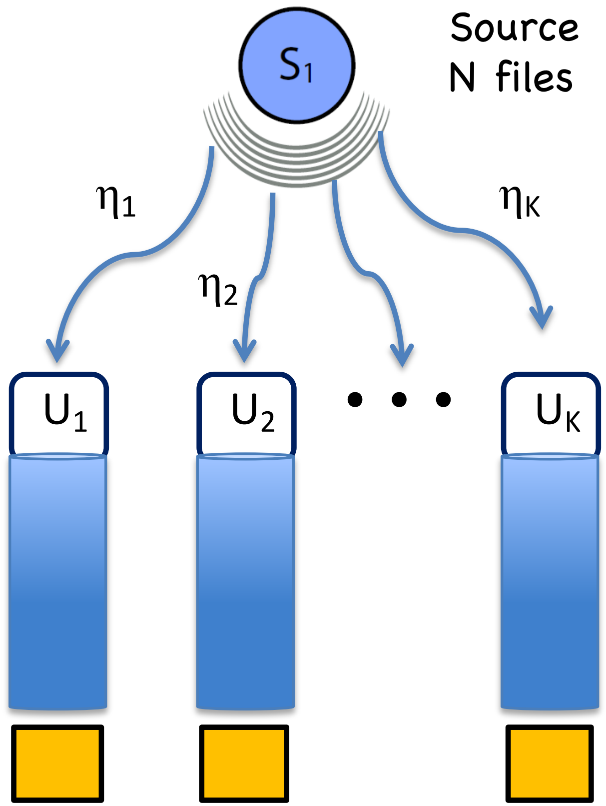
Gains of CCM theoretical unbounded

$$\mathcal{L}oad \simeq \frac{K(1-\mu)}{1+K\mu}$$

BUT Still very far from achieving these gains because of two main technical barriers

- **Coding Complexity**
  - Number of packets grows exponentially with number of caches.
  - How should F scale as a function of M,m,n to get these gains?
- **Heterogeneous Channels**
  - Different caches have different channels: worst cache channel dictates the overall performance
  - How to include channel coding in order to maintains the gains.

Heterogeneous Channels

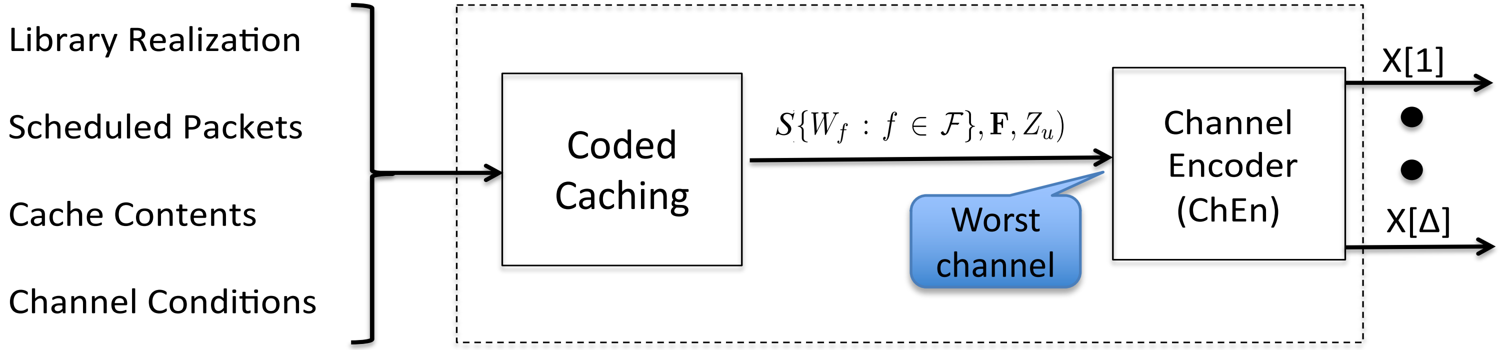


η_u channel rate of user u

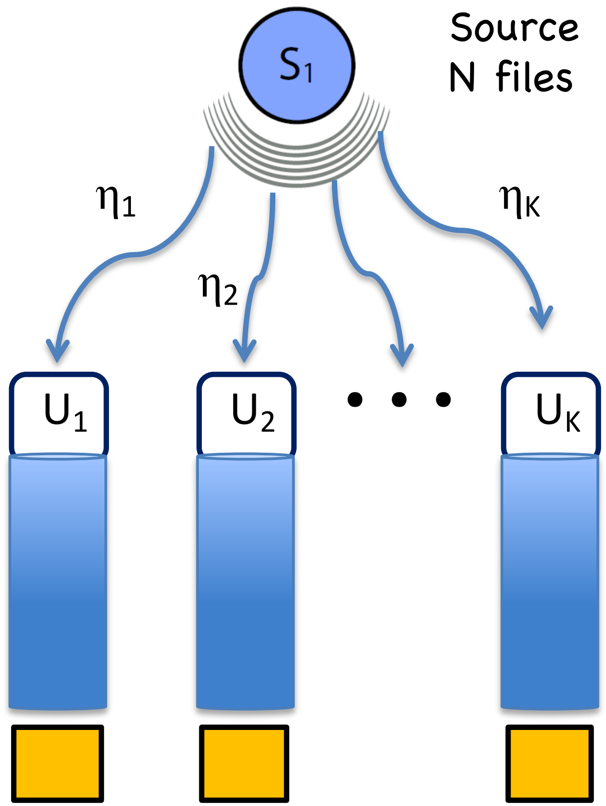
$\eta_u = \eta$ =common channel rate

Separation Source-Channel Coding theorem:

Achievable rate = η / $\mathcal{L}oad$



Heterogeneous Channels

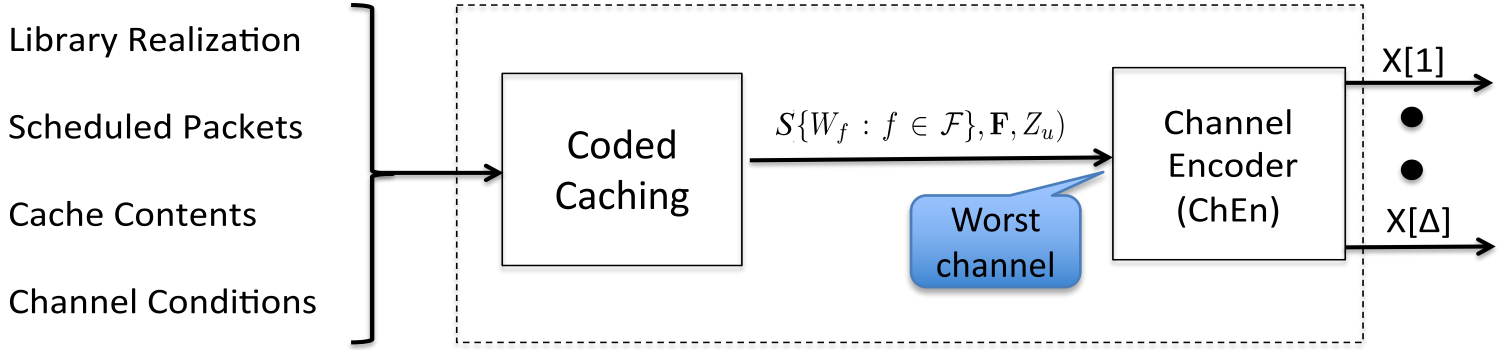


η_u channel rate of user u

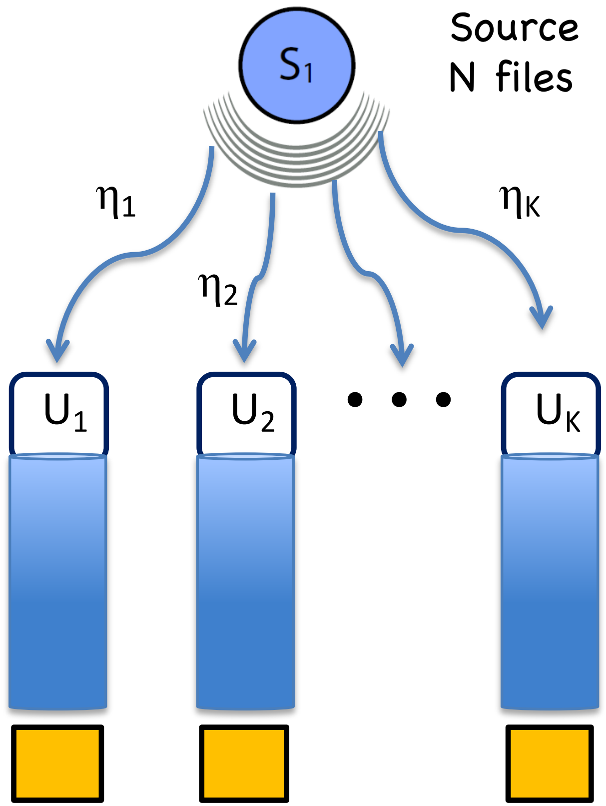
η_u channel rate different across users

Separation Source-Channel Coding theorem:

Achievable rate = η_{min} / $\mathcal{L}oad$



Heterogeneous Channels

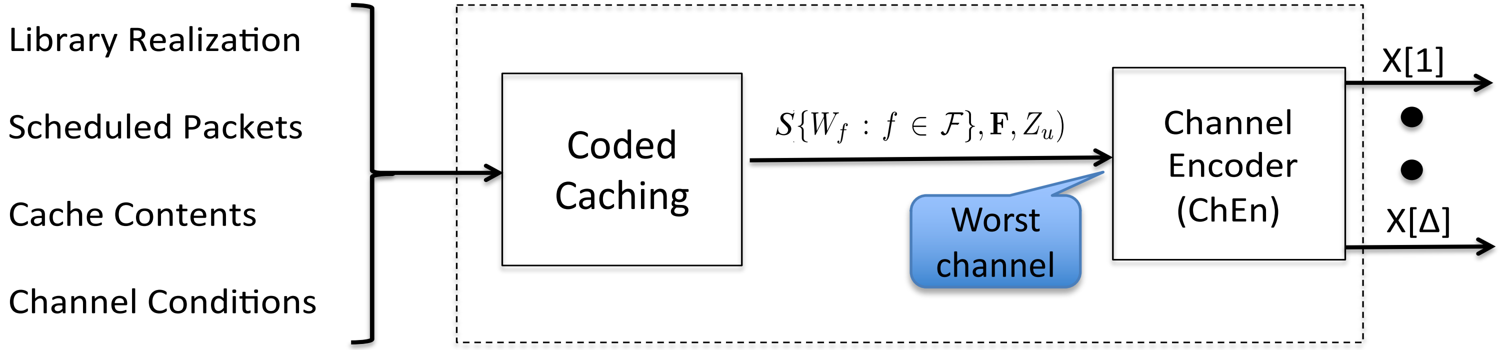


η_u channel rate of user u

To improve performance, need for joint source-channel coding scheme

Separation Source-Channel Coding theorem:

Achievable rate = η_{min} / $\mathcal{L}oad$



Heterogeneous Channels

✓ Two Caches [Asadi-Ong-Johnson, 2015]

- Capacity-memory trade off of two cache-aided receiver broadcast channel.
- Each receiver side information is part of the private message of the other.

✓ Multiple Caches divided in two classes:

Special settings

- [Karamchandani-Diggavi-Caire-Shamai, 2016]
  - Two links (1 & 2) between caches and source.
  - One class receiving only from link 1 the other from both links cache size M.
- [Bidokhti-Wigger-Timo, 2016]
  - Weak receivers with equal "large" BC erasure probabilities and cache size M.
  - Strong receivers with equal "small" BC erasure probabilities with zero cache-size.
  - This especially useful in a designing phase for dimensioning the caches

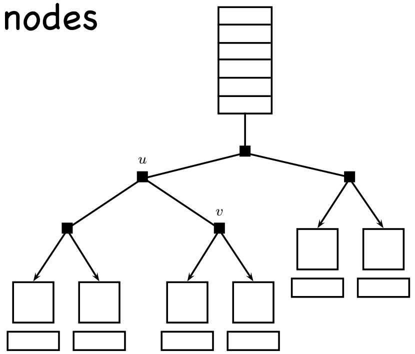
✓ General Setting [Cacciapuoti-Caleffi-Ji-Llorca-Tulino, 2016]

- Channel, cache size, demand distribution, number of requested files arbitrary across users
- Random Fractional Caching
- Channel-Aware Chromatic Index Coding

Extension to different network topologies

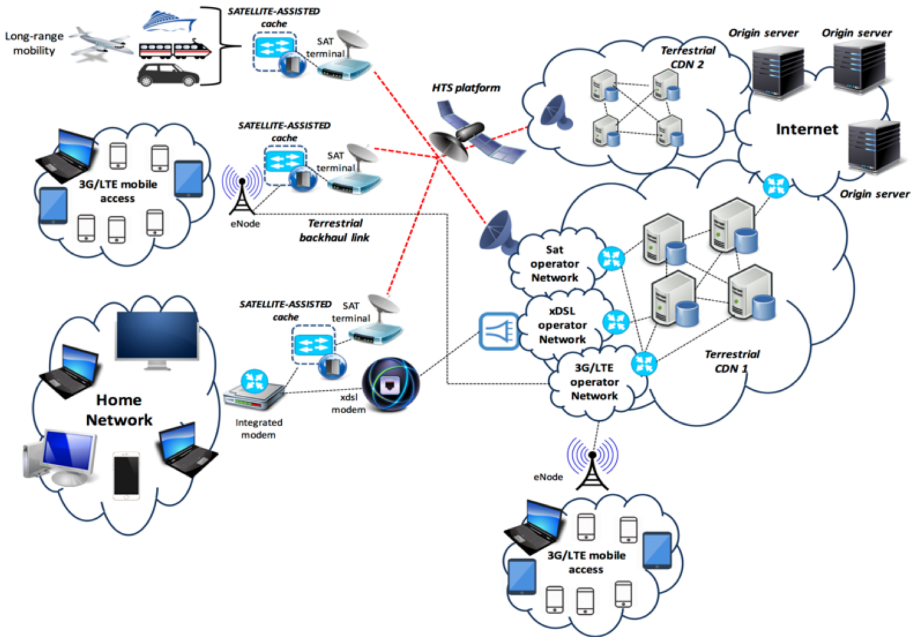
Tree Topology:

CM with routing at intermediate nodes

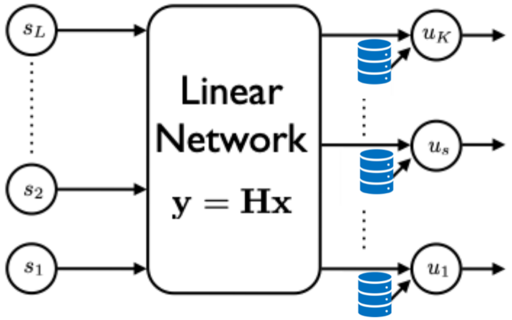


SHINE

(Secure Hybrid In Network caching Environment)

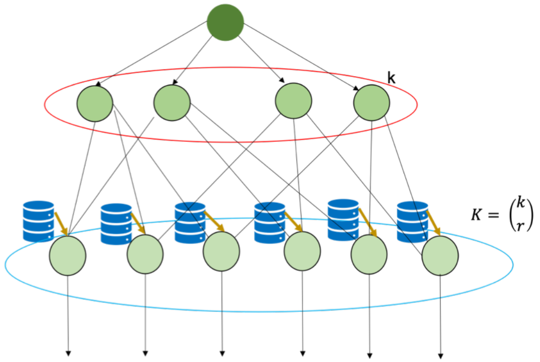


Multiserver/linear network



Shared Caches

Combination network



Combination network

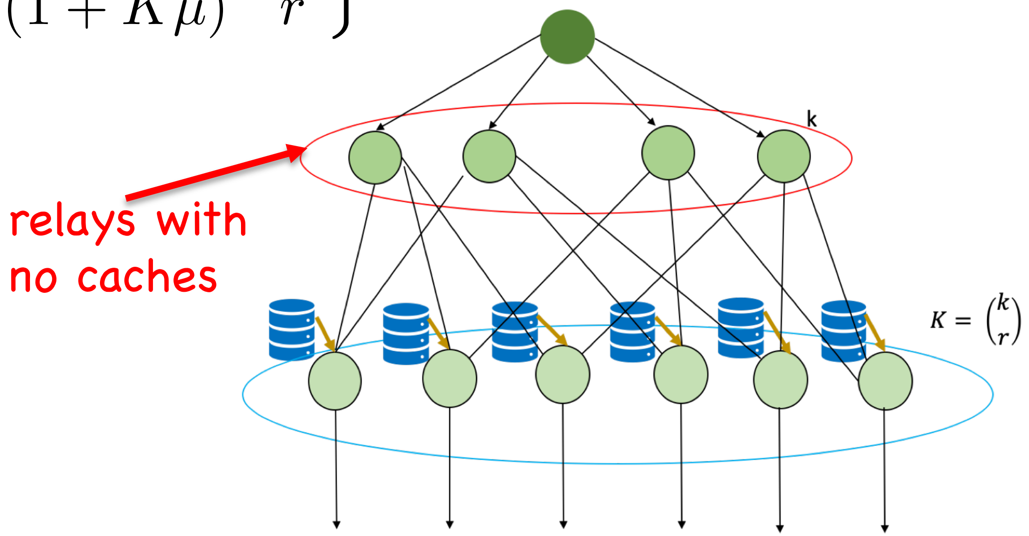
- Ji, M., Wong, M.F., Tulino, A.M., Llorca, J., Caire, G., Effros, M. and Langberg, M., IEEE SPAWC 2015 .
- M. Ji, A. M. Tulino, J. Llorca, G. Caire, *IEEE ASILOMAR*, 2015
- Kai Wan, Daniela Tuninetti, Mingyue Ji, and Pablo Piantanida, *IEEE ASILOMAR,* 2017

Simple achievable scheme: concatenation of classical Cache-Aided Coded Multicast (CCM) and MDS coding combined with naive multicasting of all the library and routing (naive unicast), is given by:

$$\text{Maximum link load} = \mathcal{L}oad \simeq \min\left\{\frac{K}{k}(1-\mu), \frac{K(1-\mu)}{r(1+K\mu)}, \frac{N}{r}\right\}$$

not optimal BUT completely topology-agnostic.

Recently extensions with caches at the relays

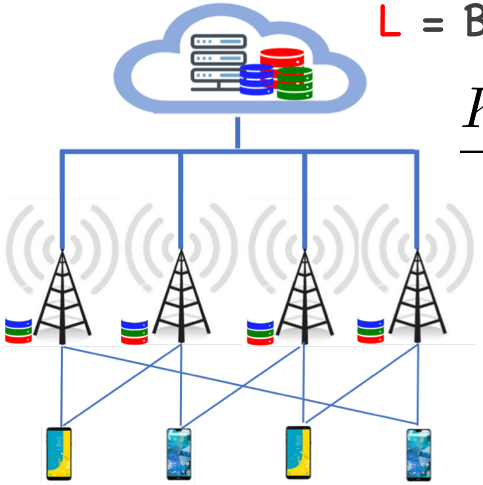


Shared Caches

- Hachem, Karamchandani, Diggavi, TIT 63(5), 2017,
- G. Vettigli, M. Ji, K. Shanmugan, J. Llorca, A. Tulino, G. Caire, MDPI Entropy, March 2019
- Parrinello, Unsal and Elia, arXiv:1809.09422, : 2018

The goal is to minimize the worst-case load over the shared link (backhaul).

Each user receives from L distinct BSs



L = BSs serving each user

$$\frac{K(1-L\mu)}{1+K\mu}$$

Each user receives from one BS with N_0 antennas

number users served by each BS $\geq N_0$

L = Number of BSs

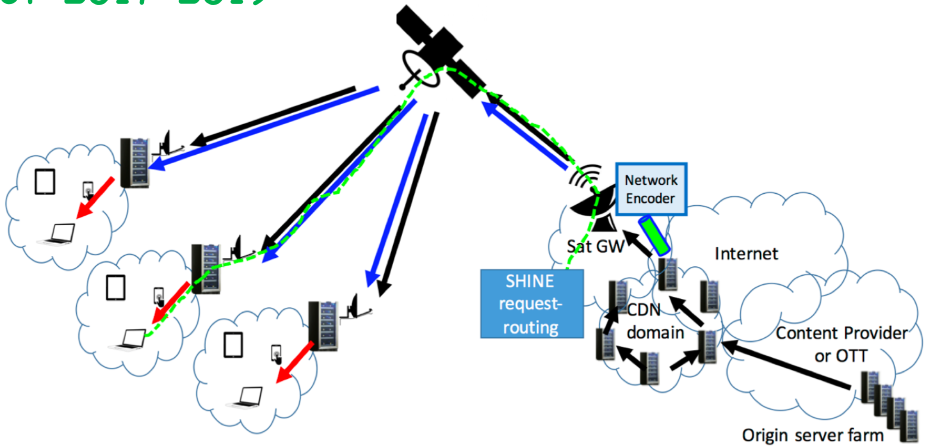
$$\frac{K(1-\mu)}{N_0(1+L\mu)}$$

Interplay between shared caches and multiple antennas:
- adding 1 degree of cache-redundancy increases a DoF to N0,
- going from 1 to No antennas reduces delivery time by N0.

SHINE

Secure Hybrid In Network caching Environment

S. P. Romano, C. Roseti, A. M. Tulino, ISNCC, 2018
SHINE: Secure Hybrid In Network caching Environment, ESA Project 2017-2019



Goal:
E2E secure delivery of multimedia content over integrated satellite-terrestrial cache-aided networks.

Combination of both unicast and network-coded multicast
Two main building blocks:

- a satellite-enabled broadcast distribution backbone leveraging the CCM in order to improve both performance and security of the transmissions;
- a MPEG-DASH/WebRTC-enabled edge distribution network.

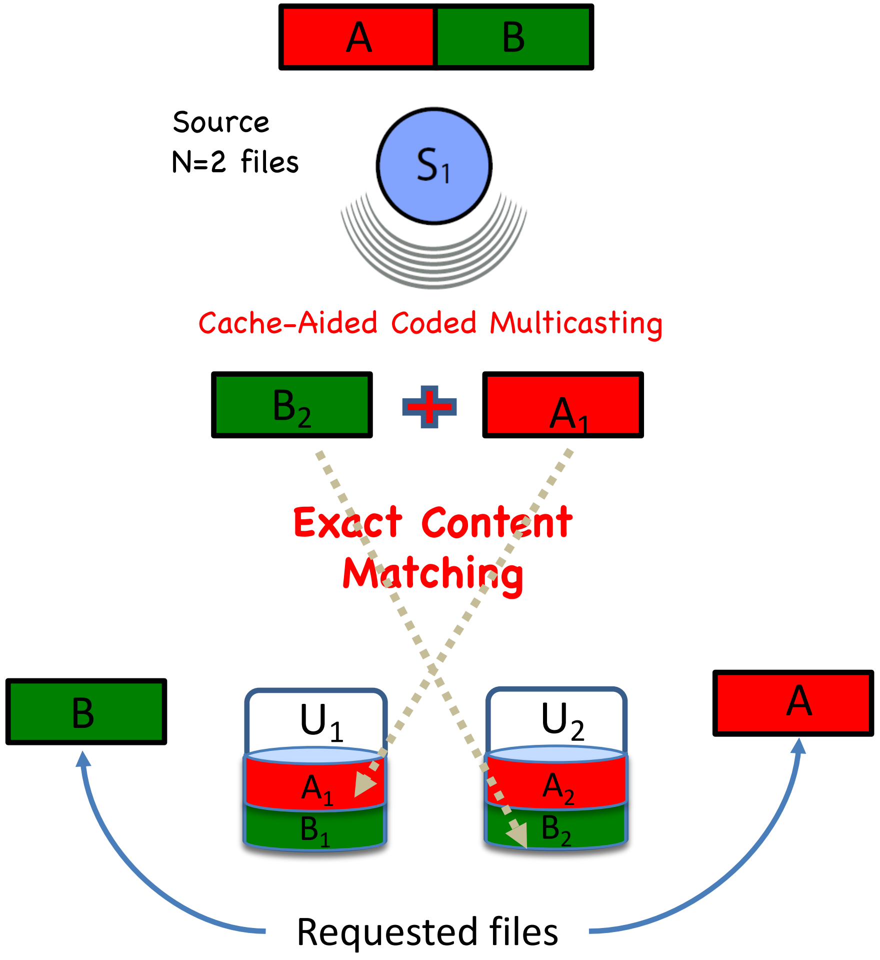
(i) relying cache-aided coded multicast to improve both performance and security of communications.

(ii) leveraging cutting-edge streaming technologies (MPEG-DASH WebRTC) to optimize E2E content distribution

Dynamic Network Compression

So far…

used previously in-network stored exact copies of the information that need to be delivered as references for network compression during delivery



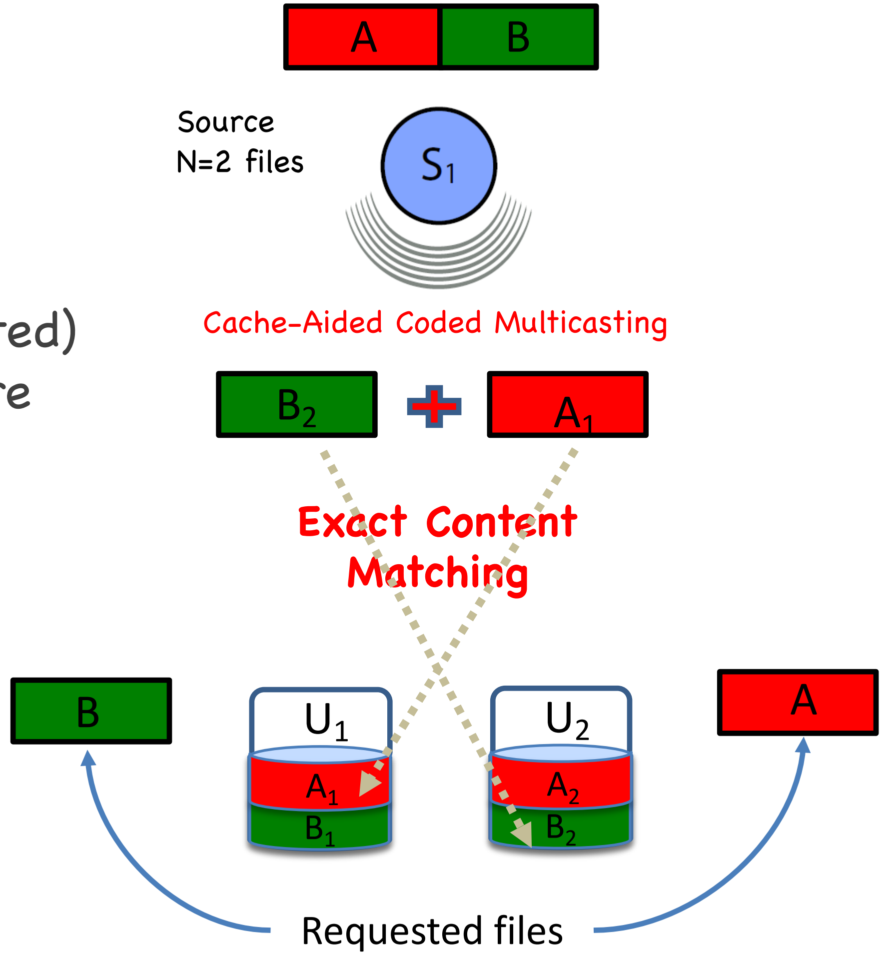
Dynamic Network Compression

So far…

used previously in-network stored exact copies of the information that need to be delivered as references for network compression during delivery

BUT

Moving towards real-time (personalized media dominated) services exact cache hits are almost non-existent.



What about
Approximate Content
Matching
(e.g. correlation)

Updated versions of dynamic data can exhibit high levels of correlation

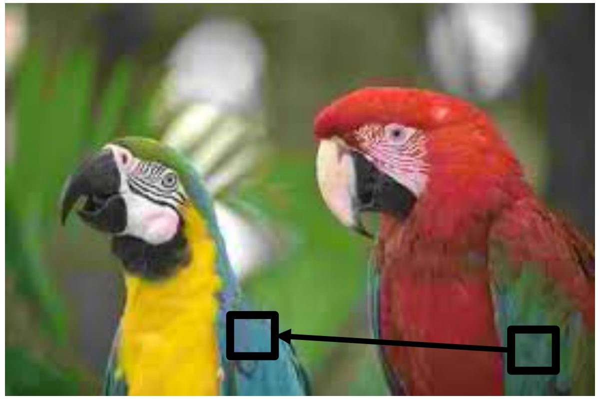
Dynamic Network Compression

Compressing information as it travels through the network

FROM STATIC LOCAL COMPRESSION TO DYNAMIC NETWORK COMPRESSION

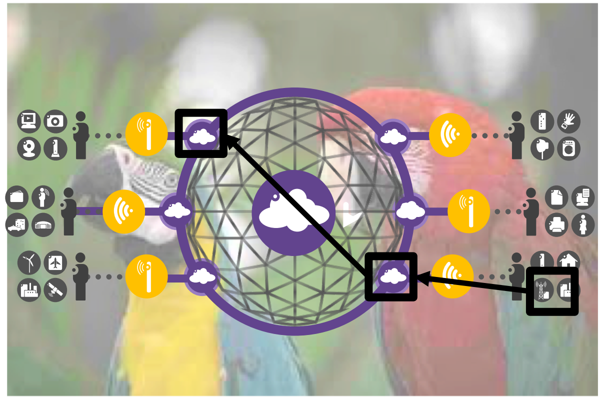
Static local compression is myopic to spatiotemporal information lifecycle

We still compress information based solely on local intra-file correlations, without taking into account increasingly relevant network-wide spatiotemporal correlations



Dynamic e2e compression adaptively exploits redundancy throughout the network

Exploiting cloud network wide spatiotemporal redundancy to push the fundamental limits of information compression



Previously stored information are exploited as references for network compression during delivery

Towards dynamic E2E network compression

Cache-Aided Coded Multicast with Correlated library

[Timo, Bidokthi, Wigger and Geiger TIT'18]:

- Lossy reconstruction.
- Two receivers and one cache, no coded multicasting.

[Op 't Veld and Gastpar ISIT'17]:

- Lossy reconstruction Gaussian sources.
- Distortion-rate-memory region two files.

[Yang and Gunduz ICC'18]:

- Specific correlation structure.
- Worst-case rate-memory trade-off.

[Hassanzadeh, Tulino, Llorca, Erkip, ITW'2016, TIT'20]

- Lossless reconstruction.
- Arbitrary correlated sources.
- Dynamic content.
- General system parameters, prove optimality in some cases.

Towards dynamic E2E network compression

Cache-Aided Coded Multicast with Correlated library

- Library Compression Approach
  - Two step approach:
    - Step 1: Sender jointly compresses the library.
      - Gray-Wyner source-coding.
    - Step 2: Correlation-unaware caching and coded multicast.
      - Multiple-request scheme.

- On-demand Compression Approach
  - Store individually compressed.
  - Deliver jointly compressed

Towards dynamic E2E network compression

Cache-Aided Coded Multicast with Correlated library

- Library Compression Approach
  - Two step approach:
    - Step 1: Sender jointly compresses the library.
      - Gray-Wyner source-coding.
    - Step 2: Correlation-unaware caching and coded multicast.
      - Multiple-request scheme.
  - **Effective for Static Library**
- On-demand Compression Approach
  - Store individually compressed.
  - Deliver jointly compressed

Towards dynamic E2E network compression

Cache-Aided Coded Multicast with Correlated library

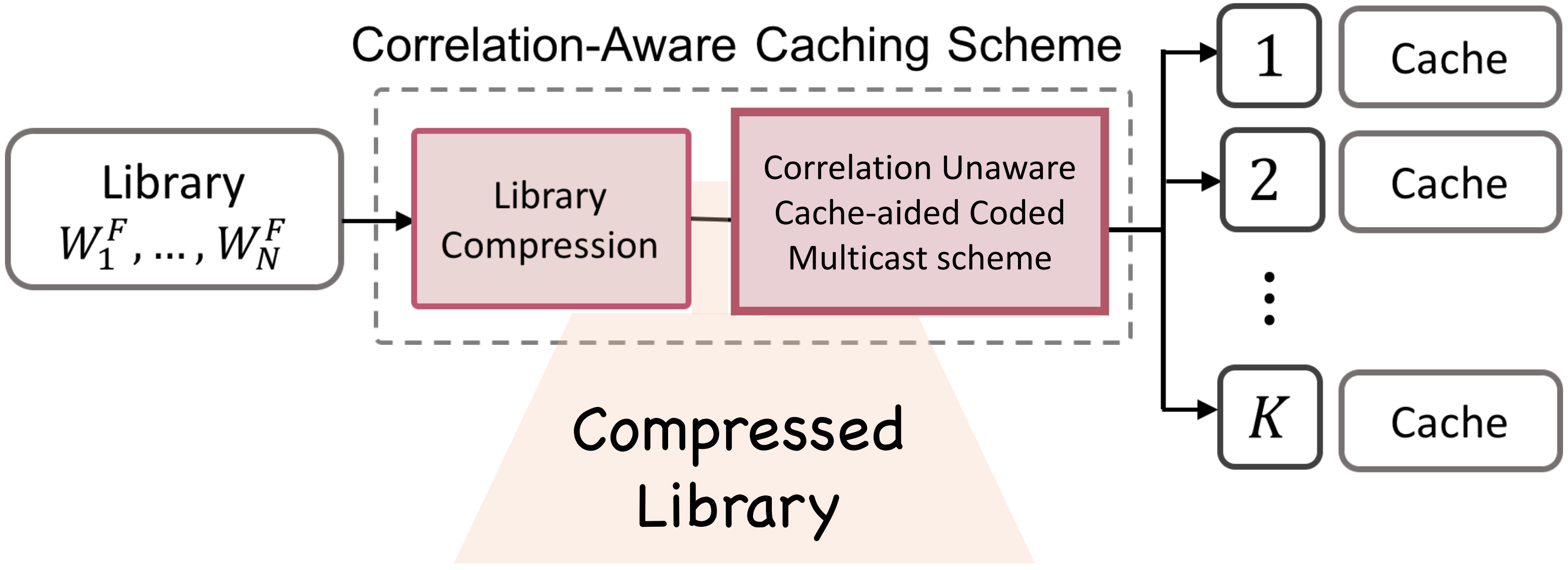
- Library Compression Approach
  - Two step approach:
    - Step 1: Sender jointly compresses the library.
      - Gray-Wyner source-coding.
    - Step 2: Correlation-unaware caching and coded multicast.
      - Multiple-request scheme.
- On-demand Compression Approach
  - Store individually compressed.
  - Deliver jointly compressed

- **Effective for Dynamic Library**

Towards dynamic E2E network compression

Cache-Aided Coded Multicast with Correlated library

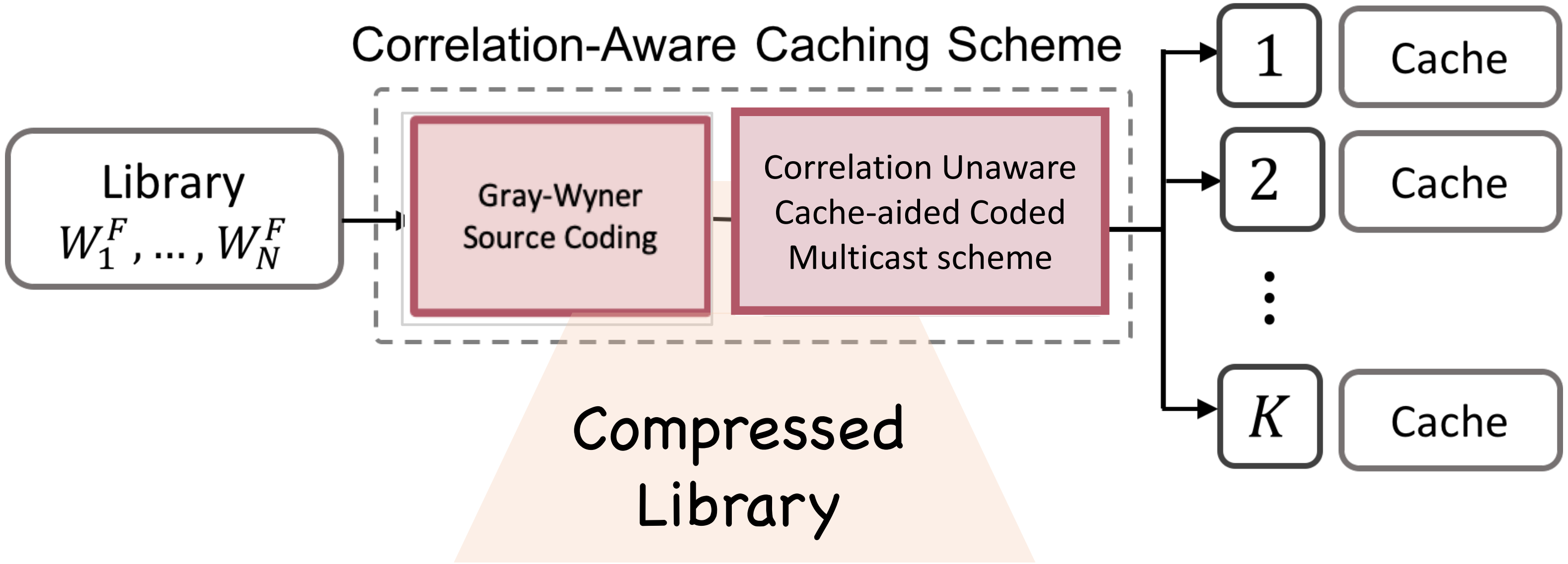
- **Library Compression Approach (two step approach):**
  - First compress the library
  - Then apply a correlation unaware CCM (Cache-aided Coded Multicast) scheme which assume independent files and consisting of
    - a cache phase (to populate caches)
    - a delivery phase



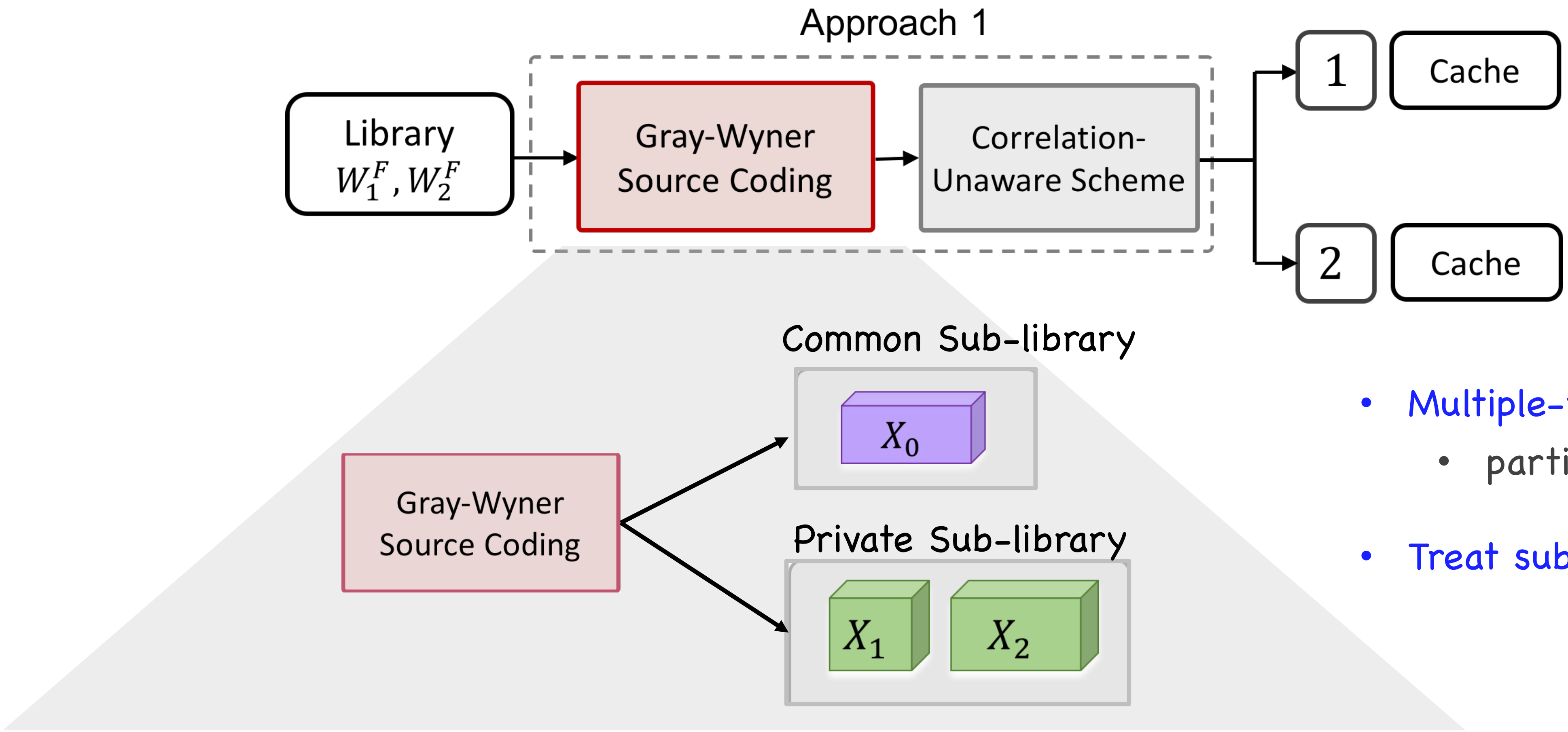
Towards dynamic E2E network compression

Cache-Aided Coded Multicast with Correlated library

- **Library Compression Approach (two step approach):**
  - First compress the library
  - Then apply a correlation unaware CCM (Cache-aided Coded Multicast) scheme which assume independent files and consisting of
    - a cache phase (to populate caches)
    - a delivery phase



Example two files

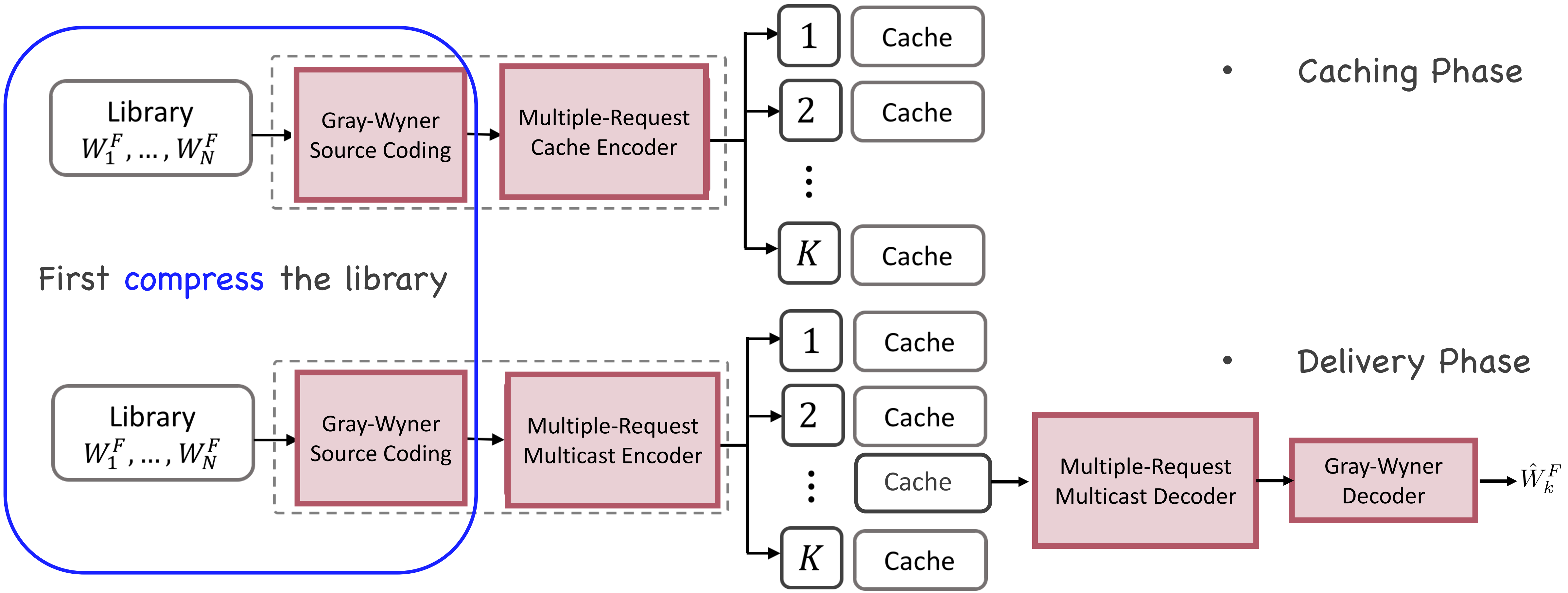


- Multiple-request scheme:
  - particular demand.
- Treat sublibraries independently.

Towards dynamic E2E network compression

Cache-Aided Coded Multicast with Correlated library

- Library Compression Approach (two step approach):

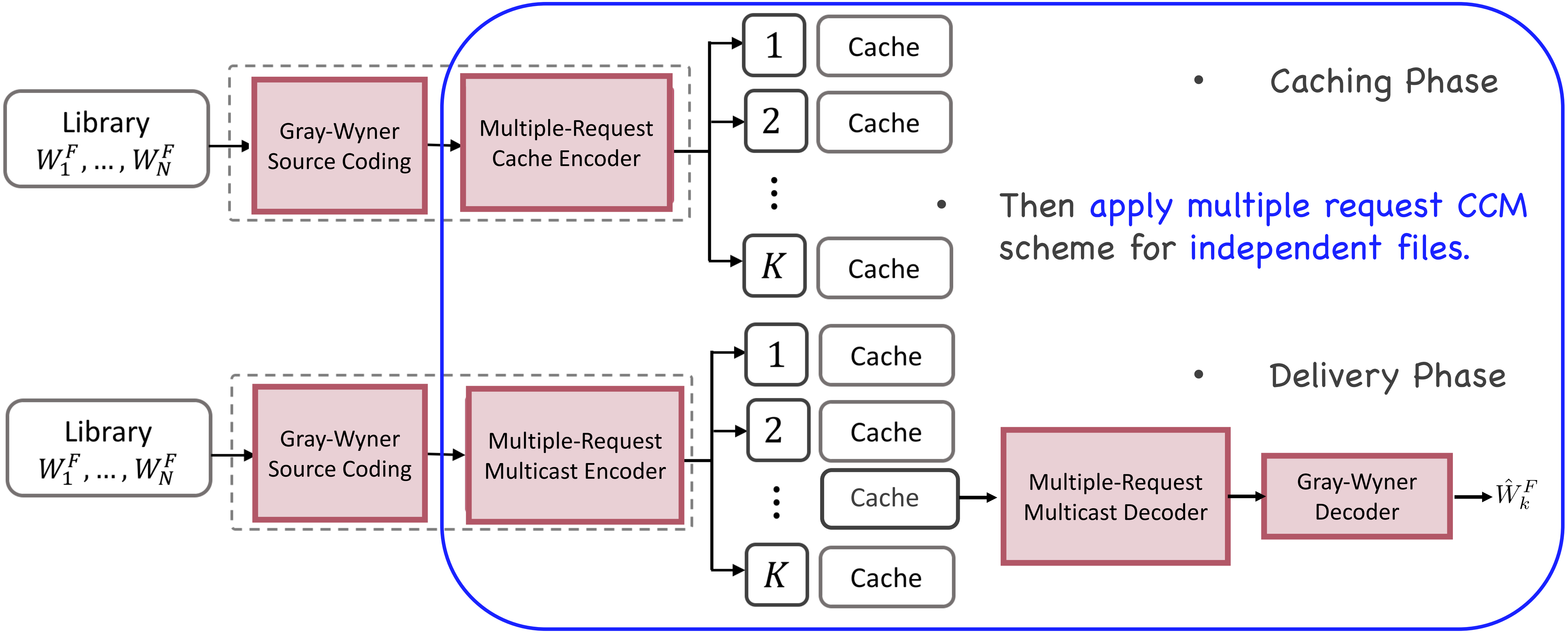


- First compress the library
- Caching Phase
- Delivery Phase

Towards dynamic E2E network compression

Cache-Aided Coded Multicast with Correlated library

- Library Compression Approach (two step approach):



- Caching Phase
- Then apply multiple request CCM scheme for independent files.
- Delivery Phase

Library Compression Approach

Optimality Results:

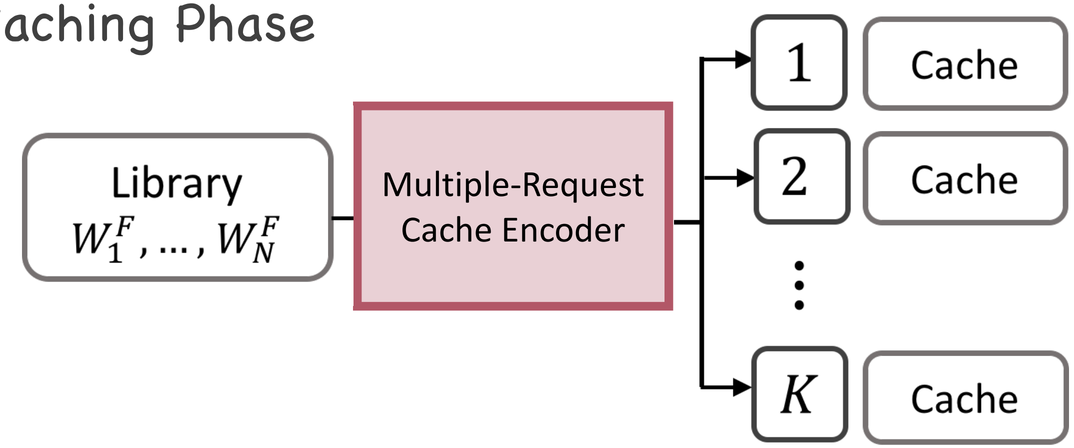
- Two files and K users:
  - Optimal for small and large memory.
  - Half of the conditional entropy of files elsewhere.
- Two files and two users:
  - Optimal over a larger region.
  - Optimal for special source.
- Extension to three files:
  - Optimal for large memory.
  - Half of $H(W_1, W_2 \mid W_3)$ elsewhere.
- Lower bound on the optimal load-memory trade-off.

Shortcomings of this Approach

- Not robust to system dynamics: a new file is added.
  - Jointly re-compressed entire library.
  - Update receiver caches.
- General setting with multiple files and receivers.

On-demand Compression Approach

- Caching Phase

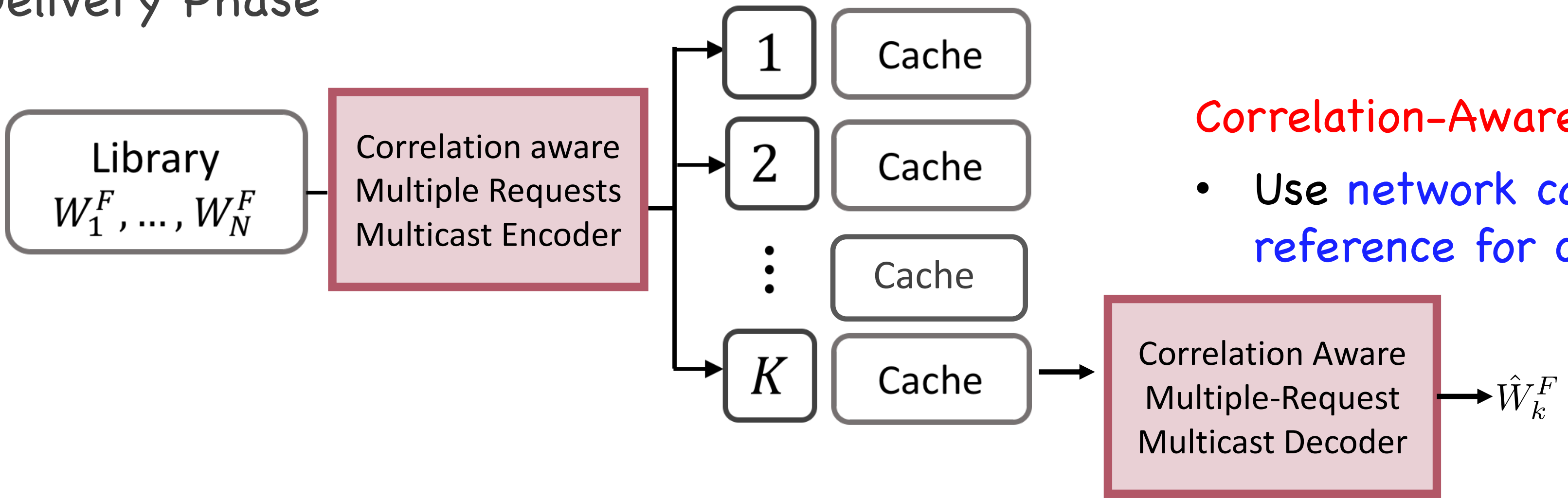


Correlation-Aware Cache Encoder.

- Divide each file into equal-size packets.
- Cache based on correlations and popularity.

Very Efficient in Dynamic content services.

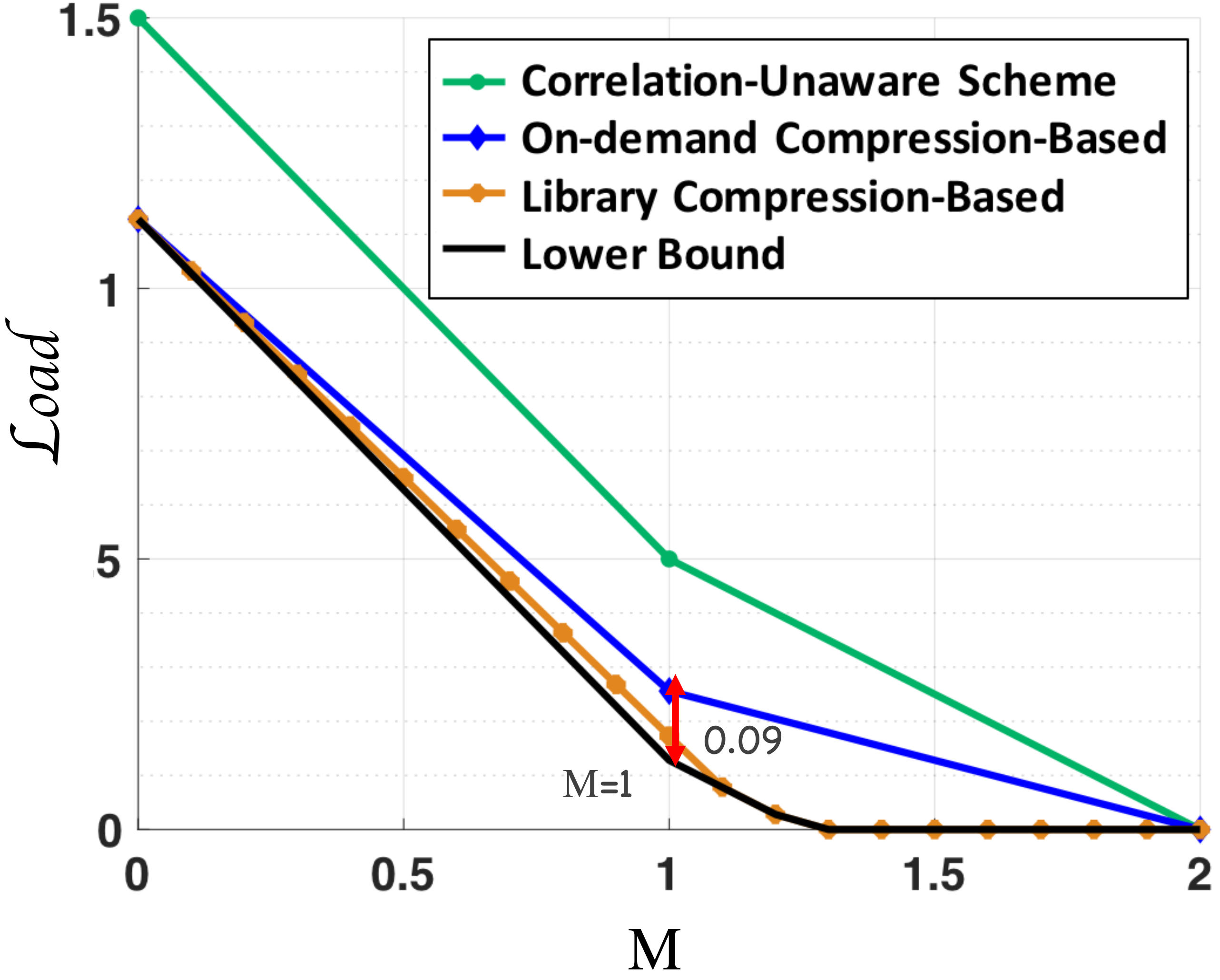
- Delivery Phase



Correlation-Aware Multicast Encoder

- Use network cached information as reference for compression during delivery.

Cache-Aided Coded Multicast with Correlated library

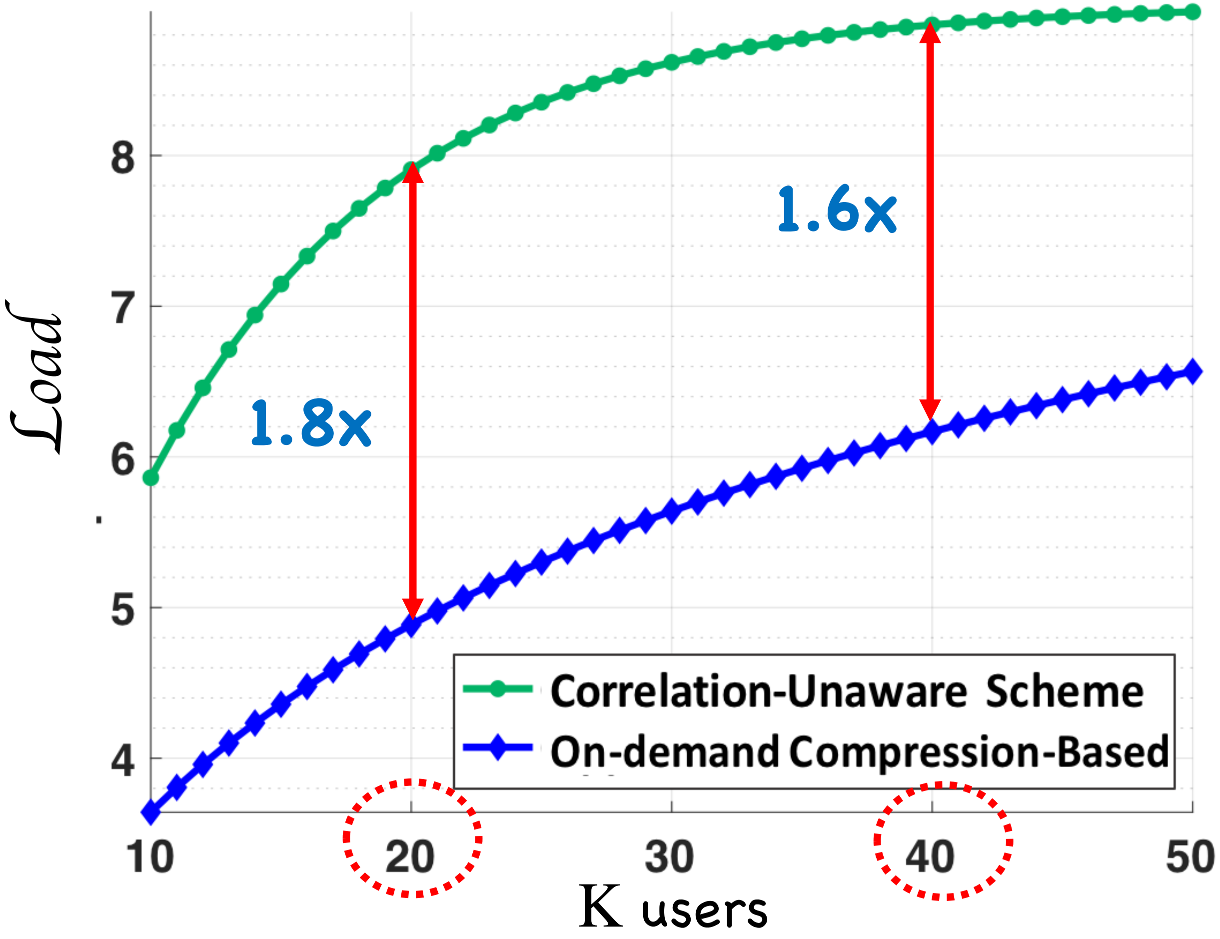


- Static library.
- Two files and two receivers.
- Deterministic cache placement.

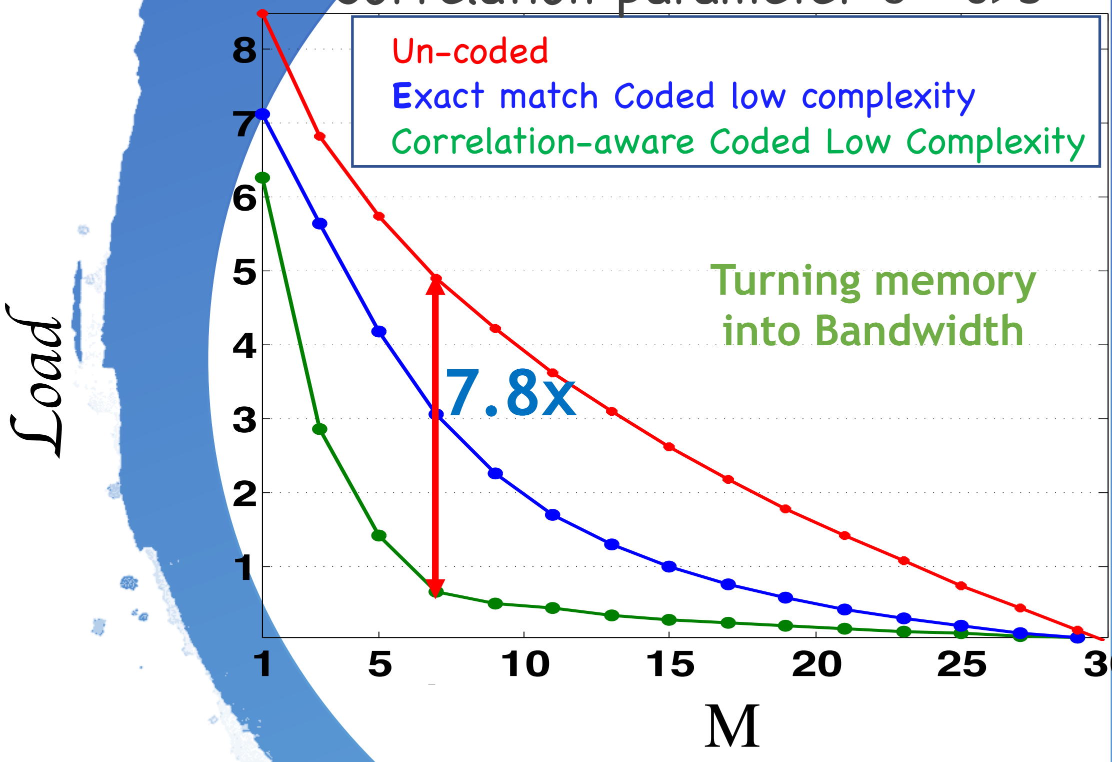
Cache-Aided Coded Multicast with Correlated library

Performance assessments

N = 1000 files.
Cache size M = 0.1× library size. Correlation parameter δ = 0▷3



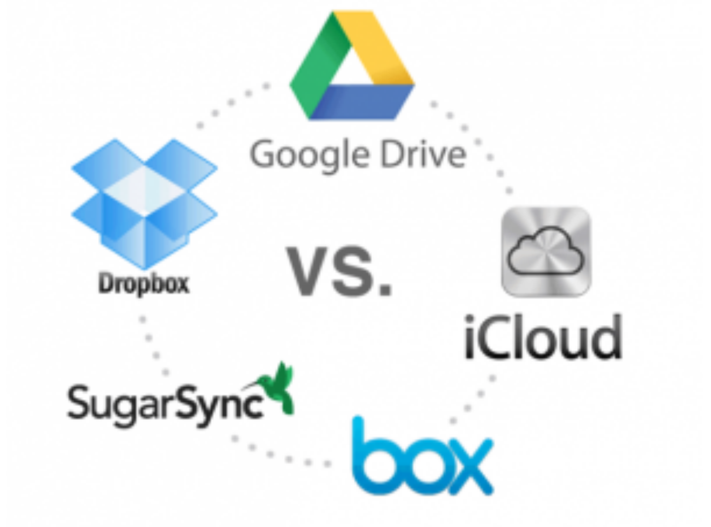
N = 30 files
K= 10 users
Correlation parameter δ = 0▷3



Efficient Storage of Dynamic Data in Distributed Clouds

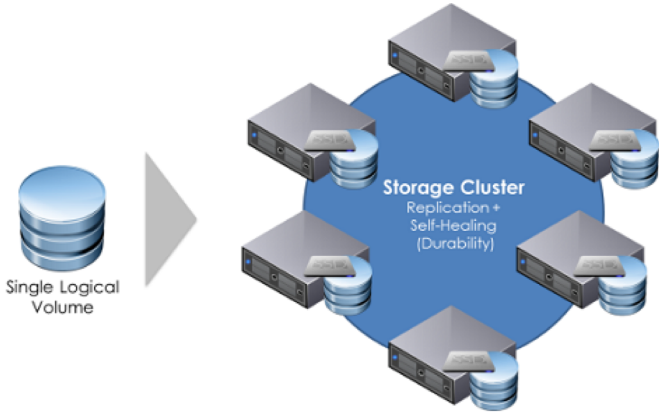
Rapid access to fresh and consistent data without costly replication

[Wang and Cadambe, TIT’14], [Ali, Cadambe, Llorca, Tulino, TC’20]



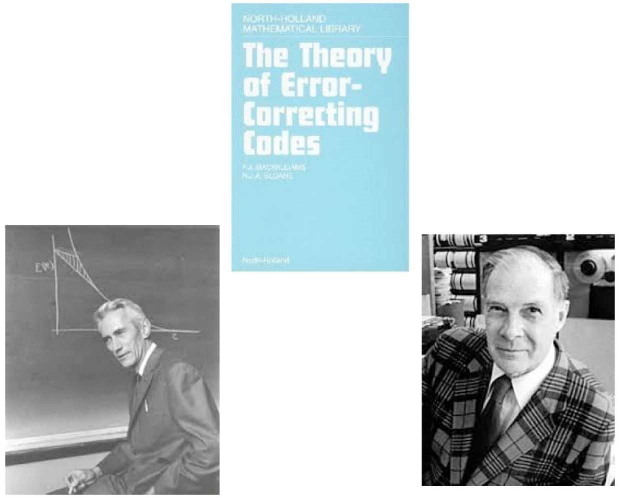
BIG CHALLENGE

Extend the benefits of distributed cloud storage (low latency access, robustness to failures) to highly dynamic applications, where the main challenges are data freshness and consistency



BASELINE

Existing systems don’t use coding and end up unnecessarily keeping old versions to ensure consistency via replication (e.g., Microsoft Azure) leading to unbearable cloud resource usage, specially for highly dynamic data.



BREAKTHROUGH

Holistic analytical understanding of the fundamental trade-offs between consistency, freshness, storage cost, and access latency. Efficient codes able to approach such fundamental trade-offs.

A NOVEL INFORMATION THEORETIC FRAMEWORK FOR CONSISTENT DELIVERY OF FRESH DYNAMIC DATA

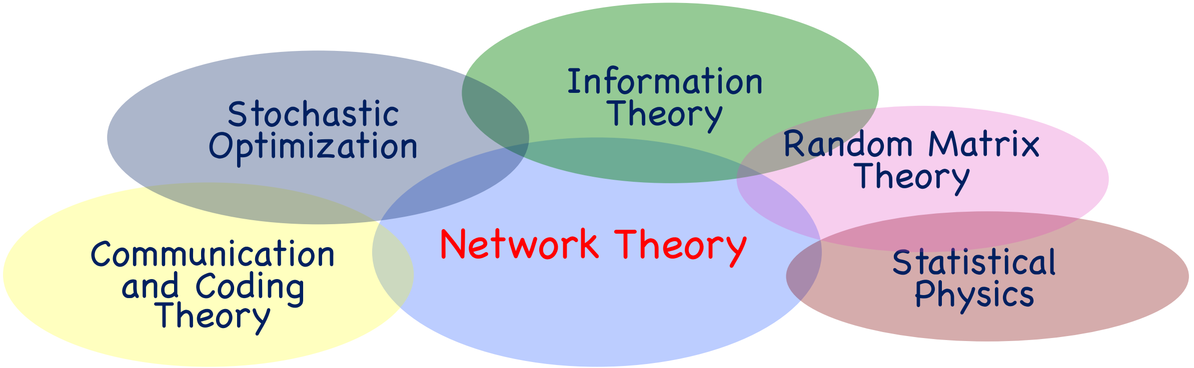
Outline

- Communication

- Content Distribution

  Efficient Content Storage and Delivery

  - Cache-aided coded multicast
  - Distributed network compression
  - Dynamic Data

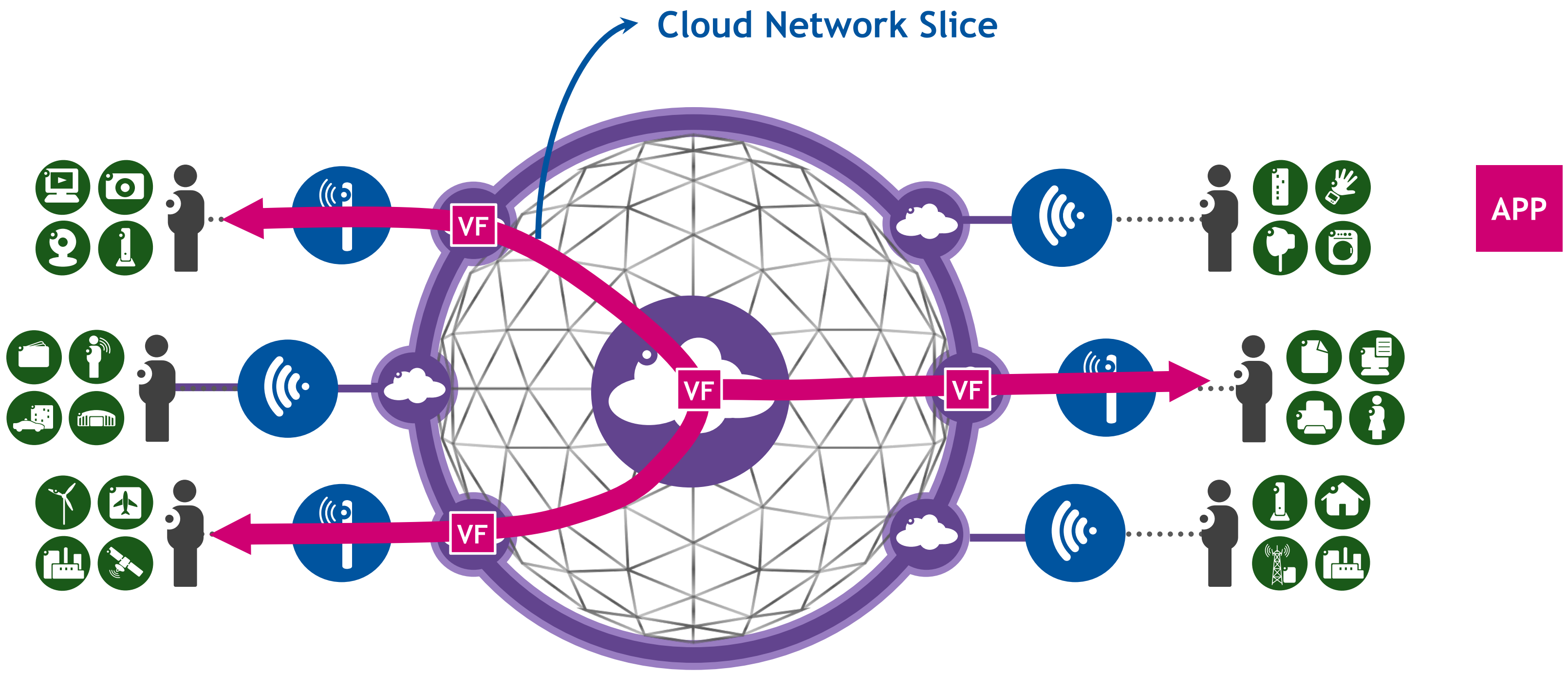


- Real-time Computation

  Efficient Service Configuration (Storage/Computation/Delivery)

  - Network Slicing (NFV/SDN)
  - Mobile Edge Computing (MEC)
  - Real-time Stream processing

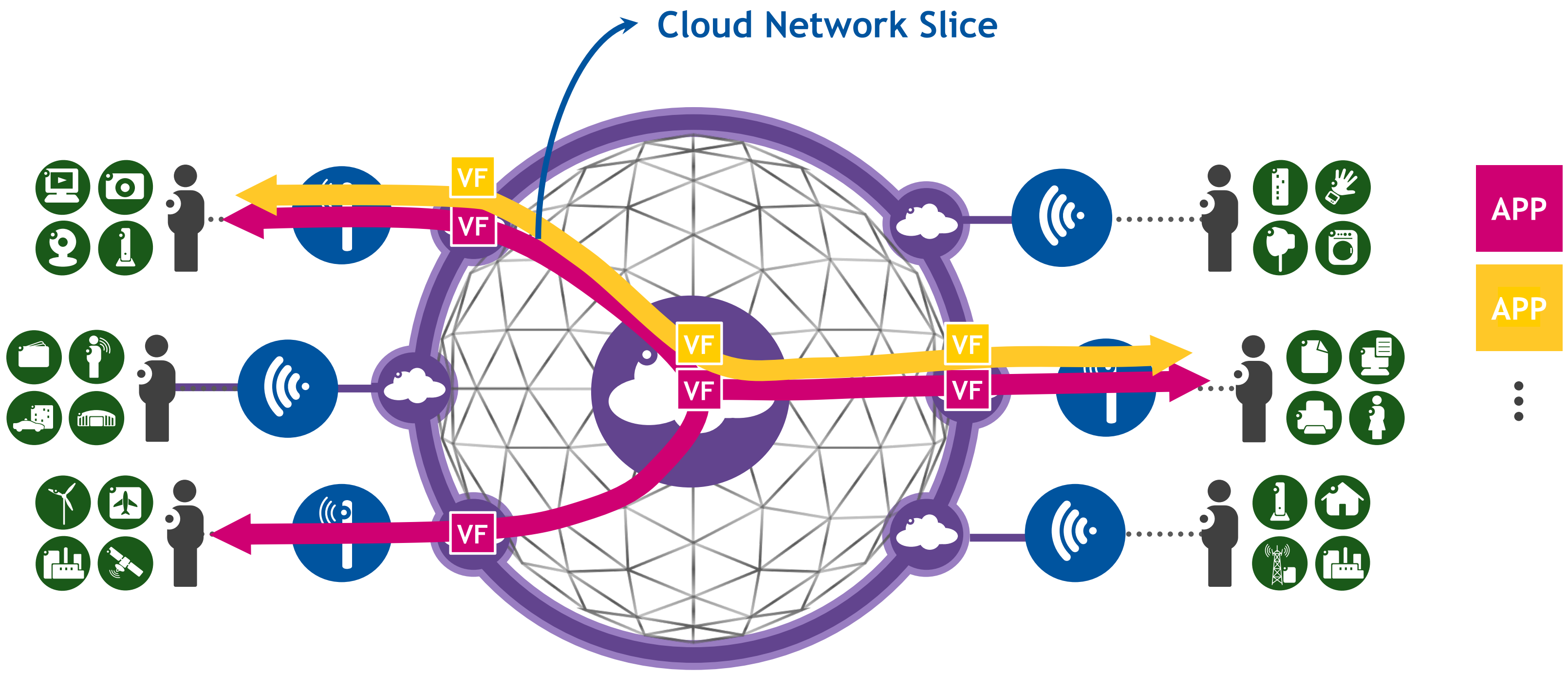
CLOUD-INTEGRATED NETWORKS AS UNIVERSAL COMPUTE PLATFORMS



Every human experience will be supported by a collection of services running over a cloud-integrated network.

M. Weldon, "The Future X Network: A Bell Labs Perspective," CRC PRESS, October 2015.

CLOUD-INTEGRATED NETWORKS AS UNIVERSAL COMPUTE PLATFORMS

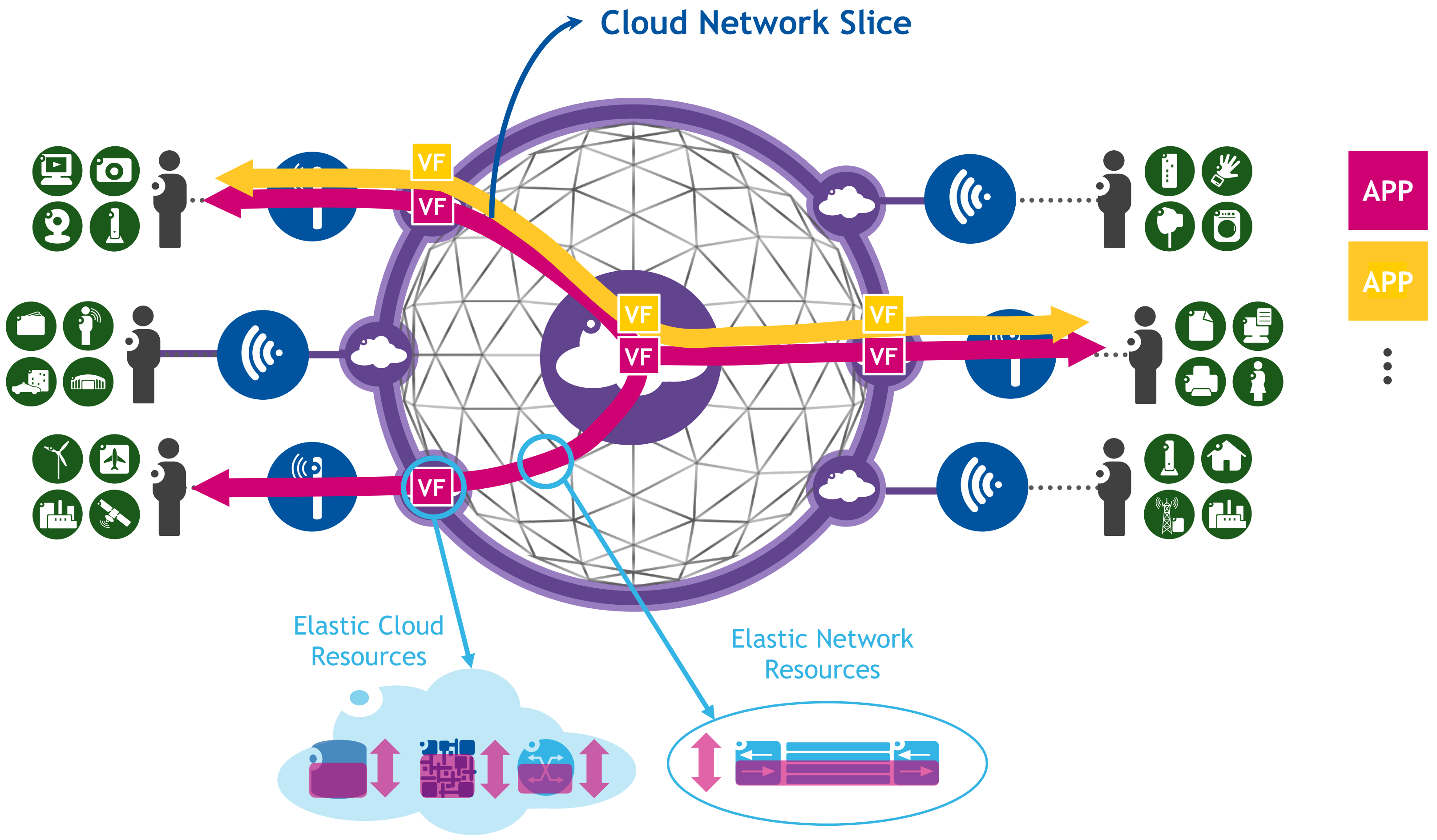


These services take information sources from the physical world, route them through multiple functions instantiated across the cloud network until delivering output flows that create some form of augmented value for the end user

M. Weldon, "The Future X Network: A Bell Labs Perspective," CRC PRESS, October 2015.

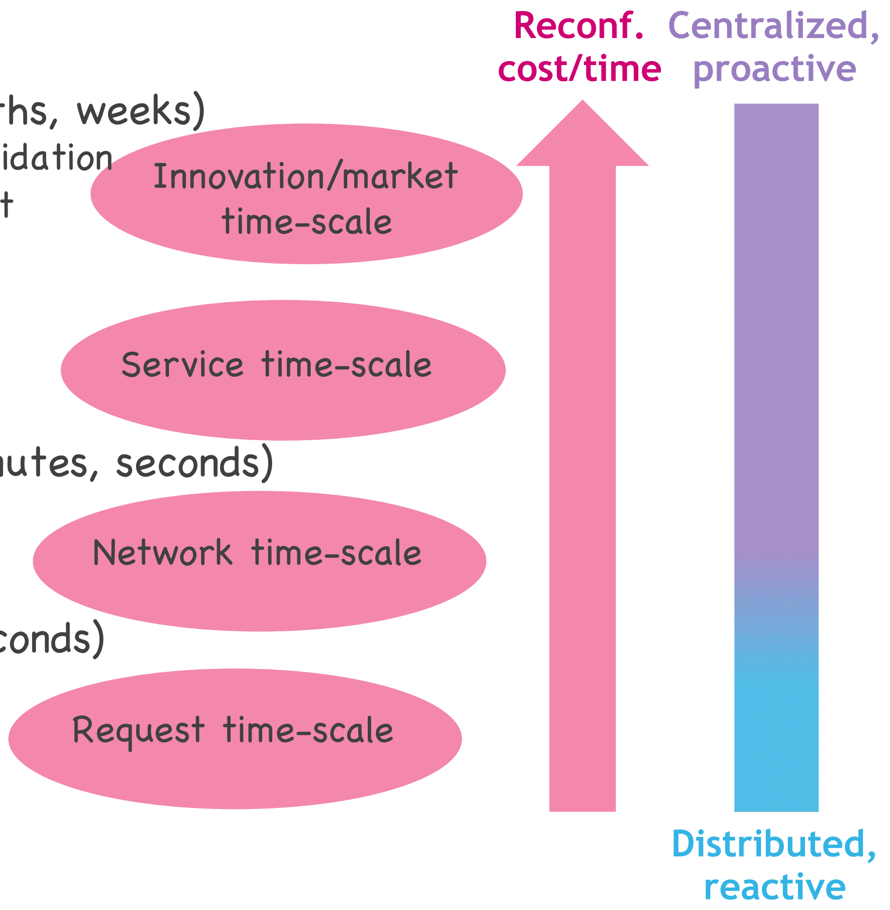
CLOUD-INTEGRATED NETWORKS AS UNIVERSAL COMPUTE PLATFORMS

- Opportunities
  - Users can consume resource- and interaction-intensive applications from resource-limited devices
  - Operators can reduce costs and create new value-added services
  - Overall sustainability
- Challenges
  - Optimized elastic consumption of compute/storage/network resources
  - **End-to-end autonomous configuration and control**



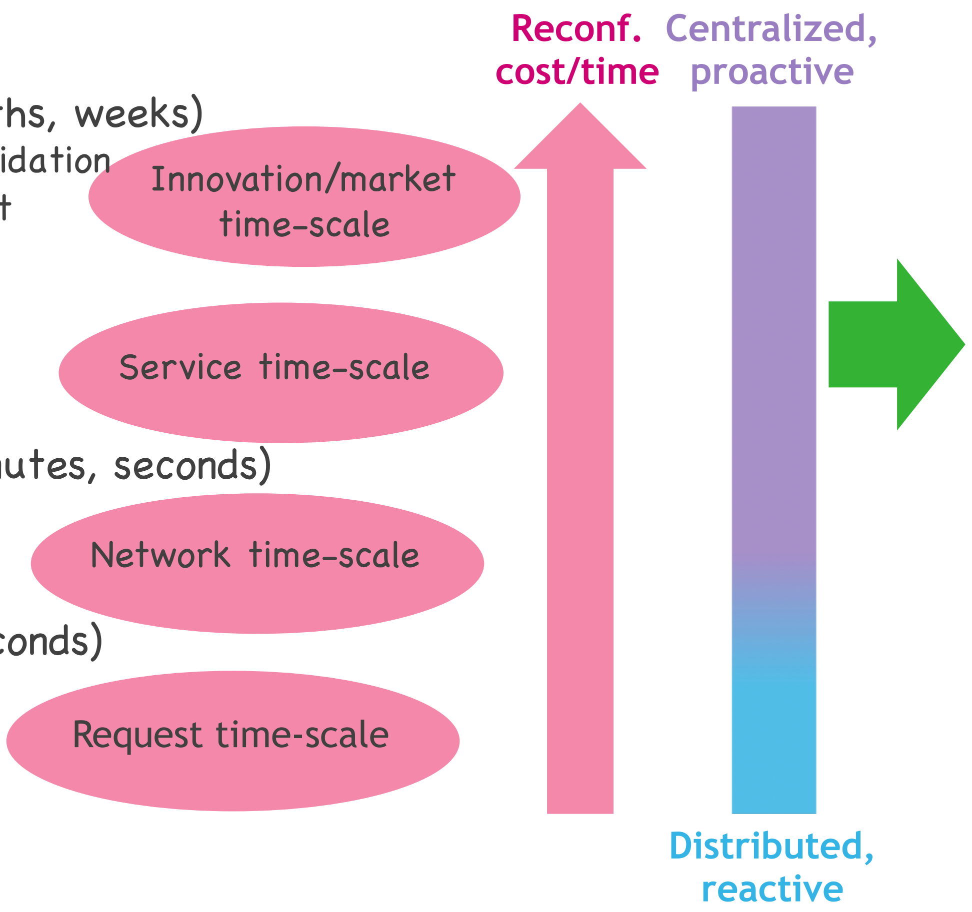
CLOUD NETWORK OPTIMIZATION AND CONTROL

- Physical resource allocation (months, weeks)
  - Physical site/link deployment/consolidation
  - Compute/storage/network equipment
- Service distribution (days, hours)
  - Data/function placement/migration
  - Cloud/network vResource allocation
- Virtual resource auto-scaling (minutes, seconds)
  - Virtual resource scale up/down
  - Virtual resource scale out/in
- Information flow (seconds, milliseconds)
  - Request routing
  - Flow scheduling
  - Load balancing



CLOUD NETWORK OPTIMIZATION AND CONTROL

- Physical resource allocation (months, weeks)
  - Physical site/link deployment/consolidation
  - Compute/storage/network equipment
- Service distribution (days, hours)
  - Data/function placement/migration
  - Cloud/network vResource allocation
- Virtual resource auto-scaling (minutes, seconds)
  - Virtual resource scale up/down
  - Virtual resource scale out/in
- Information flow (seconds, milliseconds)
  - Request routing
  - Flow scheduling
  - Load balancing

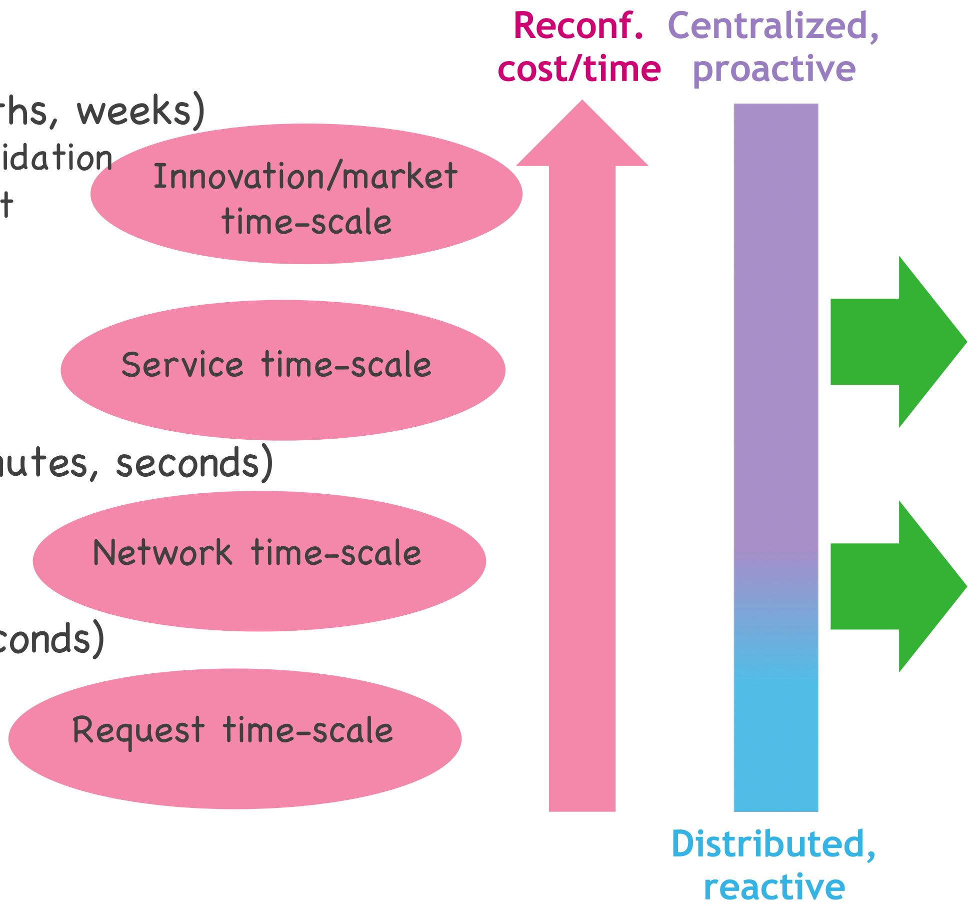


- E2E Service Optimization
  - Function placement and flow routing
  - Cloud/network resource allocation
  - Centralized solution with average demand knowledge

- Barcelo, Llorca, Tulino, Raman, "The Cloud Service Distribution Problem in Distributed Cloud Networks," IEEE ICC, 2015.
- Barcelo, Llorca, Tulino, Morell, Vicario, "IoT-Cloud Service Optimization in Smart Environments," IEEE JSAC, 2016.
- Feng, Llorca, Tulino, Raz, Molisch "Approximation Algorithms for the NFV Service Distribution Problem," IEEE INFOCOM, 2017.
- Poularakis, Llorca, Tulino, Tassiulas, "Joint Service Placement and Request Routing in Multi-Cell Edge Computing Networks," IEEE INFOCOM, 2019.
- Michael, Llorca, Tulino, "Approximation Algorithms for the Optimal Distribution of Real-time Stream-Processing Services," IEEE ICC, 2019

CLOUD NETWORK OPTIMIZATION AND CONTROL

- Physical resource allocation (months, weeks)
  - Physical site/link deployment/consolidation
  - Compute/storage/network equipment
- Service distribution (days, hours)
  - Data/function placement/migration
  - Cloud/network vResource allocation
- Virtual resource auto-scaling (minutes, seconds)
  - Virtual resource scale up/down
  - Virtual resource scale out/in
- Information flow (seconds, milliseconds)
  - Request routing
  - Flow scheduling
  - Load balancing

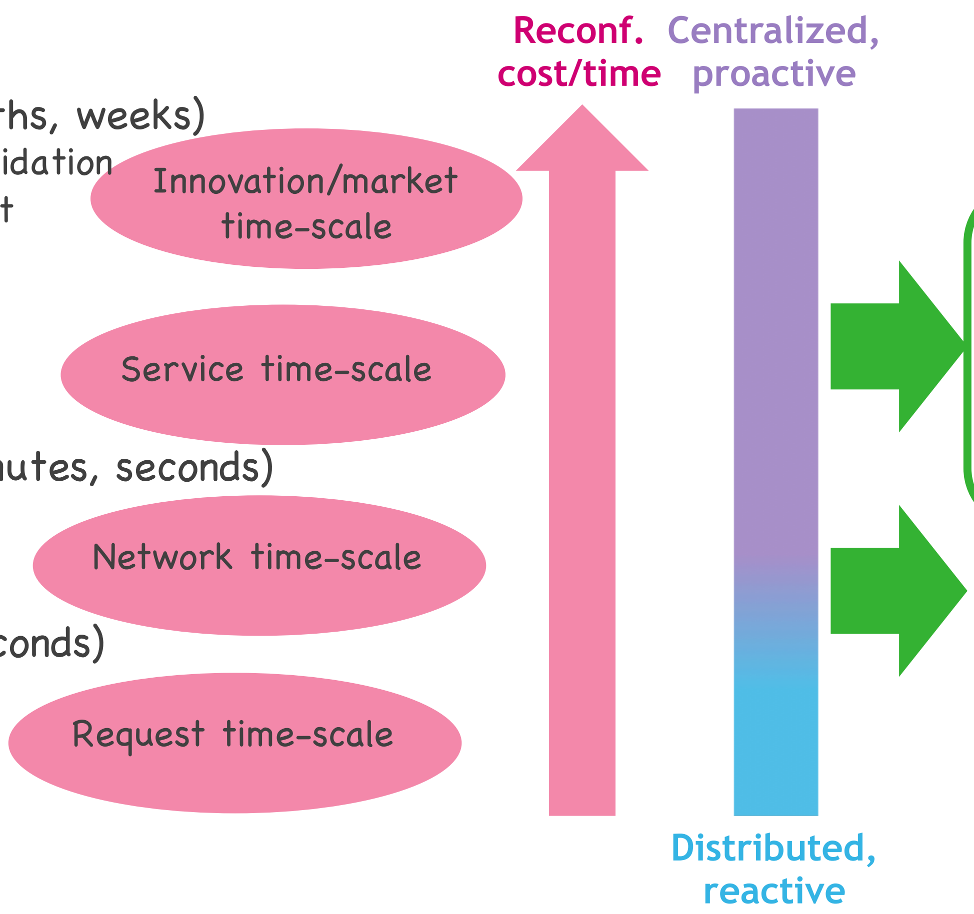


- E2E Service Optimization
  - Function placement and flow routing
  - Cloud/network resource allocation
  - Centralized solution with average demand knowledge
- Dynamic Service Control
  - Dynamic flow scheduling and virtual resource auto-scaling
  - Distributed online solution

- Feng, Llorca, Tulino, Molisch, "Dynamic Service Optimization in Distributed Cloud Networks," IEEE INFOCOM SWFAN, 2016.
- Feng, Llorca, Tulino, Molisch, "On the Delivery of Augmented Information Services over Wireless Computing Networks," IEEE ICC, 2017.
- Zhang, Sinha, Llorca, Tulino, Modiano, "Optimal Control of Distributed Computing Networks with Mixed-Cast Traffic Flows," IEEE INFOCOM, 2018.
- Feng, Llorca, Tulino, Molisch, "Optimal Dynamic Cloud Network Control," IEEE/ACM Transactions on Networking, 2018.
- Feng, Llorca, Tulino, Molisch, "Optimal Control of Wireless Computing Networks," IEEE Transactions on Wireless Communications, 2018.

CLOUD NETWORK OPTIMIZATION AND CONTROL

- Physical resource allocation (months, weeks)
  - Physical site/link deployment/consolidation
  - Compute/storage/network equipment
- Service distribution (days, hours)
  - Data/function placement/migration
  - Cloud/network vResource allocation
- Virtual resource auto-scaling (minutes, seconds)
  - Virtual resource scale up/down
  - Virtual resource scale out/in
- Information flow (seconds, milliseconds)
  - Request routing
  - Flow scheduling
  - Load balancing

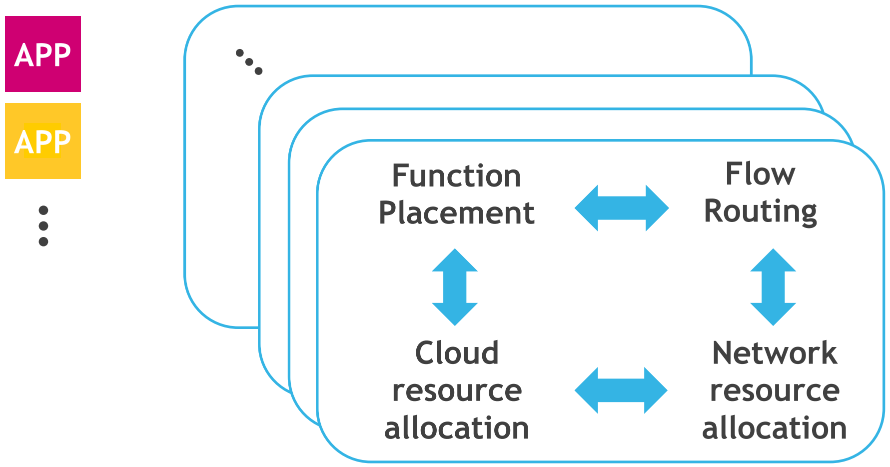


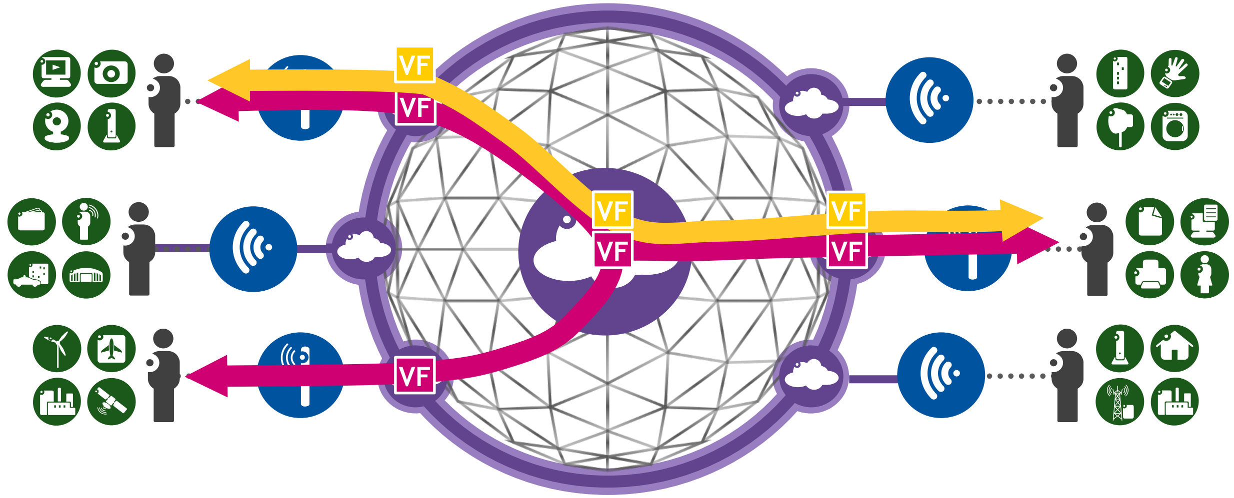
THIS TALK

- E2E Service Optimization
  - Function placement and flow routing
  - Cloud/network resource allocation
  - Centralized solution with average demand knowledge
- Dynamic Service Control
  - Dynamic flow scheduling and virtual resource auto-scaling
  - Distributed online solution

- Barcelo, Llorca, Tulino, Raman, "The Cloud Service Distribution Problem in Distributed Cloud Networks," IEEE ICC, 2015.
- Barcelo, Llorca, Tulino, Morell, Vicario, "IoT-Cloud Service Optimization in Smart Environments," IEEE JSAC, 2016.
- Feng, Llorca, Tulino, Raz, Molisch "Approximation Algorithms for the NFV Service Distribution Problem," IEEE INFOCOM, 2017.
- Poularakis, Llorca, Tulino, Tassiulas, "Joint Service Placement and Request Routing in Multi-Cell Edge Computing Networks," IEEE INFOCOM, 2019.
- Michael, Llorca, Tulino, "Approximation Algorithms for the Optimal Distribution of Real-time Stream-Processing Services," IEEE ICC, 2019

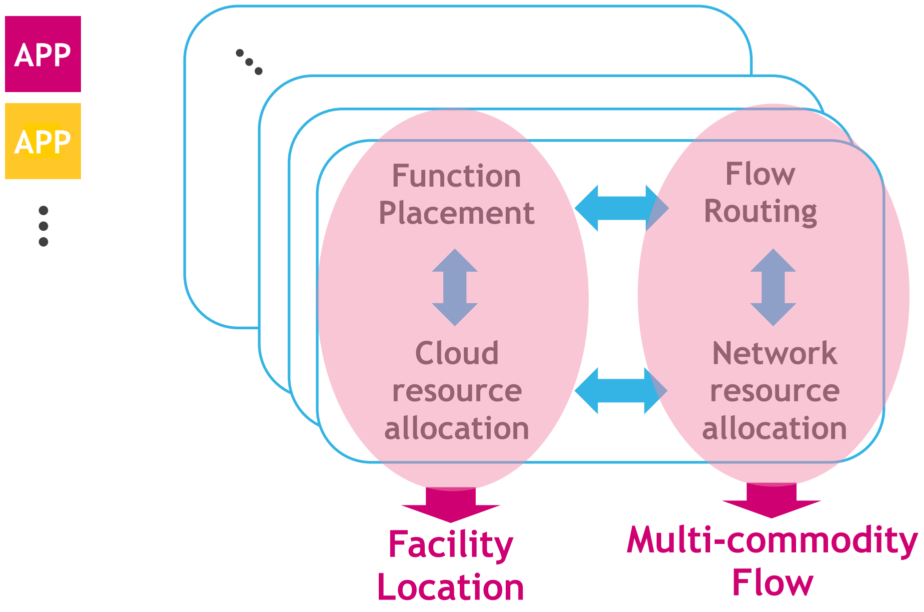
JOINT END-TO-END SERVICE OPTIMIZATION

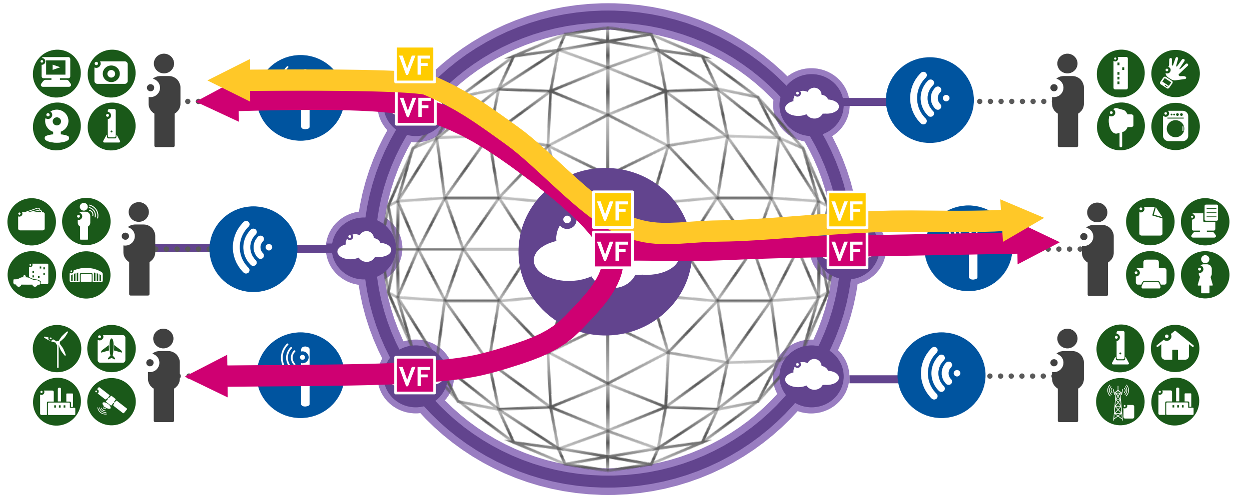




- Function placement
  - Function chaining, splitting, and replication
- Flow routing
  - Flow scaling
  - Mix of unicast and multicast traffic

EXISTING APPROACHES
COMPLEX DISJOINT SOLUTIONS

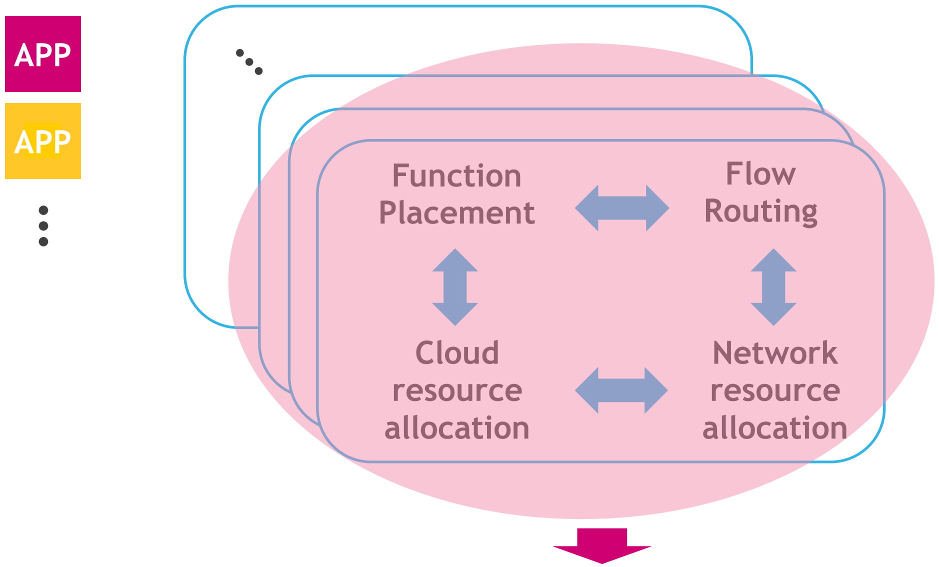


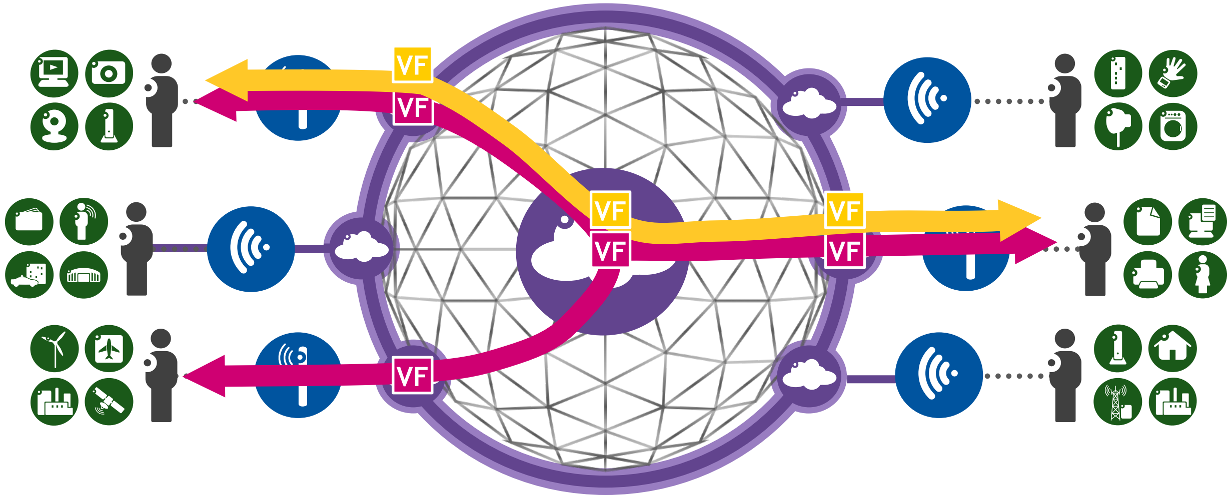


Separate data/function placement, flow routing, cloud and network resource allocation

- Driven by old vision of cloud and network separation
- No joint placement/routing optimization
- Unacceptable QoE, limited knowledge augmentation, and/or unsustainable costs with resource overprovisioning.

CLOUD NETWORK FLOW APPROACH





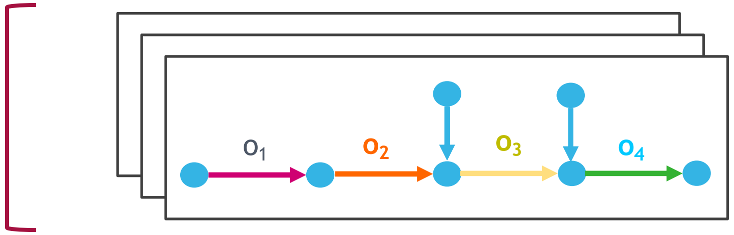
CLOUD NETWORK FLOW

- Comprehensive model
  - Arbitrary flow chaining, scaling, splitting, and replication
  - Arbitrary traffic mix (unicast and multicast flows)
  - Non-isomorphic embeddings
- Approximation guarantees

CLOUD NETWORK FLOW APPROACH

1. Service Graph

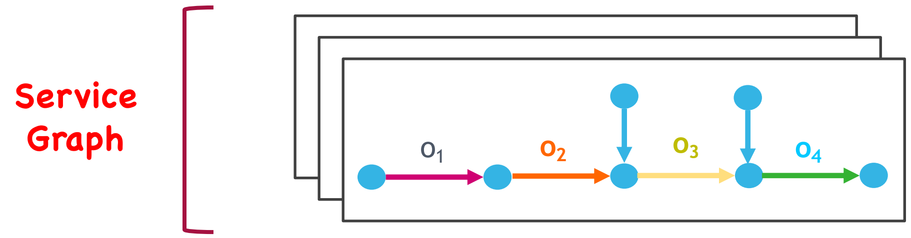
Service Graph



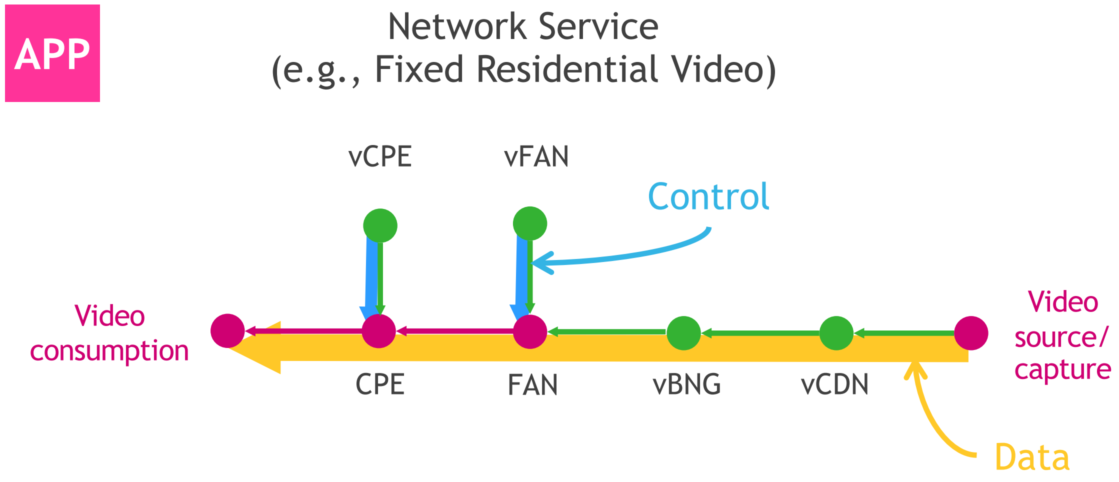
- Directed acyclic graph that encodes the relationship between service functions and associated input/output flows

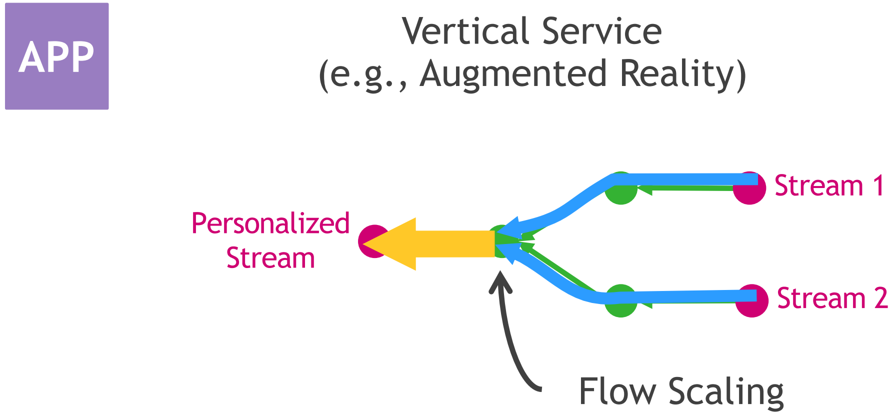
CLOUD NETWORK FLOW APPROACH:

1. Service Graph



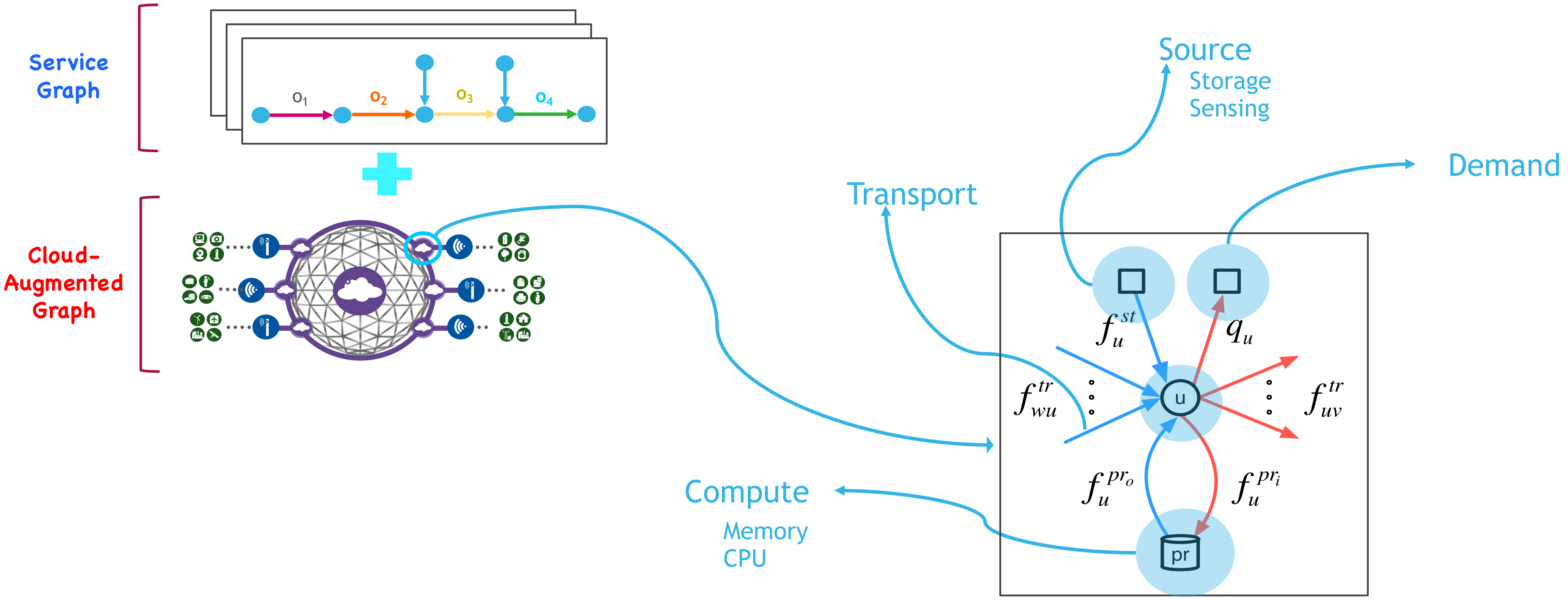
- Directed acyclic graph that encodes the relationship between service functions and associated input/output flows
- Control/data plane as well as hardware/software based functions
- Heterogeneous function complexity (proc. res. units per flow unit) and flow scaling (output flow units per input flow unit)





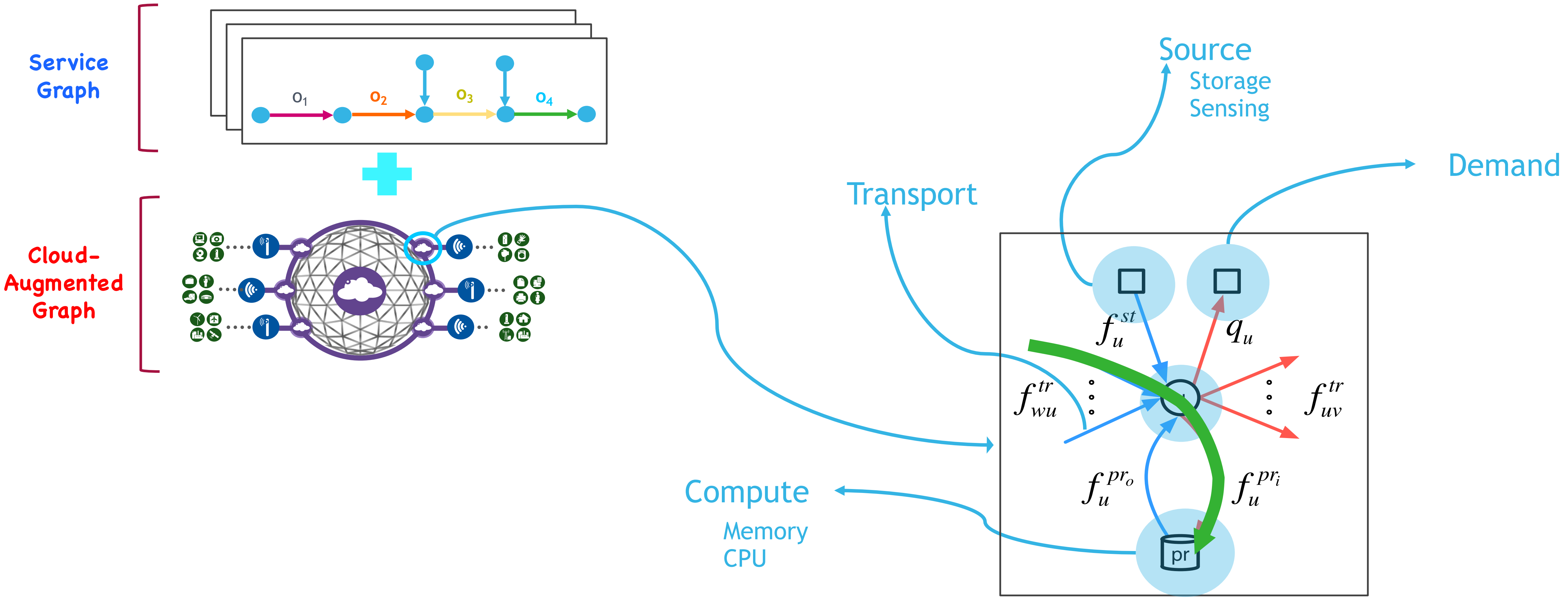
CLOUD NETWORK FLOW APPROACH

2. Cloud-augmented graph



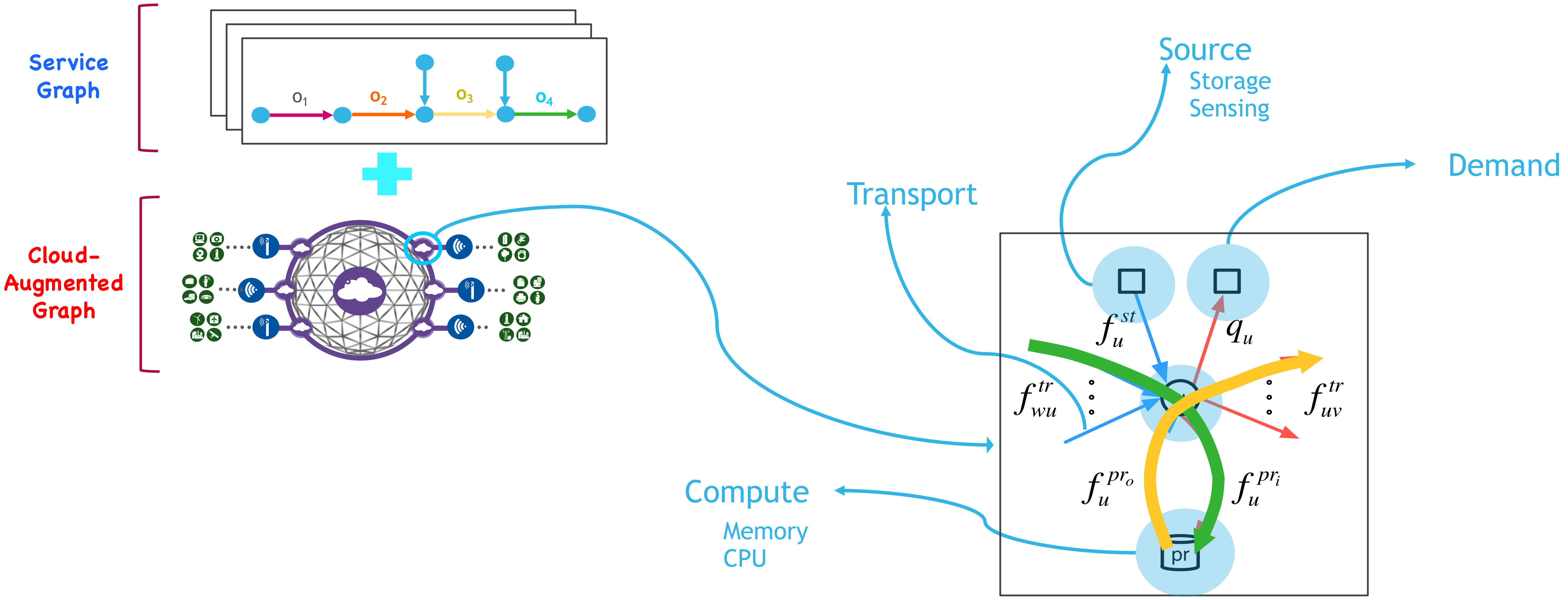
CLOUD NETWORK FLOW APPROACH

2. Cloud-augmented graph

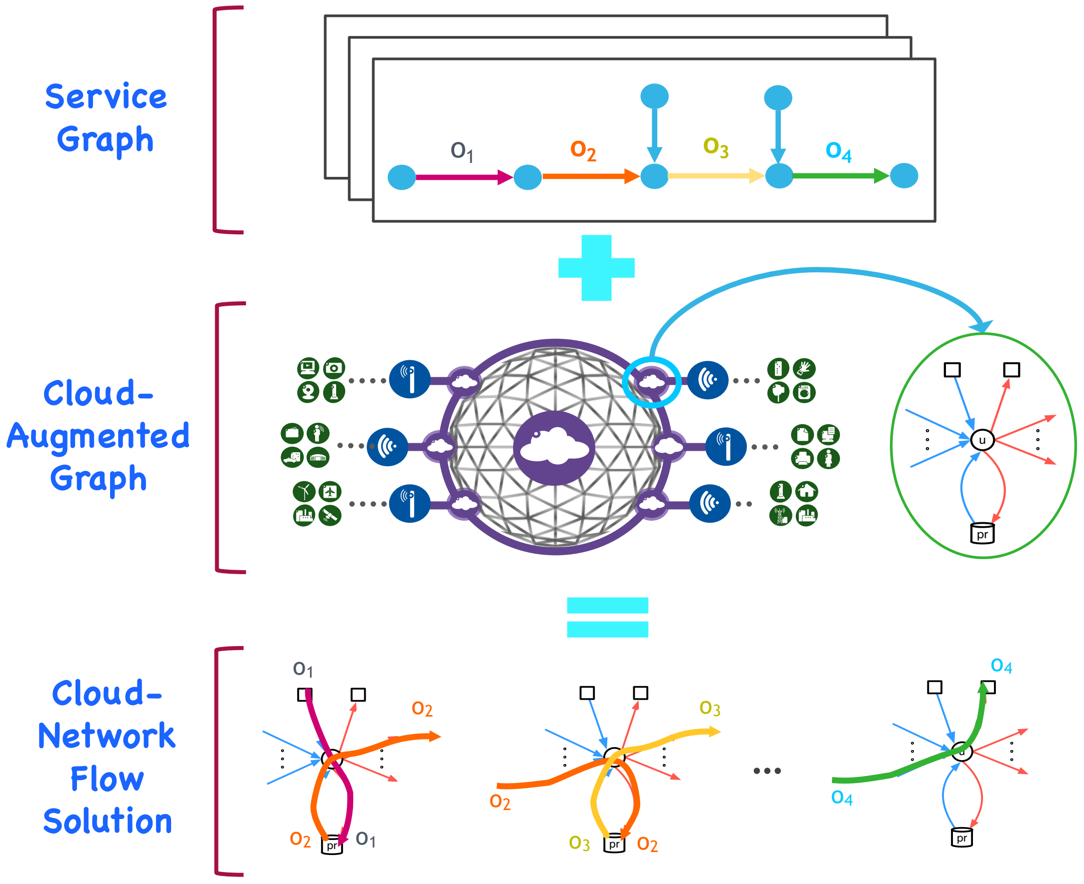


CLOUD NETWORK FLOW APPROACH

2. Cloud-augmented graph



CLOUD NETWORK FLOW APPROACH



- Mixed-cast multi-commodity-chain flow on a cloud-augmented graph
- Includes and generalizes placement and network flow problems
- Captures combined use of compute/storage/transport resources, unicast and multicast flows, and flow/function chaining, scaling, splitting, and replication
- Admits optimal polynomial time solutions under linear costs and splittable flows, and efficient approximations otherwise

CLOUD NETWORK FLOW

3. Mixed-cast chained information flow

$$\min \sum_{(u,v)} f_{uv}\, e_{uv}$$ Cost Function

s.t.

Virtual flows

$$\sum_v f_{vu}^{d,i} = \sum_v f_{uv}^{d,i} \qquad \forall u, d, i$$ Generalized Flow Conservation

$$f_{pu}^{d,i} = f_{up}^{d,j} \qquad \forall u, d, i, j \in \mathcal{Z}(i)$$ Flow Chaining

$$f_{su}^{d,i} = 1 \qquad \forall u, d, i \in \mathcal{S}(u)$$

$$f_{uq}^{d,i} = 1 \qquad \forall u, d, i \in \mathcal{Q}(u)$$ Sources and Demands

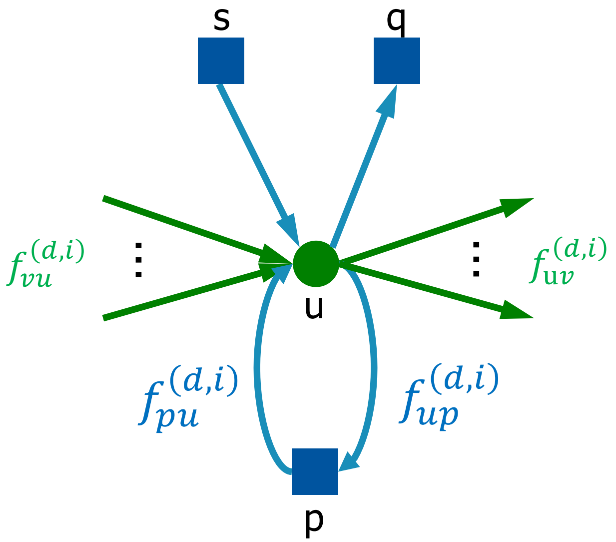
Actual flows

$$f_{uv}^{d,i} \le f_{uv}^{i} \qquad \forall (u,v), d, i$$

$$f_{uv}^{i} \le f_{uv}^{k} \qquad \forall (u,v), d, k, i \in \mathcal{K}(k)$$

$$\sum_k f_{uv}^{k} R_{uv}^{k} \le f_{uv} \le c_{uv} \qquad \forall (u,v)$$ Actual flow sizing

$$f_{uv}^{d,i}, f_{uv}^{i}, f_{uv}^{k} \in [0,1] \qquad \forall (u,v), d, i, k$$ Fractional/ Integer flows



CLOUD NETWORK FLOW

3. Mixed-cast chained information flow

$\min \sum_{(u,v)} f_{uv}\, e_{uv}$		Cost Function
s.t. $\sum_{v} f_{vu}^{d,i} = \sum_{v} f_{uv}^{d,i}$	$\forall u, d, i$	Generalized Flow Conservation
$f_{pu}^{d,i} = f_{up}^{d,j}$	$\forall u, d, i, j \in \mathcal{Z}(i)$	Flow Chaining
$f_{su}^{d,i} = 1$	$\forall u, d, i \in \mathcal{S}(u)$	Sources and Demands
$f_{uq}^{d,i} = 1$	$\forall u, d, i \in \mathcal{Q}(u)$	
$f_{uv}^{d,i} \le f_{uv}^{i}$	$\forall (u,v), d, i$	Actual flow sizing
$f_{uv}^{i} \le f_{uv}^{k}$	$\forall (u,v), d, k, i \in \mathcal{K}(k)$	
$\sum_{k} f_{uv}^{k} R_{uv}^{k} \le f_{uv} \le c_{uv}$	$\forall (u,v)$	
$f_{uv}^{d,i}, f_{uv}^{i}, f_{uv}^{k} \in [0,1]$	$\forall (u,v), d, i, k$	Fractional/ Integer flows

- Fractional flows
  - Good for network slices
  - Large aggregate flows
  - Per-flow splitting
- Integer flows
  - Good for individual services
  - Unsplittable flows

SERVICE CLASSIFICATION AND SOLUTIONS

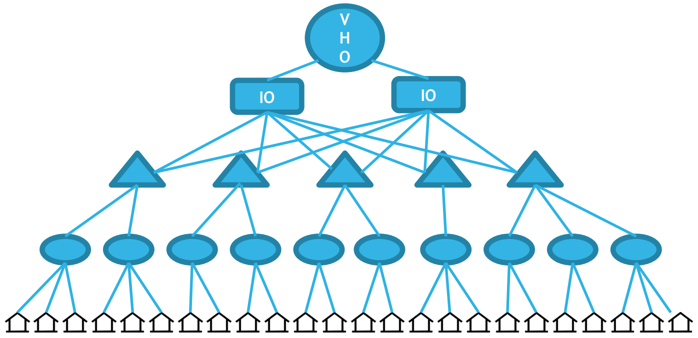
	Unicast		Multicast	
	Splittable	Unsplittable	Splittable	Unsplittable
Service Chain	Polynomial FPTAS	NP-Hard Bicriteria approx.	NP-Hard (no coding)	NP-Hard Bicriteria approx.
Service DAG	NP-Hard (no coding	NP-Hard Bicriteria approx.	NP-Hard (no coding)	NP-Hard Bicriteria approx.

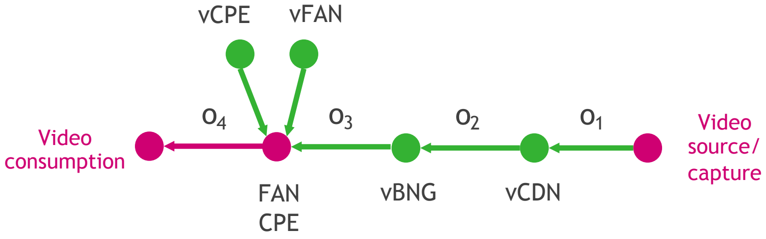
SERVICE CLASSIFICATION AND SOLUTIONS

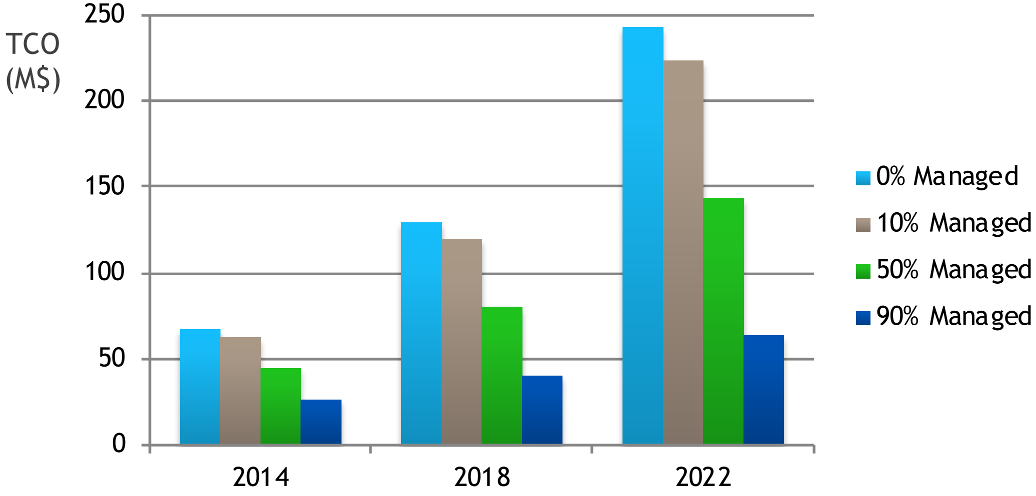
	Unicast		Multicast	
	Splittable	Unsplittable	Splittable	Unsplittable
Service Chain	Polynomial FPTAS 5G Slices	NP-Hard Bicriteria approx.	NP-Hard (no coding)	NP-Hard Bicriteria approx.
Service DAG	NP-Hard (no coding	NP-Hard Bicriteria approx.	NP-Hard (no coding)	NP-Hard Bicriteria approx. RTSP

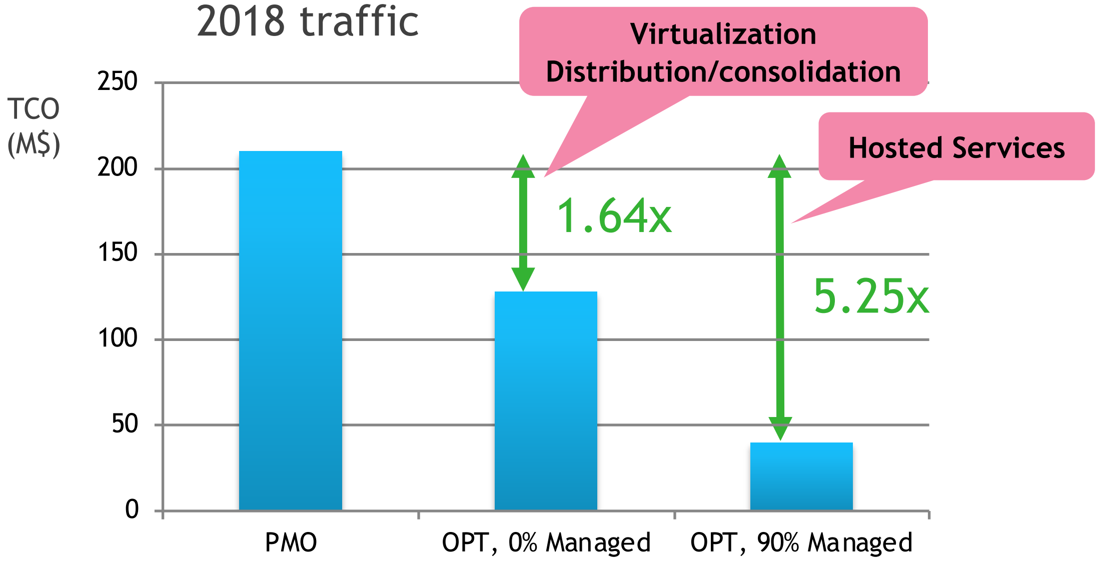
NETWORK SERVICE CHAINS

- Network: Generic US Metro
  - 4 Metro PoP, 12 Metro Agg, 60 Metro Edge
  - 10G links, CloudBand compute nodes
- Service: Fixed Residential Video
  - Data plane: vCDN, vBNG, FAN, CPE
  - Control Plane: vCDN, vBNG, vFAN, vCPE
- Demand:
  - 2014, 2018, 2022 video traffic
  - 50% VoD, 40% VS, 10% IPTV



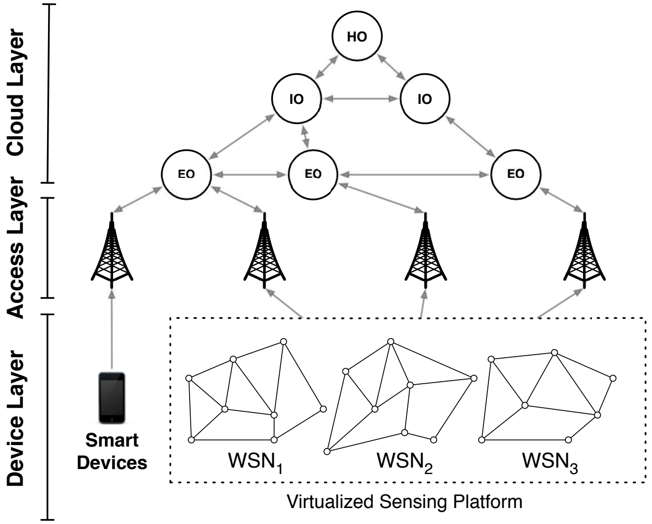




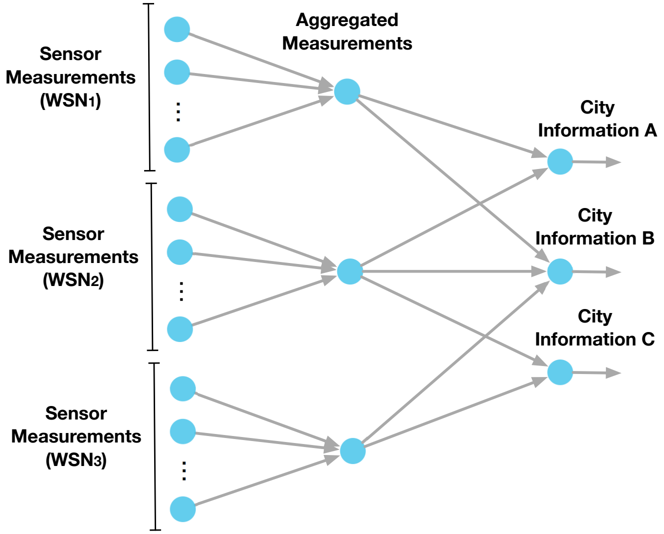


SMART CITY SERVICES

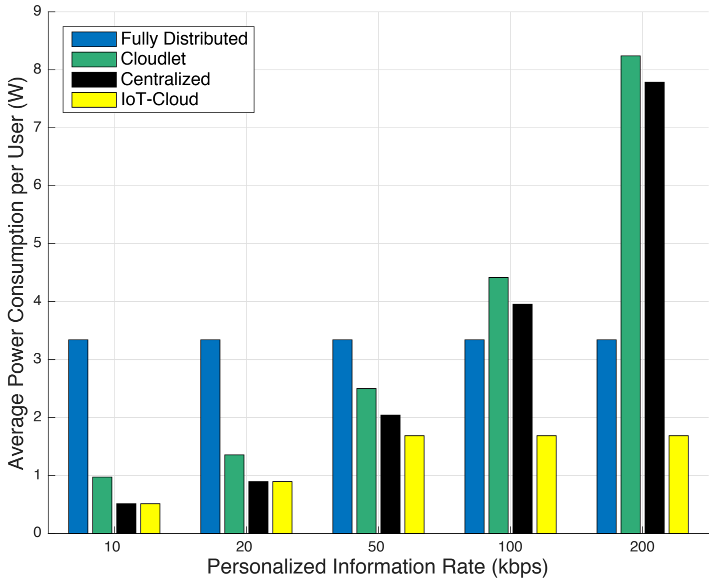
- IoT-Cloud Network:
  - Cloud layer (core, metro, edge)
  - Access layer
  - Device layer



- City Streams Service:
  - Deliver contextually relevant personalized city streams

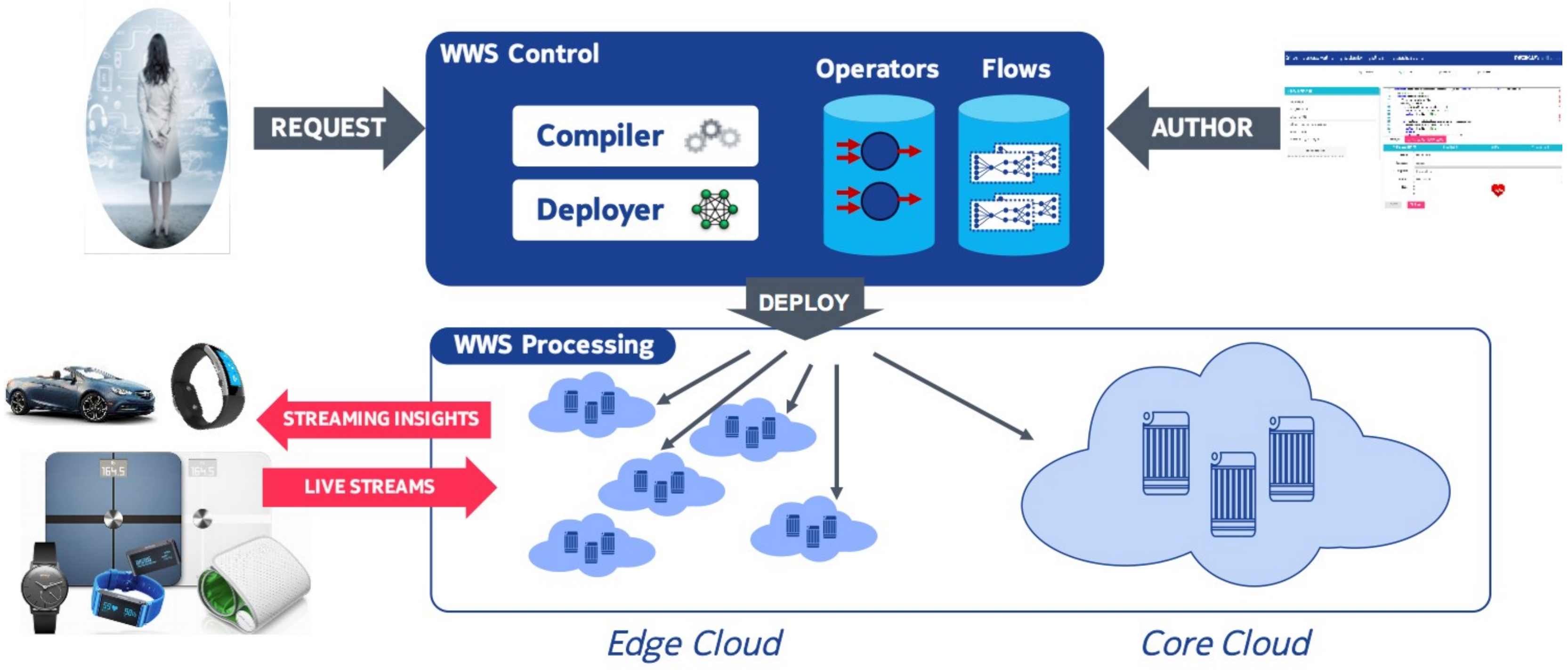


- Operational cost as a function of personalized stream data rate



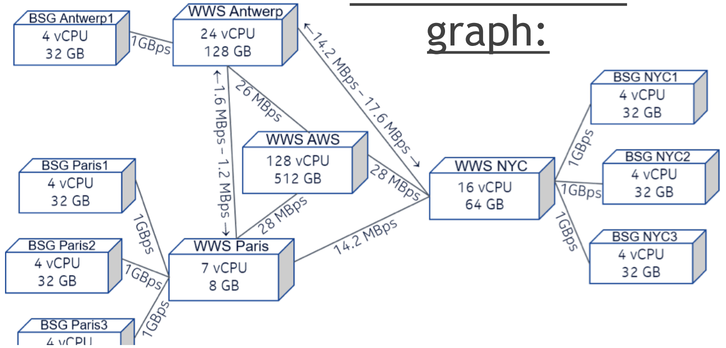
WORLD WIDE STREAMS (WWS)

- Distributed stream processing platform
- Produces and delivers streams of real-time relevance to geographically dispersed users via the real-time processing of geographically dispersed source streams

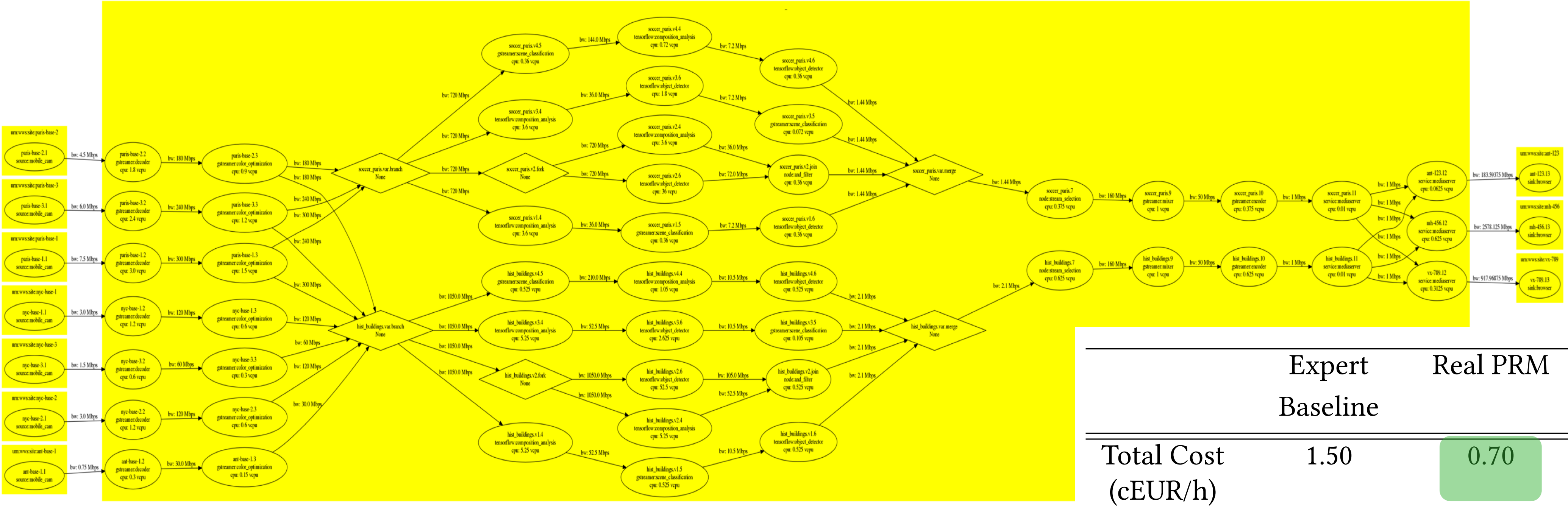


WORLD WIDE STREAMS

Cloud network graph:



Service graph:



	Expert Baseline	Real PRM	“flat”	“lucky”
Total Cost (cEUR/h)	1.50	0.70	1.68	2.60
Variant selected	Var. 1 (manual)	Var. 4 (autm.)	Var. 4 (autm.)	Var. 4 (autm.)
Placement note	“Video close to source”	Smart distrib.	Mostly AWS	Mostly AWS

2X-4X

CONCLUSIONS

- Networks are becoming universal compute platforms, able to host a variety of services and applications that can optimize the automated operation of physical systems and augment human experiences in real time.

- New mathematical tools are required to jointly optimize the allocation of compute, storage, and network resources, as well as the efficient flow of information over such highly distributed computing infrastructures.

- Dynamic cloud-network compression aims to an E2E compression of information throughout its entire lifecycle - capture/creation, upload, storage, computation, and delivery – in order to maximize conveyed information per unit cost

- Using cloud-network-wide spatiotemporal redundancy to push the fundamental limits of information compression, pioneering algorithms in network compression, including compressed video delivery with up to 8X capacity gains has been designed.

- Cloud network flow generalizes traditional network information flow models to jointly capture the efficient storage, computation, and delivery of information of real-time relevance.

- Significant efficiency improvements can be obtained via the end-to-end optimization of next generation services over distributed cloud-integrated networks.

REFERENCES – CONTENT DISTRIBUTION

1. R. Ali, V. Cadambe, J. Llorca, A. Tulino, "Fundamental Limits of Erasure-Coded Key-Value Stores with Side Information," Trans. On Communications, 2020.
2. P. Hassanzadeh, A. Tulino, J. Llorca, E. Erkip, "Rate-Memory Trade-Off for Caching and Delivery of Correlated Sources" IEEE Information on Theory, 2020.
3. P. Hassanzadeh, A. Tulino, J. Llorca, E. Erkip, Paris.a Hassanzadeh, Antonia M. Tulino, Jaime Llorca, Elza Erkip, " Trans. On Wireless Communications, 2020.
4. G. Vettigli, M. Ji, K. Shanmugan, J. Llorca, A. Tulino, G. Caire, "Efficient Algorithms for Coded Multicasting in Heterogeneous Caching Networks", MDPI Entropy, March 2019
5. P. Hassanzadeh, A. Tulino, J. Llorca, E. Erkip, "On Coding for Cache-Aided Delivery of Dynamic Correlated Content", IEEE Journal on Selected Areas in Communication, June 2018.
6. R. Ali, V. Cadambe, J. Llorca, A. Tulino, "Multi-Version Coding with Side Information," IEEE ISIT, June 2018
7. R. Ali, V. Cadambe, J. Llorca, A. Tulino, "Bridging the gap between the extremes of complete side information versus no side information in consistent distributed storage" Information Theory and Applications, 2018,
8. C. Rosetti, S. Romano, A.M. Tulino, SHINE: Secure Hybrid In Network caching Environment, IEEE International Symposium on Networks, Computers and Communications (ISNCC), 2018
9. M. Ji, A. M. Tulino, J. Llorca, G. Caire, "Order-Optimal Rate of Caching and Coded Multicasting with Random Demands", IEEE Information on Theory, Marzo 2017.
10. Y. Fadlallah, A.M. Tulino, D. Barone, G. Vettigli, J. Llorca, J.M. Gorce, "Coding for Caching in 5G Networks" IEEE Communications Magazine, Vol. 55, No. 2, pp. 106-113, 2017
11. P. Hassanzadeg, A. Tulino, J. Llorca, E. Erkip, "Broadcast Caching Networks with Two Receivers and Multiple Correlated Sources" ASILOMAR, 2017
12. . Shanmugam, A. Dimakis, J. Llorca, A. M. Tulino, "Coded Caching Main Technical Barriers: Finite Packetization and Channel Heterogeneity" ASILOMAR, 2017.
13. P. Hassanzadeg, A. Tulino, J. Llorca, E. Erkip, "Rate-Memory Trade-off for the Two-User Broadcast Caching Network with Correlated Sources" ISIT, 2017.
14. K. Shanmugam, A. M. Tulino, A. Dimakis, "Coded Caching with Linear Subpacketization is Possible using Ruzsa-Szeméredi Graphs" ISIT, 2017.

REFERENCES – CONTENT DISTRIBUTION

14. K. Shanmugam, M. Ji, A. M. Tulino, J. Llorca, A. Dimakis "Finite Length Analysis of Caching-Aided Coded Multicasting" IEEE Information on Theory, Vol. 62, No. 10, pp. 5524-5537, 2016.
15. B. Azari, O. Simeone, U. Spagnolini, A. Tulino "Hypergraph-Based Analysis of Clustered Cooperative Beamforming with Application to Edge Caching", IEEE Wireless Communications Letters, Vol. 5, No. 1, pp. 84-87, 2016.
16. A. S. Cacciapuoti, M. Caleffi, M. Ji, J. Llorca, A. M. Tulino, "Speeding up Future Video Distribution via Channel-Aware Caching-Aided Coded Multicast", IEEE Journal on Selected Areas in Communications, Vol. 34, No. 8, pp. 2207-2218, 2016.
17. P. Hassanzadeg, A. Tulino, J. Llorca, E. Erkip, "Distortion-Memory Tradeoffs in Cache-Aided Wireless Video Delivery", 22nd Annual International Conference on Mobile Computing and Networking (Mobicom'16), New York, USA, October, 2016.
18. P. Hassanzadeg, A. Tulino, J. Llorca, E. Erkip, "Memory-Rate Trade-off for Caching and Delivery of Correlated Sources," 37th IEEE Sarnoff Symposium, Newark, New Jersey, USA, September 2016. (Best paper award).
19. P. Hassanzadeg, A. Tulino, J. Llorca, E. Erkip, "Correlation-Aware Distributed Caching and Coded Delivery," IEEE Information Theory Worskhop, (ITW), September 2016.
20. P. Hassanzadeg, A. Tulino, J. Llorca, E. Erkip, "Caching-Aided Coded Multicast for Correlated Sources," IEEE International Symposium on Turbo Codes & Iterative Information Processing (ISTC), Brest, France, September 2016. (Invited Talk)
21. A.S. Cacciapuoti, M Caleffi, M. Ji, J. Llorca, A. Tulino, "On the Impact of Lossy Channels in Wireless Edge Caching", IEEE International Conference on Communications (ICC2016), 2016.
22. J. Llorca, A. M. Tulino, M. Varvello, J. Esteban, D. Perino, Member, "Energy Efficient Dynamic Content Distribution", IEEE Journal on Selected Areas in Communications, Vol. 33, No. 12, pp. 2826-2836, 2015.
23. M. Ji, A. M. Tulino, J. Llorca, G. Caire, "Caching in Combination Networks", IEEE ASILOMAR, November 2015.
24. M.Ji, K. Shanmugam, G. Vettigli, J, Llorca, A. M. Tulino, "An Efficient Multiple-Groupcast Coded Multicasting Scheme for Finite Fractional Caching", 2015 IEEE International Conference on Communications (ICC2015), London, 2015.
25. G. Vettigli, M. Ji, A. M. Tulino, J, Llorca, P. Festa, "An Efficient Coded Multicasting Scheme Preserving the Multiplicative Caching Gain" IEEE Infocom, 2015, Hong Kong, 2015.
26. M. Ji, M. Wing, A. M. Tulino, J. Llorca, G. Caire, M. Effros, M. Langberg, "On the Fundamental Limits of Caching in Combination Networks", 16th IEEE International Workshop on Signal Processing Advances in Wireless Communications, SPAWC 2015, Stockholm, Sweden, 2015

REFERENCES – CONTENT DISTRIBUTION

27. M. Ji, A. M. Tulino, J. Llorca, G. Caire, "Caching and coded multicasting: multiple requests with random demands", IEEE Information Theory Workshop, Israel, 2015.
28. P. Hassanzadeh, E. Erkip, J. Llorca, A. Tulino, "Distortion Memory Tradeoffs in Cache-Aided Wireless Video Delivery", IEEE ALLERTON, 2015.
29. M. Ji, A. Tulino, J. Llorca, G. Caire, "Caching and Coded Multicasting: Multiple Groupcast Index Coding", GlobalSIP 2014, Atlanta, Georgia, 2014.
30. M. Ji, A. Tulino, J. Llorca, G. Caire, "On the Average Performance of Caching and Coded Multicasting with Random Demands", SWCS 2014, Barcelona, Spain, 2014.
31. K. Shanmugam, M. Ji, A. Tulino, J. Llorca, A. Dimakis. "Finite Length Analysis of Caching-Aided Coded Multicasting," IEEE Allerton Conference, 2014.
32. . Llorca, A. M. Tulino, "Minimum cost caching-aided multicast under arbitrary demand" Conference on Signals, Systems and Computers, Asilomar, 2013.
33. J. Llorca, A. M. Tulino, K. Guan, J. Esteban, M. Varvello, N. Choiy, D. Kilper, "Dynamic In-Network Caching for Energy Efficient Content Delivery", INFOCOM 2013.
34. J. Llorca, A. Tulino, K. Guan, D. C. Kilper, "Network-coded caching-aided multicast for efficient content delivery", IEEE ICC 2013, Budapest, Hungary, 2013.

REFERENCES – REAL-TIME COMPUTATION

1. C.H. Wang, J. Llorca, A. Tulino, T. Javidi, Dynamic Cloud Network Control Under Reconfiguration Delay and Cost", IEEE Transactions on Networking, Januray 2019
2. K. Poularakis, J. Llorca, A. Tulino, L. Tassiulas, "Joint Service Placement and Request Routing in Multi-Cell Mobile Edge Computing Networks," IEEE INFOCOM, April 2019.
3. M. Michael, J. Llorca, A. Tulino, "Approximation Algorithms for the Optimal Distribution of Real-Time Stream-Processing Services," IEEE ICC, May 2019.
4. H. Feng, J. Llorca, A. Tulino, A. Molisch, "Optimal Control of Wireless Computing Networks," IEEE Transactions on Wireless Communications, October 2018.
5. J. Zhang, A. Sinha, J. Llorca, A. Tulino, E. Modiano, "Optimal Control of Distributed Computing Networks with Mixed-Cast Traffic Flows," IEEE INFOCOM, April 2018.
6. H. Feng, J. Llorca, A. Tulino, A. Molisch, "Optimal Dynamic Cloud Network Control", IEEE/ACM Transactions on Networking, September 2018.
7. L. Jiao, A. Tulino, J. Llorca, Y. Yin, A. Sala, "Smoothed Online Resource Allocation in Multi-Tier Distributed Cloud Networks," IEEE Transactions on Networking, June 2017.
8. H. Feng, J. Llorca, A. M. Tulino, "Impact of channel state information on wireless computing network control" ASILOMAR, 2017.
9. H. Feng, J. Llorca, A. M. Tulino, A. Molish, "On the Delivery of Augmented Information Services over Wireless Computing Networks" IEEE International Conference on Communications (ICC2017), 2017.
10. H. Feng, J. Llorca, A. Tulino, D. Raz, A. Molish, "Approximation Algorithms for the NFV Service Distribution Problem" IEEE INFOCOM, 2017.
11. M. Barcelo, A. Correa, J. Llorca, A. M Tulino, J.L. Vicario, A. Morell, "IoT-Cloud Service Optimization in Next Generation Smart Environments", IEEE Journal on Selected Areas in Communications, Vol, 34, No. 12, pp. 4077-4090, 2016.
12. L. Jei, A. Tulino, J. Llorca, Y. Jin, A. Sala, "Smoothed Online Resource Allocation in Multi-Tier Distributed Cloud Networks", IEEE International Parallel & Distributed Processing Symposium (IPDPS), 2016.
13. H. Feng, J. Llorca, A. Tulino, A. Molish, "Optimal Dynamic Cloud Network Control", IEEE International Conference on Communications (ICC2016). 2016. (Best paper award).

REFERENCES – REAL-TIME COMPUTATION

14. H. Feng, J. Llorca, A. Tulino, A. Molish, "Dynamic Network Service Optimization in Distributed Cloud Networks", IEEE INFOCOM Workshops, 2016.
15. P. Marchetta, J. Llorca, A. Tulino, A. Pescape, "MC3: a Cloud Caching Strategy for Next Generation Virtual CDNs", IEEE Networking, 2016.
16. J. Llorca, A. M. Tulino, M. Varvello, J. Esteban, D. Perino, Member, "Energy Efficient Dynamic Content Distribution", IEEE Journal on Selected Areas in Communications, Vol. 33, No. 12, pp. 2826-2836, 2015.
17. M. Barcelo, J, Llorca, A. M. Tulino, N. Raman, "The Cloud Service Distribution Problem in Distributed Cloud Networks", 2015 IEEE International Conference on Communications (ICC2015), London, 2015.
18. J, Llorca, C. Sterle, A. M. Tulino, A. Sforza. A. Esposito, "Joint Content-Resource Allocation in Software Defined Virtual CDNs", 2015 IEEE International Conference on Communications (ICC2015), London, 2015.