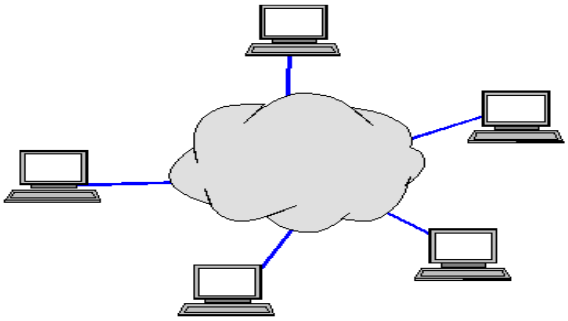
Architecting for Edge Diversity: Supporting Rich Services over an Unbundled Transport

Fahad R. Dogar, MSR
Peter Steenkiste, CMU

Outline

- What's wrong with today's transport?
- How we propose to fix it?
  → Overview of Tapa
- Three new transport services
  → Three diverse case studies

Original Internet



Included fairly homogenous hosts and networks
Network's role was limited to providing "connectivity"
Host-based applications (e.g., telnet, ftp)

3

Today's Internet – Diversity Everywhere

Heterogeneous Networks & Devices	Rich Network Services
   meraki  wimax	Akamai amazon web services facebook hulu YouTube fon

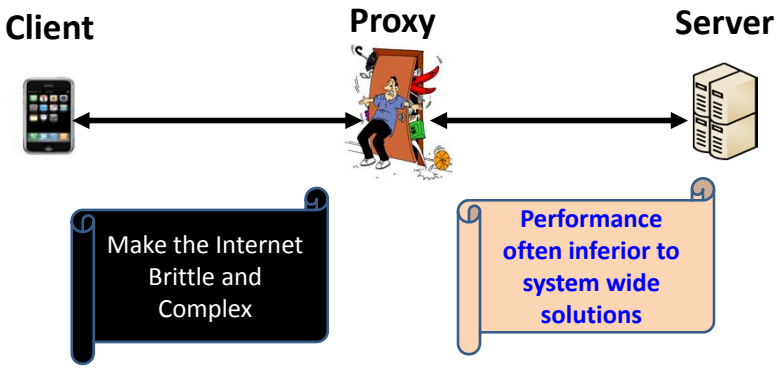
Trend towards Data oriented Apps

Dealing with diversity is difficult in today's Internet

4

Limitations of "Ad-Hoc" Solutions

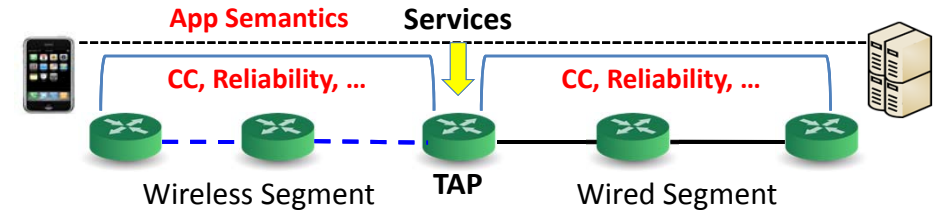
Example: Transparent Proxies



Need to address diversity in a more systematic way

5

Systematic Support for Diversity



Wired

WX

Two Key Concepts

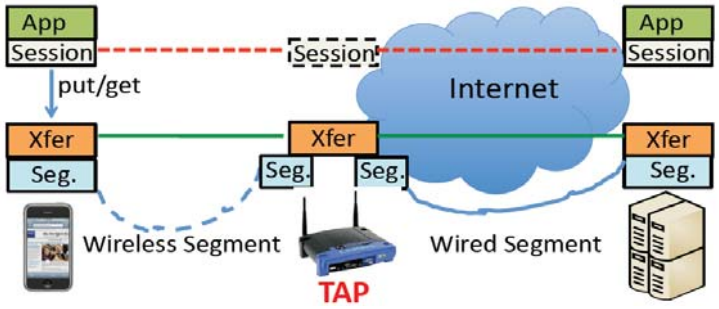
Unbundling

Visible in-path services

6

Tapa - Architecture

Synthesizes these concepts in a new transport architecture



Session-> specific application semantics (e.g., reliability, integrity)

Xfer Layer -> E2E Data Delivery

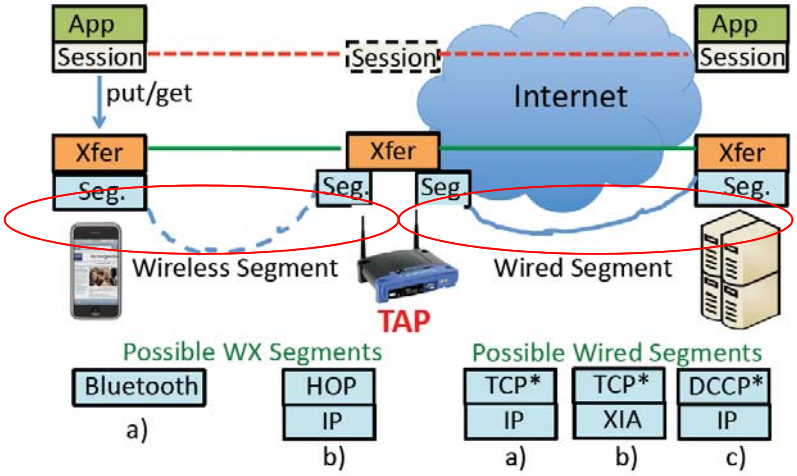
Segment Layer -> best effort data delivery across a segment

7

Tapa – Identifiers and Addresses

- Hosts
  - Self-Certified Identifiers (similar to AIP, XIA, etc)
  - Also have separate "locators" that define network attachment point – changes over time
  - Use existing mechanisms for routing
- Data
  - Tapa layers work at the granularity of ADUs
    - Segment layer transfers segments across segment
  - ADUs are identified using their hash value
    - Facilitates use of data oriented optimizations

Segment Layer

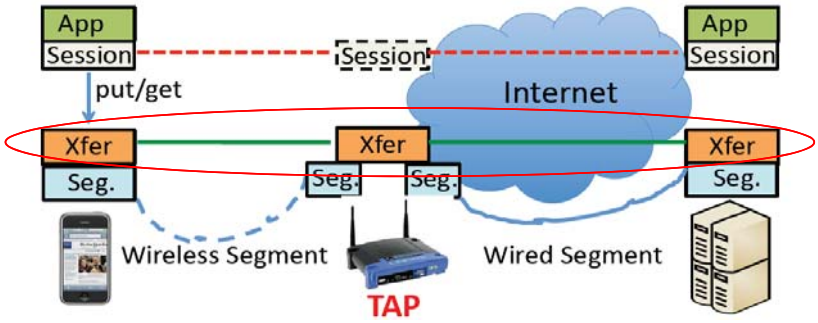


Best effort ADU delivery over a segment
Internal protocol optimized for the segment

9

Transfer Layer

E2E ADU delivery & E2E Congestion Control



Supports two modes:

Pull – Opportunistic ADU retrieval based on its id

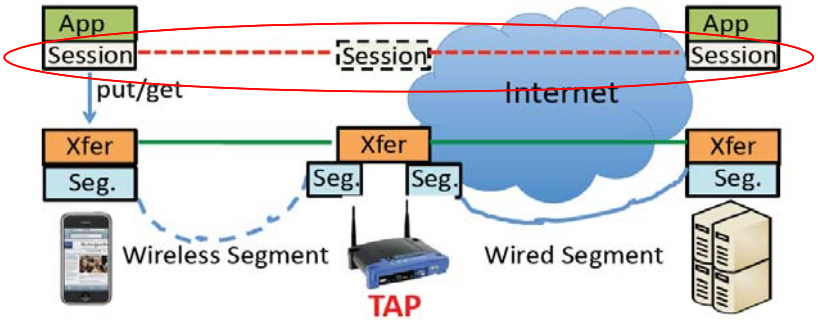
Push – Directly send ADU to the other end-point

Why do we need E2E congestion control?

10

Session Layer

Support common application semantics

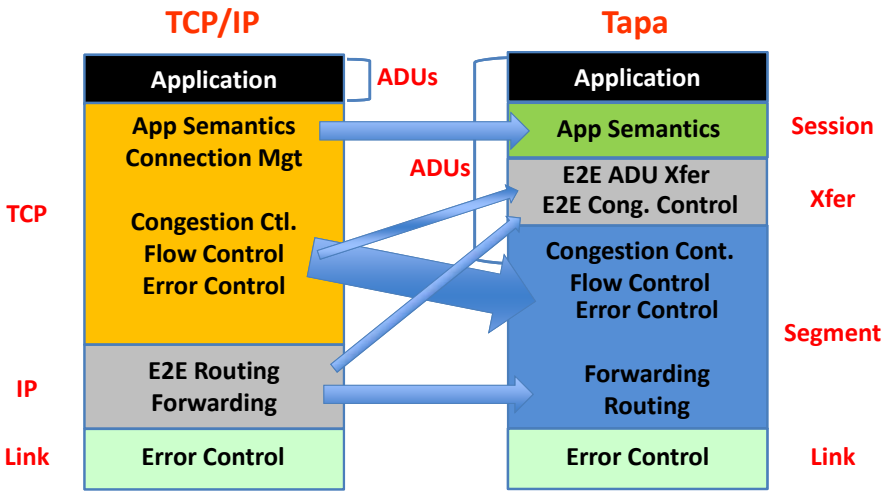


Reliability, Confidentiality, Data Integrity, and Data Ordering

Traditional E2E Semantics + New Semantics involving services

11

Tweaking Modularity

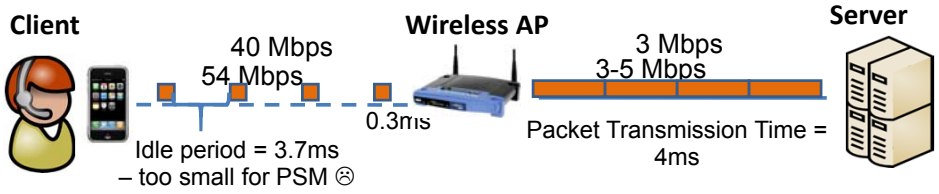


12

Tapa -- Services

- TAPs can support various services
  - E.g., Caching, Traffic Shaping, Transcoding, etc
  - Implemented at the transfer layer
  - May change session or high layer semantics
- Case Studies
  - Catnap - Traffic Shaping to Improve Energy Efficiency
  - Vigilante - Content Distribution for OSNs
  - Swift - Mobile and Wireless Optimizations

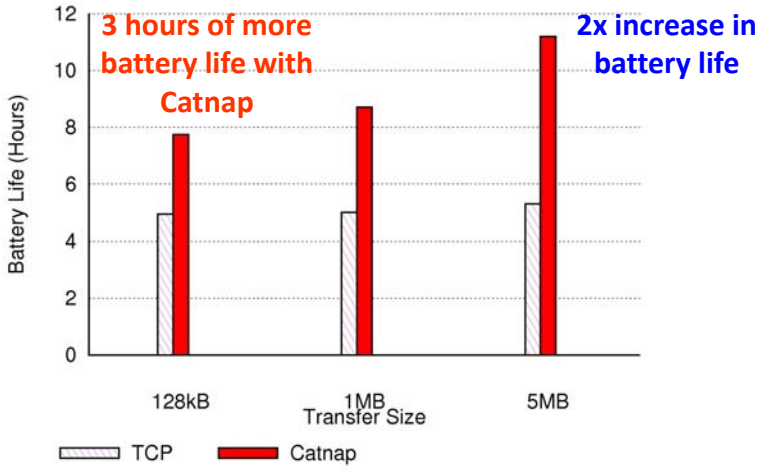
Bandwidth Discrepancy in End-to-end Transfers



- Catnap combines short idle times to save energy
- TAP buffers incoming packets while client sleeps
- Scheduler schedules burst transfer to maximize energy savings while avoiding increase in e-e delay
  - Estimates bandwidth in wired and wireless segments

14

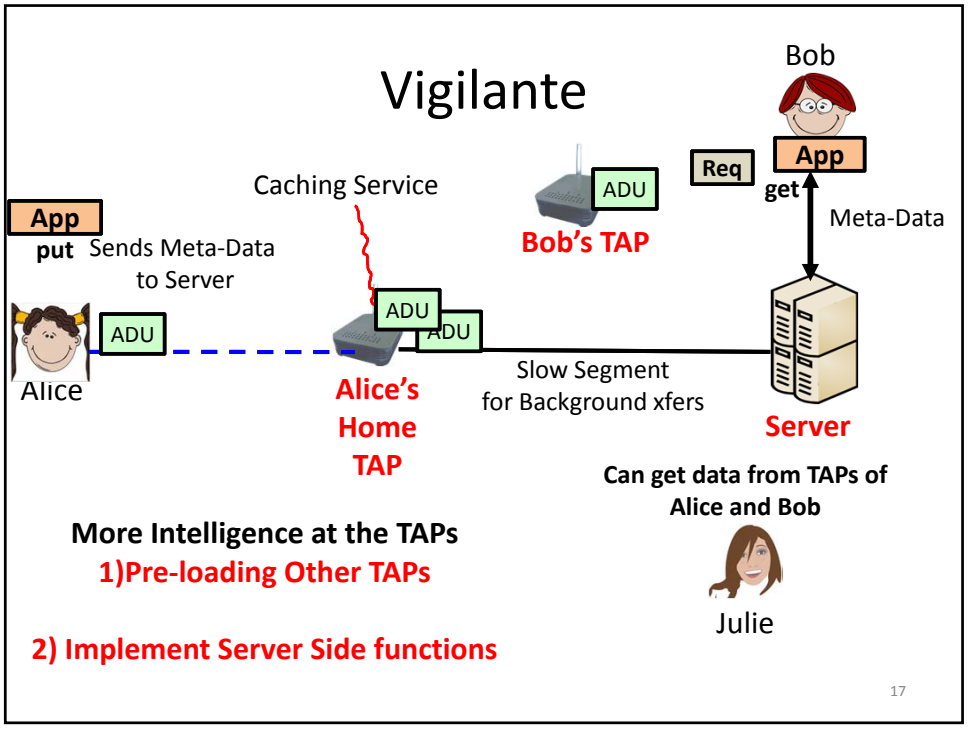
Battery Life Improvement - N810

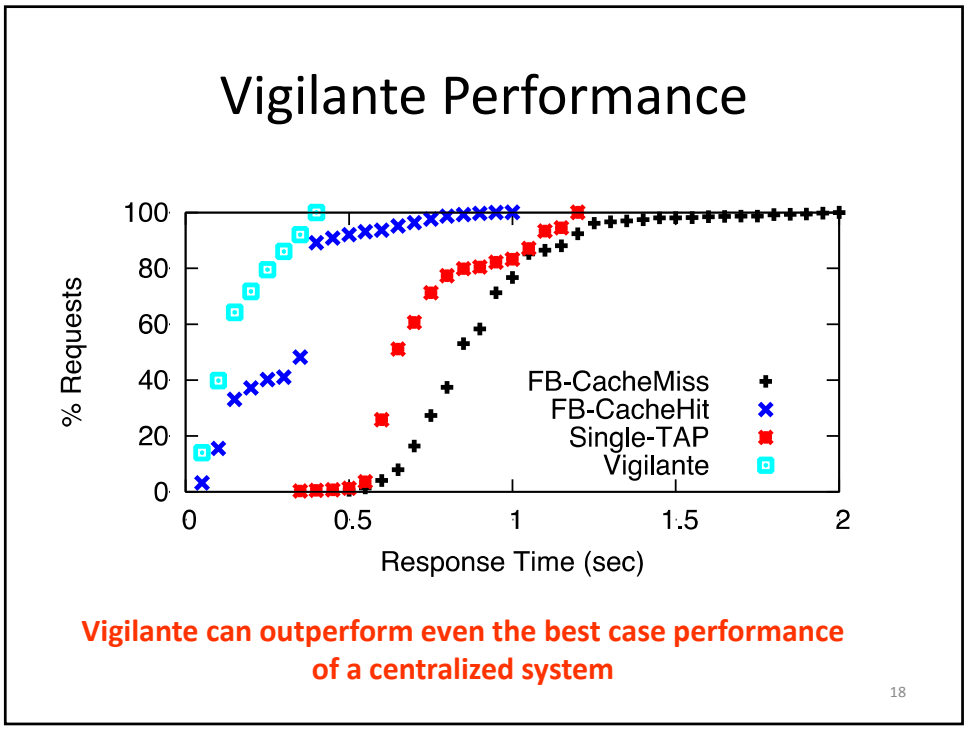


15

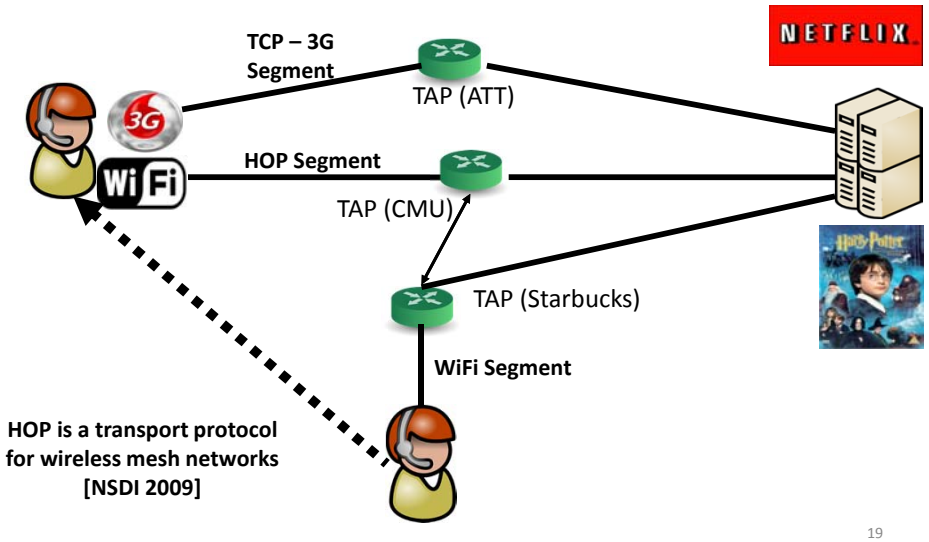
Vigilante

- Content Distribution for OSNs
  - Today's client-server design requires high cost; performance is still often poor
- Idea: Use TAPs of social networking users as opportunistic caches
  - Downloading from nearby friends can reduce latency
  - Reduces burden on the server


Vigilante
Bob
App
Req
get
Meta-Data
Caching Service
ADU
Bob's TAP
App
put
Sends Meta-Data
to Server
ADU
ADU
ADU
Alice
Alice's
Home
TAP
Slow Segment
for Background xfers
Server
Can get data from TAPs of
Alice and Bob
Julie
More Intelligence at the TAPs
1)Pre-loading Other TAPs
2) Implement Server Side functions
17

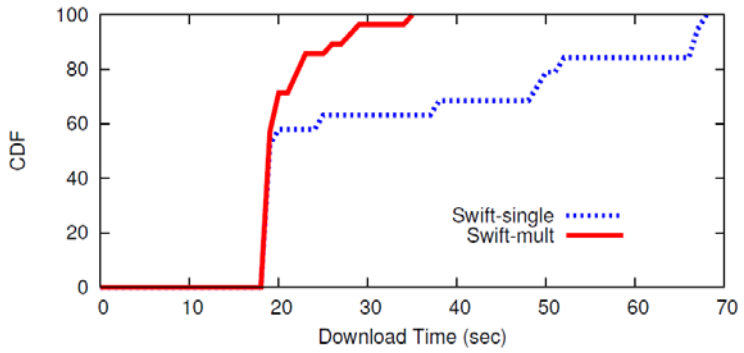

Vigilante Performance
% Requests
100
80
60
40
20
0
0
0.5
1
1.5
2
Response Time (sec)
FB-CacheMiss
FB-CacheHit
Single-TAP
Vigilante
Vigilante can outperform even the best case performance
of a centralized system
18

Swift – Optimizations for Mobile Users



19

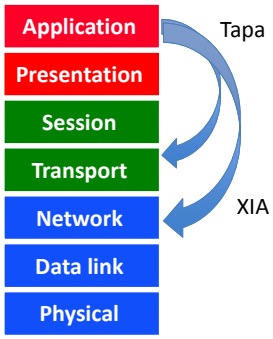
Swift in vehicular settings



Using multiple segments helps in masking disruptions

Tapa and the eXpressive Internet Architecture

- XIA offers native support for communication with content and services, besides hosts
- Tapa transport architecture carries over naturally
- Tapa content support can be pushed "into the network"
  - XIA does caching of chunks (ADUs) based on content IDs
- Applications can use service IDs for edge services



21

Related Work

- Lots of work on middleboxes
  - Mostly focus on hidden middleboxes
  - Flow versus network layer visible middleboxes
- Unbundling of the transport for middleboxes
  - Tapa more general and operates at ADU level
- Overlays: Tapa has constrained but dynamic topology, network diversity, …
- DTN: Tapa has more constrained topologies, but supports push/pull and service insertion
- Shares features with some future internet proposals

Why should I read the paper?

- Prototype Implementation and Evaluation
- How we ported a legacy application (Firefox)?
- Use of various segment protocols
  - Porting existing ones (e.g., HOP)
  - new protocols (UDP-Blast, a Lightweight UDP based protocol for WiFi)

Summary

- Increasing diversity calls for rethinking today's transport architecture
  - Really about rethinking how to modularize
- Tapa synthesizes two concepts:
  Unbundling and visible services
- Three diverse case studies demonstrate the flexibility and performance of Tapa