Enabling TDMA for Today's Wireless LANs

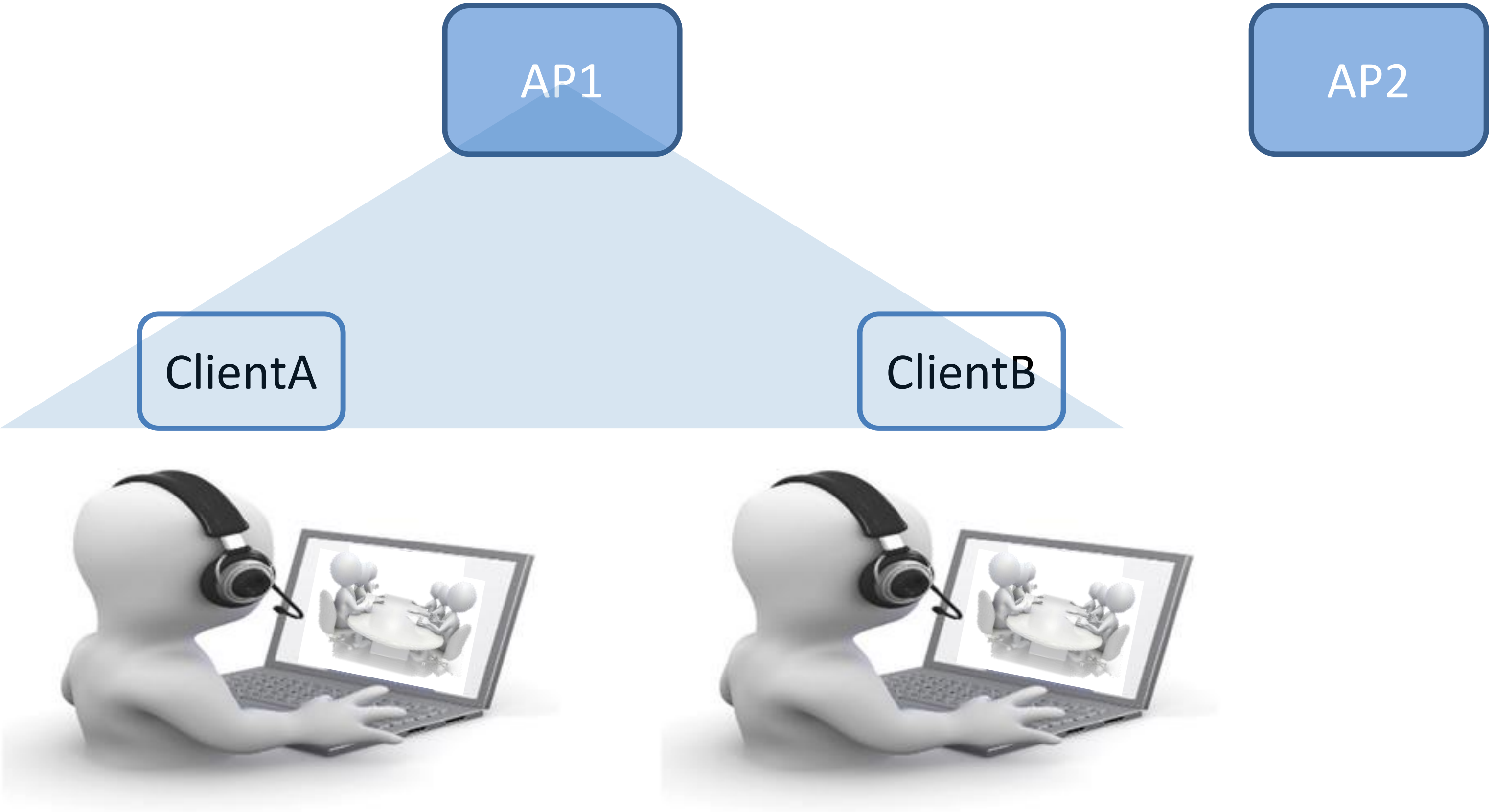
Zhice Yang[1*], Jiansong Zhang[2*], Kun Tan[2], Qian Zhang[1] , Yongguang Zhang[2]

[1]CSE, Hong Kong University of Science and Technology

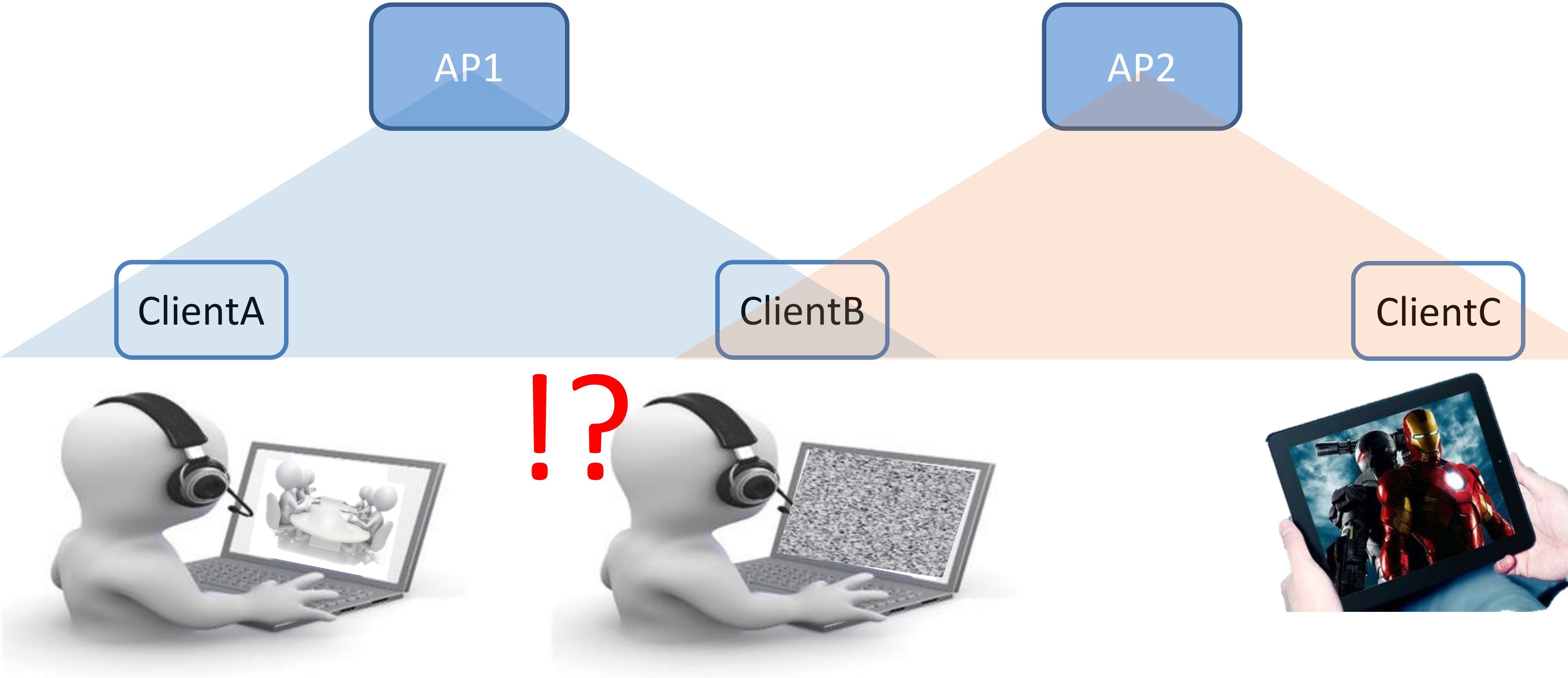
[2]Microsoft Research Asia

[*]Co-Primary Author

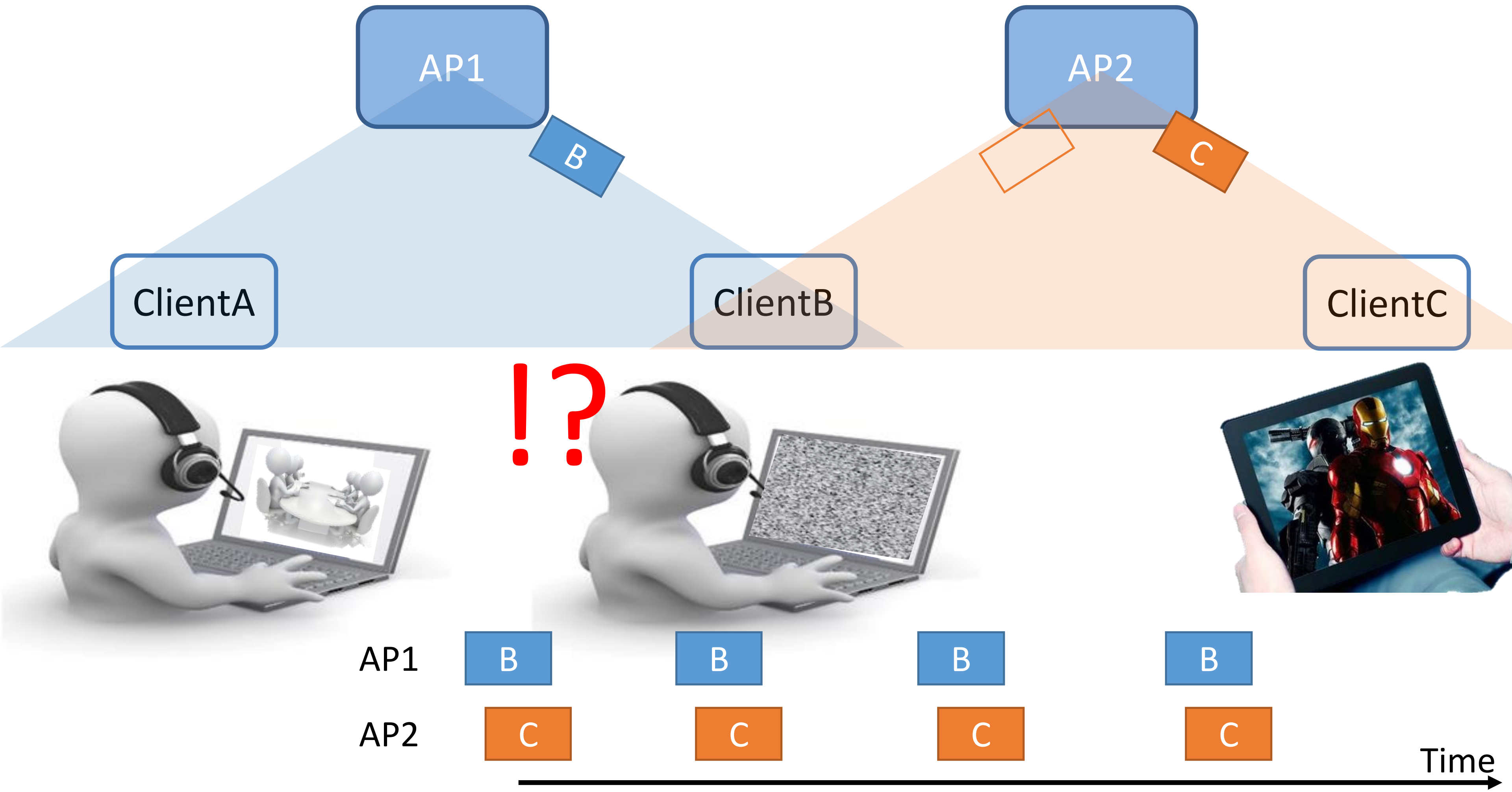
Motivation



Motivation



Motivation



Motivation

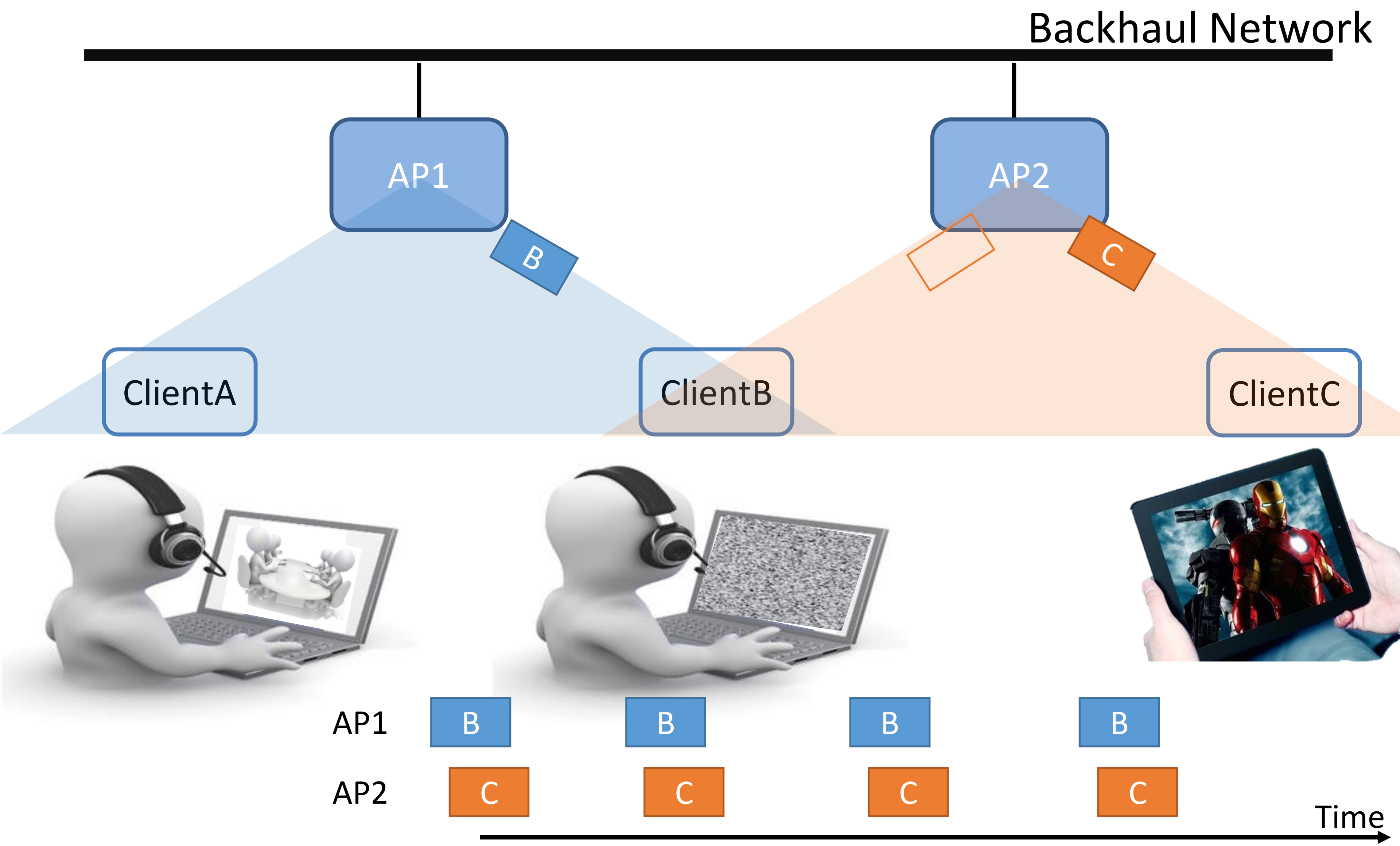
- Limitations of Distributed Coordination Function (DCF) in Current Wireless LANs
  - Weak Interference Management
  - Inefficient Channel Access
  - Lacking Guarantee in QoS

Motivation

- Limitations of Distributed Coordination Function (DCF) in Current Wireless LANs
  - Weak Interference Management
  - Inefficient Channel Access
  - Lacking Guarantee in QoS

- Demands for Higher-Efficiency Wireless Network
  - Proliferation of Wireless Devices
  - Emerging Network QoS Sensitive Applications

How to Fill the Gap?



How to Fill the Gap? -- TDMA

- Arrange the transmission of all the wireless packets in the air to
  - Manage interfering transmissions
  - Reduce contention overhead
  - Provide priorities for QoS transmissions

Research Question

- Arrange the transmission of all the wireless packets in the air to
  - Manage interfering transmissions
  - Reduce contention overhead
  - Provide priorities for QoS transmissions

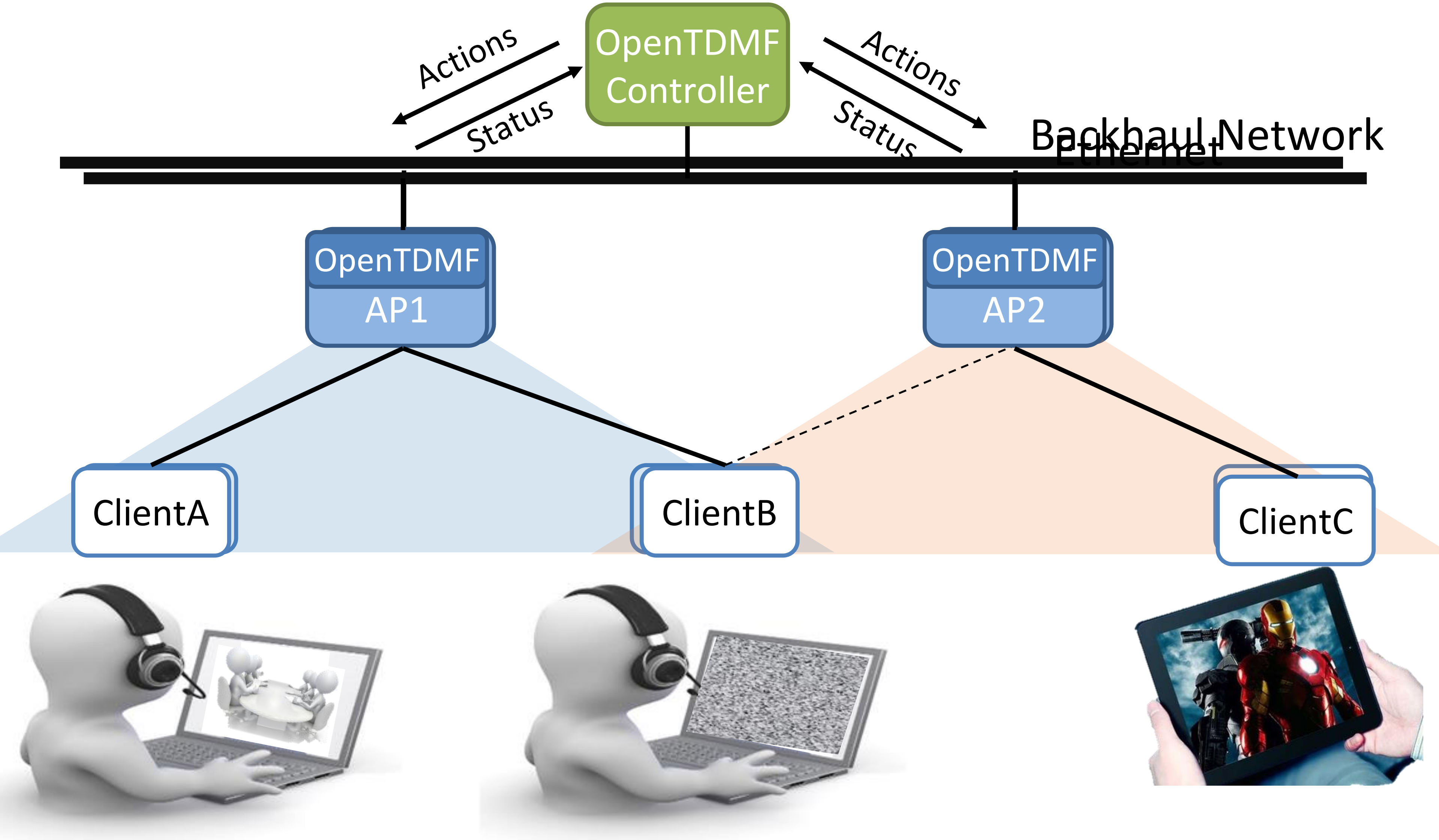
Is TDMA possible with commodity WLAN devices?

Feasibility for TDMA in WLAN

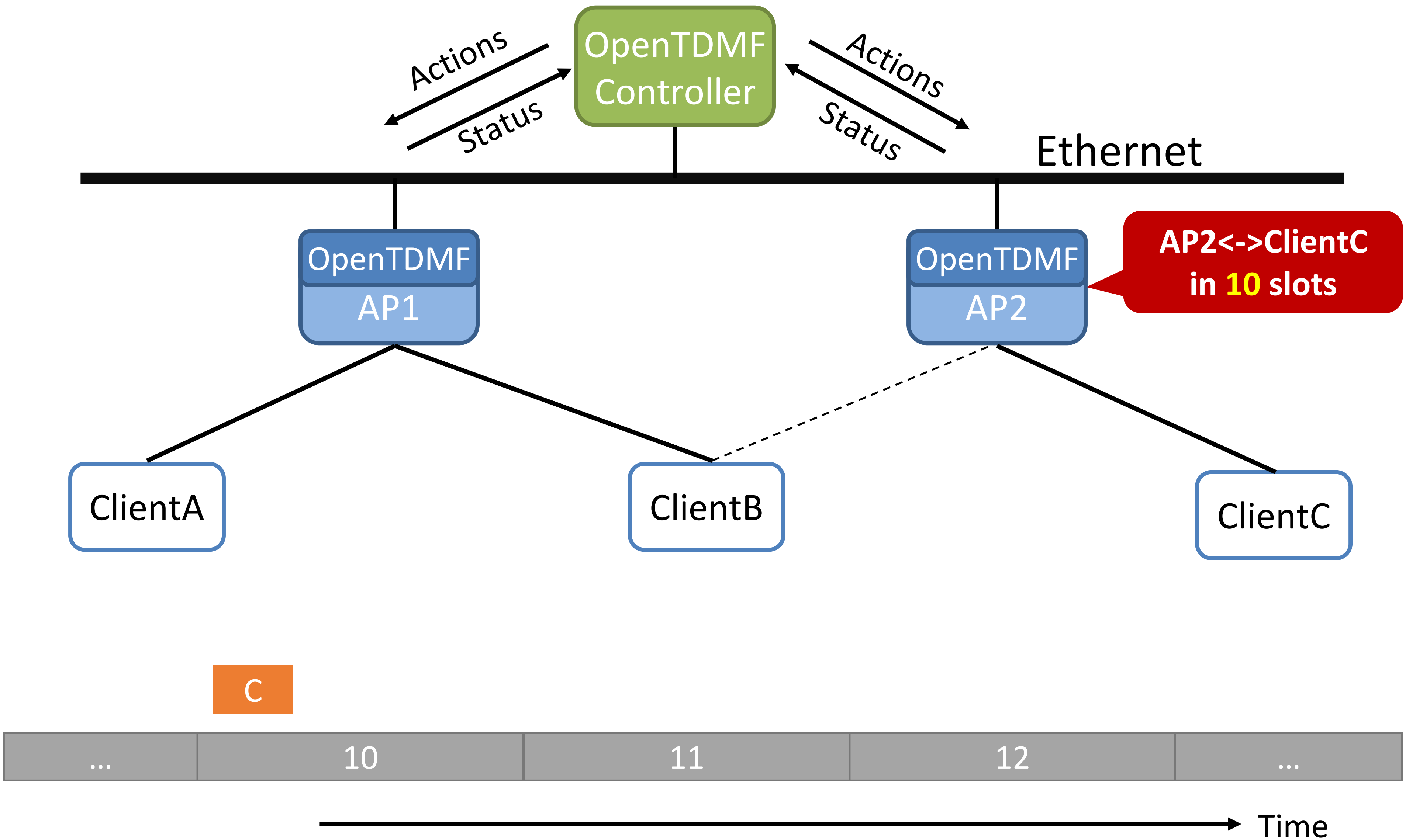
- Time Synchronization
  - Backhaul network
- Scheduling
  - Central controller

Exist

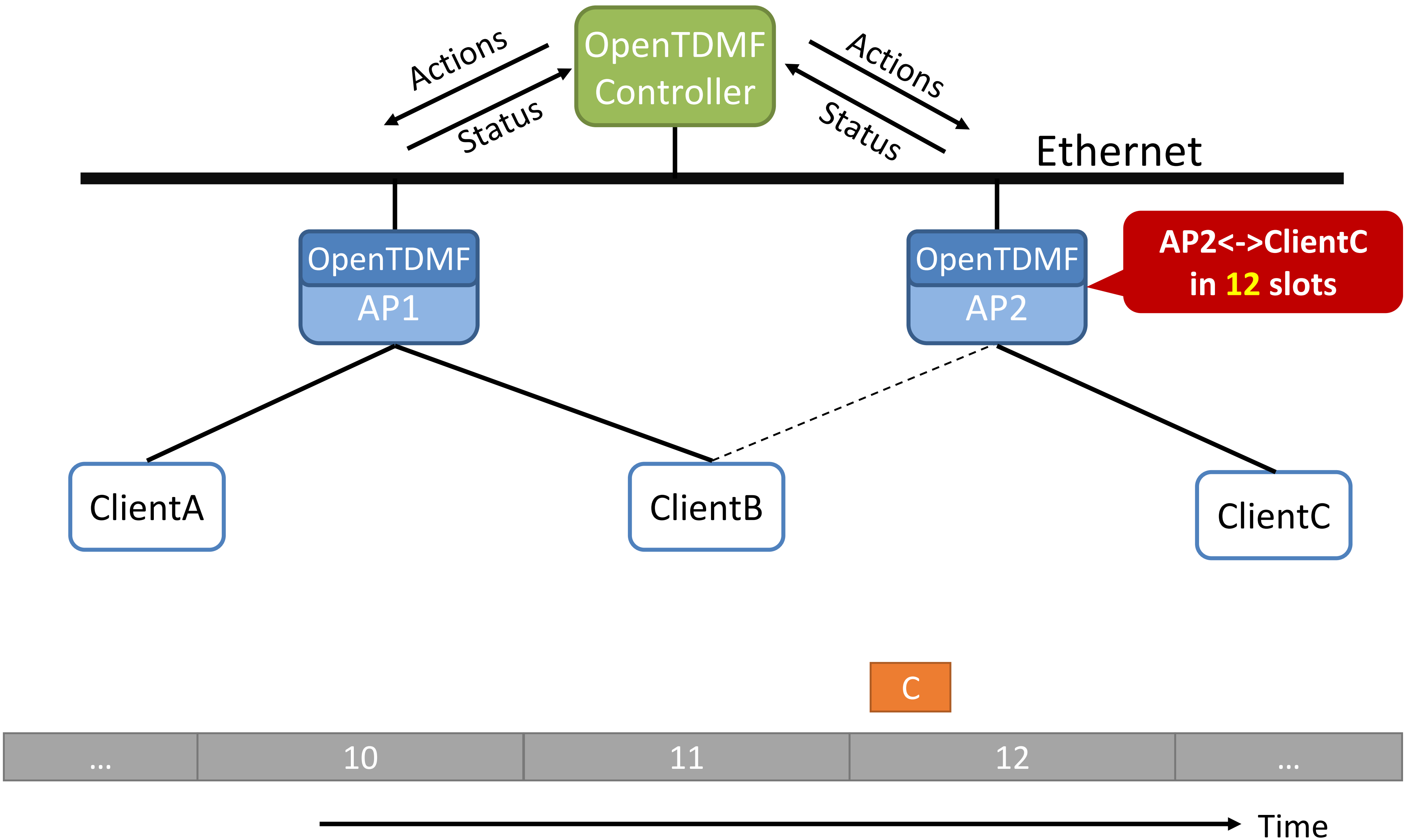
Architecture Borrowed from SDN



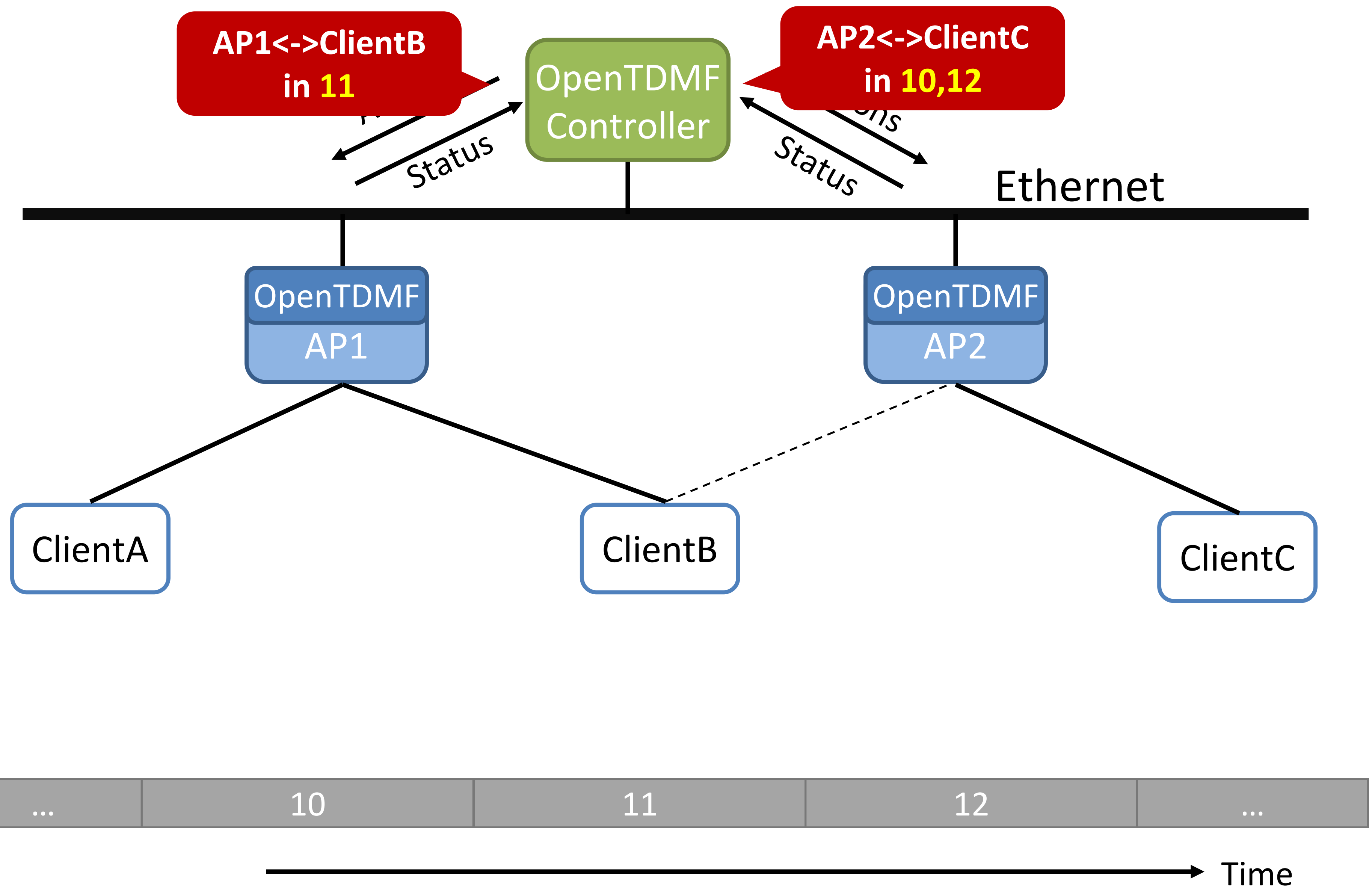
Architecture Borrowed from SDN



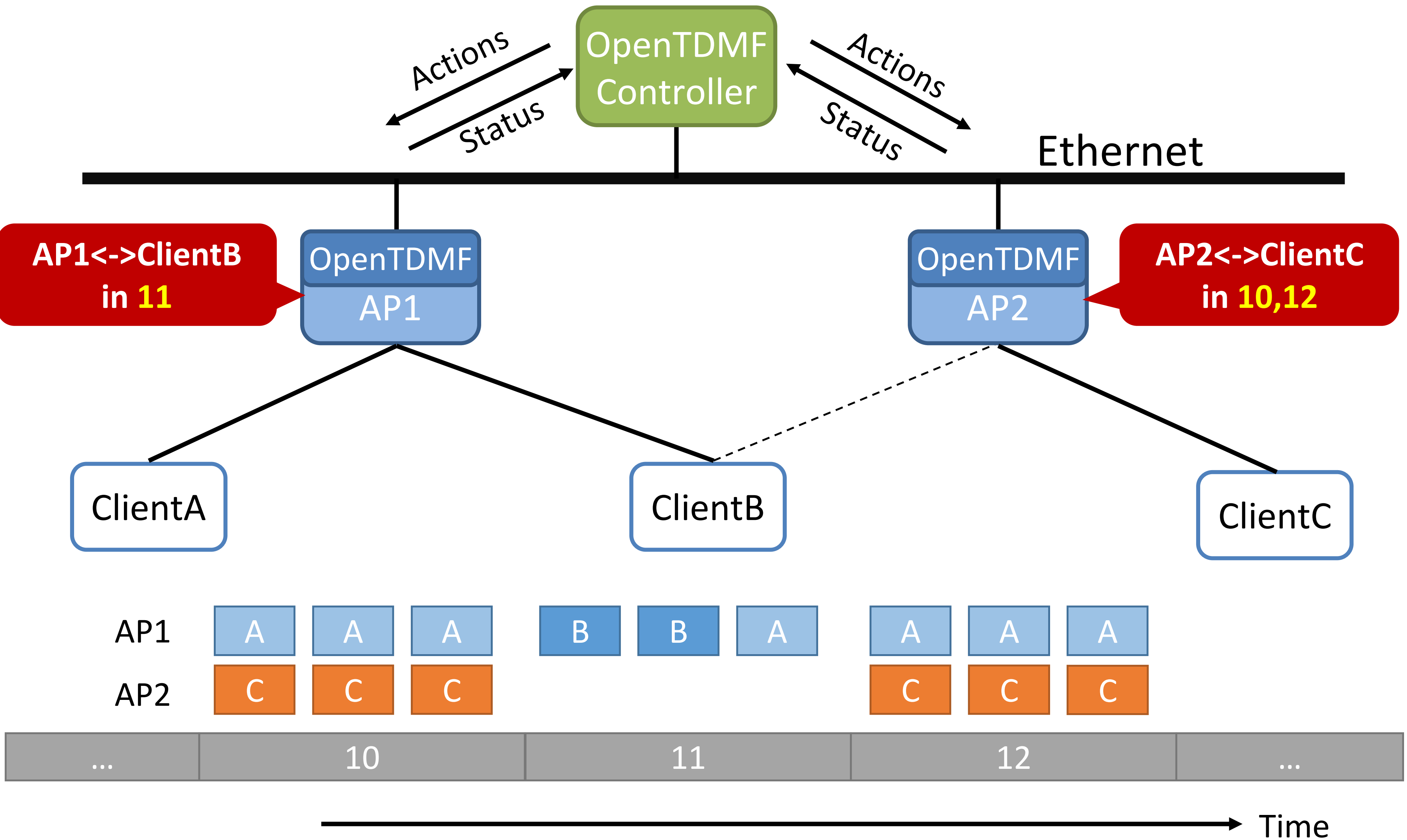
Architecture Borrowed from SDN



Architecture Borrowed from SDN



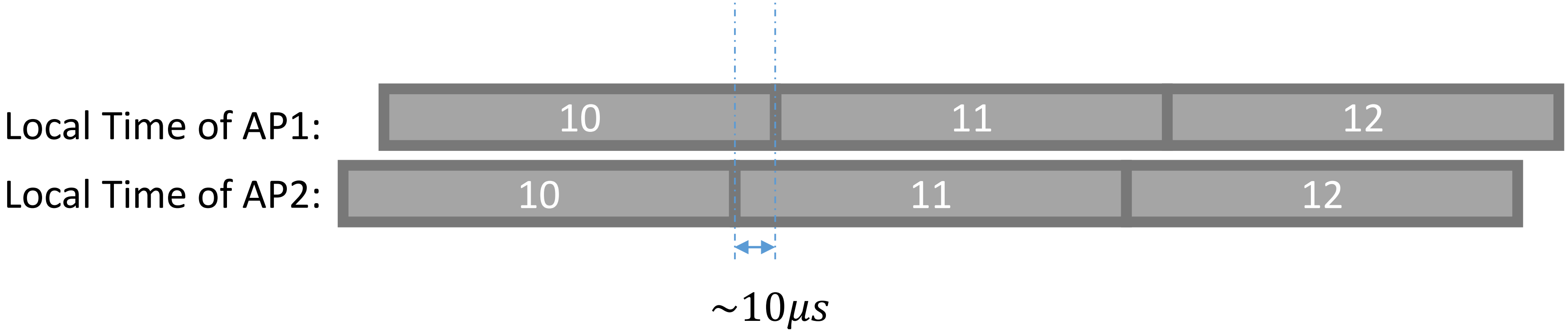
Architecture Borrowed from SDN



Challenges

- Time Synchronization

Commodity WLAN devices lack means for accurate synchronization



Challenges

- Time Synchronization

Commodity WLAN devices lack means for accurate synchronization

- Uplink Scheduling Enforcement

Commodity WLAN devices is designed for distributed access and determines channel access independently

Outline

- OpenTDMF Design
  - Time Synchronization
  - Uplink Scheduling Enforcement
- Experiment Results
- Scheduling Examples
- Conclusion

Outline

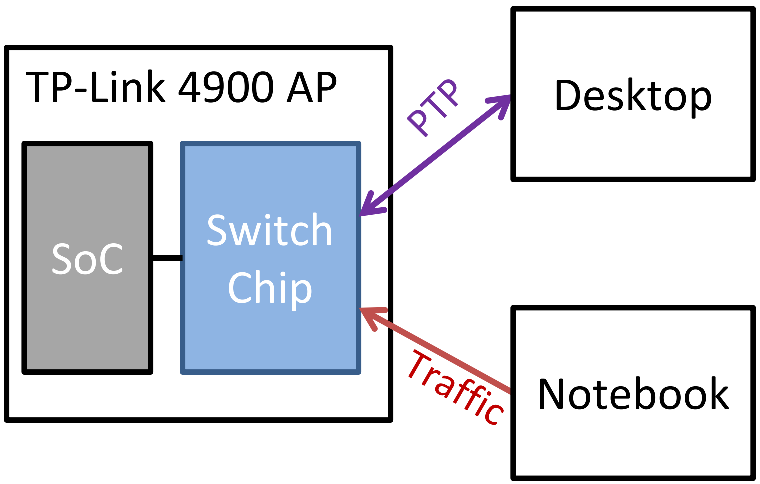
- OpenTDMF Design
  - Time Synchronization
  - Uplink Scheduling Enforcement
- Experiment Results
- Scheduling Examples
- Conclusion

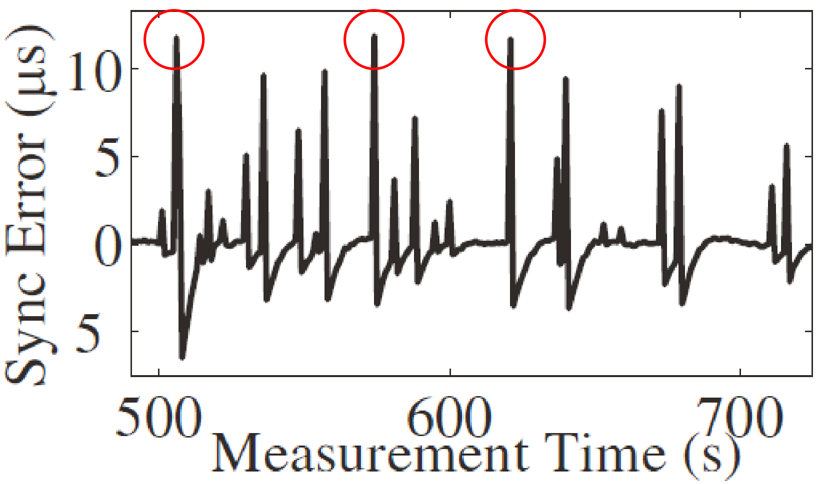
Backhaul Time Synchronization

- Using IEEE 1588 Precise Time Protocol (PTP) to Synchronize the Wired APs
  - Assumption: The network delay is symmetrical

The Problem

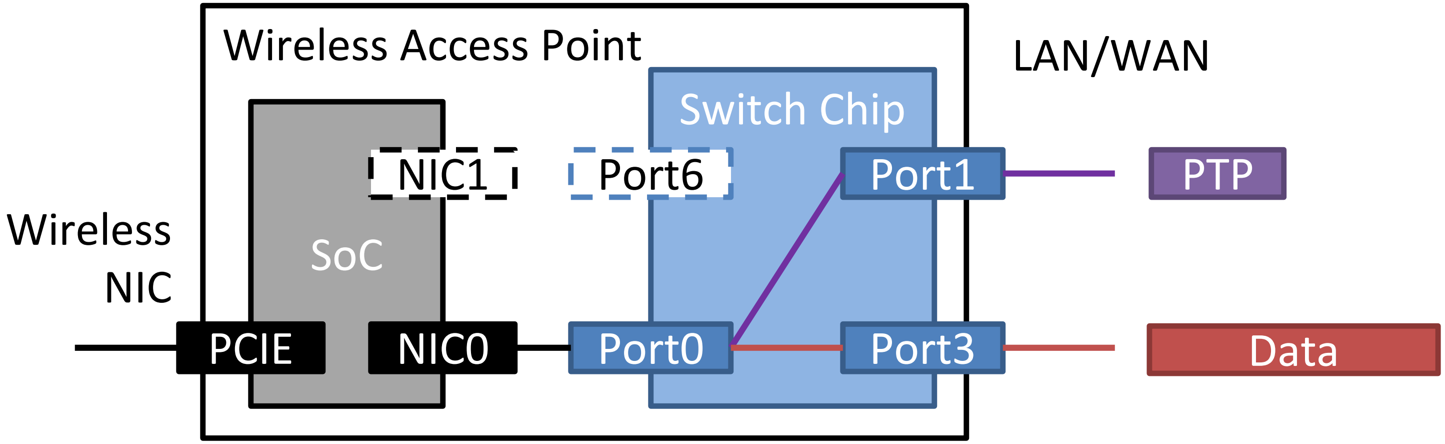
- Large Variation





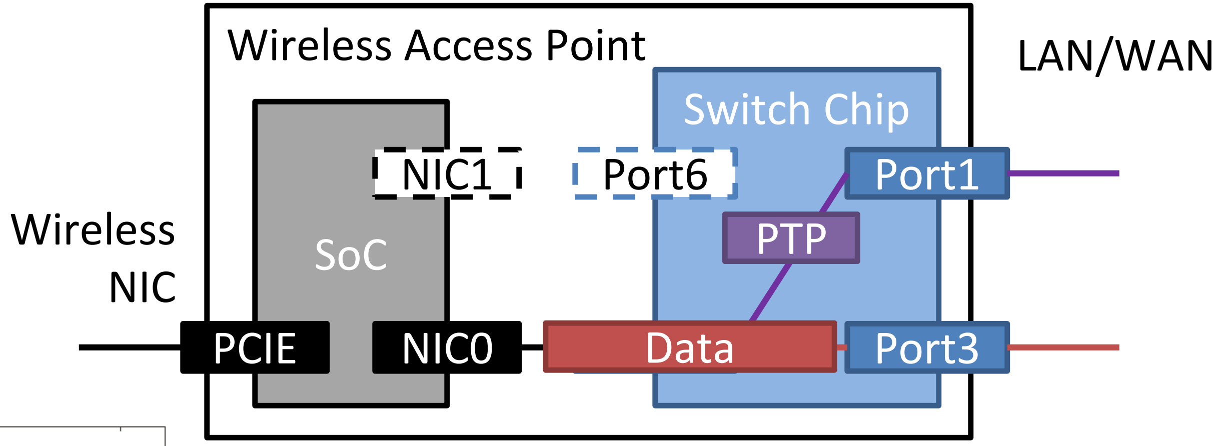
Analyze the Problem

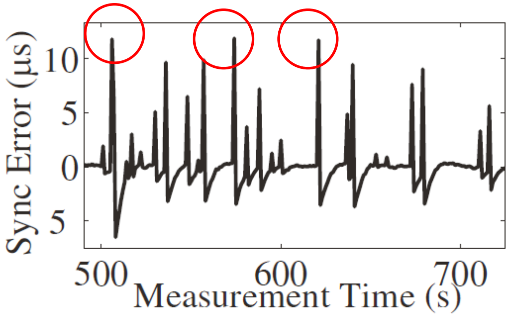
- The Architecture of the Commodity AP Introduces Variance in the Delay Measurement



Analyze the Problem

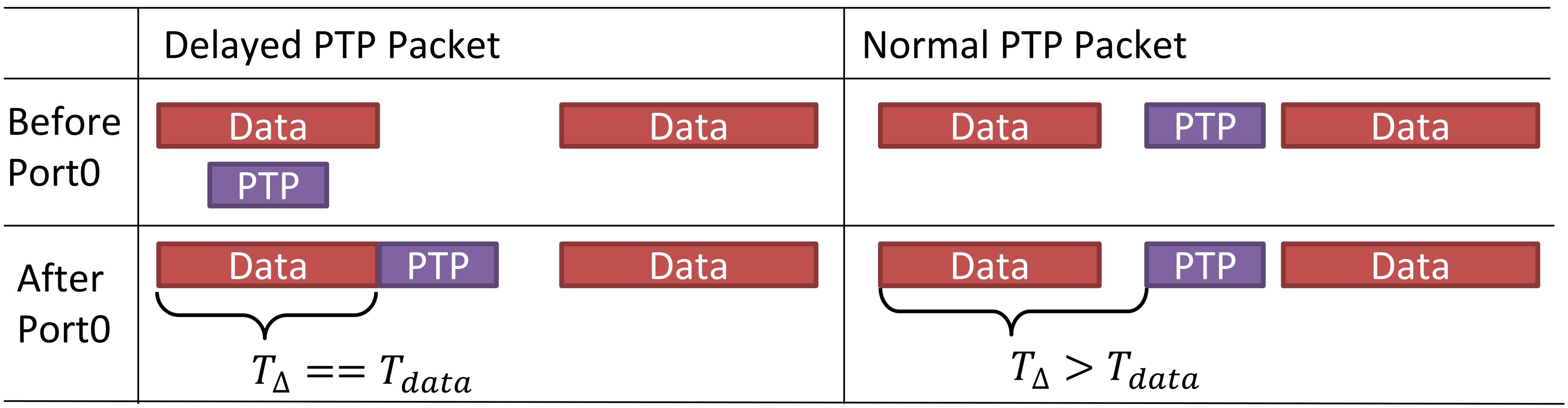
- The Architecture of the Commodity AP Introduces Variance in the Delay Measurement





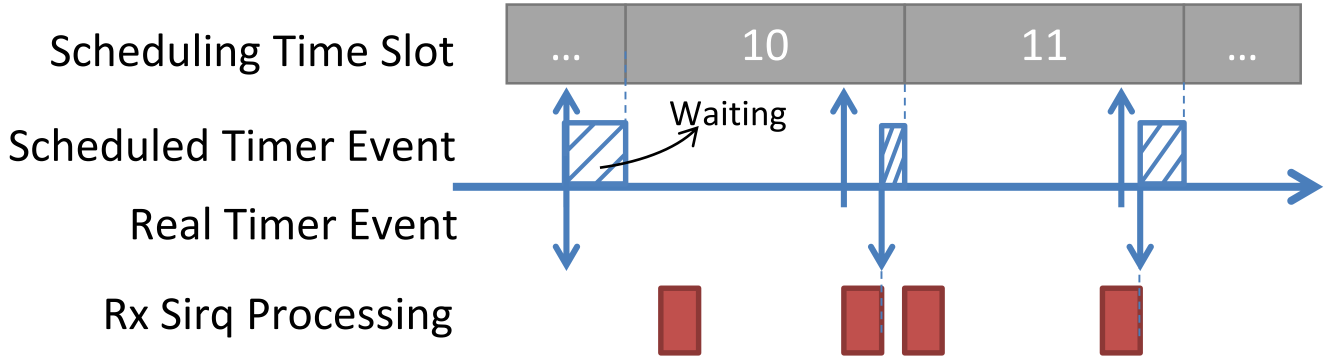
Delayed!

Our Solution



Scheduling Enforcement

- Use busy waiting to ensure accurate software timer event



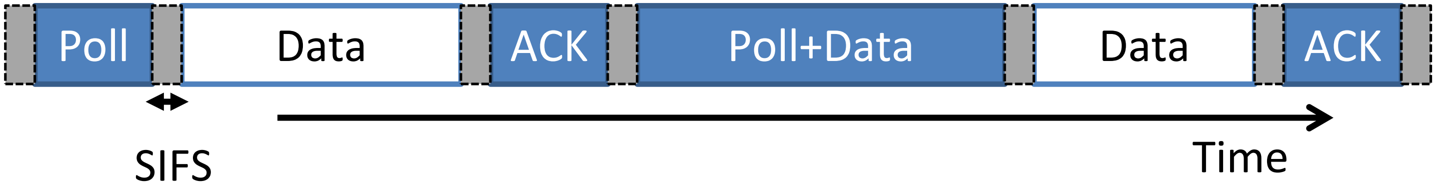
- Use transmission gate handler in WiFi chip to ensure accurate transmission control

Outline

- OpenTDMF Design
  - Time Synchronization
  - Uplink Scheduling Enforcement
- Experiment Results
- Scheduling Examples
- Conclusion

Uplink Control

- Polling for Uplink Transmission
  - Poll packet is a normal packet with a poll flag in the control filed of the MAC header
  - Clients response the poll packet with the uplink data
  - Uplink can be treated as downlink



First Transmission Problem

- AP needs to know about the packet queue information in clients to schedule polling
  - Clients piggyback queue information in every uplink packet
- AP doesn’t know when the client want to transmit the first uplink packet

Group Polling

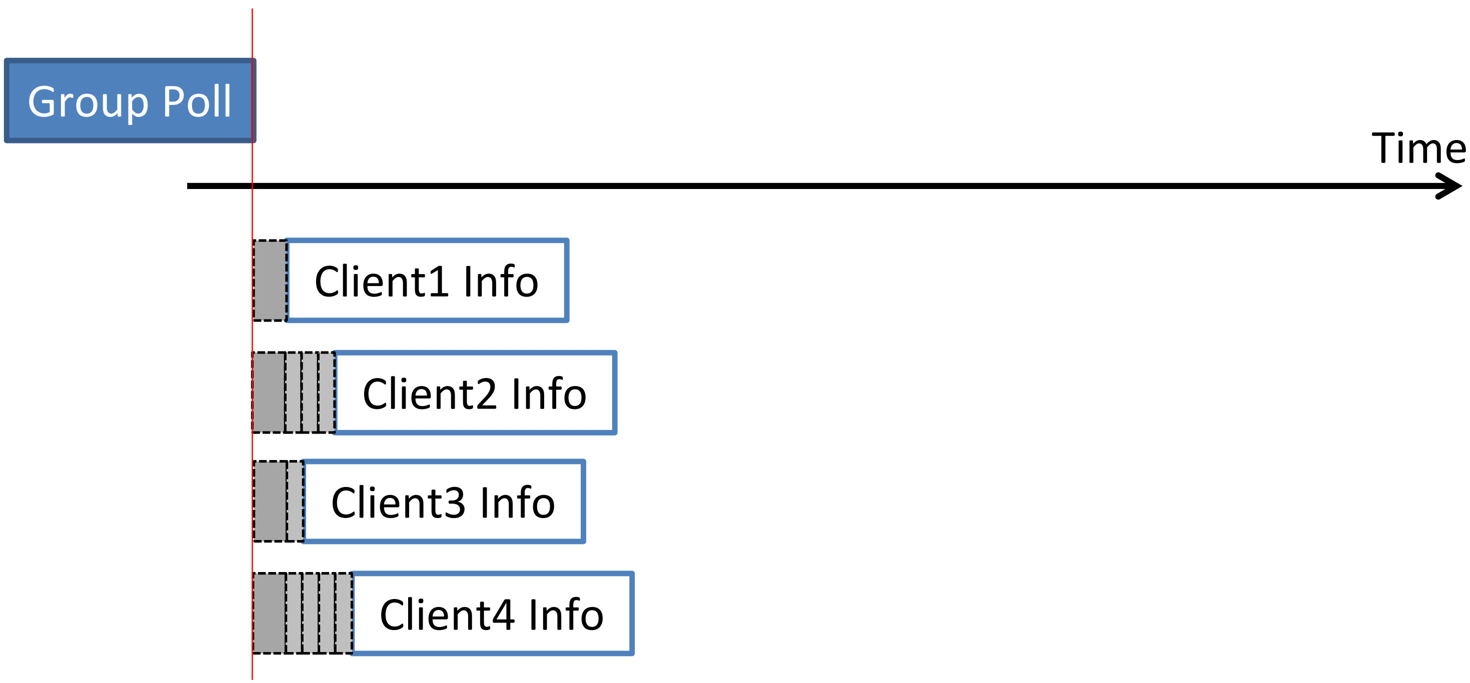
- Group polling for the first transmission
  - Group poll packet is a poll packet with group address

	Client1	Client2	Client3	Client4
Random SIFS	+0 slots	+3 slots	+1 slots	+4 slots

Group Polling

- Group polling for the first transmission
  - Group poll packet is a poll packet with group address

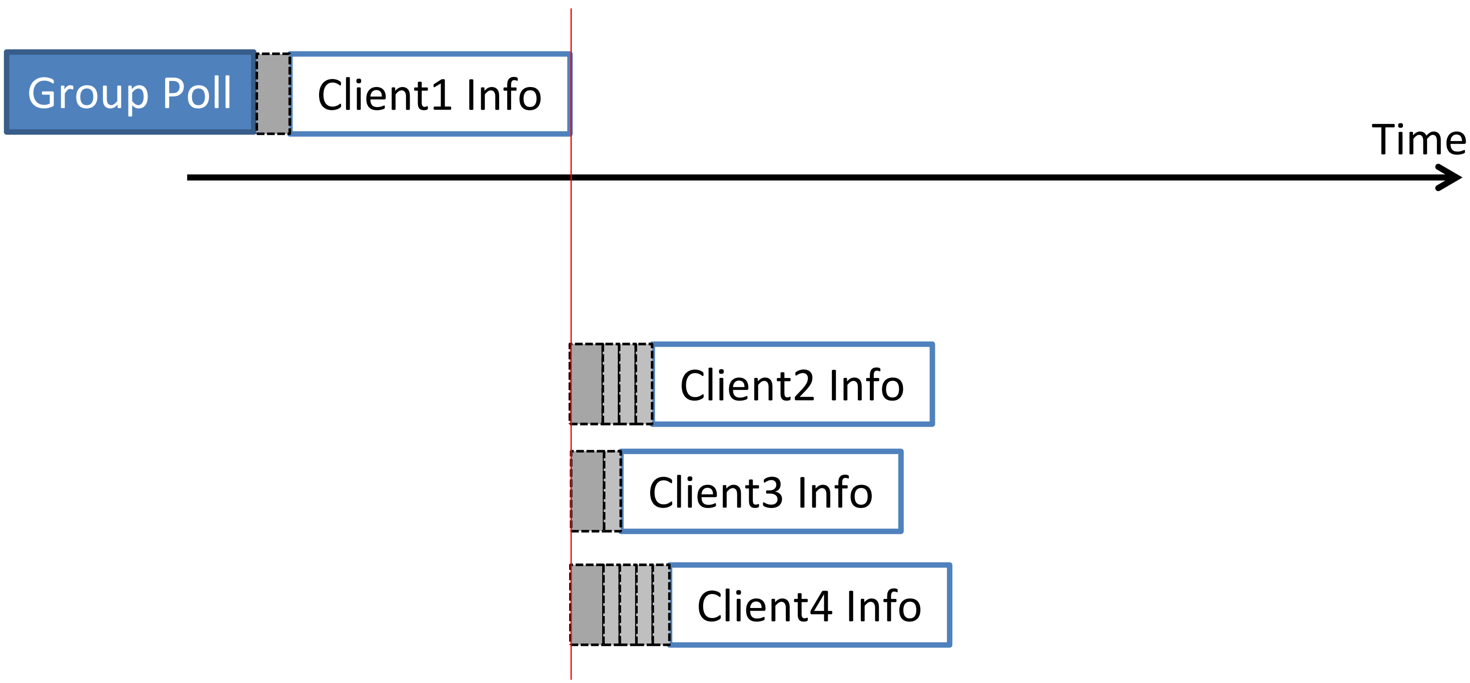
	Client1	Client2	Client3	Client4
Random SIFS	+0 slots	+3 slots	+1 slots	+4 slots



Group Polling

- Group polling for the first transmission
  - Group poll packet is a poll packet with group address

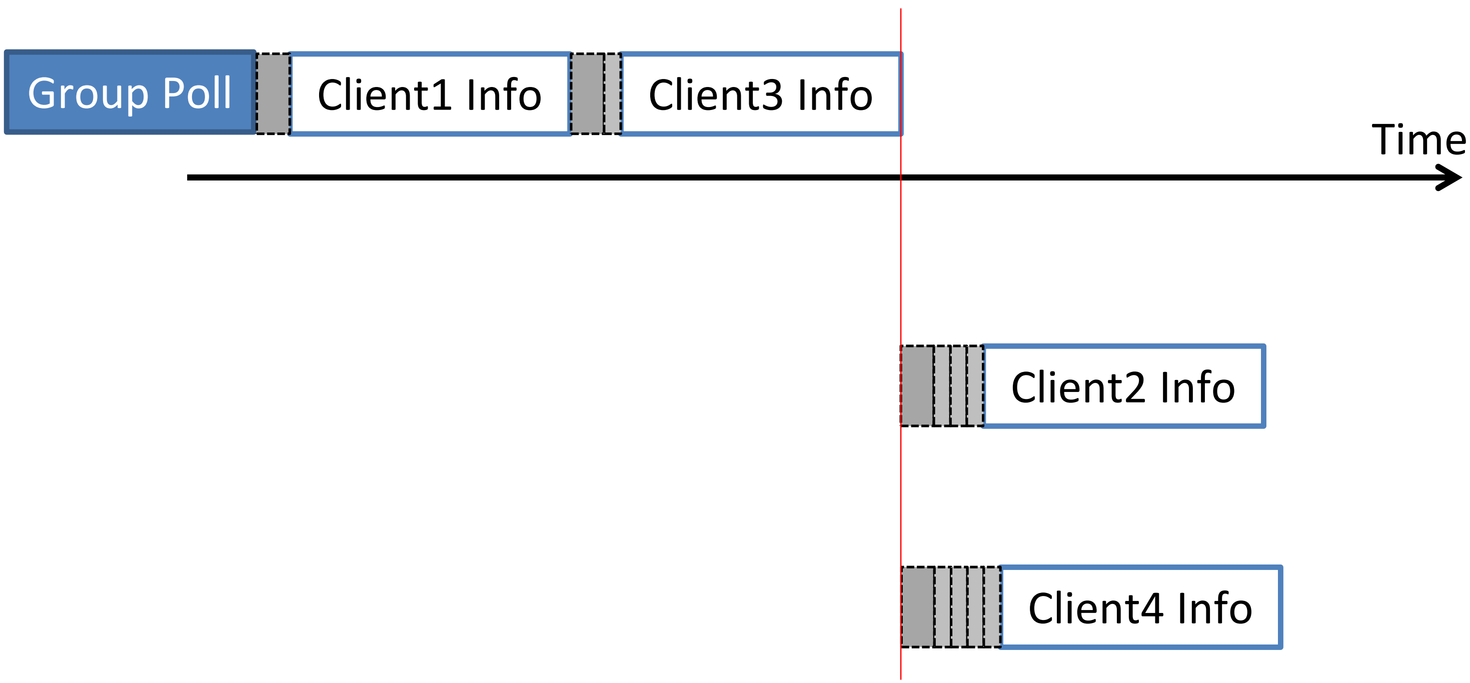
	Client1	Client2	Client3	Client4
Random SIFS	+0 slots	+3 slots	+1 slots	+4 slots



Group Polling

- Group polling for the first transmission
  - Group poll packet is a poll packet with group address

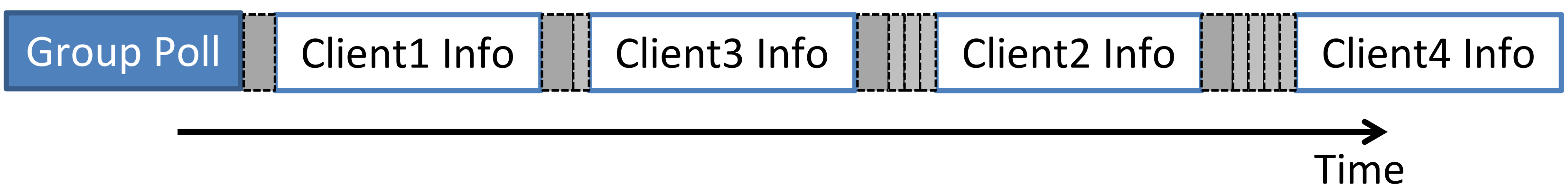
	Client1	Client2	Client3	Client4
Random SIFS	+0 slots	+3 slots	+1 slots	+4 slots



Group Polling

- Group polling for the first transmission
  - Group poll packet is a poll packet with group address

	Client1	Client2	Client3	Client4
Random SIFS	+0 slots	+3 slots	+1 slots	+4 slots



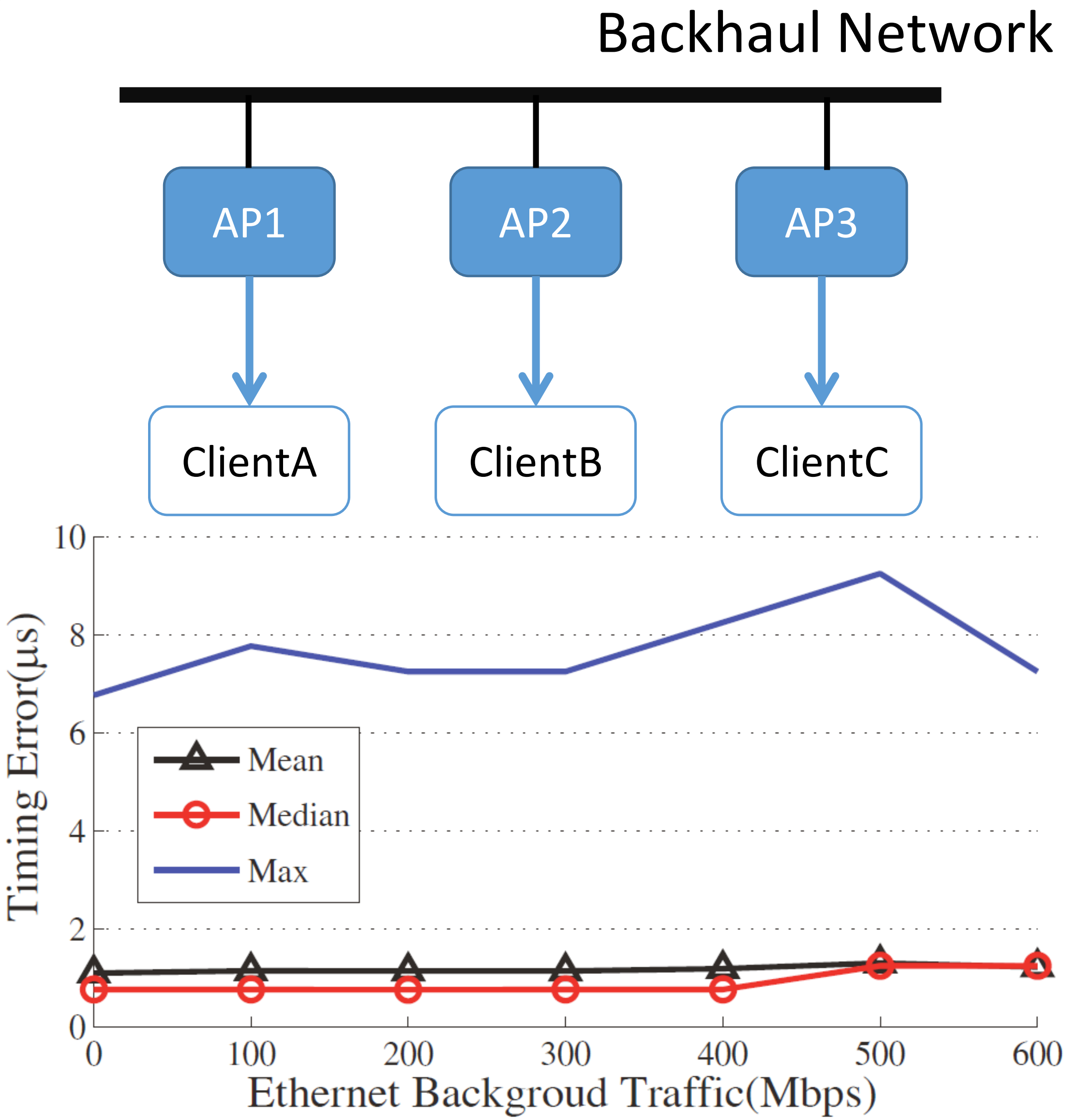
Outline

- OpenTDMF Design
  - Time Synchronization
  - Uplink Scheduling Enforcement
- Experiment Results
- Scheduling Examples
- Conclusion

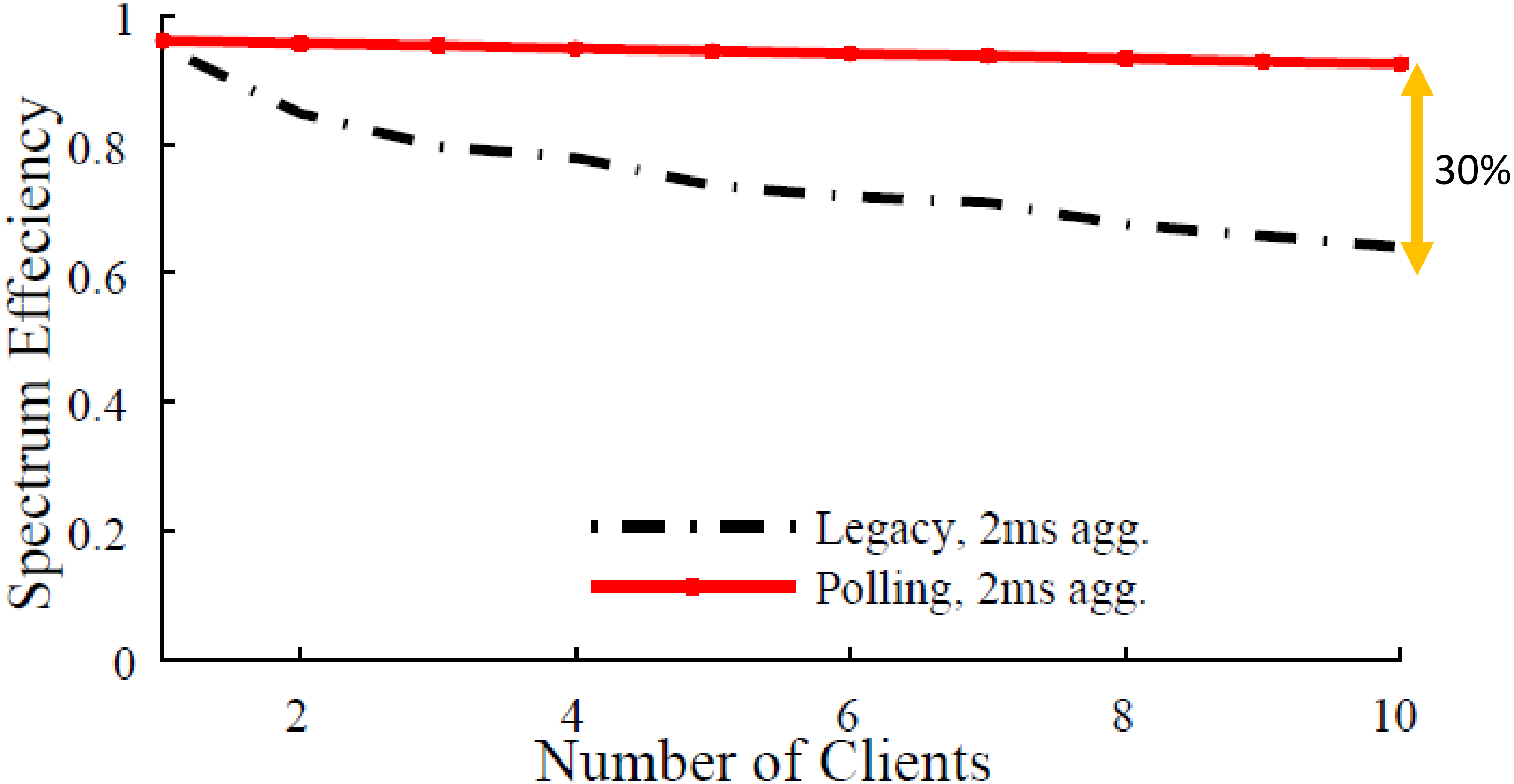
Implementation

- Implemented with TP Link 4900 with Atheros AR9381 and AR9580 WiFi Chip
- Modified ath9k driver
- Modified linuxptp program

Timing Error



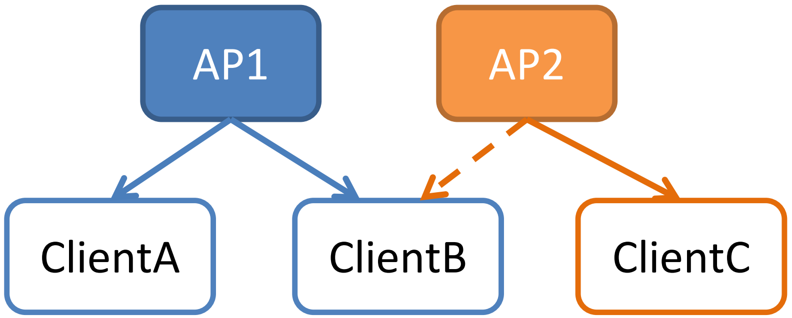
Uplink Efficiency



Outline

- OpenTDMF Design
  - Time Synchronization
  - Uplink Scheduling Enforcement
- Experiment Results
- Scheduling Examples
- Conclusion

Topology and Policy



Knowledge of the Controller

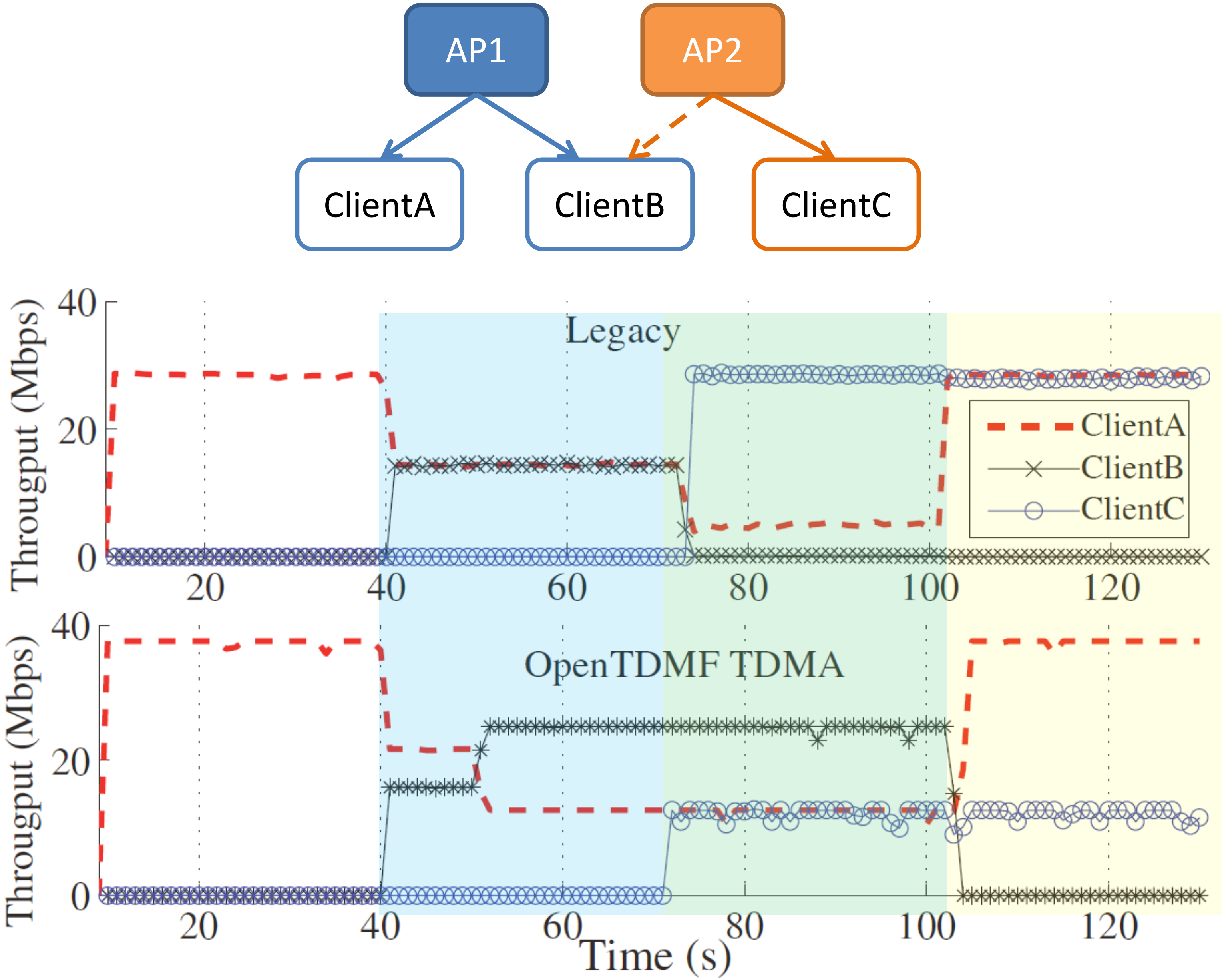
AP1<->B conflicts with AP2<->C & AP1<->B requires high priority

Control Policy for AP1

Flow ID	Time Slots	Priority
AP1<->B	1,2 mod 3	High
AP1<->A	ALL	Normal

Control Policy for AP2

Flow ID	Time Slots	Priority
AP2<->C	0 mod 3	Normal



AP1
AP2
ClientA
ClientB
ClientC
Legacy
OpenTDMF TDMA
Througput (Mbps)
Time (s)
ClientA
ClientB
ClientC

Outline

- OpenTDMF Design
  - Time Synchronization
  - Uplink Scheduling Enforcement
- Experiment Results
- Scheduling Examples
- Conclusion

Conclusion

- Thoughtful study of accurate synchronization in commodity AP
- Enable polling based uplink transmission in commodity WiFi chips
- Build the OpenTDMF system and validate the feasibility of TDMA in commodity WLANs.

Thank you !