PiCasso: Information-Centric Edge Computing Platform for Community Mesh Networks

Adisorn Lertsinsrubtavee
University of Cambridge

IRTF GAIA Meeting, IETF 101, London
22 March 2018
Environment: Data centers
Technology: Heavy weight VMs
Hardware: Hi-end server grade machines

Facility: Reliable electricity supply. Hi-speed LANs. Stable network connectivity
EDGE COMPUTING VS COMMUNITY NETWORKS

Service Providers

Data center

Internet

Gateway

Constrained Backhaul

Community Network

➤ Low speed bandwidth
➤ Intermittent link
➤ Poor Quality of Service
➤ High latency
➤ Service unreachable
DISTRIBUTE SERVICE TO THE EDGE

Running Service at the edge
- Easy hosting and migration of local services could be highly beneficial in community networks.

Lightweight Virtualisation technology
- Allow very small (e.g., 2MB) virtual machine services to be migrated across a network with very little cost.

Overcome the intermittent connectivity
- Rather than hosting services in fixed predetermined locations, service can be retrieved from anywhere.
PICASSO: SYSTEM OVERVIEW

Service

Service

Service

Service Controller

Networking (NDN - DTN)

Edge Network

Edge Network

Automatic Decision Making
- Service Deployment
- Where to and When to place service?
  - QoS, SLA
  - Network condition
  - Hardware resources

Monitoring System
- Node usage (CPU, Memory)
- Network condition

Smart Forwarding
- Named Data Networking (NDN)
- Delay Tolerant Networking (DTN)
  - Name based routing
  - Dynamic in-network caching
  - Store-and-Forward fashion
PICASSO: ARCHITECTURE
PICASSO: MODULES

- Monitoring DB
- Service Repo
- Monitoring Manager
- Decision Engine
- NFD Forwarding
  - TCP
  - UDP
  - DTN
  - Network Interfaces

- Service Controller

- NFD Forwarding
  - TCP
  - UDP
  - DTN
  - Network Interfaces

- Forwarding Node

- Docker Engine
  - Service
  - Service
  - Service

- Monitoring Agent

- Service Execution Gateway

- NFD Forwarding
  - TCP
  - UDP
  - DTN
  - Network Interfaces
PICASSO DEPLOYMENT IN GUIFI.NET
Select replica to host the service

- **Service**: webserver (300 bytes payload)
- **Factor**: k=1, k=2
- **Tools**: Apachebench (generate concurrent HTTP requests)
- **Metrics**: HTTP Response time
- **Comparison**: Random strategy

- **HANET** selects better node to host the service (enough CPU, better bw)

- **HANET** k=2, 90% of requests achieve response time less than 500 ms
SERVICE DELIVERY COST IN GUIFI.NET

Delivery Service image from Service Repo to all SEGs

- **Service**: debian image (145 MB)
- **Metrics**: Delivery Cost
- **Comparison**: HCN (IP unicast)

- PiCasso node automatically retrieves the data chunks from the nearest cache
- HCN: Every request is forwarded to Service Repo

- **Average Delivery cost**: PiCasso (70s) HCN (155s)
- **54% improvement**
TRAFFIC CONSUMPTION OF SERVICE DELIVERY

Heavy traffic consumption links

HCN

PiCasso

➤ PiCasso consumes less traffic consumption up to 43.24%
THANK YOU  Q&A

adisorn.lertsinsrubtavee@cl.cam.ac.uk
https://github.com/AdL1398/PiCasso
Planning to come to IETF 103 in Bangkok? wait there's MORE.
AINTEC 2018
ASIAN INTERNET ENGINEERING CONFERENCE

November 12-14, 2018
Bangkok, Thailand
https://interlab.ait.ac.th/aintec2018/

Image by Warawut Srisopark - Own work, CC BY-SA 3.0/70% Opacity