Control, processing, data.
When models mess around implicits and explicits

a Socratic Method discussion

Rui L. Aguiar ruilaa@ua.pt
Universidade de Aveiro
Instituto de Telecomunicações

Objective

To reflect on what are we doing with

Flexible – “able to be easily modified to respond to altered circumstances.”

Programmable – “able to be provided with coded instructions for the automatic performance of a task.”

Infrastructures – “the basic physical and organizational structures and facilities needed for the operation of a society or enterprise”

And what we are doing with them…

… hopefully relying in the knowledge of the audience.
the background:

Software-Defined Networks?

Developed at Stanford in 2008

Concept “new”
Although is hard to define what is the difference from previous telecom strategies.

Decoupling data plane from control plane

Overlay network
Keywords for the future

• In-networking computing
• Native AI processing
• Distributed computing interconnection
• Embedded Edge computing

Lost separation between computing and networking.
Lost separation between application and in-networking functions
Lost separation on location and functions

Let's ask a friend for help

The Death of Socrates
Painting by Jacques-Louis David (1787), neoclassic
Developed at Stanford in 2008

*Concept "new"
Although hard to define what is the difference from previous telecom strategies.

Decoupling data plane from control plane

Overlay network

IETF COPS (98)
3GPP PCRF (05)
IETF Forces (01)
Considering SDN…

Openflow:
• Expected some sort of “functional commonality” in the “boxes”.
• The focus was on separation of control and dataplane, plus a strict controlling architecture concept

These concepts existed in different styles in the past.

The differences are on the distribution **DEGREE** and the overall **INTERRELATION** between Control and Data elements

_Did we ever had a non-programable infrastructure? (Please discuss it with CISCO)._

Extending the concept to network elements…

What have we found:
• In reality: extremes will (do) not work
• A trade-off will be required between ideal SDN and network performance
• Transition will be challenging:
  • Green-field deployments (native) vs. Evolution (integration)
  • Operation of hybrid boxes (switches)
So nice of you to bring this topic of programming into the discussion. This P4 looks really great.

Can we jointly reflect one minute on this?

Do not overcomplicate: let's start with OF

OpenFlow
Open-Source configuration and control protocol
Manufacturer-independent
Objective: enabler for researchers to develop new networking protocols
So...

A packet arrives and is looked into a LUT?

Yeah...

And how fast you need to do that?

Hum... let's say we have an edge with 10Gbps input, and headers with 200 bits, this gives 20nsec to process the look-up for runtime processing.

Excellent! Seems a lot of time. How many searches can you do in that time?

Well, there are some optimum mechanisms for search, with some previous requirements in sorting previously...

Do not overcomplicate, please, we are just trying to understand the process...

Hum... with a 5GHz clock, with a cached LUT, this means about 100 searches, and in a linear approach, this mean a LUT about 200 entries.

Excellent again! And now for processing?

With an average packet size of 10Kbits, then we have about 1Mpackets/sec, so about 1 μsec, leading to about 5000 clock cycles.

Thank you for explaining.

You are telling me that in all these discussions you are considering about 200 different types of processing code, with a complexity of about 5K clock cycles?

This is now clear. Finally I understand what you want to do in the edge!

Can we do this exercise for the core now?
But I am fine if we stay on the “ideals world” for these “flexible programmable infrastructures”. Can we jointly reflect one minute on this?

How is this structured then?

Inherits visions from IPSphere
Reflects Operators deployment
Can we restructure this in a slightly different way?

For simplifications let’s call this as basic functions:
- Routing/connectivity
- Security/access control
- QoS/Differentiation

We can define network slices with these features.

But are the links all the same?
And how do you bootstart?

This “dataplane” appears only when you have data.

→ Before the dataplane, you need to have a way of putting a control plane operational.
→ For having a control plane operational, you need to have a way of communicating at the physical level.
→ For communicating at the physical level, you need to have a way of at-least communicating properly between two different nodes, in a meaningful way.
→ For communicating in a meaningful way, you need to have running code that does understand the most primitive needs of communication.

But then… we are always stuck with different levels of overlays?

P4 is the same as… a CDN, … a P2P, or something similar?

What can I conclude

It looks like you lost your bearings

You do not know what computations you are doing in the network
You do not know what you can be doing in a specific network point
You do not know that you live in a technology continuum, where all aspects are always present in some component, and you keep mentally falling for the simpler binary views of the world

For the future – if you want to have 6G as a successful technology – if you rethink your traditional ways of looking into computing and communicating, of looking into central and distributed computation, you cannot simply change one of your assumptions. It you mess around, look at the new system as requiring a complete overhaul on the way you think about it.
Flexible Programmable Infrastructures
online, December 1st 2022

Control, processing, data.
When models mess around implicits and explicit

a Socratic Method discussion

Rui L. Aguiar ruilaa@ua.pt
Universidade de Aveiro
Instituto de Telecomunicações

Positions open for PhD students and researchers

creating and sharing knowledge for telecommunications