Opening Statement

Stefan Schmid (TU Berlin)

CNSM 2022 Panel @ Thessaloniki, Greece

Opening Statement

Stefan Schmid (TU Berlin)

CNSM 2022 Panel @ Thessaloniki, Greece

Kudos: Oliver Michel

Opening Statement

Stefan Schmid (TU Berlin)

CNSM 2022 Panel @ Thessaloniki, Greece

The Programmable Data Plane: Abstractions, Architectures, Algorithms, and Applications

OLIVER MICHEL, Faculty of Computer Science, University of Vienna, Austria
ROBERTO BIFULCO, NEC Laboratories Europe, Germany
GÁBOR RÉTVÁRI, Budapest University of Technology and Economics (BME), Hungary
STEFAN SCHMID, Faculty of Computer Science, University of Vienna, Austria

Programmable data plane technology enables the systematic reconfiguration of the low-level processing steps applied to network packets and is a key driver in realizing the next generation of network services and applications. This survey presents recent trends and issues in the design and implementation of programmable network devices, focusing on prominent architectures, abstractions, algorithms, and applications proposed, debated, and realized over the past years. We elaborate on the trends that led to the emergence of this technology and highlight the most important pointers from the literature, casting different taxonomies for the field and identifying avenues for future research.

Kudos: Oliver Michel

Exciting Times

Great time to be a CNSM/TNSM researcher!

- ⇢ Today: great choice of diverse in-network acceleration and processing platforms
  - ⇢ E.g., high-density reconfigurable match-action dataplanes, smartNICs, kernel bypass frameworks, etc.

Exciting Times, But…?

Great time to be a CNSM/TNSM researcher!

- ⇢ Today: great choice of diverse in-network acceleration and processing platforms
  - ⇢ E.g., high-density reconfigurable match-action dataplanes, smartNICs, kernel bypass frameworks, etc.

- ⇢ But a challenge: how to design a service management framework to harness these diverse resources? How to ensure a coherent service over such heterogeneous infrastructure?

Exciting Times, But…?

Great time to be a CNSM/TNSM researcher!

⇢ Today: great choice of diverse in-network acceleration and processing platforms

  ⇢ E.g., high-density reconfigurable match-action dataplanes, smartNICs, kernel bypass frameworks, etc.

⇢ But a challenge: how to design a service management framework to harness these diverse resources? How to ensure a coherent service over such heterogeneous infrastructure?

⇢ Maybe this is even impossible?

I think…

- ***Trend*** is clear and unstoppable: more and more specialization and diversity: SmartNICs vs Tofino vs …
- Based on ***different interfaces***, capabilities, devices speak different programming languages, etc.
- There is ***no alternative***: have to be unified by network management, and provide single interface
- This will be hard and ***challenging***: high hardware diversity (e.g., monitoring everything vs sketch), components in hardware vs in software, etc.

and that …

- ⇢ This is ***feasible***, and ***depends on application***, e.g., in monitoring context first solutions exist
- ⇢ Example: ***Sonata*** is based on abstract stream processing graph, which is ***compiled*** to diverse hardware
- ⇢ And we are already ***on a good track*** to such unifying frameworks! E.g., ***PISA architecture***!
  - ⇢ We need more work in this direction!
  - ⇢ Management frameworks do not have to know about every detail of the device.

Interesting Questions (1)

- ⇢ What is a good ***abstraction layer***? A language?
- ⇢ How much ***domain knowledge*** about devices is needed?
- ⇢ How to obtain information? It would be great to have ***inspection***, to have an automated agent, or so.
  - ⇢ Or do we need to map facts like “Tofino has 12 stages” manually?!
- ⇢ How can we determine whether/how an application can be ***split*** across hardware? Maybe by just looking at the ***workload***? Or by simulation?
  - ⇢ May be simpler than program analysis! And does not require a unified data plane language…

Interesting Questions (2)

- ⇢ What is ***meaningful*** to split? Not everything can/should be run on a switch for example! (In contrast to some research…!)
- ⇢ I believe in ***self-adjusting solutions***: can we auto-tune our systems? Also depends on application, e.g., Neural Networks may be automatically reduced in size („model pruning“) to be runnable on a device.
- ⇢ Role of ***AI***?

Looking forward to discuss!