Efficient Scheduling for Concurrently Executed Network Packet Processing Applications on Heterogeneous Hardware Architectures

Giannis Giakoumakis, Eva Papadogiannaki, Giorgos Vasiliadis and Sotiris Ioannidis
FORTH

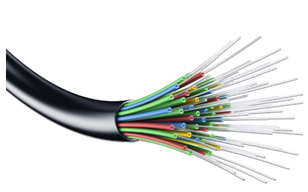
Motivation: As network traffic speeds and volumes grow how could we utilize system resources that run typical network applications in order to increase performance while decreasing power consumption?

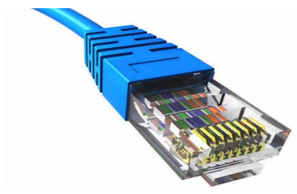
Challenges: Different levels of heterogeneity, each of them introducing its own challenges!

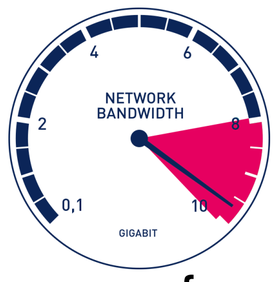
- Typical network applications may have different needs in terms of resources
- In fact, the workload can also affect how resource hungry an application will be

	CPU	iGPU	dGPU
MD5	–	✓	✓
AES	✗	–	✓
CRC	–	✓	–
DPI	✗	✓	✓
SHA1	✓	–	✗

✓ best fit
– avg fit
✗ worst fit

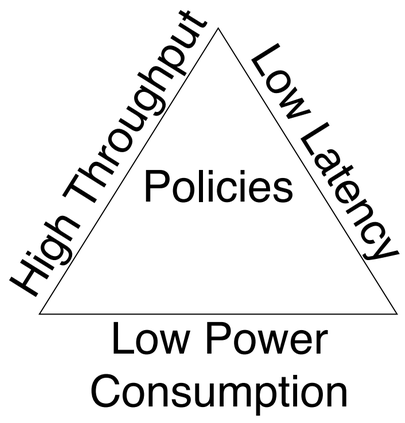




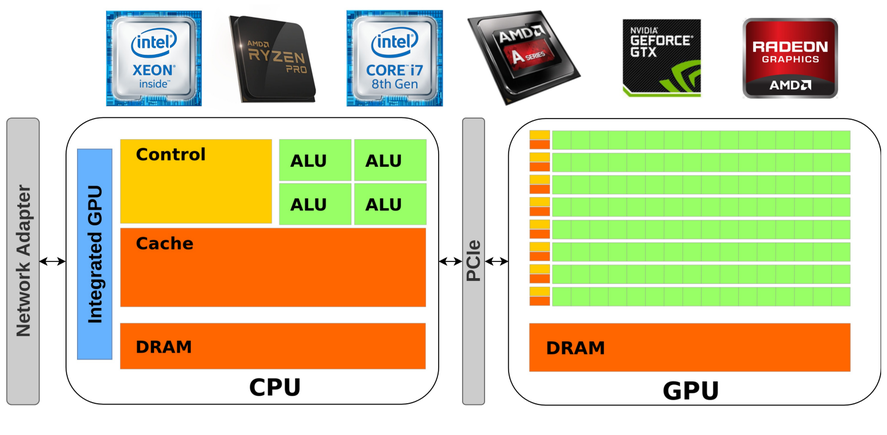


- Network level heterogeneity occurs from traffic fluctuations
- Since the network provides the workload, real-time systems should take such changes into consideration

- Human level heterogeneity as the end user decides how the system should be refined based on personalized needs



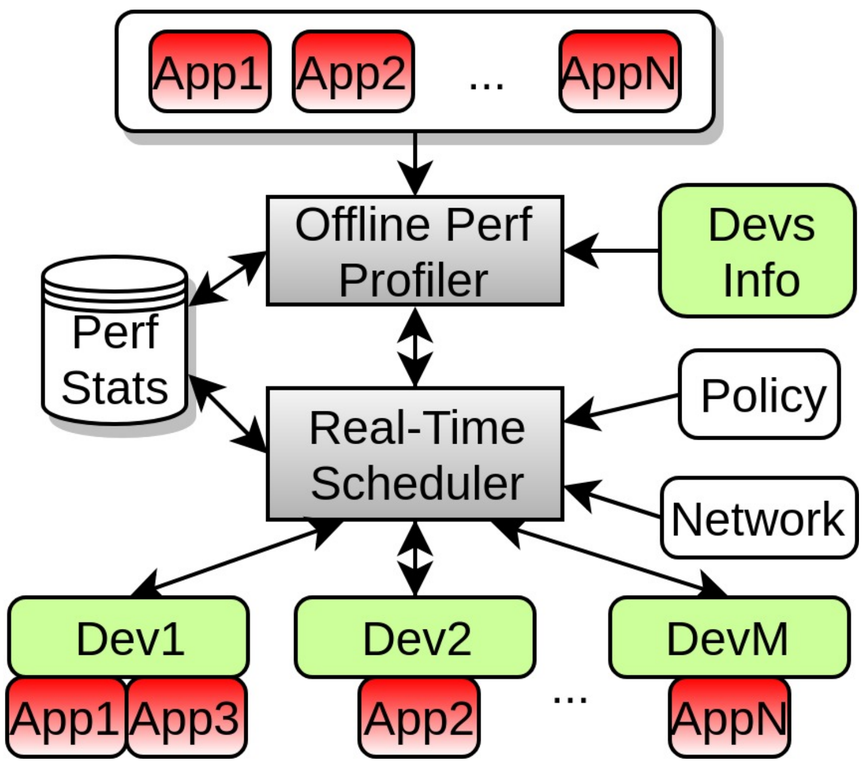
- Heterogeneity in commodity hardware (many different vendors & microarchitectures)
- Different device types, each one performs well under specific workloads
- Profiling of commodity hardware devices is not unified, in need of specialized hardware to measure power consumption



Design & Implementation

- We implement a mechanism that can run multiple applications on each physical device of a system by using OpenCL framework. To achieve that we either use OpenCL fission extension to pin applications to specific CPU compute units or use a stream-like technique to concurrently execute multiple application instances on GPU devices.
- Then we implement a real-time scheduler coupled with an offline profiler to efficiently distribute workload to each device
- Finally to measure power consumption we implement a power API, thus detaching any special sensors from the PSU

High-level System Overview



Goals of our scheduler
Efficient scheduling of concurrently running network packet proccessing applications after performing offline analysis
Ability to map N applications to M devices to achieve low power consumption, hide PCIe copies latency, etc
Ability to periodically measure power consumption of every device without using any specialized hardware or external sensors
Adapt as quick as possible to network traffic rate changes by rescheduling the workload on each device.
Be flexible and scalable (user can easily add more applications, change the underlying hardware by adding or changing devices, etc)

This work has been partially supported by the European Commission under the Horizon 2020 Program through the I-BiDaaS project under Grant Agreement 780787.