

# Taming Software Complexity in Emerging Computer Architectures for Data Intensive Workloads

Aayush Patel, Elliot Wasem, Wei Wang, Dr. Antonio Barbalace

apate139@stevens.edu, ewasem@stevens.edu, wwang88@stevens.edu, abarbala@stevens.edu  
Department of Computer Science, Stevens Institute of Technology



## Today

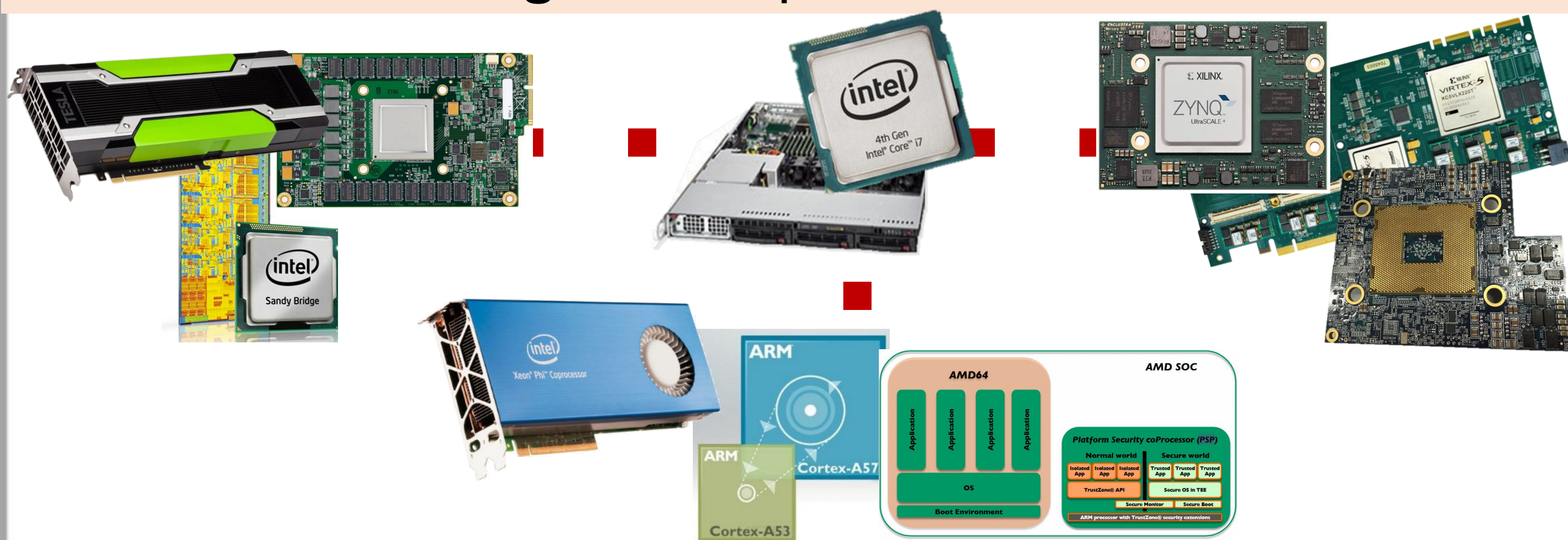


- With the advent of big data, TBs of data are produced per day from a myriad of internet-connected devices
- Artificial intelligence and machine learning technologies are keeping a stake in extracting useful information from this data

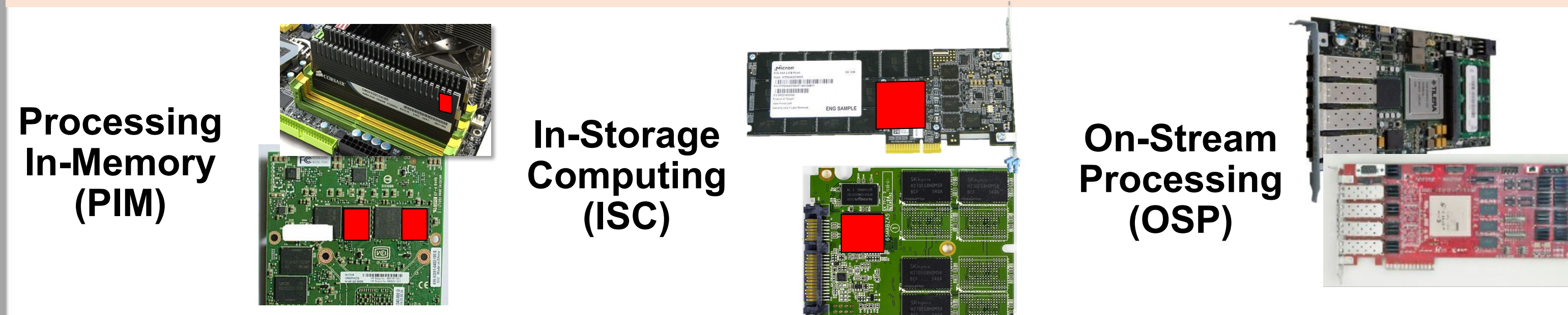
## Hardware Problem - Solved

- **Data extraction must be done in reasonable time in order for the data to be useful for decision making**
- Traditional computer hardware cannot keep up with the challenges, therefore modern hardware includes

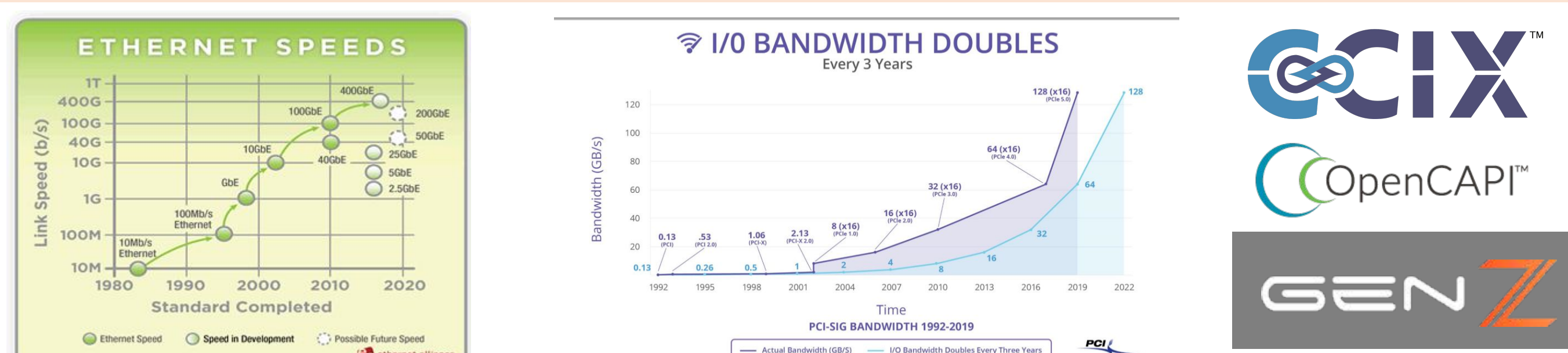
- **Heterogeneous processors**



- **Near data processors (NDP)**

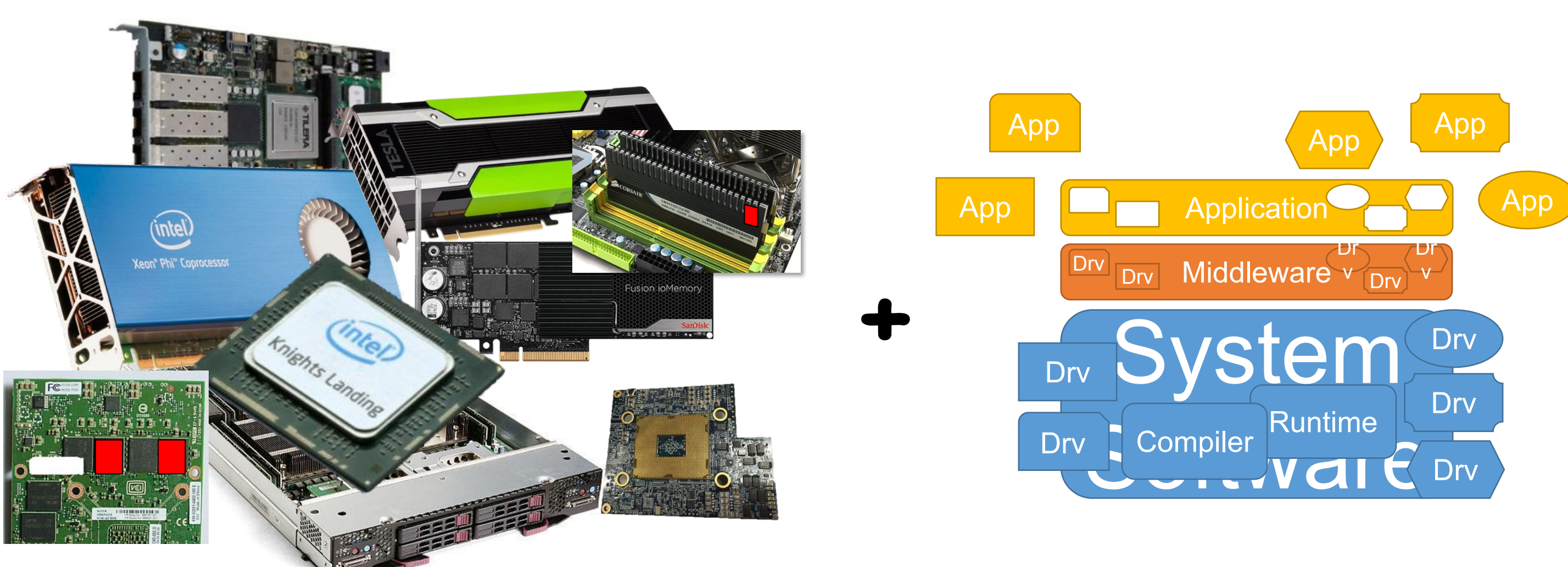


- **Faster network and interconnect**



## Software Problem - Open

- **Software plays a fundamental role to extract data in a reasonable time**
- But the above hardware is very difficult to use and extremely complicated to program
- Thus **making hardware very hard to exploit it**



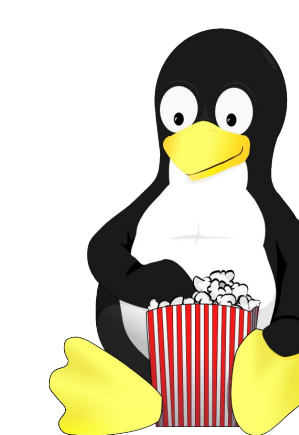
## Idea

**Abstracts away hardware differences,**

- allowing a program to run on multiple processors (CPU, GPU, FPGA, etc.) as if they were the same
- Move program on different processors at **any time**
  - to improve performance, energy, determinism, fault-tolerance, security, etc.

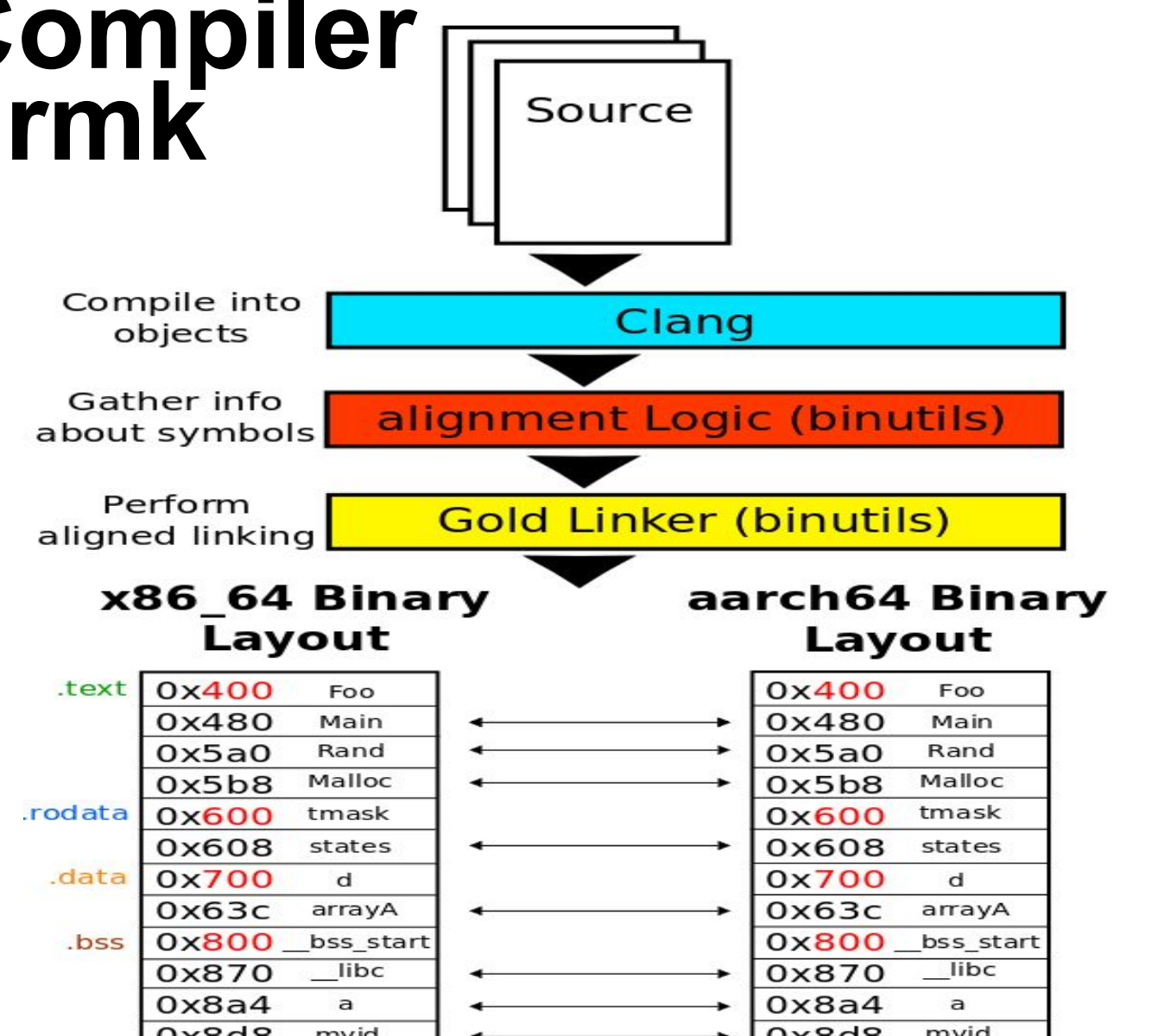
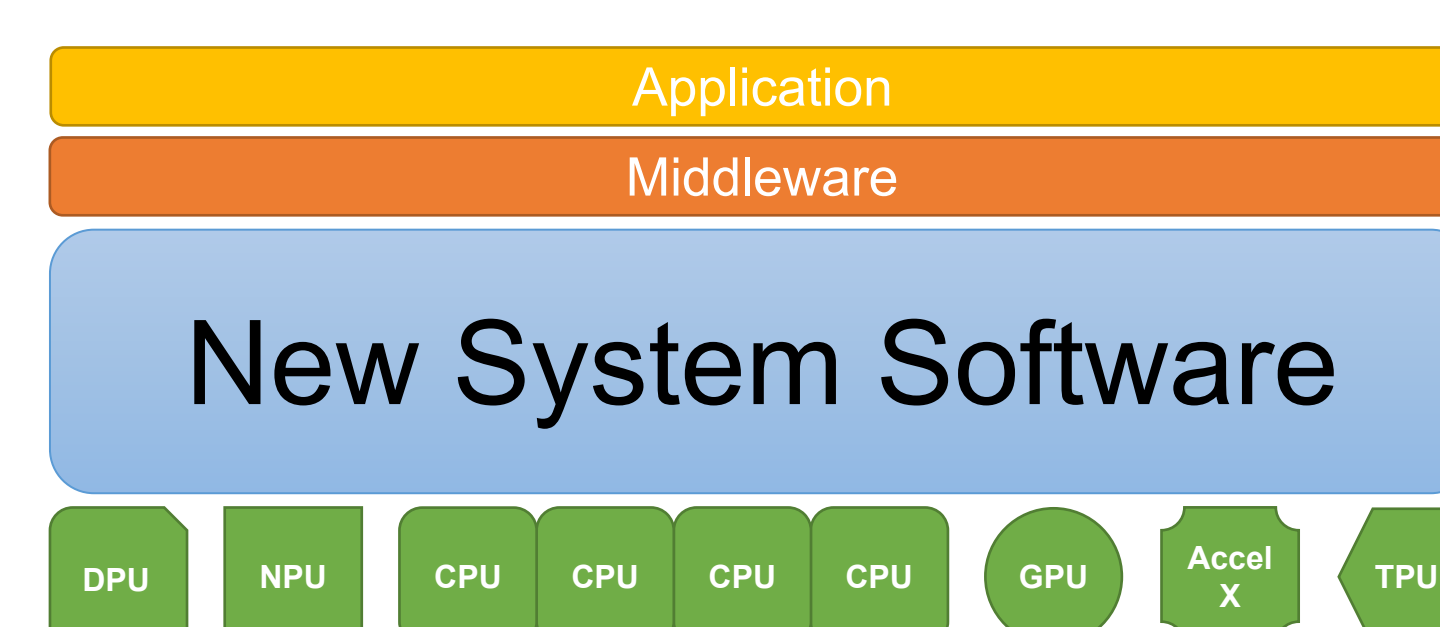
## Bridging Heterogeneity and NDP

- Different ISAs have incompatible instruction sets, so a program compiled for one ISA cannot run on another
- To circumvent this issue, binaries for programs can be produced for all available ISAs on a system and combined into one program file
- Steps can then be taken to ensure that smooth migration can occur between the different processor architectures during process execution
- Distributed shared memory to provide a consistent and uniform address space



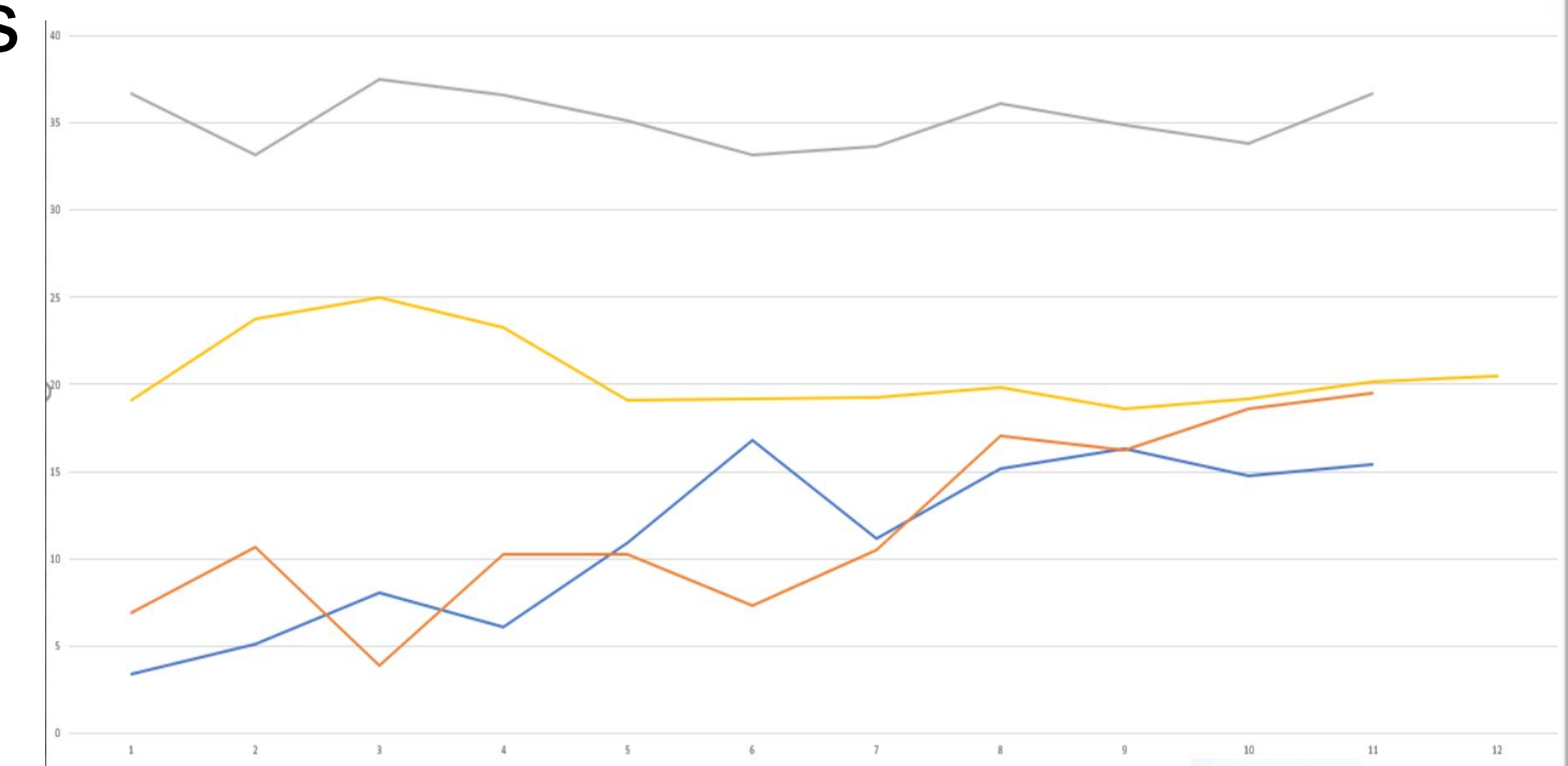
Popcorn  
Linux OS

Popcorn  
Compiler  
Frmk



## Handle New Networking and Interconnects

- As interconnects are becoming increasingly faster CPUs are becoming the bottleneck
- Applying Popcorn Linux's approach on intelligent NICs to enable higher data transfers (up to 160GbE)
- At the same time Popcorn Linux allows ease of programmability of the computer system
- Possibly reducing the time to market and the cost of emerging systems



## Acknowledgment

This research is supported by the **US Department of Defense** under contract N00014-16-1-2104. We thank our colleagues from **Virginia Tech** who provided insight and expertise that greatly assisted the research.

