Aalto University
School of Electrical
Engineering

# High-performance software directed packet routing

*Performance Measurements*

Nuutti Varis, Jukka Manner

# Agenda

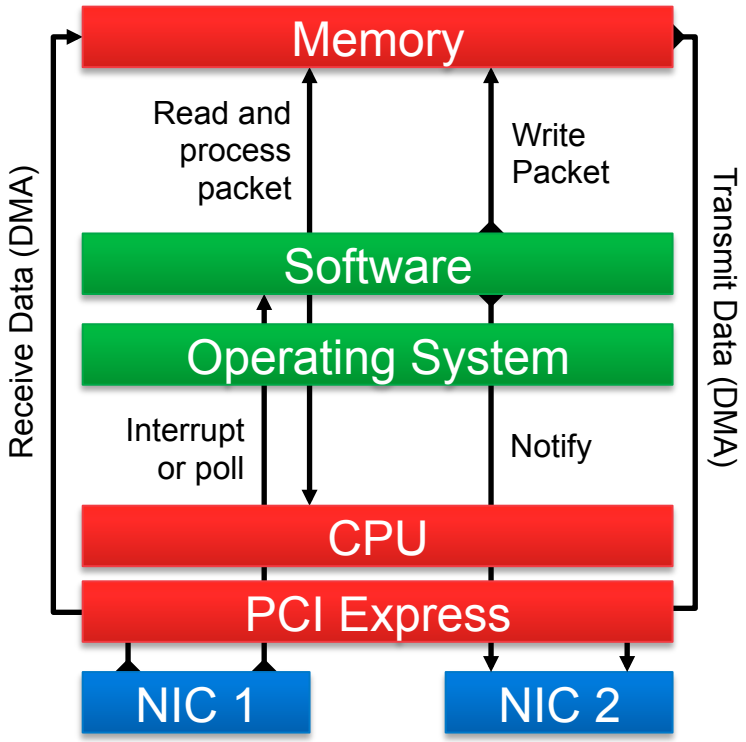
- Motivation
- Generic OS packet forwarding
- Test environment
- Results
- Conclusion

# Motivation

- Cheap general purpose off the shelf hardware has become powerful in recent years
- High-performance, commodity hardware based packet processing uses
  - An off-the-shelf hardware platform, e.g. processor(s), memory, NICs, …
  - A general purpose OS, such as Linux
  - Specialized packet processing software and NIC driver

- *We were interested in comparing the performance of the last two Intel microarchitectures in different scenarios*

# Generic OS packet forwarding

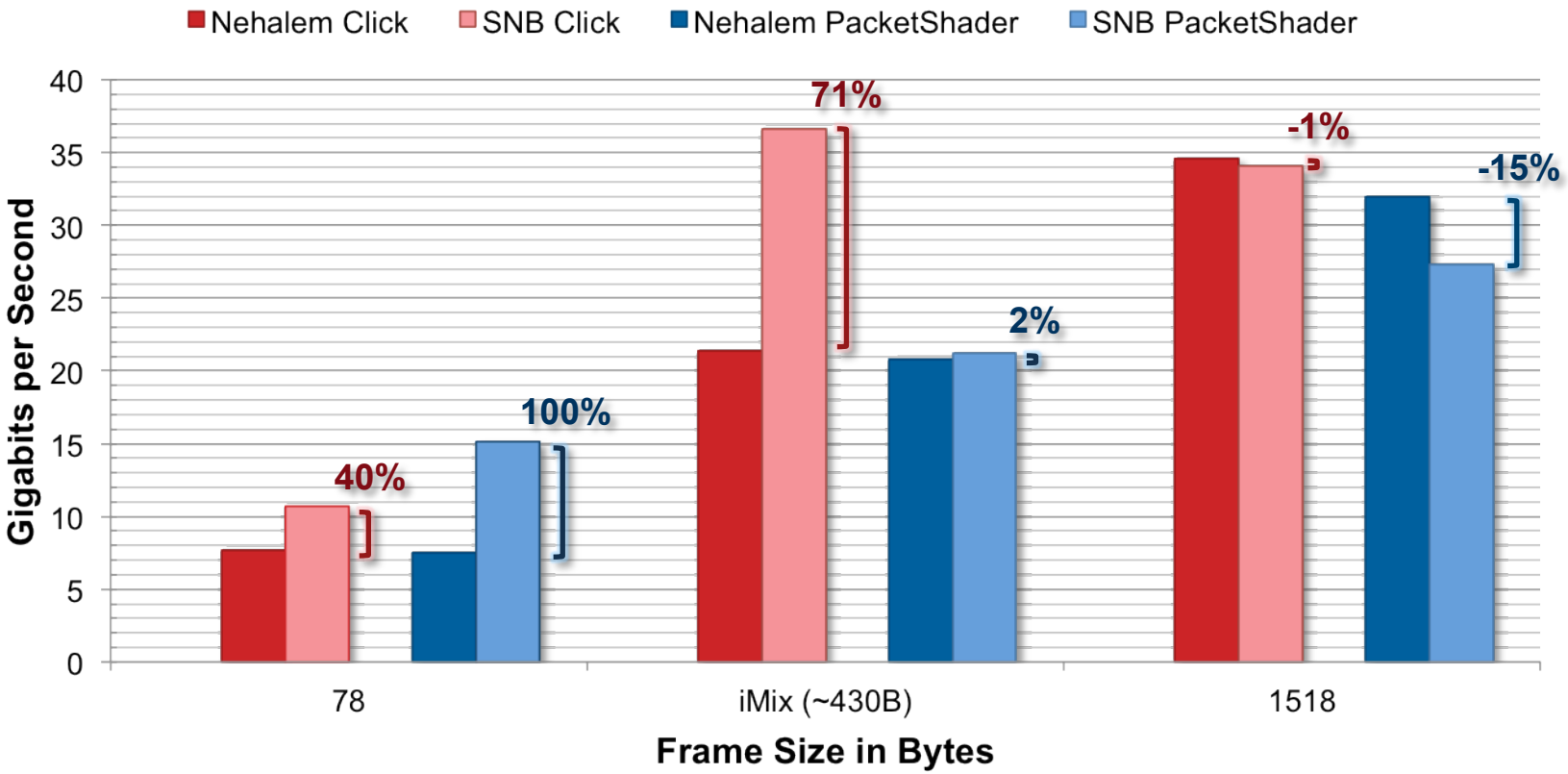



1. Packets are received directly to memory using DMA
2. Software processing is triggered typically via interrupting, polling, or a hybrid scheme
3. Packet processing usually has many stages; requires CPU cycles and memory accesses
4. Once packet is processed, it is copied to destination NIC buffer
5. Destination NIC is notified of a pending transfer
6. NIC uses DMA to transfer packet directly from memory

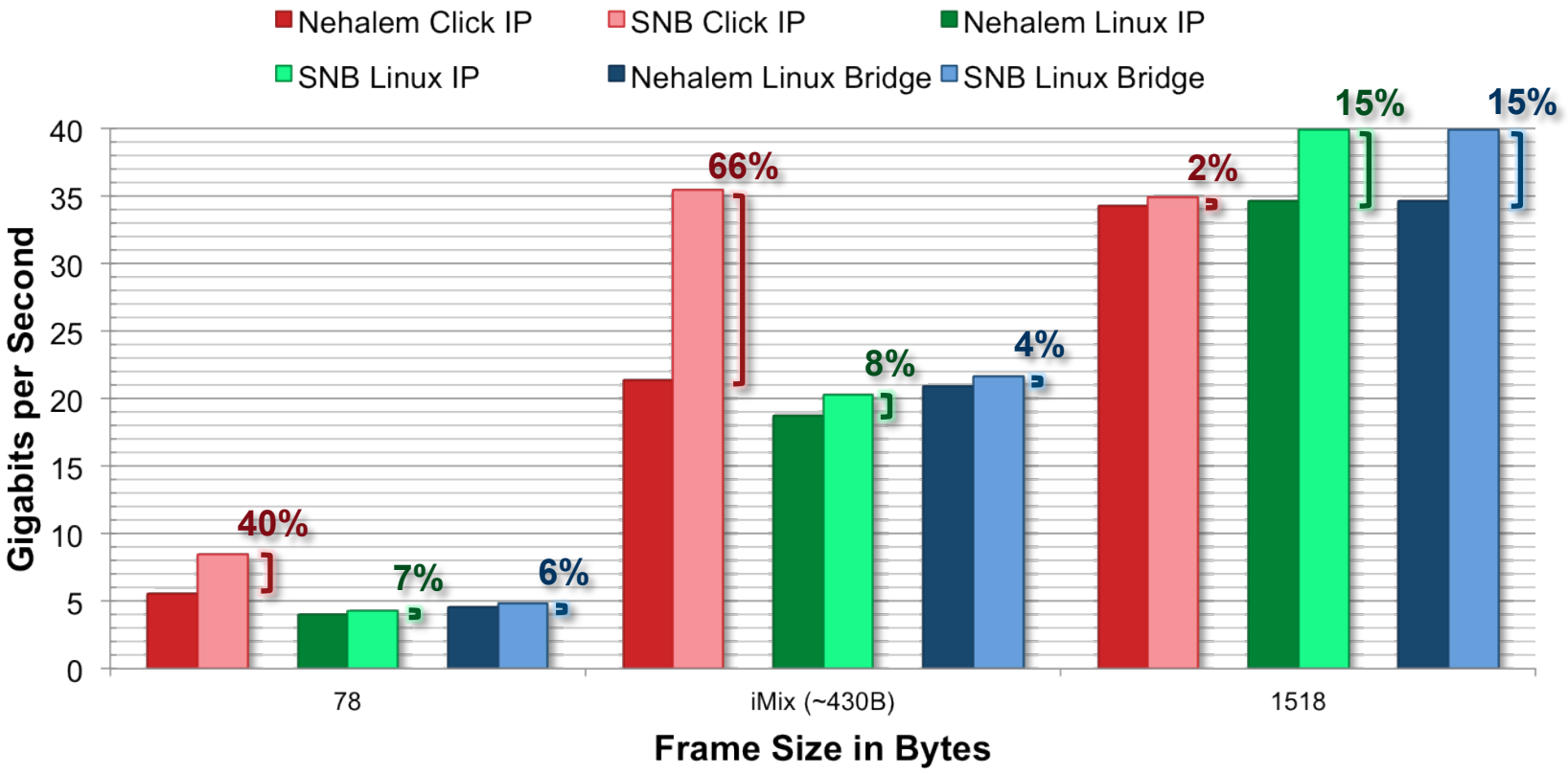
# Test Environment

- Two devices (Nehalem and Sandy Bridge) with a single CPU (4 cores), using 4x 10GBps Ethernet ports
- Test both raw and processed I/O performance
  - Raw I/O tests using two software packet processing engines: Click and PacketShader
  - Processed I/O tests using Linux bridging and IP forwarding and a simple Click IP forwarding
- Measure the maximum lossless throughput using three frame size categories
  - Small (78 bytes) and large (1518 bytes) fixed size frames
  - A frame size distribution (”Internet Mix”) modeling typical Internet TCP traffic

# Raw I/O Throughput



# Processed I/O Throughput



# Conclusions

- Specialized packet processing software can improve small frame performance significantly
  - Small frame processing is the most resource intensive task
  - Currently unknown issues bottleneck large frame throughput
- Linux network stack overhead eats most of the performance benefits of Sandy Bridge
  - Performs well with large frames, where network stack overhead in general is hidden by the interarrival time for large frames
- Our poster contains results on the effect of Linux parameter exploration on packet processing performance