StreamBox: Modern Stream Processing on a Multicore Machine

Hongyu Miao*, Heejin Park*, Myeongjae Jeon^, Gennady Pekhimenko^, Kathryn S. McKinley^, and Felix Xiaozhu Lin*
*Purdue ECE ^Microsoft Research #Google
xzl@purdue.edu

1. Motivation

• Single multicore machine for stream processing:
  • Terabyte DRAM, large numbers of cores, and fast I/O
• Challenges of stream processing on a multicore machine:
  • Handling out-of-order input data
  • Exploiting parallelism to harness tens of CPU cores
  • Exploring memory hierarchy to minimize data move
  • Achieving both high throughput and low latency

2. Key Mechanism: Cascading Containers for Processing Stream Epochs in Parallel

3. Key Results

• Built StreamBox from scratch in 23K SLoC C++
• Designed Cascading Container mechanism for processing out-of-order stream in high concurrency
• Achieved both high throughput and low latency -- 20x lower than popular large-scale streaming engines

4. Ongoing and Future Work

• Optimizing streaming operator performance
• Making StreamBox dataflow NUMA-friendly
• Exploiting heterogeneous memory architecture, e.g. Intel Knights Landing
• Guaranteeing data security, e.g. confidentiality and integrity, during stream processing