IoT devices produce a lot of data that is processed with ML algorithms.

**Challenges with IoT hardware**
- IoT hardware have limited resources such as processing, storage, power etc.
- ML Processing has to be offloaded to a distant server
- Offloading adds latency, consumes network bandwidth

**Solution: Process ML applications in Edge Network**
- Edge servers: more compute & storage than IoT devices
- Edge servers closer to IoT devices: reduces the network latency, enabling real-time processing
- Reduce backbone network bandwidth consumption
- Edge servers can aggregate data from various devices to generate a holistic view

---

**Challenges of using GPU in Edge Servers**
- ML applications exploit GPUs for speeding up computation.
- But: getting data to/from GPUs causes additional latency

**Challenges with GPU**
- GPU are PCIe resident devices and all data is transferred to GPU via DMA using PCIe BUS
- NVIDIA GPU’s require the data to be stored in page-locked “pinned” host memory to initia DMA
- Initiating data transfer from host one packet at a time has high overhead

**Network Function Virtualization Platform**
- NFV is framework to virtualize network function (NFs): load balancers, firewall, IDS etc.
- We are using OpenNetVM (ONVM) NFV platform in a COTS edge server to host NFs and ML applications in containers.
- ONVM uses the DPDK library and shared memory optimized for fast packet processing.
- Packets that arrive in the edge server go into a shared memory location before being accessed by NFs and ML applications.

**Our Solution (NetML)**
- We used a data transfer kernel using NVIDIA CUDA’s universal virtual addressing (UVA) to access and transfer the data from payload of the packet to the GPU.

---

**Results**
- Time Spent on Each Phase of Execution
- Image Processing in Packets
- First Packet Arrives
- First Packet Arrives Data Moving Kernel