Presentation Focus

- Heterogeneous Compute in a Cloud Native Environment

Who Am I?

- I graduated in 1987 from Purdue University with a degree in Electrical Engineering
- I’ve been working in high tech for 35 years
  - Motorola, Five Silicon Valley Startups, F5 Networks, Amazon, Intel (for about one year now)
- The first half of my career as a Software Design Engineer, ASIC Design Engineer, FPGA Design Engineer
- The second half of my career leading design and architecture organizations
- Primary focus is in Networking and Computer Architecture
- At Intel my role is a staff level strategy position where I focus on how Intel HW/SW/Systems enable future Cloud Computing. Focused primarily on FPGAs.
FPGAs in a Multi Cloud Environment

Public Core
- Compute
- Network
- Storage

Public Edge
- Compute
- Network
- Storage

Private Core
- Compute
- Network
- Storage

Private Edge
- Compute
- Network
- Storage

Telemetry
Security
Orchestration

Source
Open
Repository

OpenCL™
System Verilog

oneAPI
Multi-Cloud & Hybrid-Cloud

- Public Cloud scale and Private Cloud specialization
- Cloud federation and orchestration supports distributed solutions on common stacks
- DevOps development flows allow for speed and distributed application integration
- Library Based Repos support open domain specific full-stack solutions, community and reuse
- End-to-end systems that orchestrate compute across Application:Network:Storage layers
Heterogeneous Compute

Application/Infrastructure/Service Stack

- Apps (And App Services)
- Service Mesh
- Orchestration
- Containerization
- Hypervisor
- VMs
- gRPC/HTTP/TCP/IP
- Operating System

- BIOS
- CPU
- GPU
- FPGA
- Memory
- Other (TPU)

- Customer Apps
- CSP Apps
- Istio, HashiCorp
- Kubernetes, CNI
- Docker
- KVM
- VMWare
- Linux
- Linux
- Microservice Execution Units
- DRAM, SRAM, HBM, Optane, SSD, HDD
- NVMe, Ethernet, PCIe, CXL

XPU

- intel.
- AGILEX
- eASIC
- XEON
- Habana
- Ponte Vecchio

DevOps Stack

- Monitor
- Operate
- Deploy
- Release
- Test
- Build
- Code

- Dog, Splunk, Prometheus
- Tableau, Terraform
- Ansible, Puppet, Chef
- Jenkins
- TestRail, Junit, Selenium
- Linux, Artifactory
- Git
- Linux
- Artifactory
- Git

“What Would Software Do?”
Cloud Native Stacks allow for low friction integration and scale
DevOps Development/Test/Release/Operate allows for Speed and Reuse
Make heterogeneous compute accessible to more developers
A Microservice Life Cycle

1. Cloud Native Developed
2. Service Advertisement
3. Service Registry
4. Services Discovery
5. Service Delivery

- Function
- Vertical Performance
- Performance/Watt
- Performance/$
- Latency
- Jitter
- Horizontal Scalability

Service Registry

- Service
- IP/Port
- Service State

Orchestrator

- REST API
- gRPC

μService

- FPGA
- CPU
- GPU
- XPU

Common Cloud Native Stacks
- Service Aware Orchestration
- Applications use the best execution unit for the workload
- With Microservice level granularity
Cloud Layers

- Apps and App Services
- Compute Instances
- VPC Infrastructure
- Storage Network
- External I/O
Cloud Layering allows for scale, services management, security

Locate the workload in the stack layer on the compute platform best suited for execution

Hardware acceleration at Compute, Network, Storage Layers

Cloud Provider and Customer Persona Evolution

Infrastructure as a Service, Platform as a Service, Managed Services

Multi-tenant, composable, infrastructure

Monolithic Applications

Virtual Machines supports Lift and Shift on known framework

Cloud Native Containerization of Workloads. Composable, Disaggregated, Distributed

Orchestration of Application, Network & Storage Services

DevOps Development, Test, Release, Operate

- Speed and reuse
- Make heterogeneous compute accessible to more developers