Sharing, Protection, and Compatibility for Reconfigurable Fabric with *AmorphOS*

AHMED KHAWAJA[1], JOSHUA LANDGRAF[1], ROHITH PRAKASH[1]
MICHAEL WEI[2], ERIC SCHKUFZA[2], CHRISTOPHER J. ROSSBACH[1,2]

[1]UT AUSTIN

[2]VMWARE RESEARCH GROUP

PUBLISHED IN OSDI 2018

Goals

Protected Sharing/Isolation

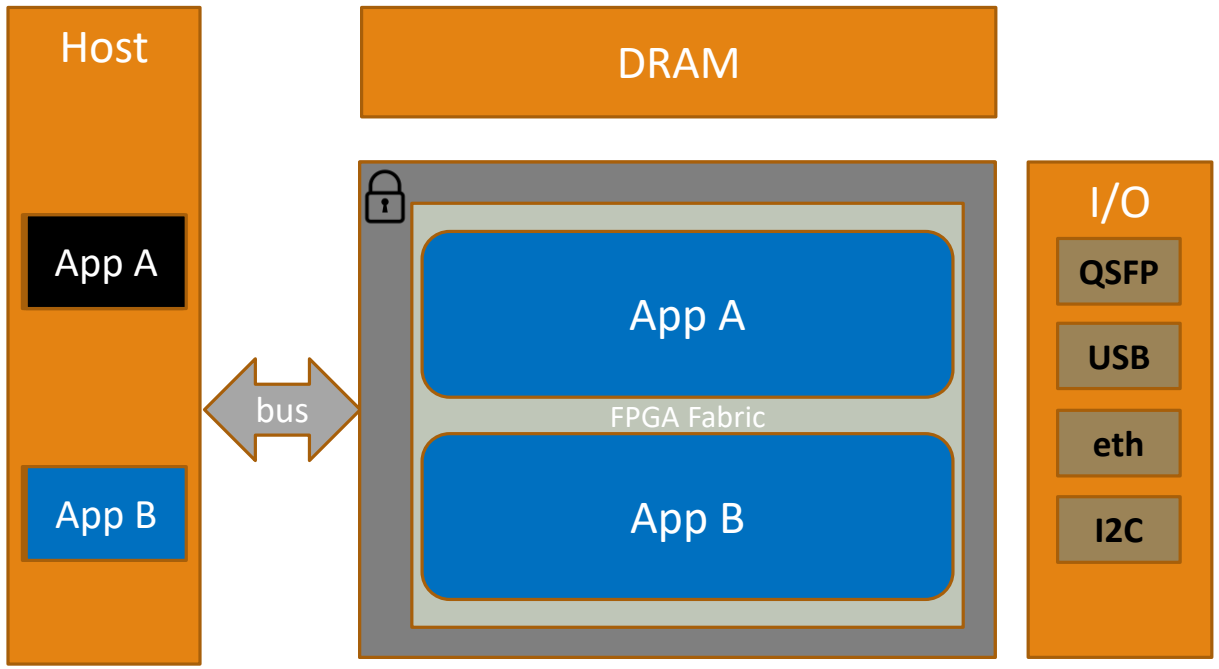
- Mutually distrustful applications

Compatibility / Portability

- HDL programming model
- Accelerators written to AmorphOS interfaces
- ***15+ benchmarks run unchanged on Microsoft Catapult and Amazon F1***

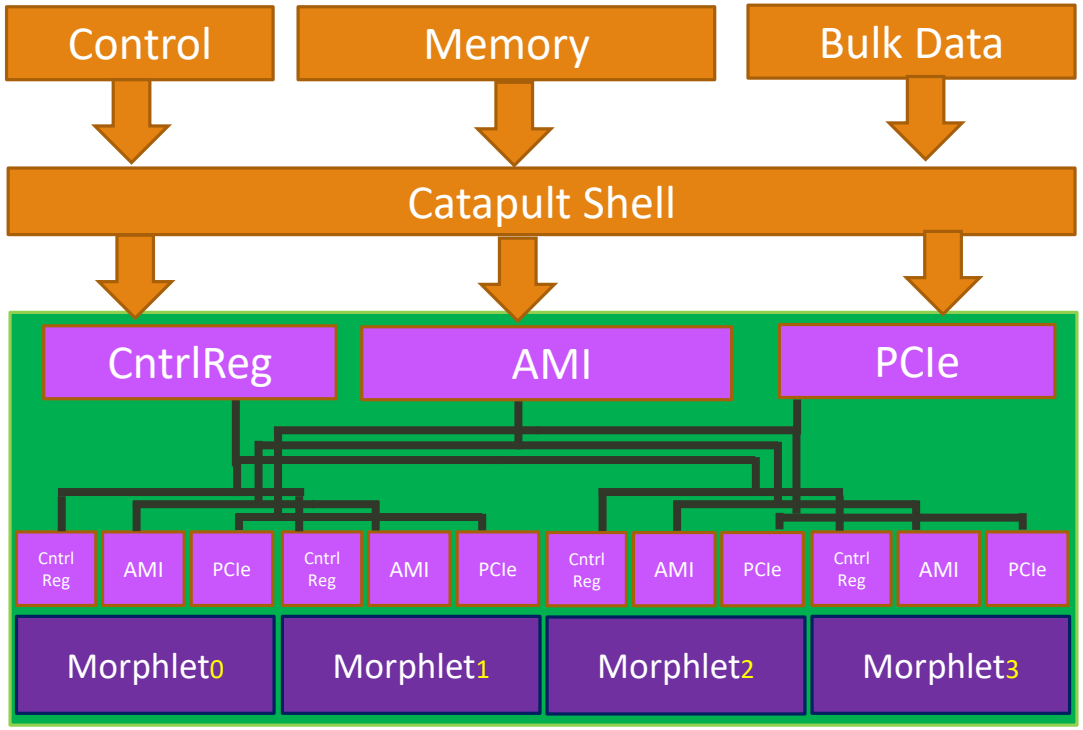
Elastic Scalability

- User logic scales with resource availability
- Multiplex fabric in time *and* space
- Avoid Partial Reconfiguration (PR)
- Avoid fixed slots



AmorphOS Abstractions

- **Zone:** Allocatable Unit of Fabric
  - 1 Global zone
  - N dynamically sized, sub-dividable PR zones
- **Hull**: OS/Protection Layer
  - Memory Protection, I/O Mediation
  - Interfaces form a compatibility layer
- **Morphlet:** Protection Domain
  - Extends Process abstraction
  - Encapsulate user logic on global or PR zone
- **Registry:** bitstream cache
  - Hides latency of place-and-route (PaR)



Open Source Soon

www.amorphos.io

https://github.com/afkhawaja/amorphos

Supported Platforms

- Microsoft Catapult (TACC)
- Amazon F1 FPGA Cloud Platform
- Intel Stratix 10 (in progress)

Workloads

- DNNWeaver
- CHStone
- Crypto Mining
- Memory Synthetics
- TVM (in progress)