Flash Memory Summit

# Reliability, Availability and Serviceability (RAS) for CXL

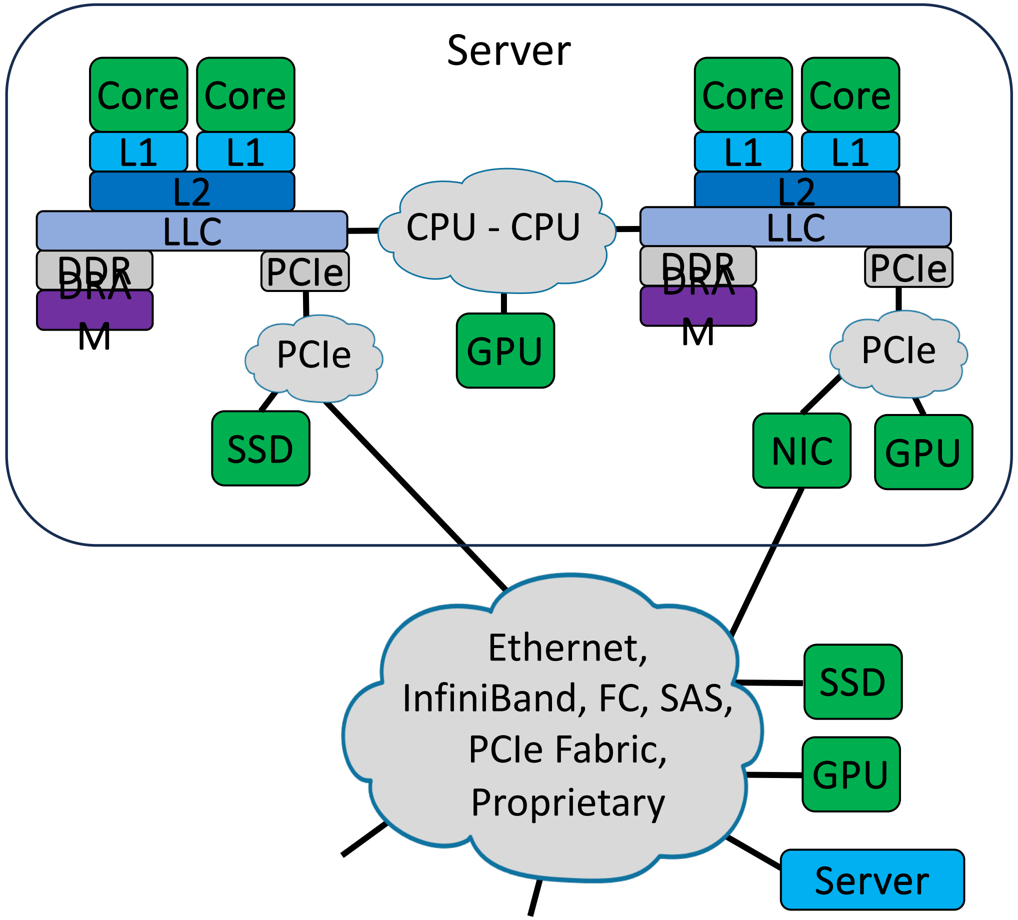
Presenter: Larrie Carr, VP Engineering, CXL Solutions, Rambus

# Overview

- RAS as a Feature of CXL Systems
- CXL Design Challenges
  - Reliability
  - Availability
  - Serviceability
- Final Thoughts

# Before CXL…

- An explosion of connectivity, purpose-built for the application needs
  - Media, data type, IO size, scale, distance, etc.
- Fabrics are created as the connectivity scale increases beyond simple point-to-point
- CPU and attached DRAM treated as one unit – failure of one is a failure for all
- RAS is inherent feature of connections and fabrics…



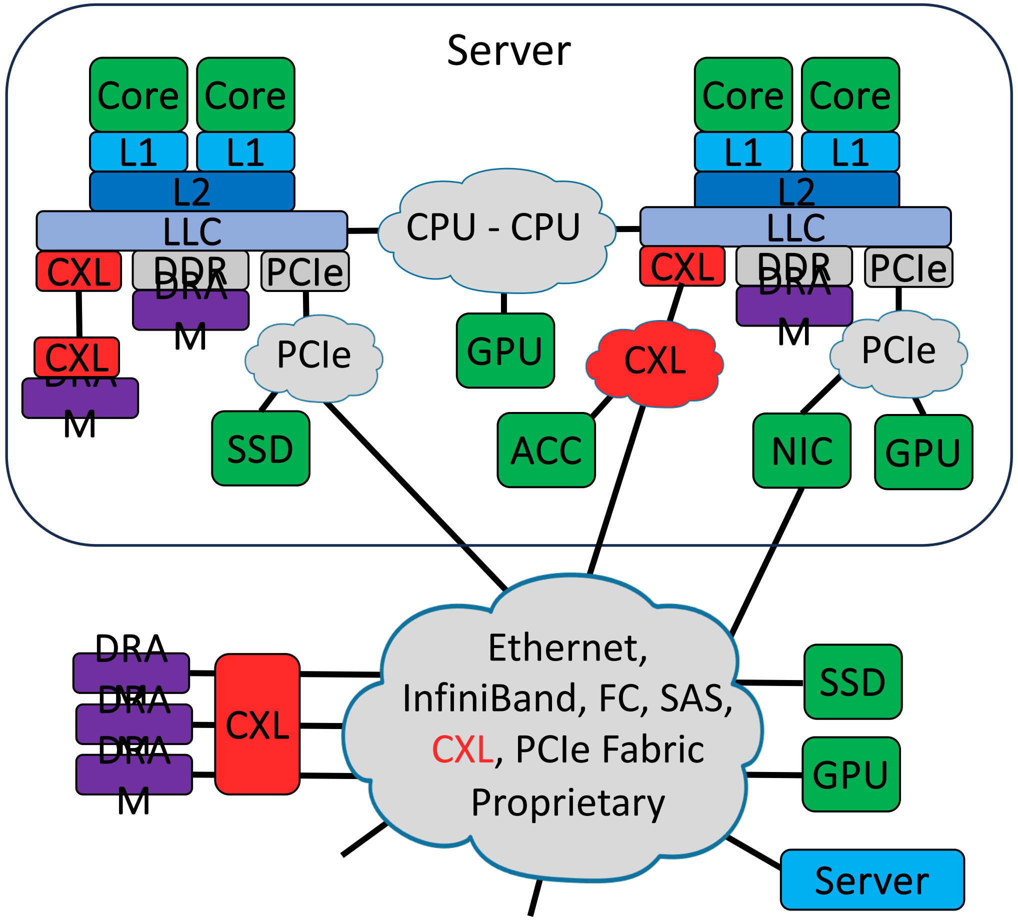
# After CXL…

- CXL is more than an interface on processors

- Physical seperation, switching, new elements all contribute to a new fabric

- Like most standards, CXL leaves a solution's RAS requirements to the architects and implementers

- Need a fabric RAS mindset even when point-to-point…
  - "CXL fabric manager" in the CXL standard



# Reliability

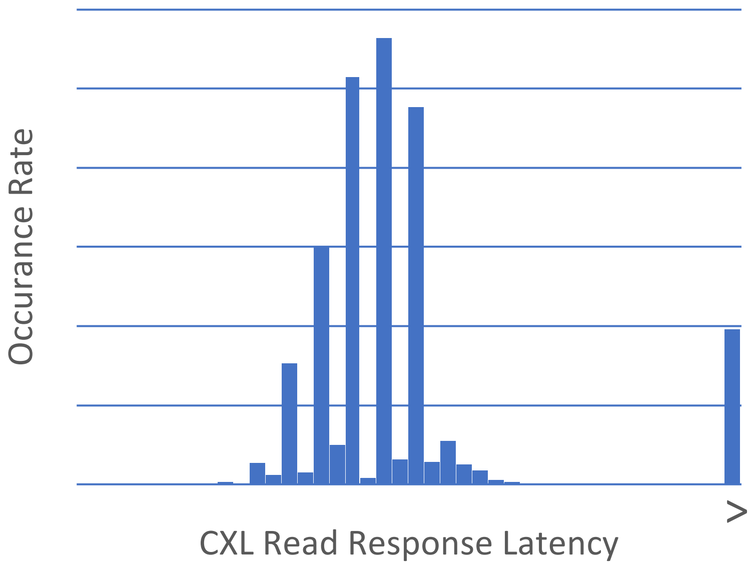
- Most likely the best understood of the three themes, but can easily miss the point
  - Reed-Solomon & “Chip Kill” for DRAM ECC, for example
- At scale, maintaining “data integrity” is the key design requirement - must avoid:
  - Silent data corruption or misdirection
  - Marking corrupted data as good
  - Corruption/loss without consequential/reporting action
- Brings in new design techniques to mitigate soft-error, early-life and aging related failures

# Availability

- FIT rates important as the system scale expands
  - "Blast radius" moves beyond a single server

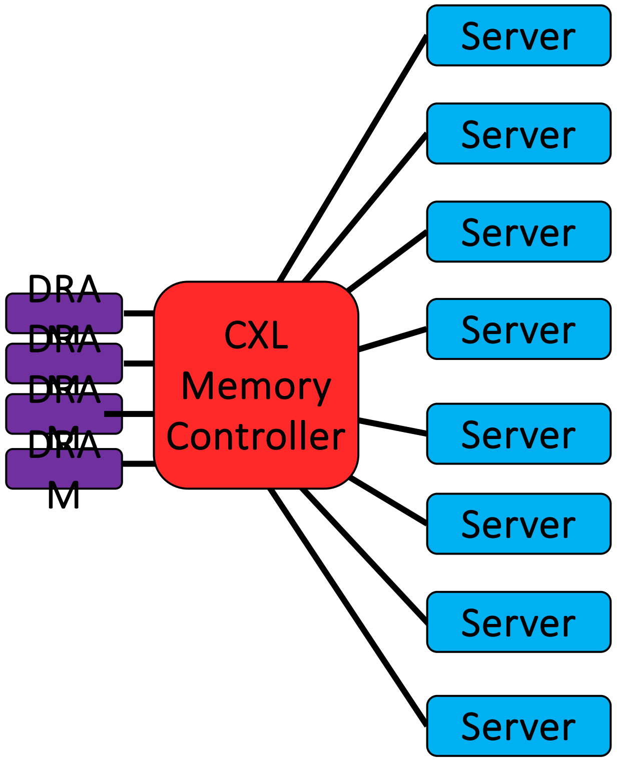
- Advanced Availability with CXL difficult given:
  - CPU's don't tolerant system components disappearing
  - Limited software stacks for consequential actions
  - No redundancy

- Latency falls into this theme given the negative impact on system performance
  - More opportunity for software to respond
    (example new memory tiers with hot page detection)



# Serviceability

- Worst of the three themes for adoption and innovation – debug can be a “customer feature”
- CXL 3.0’s Back Invalidate provides CXL controllers limited data locality control in a logical domain
  - A CXL solution can share memory regions between servers, unlike pooling’s independent memory regions
- A Software Engineer says his shared-memory message queues between servers are getting corrupted – what is the debug strategy?
  - Realize that everything is probably new… software, servers, management, controllers..



# Final Thoughts

- CXL is experiencing an extended Proof-of-Concept phase
  - Everyone is innovating and learning
  - Impacts on architecture, TCO, software, etc being worked through
- Given the hardware relationship between CPU and CXL devices, little opportunity for software to intervene to muddle through problems
- RAS requirements are driven by the ecosystem, not by the CXL standard
  - Good defensive design will ensure the success of CXL