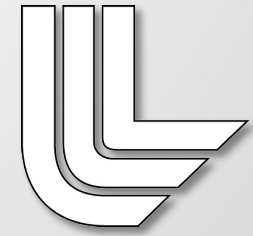


# Resilience Panel

## *Resilience in the X-Stack With a special look at Tools*



**Martin Schulz**

Lawrence Livermore National Laboratory

X-Stack 2 PI-Meeting ♦ May 29<sup>th</sup>, 2014

<http://scalability.llnl.gov/>

**This work was performed under the auspices of the U.S.  
Department of Energy by Lawrence Livermore National  
Laboratory under Contract DE-AC52-07NA27344.**



# How Does Resilience Fit into the X-Stack?

- **What features of other levels of X-Stack should resilience depend on?**
  - No single level can cover resilience alone
  - Recover what you can at each level, but every layer needs to be able to
    - Be able to report failures up
    - Be able to accept failure being handed up
  - Need for integration / coordination across layers -> clean failure semantics
- **How can resilience schemes best exploit application, RT, or PM semantics?**
  - Semantics that defines locality can help determine impact
  - Identify tainted regions, recovery needs, dependencies, ...
- **What is missing from any layer to make resilience schemes succeed?**
  - Information interfaces that allow root cause analysis
  - Clean failure propagation semantics
  - Recovery and clean-up hooks
- **What is the impact on resilience of the wide range of expected scenarios?**
  - Traditional models with strict SPMD semantics are harder
  - New models are a large step forward
  - If applications can deal with changing resources, they can deal with resilience
  - Question: can we get there with legacy applications?

# A Practical and Holistic View on Resilience

- **Vertical integration**

- Failure information must be able to travel up the stack
- Programming models must expose locality and containment
- Clean-up hooks in case of failures would be helpful for local recovery

- **Balance between machine and application view**

- We can't let applications deal with all failures and reduced resources
  - Just saying “there is a failure, deal with it” won't work
- We can't hide failures from applications completely
  - We won't end up with a full “runtime will fix it” approach

- **Close interaction with resource managers needed**

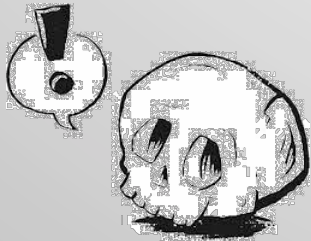
- Recovery of resources if possible
- External constraints will impact recovery actions

# Resilience from the Tools Perspective



- **Resilience impacts the functionality of performance tools**

- Transparent failure detection & recovery acts basically as a noise event
- Performance measurements become unreliable or invalid
- Resilience events can cause ripple effects



- **To show it or Not to show it, that is the question!**

- Tools need to show “clean” performance measurements
- Tools need to show that something went wrong

- **Need for a new set of tools specifically for resilience**

- Analyze the impact and source of failures
- Understand the impact of recovery

- **Even if you think you have everything covered, don't forgot to pay the PIPER!**

