The State of Simulation
Where We Are
What is Wrong
Where We Want to Be

Arun Rodrigues

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company,
for the United States Department of Energy's National Nuclear Security Administration
under contract DE-AC04-94AL85000.

Where We Are

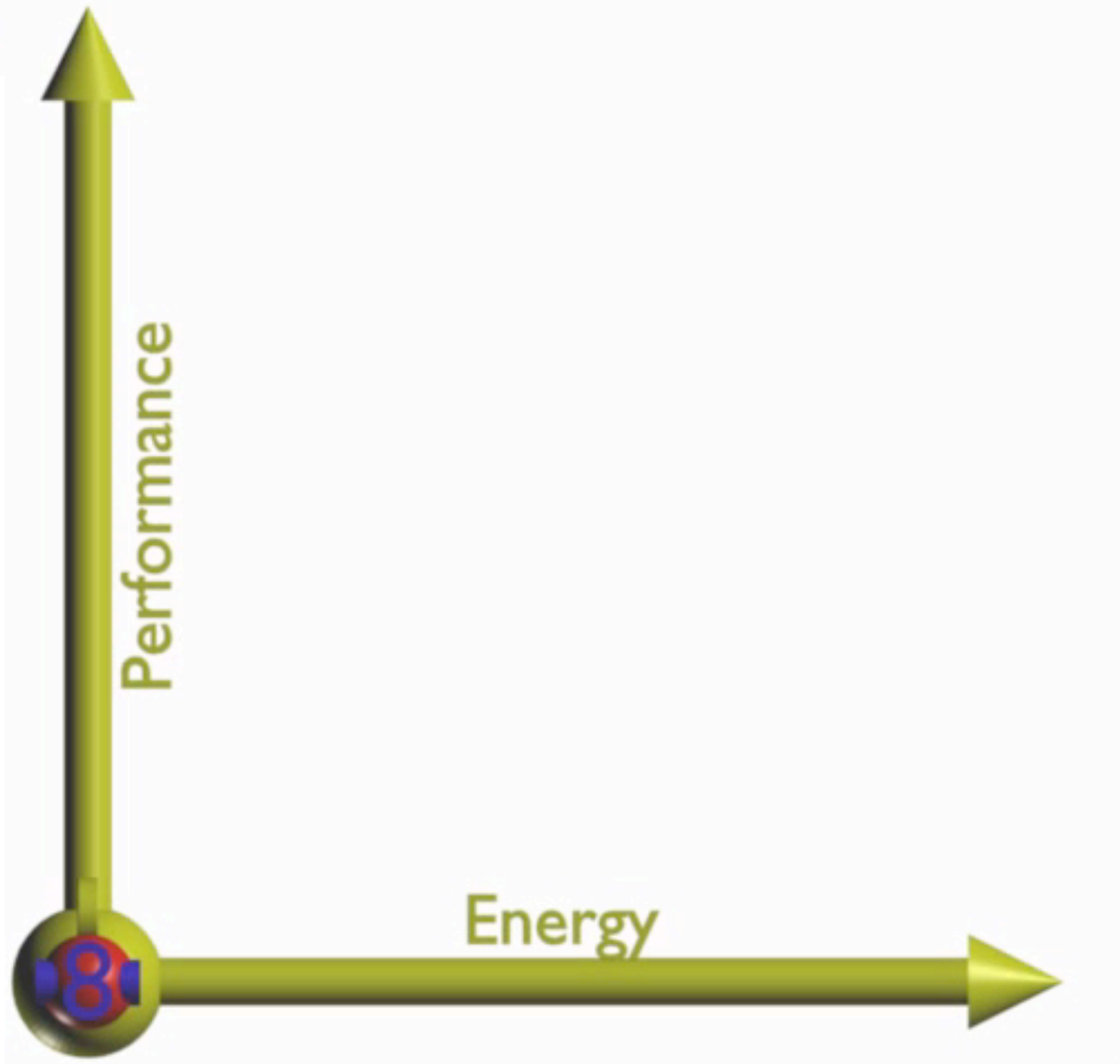
Major Architectural Simulation Challenges Are Increasing

- **Multiple Objectives**
  - Performance used to be only criteria
  - Now, Energy, cost, power, reliability, etc...
- **Scale & Complexity**
  - Many system characteristics require detail to measure
  - Detailed simulation takes too long
  - Application complexity increasing
- **Accuracy**
  - Systems more complex
  - Vendors don't reveal necessary details

Sandia National Laboratories

Major Architectural Simulation Challenges Are Increasing

- **Multiple Objectives**
  - Performance used to be only criteria
  - Now, Energy, cost, power, reliability, etc...
- **Scale & Complexity**
  - Many system characteristics require detail to measure
  - Detailed simulation takes too long
  - Application complexity increasing
- **Accuracy**
  - Systems more complex
  - Vendors don't reveal necessary details



Sandia National Laboratories

View of the Simulation Problem

Scale.....

Many Cores + Memory **X** Many Many Nodes **X** Many Many Many Threads

Multiple Audiences.....

Network Processor System **X** Application writers purchasers designers **X** system procurement algorithm co-design architecture research language research **X** present systems future systems

Complexity.....

Multi-Physics Apps Informatics Apps **X** Communication Libraries Run-Times OS Effects **X** Existing Languages New Languages

Constraints.....

Performance Cost | Power Reliability | Cooling Usability | Risk Size

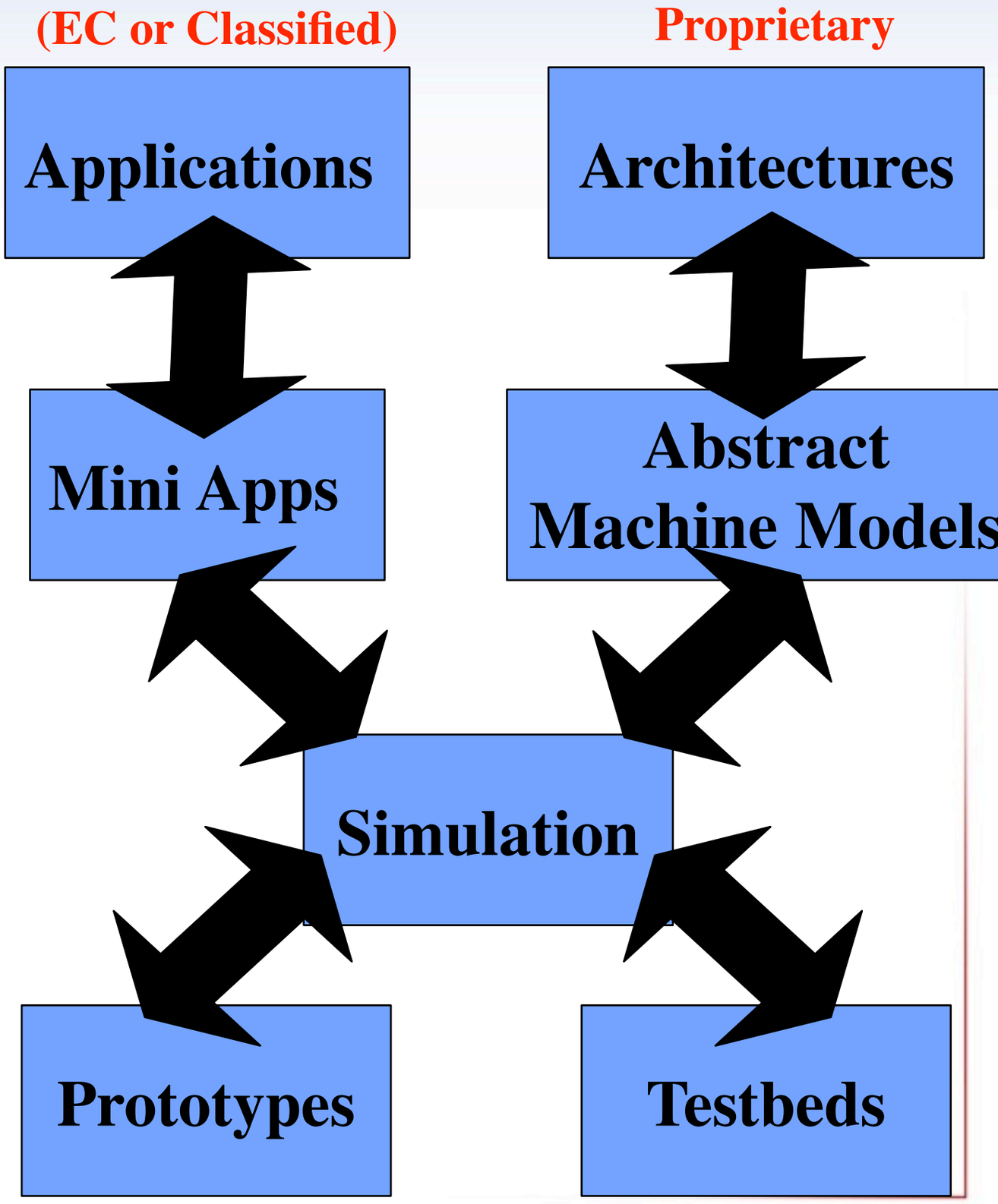
NNSA National Nuclear Security Administration

Sandia National Laboratories

Simulation & CoDesign

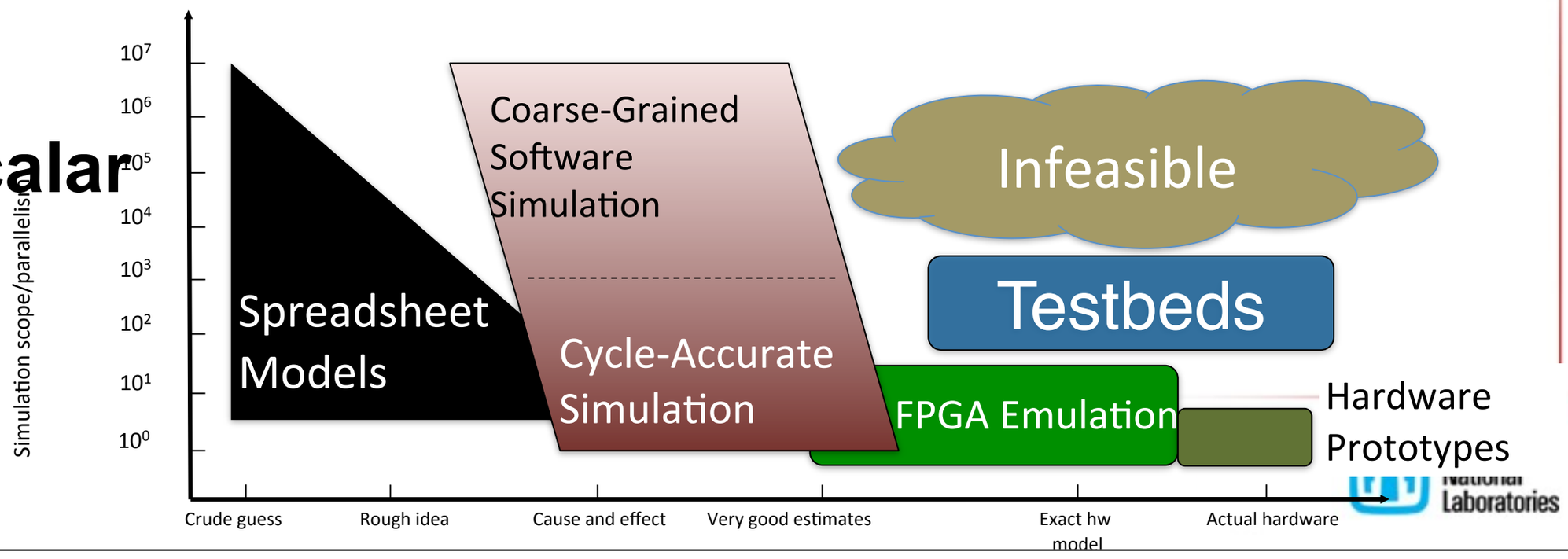
- **Multi-Level Design Strategy**
  - **Allows exploration of multiple types of abstract machine models**
  - **Multiple Levels of Scale**
  - **"Clearinghouse" of ideas**

	Back of the Envelope
Analysis	Analytical Models
High Level	State Machines
Behavioral	Cycle-Approximate,
Hardware Prototype	Prototypes



A (Very) Incomplete List...

- Sniper
- Graphite
- CMPSim
- SIMICS
- SESC
- ASPEN
- OMNet++
- Phoenix
- RAMP
- SimpleScalar
- BigSim
- gem5
- FAST
- SST
- Manifold
- Zesto
- DRAMSim
- MacSim
- TwinCAM
- CAPA
- MARSx86
- HotSpot
- Orion
- IntSim
- McPAT
- PIN
- (at this point, I got tired)



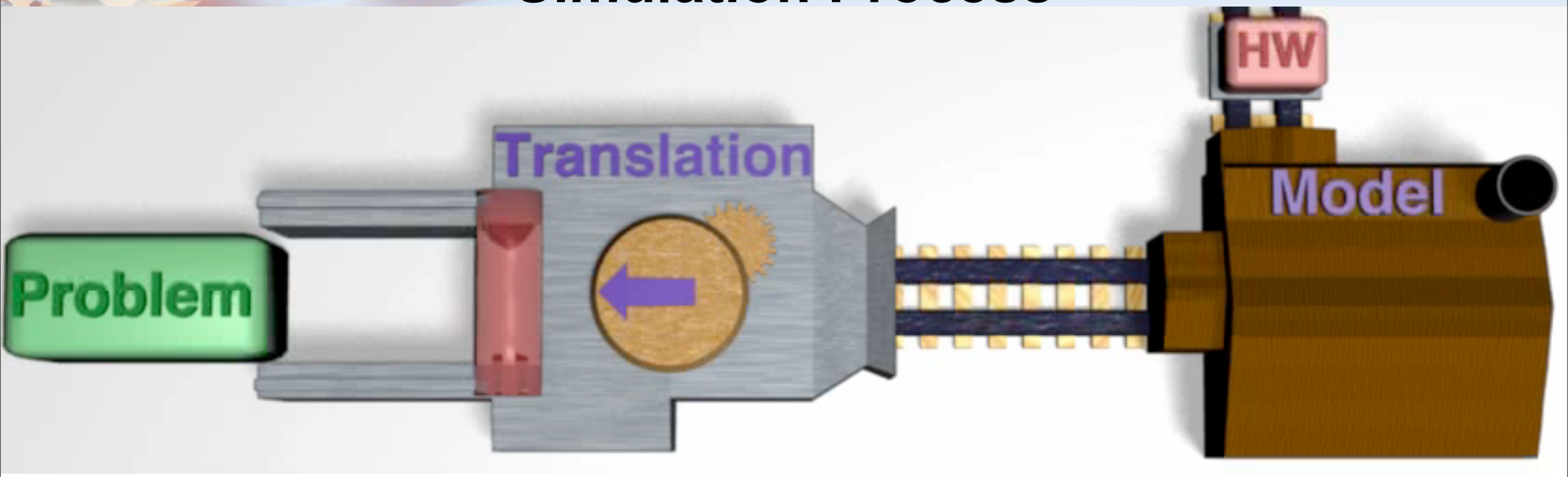
Simulation Process

- **Problem & Software**
  - **Execution vs. Trace vs. Stochastic vs. State Machine**
- **Model**
  - **Emulation vs. Simulation vs. FPGA**
  - **Cycle-accurate/Cycle-Approximate vs. CPI=X**
- **Hardware Representation**
  - **Exact & specific to very general**
- **Hybrid (any or all of the above)**

NNSA National Nuclear Security Administration

Sandia National Laboratories

Simulation Process



- **Problem & Software**
  - **Execution vs. Trace vs. Stochastic vs. State Machine**
- **Model**
  - **Emulation vs. Simulation vs. FPGA**
  - **Cycle-accurate/Cycle-Approximate vs. CPI=X**
- **Hardware Representation**
  - **Exact & specific to very general**
- **Hybrid (any or all of the above)**

NNSA National Nuclear Security Administration

Sandia National Laboratories

What is Wrong

Best: Diversity

Best & Worst Aspects of Today's Infrastructure

Best: Diversity

Worst: Diversity

Little Interoperability / Reuse

Poor Maintainability

Poor Documentation

"Black Box" Effect

Little Trust

Software Engineering

•Mike Kistler (IBM)

- Core Services
  - Must provide the “right” abstractions
  - Must be high-performing
  - Must enable parallel execution
- Simulator construction is primarily a **software engineering** activity
  - Reuse is essential
  - The simulation infrastructure must be **modular**

•Ali Saidi & Steve Reihardt (ARM/AMD)

- Full-time maintenance and support staff
  - These are large projects: gem5 is ~200,000 lines of code
  - Need to handle the unglamorous side of software and tool development: regression testing, validation, porting, etc.

•David Wood

- Use all methods as appropriate
  - Computer evaluation methods are not one size fits all
- More of the community’s activity should benefit the community
  - Expand the use of open source
- Treat computer evaluation as a science

NNSA National Nuclear Security Administration

Sandia National Laboratories

Challenge of Validation

- We don't have enough details for "fully" accurate simulation
  - –Vendors IP consideration
  - –Complexity vs. Flexibility
- Standards for "how accurate is enough" vary
  - –Models validated at different levels, with different methodologies
- Assumptions poorly communicated
- Lot of Validation in Isolation
- Fundamental question

Is Simulation Useful?

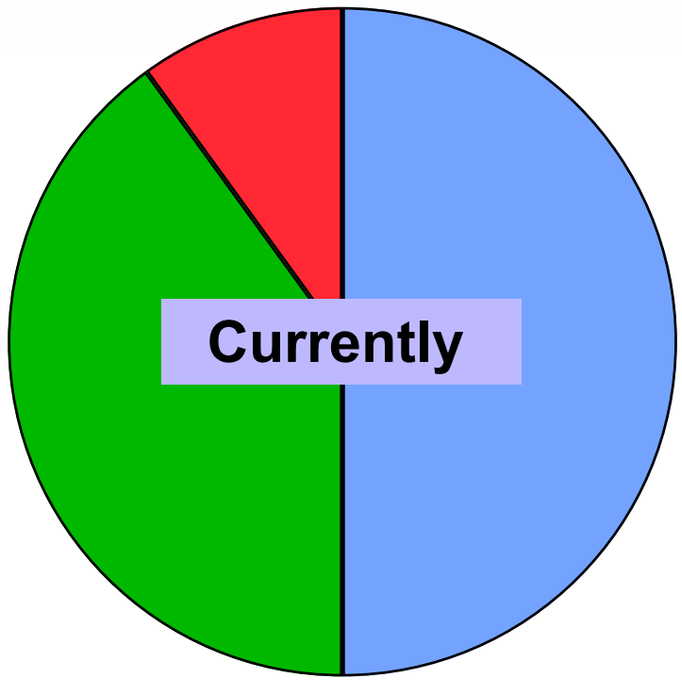
Component	Method	Error
DRAMSim	RTL Level validation against Micron	Cycle
Generic Proc	Simplescalar SPEC92 Validation	~5%
NMSU	Comparison vs. existing processors on SPEC	<7%
RS Network	Latency/BW against SeaStar 1.2, 2.1	<5%
MacSim	Comparison vs. Existing GPUs	Ongoing <10% expected
Zesto	Comparison vs several processors, benchmarks	4-5%
McPAT	Comparisons against existing processors	10-23%
GeM5	Comparisons against existing processors	Ongoing 5-20%

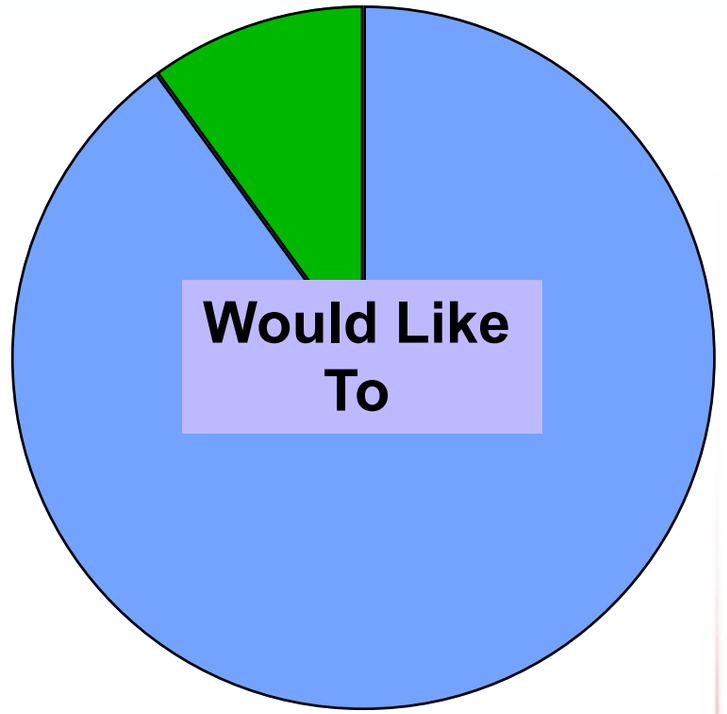
National Laboratories

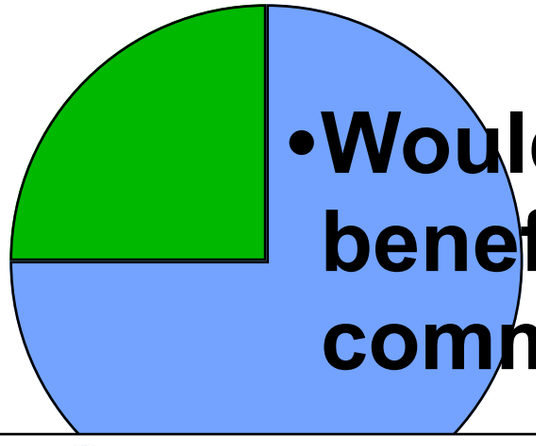
Where We Want To Be

HPCAS Survey

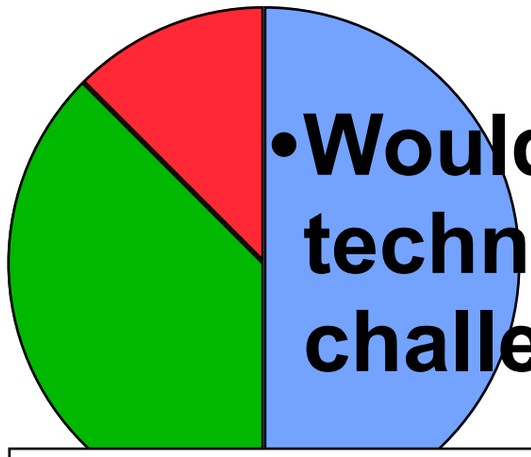
- How much do you currently use simulators? how much would you like to?



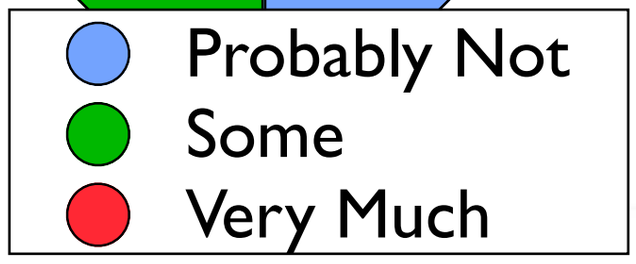




- Would your simulator benefit from a common Framework?



- Would there be major technical integration challenges?



NNSA National Nuclear Security Administration

Sandia National Laboratories

Ideal World

- Common Simulation Environment with dozens of interoperable component models
- Include models for power, energy, temperature, reliability, and cost
- Open non-restrictive license
- PARALLEL & Fast
- Long Term support
  - Documentation
  - Reuse
- Accepted by community
  - Validated to known standard with uniform methodology
  - Easy to replicate results
- Multi-level: Analytical to Behavioral to Cycle-Level to Hardware in the Loop

NNSA National Nuclear Security Administration

Sandia National Laboratories

SST Simulation Project Overview

Goals

- Become the standard architectural simulation framework for HPC
- Be able to evaluate future systems on DOE workloads
- Use supercomputers to design supercomputers

Status

- Includes parallel simulation core, configuration, power models, basic network and processor models, and interface to detailed memory model
- http://code.google.com/p/sst-simulator/

Technical Approach

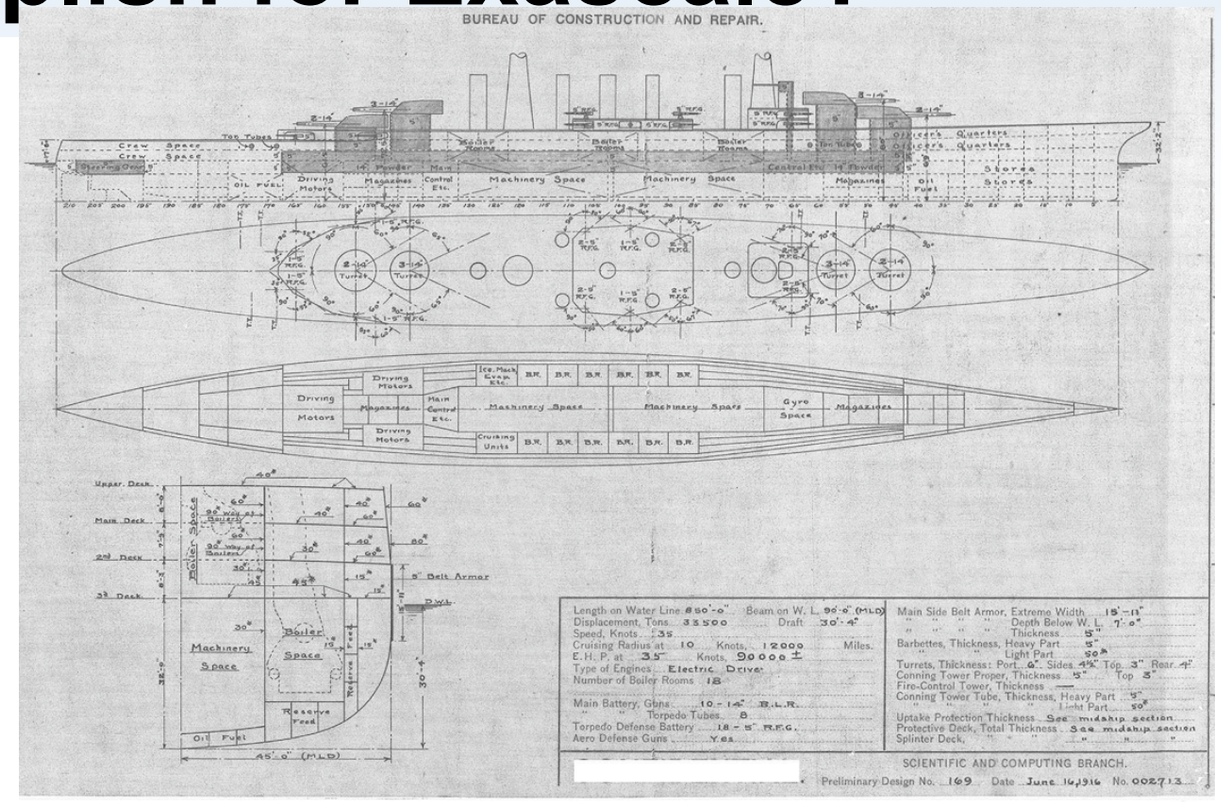
- Parallel
  - Parallel Discrete Event core with conservative optimization over MPI
- Holistic
  - Integrated Tech. Models for power
  - McPAT, Sim-Panalyzer
- Multiscale
  - Detailed and simple models for processor, network, and memory
- Open
  - Open Core, non viral, modular

Consortium

- "Best of Breed" simulation suite
- Combine Lab, academic, & industry

What Can We Accomplish for Exascale?

- Component & System validation is difficult, too many unknowns for high accuracy
- Many ‘consumers’
- Is simulation still useful?
- Historical Analog
  - US Navy would develop “Spring Style” design sketches of warships for use in war games (simulations) and as specification to “vendor” shipyards
  - Combination of abstract design & simulation allow users & vendors to perform co-design
    - “Software Team”: Naval War College developing tactics
    - “Hardware Vendor”: Shipyards building the ship



US Navy “Spring Style”

Role of Simulation for Exascale

- Required for effective co-design, but need to address more audiences
- It is not our role to simulate exactly what an Exascale System will look like today
- We should be developing "Spring Styles"
  - Abstract models which inform the hardware designers of our requirements
  - Models that the application teams can use to understand how future architectures will impact their applications

- We need to harness diversity, not fight it
- We need long-term support
- We need consensus
- How do we get there?

NNSA National Nuclear Security Administration

Sandia National Laboratories

How do we get there?

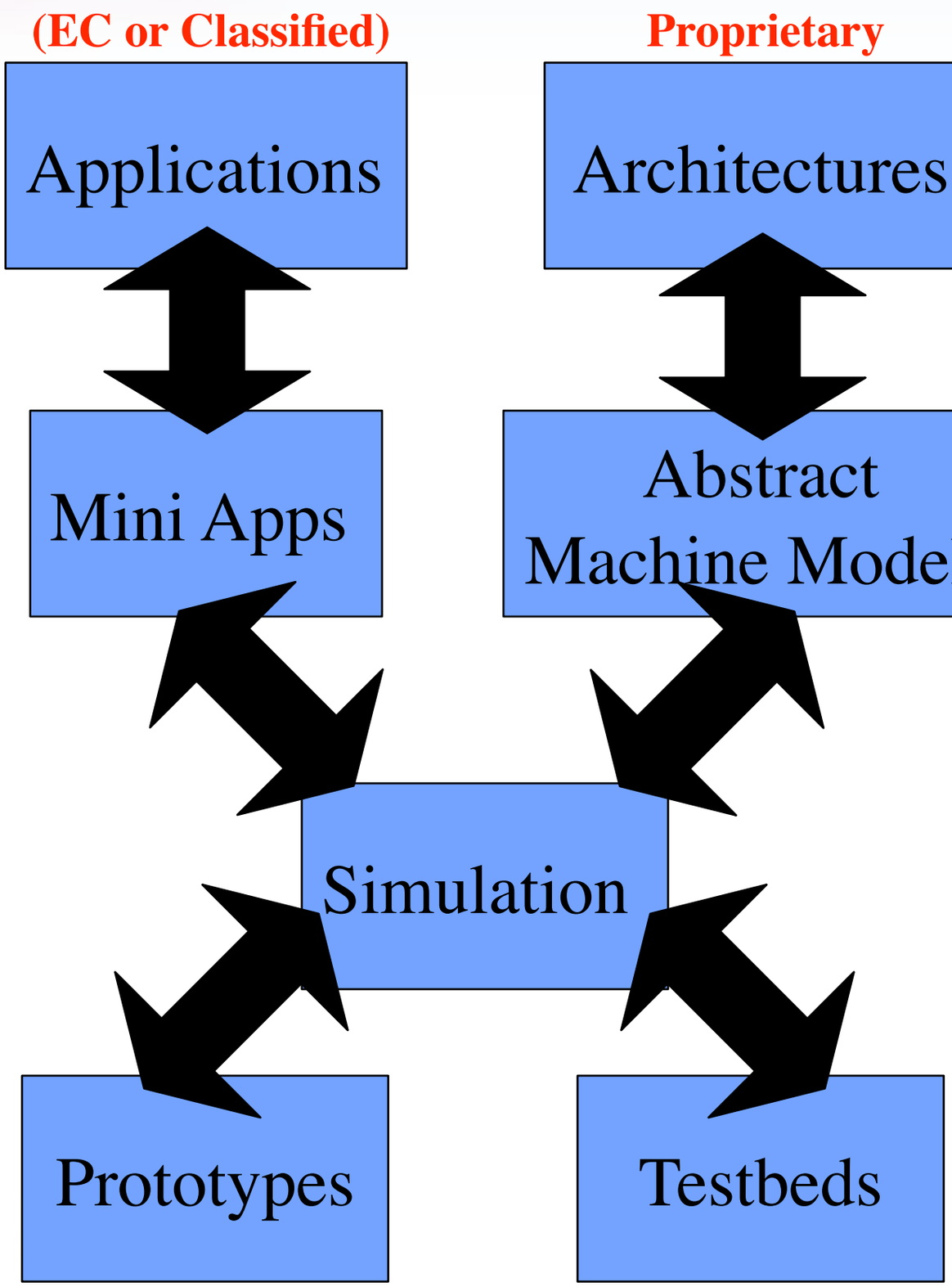
HPCAS Possible Plan of Action: Standard Simulation Interface

- Workshops to define interface
  - Limited mandate
    - Define minimal subset of interfaces to promote adoption
    - Phased approach (core + optional chapters)
    - Three phases
      - Define interface sets: priority/required-or-not, then define interfaces
      - Define Core interfaces
      - Define Component Interfaces
- User Group to guide towards industrial strength
- Encourage multiple existing projects to adopt interface
- Calls for... / Follow-on projects for...
  - Implementations of interface
  - Porting components to plug into interface

- Need to start with clear, limited, attainable goals
- Study examples we already have
- Long-Term Multi-Agency Funding
- Analogy to MPI

Bonus Slides

Simulation is the Nexus of CoDesign



- CoDesign needs a meeting point between applications and architectures
- Full Apps & Archs. are too complex to easily reason about, so we create proxies
- Simulation provides a way...
  - ...to combine and test AMMs and MiniApps
  - ...for application writers to test ideas on machines that don't exist
  - ...for architects to understand evolving application proxies

Sandia National Laboratories

Who Can Save Us?

- Why Not Just Industry?
  - Industry focused on specific products, often no system view
  - Labs are 'neutral' - able to work with everyone, don't compete with anyone, able to keep a secret
  - We can work with Industry and academia to provide a system & application view they might not have
- Why Not Just Academia?
  - Labs can provide long-term support, software engineering (i.e. we can work on something even if it doesn't become a paper)
- Labs bring
  - Neutrality
  - Focus: HPC Long-Term Development System-Level
  - Application Knowledge
  - Look for big changes - out the box ideas, long term

Component Validation

- Strategy: component validation in parallel with system-level validation
- Current components validated at different levels, with different methodologies
- Validation in isolation

- What is needed
  - Uniform validation methodology (apps)
  - System (multi-component) level validation
  - Learn from multi-level physics apps

Component	Method	Error
DRAMSim	RTL Level validation against Micron	Cycle
Generic Proc	Simplescalar SPEC92 Validation	~5%
NMSU	Comparison vs. existing processors on SPEC	<7%
RS Network	Latency/BW against SeaStar 1.2, 2.1	<5%
MacSim	Comparison vs. Existing GPUs	Ongoing <10% expected
Zesto	Comparison vs several processors, benchmarks	4-5%
McPAT	Comparisons against existing processors	10-23%
GeM5	Comparisons against existing processors	Ongoing 5-20%