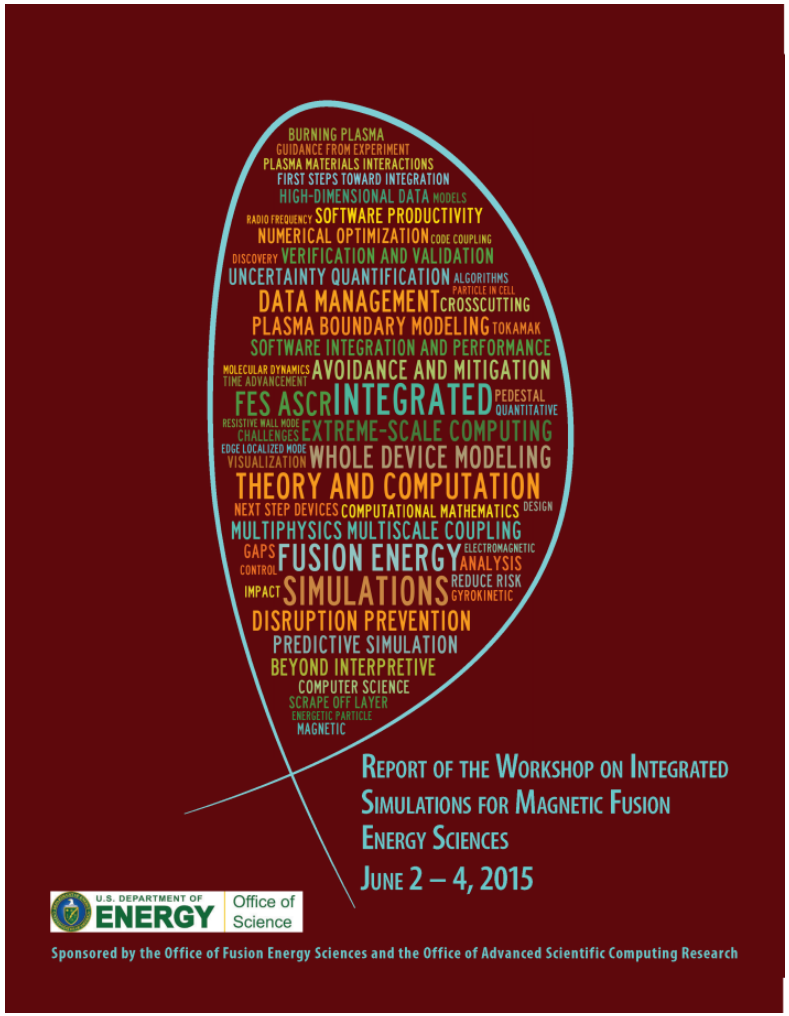
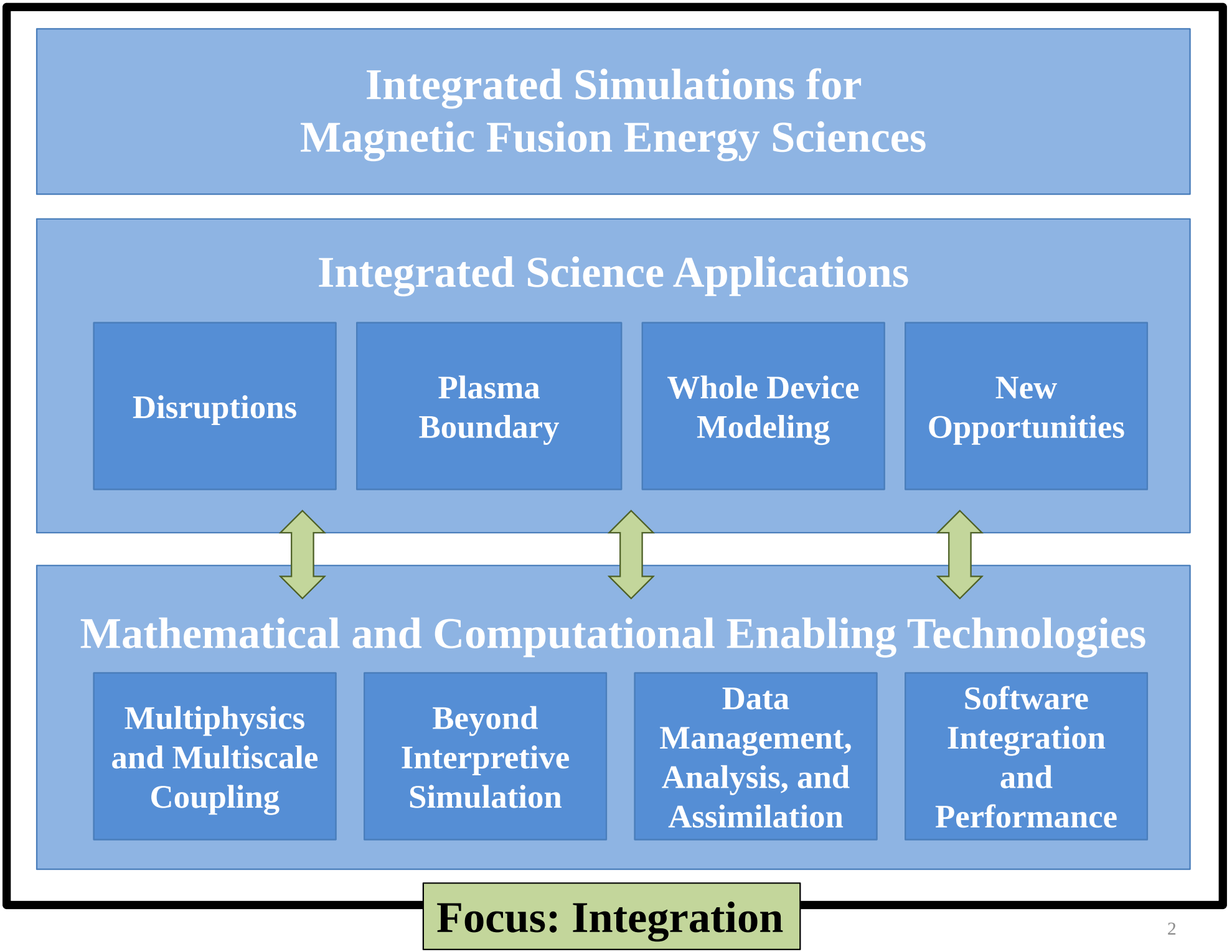
Software Needs for Integrated Simulations for Magnetic Fusion Energy Sciences

David E. Bernholdt
Oak Ridge National Laboratory
and
Robert F. Lucas
ISI/USC
Software Integration and Performance Panel Chair and Co-Chair

Paul Bonoli
Massachusetts Institute of Technology
and
Lois Curfman McInnes
Argonne National Laboratory
Workshop Chair and Co-Chair (and slide makers)



Workshop report: http://science.energy.gov/~/media/fes/pdf/workshop-reports/2016/ISFusionWorkshopReport_11-12-2015.pdf


Integrated Simulations for
Magnetic Fusion Energy Sciences
Integrated Science Applications
Disruptions
Plasma Boundary
Whole Device Modeling
New Opportunities
Mathematical and Computational Enabling Technologies
Multiphysics and Multiscale Coupling
Beyond Interpretive Simulation
Data Management, Analysis, and Assimilation
Software Integration and Performance
Focus: Integration

Software Integration and Performance

Panel Chair: David Bernholdt (Oak Ridge National Laboratory)

Panel Co-Chair: Robert Lucas (University of Southern California, ISI)

Panel Members:

John Cary[2] (Tech-X Corporation)

Milo Dorr (Lawrence Livermore National Laboratory)

Alice Koniges (Lawrence Berkeley National Laboratory)

Orso Meneghini[2] (General Atomics)

Boyana Norris (University of Oregon)

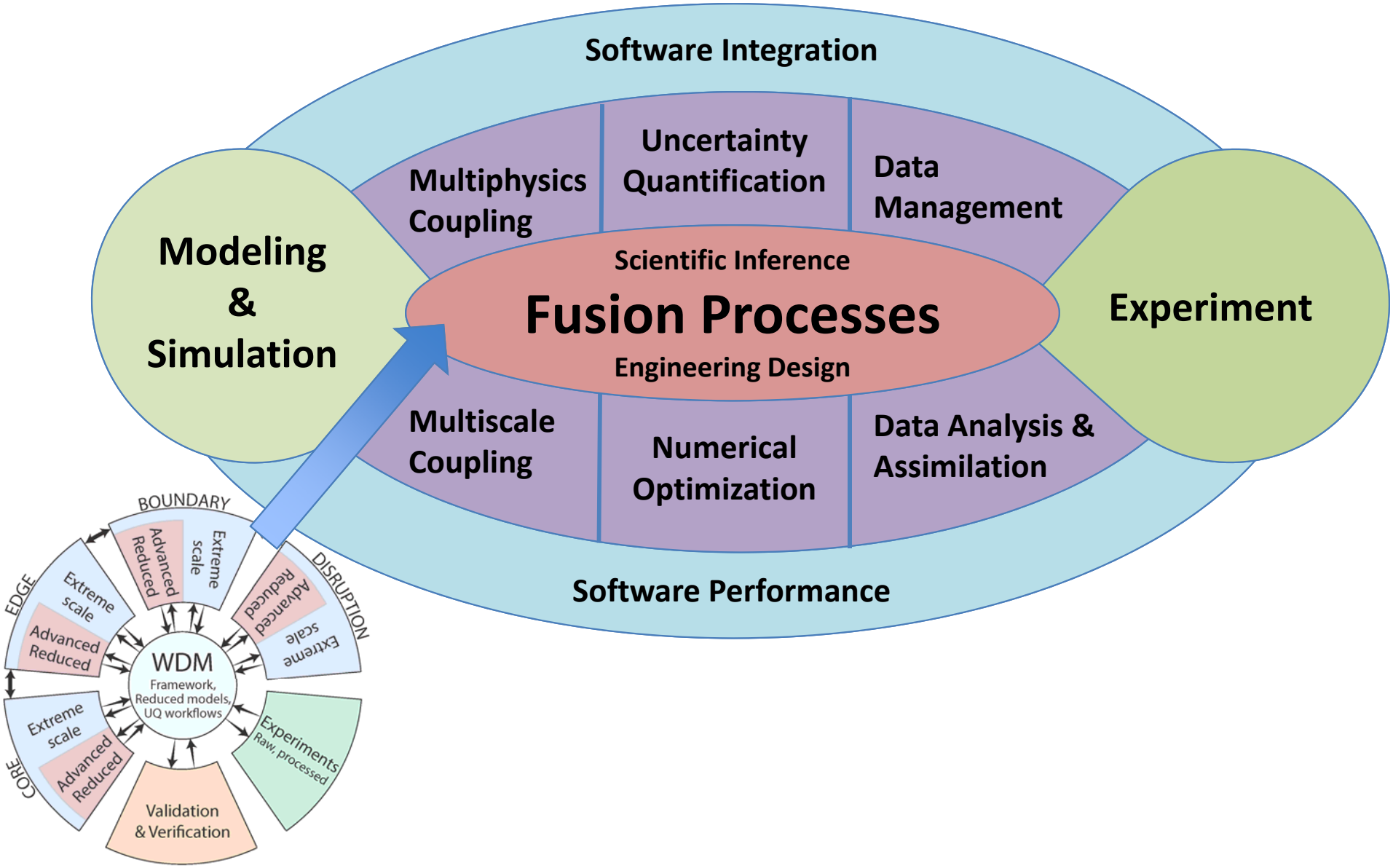
Francesca Poli[2] (Princeton Plasma Physics Laboratory)

Brian Van Straalen (Lawrence Berkeley National Laboratory)

Patrick Worley (Oak Ridge National Laboratory)

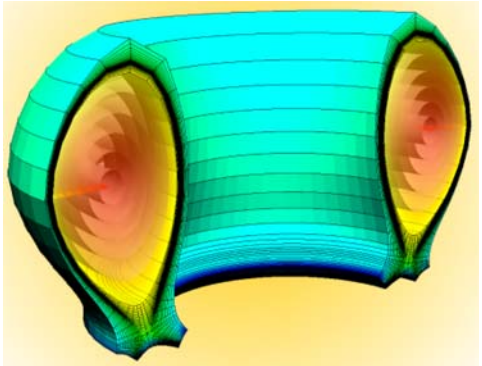
[2] Crosscutting expert from FES

Software integration and performance: Permeate all aspects of work



Software integration & performance

- **Code design, maturity, and integration**
  - Different codes for different purposes; need to (re)structure codes to make them more readily composable
  - Common for single-physics codes to be in both standalone and integrated contexts
  - Useful design pattern: ‘Component’ approach, with interchangeability of conceptually similar codes
- **Performance and portability**
  - Must plan for emerging extreme-scale architectures: performance-aware software
  - Understanding performance in coupled contexts
  - Need to expose performance models and performance variation
- **Culture, community, and governance issues**
  - Sharing code; institutional investments in own codes; tension between ‘research’ and ‘production’ software
- **Software productivity and software engineering for integrated fusion applications**
  - Methodologies for revision control, build systems, bug tracking, documentation, refactoring, interoperability, performance portability, etc.
  - Testing (unit, integration, system level, performance, etc.)
- **Recent progress**
  - Fusion proto-FSPs (FACETS, SWIM, CPES); SciDAC projects: AToM, EPSI
  - SUPER SciDAC Institute, IDEAS software productivity project
- **Related work**
  - EU Integrated Tokamak Modeling, ITER’s Integrated Modeling and Analysis Suite (IMAS): compatibility useful and desirable



Coupled core-edge simulation using FACETS

Software Integration and Performance:
Priority Research Directions

- **[PRD-Software-1]** Implement software engineering best practices, consistently, throughout the fusion integrated simulation community.
- **[PRD-Software-2]** Bring together fusion researchers, applied mathematicians, and performance experts to focus on the performance and portability of fusion codes on current and future hardware platforms.
- **[PRD-Software-3]** Develop community standards and conventions for interoperability.
- **[PRD-Software-4]** Develop best-practice guidelines and recommendations to address the particular software engineering challenges of integrated simulation.
- **[PRD-Software-5]** Perform research on the computer science of code composition.
- **[PRD-Software-6]** Determine a strategy to ensure the sustainability of key fusion integrated simulation infrastructure for long enough to establish a sustainable community of developers and users around it.