

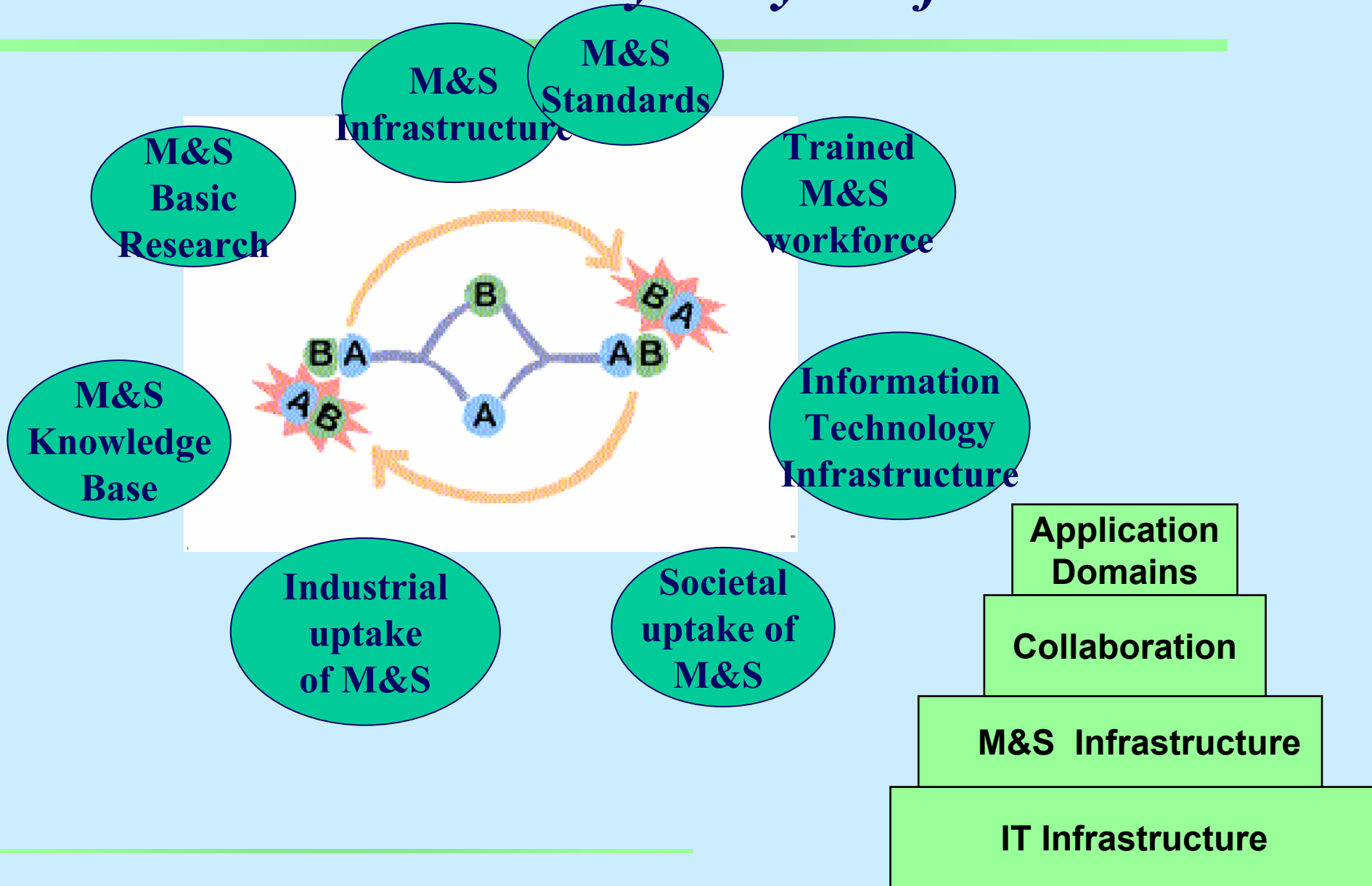
*Arizona Center for
Integrative Modeling and Simulation/
High Performance Distributed Computing Lab*

Bernard P. Zeigler
Salim Hariri
Hessam S. Sarjoughian

**The University of Arizona
Arizona State University**

web: www.acims.arizona.edu

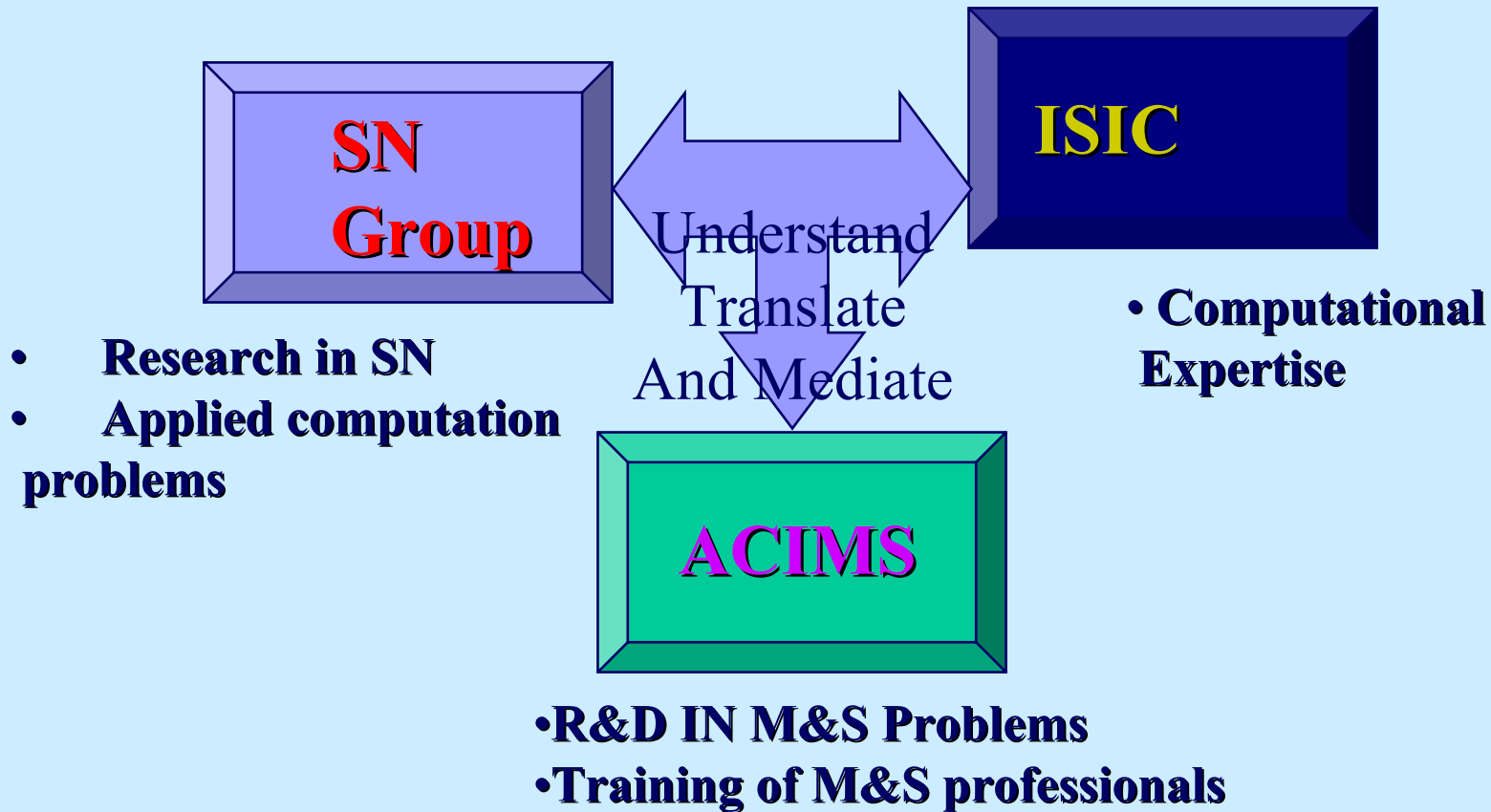
The Autocatalytic Cycle of M&S



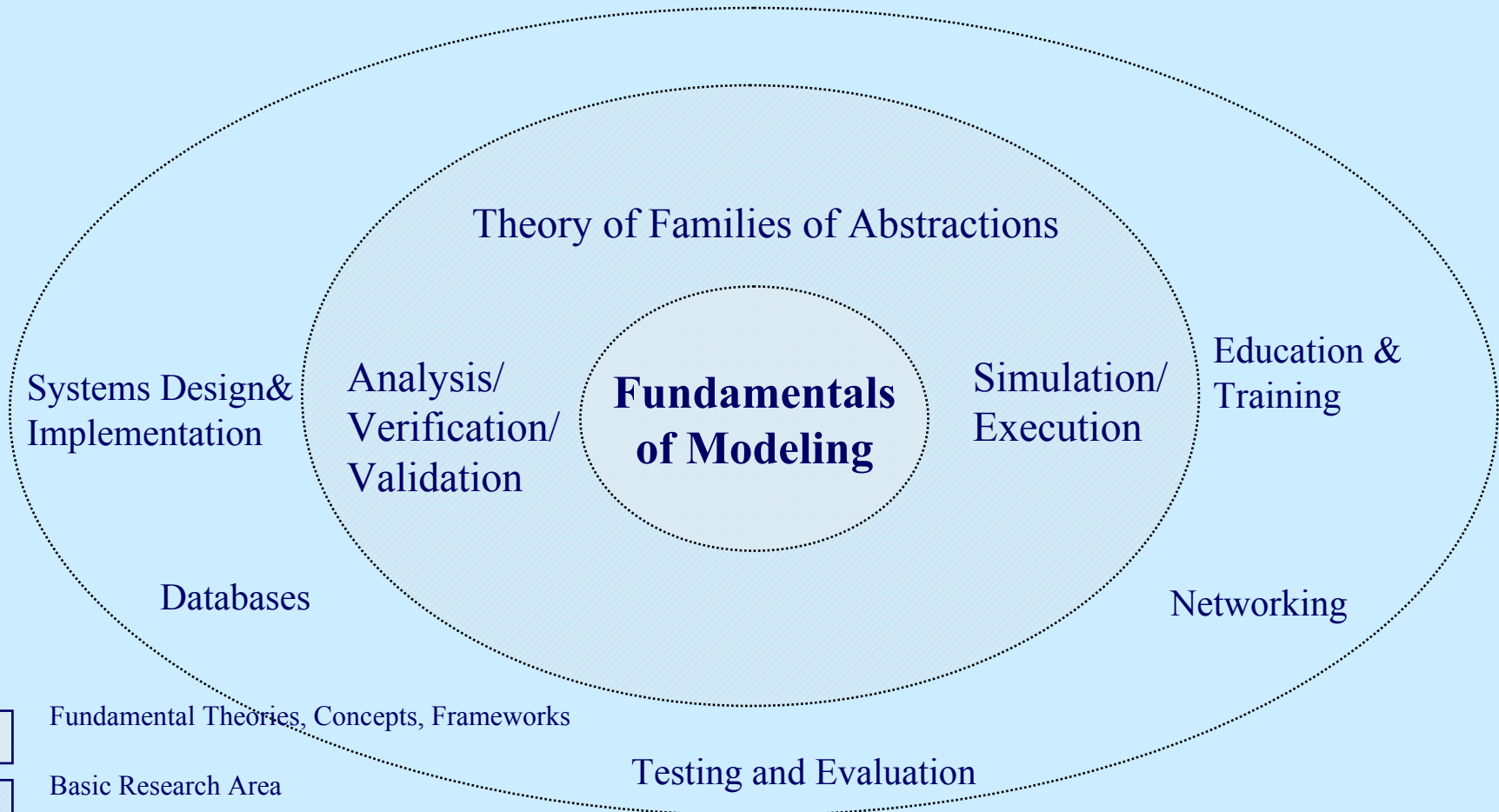
ACIMS Missions

- **Research Mission**
 - basic research on M&S infrastructure
 - apply to challenging problems, e.g. astrophysical systems
 - **Education Mission**
 - degree programs in M&S discipline
 - training of M&S professionals
 - short courses, tutorials
 - **Service Mission**
 - support M&S needs of government and industry
 - engage/spin off companies to market and support developed software
-

Interaction Among SN Group, ACIMS and ISIC



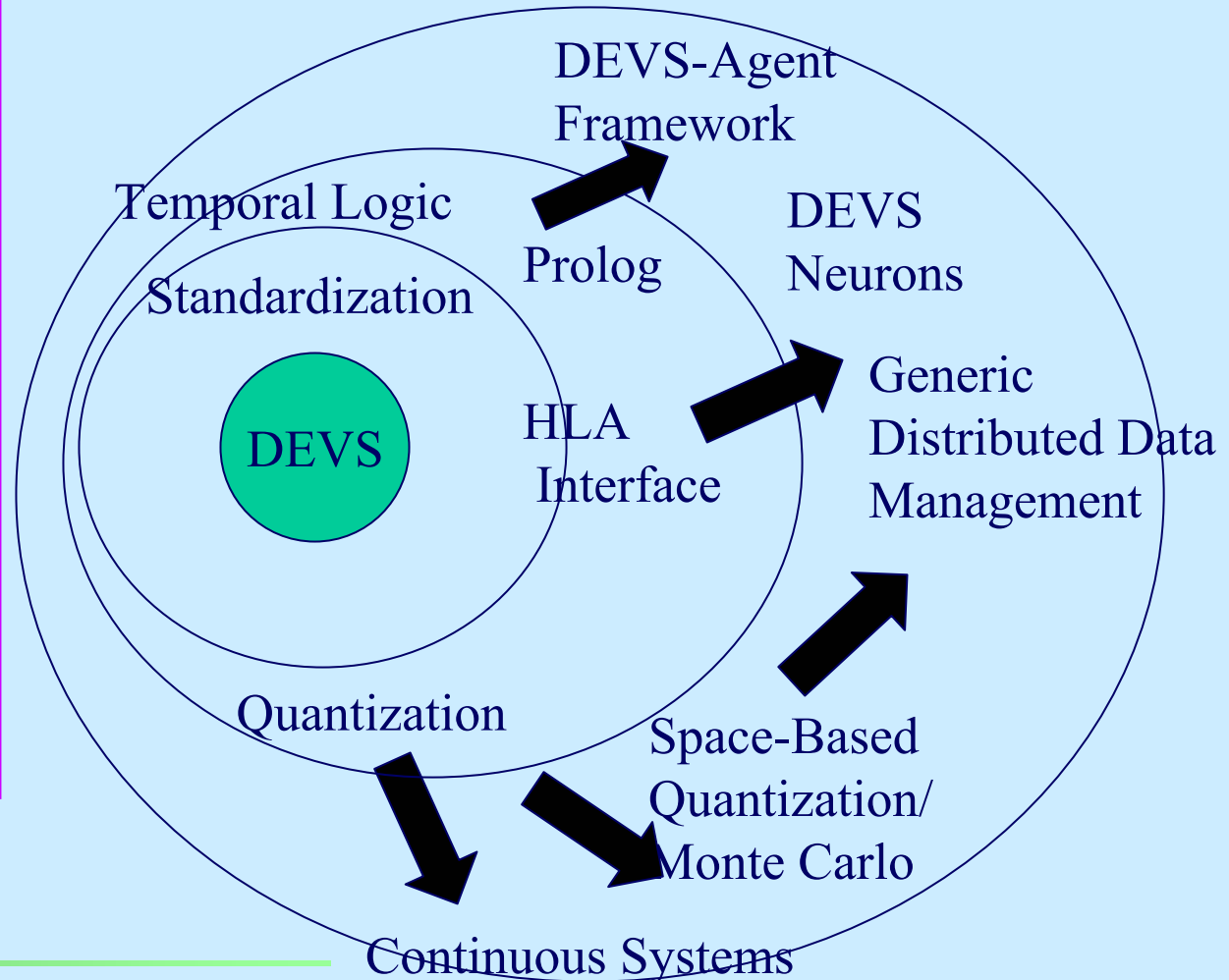
Intellectual Foundations



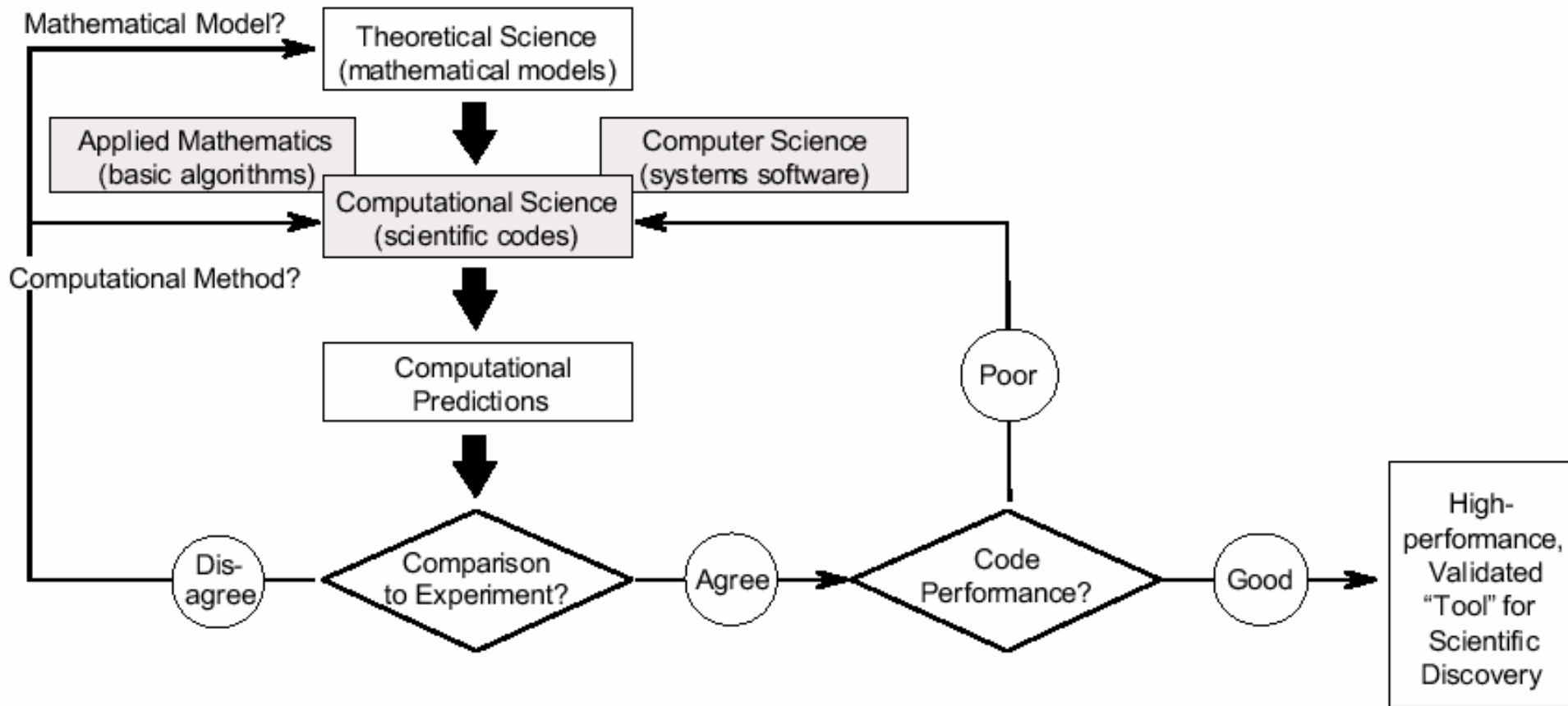
-  Fundamental Theories, Concepts, Frameworks
 -  Basic Research Area
 -  Application Research Areas
-

DEVS-Based Research Directions

- DEVS = Discrete Event System Specification
- Provides sound M&S framework
- Derived from Mathematical dynamical system theory
- Supports hierarchical, modular composition

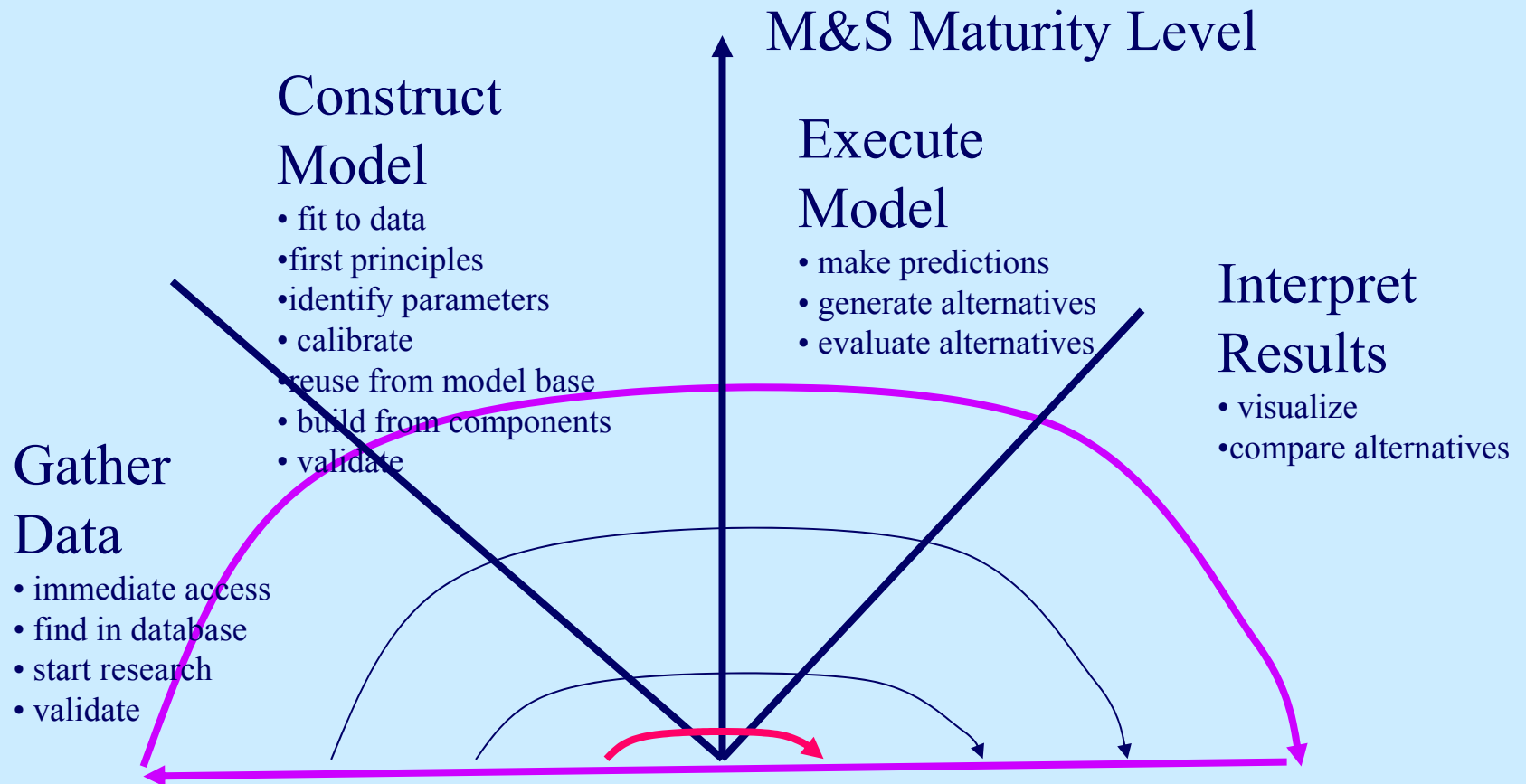


Workflow for the development of scientific M&S codes*



*SciDAC, DOE Report, 2000

M&S Spiral Development Process



*Critical Attributes of M&S software**

- ◆ Performance
- ◆ Portability
- ◆ Adaptability
 - needed because lifetime of code is much longer than theory, methods or technology

Need architecture which is

- ◆ Layered, to facilitate portability
- ◆ Modular, to facilitate adaptability
- ◆ Positive Example: NWCHEM
 - runs on broad range of computers
 - layering allows easy portability
 - modularity allows incorporation of latest sci. into codes
 - high performance: executed on parallel machine

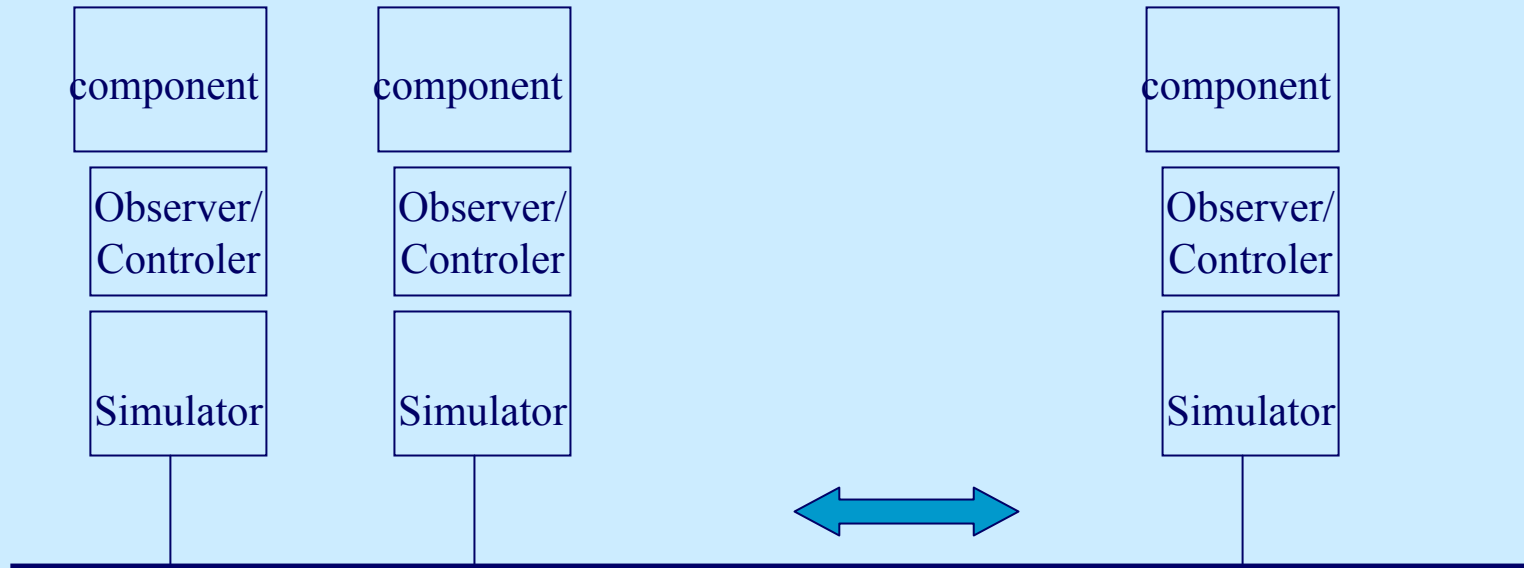
*SciDAC, DOE Report, 2000

BUT, are layering and modularity at odds with performance?

Not necessarily:

- ◆ they facilitate runtime steering
 - observability
 - controlability
 - amendment/migration
 - execution optimization conditioned to state of computation
-

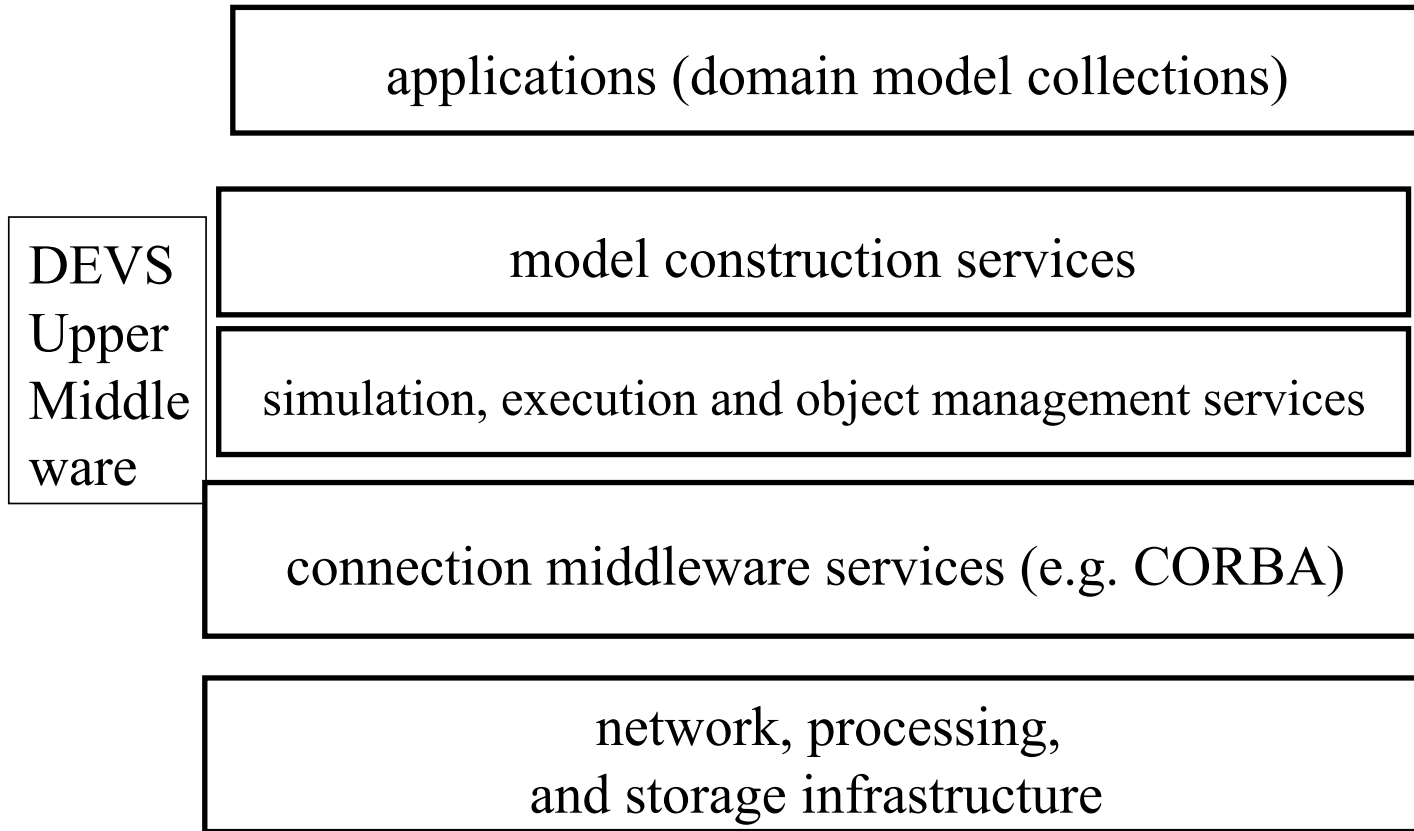
Three Layer Computational Architecture Facilitated by Component Framework



Componentization and separation of code from execution allows

- insertion of observer/controller layer
- online observation of state of computation
- online reallocation of processing resources to components
- e.g., remapping of components to processors

DEVS Middleware



More Information

- ◆ Zeigler, B. P., T. G. Kim, H. Praehofer (2000). *Theory of Modeling and Simulation*. New York, NY, Academic Press.
- ◆ Y. Moon, B.P. Zeigler and G.J. Ball, "The DEVS Environment for High-Performance Modeling and Simulation.", *IEEE Computational Sci. and Engr*, 1997, 4(3): p. 61-71.
- ◆ Zeigler, B.P, H.S. Sarjoughian, and H. Praehofer, (2000) "Theory of Quantized Systems: DEVS Simulation of Perceiving Agents", International Journal of Cybernetics and Systems, Vol. 31, No. 6.

Many more publications and downloadable software are available from Website: www.acims.arizona.edu