DEVS Background

for Model-Based System Engineering

Bernard P. Zeigler
Emeritus Professor University of Arizona

RTSync Corp.
6909 W Ray Rd STE 15-107
Chandler, AZ 85226 USA

Overview

RTSync Corp. Arizona and Maryland

Focus:

Model-based System Engineering (MBSE), Discrete Event System Specification (DEVS) modeling and simulation methodologies and software toolset

Research and Development products

- ***Automated Test Case Generator (ATC-Gen) :*** tactical data conformance testing
- ***MS4 Modeling & Simulation Environment (MS4 Me)*****: DEVS IDE**
- ***Automated Scenario Selection/Construction (ASC) :*** SES-based Simulation generated search
- ***Collaborative Architecture for Model-level Integration (CAMLI):*** DEVS-based Re-usable Component Development

Related Contract Support and Grants:

Missile Defense Agency, SBIRs,

Korean Advanced Defense Development

DoD Info Sys Agency JITC, Dept of Homeland Security, National Science Foundation, NASA,

Collaborations:

- **Business:** Prime Group Solutions (Arizona: MBSE), nSi (Huntsville: Missile Defense M&S)
- **FFRDC**: Oak Ridge National Lab, M&S Group
- **University**: Arizona Center for Integrative Modeling and Simulation (ACIMS) (https://acims.asu.edu) Arizona State Univ, Univ of Arizona, Georgia State Univ, Univ of Corsica, Univ of Nice and CNRS in France, Univ of Wismar, Germany

Books :

"Model Engineering for Simulation", Editors: L. Zhang, B. P. Zeigler, and L. Lian

"Theory of Modeling and Simulation Third Edition", B. P. Zeigler. Alexandre Muzy, E. Kofman, 2018

"Guide to Modeling and Simulation", B. P. Zeigler and H. Sarjoughian , 2nd Edition, 2017

"Modeling and Simulation-based Data Engineering Introducing Pragmatics into Ontologies for Net-Centric Information Exchange", B. P. Zeigler and P. Hammonds, 2007

Contacts

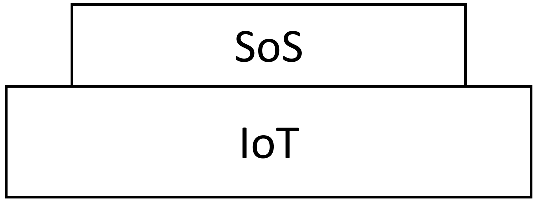
For more information , contact info@rtsync.com or POCs ;

Dr. Bernard P. Zeigler Chief Scientist zeigler@rtsync.com

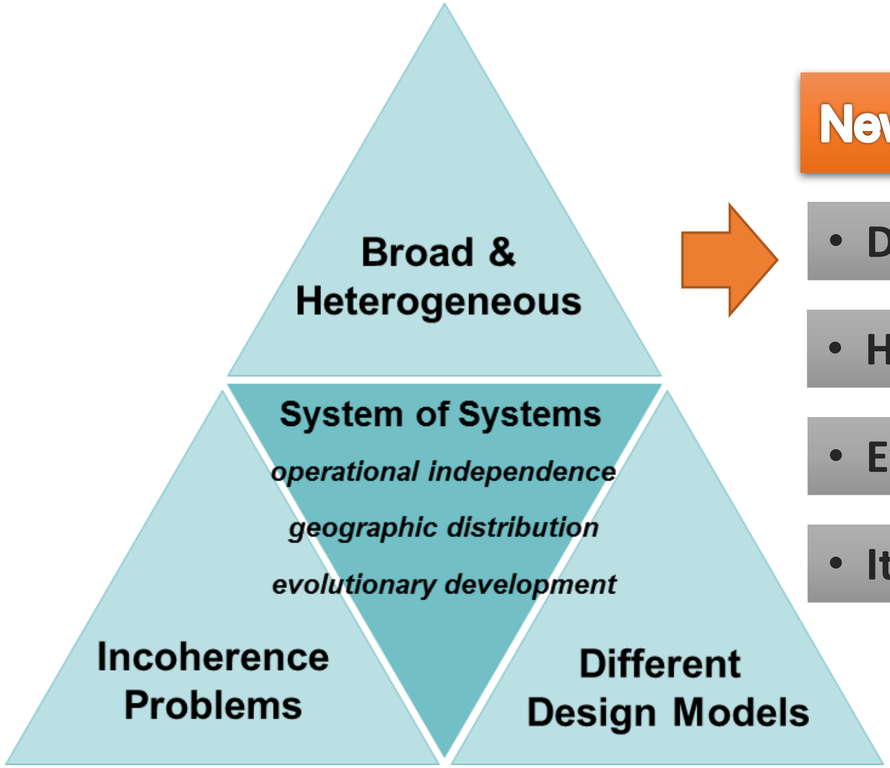
Dr. Doohwan Kim President & CEO dhkim@rtsync.com

System of systems (SoS) **is a collection of task-oriented systems that pool their resources and capabilities together to obtain a new, more complex, 'meta-system' which offers more functionality and performance than simply the sum of the constituent systems** **(from Wikipedia)**

- *<u>Energy</u> : Smart grid system, nuclear power plant infrastructure,*
- *<u>National Infrastructure</u> : intelligent transportation infrastructure*
- *<u>Defense</u> : C4ISR, Net-centric warfare*
- *<u>Security</u> : Homeland Security, Boarder Surveillance*
- *<u>Space</u> : Planetary colonies*



- System of Systems(SoS) contains high degree of complexity and current methodology is not capable of designing and development as well as operating the SoS.



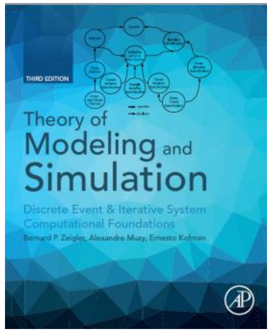
New Capability for Development Methodology

- Different level of system abstraction
- Hierarchical representation of system structure
- Evaluation of different design alternatives
- Iterative development methodology

Theory of Modeling and Simulation

Levels of System Specification

Level	Specification Name	What we know at this level	Example: A Person in a Conversation
0	Observation Frame	How to stimulate the system with inputs; what variables to measure and how to observe them over a time base;	The person has inputs and outputs at the usual cognitive level, such as streams of words
1	I/O Behavior	Time-indexed data collected from a source system; consists of input/output pairs	For each input that the person recognizes, the set of possible outputs that the person can produce
2	I/O Function	Knowledge of initial state; given an initial state, every input stimulus produces a unique output.	Assuming knowledge of the person's initial state when starting the conversation, the unique output response to each input.
3	State Transition	How states are affected by inputs; given a state and an input what is the state after the input stimulus is over; what output event is generated by a state.	How the person transits from state to state under input words and generates output words from the current state
4	Coupled Component	Components and how they are coupled together. The components can be specified at lower levels or can even be structure systems themselves – leading to hierarchical structure.	A description of a person's I/O behavior in terms of neural components and their interaction by spikes is at this level.



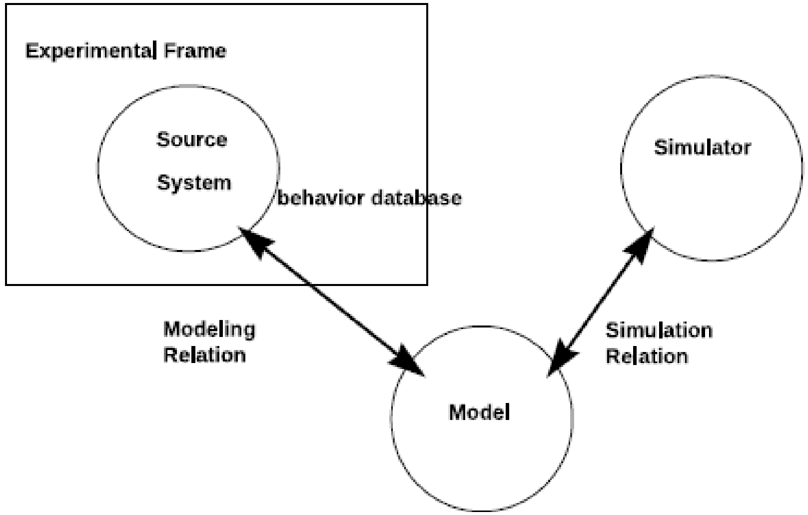
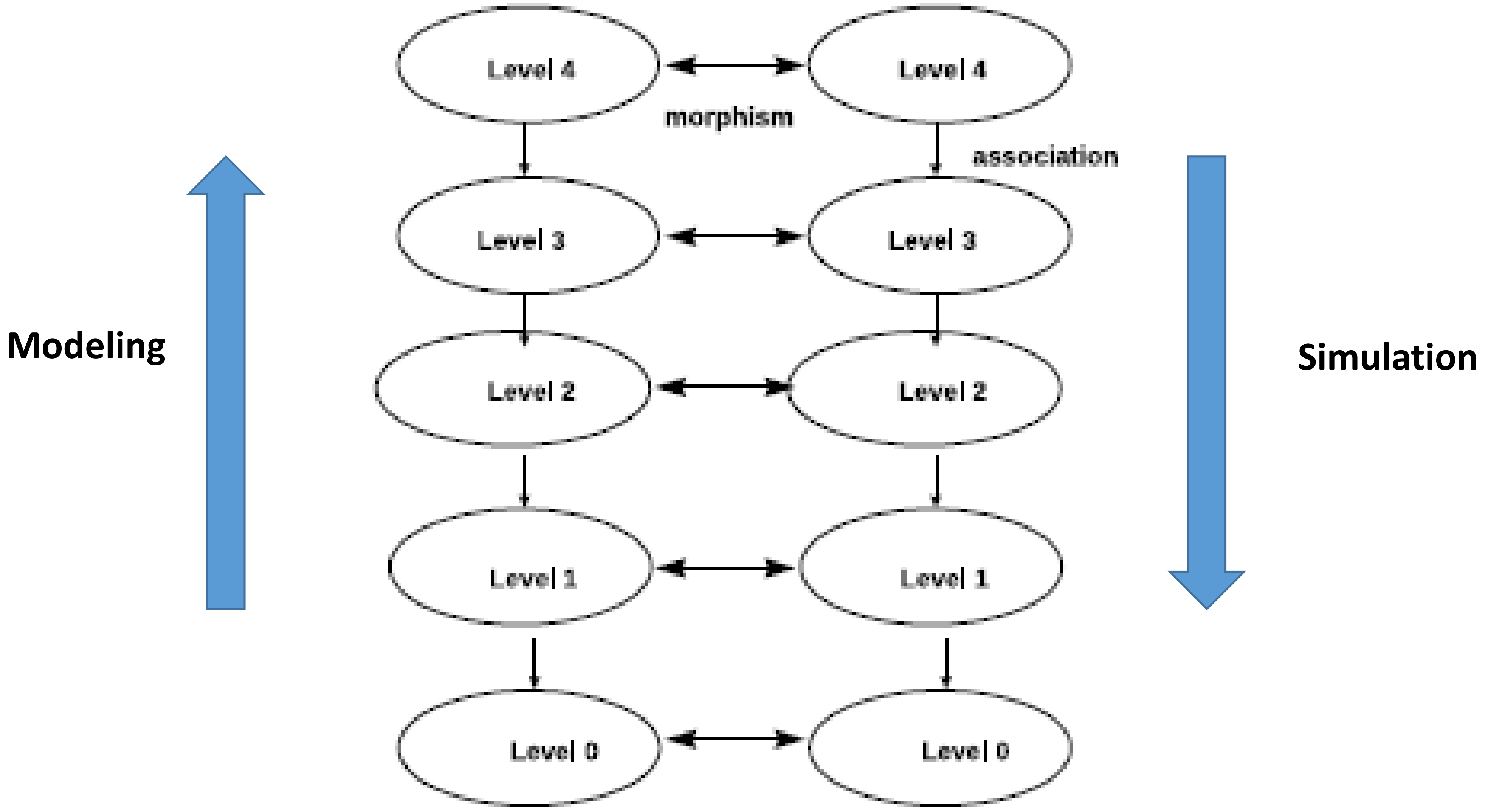


FIGURE 2.1

The Basic Entities in M&S and Their Relationships.

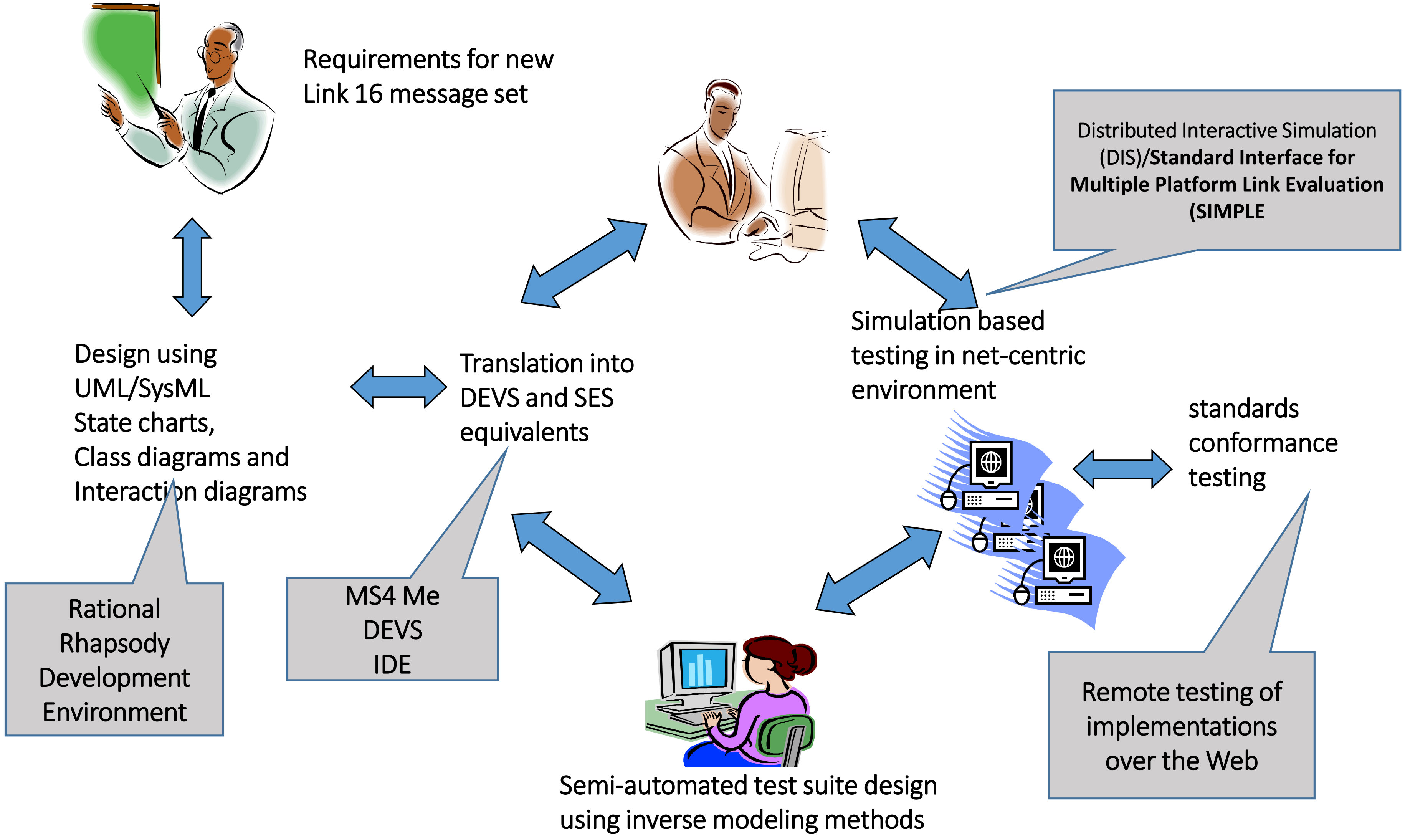
Equivalences at each Level of System Specification



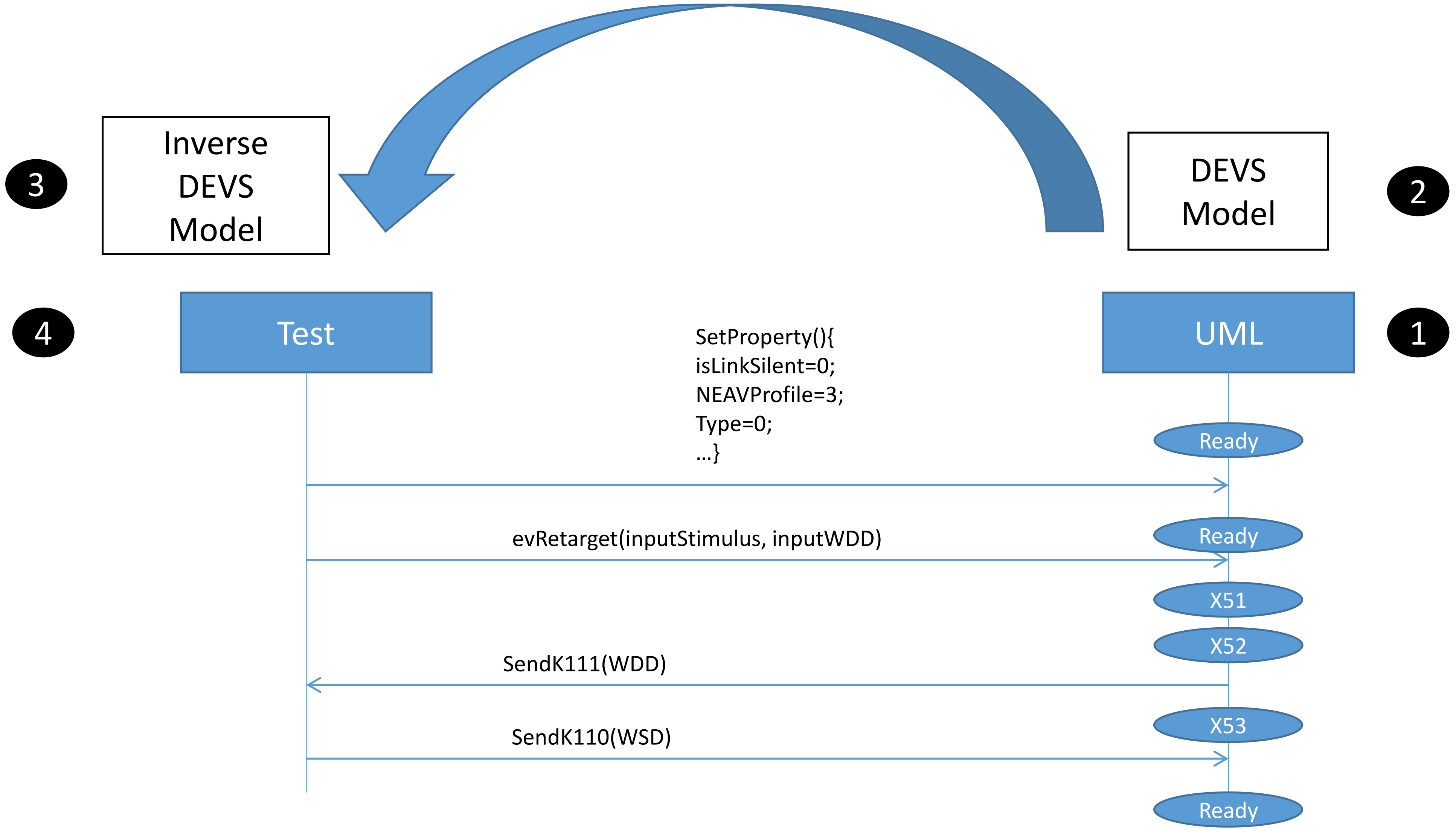
DEVS Systems-based Simulation is applicable to wide spectrum complex systems

- Theoretical – supporting application
  - Closure under coupling, universality, uniqueness, relation to other formalisms
  - Hierarchical Model Construction supports complex system development
  - Supports the correctness of the algorithms and validation of the executing models, e.g., time management is rigorously defined
  - Discrete Event System Specification (DEVS) formalism can be easily expanded beyond discrete-event world to continuous characteristics of system. (Hybrid-DEVS)
- In Application
  - Models, Simulators and Experimental Frames are distinct entities with their own software representations
  - Precise and well-defined mathematical representation
  - Models/Experiments are developed systematically for interoperability
  - Repositories of models/experiments created and maintained systematically
  - Components can be easily reused for constructing new models
  - Discrete-event basis improves performance (e.g. no need for a global clock to control timing)
  - DEVS software can be deployed on distributed computing environments and interact with heterogeneous M&S system.

DEVS Model-Based Systems Engineering



DEVS-Based Test Development

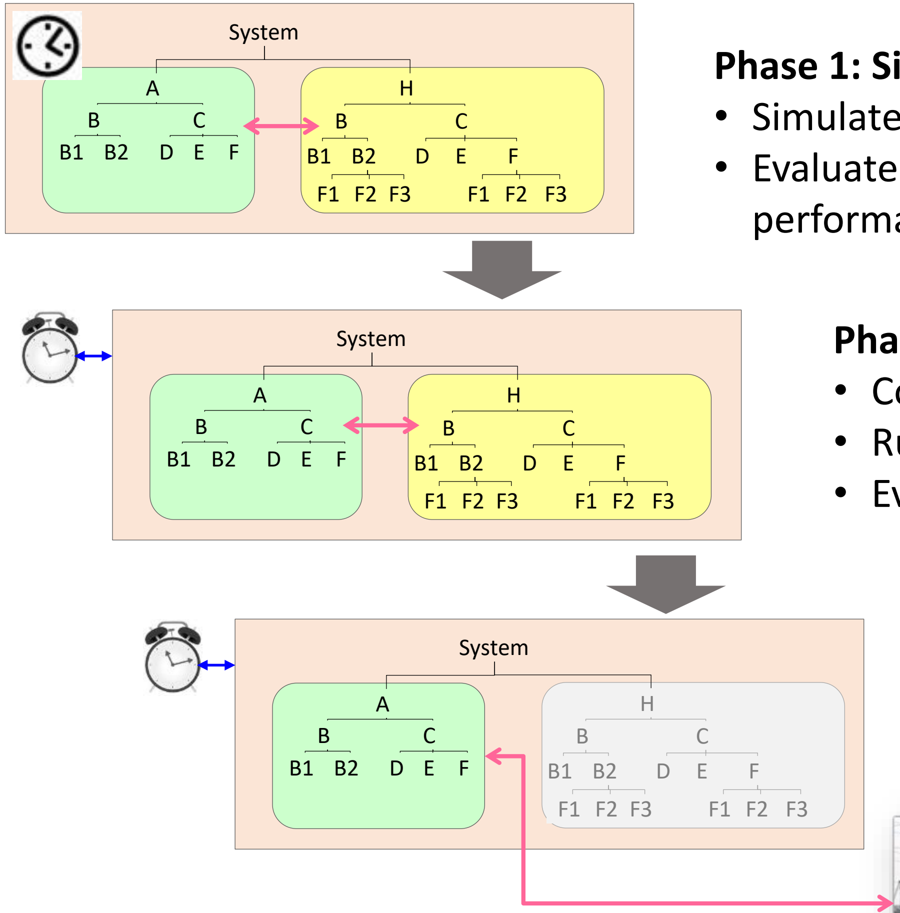


Transform models for real time application

- Same DEVS models can be designed, evaluated and transformed to operating system for actual real application.

Monitor & Control System (ex.. SCADA)

System under test (ex.. Plant, Power System)



Phase 1: Simulation by simulation clock

- Simulate models and SES with simulation clock
- Evaluate various design specifications and performance

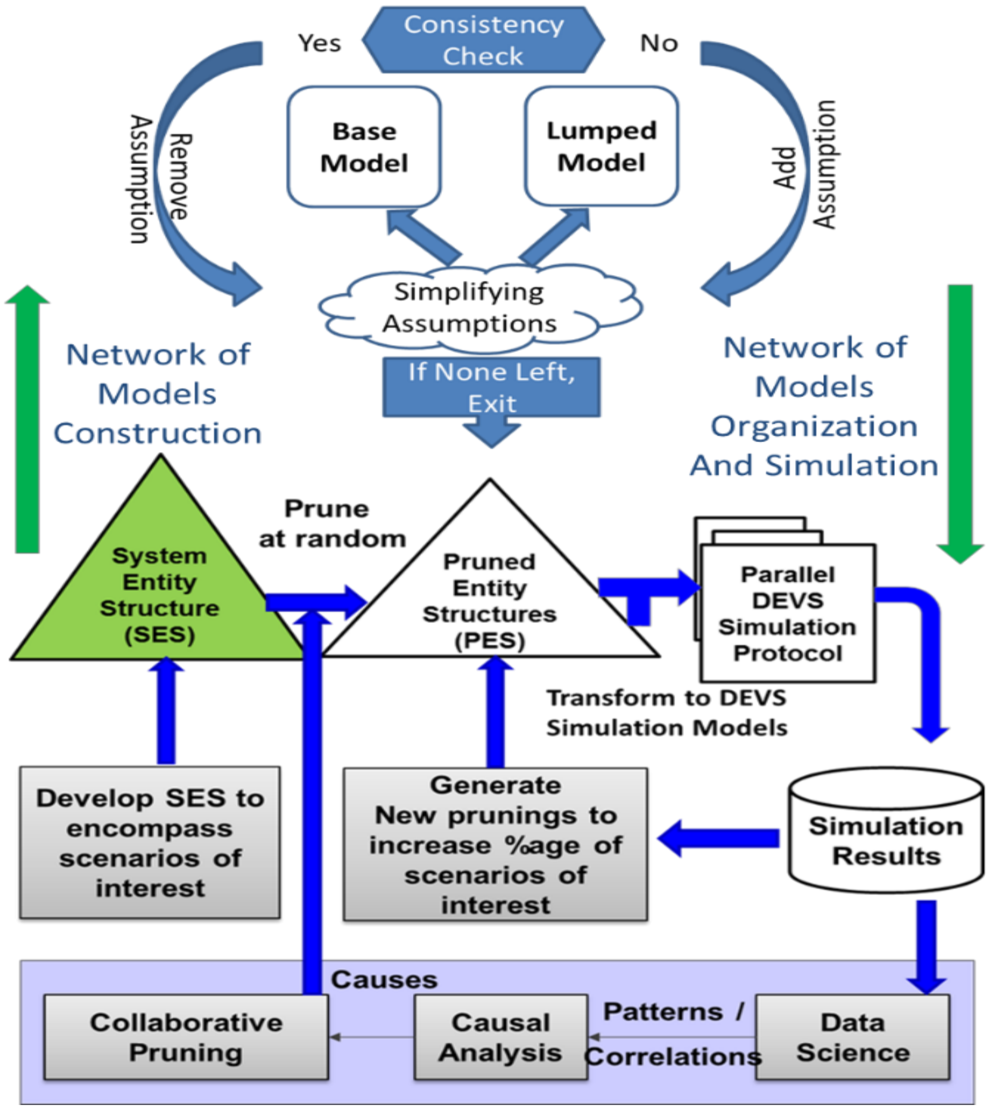
Phase 2 : Simulation by real-time clock

- Continue to use same models of phase 1
- Run simulation with real-time clock
- Evaluate real-time characteristics of system

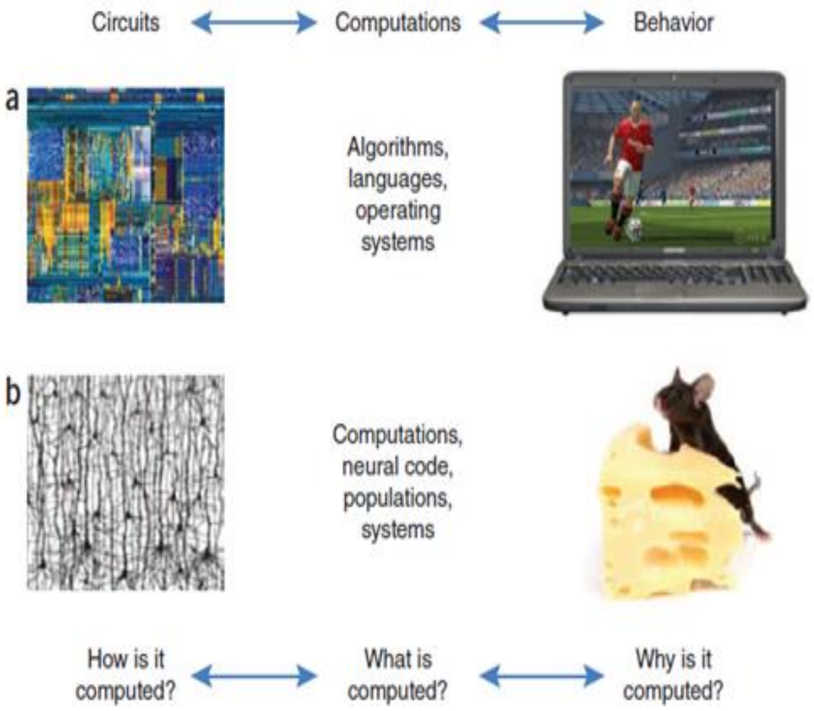
Phase 3 : Transform models to real application

- Continue to use same models of phase 2
- Directly interface to real world application

DEVS-Based Multi-Resolution Model Development

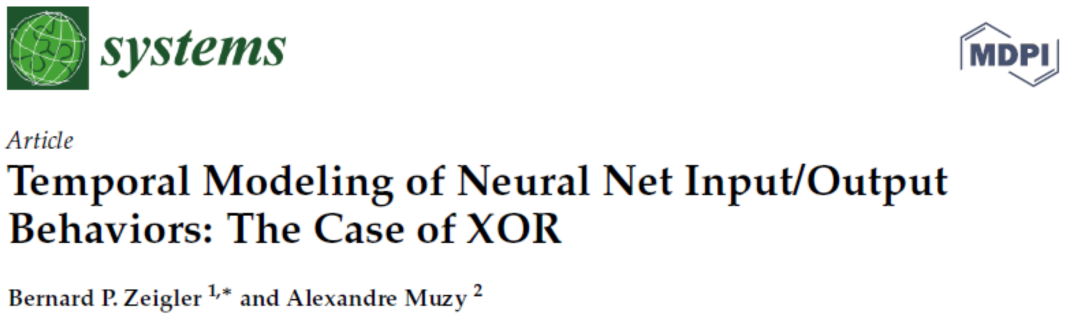


Course in Neurocognition with A. Muzy at U. Nice



Some implications:

- DNN may overemphasize recognition
- May be impractical in real-time applications

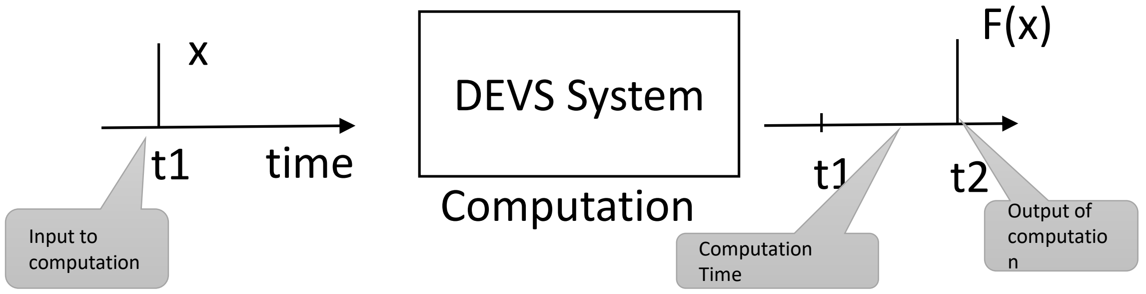

systems

MDPI

Article

Temporal Modeling of Neural Net Input/Output Behaviors: The Case of XOR

Bernard P. Zeigler [1,*] and Alexandre Muzy [2]



Some Considerations in temporal computation

- Time dispersion of input events
- Coincidence of events
- End-to-end computation time
- Waiting time before a new input can be submitted
- Accuracy/Speed Tradeoff