



Integrity ★ Service ★ Excellence

Systems and Software

06 March 2013

Dr. Kathleen M. Kaplan

AFOSR

Air Force Research Laboratory

Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE 06 MAR 2013		2. REPORT TYPE		3. DATES COVERED 00-00-2013 to 00-00-2013	
4. TITLE AND SUBTITLE Systems and Software				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Office of Scientific Research ,AFOSR/RTC,875 N. Randolph,Arlington,VA,22203				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution unlimited					
13. SUPPLEMENTARY NOTES Presented at the AFOSR Spring Review 2013, 4-8 March, Arlington, VA.					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT Same as Report (SAR)	18. NUMBER OF PAGES 11	19a. NAME OF RESPONSIBLE PERSON
a. REPORT unclassified	b. ABSTRACT unclassified	c. THIS PAGE unclassified			



2013 AFOSR SPRING REVIEW



NAME: Systems and Software

BRIEF DESCRIPTION OF PORTFOLIO:

- **Enable quantifiable performance evaluation of critical systems**
- **Manage environments in order to preserve vital mission functions**
- **Comprehensively understand distributed effects in large infrastructures to predict global system failures**

LIST SUB-AREAS IN PORTFOLIO:

- **Composeable Dynamic Models**
- **Formal Analysis and Verification**
- **Assessment/Repair of Failure**



Current Program Scope



- **Composeable Dynamic Models**
 - New programming languages or language constructs reduce errors at run-time
 - Domain-specific languages enhance capabilities for code generation
- **Dynamic Formal Analysis and Verification**
 - Verification of system properties based on formal specifications
- **Assessment/Repair of Failure**
 - Abstract models of systems and their interactions facilitate automated generation of code



Scalable Model Checking

C. Tinelli U Iowa, C. Barret, NYU

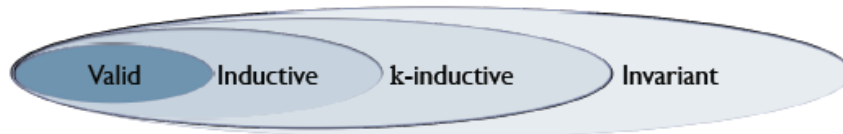


Approach: Formal verification suffers from state space explosion.
Compactly represent logical symbols in scalable nested satisfiability modulo theory (SMT)

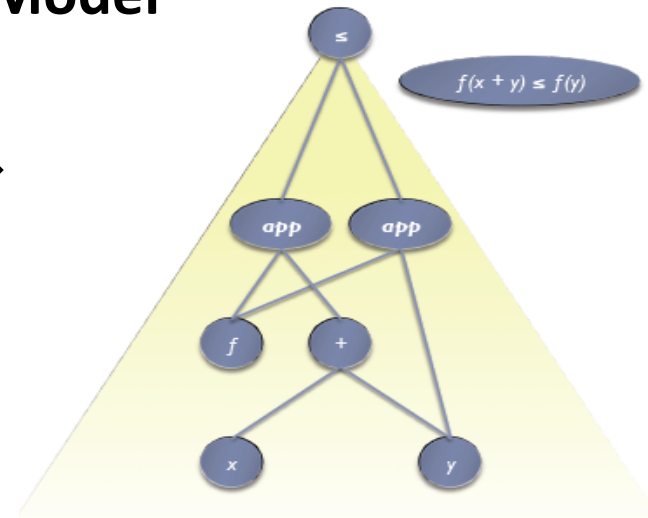
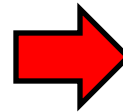
Payoff: More scalable verification to handle large heterogeneous systems

Compact SMT Language

- ▶ **Valid:**
 - ▶ satisfied by all states in Q
- ▶ **Inductive:**
 - ▶ $I(s_0) \models P(s_0)$,
 - ▶ $P(s_n), T(s_n, s_{n+1}) \models P(s_{n+1})$
- ▶ **k-inductive:**
 - ▶ $I(s_0), T(s_0, s_1), \dots, T(s_{k-1}, s_k) \models P(s_0), \dots, P(s_k)$,
 - ▶ $T(s_n, s_{n+1}), \dots, T(s_{n+k}, s_{n+k+1}), P(s_n), \dots, P(s_{n+k}) \models P(s_{n+k+1})$
- ▶ **Invariant:**
 - ▶ satisfied by all reachable states of S



Improved Lower Dimensional Model





Stochastic Methods for Dynamic Scalable System Verification

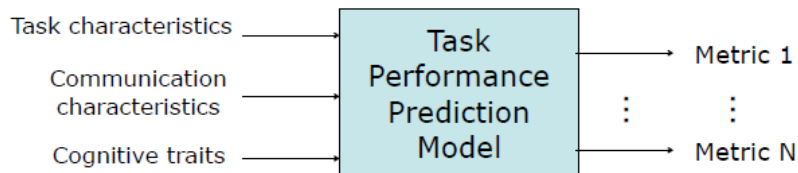
Mark Campbell, Cornell



Approach: Heterogeneous and uncertain states characterize system performance across multiple levels of software. Using stochastic models can enable robust characterization for system performance verification

Payoff: Computationally tractable ways of system performance verification at multiple layers of software including human interaction

System Performance Model

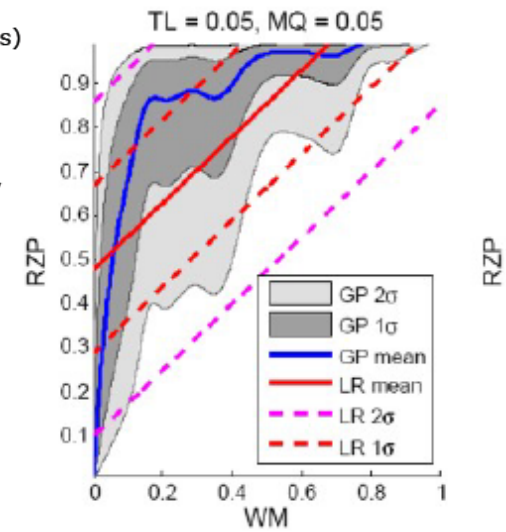


System Performance Model

TL = Task load (# enemies)

MQ = Message quality
(% relevant messages)

WM = Working memory
(OSPAN)
= Gaussian noise
(fixed std dev)





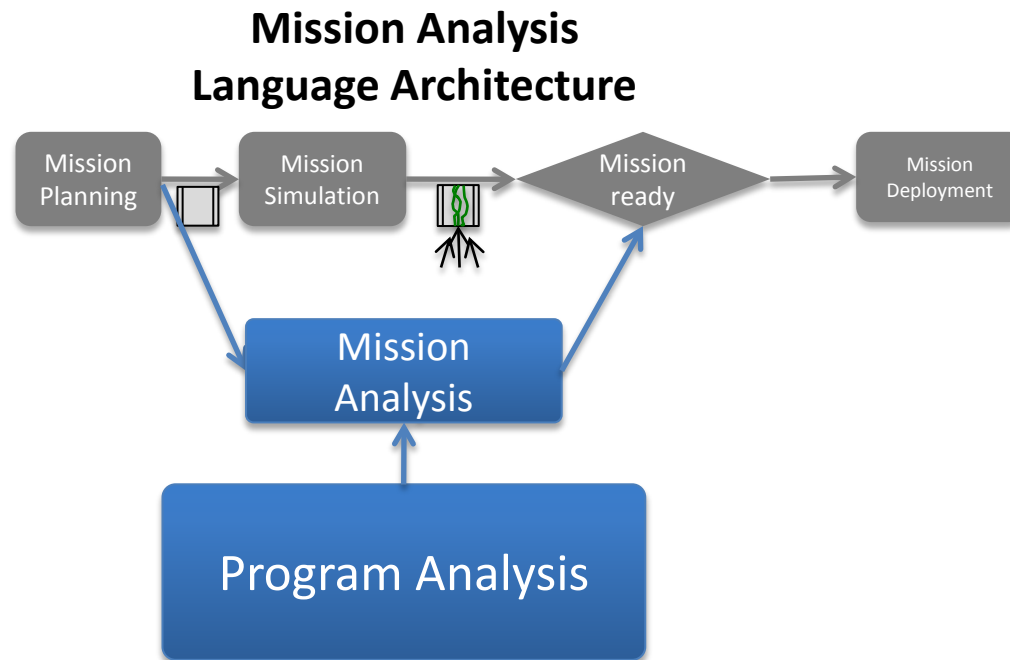
Mission Verification

Elbaum, Dwyer U. Neb., Rosenblum



Approach: Develop a language to represent mission scenarios tied to integrated distributed software architecture.

Payoff: Verify global mission properties as function of lower level software constructs for quantifiable fault tolerance in achieving mission objectives



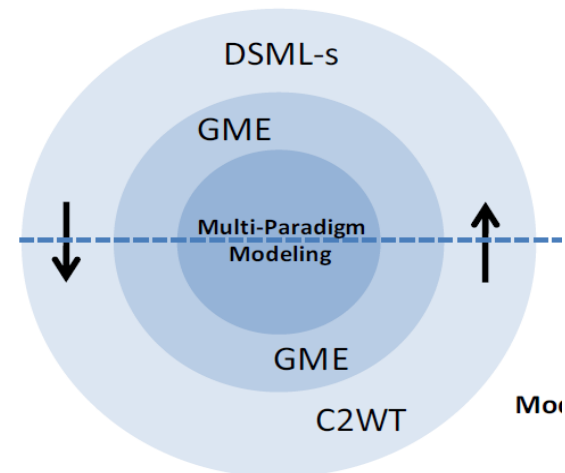


Systems and Software

AFRL Tech Directorate Interest/Coordination



- Information Directorate
 - Systems and Software Producibility
 - Multi-core Computing
- Air Vehicles
 - Flight-critical systems and software
 - Mixed-criticality architectures
- Human Effectiveness
 - Modeling of human-machine systems
 - Meta-information portrayal STTR
 - Robust Decision Making
 - Large Scale Cognitive Modeling/C2WT





Increased Scale/Integration via DSMLs Anchored in DEVS



DEVS (*discrete event system specification*)

- Formal rigor
- Model reusability
- Interoperability

A *discrete event system specification (DEVS)* is a mathematical structure (7-tuple)

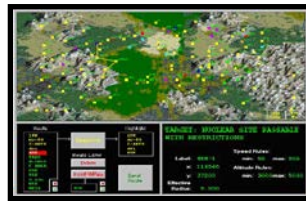
$$M = \langle X, S, Y, \delta_{int}, \delta_{ext}, \lambda, ta \rangle$$

where

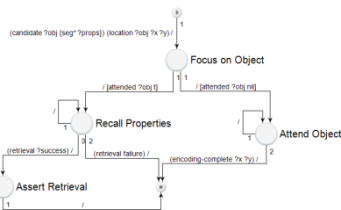
X	is the set of input values
S	is a set of states
Y	is the set of output values
$\delta_{int} : S \rightarrow S$	is the internal transition function
$\delta_{ext} : Q \times X \rightarrow S$	is the external transition function
$\lambda : S \rightarrow Y$	is the output function
$ta : S \rightarrow R_{0,\infty}$	is the time advance function



Navigator



Plans routes from targets to targets under constraints



Domain-Specific Languages

- Tailored for cognitive modeling
- Semantically anchored in DEVS



High-Performance Computing

- Scalable simulation infrastructure
- Exploiting 25 years of DEVS

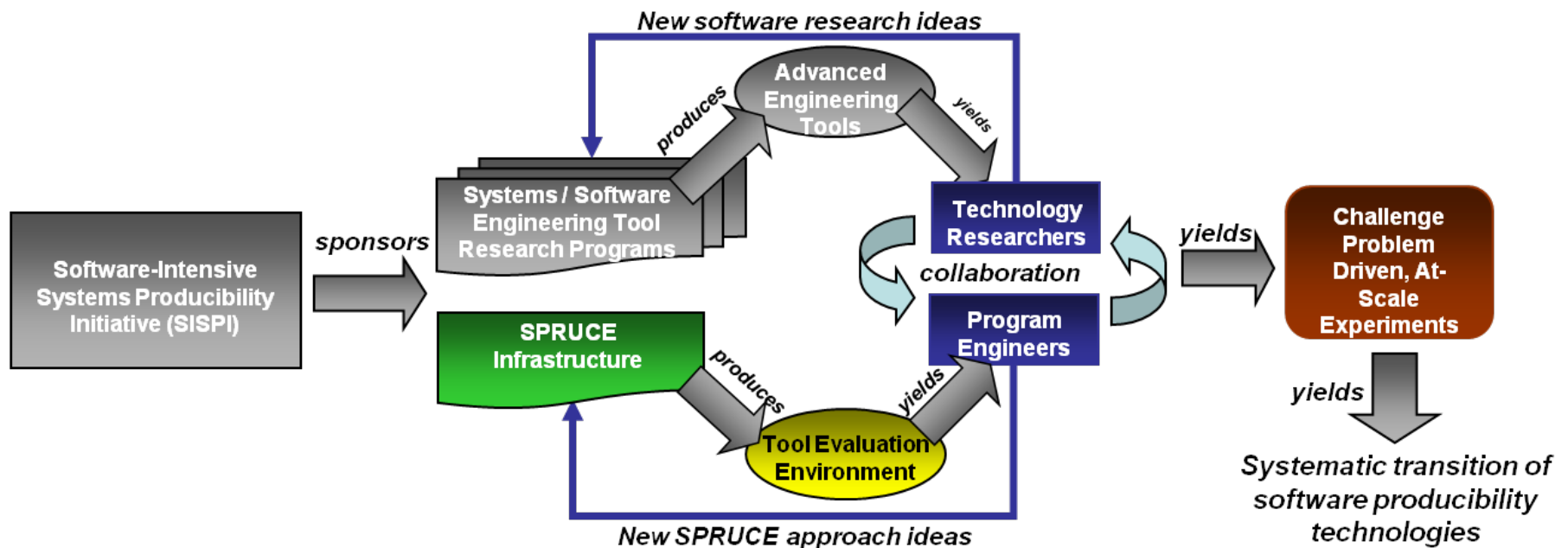


Drager/RI



Approach: Use parallel processing resources and network infrastructure as means of emulating and detecting system faults

Payoff: Far fewer defects and more detailed assessment of integrated system performance





Collaborations at AFOSR

- **Information Operations and Security**
 - Fundamental software constructs for system security
- **Information Fusion**
 - Signal and sensor processing for integration of large data into systems architectures
- ***Complex Networks***
 - Mathematical and statistical methods for network and networked systems
- ***Foundations of Information Systems***
 - Measurement and statistical verification for software, network, and hardware
- **Computational Mathematics**
 - Methods of computational modeling of large complex physical processes
- **Dynamic Data Driven Applications Systems**
 - Strategies for real time feedback of data into distributed computational processes
- **Optimization and Discrete Mathematics**
 - Optimization strategies and algorithms for discrete computational processes
- ***Dynamics and Control***
 - Dynamical systems theory for assessment of performance of control architectures



Systems and Software

Agency Interaction & Funding Agencies



Agency Interactions:

- **OSTP/NITRD Coordinating Group**
 - High Confidence Systems and Software (HCSS)
- **ASDR&E**
 - Software Producibility Initiative
- **NSF**
 - Cyber Physical Systems
- **NASA**
 - V&V of Flight Critical Systems
 - Ames Research Laboratory

Other Funding Agencies

- **ARO**
- **ONR**
- **DARPA**