

MODEL-BASED SYSTEMS ENGINEERING FOR CYBER-PHYSICAL SYSTEMS

Mark Austin, Department of Civil and
Environmental Engineering and ISR, University of
Maryland, College Park, MD 20742, USA.

Keynote Presentation

International Conference on Systems (ICONS 2014),

Nice, France, February 26, 2014



The
Institute for
Systems
Research

THE A. JAMES CLARK SCHOOL of ENGINEERING
UNIVERSITY OF MARYLAND

©2010 University of Maryland

OVERVIEW

Model-Based Systems Engineering at Maryland

- Pathway to University of Maryland
- Model-Based Systems Engineering (MBSE) at ISR
- Our view of MBSE and CPS.

Current Research on CPS

- Ontology-enabled traceability. Semantic platforms for simulation and control of trains and buildings.
- Ontologies of time and time-based reasoning. Ontologies of time and space, and spatio-temporal reasoning.

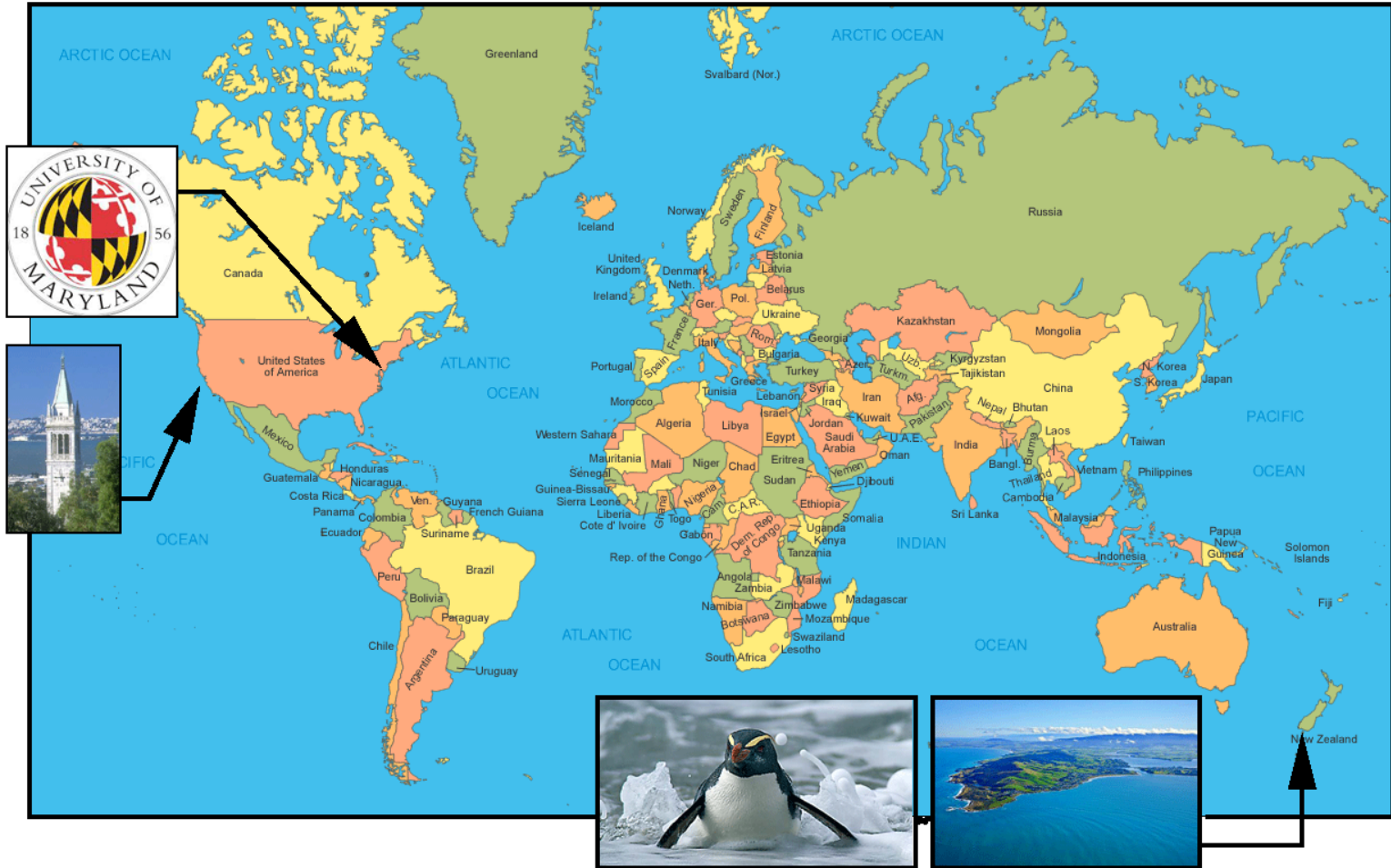
Acknowledgements / Co-Workers

- **At UMD:** Cari Wojcik, John Baras, Reza Ghodssi, Matt Mosteller, Karam Rajab, Nefretiti Nassar, [Parastoo Delgoshaei](#), Eddie Tseng, and [Leonard Petnga](#).
- **At NIST:** Conrad Bock, Daniel Veronica and Amanda Pertzborn.

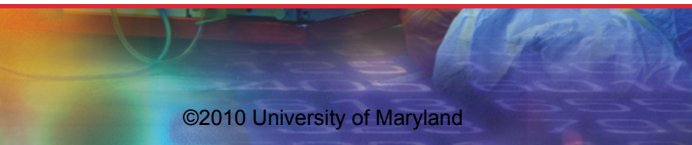


BACKGROUND

Pathway to University of Maryland



The
Institute for
Systems
Research
THE A. JAMES CLARK SCHOOL of ENGINEERING
UNIVERSITY OF MARYLAND



INSTITUTE FOR SYSTEM RESEARCH

38 joint appointment faculty,
26 affiliated faculty, **7** research
 faculty, **1** Professor of the
 Practice
 in **4** colleges and **14**
 departments

15 postdoctoral researchers
 leveraging research programs

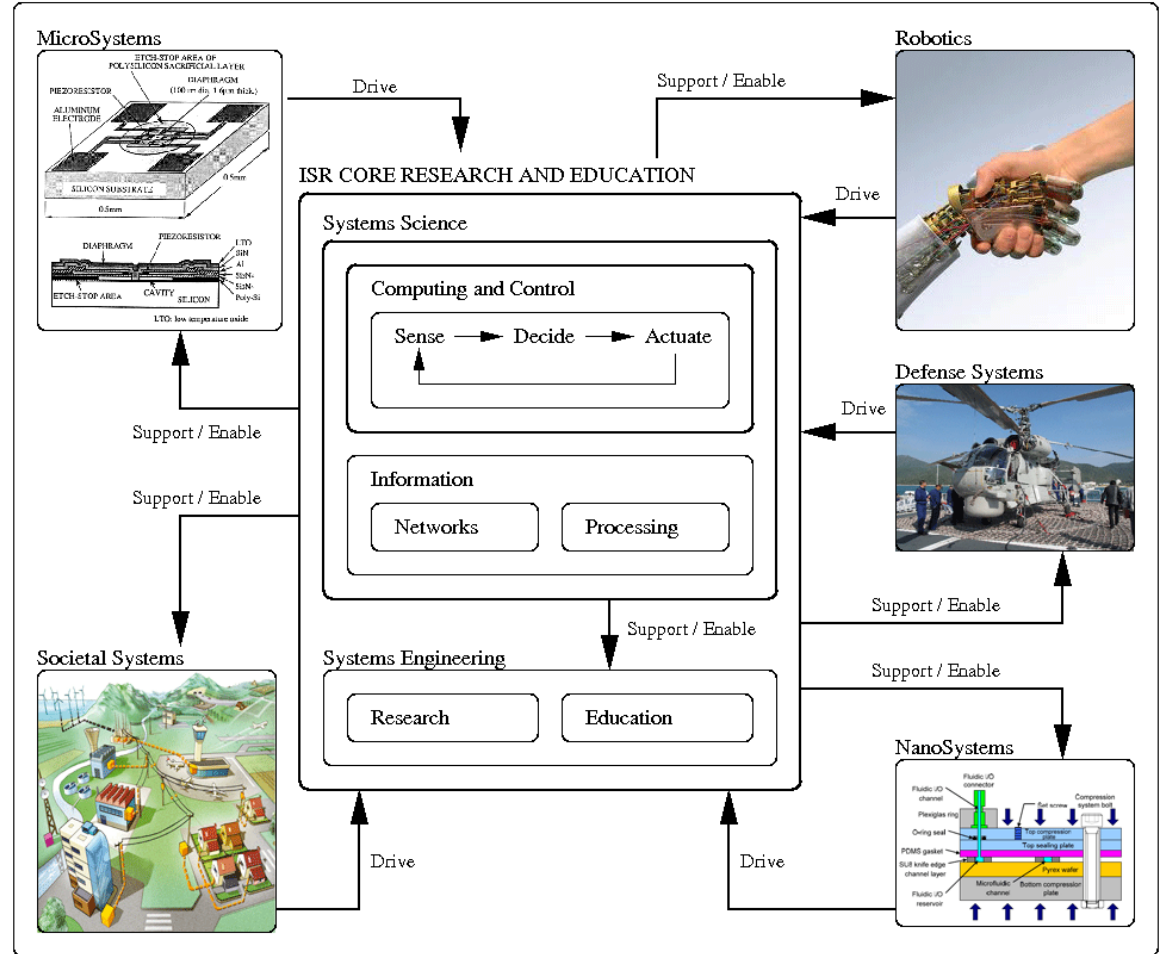
200+ research graduate
 students

45 MS Systems Engineering/
 ENPM graduate students

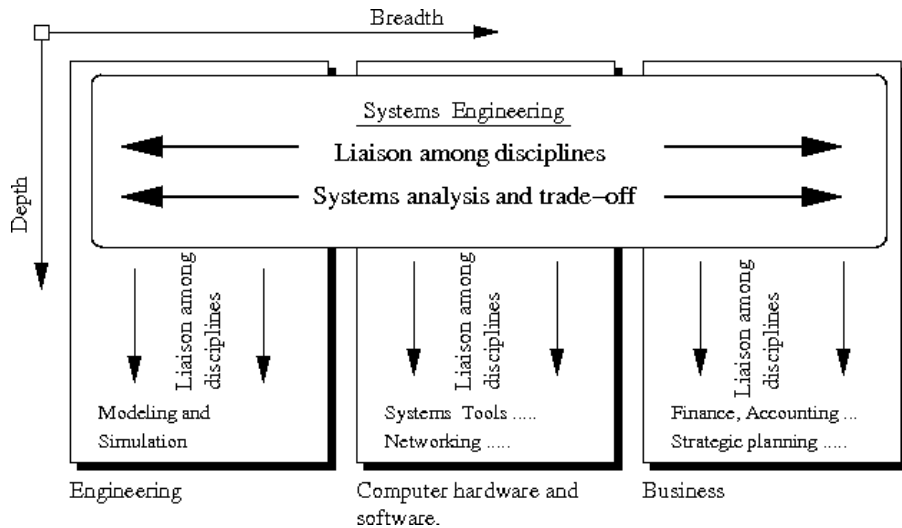
55 undergraduate students in
 SE Projects course

9 undergraduate students in
 research programs

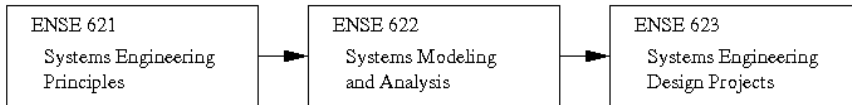
ISR APPLICATION LAYER



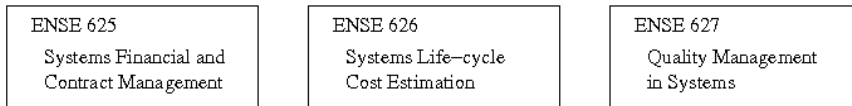
SYSTEMS ENGINEERING EDUCATION AT ISR



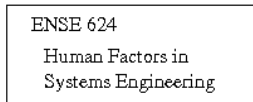
Engineering School : Systems Engineering Core



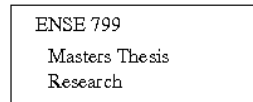
Business School Components



Department of Computer Science



Multi-disciplinary Research



NEW FOR FALL 2010

ENES 489P

SPECIAL TOPICS IN ENGINEERING

HANDS-ON SYSTEMS ENGINEERING PROJECTS

WOULD YOU LIKE TO UNDERSTAND:

- How to master system complexity?
- How to build systems to meet time and budget requirements?
- How to build systems that can be easily verified and validated?
- How to control risk?
- How to design safe systems?

This course will be a great opportunity for senior-level undergraduates and graduate students in all engineering disciplines. You'll get the chance to work in teams on hands-on, complex systems design in collaboration with industry and government experts.

Be among 10 select groups in the country to be introduced to the new area of systems engineering. Systems engineering is rapidly developing as a much-sought-after career path for engineers of all kinds and is proven to be a critical factor for U.S. competitiveness.

Get ahead of your class and get introduced to the emerging model-based systems engineering discipline!

INSTRUCTORS Professor Mark A. Austin and Professor David L. ...

LECTURE NOTE TIME CHANGE Tuesdays, 5:00-6:00 p.m. SEIL Lab, 2250 ...

LAB Thursdays, 3:30-6:00 p.m. SEIL Lab, 2250 ...

CLASS LIMIT 20 students

LEARNING OBJECTIVES ...

3 CREDITS

www.isr.umd.edu

COMING IN SPRING 2014 !

ENCE 688R

ADVANCED TOPICS IN CIVIL ENGINEERING

CIVIL INFORMATION SYSTEMS

WOULD YOU LIKE TO UNDERSTAND:

- How to develop engineering software in Java and Python?
- How to develop graphical user interfaces in Java?
- How to use Java and Python with XML?
- How to model the structure and behavior of civil systems?
- Data structures and algorithms for the modeling and analysis of networked infrastructure systems?

GOALS

This course will be a hands-on introduction to engineering software development for the model-based design of civil infrastructure systems. You will learn how to model and analyze the structure and behavior of civil systems drawn from building and urban infrastructure.

COMING IN FALL 2013 !

ENSE 698E

SPECIAL TOPICS IN SYSTEMS ENGINEERING

SENSOR SYSTEMS

COURSE DESCRIPTION

This course covers sensors and sensing systems for a variety of engineering applications from domains including but not limited to aerospace, civil, electrical, mechanical, and systems engineering.

The course will comprehensively cover all aspects of sensor systems, including: sensor mechanisms, physics-based analytical and simulation models of sensors, signal representation, signal processing, noise and other sources of interference, interfaces, statistical analysis of sensor outputs, networks of sensors, and system-level analysis and design including specifications, requirements, and validation.

Following current industry paradigms, students will work in teams to complete a design project addressing simulation analysis and optimization of a sensor system. This course encourages multidisciplinary participation and will be offered to graduate students in all departments of the A. James Clark School of Engineering.

INSTRUCTORS Associate Professors Pamela Abshire and David Lovell

LECTURE Tuesday and Thursday 12:30-1:45 p.m., 3102 EGR

CLASS LIMIT 20 students

CROSS-LIST ENCE 688R

CREDITS 3 (three)

LEARN MORE ONLINE!

ter.ps/ENSE698E



MSSE PROGRAM

*Model-Based Systems Engineering in a Systems Research Environment
(30-credits)*

MBSE Design Sequence (9-credits)

- Systems Concepts, Issues, and Processes
- Systems Requirements, Design, and Trade-off Analysis
- Systems Projects, Validation and Verification

SE Specialty Courses (9-credits)

- Human Factors in SE
- Systems Life Cycle Analysis and Risk Management
- System Quality and Robustness Analysis

Thesis (6-credits)



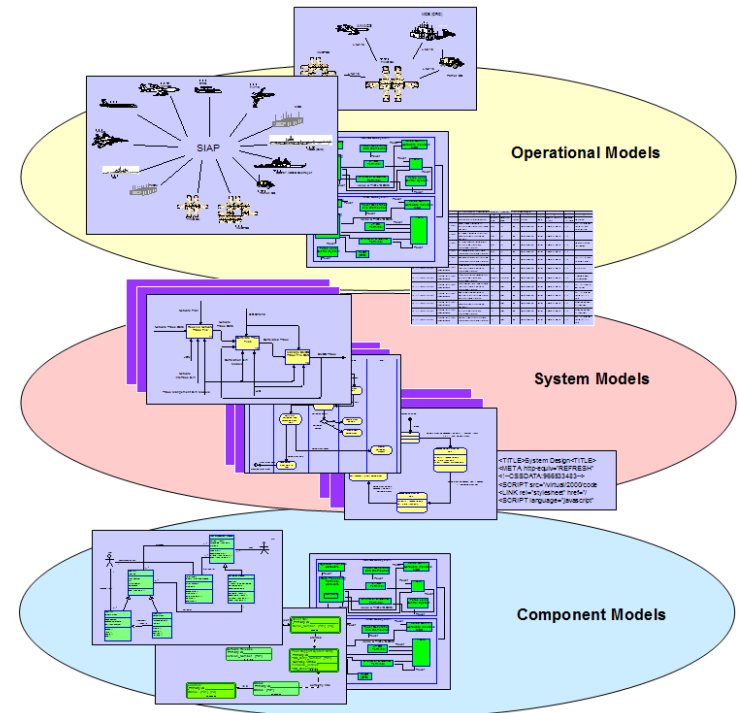
MODEL-BASED SYSTEMS ENGINEERING

Definition and Scope

- Formalizes the development of systems through the use of models.
- Broad in scope, across multiple stages of system development and multiple physics.

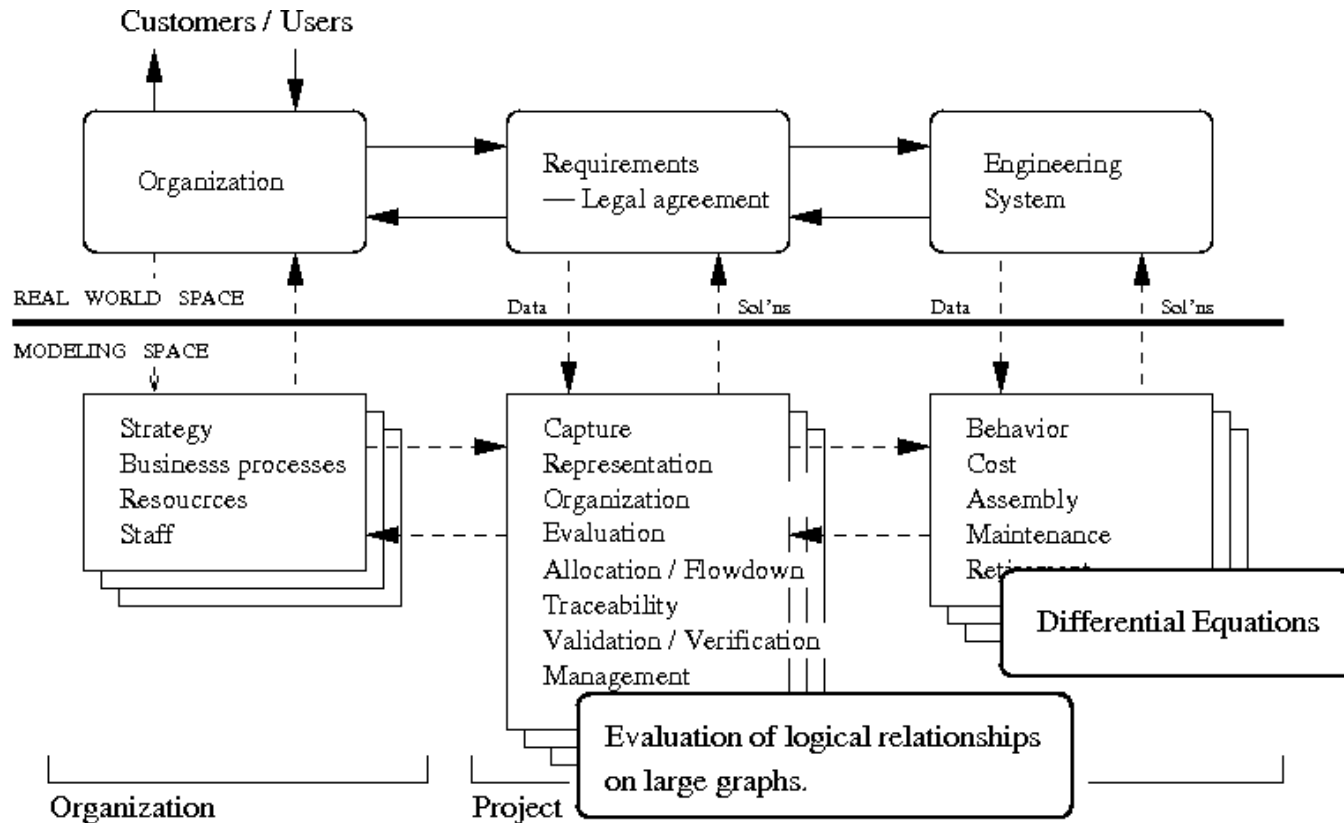
Benefits of MBSE

- Allows for the development of virtual prototypes.
- Facilitates communication among disciplines in team-based development.
- Enables semi-formal and formal approaches to system assessment.
- Management of system complexity.



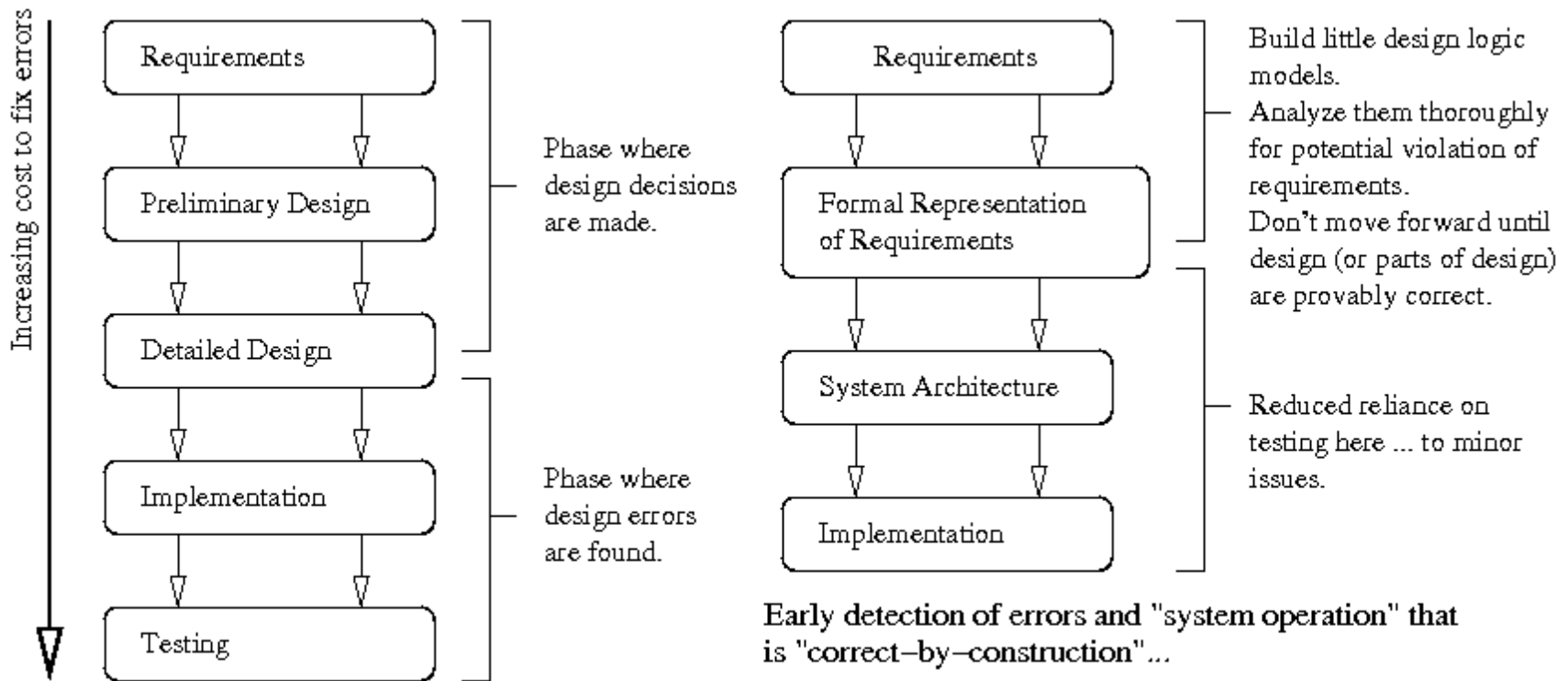
MODEL-BASED SYSTEMS ENGINEERING

Tenet 1: Create Big-Picture View and Emphasize Model-Based Systems Engineering.
The mathematics needed for formal approaches to systems engineering is foreign to many engineers.



MODEL-BASED SYSTEMS ENGINEERING

Tenet 2: Emphasize Disciplined Approaches to Design. Techniques include decomposition, abstraction, and formal analysis.

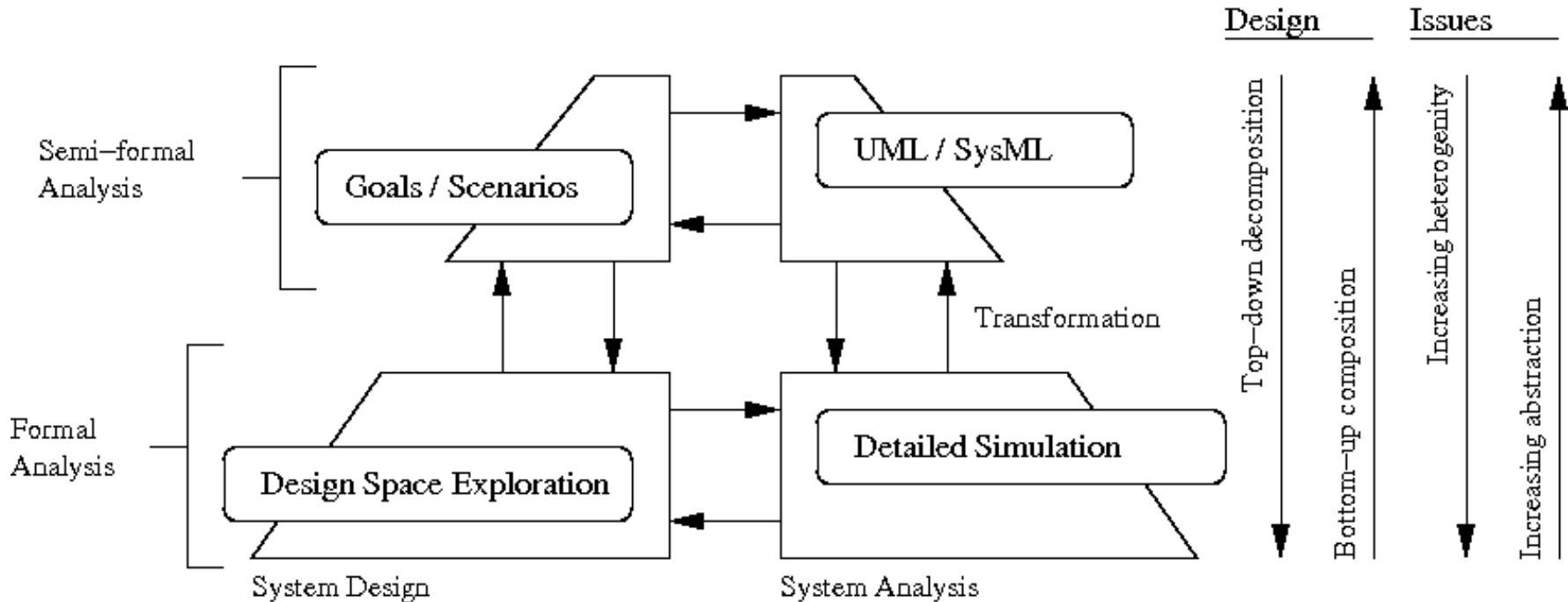


Traditional Approach to Design and Test...



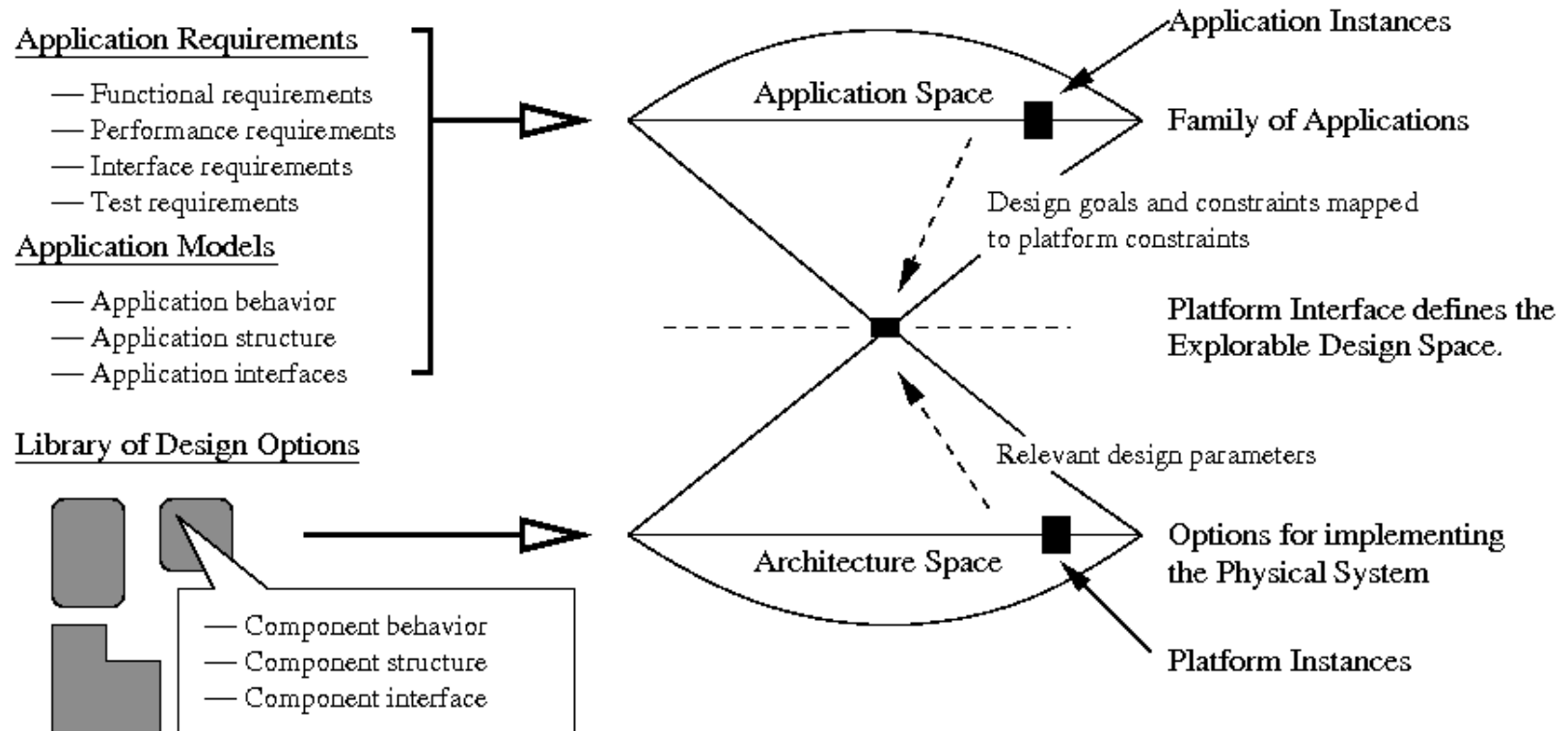
MODEL-BASED SYSTEMS ENGINEERING

Tenet 3: To keep the complexity of design activities in check, we need to employ mixtures of semi-formal and formal approaches to system development.



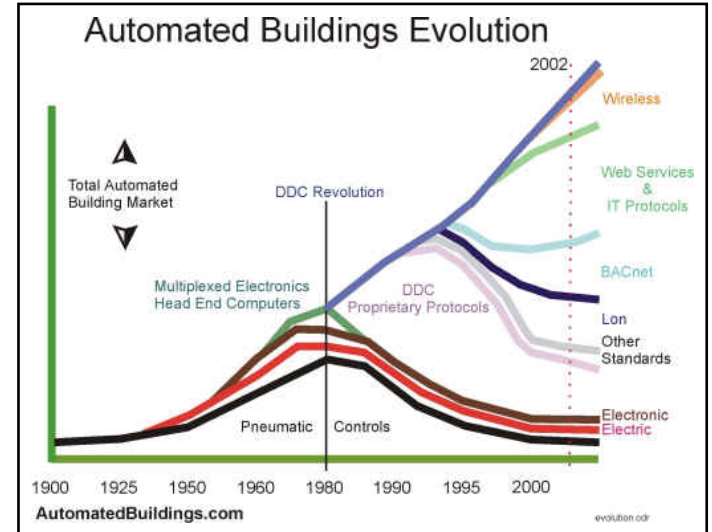
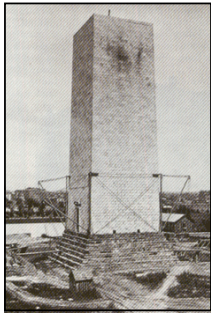
MODEL-BASED SYSTEMS ENGINEERING

Tenet 4: Use platform abstractions for system-level design.

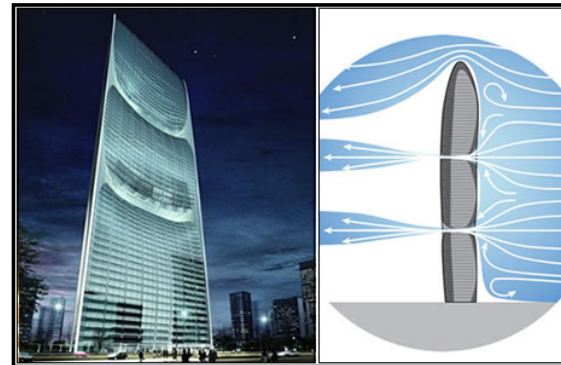


MODEL-BASED SYSTEMS ENGINEERING

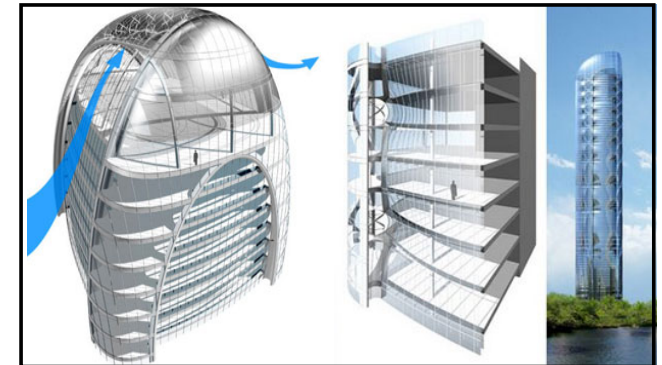
Motivating Application Area 1: Buildings!



Pearl River Tower Complex



Green Technology Tower — Architectural Proposal for Chicago



MODEL-BASED SYSTEMS ENGINEERING

Design Platform Stack



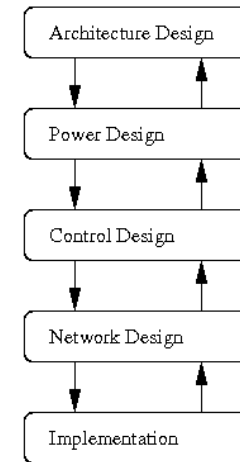
Research: Design of a scalable and extensible platform infrastructure



Factors Driving Design

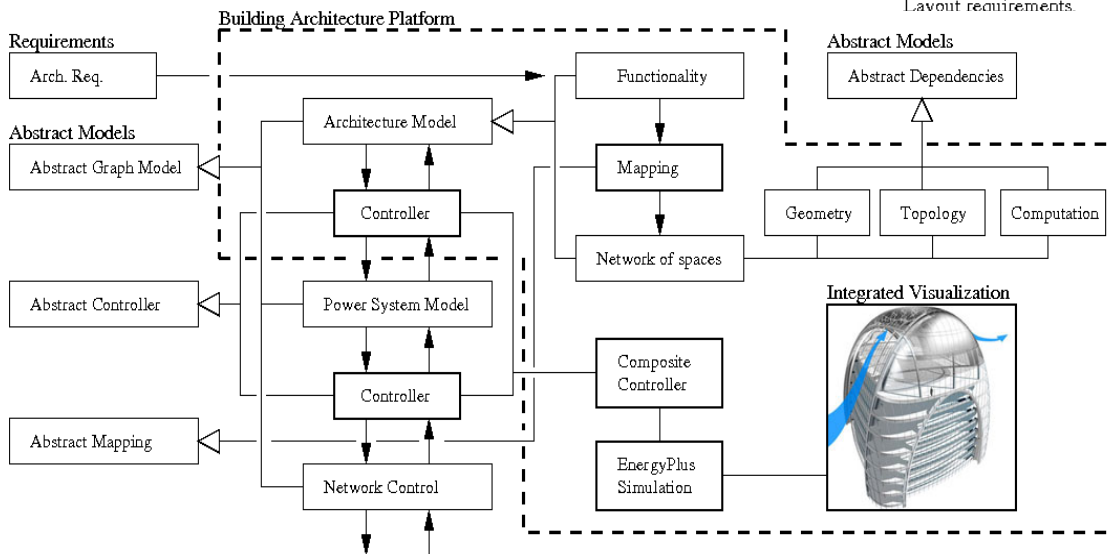
- Architectural requirements.
- Occupancy requirements.
- External loads (gravity, thermal, ...)
- Ventilation requirements.
- Energy generation requirements.
- Sequence of operations.
- Comfort requirements.
- Control speed requirements.
- Sensor and actuator requirements.

Design Flow



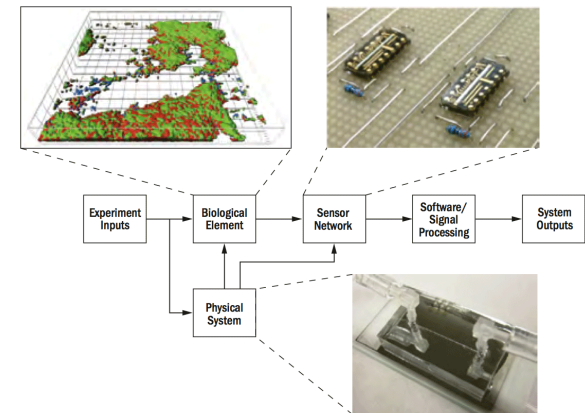
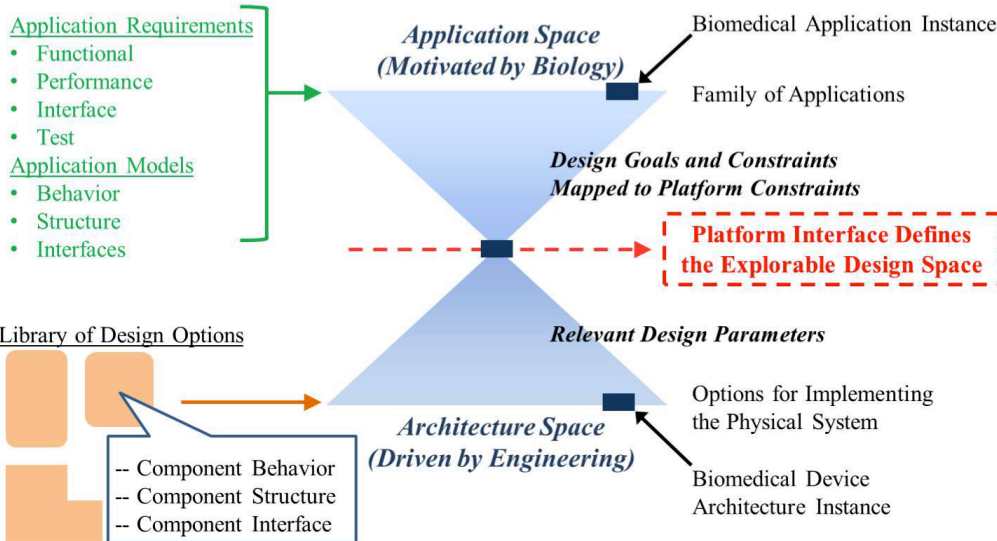
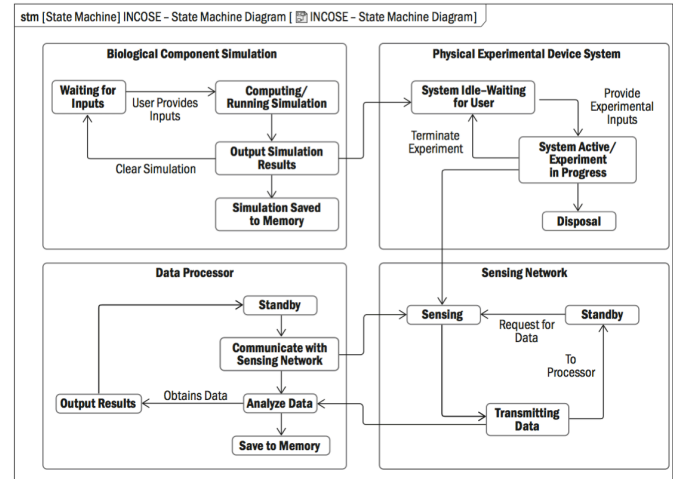
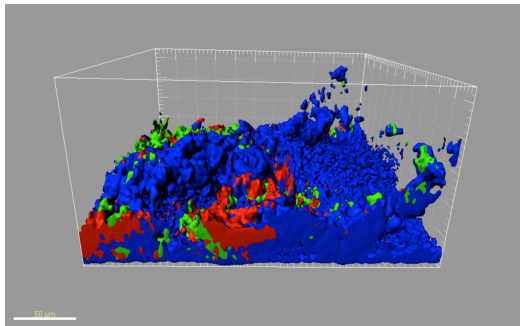
Performance

- Maximum ventilation.
- Maximum power generation.
- Cost estimates.
- Minimum response time.
- Control accuracy.
- Maximum available bandwidth.
- Maximum computational speed.
- Maximum storage size.
- Actual ventilation.
- Actual power generation.
- Actual network speed.
- Actual layout constraints.
- Actual installation cost.

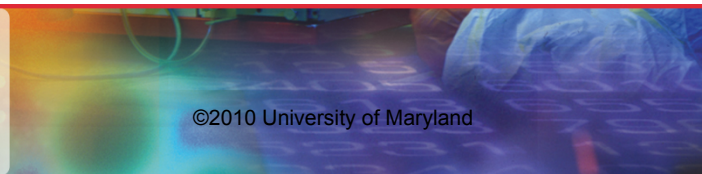


MODEL-BASED SYSTEMS ENGINEERING

Motivating Application Area 2: Platforms for Biomedical Experimental Research



Source: Mosteller et al., 2012



PART 1

MODEL-BASED SYSTEMS ENGINEERING OF CYBER-PHYSICAL SYSTEMS AT MARYLAND



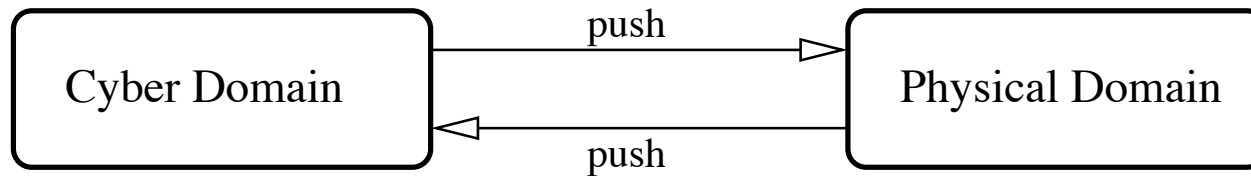
The
Institute for
Systems
Research

THE A. JAMES CLARK SCHOOL of ENGINEERING
UNIVERSITY OF MARYLAND

©2010 University of Maryland

CYBER-PHYSICAL SYSTEMS

Our Definition of CPS



C-P Structure

Cyber capability in every physical component.

Executable code

Networks of computation

Heterogeneous implementations

Spatial and network abstractions

-- physical spaces

-- physical and social networks.

-- networks of networks

Sensors and actuators.

C-P Behavior

Dominated by logic

Control, communications

Stringent requirements on timing

Needs to be fault tolerant

Physics from multiple domains.

Combined logic and differential equations.

Not entirely predictable.

Multiple spatial- and temporal- resolutions.



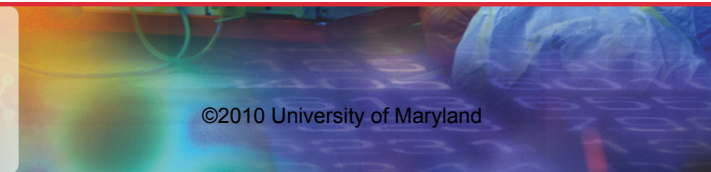
PATHWAY TO RESEARCH IN CPS

Scientific and Technical Challenges

- **Modeling:** Lack of an **integration science** with needed mathematical foundations
- **Design:** Weak procedures for handling meta domains (**time, space, ..**).
- **Operation and Decision:** High dependence on embedded and local computational intelligence

Research Opportunities

- Design cyber that can reason with **physical quantities** (not just numbers), **time and space**.
- Design **component hierarchies** and **networks**, and component ports that work with physical quantities.
- Embed physical quantities, **ontologies**, and **reasoning capability** deeply into **scripting languages**. Script and solve practical applications.



ENCE 688R: CIVIL INFORMATION SYSTEMS

Pilot Course: Understand data structures and algorithms for modeling and analysis of **networked infrastructure systems**. Focus on object-oriented solutions and use of **software design patterns**.

Hands-On: Software development in Java, Python, Jython and XML.

Mechanism: Mixtures of Civil Engineering and Systems Engineering graduate students.

Class projects:

- Component based modeling of networked systems.
- Transportation route selection.
- Network-based modeling of cities.
- Dam modeling and visualization.
- European Gas Network Modeling.
- Ontology modeling and rule-based reasoning.

COMING IN SPRING 2014 !

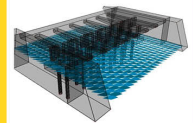
ENCE 688R

ADVANCED TOPICS IN CIVIL ENGINEERING

CIVIL INFORMATION SYSTEMS

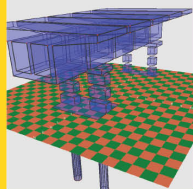
WOULD YOU LIKE TO UNDERSTAND:

- How to develop engineering software in Java and Python?
- How to develop graphical user interfaces in Java?
- How to use Java and Python with XML?
- How to model the structure and behavior of civil systems?
- Data structures and algorithms for the modeling and analysis of networked infrastructure systems?



GOALS

This course will be a hands-on introduction to engineering software development for the model-based design and operational management of modern civil systems. Students will learn how to model the structure and behavior of civil systems, and then develop object-oriented software solutions for specific civil systems applications. Motivating case studies will be drawn from road, rail, and utility networks, networked building services, and spatial modeling for buildings and urban areas.



INSTRUCTOR Professor Mark A. Austin
LECTURE Tuesday and Thursday, 5:00-6:15 p.m. 1104 EGR
CLASS LIMIT 30 students **3 CREDITS**

LEARN MORE ONLINE!

ter.ps/ence688r



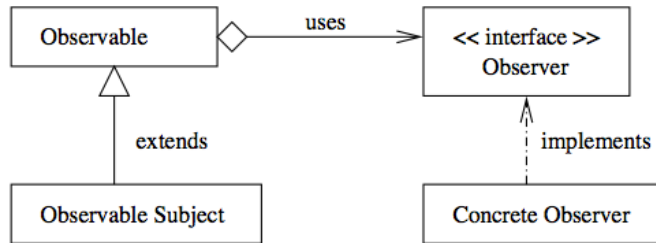
The
Institute for
Systems
Research

THE A. JAMES CLARK SCHOOL of ENGINEERING
UNIVERSITY OF MARYLAND

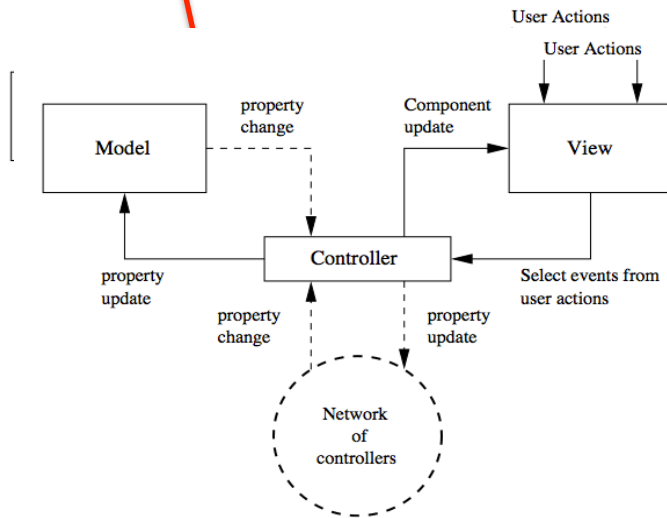
©2010 University of Maryland

SOFTWARE DESIGN PATTERNS (NETWORKS)

Observer Pattern

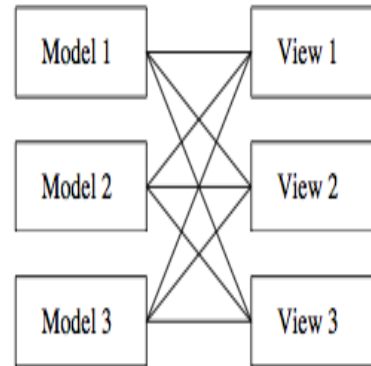


Hybrid MVC Pattern

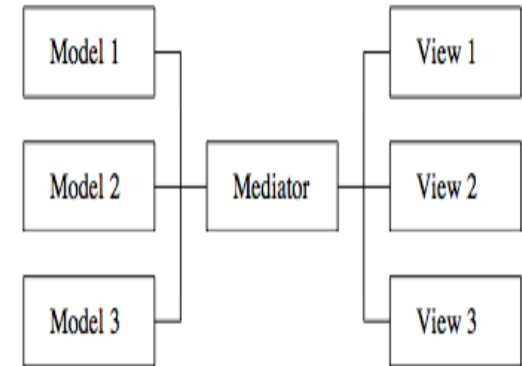


Mediator

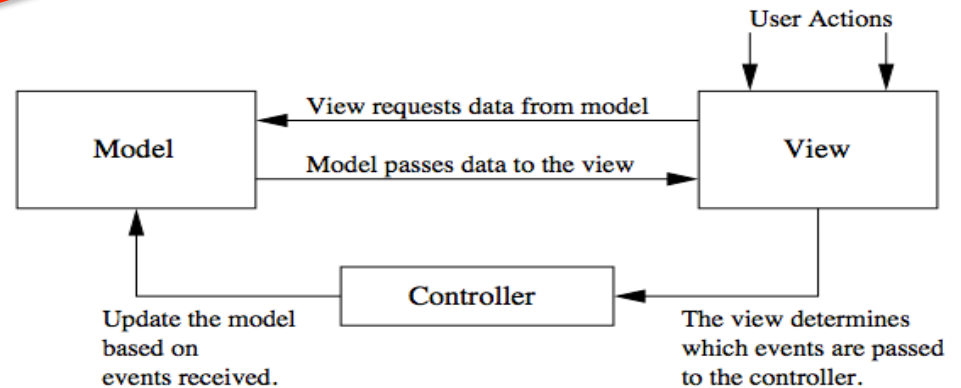
Point-to-Point Connection



Mediator Design Pattern

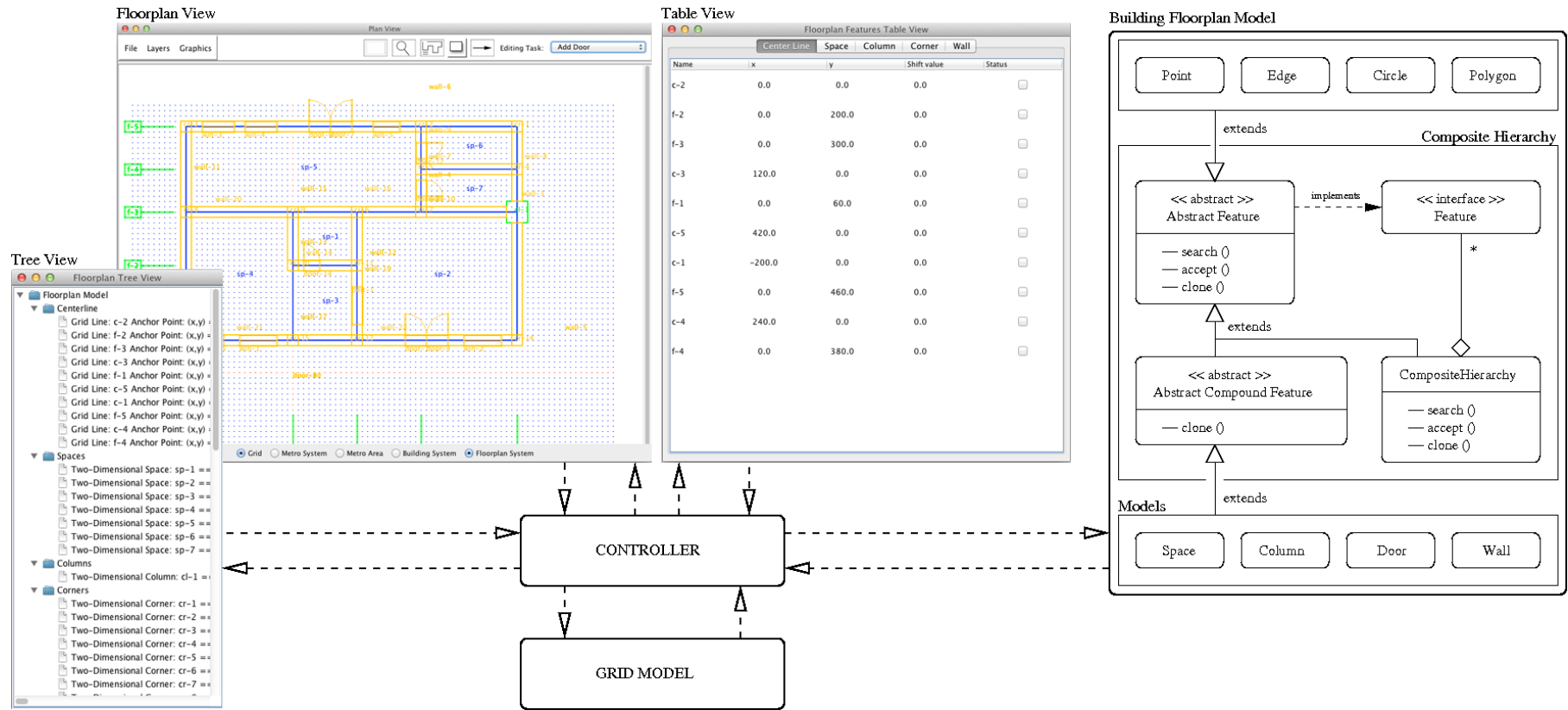


Model-View-Controller (MVC)



ENCE 688R / MSSE PROJECT WORK

BUILDING FLOORPLAN EDITOR

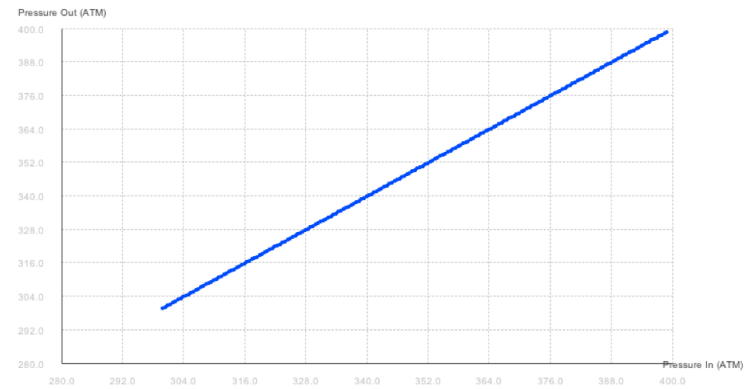
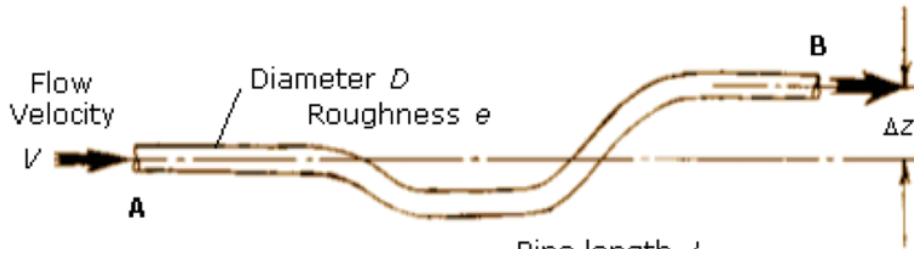
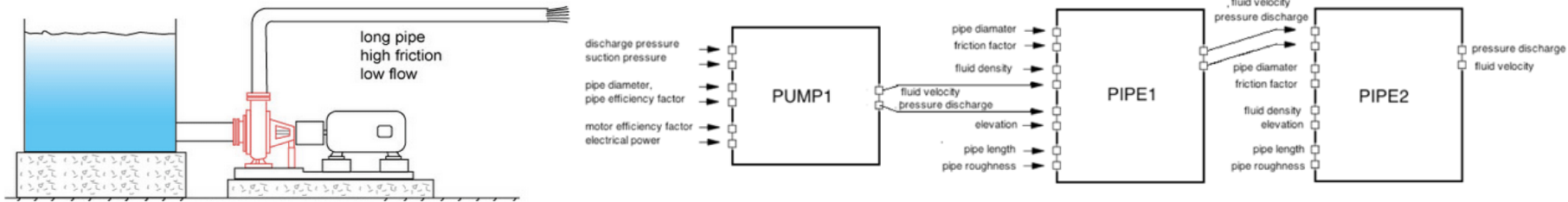


Source: Eddie Tseng, MSSE Graduate Student.



ENCE 688R / MSSE PROJECT WORK

COMPONENT-BASED MODELING OF PIPE NETWORKS



$$P_b[\text{Pa}] = P_a[\text{Pa}] - \rho[\text{kg}/\text{m}^3] * g[\text{m}/\text{s}^2] * \left(\Delta z[\text{m}] + f[\text{dimensionless}] \frac{L[\text{m}]}{D[\text{m}]} \frac{V^2[\text{m}^2/\text{s}^2]}{2g[\text{m}/\text{s}^2]} \right)$$

Source: Karam Rajab, MSSE Graduate Student.



ENCE 688R / MSSE PROJECT WORK

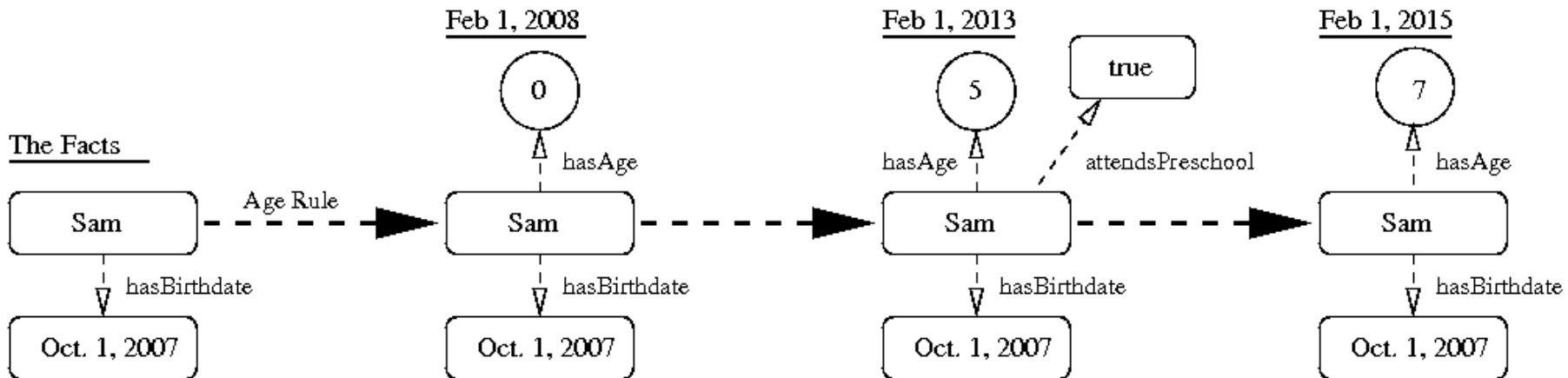
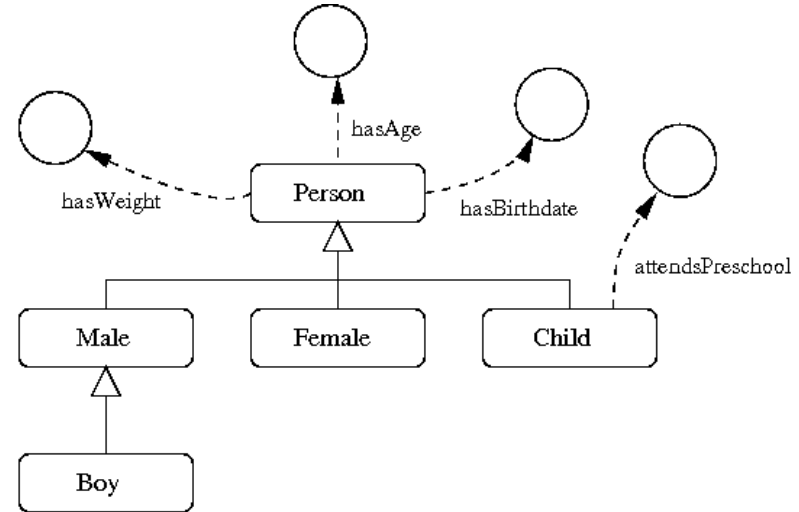
ONTOLOGY MODELING AND RULE-BASED REASONING

Fact. Sam is a boy. He was born October 1, 2007.

Rule 1: For a given date of birth, a built-in function getAge() computes a person's age.

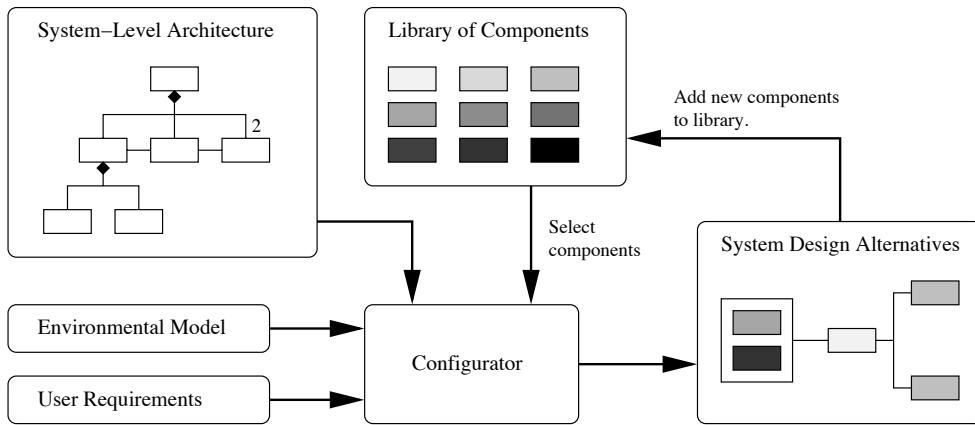
Rule 2: A child is a person with age < 18.

Rule 3: Children who are age 5 attend preschool.



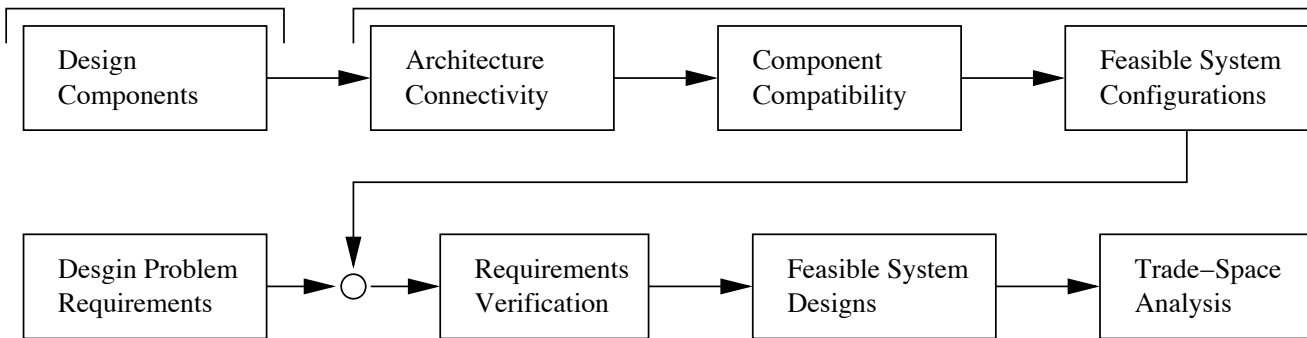
ENCE 688R / MSSE PROJECT WORK

DESIGN AND TRADE-OFF ANALYSIS WITH RDF GRAPHS

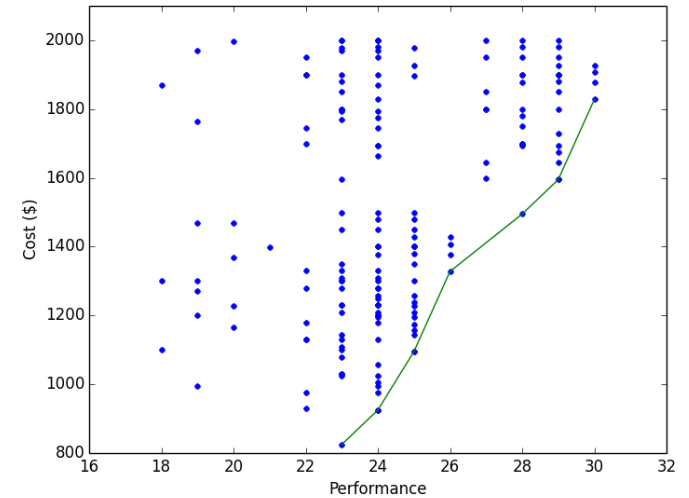


Problem Definition
RDF Graph Models

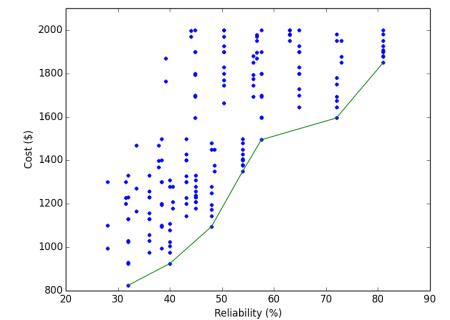
Inference-Rule Driven Graph Transformations



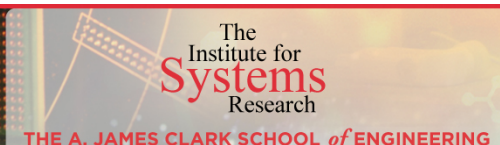
Trade-off Analysis: Min Cost versus Max Performance



Trade-off Analysis: Min Cost versus Max Reliability



Source: Nefretiti Nassar, MSSE Graduate Student.



CURRENT RESEARCH ON CPS

Semantic Platforms for Design and Management of Trains and Buildings

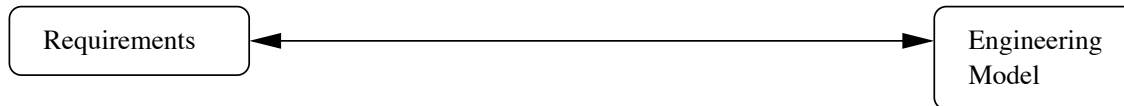
Civil Systems Ph.D. Student: Parastoo Delgoshaei (2010-present)



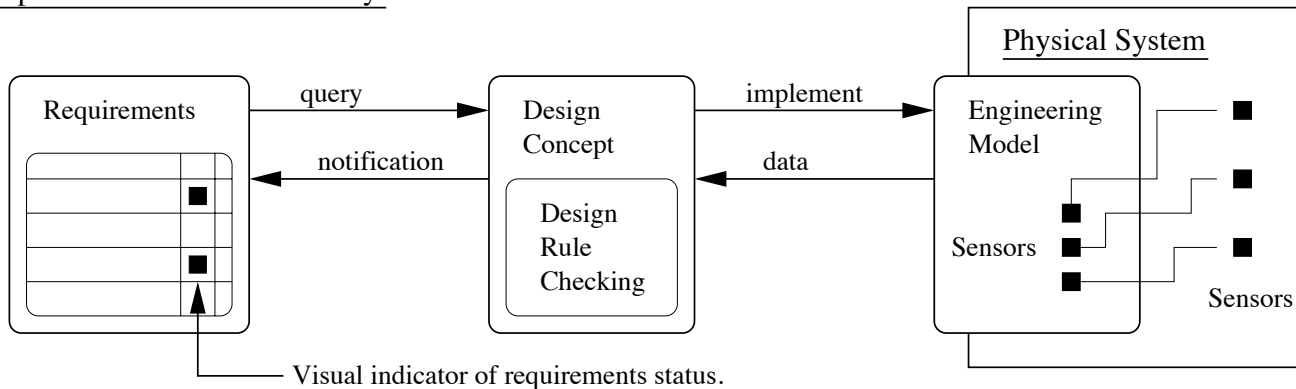
TRACEABILITY MECHANISMS

New idea (2005): Ontology-enabled Traceability Mechanisms

State-of-the-Art Traceability



Proposed Model for Traceability



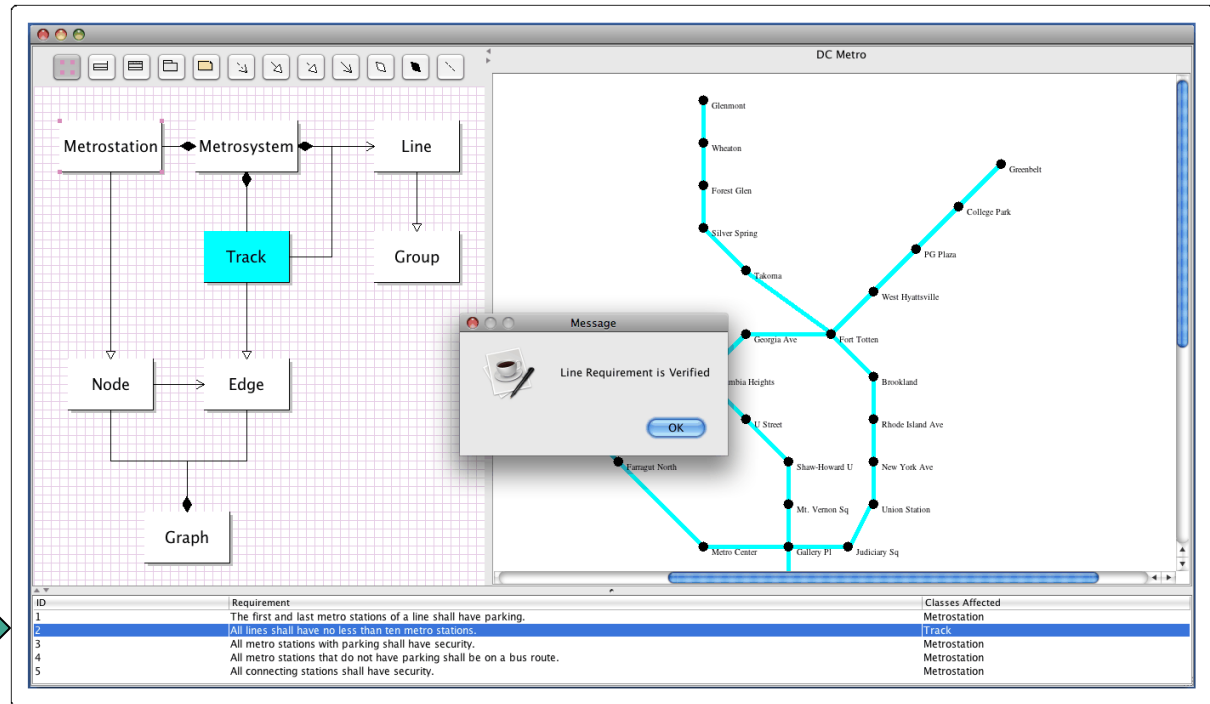
Approach: Requirements are satisfied through implementation of design concepts. Now traceability pathways are threaded through design concepts.

Key Benefit: Rule checking can be attached to “design concepts” (ontology), therefore, we have a pathway for early validation.



ONTOLOGY – ENABLED TRACEABILITY (WITH VERY BASIC RULE CHECKING)

Key Advantage: Design rules and procedures for design rule checking can be attached to ontologies

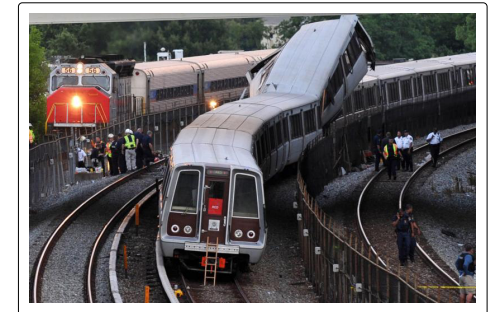
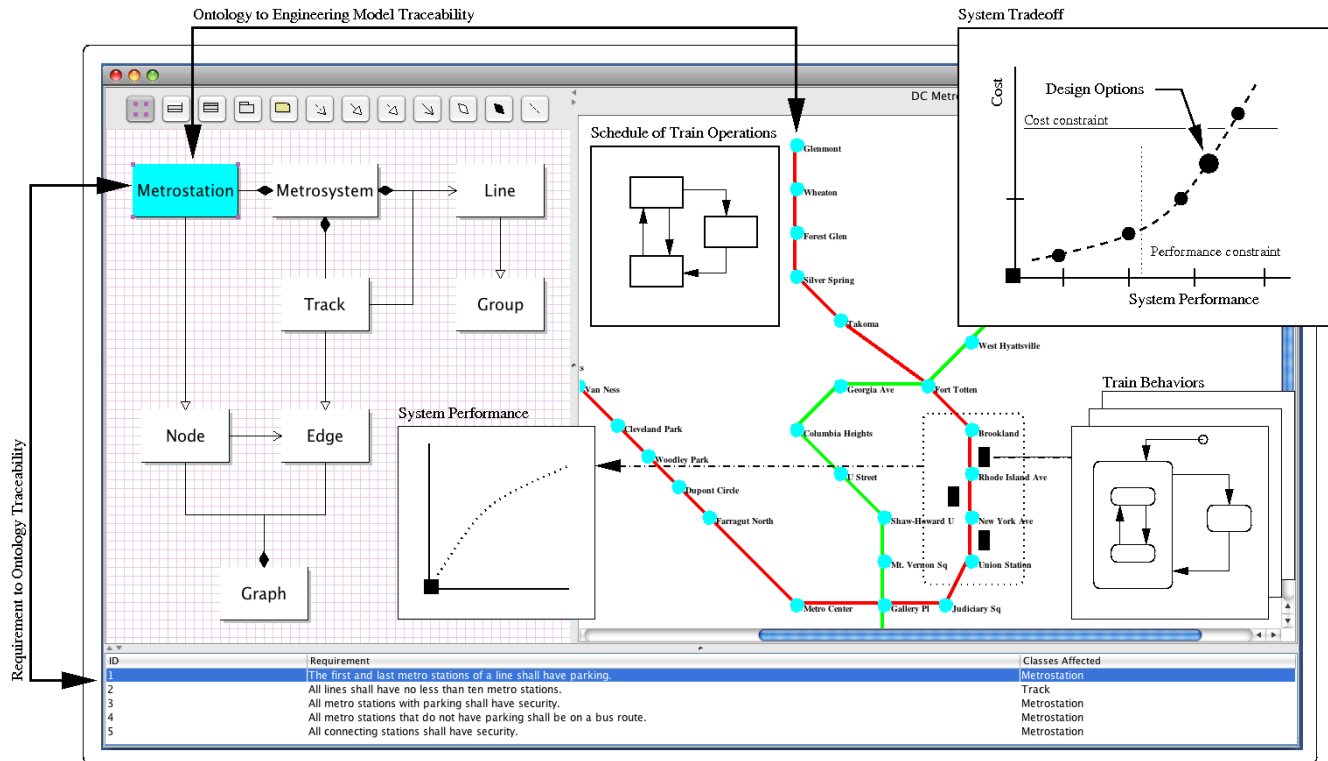


Design rule checking is triggered by double clicking on a requirement. Visualization shows the extent of ontologies and engineering entities involved in the rule checking.

Source: Cari Wojcik, 2006.



VISION FOR VERSION II (2010)



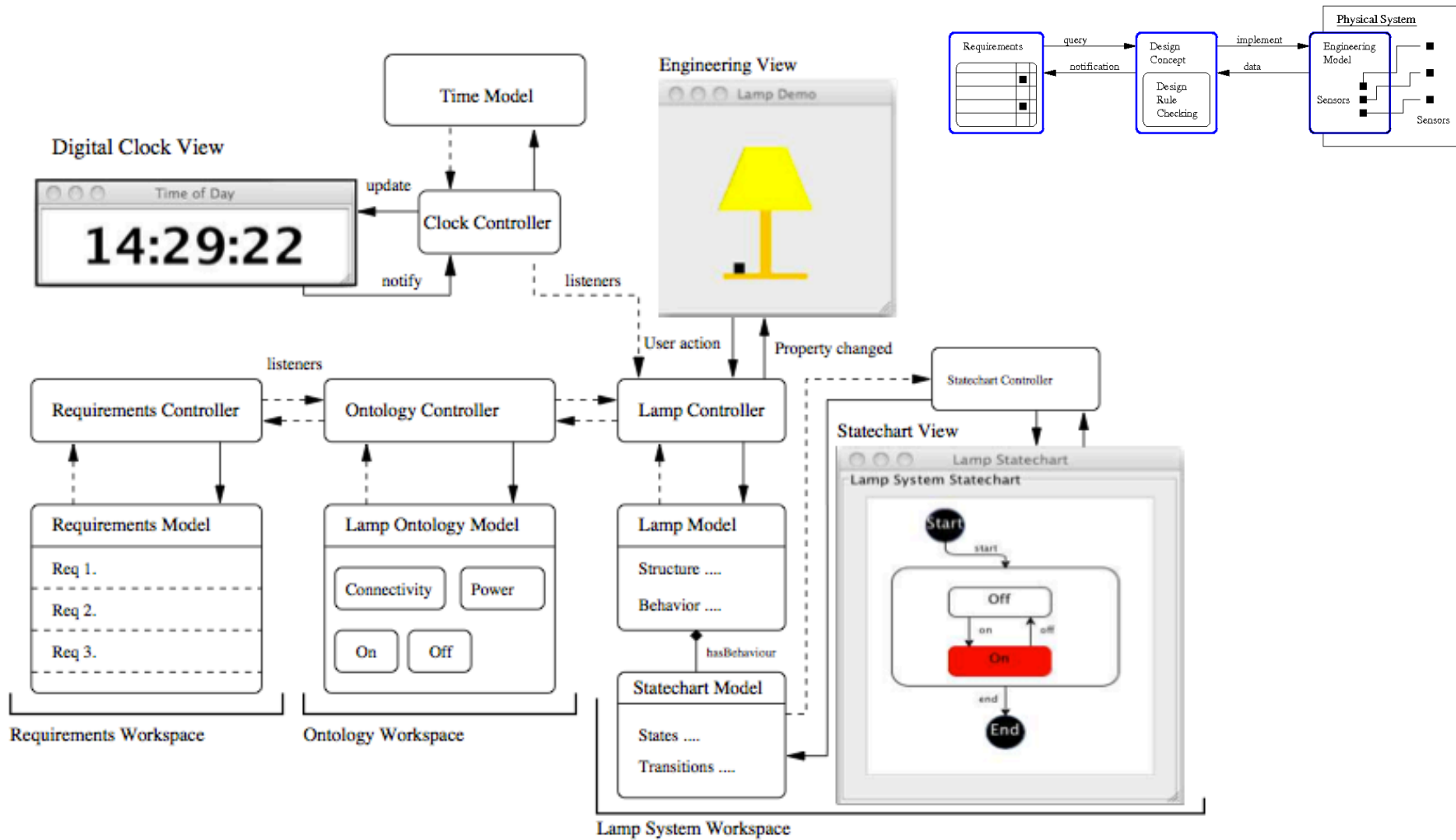
June, 2009.

Re-design implementation to maximize use of software design patterns.
 Model system schedules and train behavior with finite state machines.

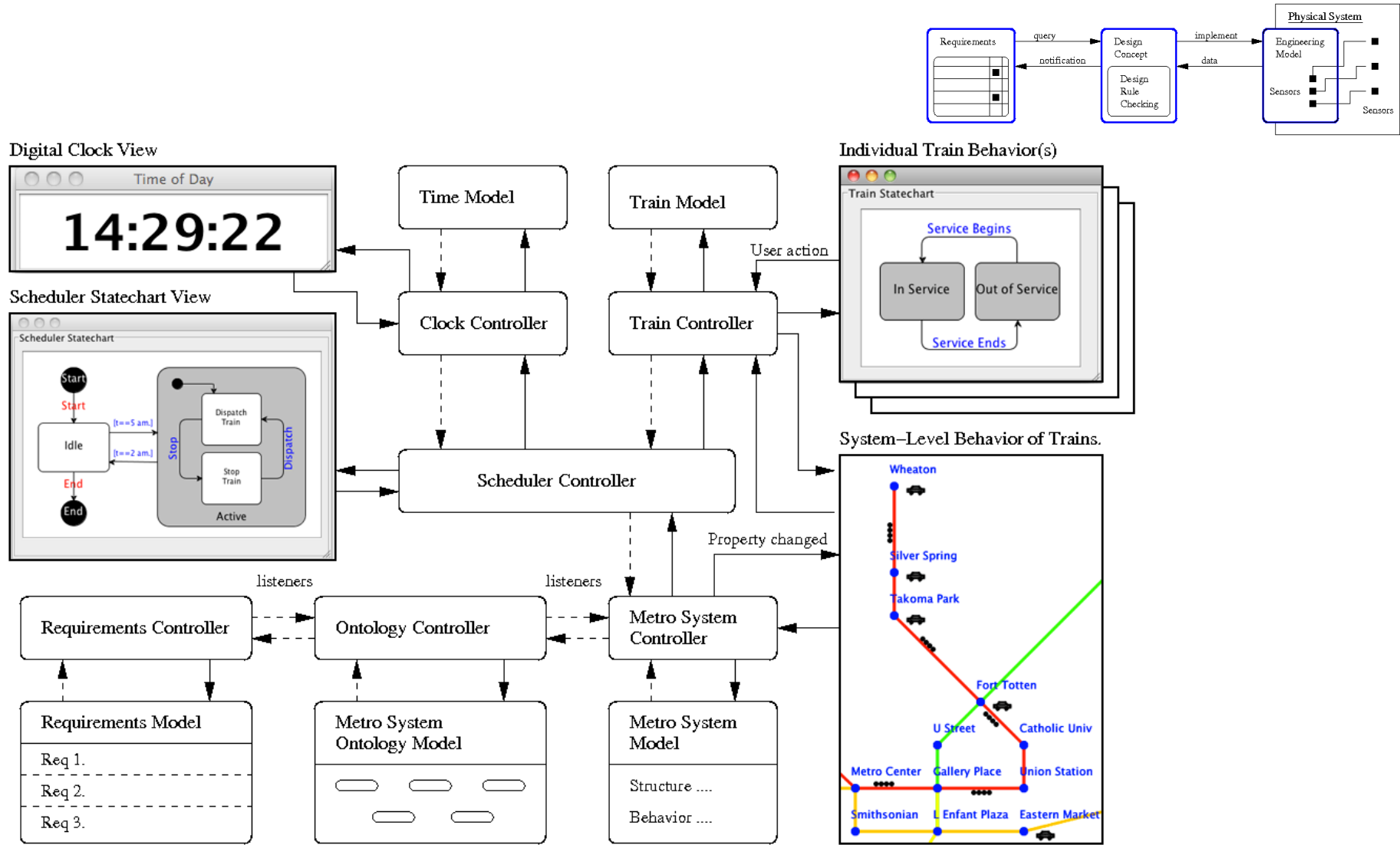
System-level behavior will correspond to a network of communicating finite state machines.



SIMPLE LAMP ARCHITECTURE



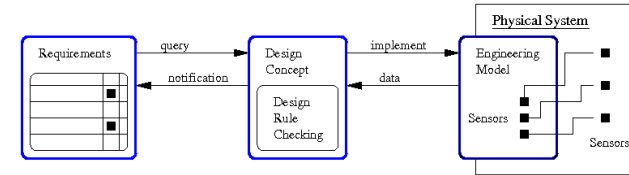
RAILWAY SYSTEM ARCHITECTURE



REQUIREMENTS-TO-STATECHART TRACEABILITY

Requirement level (textual representation)

The metro system will start working at 5 am.



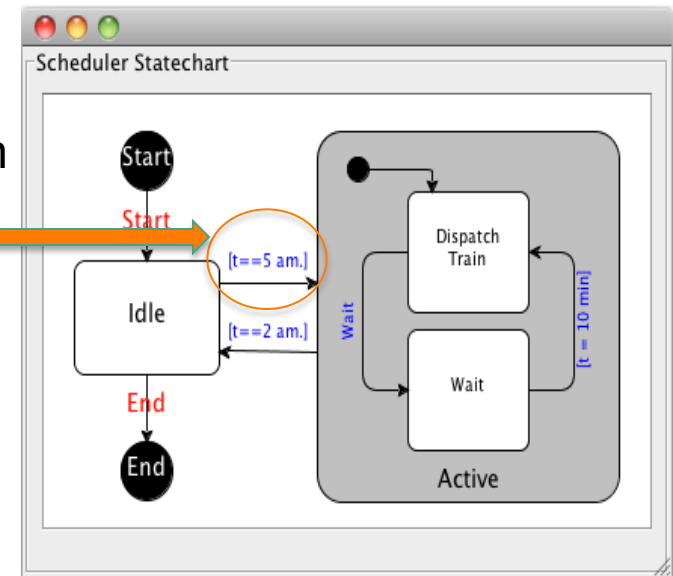
Rule level (SWRL)

```

scheduler(?s)^ hasTime(?s,?t) ^ swrlb:greaterThan(?t,5) ^ train(?tr)
^ isAvailable(?tr,true)=>sendTrain(?s,?tr)
    
```

Guard Statement

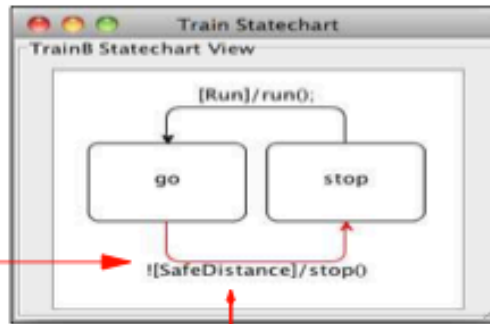
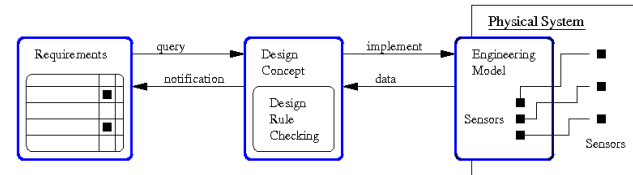
The transition from idle to active is conditional on “ [t == 5 am.]” evaluation results.



Expected Behavior

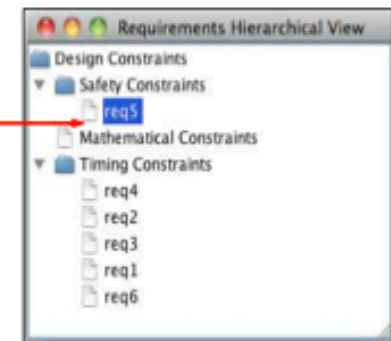
- The scheduler statechart will transition from idle to active at 5:00 am.
- The statechart of at least one train will transition to the “At Station” state.

STATECHART TO REQUIREMENT TRACEABILITY



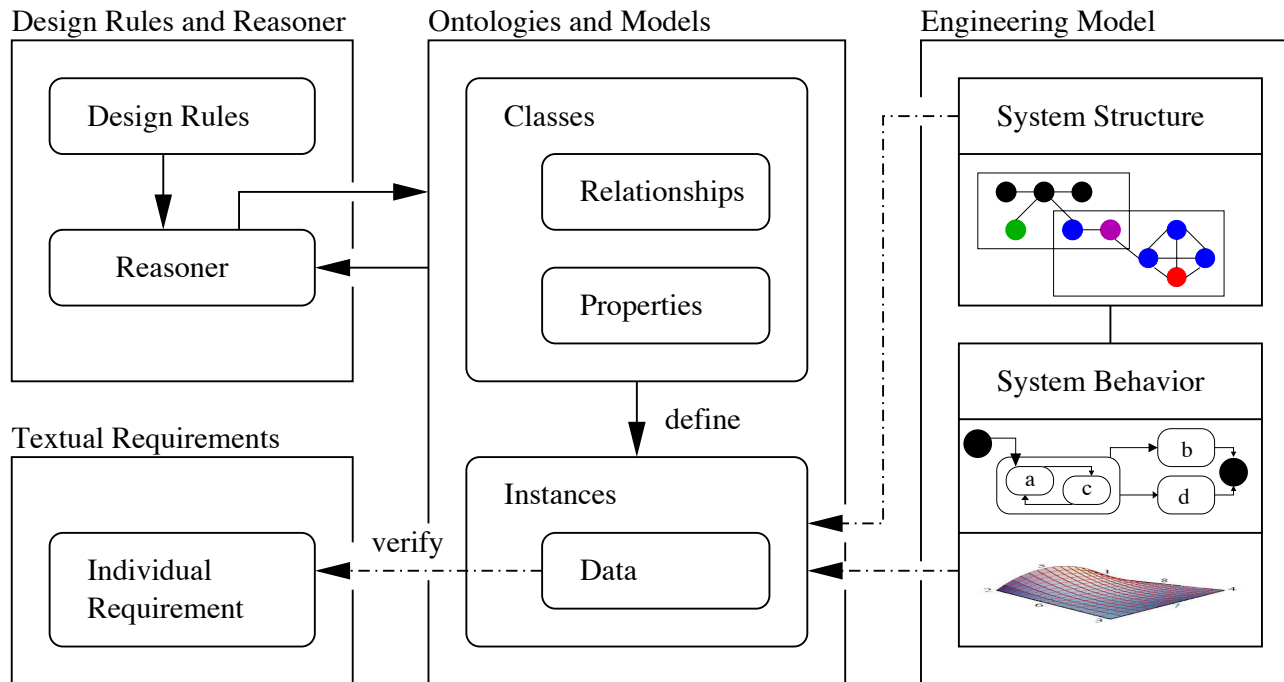
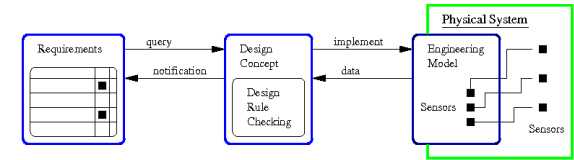
Washington DC Metro System (Table View)

Requirement ID	Description	Priority	Status
req4	During rush service trains run mo...	false	
req5	The distace between two trains s...	true	
req2	Metro will be closed at midnight S...	false	
req3	Metrorail will operate rush hour s...	false	
req1	Metro will be open at 5 a.m. wee...	false	
req6	During normal service hour trains...	false	



FROM TRAINS TO BUILDINGS (2013-2014)

Parastoo finishes MSSE Degree in Dec. 2012.
 Matriculates to Ph.D. in Civil Systems in Jan. 2013.

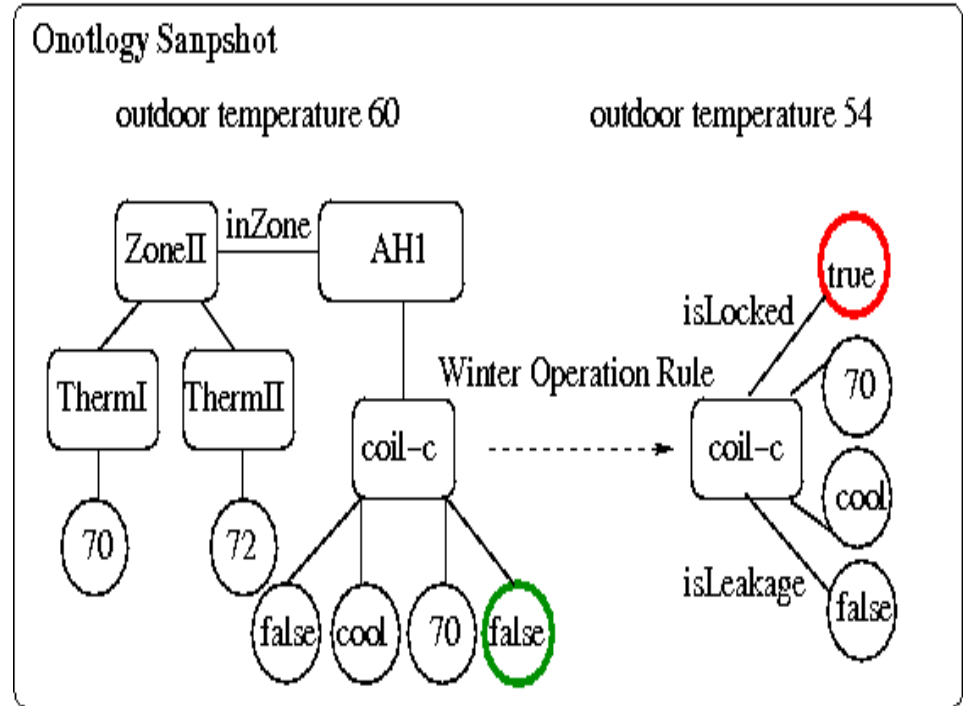
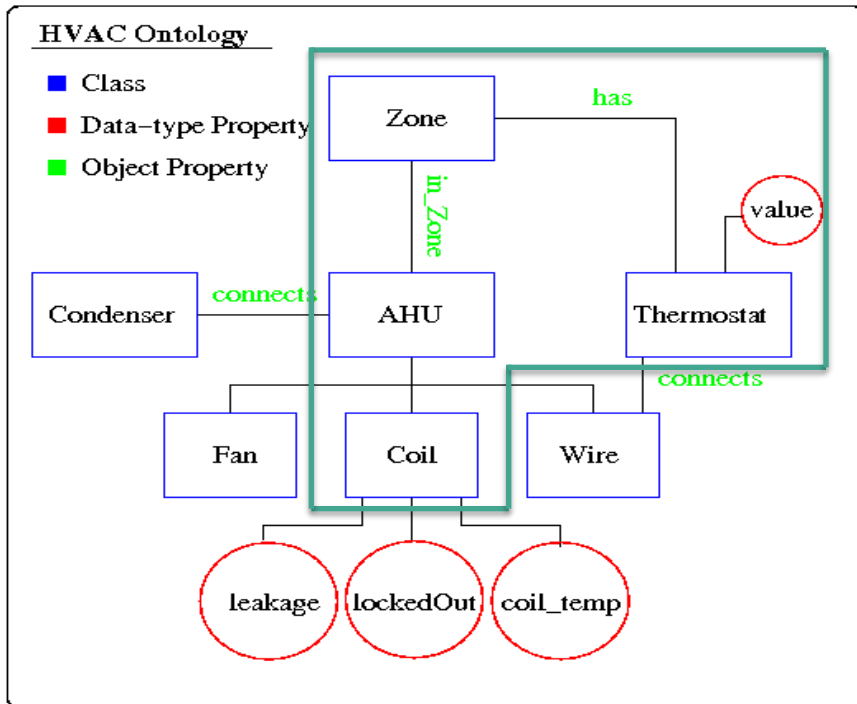


- Remarks
- System structures are modeled as networks and composite hierarchies of components.
 - Behaviors will be associated with components.
 - Discrete behavior will be modeled with finite state machines.
 - Continuous behavior will be represented by partial differential equations.

INFERENCE RULES FOR HVAC ONTOLOGY

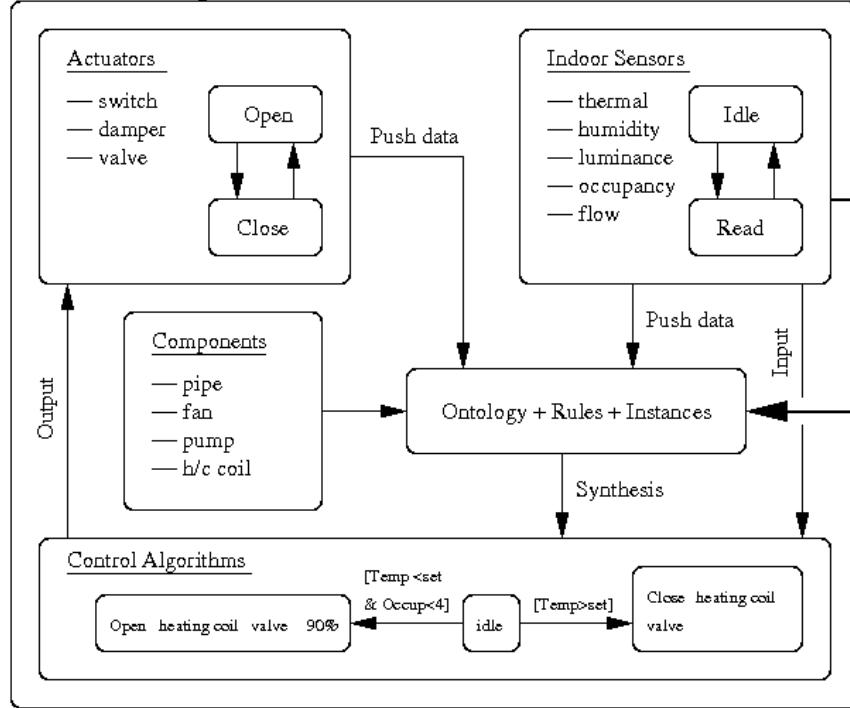
Requirement: Cooling coil will be locked out for winter operation (55 F)

Rule: (?cc RDF:type Cooling) (?cc ont:isLocked? ?I) (?out_temp ont:hasValue ?v) lessThan(?v,55) ->(?I, true)

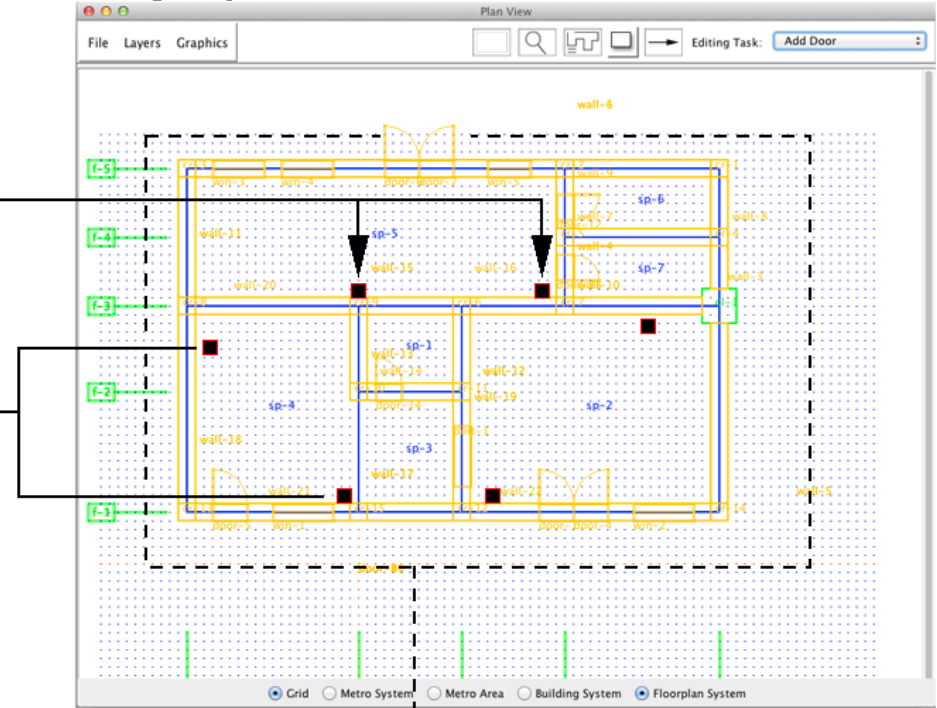


SYSTEM-LEVEL DESIGN / SIMULATION / CONTROL

Behavior Modeling and Control



Building Floorplan / Architecture



Automated synthesis of building simulations.

Building Simulation

- Time history simulation.
- Performance assessment.



SCRIPTING LANGUAGE DESIGN (2013)

Problem Statement and Approach

- Support the **integration of physical components** and **computation of discrete and continuous behaviors**.
- Units are embedded within the basic data types, physical quantities, matrices of physical quantities, and branching and looping control.

Assignment Statements

```
// Setup parameters for tank 02 ....

area02 = 5 m^2;
h02    = 1 m;

// Setup parameters for pipe connecting tanks 01 and 02 ....

pipeRadius    = 10 cm;
pipeArea      = PI*pipeRadius^2;
pipeLength    = 5 m;
pipeRoughness = 0.005;

// Setup parameters for fluid contained in the pipe and tanks ....

rho = 1000.0 kg/m^3; // density of water ...
g   = 9.81 m/sec^2;  // acceleration due to gravity ...
```

Looping Constructs

```
x = 0 cm;
while ( x <= 10 cm ) {
    print "*** x = ", x;
    if ( x <= 5 cm ) {
        x = x + 1 cm;
    } else {
        x = x + 2 cm;
    }
}
```

Matrices

```
Force    = [ 2 N, 3 N, 4 N ];
Distance = [ 1 m; 2 m; 3 m ];
Work     = Force*Distance;
```

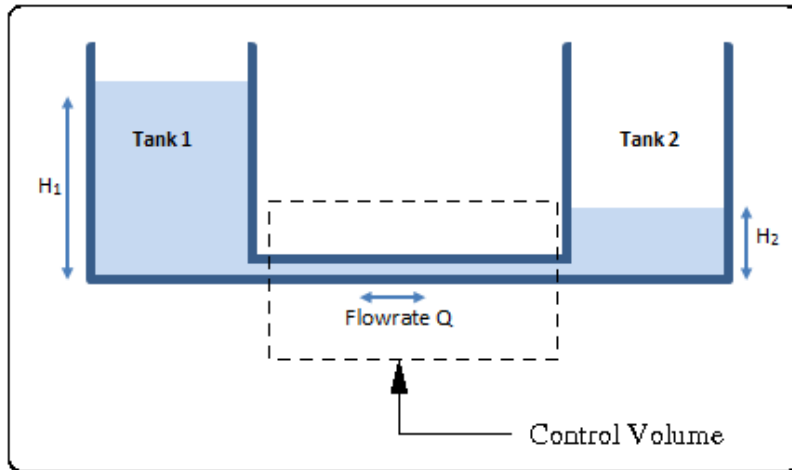
```
Matrix: Force
row/col      1          2          3
units        N          N          N
1            2.00000e+00 3.00000e+00 4.00000e+00
```

```
Matrix: Distance
row/col      1
units        m
1            1.00000e+00
2            2.00000e+00
3            3.00000e+00
```

```
Matrix: Work
row/col      1
units        Jou
1            2.00000e+01
```

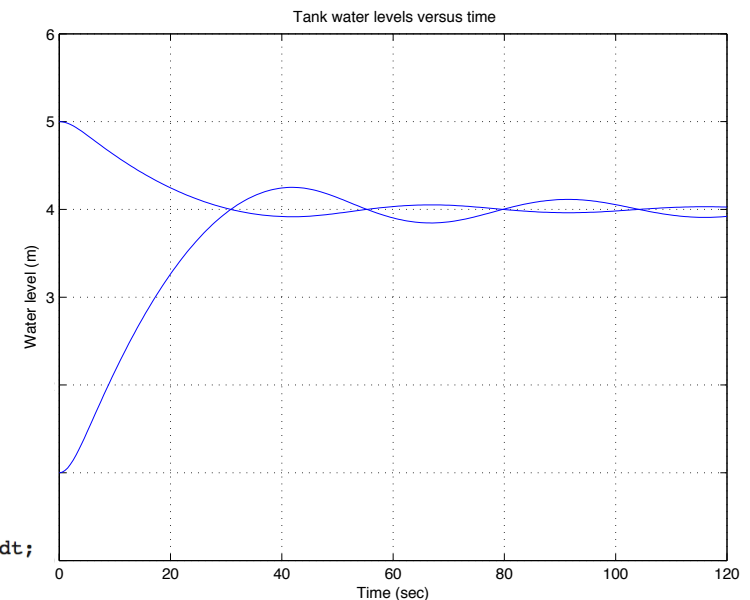


A SIMPLE PROBLEM – FLOW BETWEEN TANKS



```
[java]
[java] Matrix: response01
[java] row/col          1          2          3          4
[java]      units          sec          m          m          m^3/sec
[java] 1          0.00000e+00  5.00000e+00  1.00000e+00  0.00000e+00
[java] 2          5.00000e-01  4.99795e+00  1.00616e+00  1.23276e-01
[java] 3          1.00000e+00  4.99184e+00  1.02449e+00  2.43276e-01
[java] 4          1.50000e+00  4.98189e+00  1.05434e+00  3.53771e-01
```

... lines of output removed ...



$$\left[\frac{dv(t)}{dt} \right] + \left[\frac{f_1}{2D} \right] v(t)|v(t)| = \left[\frac{g}{L} \right] [H_1(t) - H_2(t)].$$

```
// Compute simulation response ...
```

```
for (i = 1; i < nsteps; i = i + 1) {
```

```
    // Compute fluid velocity update ...
```

```
    velocityFluid = pipeRoughness/(4.0*pipeRadius)*velocityOld*Abs(velocityOld)*dt;
    velocityUpdate = g/pipeLength*( h01Old - h02Old )*dt;
    velocityNew    = velocityOld + velocityUpdate - velocityFluid;
```

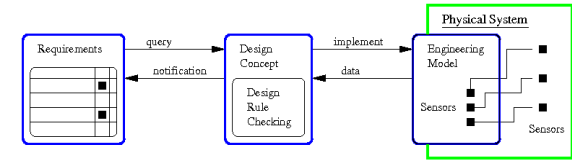
```
    // Update water depths in tanks 01 and 02 ...
```

```
    h01New = h01Old - (pipeArea/area01)*(velocityOld+velocityNew)*dt/2.0;
    h02New = h02Old + (pipeArea/area02)*(velocityOld+velocityNew)*dt/2.0;
```

Near-term goal: support for computational fluid dynamics.

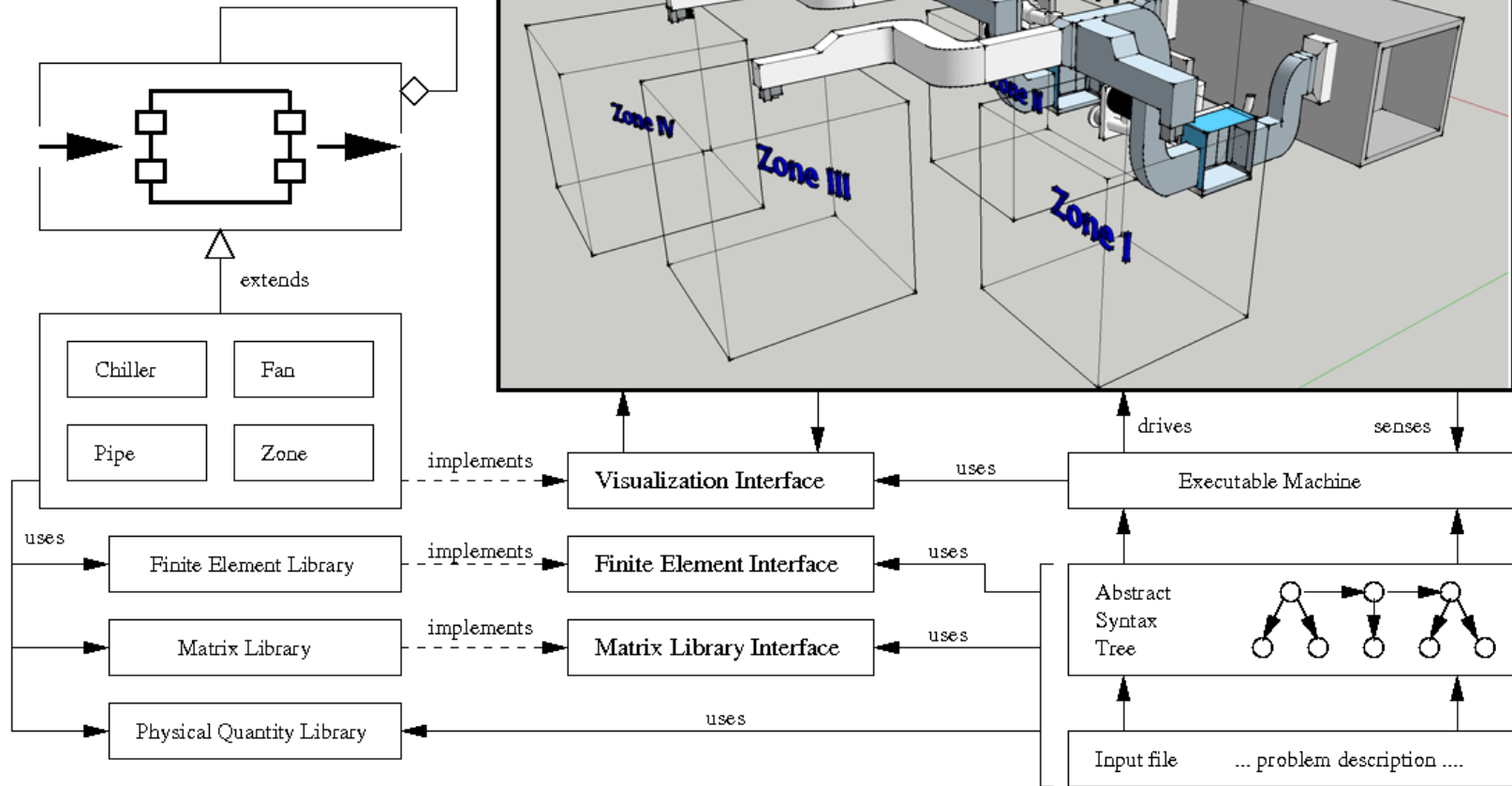


FUTURE PLANS



Simulation Framework and 3D Visualization

Component Modeling Framework



Acknowledgement: Amanda Pertzborn



CURRENT RESEARCH ON CPS

Ontologies of Time and Time-Based Reasoning.
Spatio-Temporal Systems.

Civil Systems Ph.D. Student: Leonard Petnga (2012-present)



The
Institute for
Systems
Research

THE A. JAMES CLARK SCHOOL of ENGINEERING
UNIVERSITY OF MARYLAND

©2010 University of Maryland

PROBLEM STATEMENT

What are Cyber-Physical Systems?

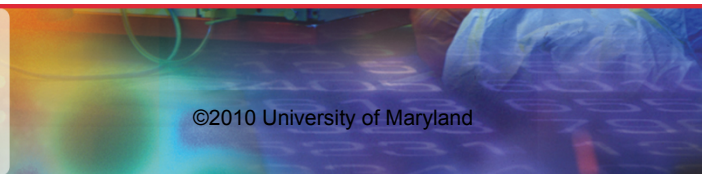
Systems with tight integration of *networked computational* and *physical* elements (Lee, 2010)



Mathematical model:	FSM	Graph	ODE
Time model:	Discrete	Discrete	Continuous
Space model:	Discrete	Discrete	Discrete/Continuous
Role:	Decision/Computation/Reasoning	Communication	Sensing/Actuation

Scientific and Technical Challenges :

- **Modeling:** Lack of “Integration science” with needed mathematical foundations
 - **Design:** Weak “procedures” for handling **meta domains** (time, space, ..) critical to system “ity”
 - **Operation/Decision :** High dependence on embedded/local computational intelligence
- ➔ **Right physical action at the right time and right place are critical to ... safety!**
- ➔ **How to embed physical semantics in cyber models for smartness**



OBJECTIVE AND SCOPE

Investigate and understand

- Meta domains especially **temporal** and **spatial theories**
- **Ontologies** roles in deriving formal, precise models and **architecting CPS**
- **DL-based semantics Reasoning** for achieving **System level safety** in CPS
- **Allen's Temporal Interval Calculus** and **Region Connection Calculus (RCC)**
- **Semantic web technologies**

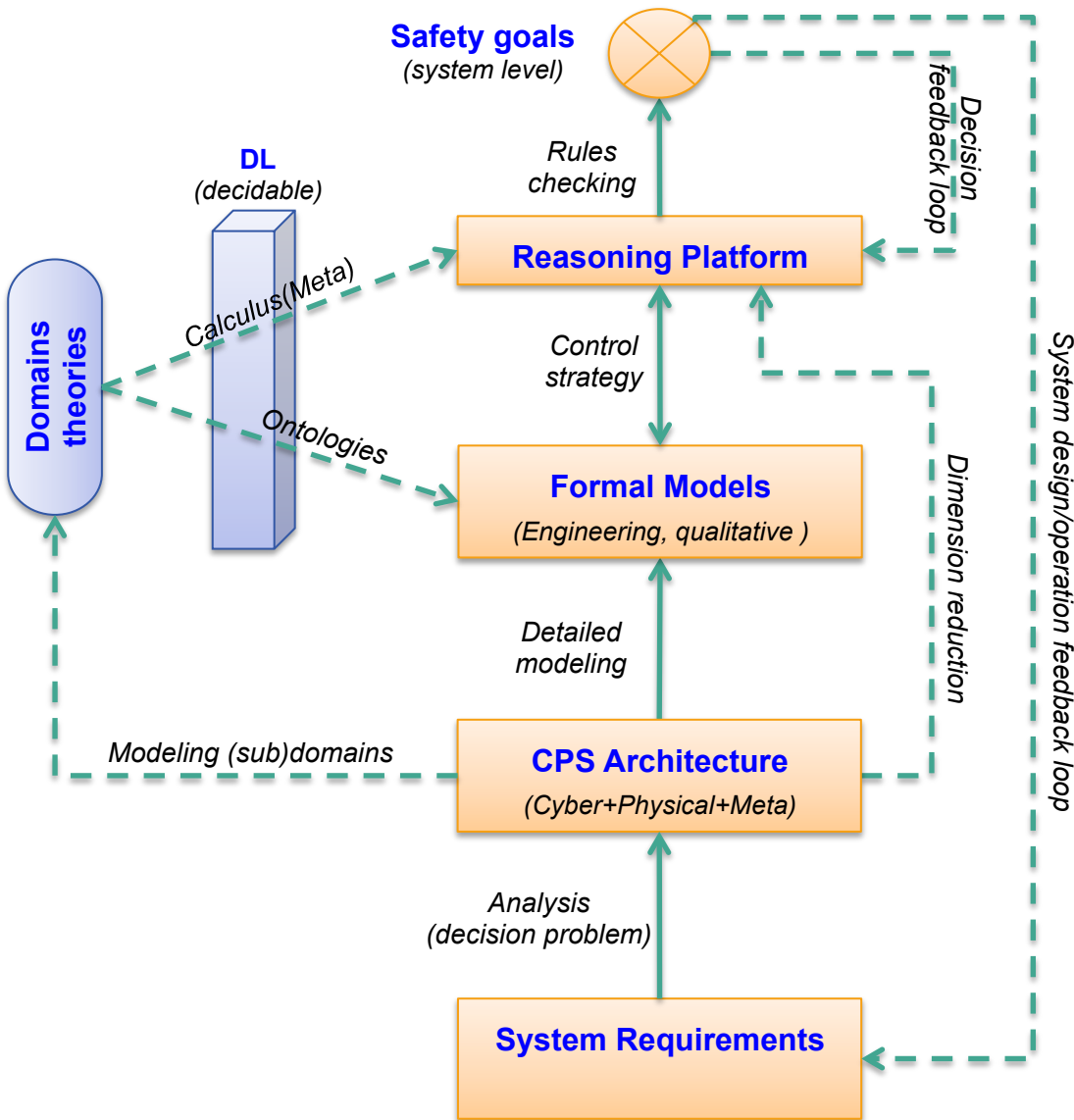
To conceptualize formal procedures and reasoning framework enabling HW-SW co-design, system-level safety study and domain-specific semantics in MBSE of CPS.

Implement: integration mechanisms between temporal and non-temporal domains in safety-critical systems in general and CPS in particular.

Applications: Civil Systems (traffic system, connected vehicles, automated aircraft taxing), robots (automated warehouse), aeronautic (UAV fleet), energy (wind farm).



OUR APPROACH: What's New?



- ✓ **Modular framework** : domains and rules (theories) formally defined; Ontologies integrity .
- ✓ **Reasoning-enabled** : DL- semantics for decision making; Handling of physical quantities.
- ✓ **Interface ontologies** : define and link primitive-domains.
- ✓ **Formal analysis of system's safety properties**: decision trees for safety requirements (hard constraints) checked;



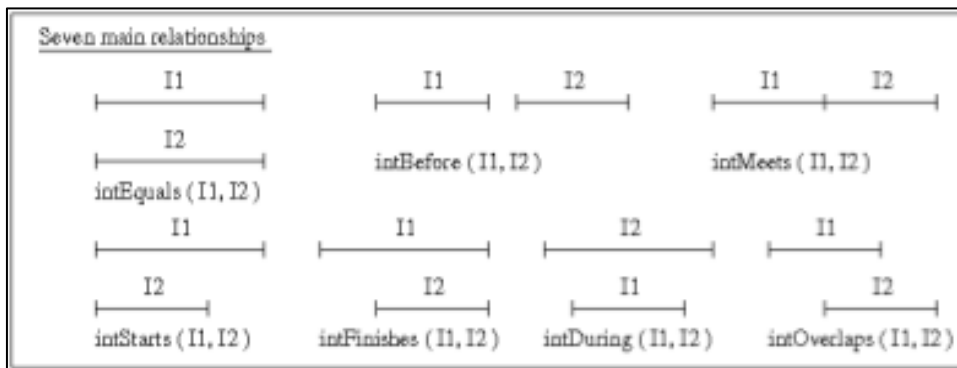
OUR APPROACH: **Scope**

**CPS for which safety and performance depend
on correct time/space-based prediction of the
future state of the system.**



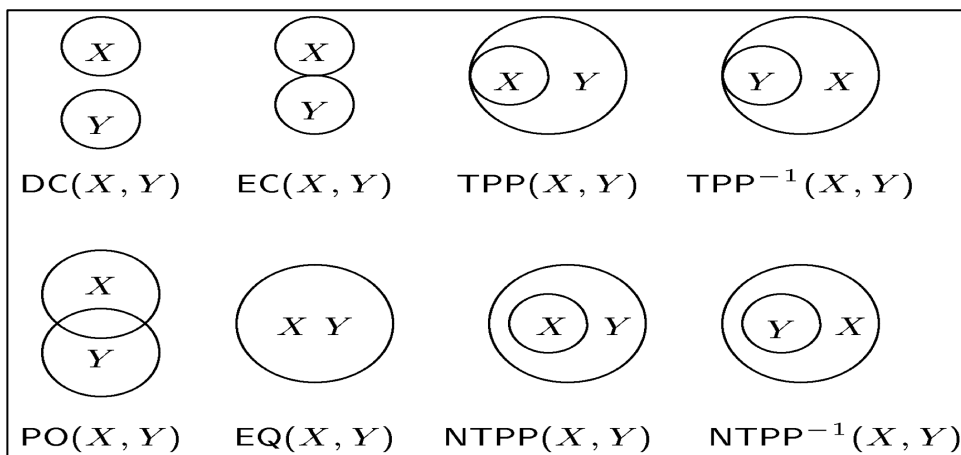
SUPPORTIVE SEMANTICS : TEMPORAL AND SPATIAL CALCULUS

◆ Allen's Temporal Interval Calculus



- **Interval-based theory:** proper time intervals defined from time instants
- **Restricted axioms** ensuring time reasoning **decidability** (OWL DL)
- Ex. *intOverlaps* (FOL)

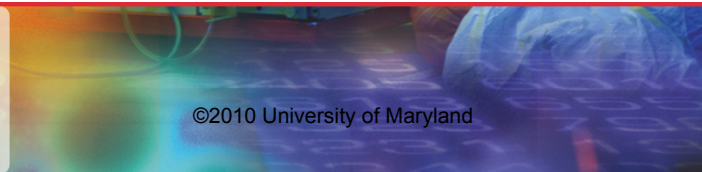
◆ Region Connection Calculus(RCC)



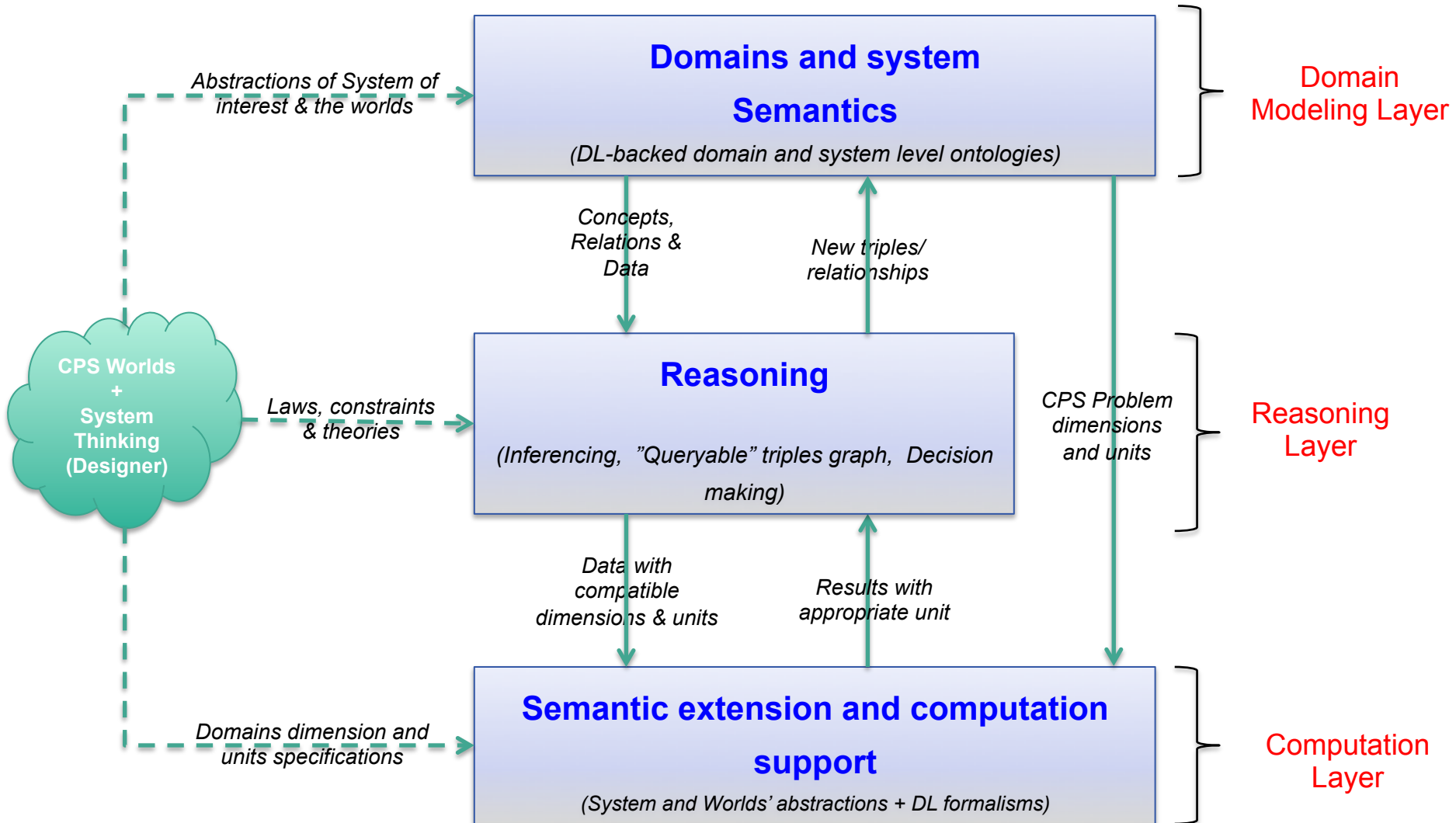
- **Region-based spatial model:** Regular 2D spaces, convex shapes
- **"Maximal fragment" satisfiable in polynomial time** (Renz 1999)
- Ex. *disjoint* (DC)

$$(\forall x \forall y (DC(x, y) \leftrightarrow \neg C(x, y)))$$

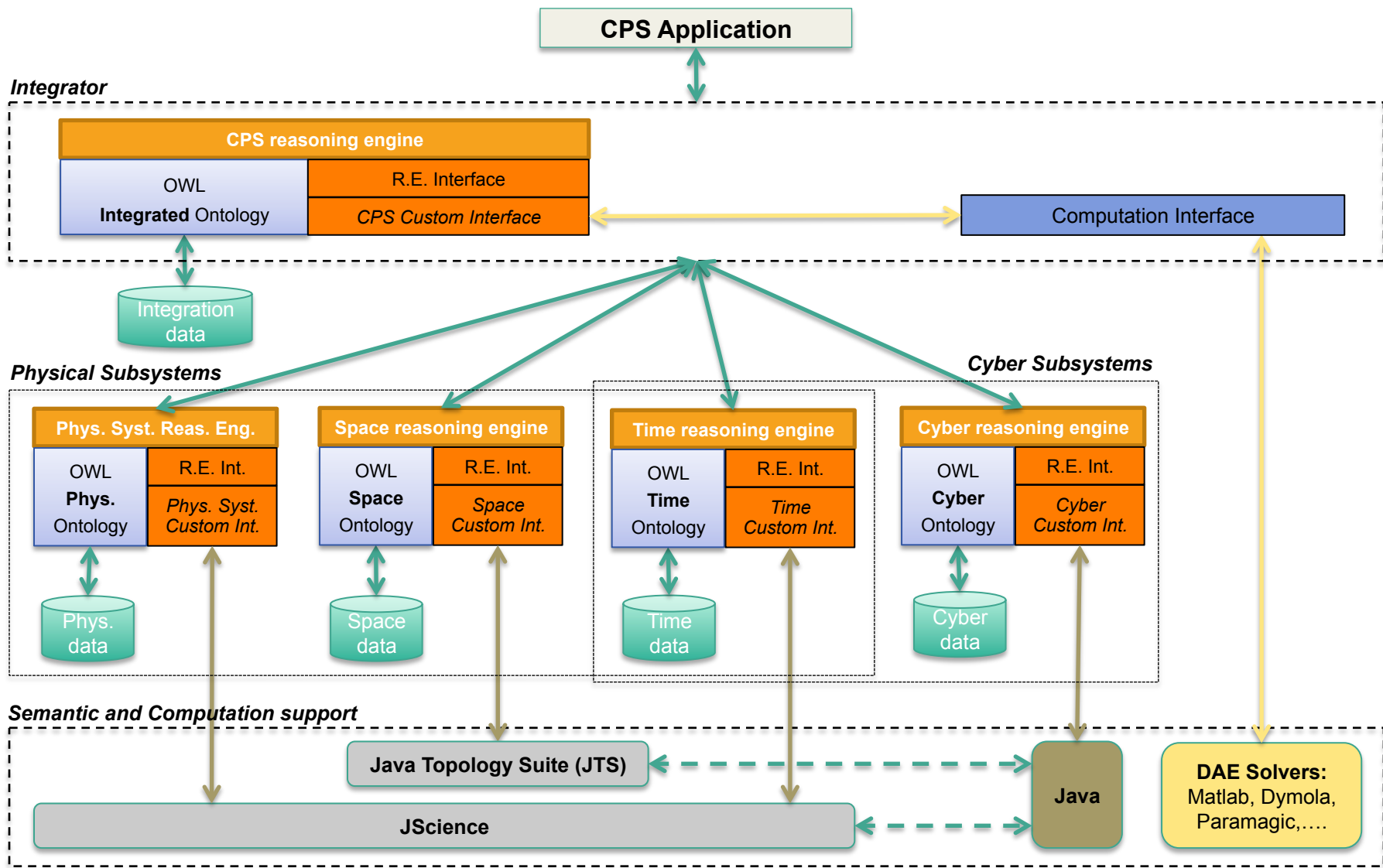
➔ Trade-off between expressiveness and computability for satisfiability!



SYSTEM ARCHITECTURE - HIGH-LEVEL



SYSTEM ARCHITECTURE – COMPONENTS AND INTEGRATION



Applications

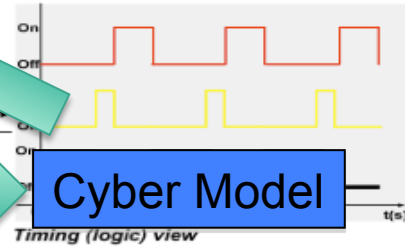
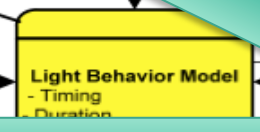
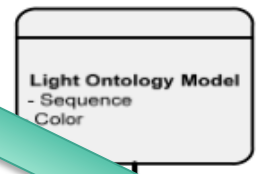
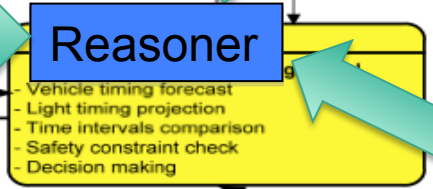
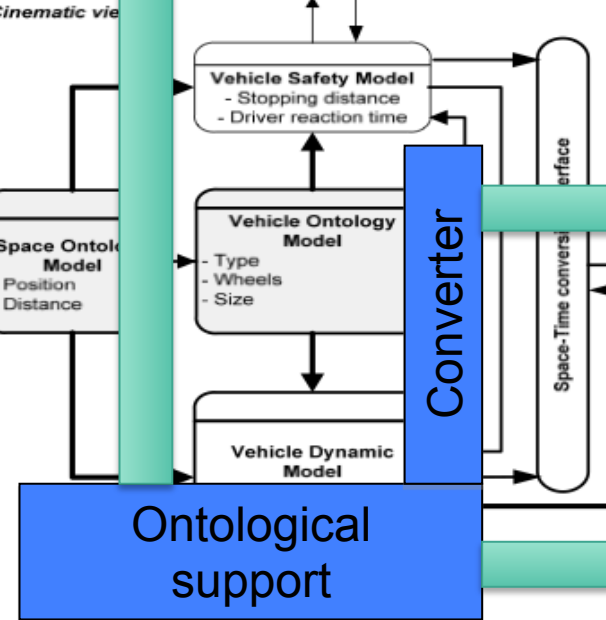
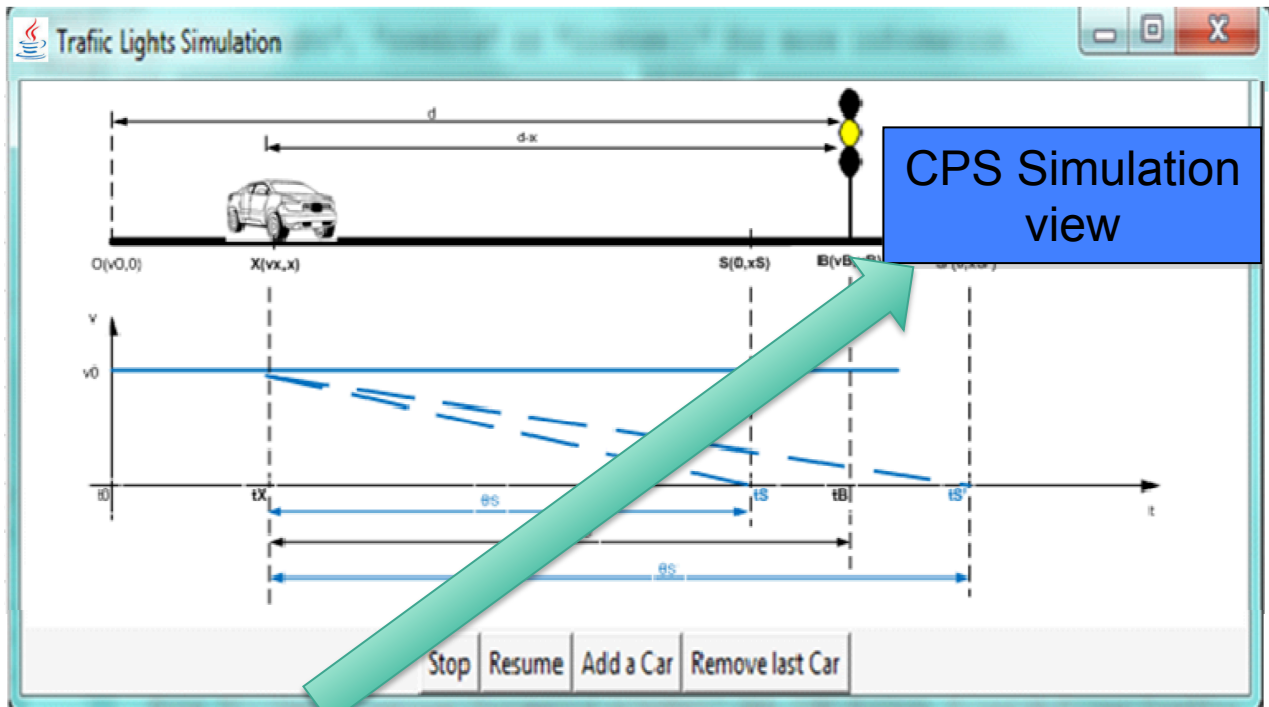
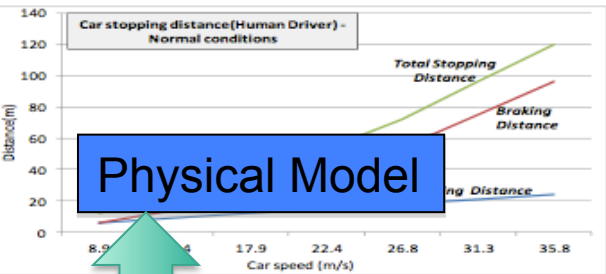
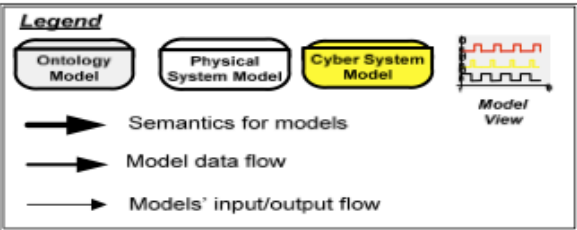


The
Institute for
Systems
Research

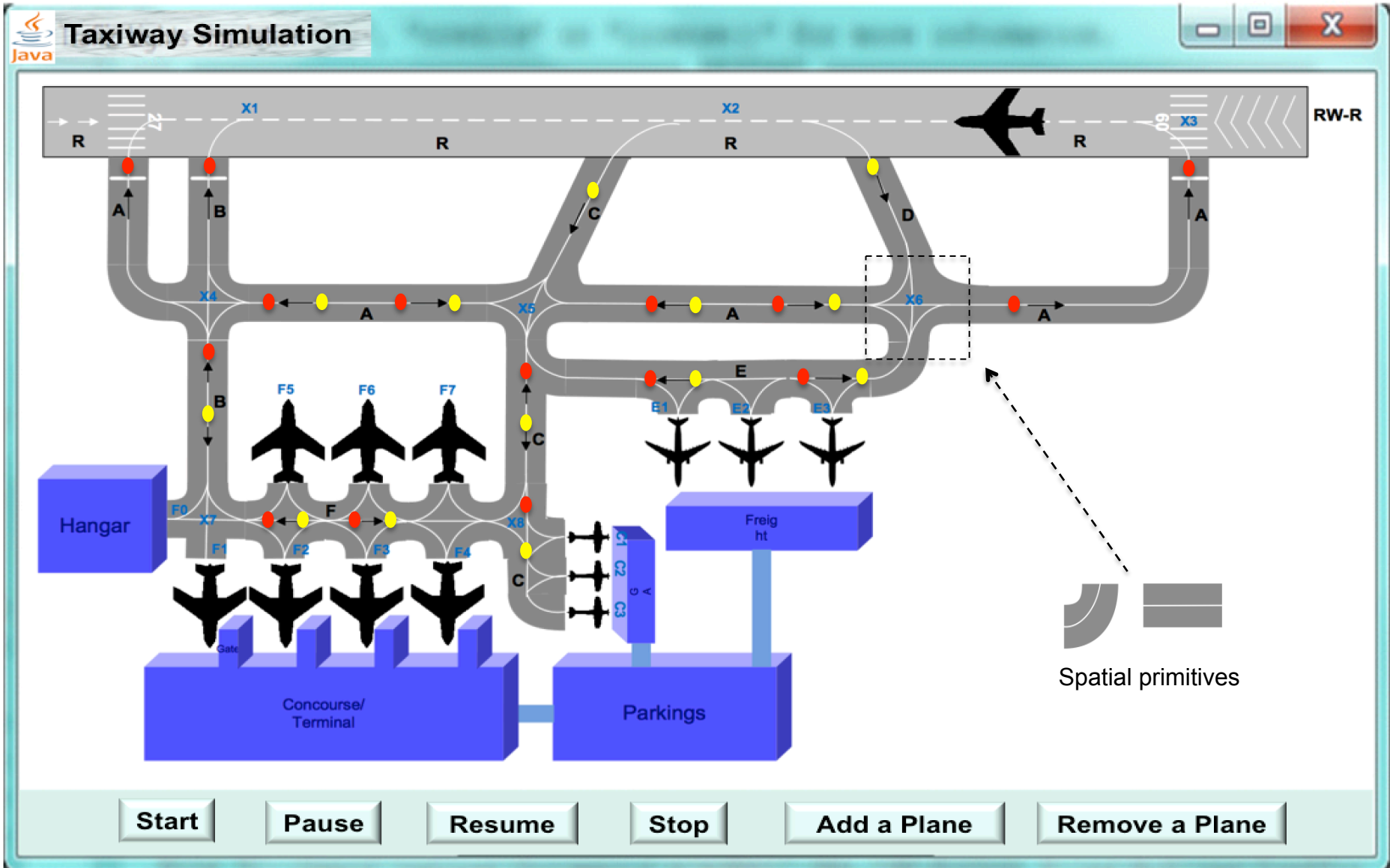
THE A. JAMES CLARK SCHOOL *of* ENGINEERING
UNIVERSITY OF MARYLAND

©2010 University of Maryland

APPLICATION 1 : DILEMMA ZONE PROBLEM AT TRAFFIC INTERSECTION



APPLICATION 2 : AUTOMATED TAXIWAY SYSTEM AT AIRPORT



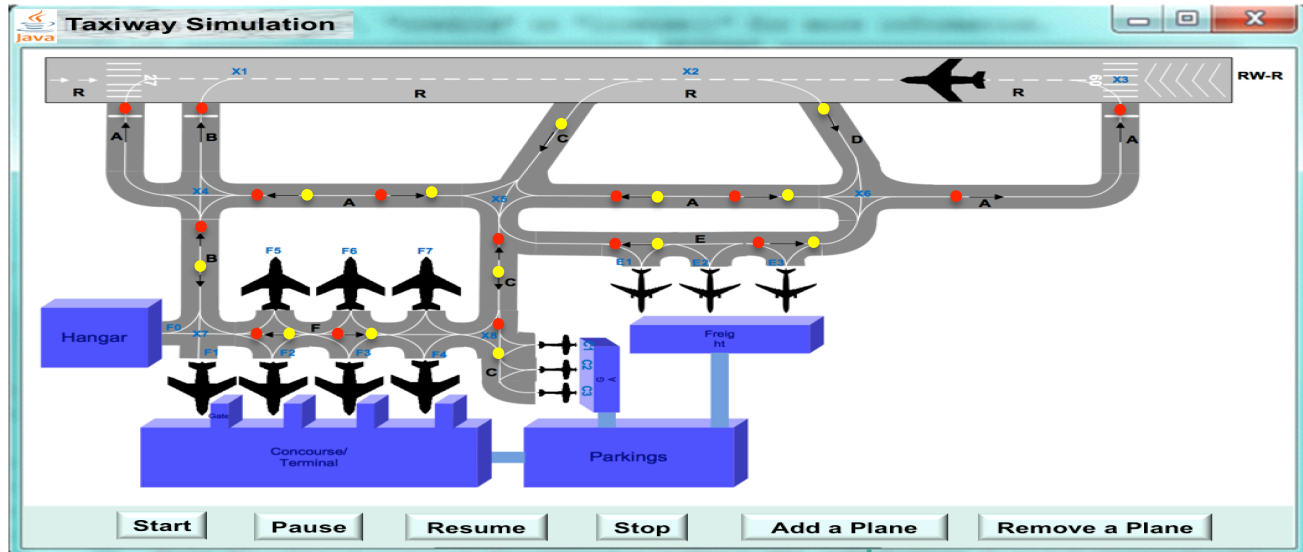
APPLICATION 2 : AUTOMATED TAXIWAY SYSTEM AT AIRPORT

Reasoning view

```

[3:00]
[3:00] ----- Path to DRIVER'S FINAL DECISION -----
[3:00]
[3:00] Resources: CarEntity members with defined/calculated "Stopping Distance" parameter ...
[3:00]
[3:00] The database contains stoppingDistance: for:
[3:00] Individual: http://www.isi.edu/~pon/dan/line/cor.uw/#2094FordTaurusS3S has stoppingDistance: 2428.8338800000004
[3:00] Individual: http://www.isi.edu/~pon/dan/line/cor.uw/#2010Ford800SUT has stoppingDistance: 629.592668000000L
[3:00]
[3:00]
[3:00] Resources: CarEntity members with defined/calculated "distance to light" parameter ...
[3:00]
[3:00] The database contains distanceToLight: for:
[3:00] Individual: http://www.isi.edu/~pon/dan/line/cor.uw/#2094FordTaurusS3S has distanceToLight: 2288.0
[3:00] Individual: http://www.isi.edu/~pon/dan/line/cor.uw/#2010Ford800SUT has distanceToLight: 490.0
[3:00]
[3:00]
  
```

Simulation Framework and 2D Visualization



Visualization Interface

Semantic modeling

Semantic extension & computation support

Domain & integration ontologies

Engineering modeling

Model object types (Physical-Engineering)

Libraries (computation & Physical Quantity)

Visualization Interface

Computation Interface

DBMS + Extension

Database

Executable Machine

Decision Tree

Dimensional Reduction

System requirements



$$X_{space} \rightarrow P_{space}$$

Problem analysis



The
Institute for
Systems
Research

THE A. JAMES CLARK SCHOOL of ENGINEERING
UNIVERSITY OF MARYLAND

REFERENCES

- Austin M.A. and Wojcik C.E., Ontology-Enabled Traceability Mechanisms, INCOSE 2010 ,Chicago, USA, July 12-15, 2010.
- Delgoshaei P. and Austin M.A., Software Patterns for Traceability of Requirements to Finite-State Machine Behavior: Application to Rail Transit Systems Design and Management, INCOSE 2012, Rome, Italy, 2012.
- Petnga L. and Austin M.A., Ontologies of Time and Time-Based Reasoning for MBSE of Cyber-Physical Systems, CSER 2013, Georgia Institute of Technology, Atlanta, GA, March 19-22, 2013. **Best Conference Paper Award.**
- Mosteller M., Austin M.A., Yang S., and Ghodssi R., Platforms for Engineering Experimental Biomedical Systems, INCOSE 2012, Rome, Italy, 2012. **Outstanding Systems Engineering Paper Award.**
- Delgoshaei P., Austin M.A., and Veronica D.A., A Semantic Platform Infrastructure for Requirements Traceability and System Assessment, 9th International Conference on Systems (ICONS 2014), Nice, France, February 23-27, 2014.
- Nassar N. and Austin M.A., Model-Based Systems Engineering and Trade-Off Analysis with RDF Graphs, CSER 2013, Georgia Institute of Technology, Atlanta, GA, March 19-22, 2013.



Thank You

E-mail: austin@isr.umd.edu

Web site: <http://www.isr.umd.edu/~austin>



The
Institute for
Systems
Research

THE A. JAMES CLARK SCHOOL *of* ENGINEERING
UNIVERSITY OF MARYLAND

©2010 University of Maryland