The Institute for Systems Research

"COMPASS: Component-based Architectures for Systems Synthesis"

John S. Baras

Institute for Systems Research
Department of Electrical and Computer Engineering
Fischell Department of Bioengineering
Applied Mathematics, Statistics and Scientific Computation Program
University of Maryland College Park

Invited keynote address

2012 MODPROD

February 8, 2012

Linkoping University, Sweden

Acknowledgments

- **Joint work with:** Shah-An Yang, Mark Austin, Kiran Somasundaram, Vahid Tabatabaee, Kaustubh Jain, Senni Perumal, George Theodorakopoulos, Dimitrios Spyropoulos, Yuchen Zhou, Leonard Petnga
- **Sponsors:** NSF, AFOSR, ARO, ARL,NSF, DARPA, SRC, NIST, Lockheed Martin, Northrop Grumman, Telcordia

“The Nation that has the System Engineers has the Future”

John S. Baras, *Systems and Signals*, Vol. 4.2, May 1990

The Next Frontier in Engineering Research and Education

- First 25 years of the 21st century will be **dominated** by advances in methods and tools for the **synthesis of complex engineered systems to meet specifications in an adaptive manner**
- Evident from the areas emphasized by governments, industry and funding agencies world-wide:

  - **energy and smart grids**
  - **biotechnology**
  - **systems biology**
  - **nanotechnology**
  - **the new Internet**
  - **collaborative robotics**
  - **software critical systems**
  - **homeland security**
  - **materials design at sub-molecular level**
  - **network science**
  - **environment and sustainability**
  - **intelligent buildings and cars**
  - **customizable health care**
  - **pharmaceutical manufacturing innovation**
  - **broadband wireless networks**
  - **sensor networks**
  - **transportation systems**
  - **security-privacy-authentication in wireless networks**
  - **cyber-physical systems**
  - **web-based social and economic networks**

THE NEXT FRONTIER IN ENGINEERING RESEARCH AND EDUCATION (CONT.)

- Encounter frequently ***system of systems***
- Complexity manifests itself through **heterogeneity of subsystems and components**
- The synthesis of complex engineered and other systems from components so as to meet specifications and the associated education **represent the next frontier in engineering research and education**
- It is the frontier that **will determine the next generation leaders among Universities and industry**

The Critical Role of IT

- Possible to undertake a successful research and education program to accomplish this vision is IT -- namely **networked embedded systems**
- Through embedded systems the heterogeneity of the various physical components is **translated into a common language** where design can be integrated
- Networked embedded systems have revolutionized cars, networks, energy, biology and many other fields; at scales from nano to macro
- Implied **programmability and re-programmability** has immense consequences

NETWORKED EMBEDDED SYSTEMS AND CPS SEI

Automotive

Industrial automation

Robotics

Aeronautics

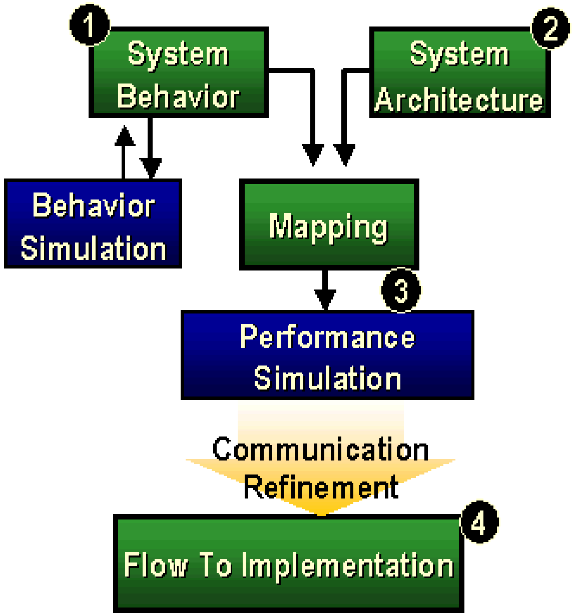
Elevators

Building automation

Balance Between System Behavior and System Structure

- With the exception of VLSI and (partly) embedded systems design, design and synthesis methodologies at the system level emphasize in an unbalanced way System Behavior (dynamics, functionality)
- *What is missing is System Structure!* **Namely, the system components and their physical realizations; "the platform"**

  "Platform-Based design" is a simple example



- **Systems Engineering is much harder than Software Engineering, because the design rules predicated by the physics of implementation (electrical, chemical, mechanical, hybrid, etc.) must be satisfied.**
- **Physics of implementation must be also selected: Multi-physics models and design**

Virtual Engineering Everywhere

Helping over 30 different teams and skills in the company work together

Linking over 40 different EE design representations throughout the entire development process

Ensuring that the EE design flow is integrated at the same level of quality and performance as the 3D CAD system

Model based design and executable specification in the OEM/supplier chain

Albert Benveniste -- *INRIA*

Virtual Engineering Everywhere CAD models

Helping over 30 different teams and skills in the company work together

Linking over 40 different EE design representations throughout the entire development process

Ensuring that the EE design flow is integrated at the same level of quality and performance as the 3D CAD system

Model based design and executable specification in the OEM/supplier chain

Albert Benveniste -- *INRIA*

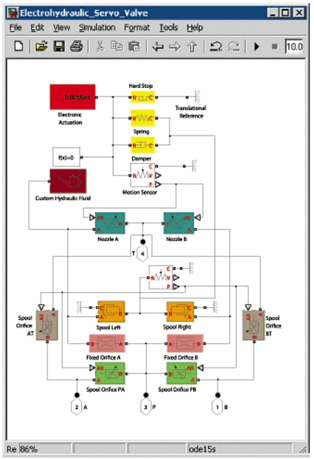
Virtual Engineering Everywhere Multi-Physics models

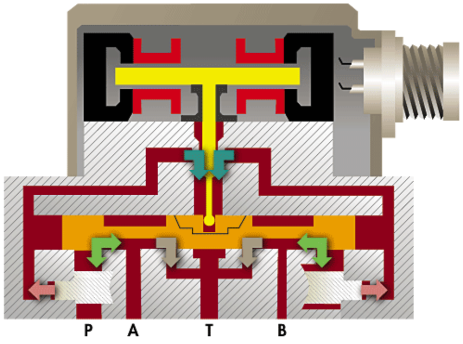
Helping over 30 different teams and skills in the company work together

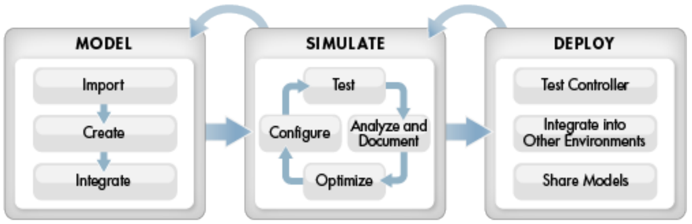
Linking over 40 different EE design representations throughout the entire development process

Ensuring that the EE design flow is integrated at the same level of quality and performance as the 3D CAD system

Model based design and executable specification in the OEM/supplier chain







Albert Benveniste -- *INRIA*

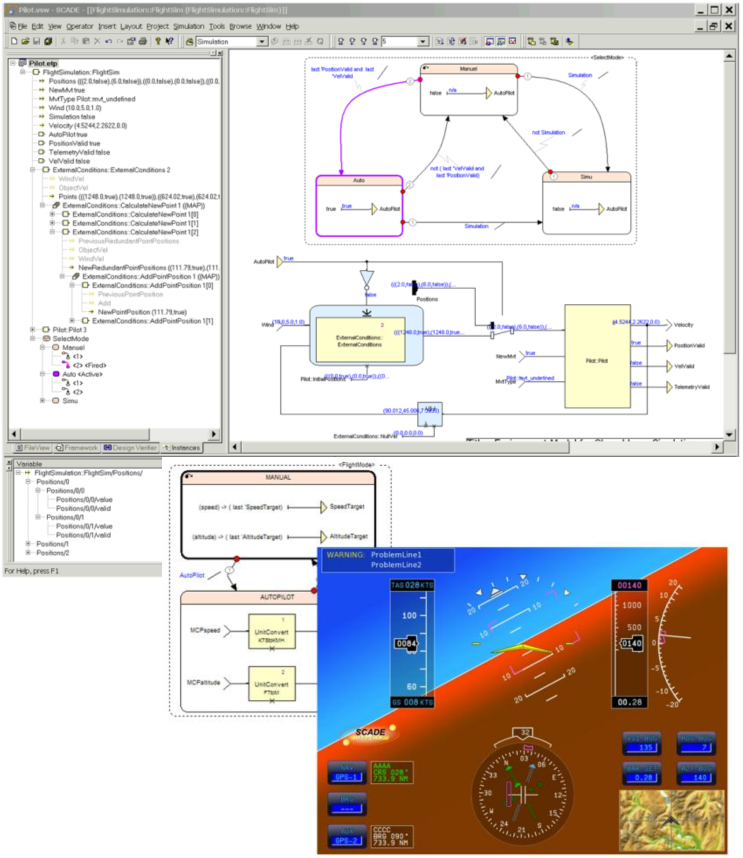
Virtual Engineering Everywhere Embedded Software

Helping over 30 different teams and skills in the company work together

Linking over 40 different EE design representations throughout the entire development process

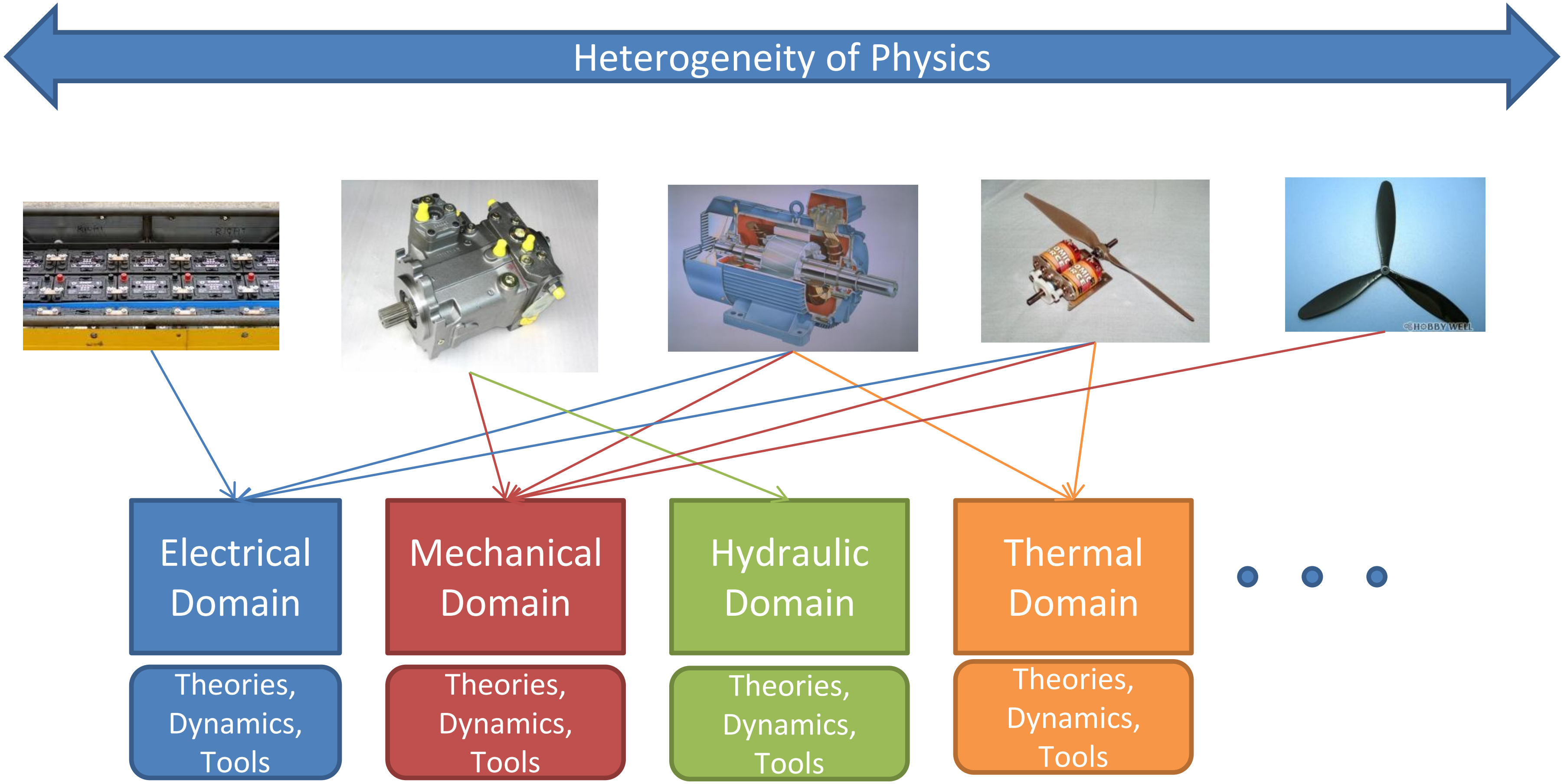
Ensuring that the EE design flow is integrated at the same level of quality and performance as the 3D CAD system

Model based design and executable specification in the OEM/supplier chain



Albert Benveniste -- *INRIA*

Model Integration Challenge: Physics



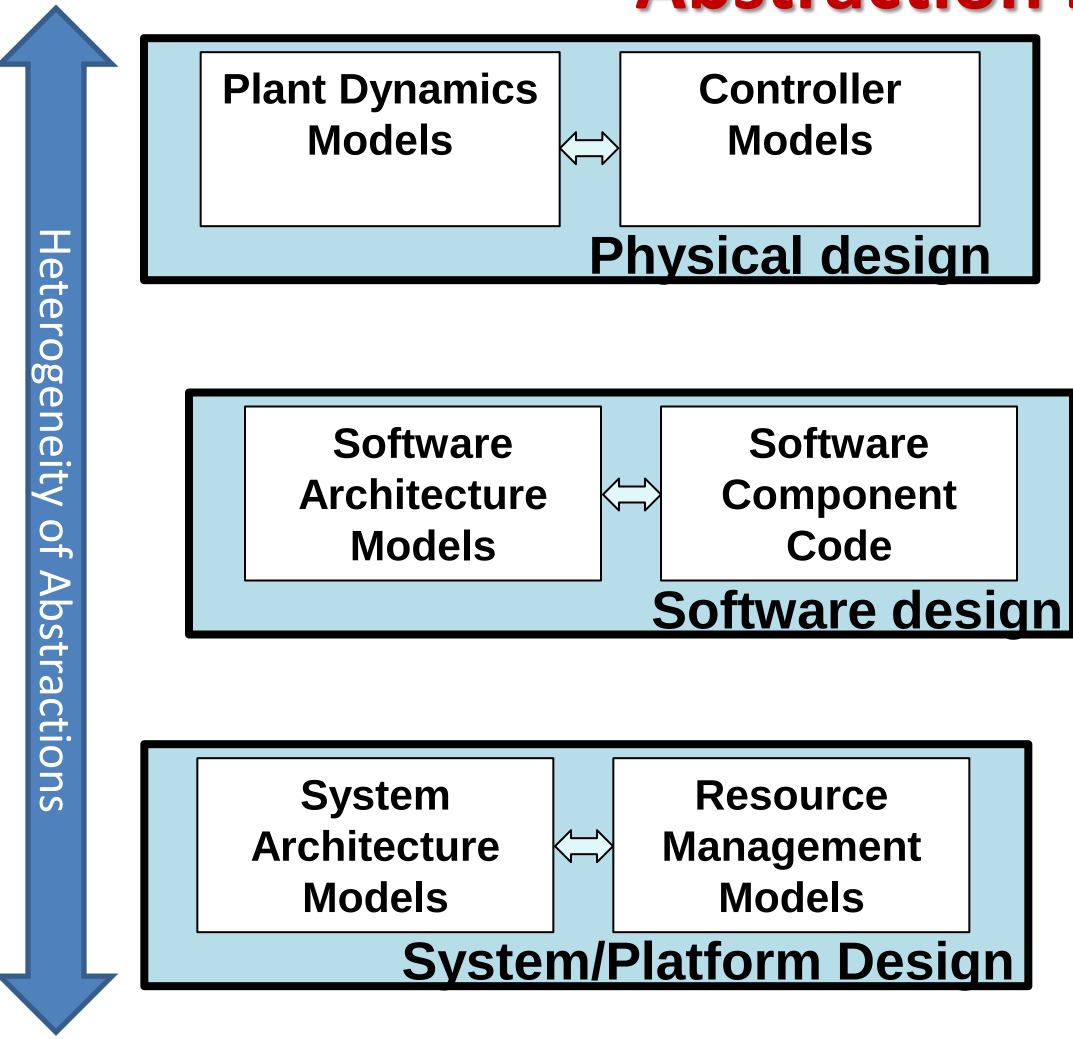
Physical components are involved in multiple physical interactions (multi-physics)
Challenge: How to compose multi-models for heterogeneous physical components

Janos Sztipanovits – *Vanderbilt Un.*

Model Integration Challenge: Abstraction Layers

The Institute for Systems Research

University of Maryland 1856



Dynamics: $B(t) = \kappa_p(B_1(t),...,B_j(t))$

- *Properties*: stability, safety, performance
- *Abstractions*: continuous time, functions, signals, flows,…

Software : $B(i) = \kappa_c(B_1(i),...,B_k(i))$

- *Properties*: deadlock, invariants, security,…
- *Abstractions*: logical-time, concurrency, atomicity, ideal communication,..

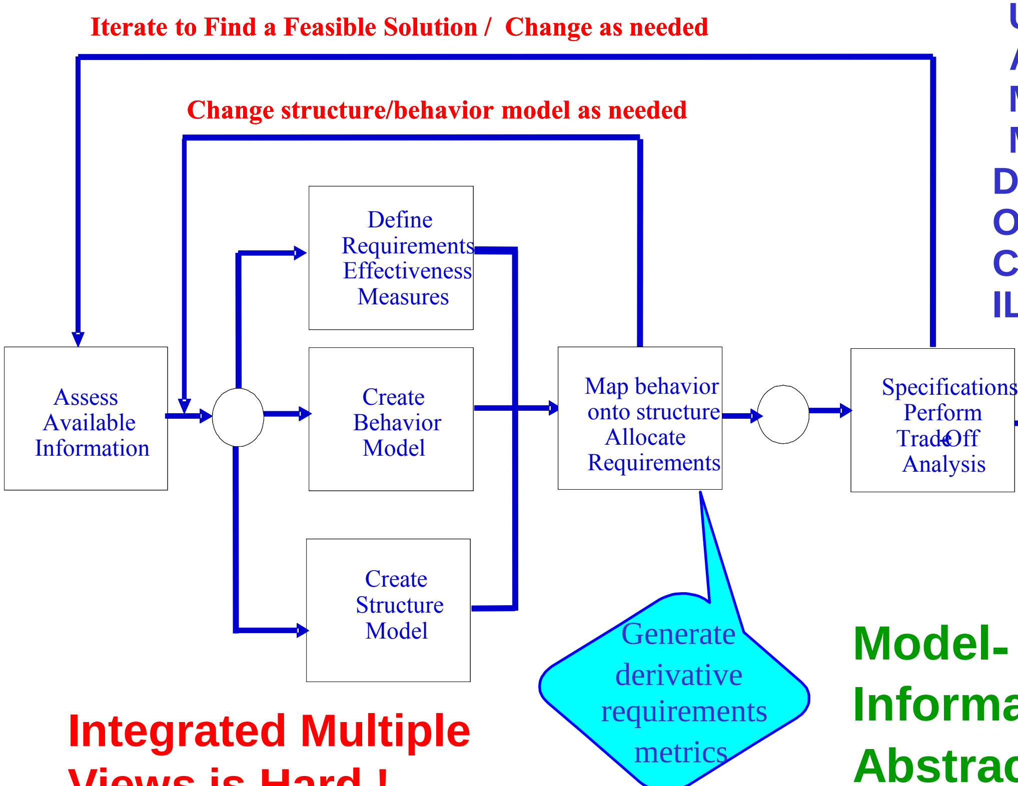
Systems : $B(t_j) = \kappa_p(B_1(t_i),...,B_k(t_i))$

- *Properties*: timing, power, security, fault tolerance
- *Abstractions*: discrete-time, delays, resources, scheduling,

Cyber-physical components are modeled using multiple abstraction layers
Challenge: How to compose abstraction layers in heterogeneous CPS components?

COMPONENT- BASED SYSTEM SYNTHESIS

Integrated System Synthesis - Tools & Environments missing



Model - based
- UML - SysML
- Rapsody
- UPPAAL
- Artist Tools
- MATLAB, MAPLE
- Modelica

DOORS, etc
OPCAD
CPLEX, SOLVER, ILOG

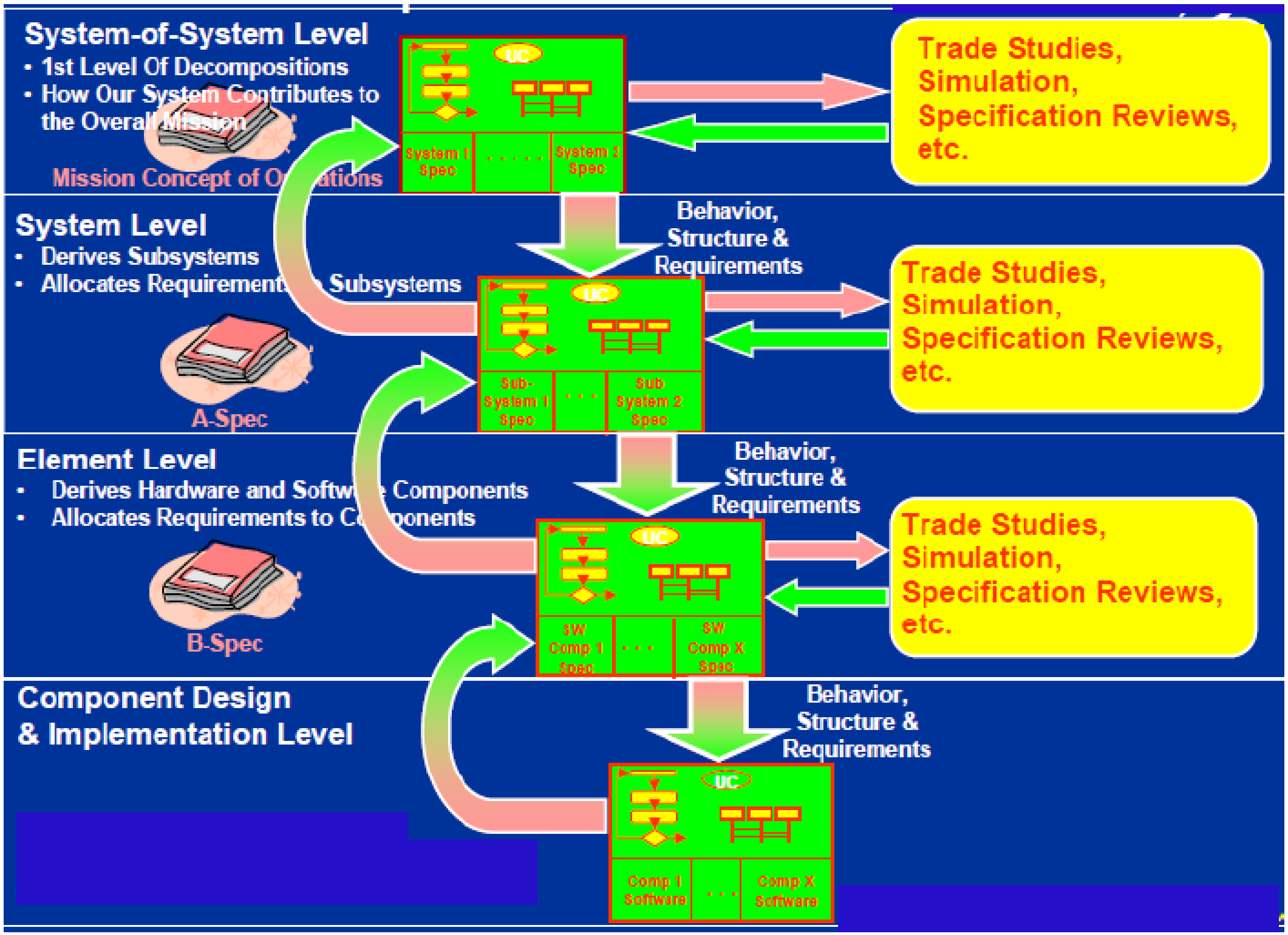
Integrated Multiple Views is Hard !

Model- Based Information - Centric Abstractions

CORE SE TOPICS

- Object Oriented modeling and beyond
- Automata, languages, design rules
- Trade-off analysis and multi-objective optimization
- Testing, validation, behaviors
- Logic programming and optimization
- Performance over time, hybrid systems
- Simulation and performance analysis

Layered MBSE -- Hierarchies

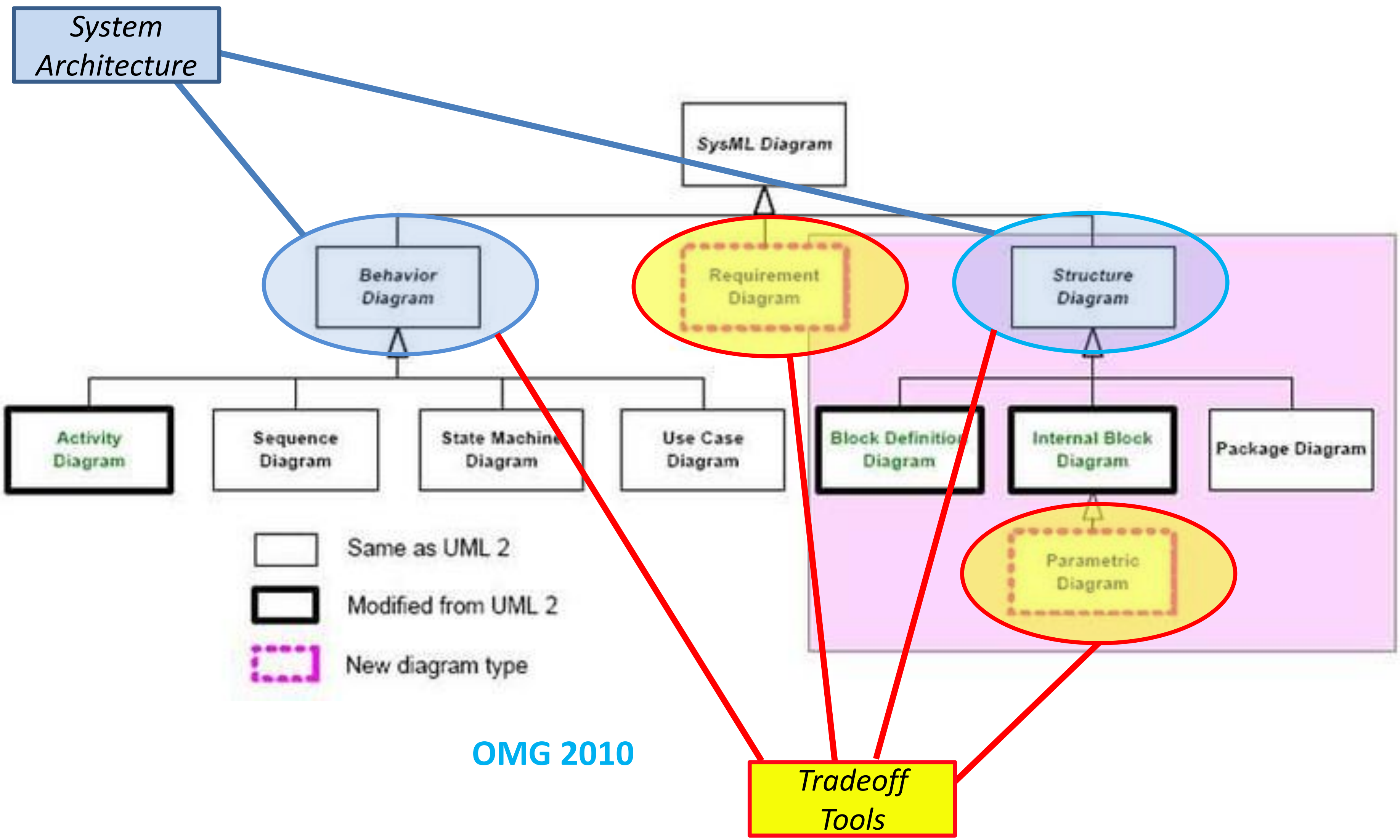


(Watson 2008, Lockheed Martin)

SysML at a Glance

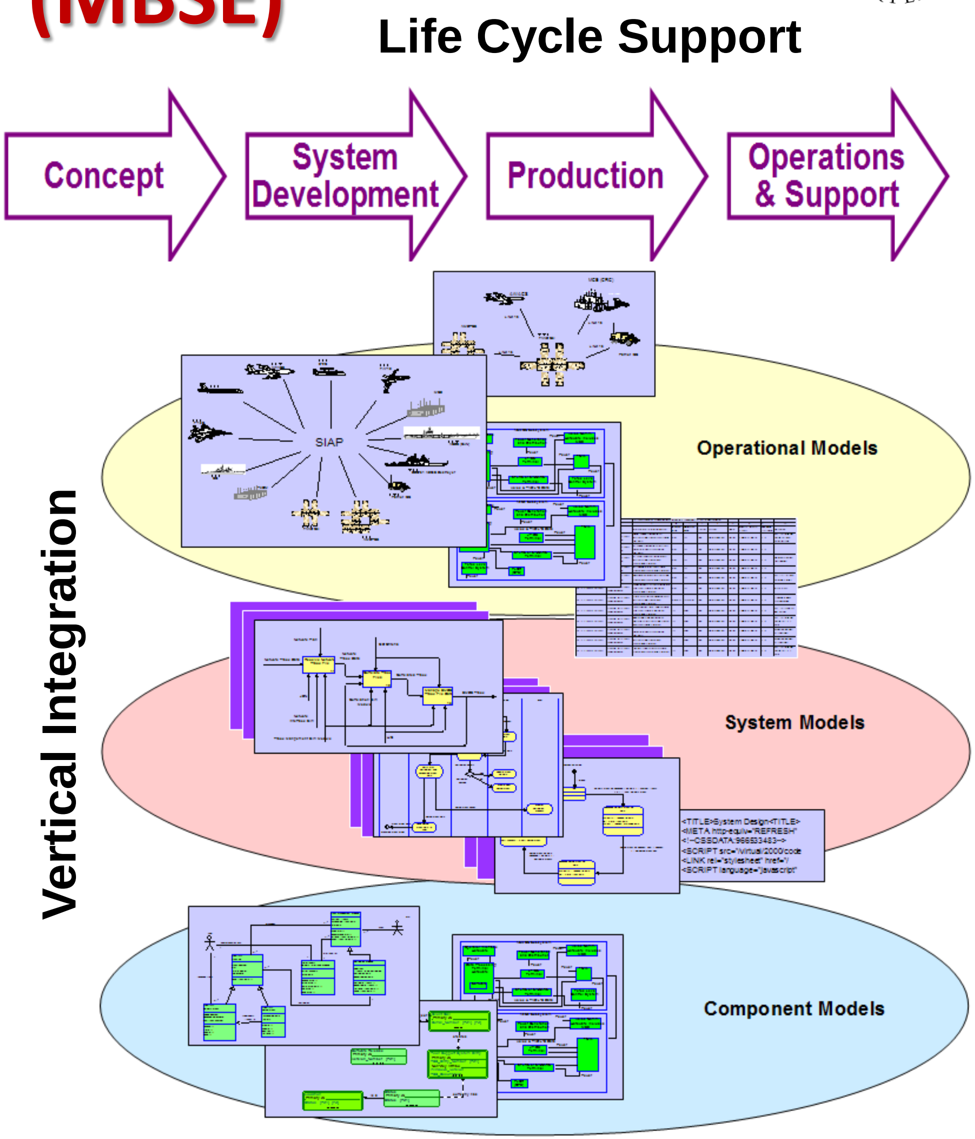
- SysML is a general purpose modeling language for systems engineering applications.
- SysML supports the analysis, specification, design, verification and validation of complex systems .
- SysML is intended to specify and architect systems and its components that can then be designed using other domain specific languages.
- SysML is an open source modeling standard supported by OMG .
- SysML can be used as an **integration framework** for multiple heterogeneous systems, subsystems and components modeling and analysis tools .
- SysML models of **system behavior** and **structure** can serve as the unifying **system architecture model** of a complex system or system of systems (SoS).

SysML Taxonomy

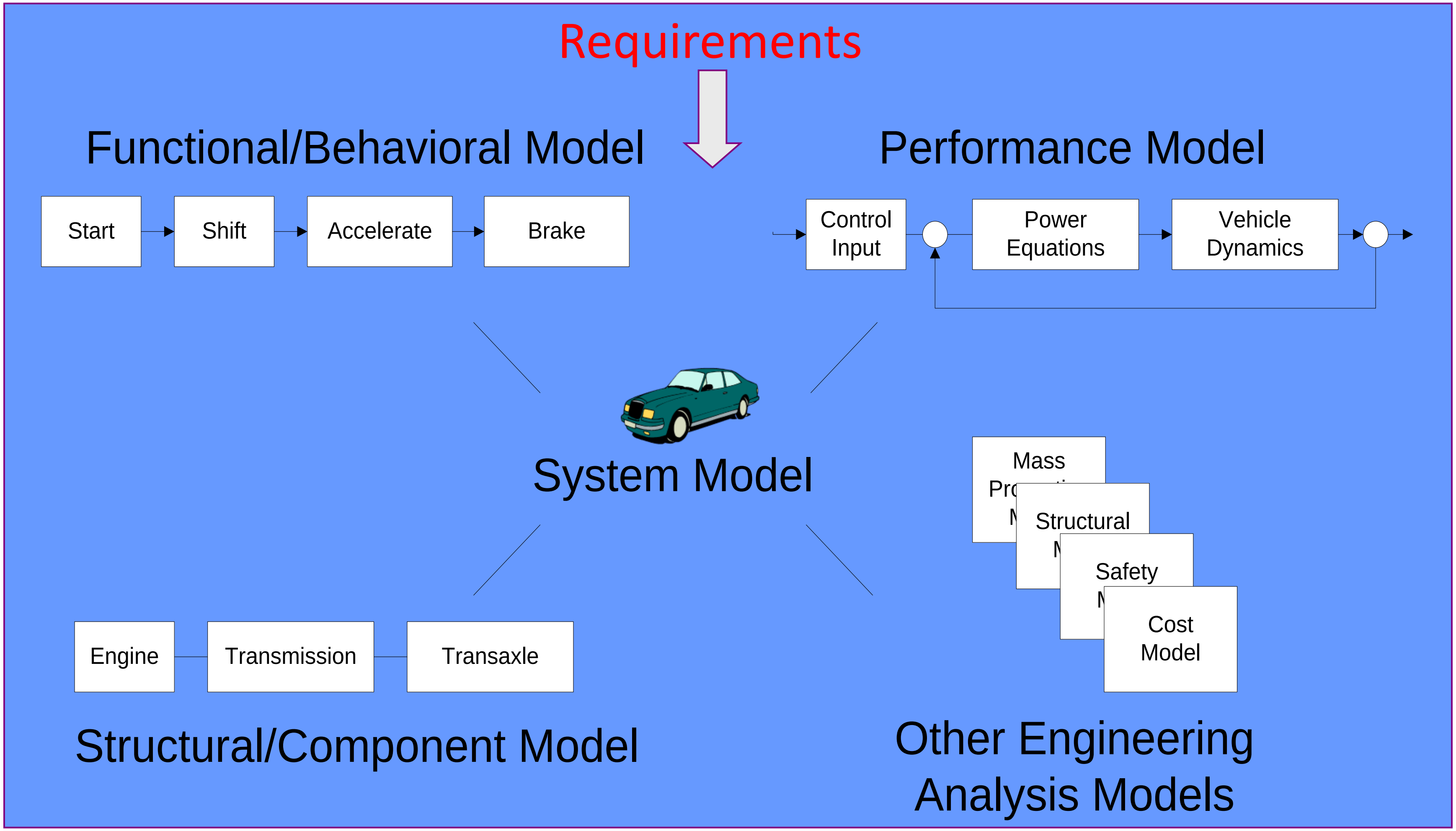


MODEL- BASED SYSTEMS ENGINEERING (MBSE)

- **Formalizes the practice of systems development through use of models**
- Broad in scope
  - Integrates with multiple modeling domains across life cycle from system of systems to component
- Results in quality/productivity improvements & lower risk
  - Rigor and precision
  - Communications among system/project stakeholders
  - Management of complexity

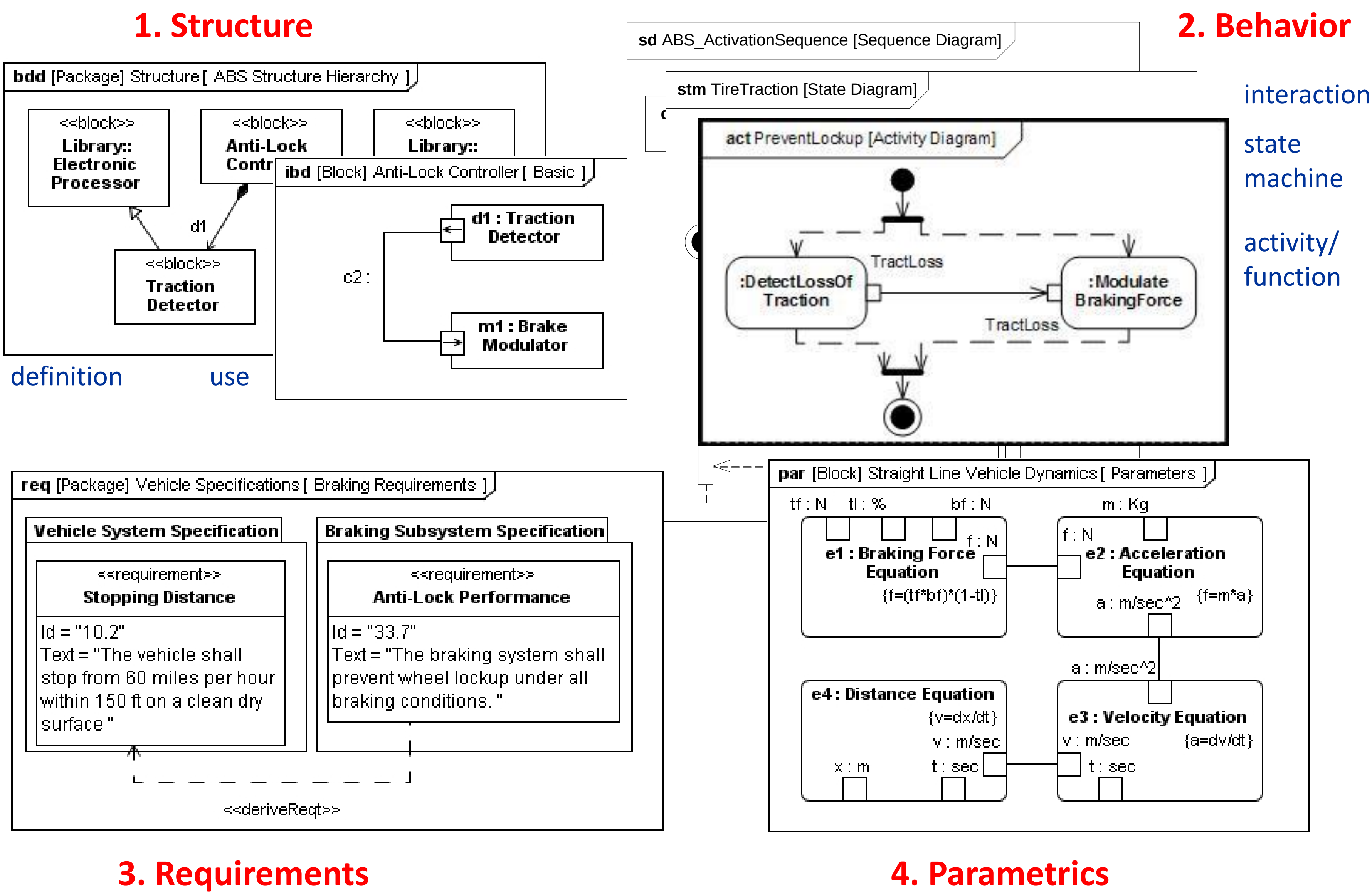


SYSTEM MODELING

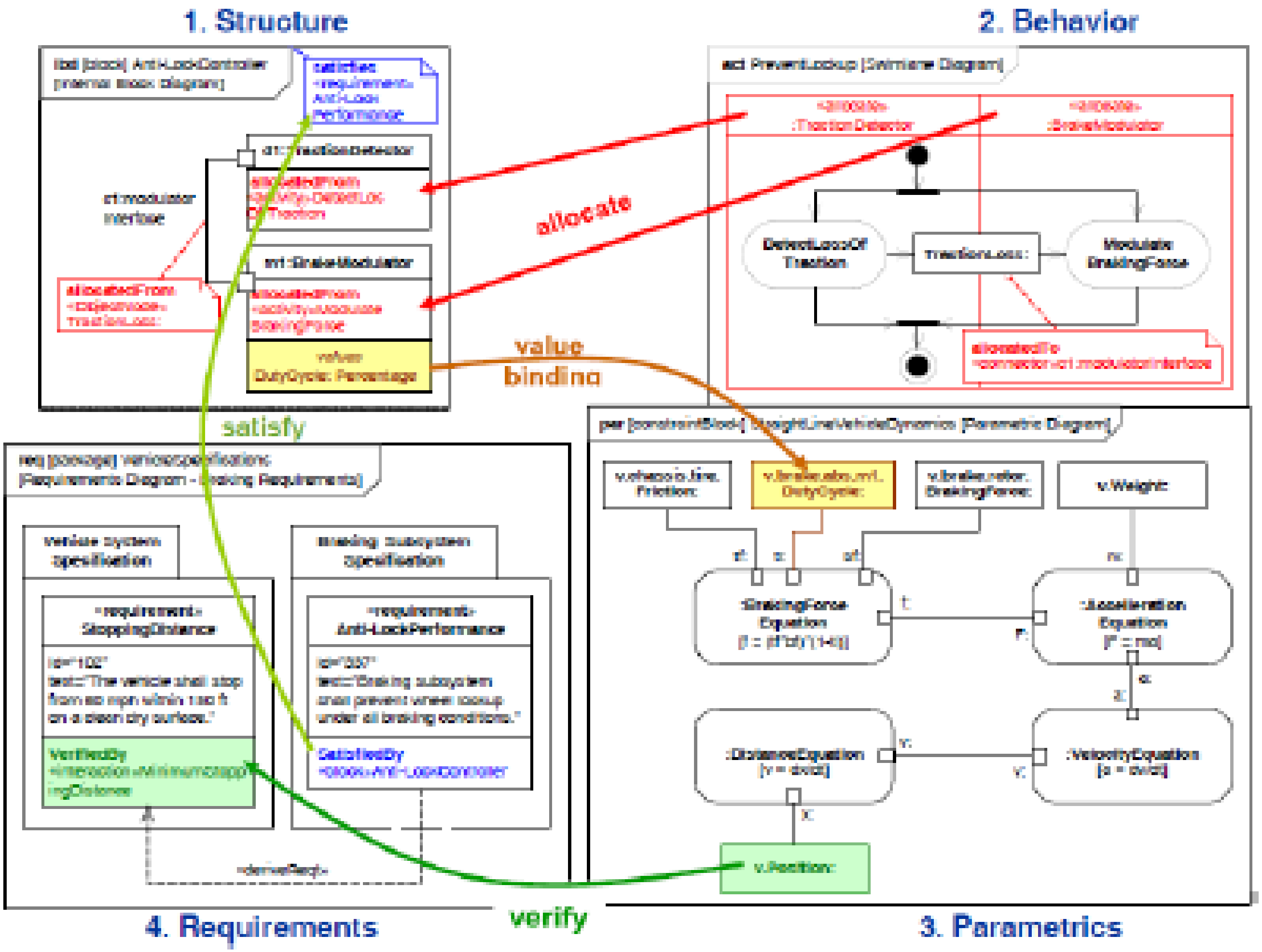


Integrated System Model Must Address Multiple Aspects of a System

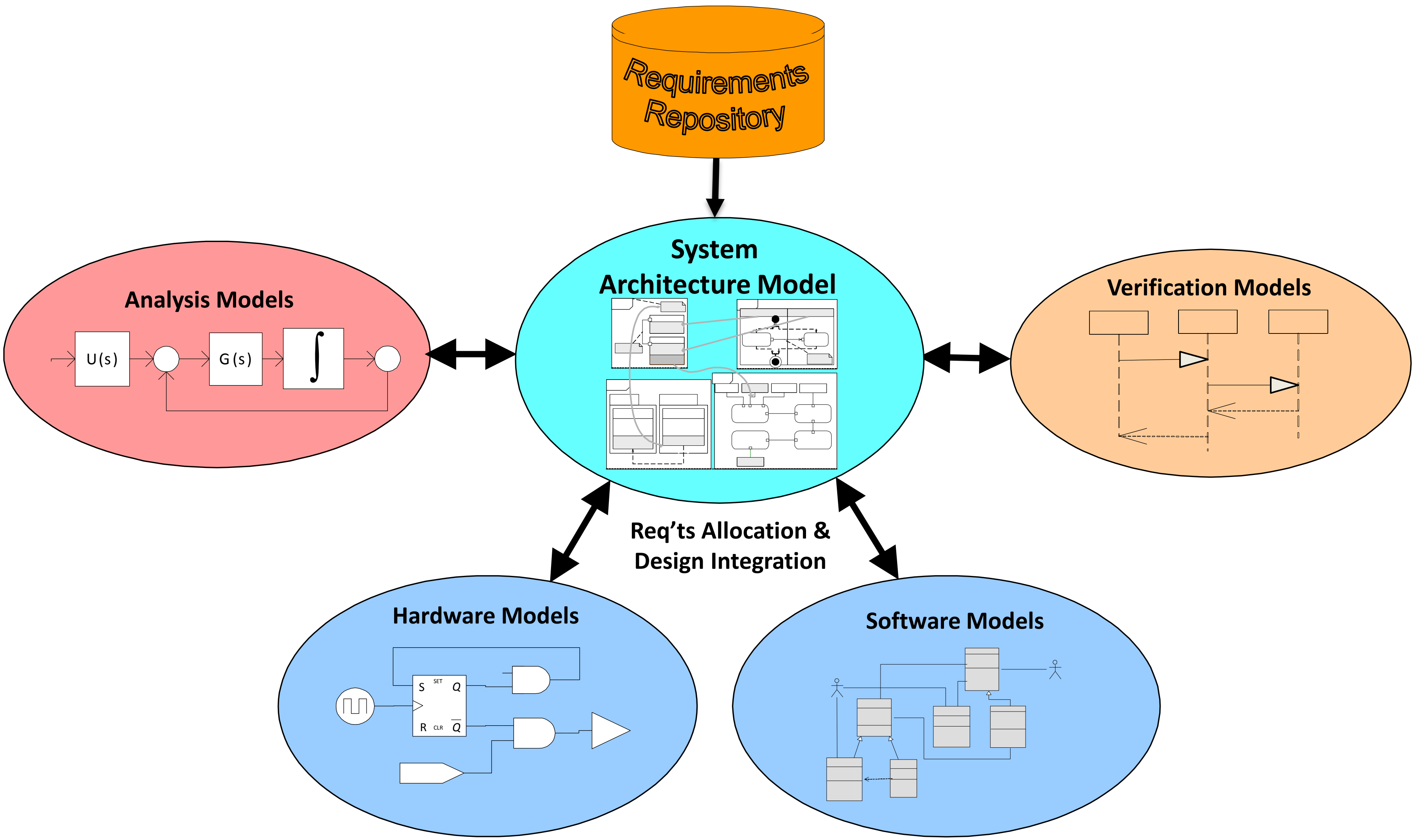
Four Pillars of SysML



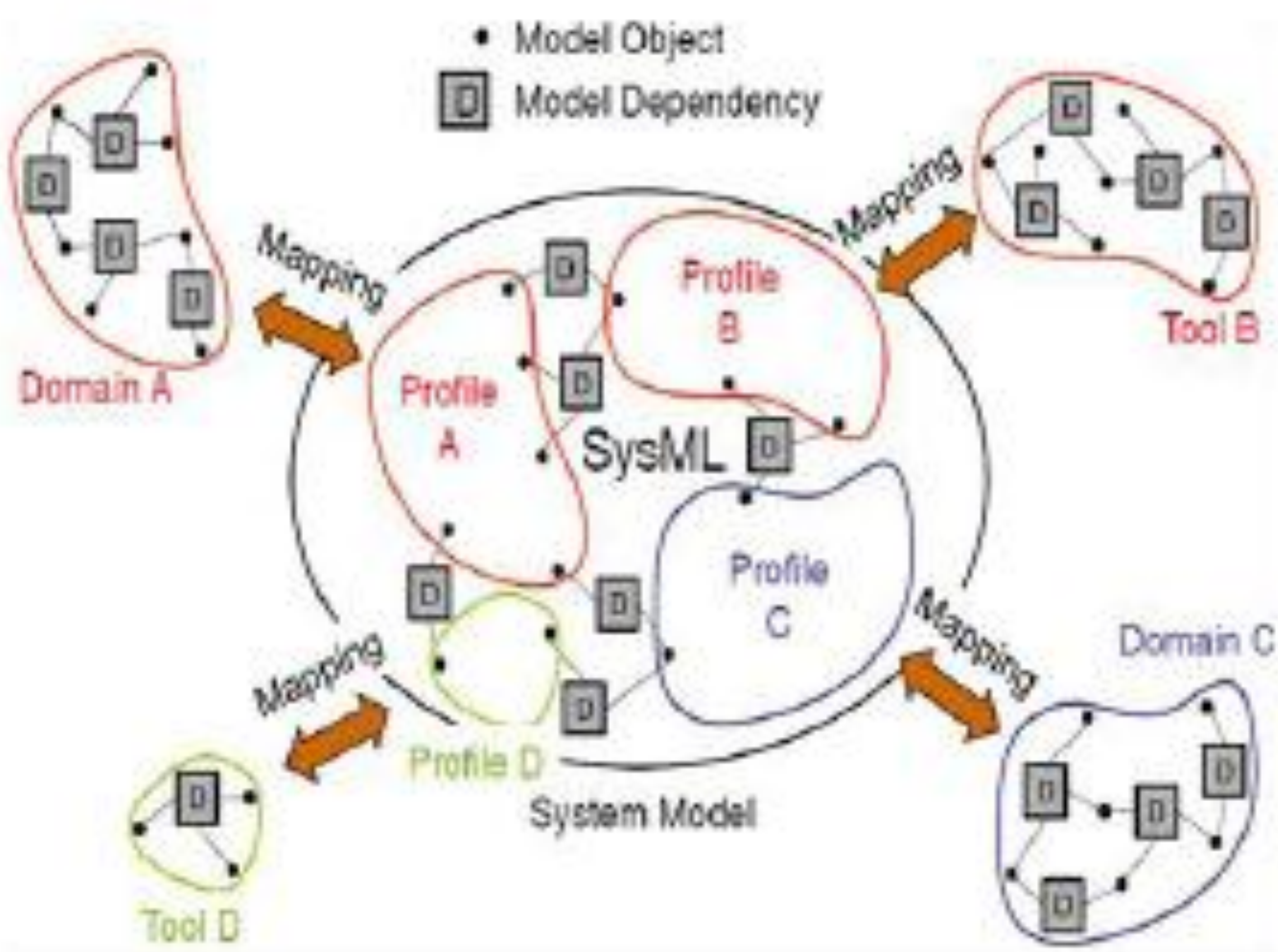
SysML


1. Structure
2. Behavior
allocate
value
binding
satisfy
verify
4. Requirements
3. Parametrics

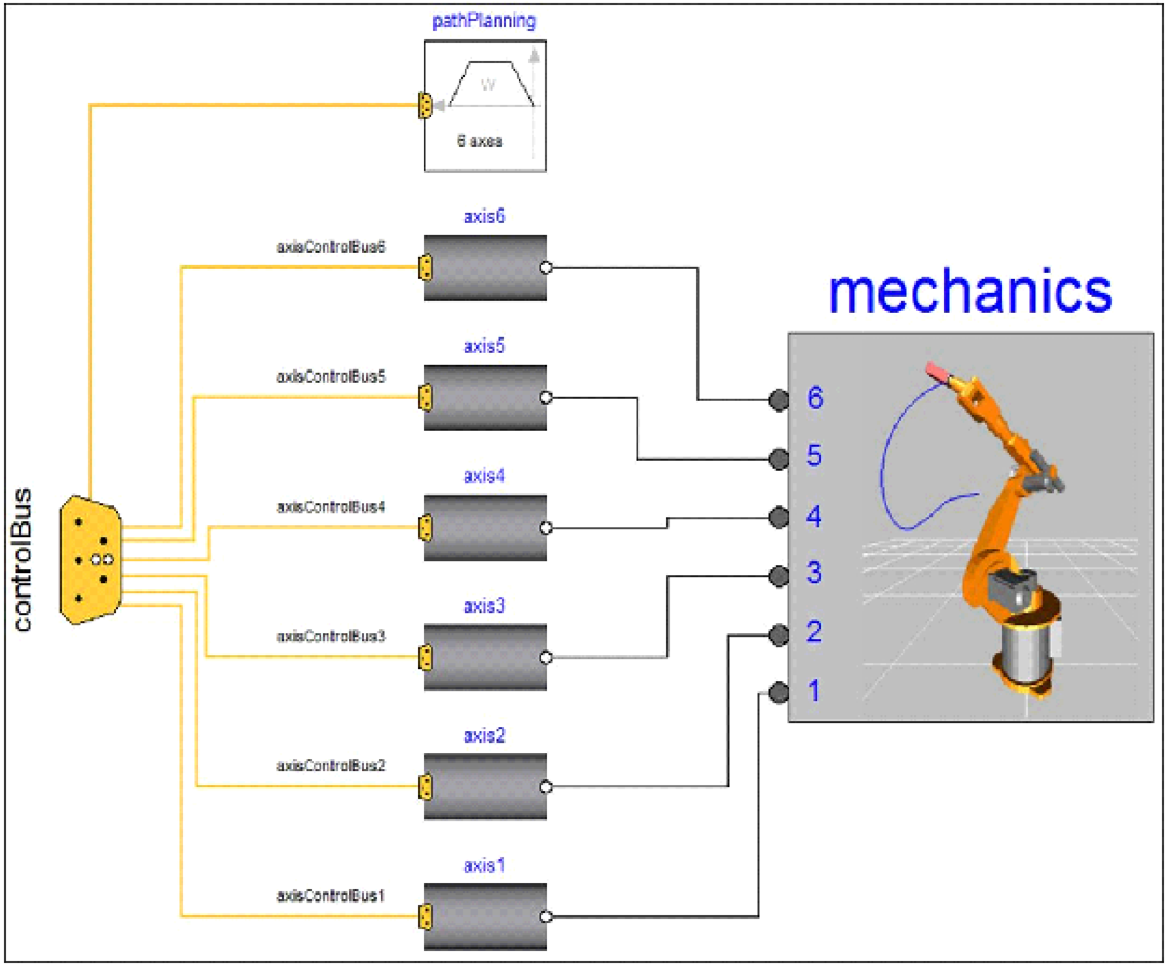
Using *System Architecture Model* as an Integration Framework

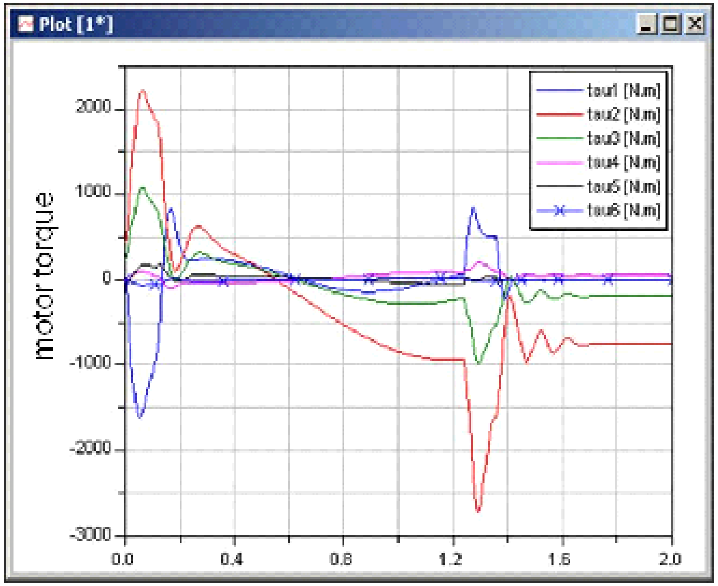


System Modeling Transformations



SysML-Modelica Robot Example: Modelica Model with Simulation Results

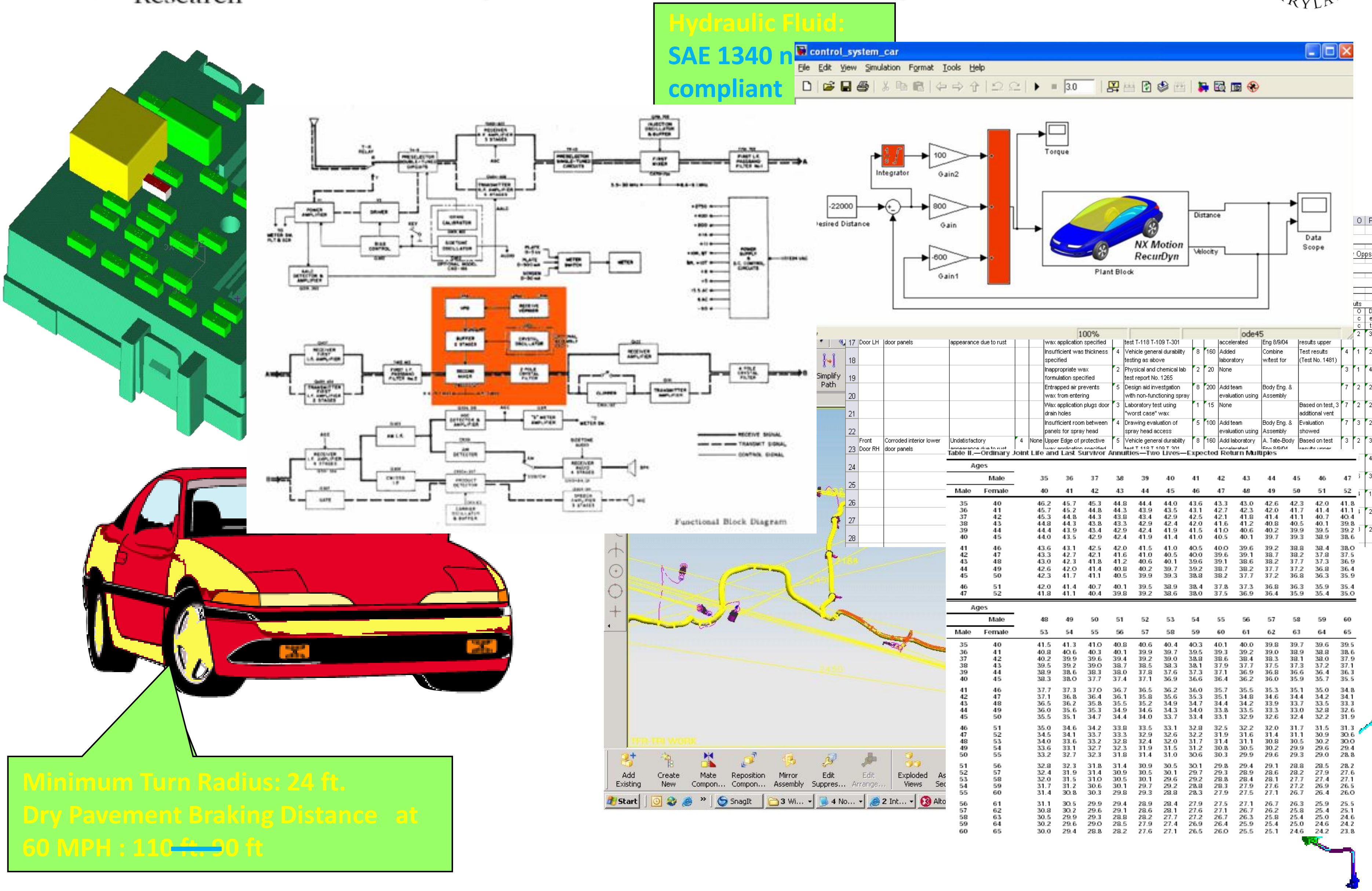




The Institute for Systems Research

Support Impact Assessment
(MBSE Vision, INCOSE)

University of Maryland 1856



COMPONENT- BASED HETEROGENEOUS NETWORK SYNTHESIS

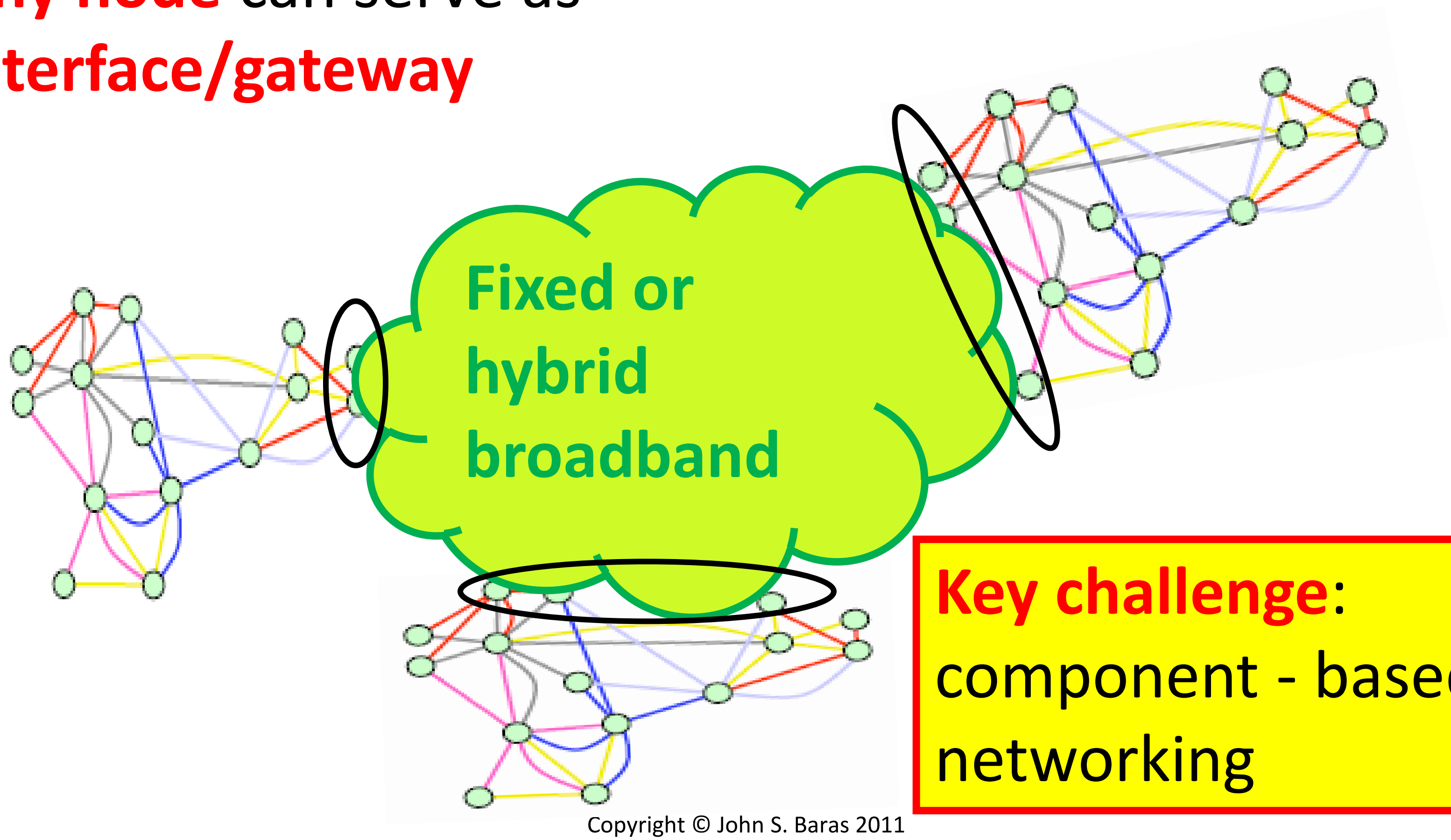
- **How to synthesize resilient, robust, adaptive networks?**

  → ***Component-Based Network Analysis & Synthesis (CBN)***

- **Components: modularity, cost reduction, re - usability, adaptability to goals, new technology insertion, validation and verification**
- **Interfaces: richer functionality– intelligent/cognitive networks**
- **Theory and Practice of Component-Based Networks**
  - **Heterogeneous components and compositionality**
  - **Performance of components and of their compositions**
  - **Back and forth from performance - optimization domain to correctness and timing analysis domain and have composition theory preserving component properties as you try to satisfy specs in both domains**
- **From communication to social, from cellular to transportation, from nano to macro networks**
- **Critical theory and methodology for Networked Embedded Systems, Cyber-Physical Systems, Systems Biology**

Dynamic Interconnection and Interoperability

- Broadband wireless nets capable for **multiple dynamic interface** points
- **Any node** can serve as **interface/gateway**



Key challenge: component - based networking

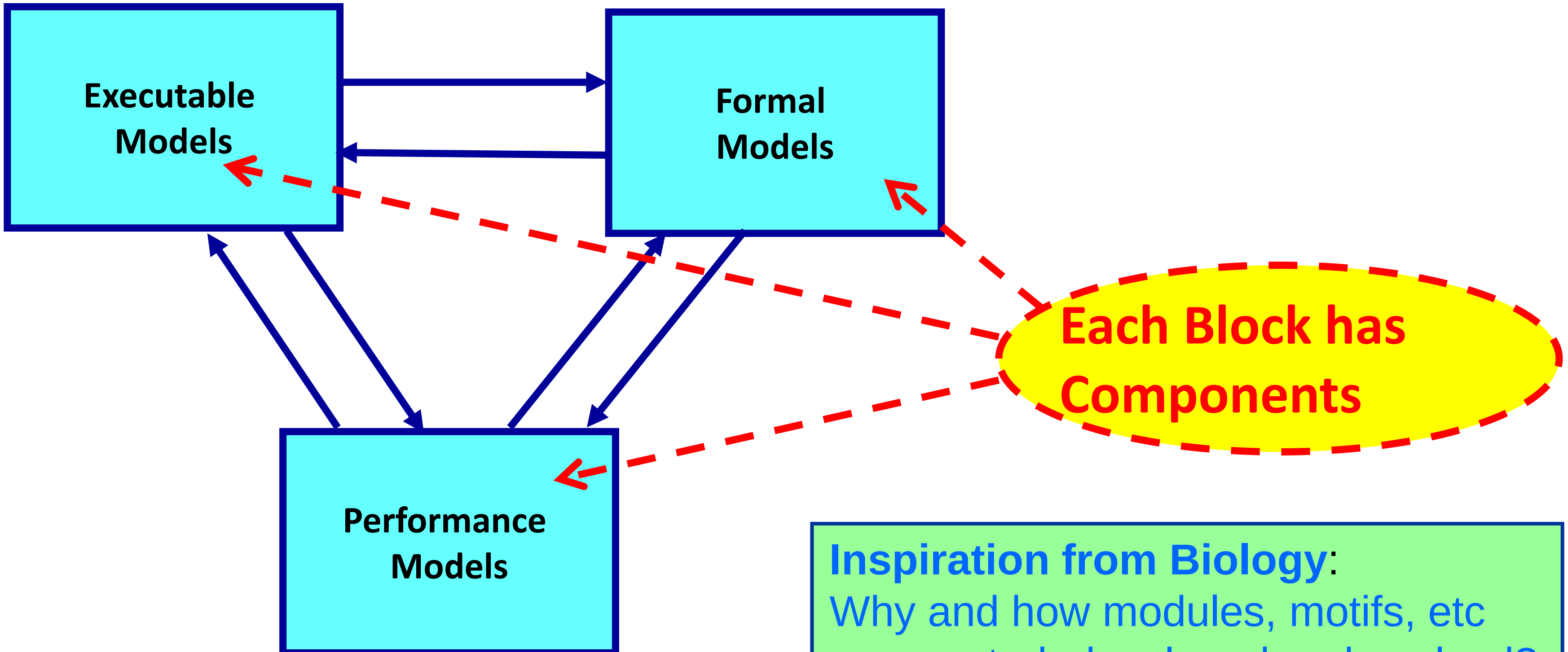
NETWORKS: DIFFERENT LINKED VIEWS

Networks:

- as **distributed, asynchronous, feedback (many loops), hybrid automata (dynamical systems)**
- as **distributed asynchronous active databases** and **knowledge bases**
- as **distributed asynchronous computers**

- **Can we**:
- **Develop a taxonomy of network structure vs network functionality?**
- **A theory of modularity and compositionality for networks?**

Component-Based Heterogeneous Networked Systems Synthesis



Inspiration from Biology:
Why and how modules, motifs, etc are created, developed and evolved?

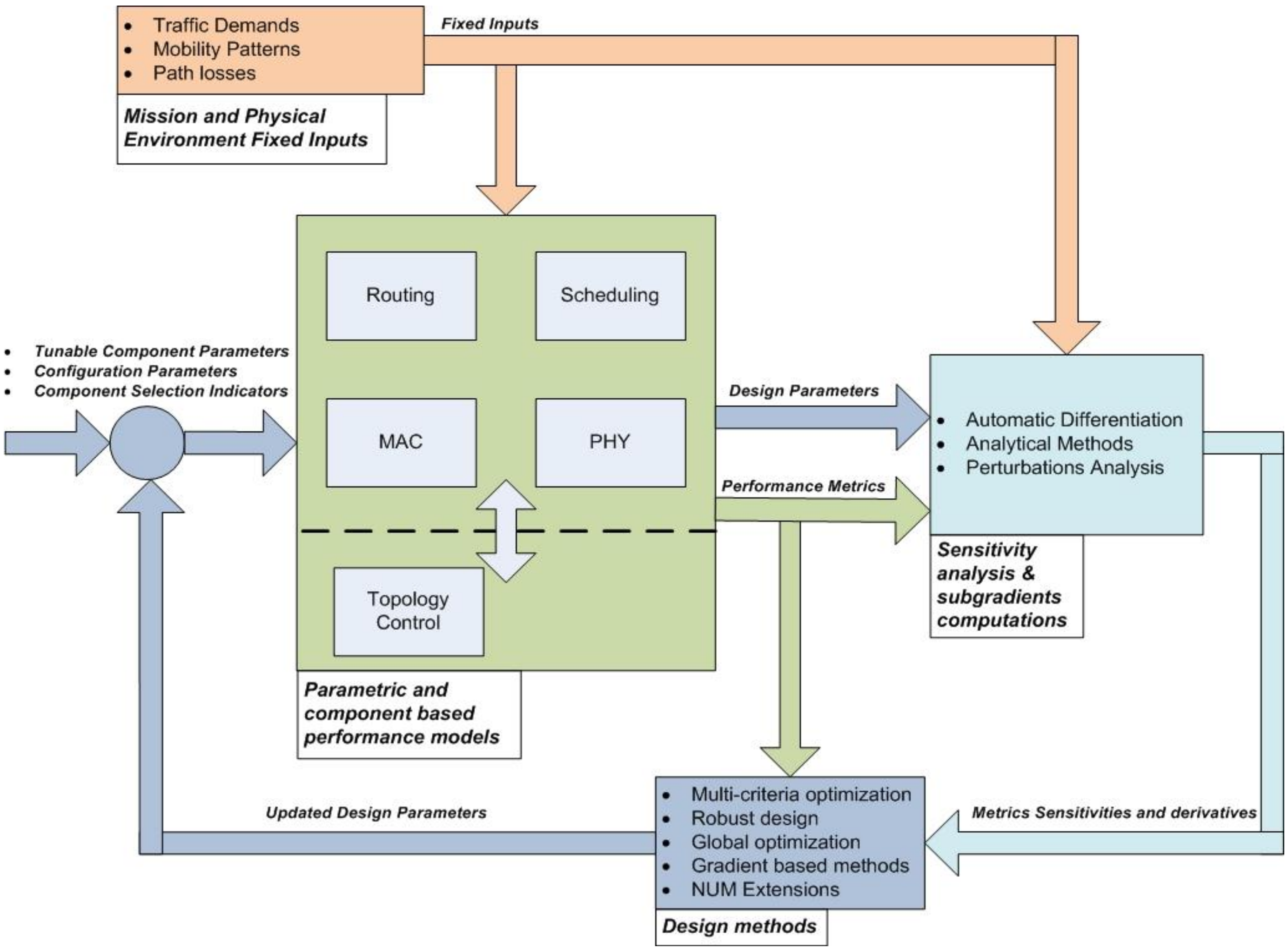
Grand challenge: Develop this framework for distributed, partially asynchronous systems, with heterogeneous components and time semantics

Model Based Wireless Network Design

- Objective: design mobile ad hoc wireless networks with predictable performance to meet specifications
- Approach: Combination of analytical and numerical performance assessment, linked with sensitivity analysis and design methodologies
- Challenges:
  - Development of simple analytical/numerical performance models that will achieve desired accuracy.
  - Identify metrics of component-level performance which are strongly correlated to network-level performance.
  - Development of design methodologies for components

Model-Based Design Tool

Inputs, components, design parameters, sensitivity analysis, optimization ….



MAC AND ROUTING COMPONENTS

- **Objective**
  - Design MANET adaptable to missions with predictable performance
- **Approach**
  - **Break traditional layers to components!** Develop component-based models MANET that considers cross-layer dependency to improve the performance
  - Study the effect of each component on the overall MANET performance

- **Routing Components** – routing protocols like OLSR [Baras08]
  - Neighbor Discovery Component (NDC)
  - Selector of Topology Information to Disseminate Component (STIDC)
  - Topology dissemination Component (TDC)
  - Route Selection Component (RSC)
- **MAC Components** – based on CSMA-CA MAC protocols like IEEE 802.11 [Baras08], and on schedules based MAC (USAP) [Baras09]
  - Scheduler
  - MAC

STIDC Benefits and Approach

- STIDC selects a subset of links to be broadcasted
- STIDC is a local pruning method for link selection
- STIDC reduces the broadcast storm problem of TDC
- OLSR uses set cover methods for MPR selection
- There are metrics that capture the stability of the MANET links

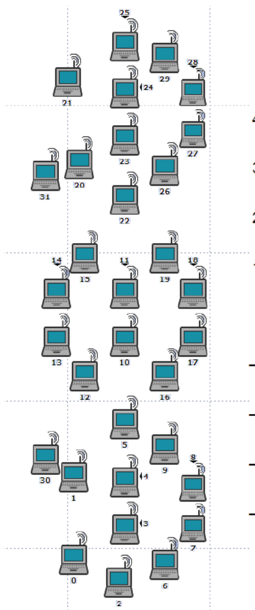
Stable Path Topology Control (SPTC) that accounts for stability metrics in link selection

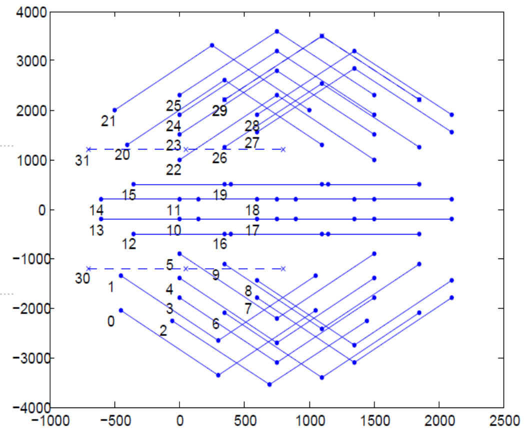
Challenges

- Most local pruning algorithms proposed **do not guarantee QoS optimal paths** for routing.
  - In most cases, they **only guarantee connectivity**

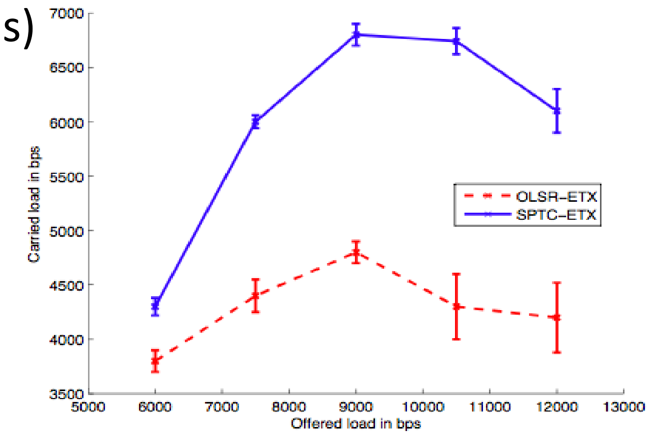
- Non-triviality for preserving QoS optimal paths in local pruning algorithms:
  - **Preserving global properties from only local observations**

3 Platoon Mobility Scenario





Long connection from 20 to 0 (platoon heads)



	OLSR-ETX	SPTC-ETX
Saturation CL	~ 2 Mbps	~ 2 Mbps
TC message rate	923 kbps	890 kbps

Type	Connection	Offered-load
Intra-platoon	(1,3),(2,9),(4,6),(7,5),(20, 29), (14,17),(16,11),(17,18),(19,12), (21,22),(23,27),(23,28)	12 kpbs
Inter-platoon	(1,18) (20,11),(20,0) (10,1),(21,10)	2.4 kbps 6 kbps 12 kbps

STIDC Design Summary

- We have developed a local pruning mechanism that ensures that globally optimal routing paths are preserved in the pruned graph
- SPTC-ETX, when compared to OLSR-ETX
  - **Carries more traffic** – stable paths are long-lived ➔ long-lived sessions
  - **Fewer topology changes** – stable links are long-lived ➔ stable routing graph

Component Based Networking: Network MBSE for MANET

J. Baras UMD), V. Tabatabaee (Broadcom), K. Somasundaram (Qualcomm) and M. Austin (UMD)

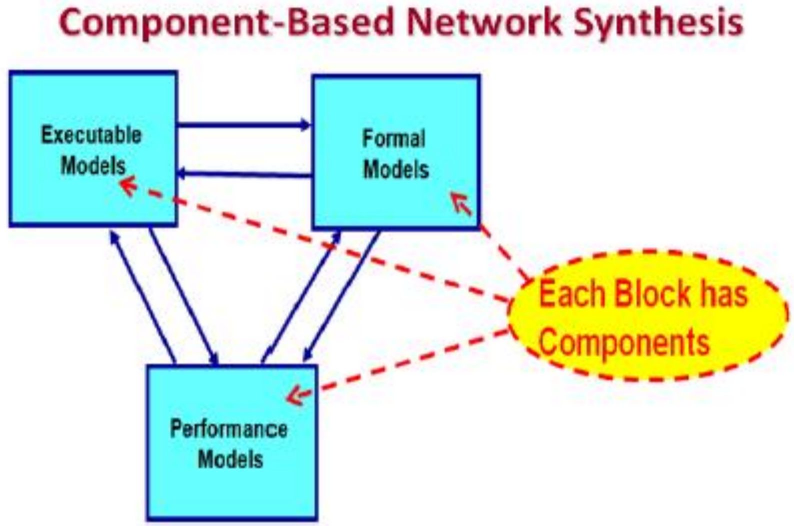
The Challenge & Need:
Design DoD and Commercial MANET Adaptive to Dynamic Mission Requirements

Dynamic Interconnection and Interoperability

- Broadband wireless nets capable for multiple dynamic interface points
- Any node can serve as interface/gateway

Fig.1: Intelligent Wireless Multi-Nets

Fig.2: Component Based Networking



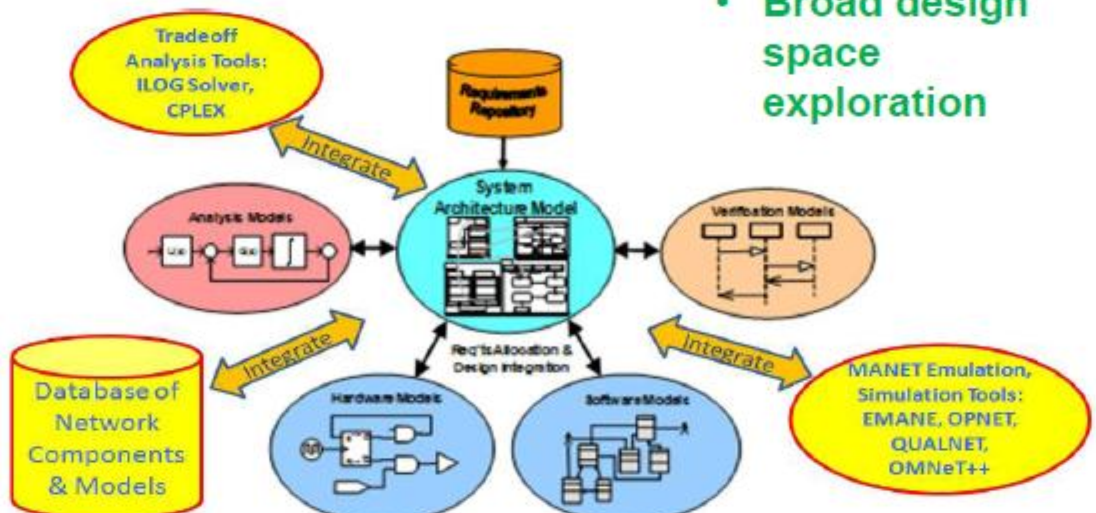


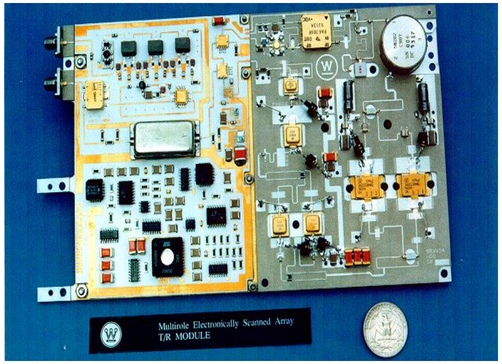
Fig. 3: Network MBSE Toolset : integrating **SysML** Architecture Model with DB of network models, emulation-simulation models, tradeoff tools

BENEFITS

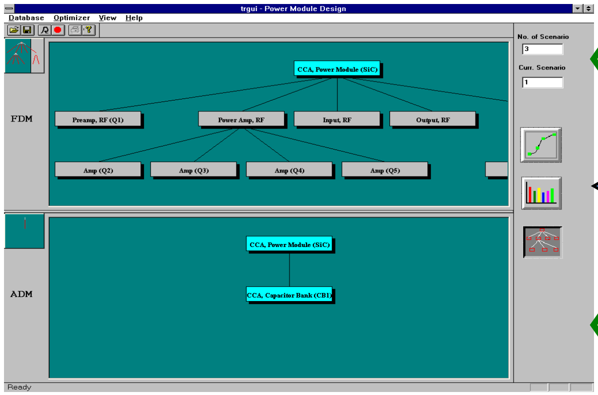
- Reduced MANET cost and fielding time
- Modularity and re-use
- Increased agility in designing, modifying and fielding new MANET
- Broad design space exploration

The Institute for Systems Research

Integrated Product and Process Design of T/R Modules



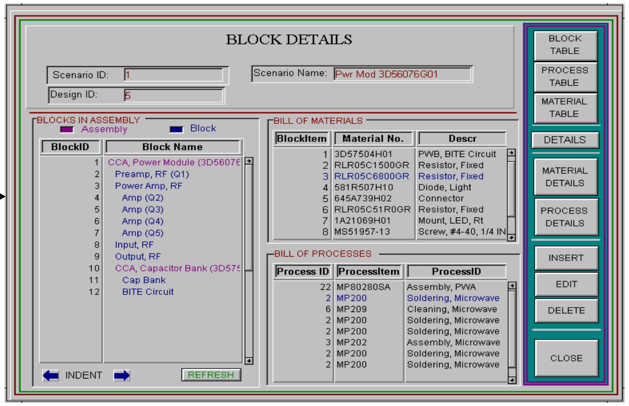
PROBLEM **Integrate Electronic and Mechanical Design**
information interchange among tools used by designers
Identify alternative components
integration with part catalogs, corporate databases
Help generate and evaluate alternative designs
estimate cost, manufacturing time, reliability, etc.evaluate tradeoffs
Help generate process plans
process parameters, time estimates, etc.



Assembly Data Model Process Designer View

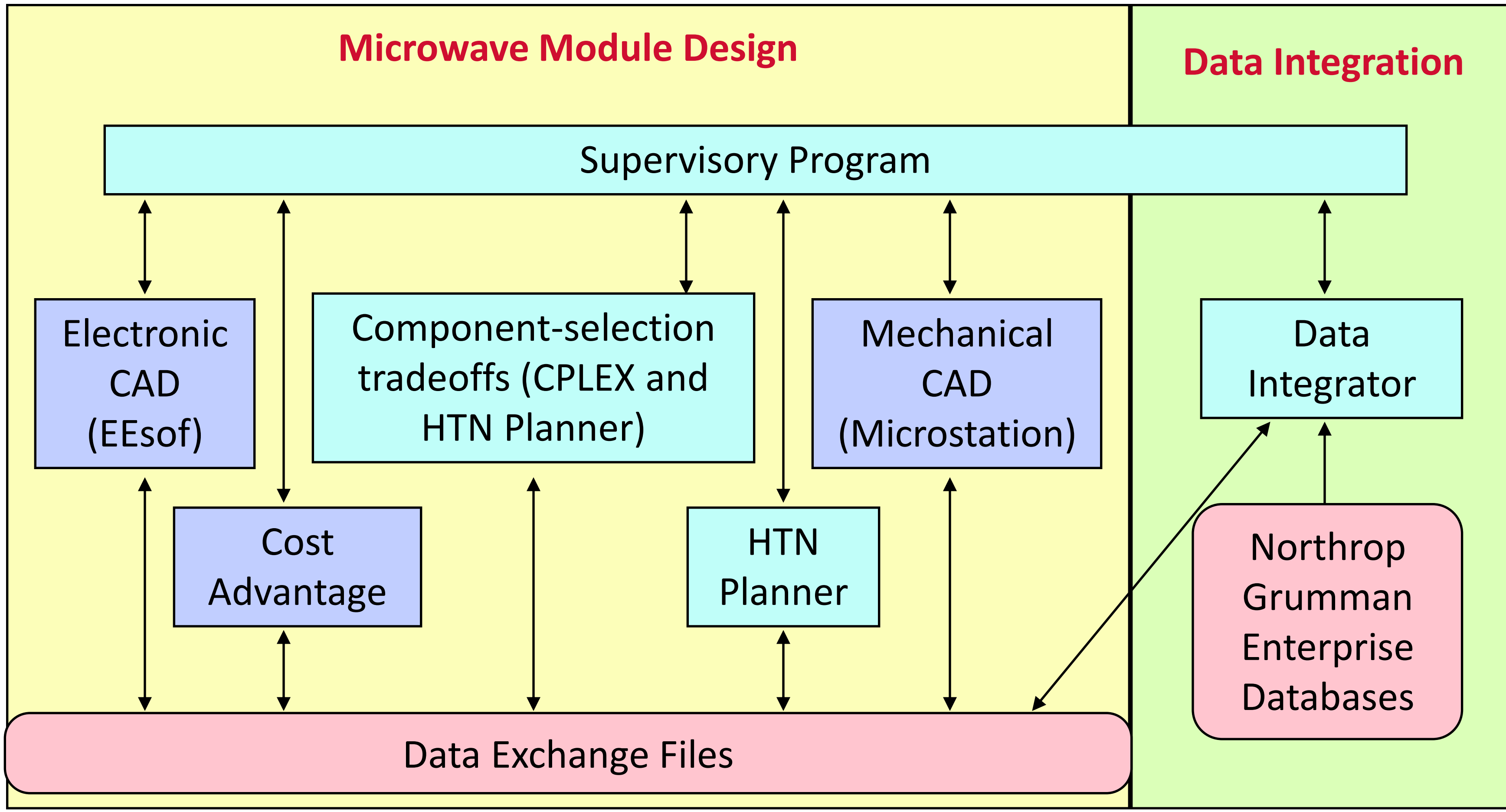
Entity-Relation Diagrams

Functional Data Model Product Designer View



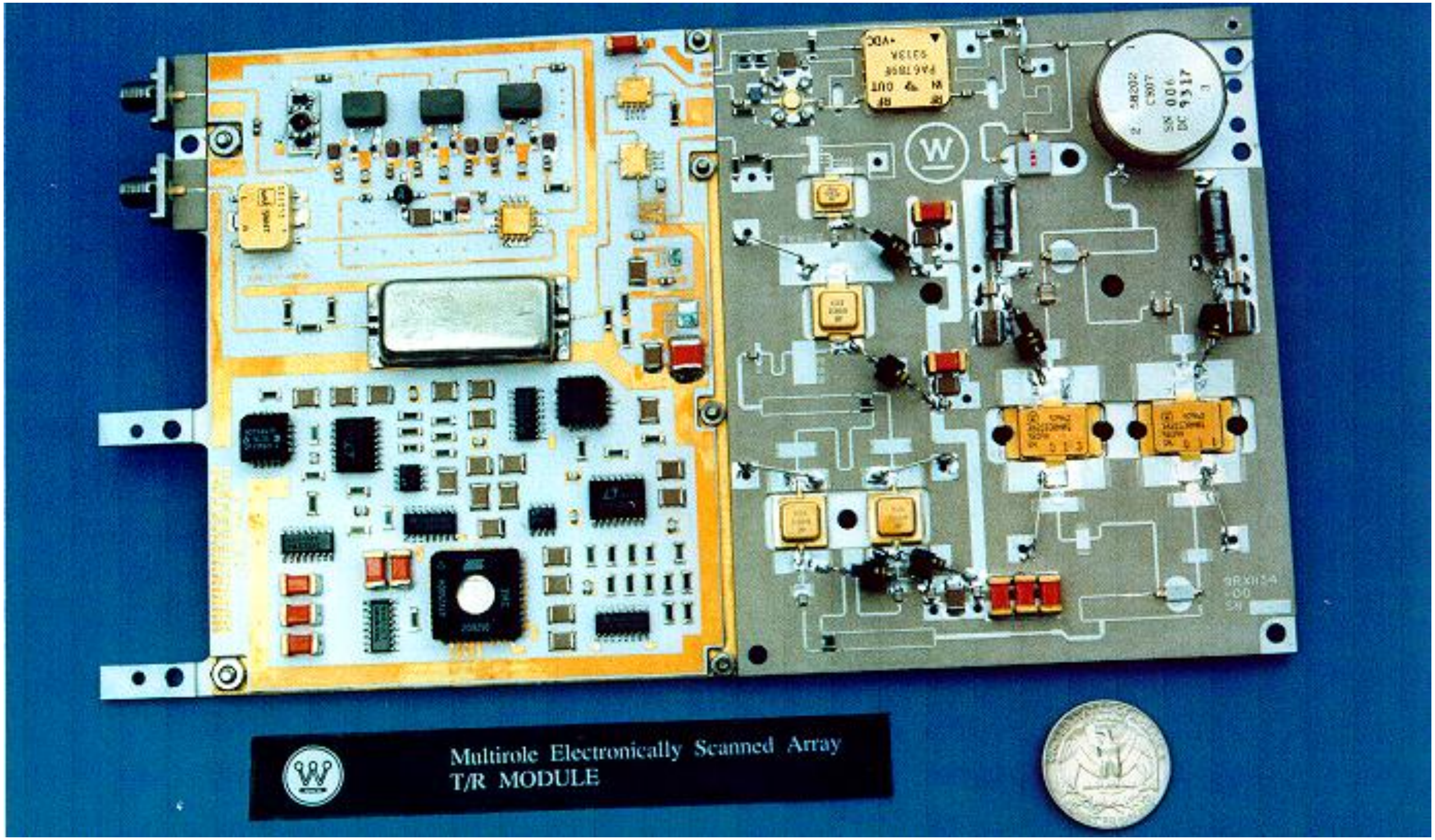
SOLUTION **Object-Relational Databases and Middleware to integrate heterogeneous distributed data sources:**
multi-vendor DB, text, data, CAD drawings, flat, relational, object DBs
Entity-Relation Diagrams to provide multiple expert views of the data and integrate product and process design phases into a single system environment
Hierarchical Task Network planning to explore alternate options at each level of the product: parts and material, processes, functions assemblies
Multicriteria Optimization for trade-offs: cost, quality, manufacturability, ...

IPPD System Architecture

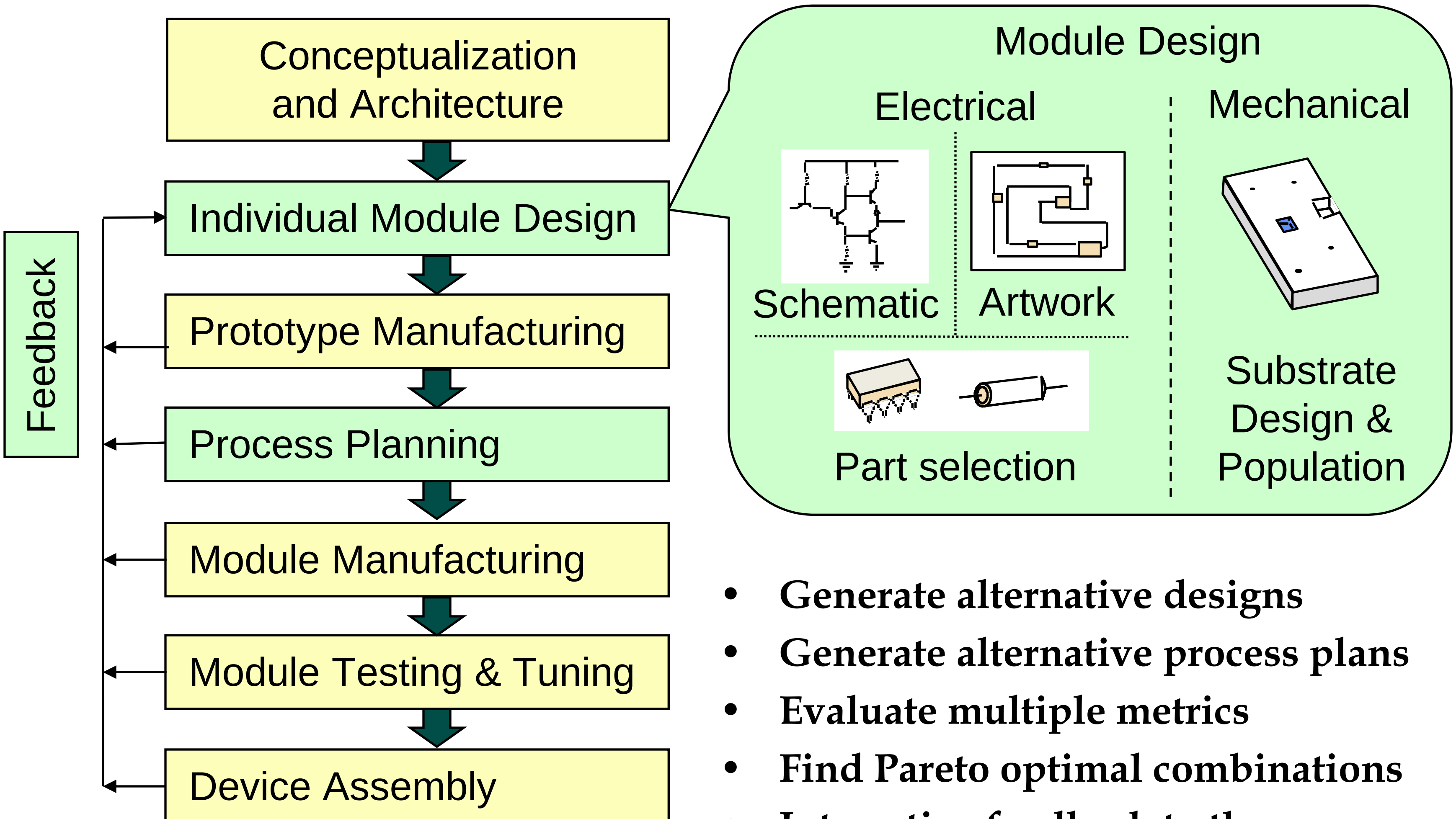


Microwave Transmit/Receive Modules

- 1-20 GHz frequency range (radars, satellite communications, etc.)
- Difficult and expensive to design and manufacture



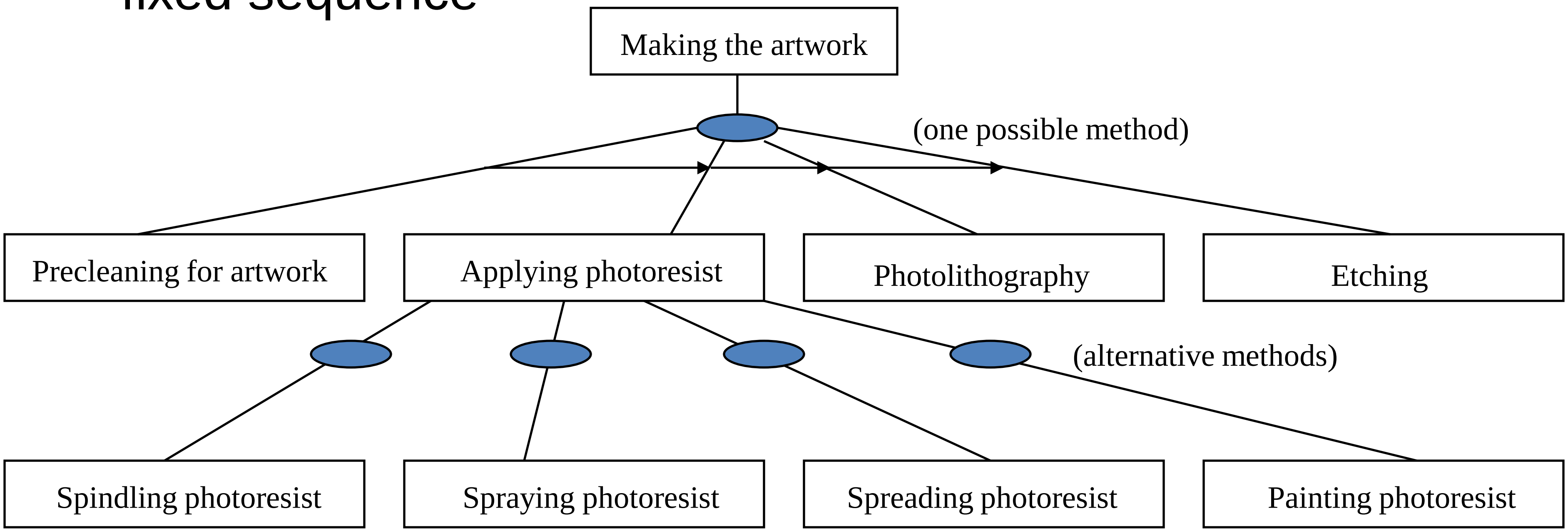
IPPD for Microwave Module Design and Manufacturing



- **Generate alternative designs**
- **Generate alternative process plans**
- **Evaluate multiple metrics**
- **Find Pareto optimal combinations**
- **Interactive feedback to the user**

Electronic Manufacturing Processes

- Process structure is more complicated - but it decomposes naturally into tasks and subtasks that are performed in a fixed sequence



- **Develop plan details depending on the details of the design**

The Metrics

- We evaluate the design with respect to five metrics:
  - **Cost defined as the sum of material costs, process runtime costs, and process setup costs**
  - **Manufacturing yield, defined as the product of process yields and part yields**
  - **Supplier lead time, defined as the maximum of the delivery lead times of the selected suppliers**
  - **Total number of suppliers selected**
  - **Quantity discounts associated with placing more orders with the same supplier**

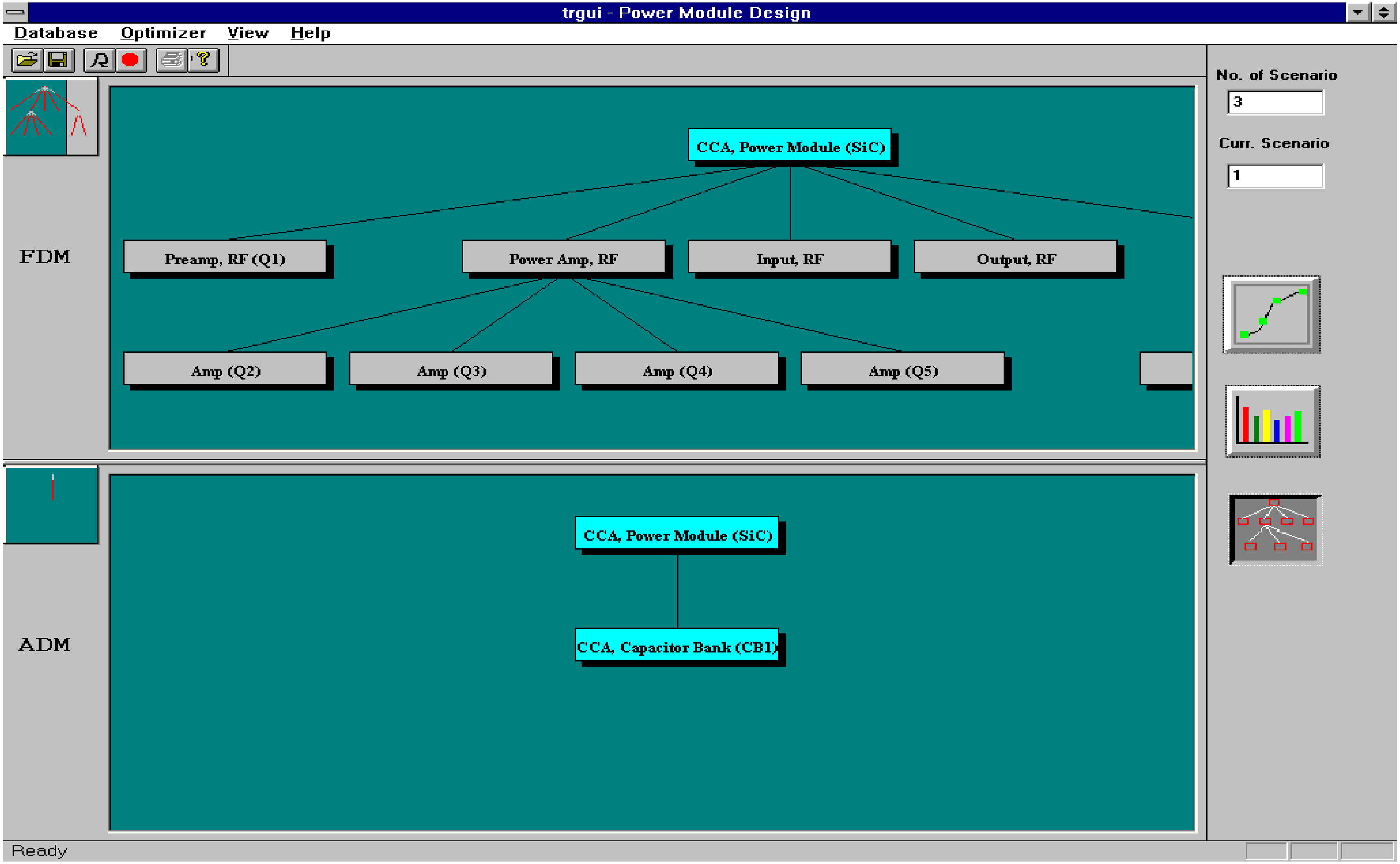
Multiple Object Views of Data

- Functional Data Model (FDM)
  - Functional block elements (FBE)
  - Functional bill of material (FBOM)
  - Functional list of processes (FLOP)

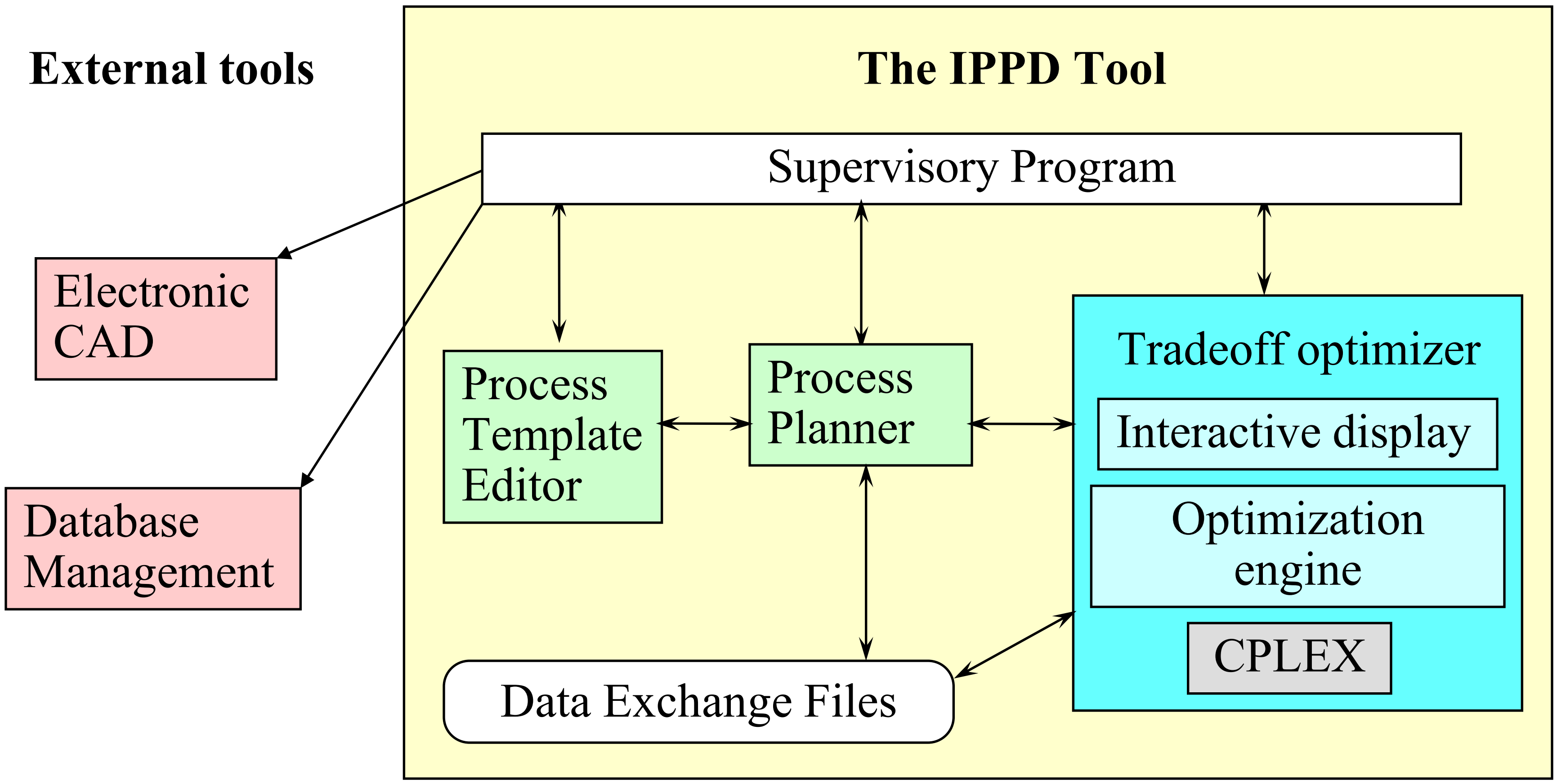
- Assembly Data Model (ADM)
  - Each assembly is a manufacturable unit
  - Map FDM to ADM
  - Bill of materials (BOM) for each assembly

- List of processes (LOP)

FDM and ADM Implementation

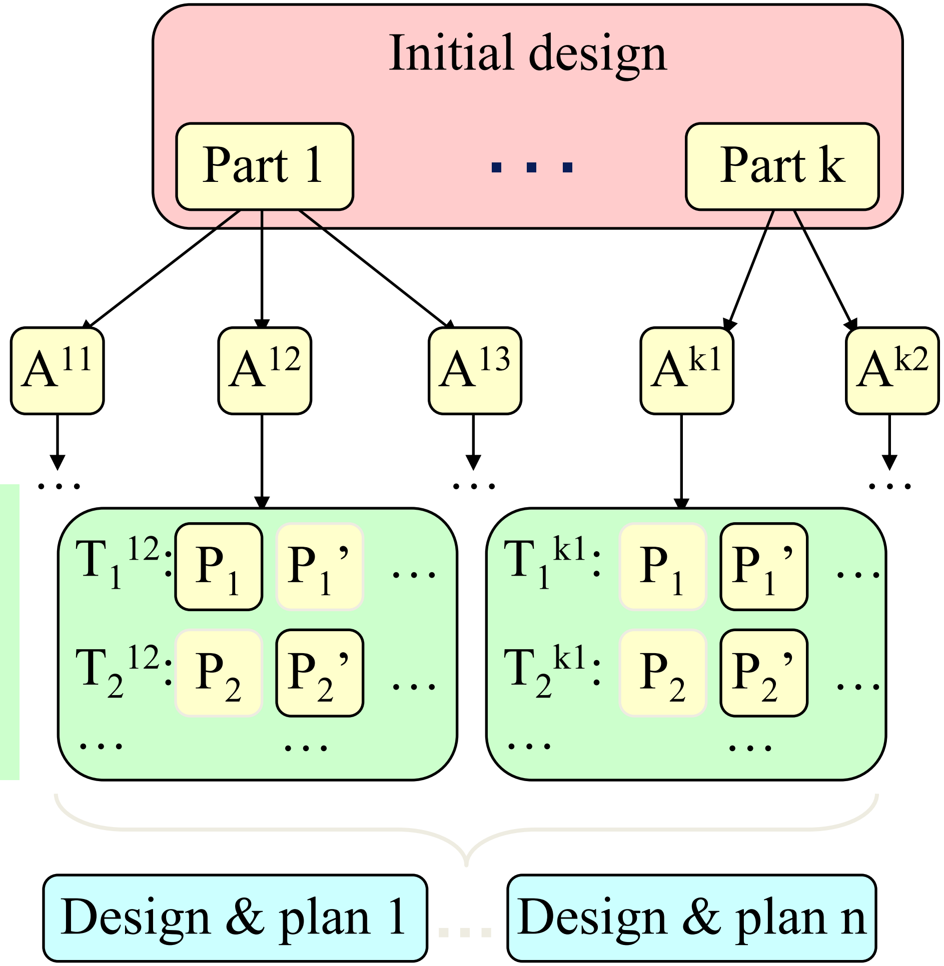


System Architecture



Hierarchical Task Network Planning for ProcessPlanning

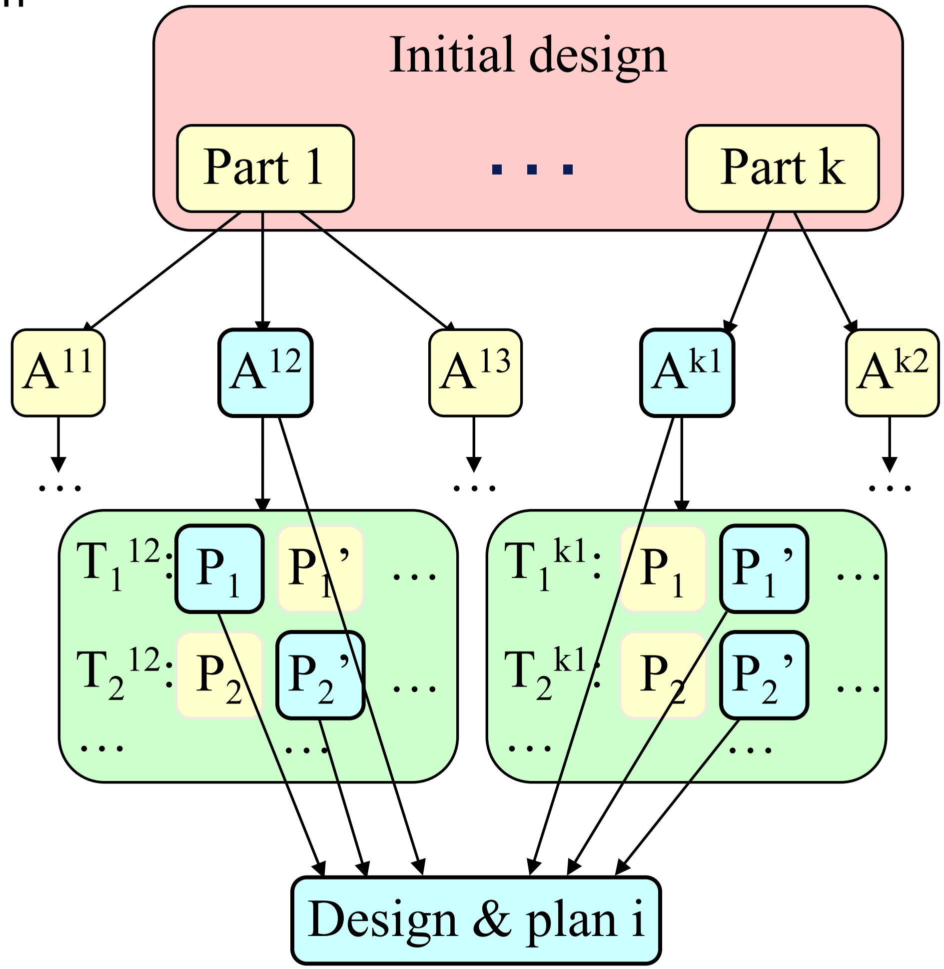
- For use in a factory setting
  - Must be easy for non-researchers to understand and maintain
- Input: alternatives for each part in the design
- Output: plan fragments for each alternative part
  - alternative processes for each task
- Don't yet want entire plans
  - Later, will use tradeoff analysis to combine the plan fragments into plans



Tradeoff Analysis

- For each part, for each task to perform on the part, the process planner finds
  - All applicable alternative processes
  - Setup time, run time, & yield for each

- Tradeoff analysis
  - Choose a combination of alternative parts
  - For each chosen part, choose a sequence of processes
- Find collections of choices that have Pareto optimal values for
  - Total cost
  - Total time
  - Total number of suppliers
  - Yield



Multicriteria Optimization Formulation

Decision Variables

$x_i = \begin{cases} 1 \text{ if node } i \in V \text{ is selected} \\ 0 \text{ otherwise} \end{cases}$

$x_{ip} = \begin{cases} 1 \text{ if arc } (i,p) \in E \text{ is selected} \\ 0 \text{ otherwise} \end{cases}$

$y_p = \begin{cases} 1 \text{ if process } p \in P \text{ is selected} \\ 0 \text{ otherwise} \end{cases}$

For each component, choose exactly one among its alternatives

$$\sum_{j \in \upsilon_k} x_j = 1 \quad k \in \upsilon$$

For each process related to a component, choose exactly one among its alternative

$$\sum_{p \in P_{ji}} x_{pj} = x_j \quad \forall p, \quad i \in Pj$$

Selection of processes

$$y_p \geq x_{pj} \quad \forall p, \quad j$$

Selection of suppliers

$$w_i \geq x_j \quad \forall i \in S, \; j \in S_i$$

Integrality

$$y_p, \; x_{pj}, \; w_s \in \{0, \; 1\} \quad \forall j, \; p, \; s$$

Material cost:

$$C_m = \sum_i n_i c_i x_i$$

Runtime cost:

$$C_r = \ell \sum_{p,j} t_{pj} x_{pj}$$

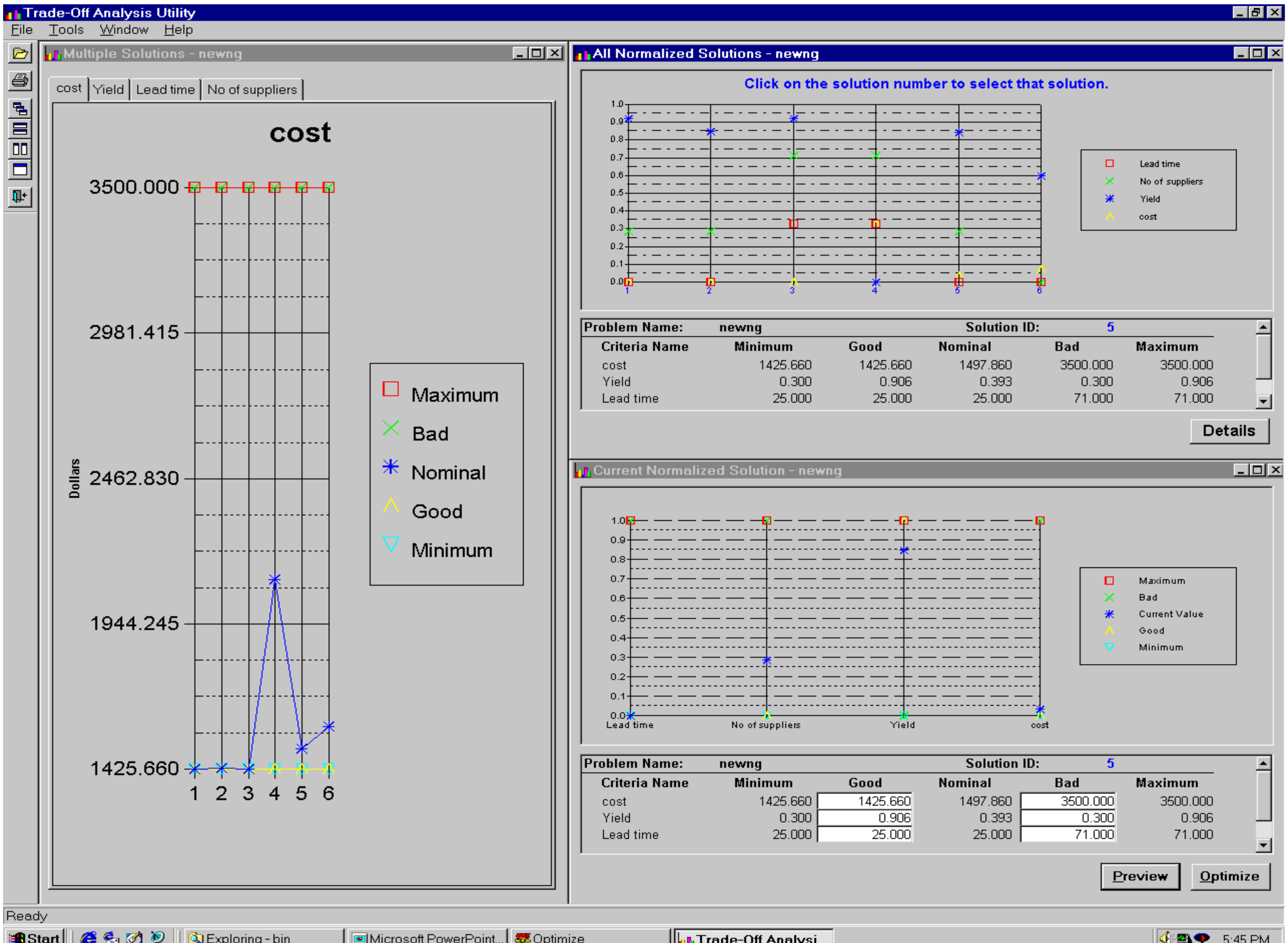
Setup cost:

$$C_p = \frac{\ell}{b} \sum_p t_p y_p$$

Total cost:

$$C = C_m + C_r + C_p$$

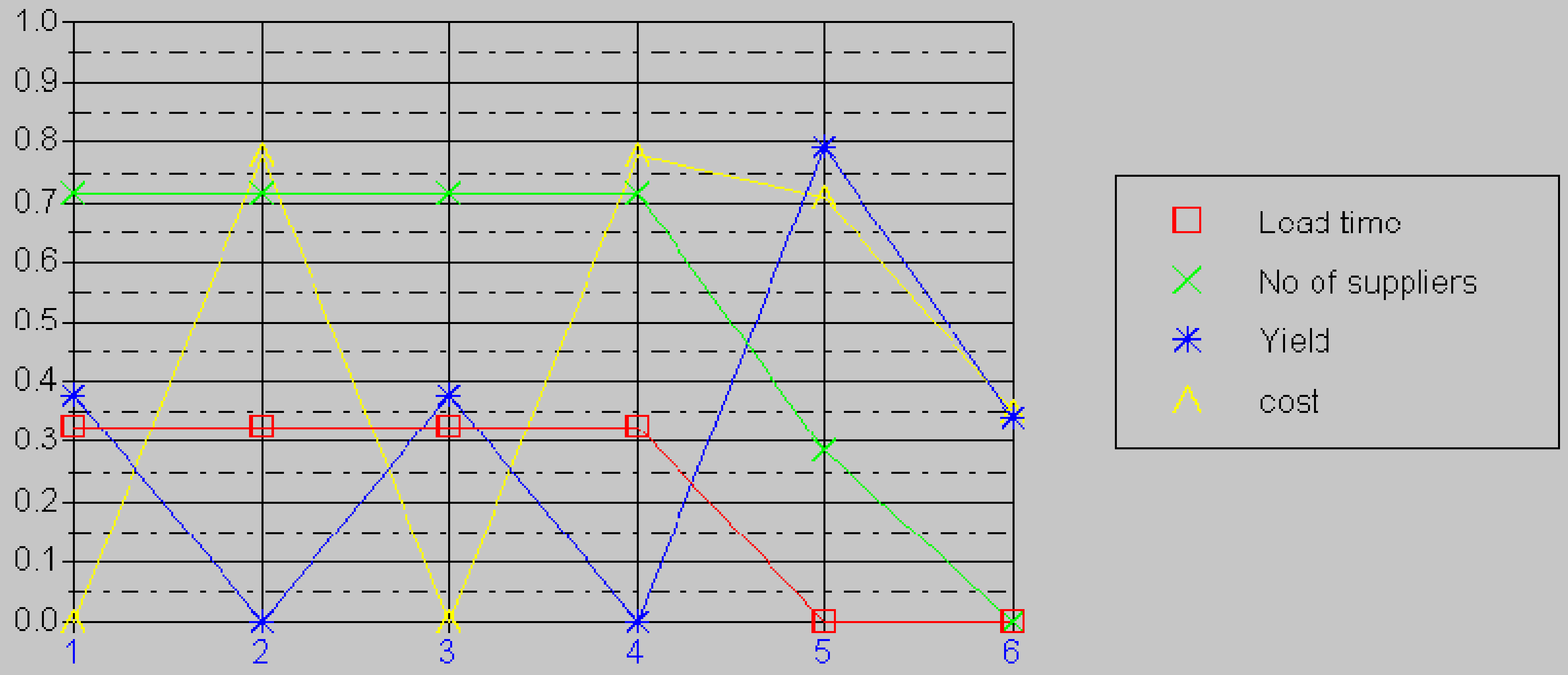
Tradeoff Analysis via Multicriteria Optimization



Tradeoff Analysis via Multicriteria Optimization (cont.)

Initial Normalized Solutions - p1a

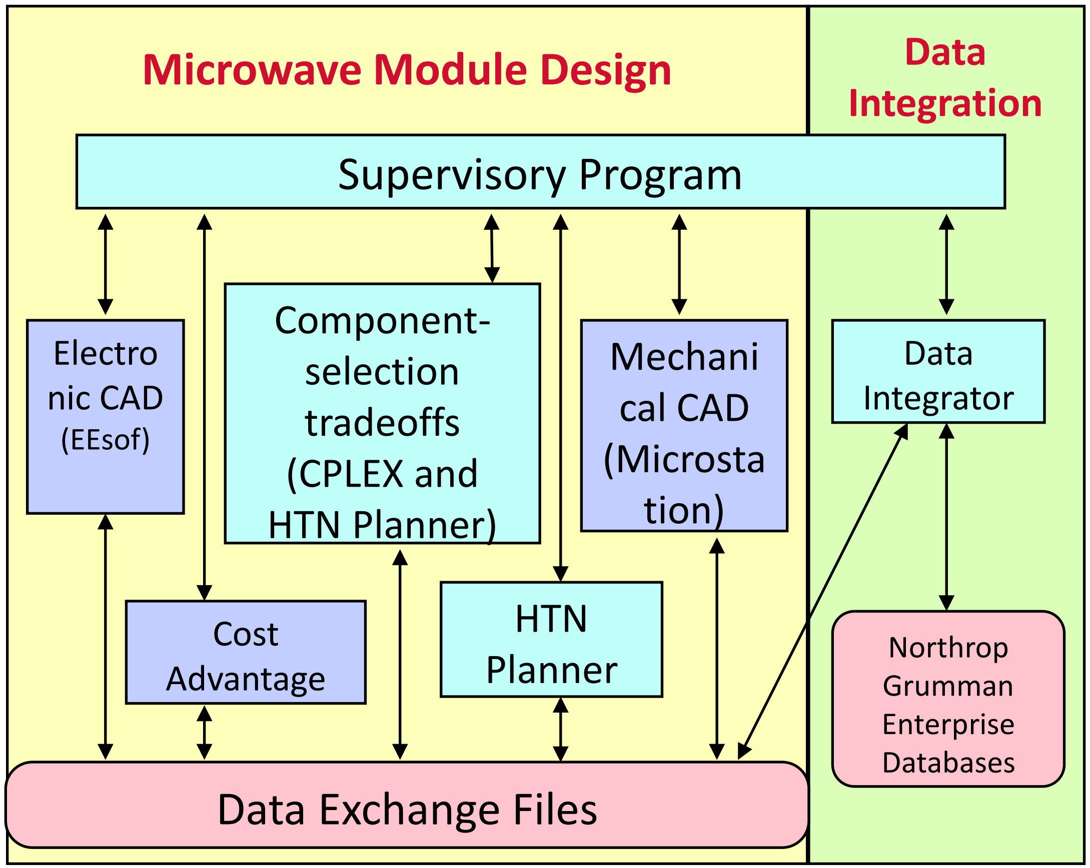
Click on the solution number to select that solution.



The file containing the description of this problem cannot be found. Ensure there is a file called P1A.TXT in the same directory as the LP and NF files.

Current Work: From IPPD to Health CARE

IPPD System Architecture



ISR Innovations:

- **Object-oriented modeling** of parts and processes (system components)
- Respect the **need for integration** with company tools (some proprietary)
- IPPD environment independent from company data warehouse contents
- Integration of **many discipline design** tools
- **Multiple views** of data: Functional data model, Assembly data model
- **Object-relational** databases
- **Hierarchical task network planning** for process plan generation
- Tradeoff analysis via **multi-criteria optimization** involving **numerical and Boolean** variables

Process Models and efficiency Analysis of critical hospital systems (ICU):

Process Models, Equipment Models, Cost Models, Tradeoffs and Efficiency Improvement

MBSE Approach to Energy Efficient Buildings

Buildings Design Energy and Economic Analysis

Windows and Lighting

HVAC

Domestic/International Policies, Regulation, Standards, Markets

Demonstrations, Benchmarking, Operations and Maintenance

Natural Ventilation, Indoor Environment

Networks, Communications, Performance Database

Sensors, Controls, Performance Metrics

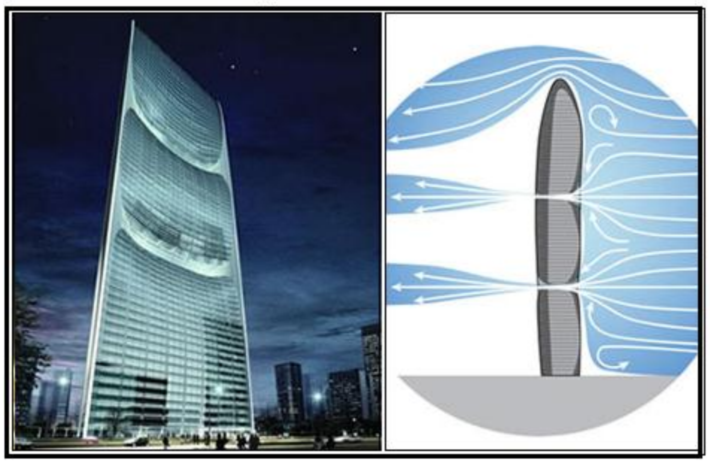
Power Delivery and Demand Response

Building Materials, Misc. Equipment

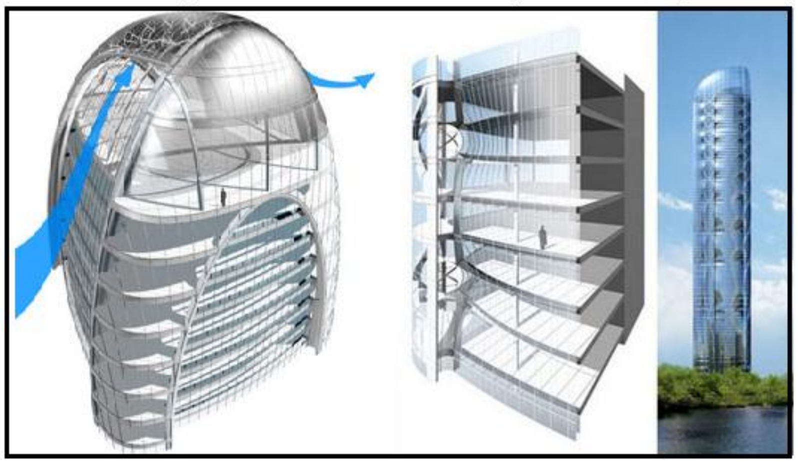
Cyber-Physical Building Systems

- **Research focus:** Platform-Based Design for Building-Integrated Energy Systems.

Pearl River Tower Complex



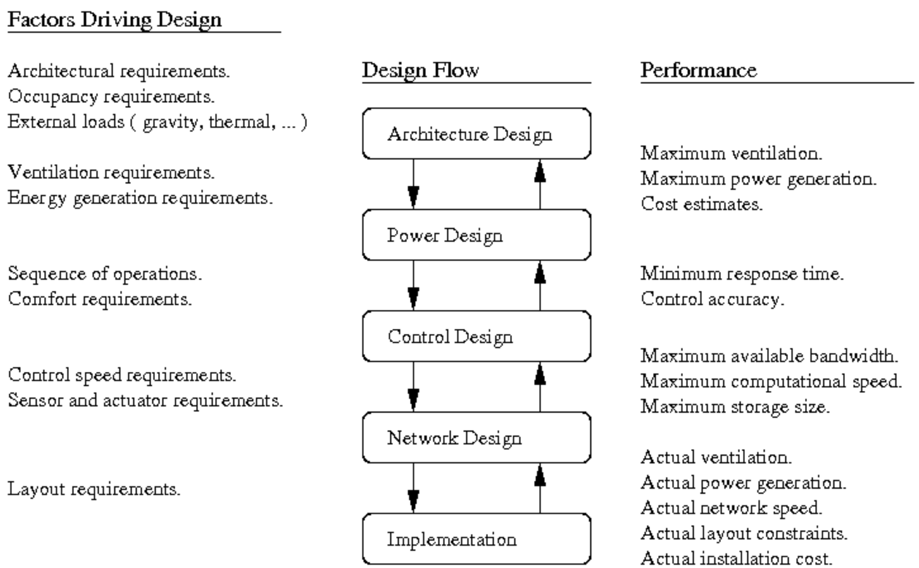
Green Technology Tower –– Architectural Proposal for Chicago

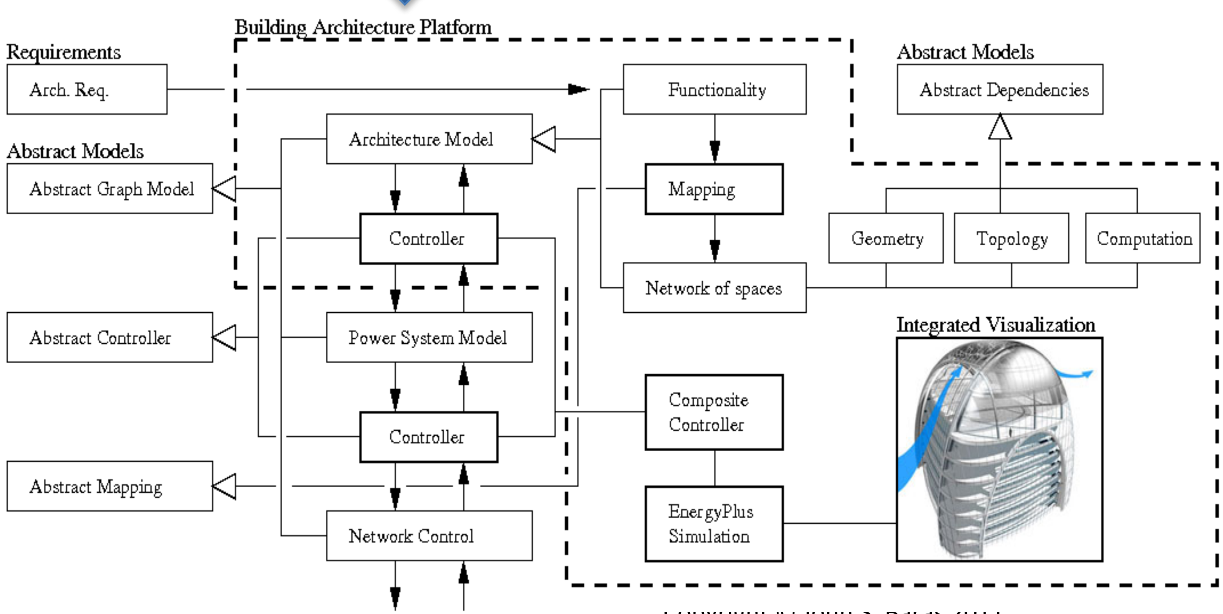


Cyber-Physical Building Systems Design

- Design Platform Stack

Research: Design of a scalable and extensible platform infrastructure





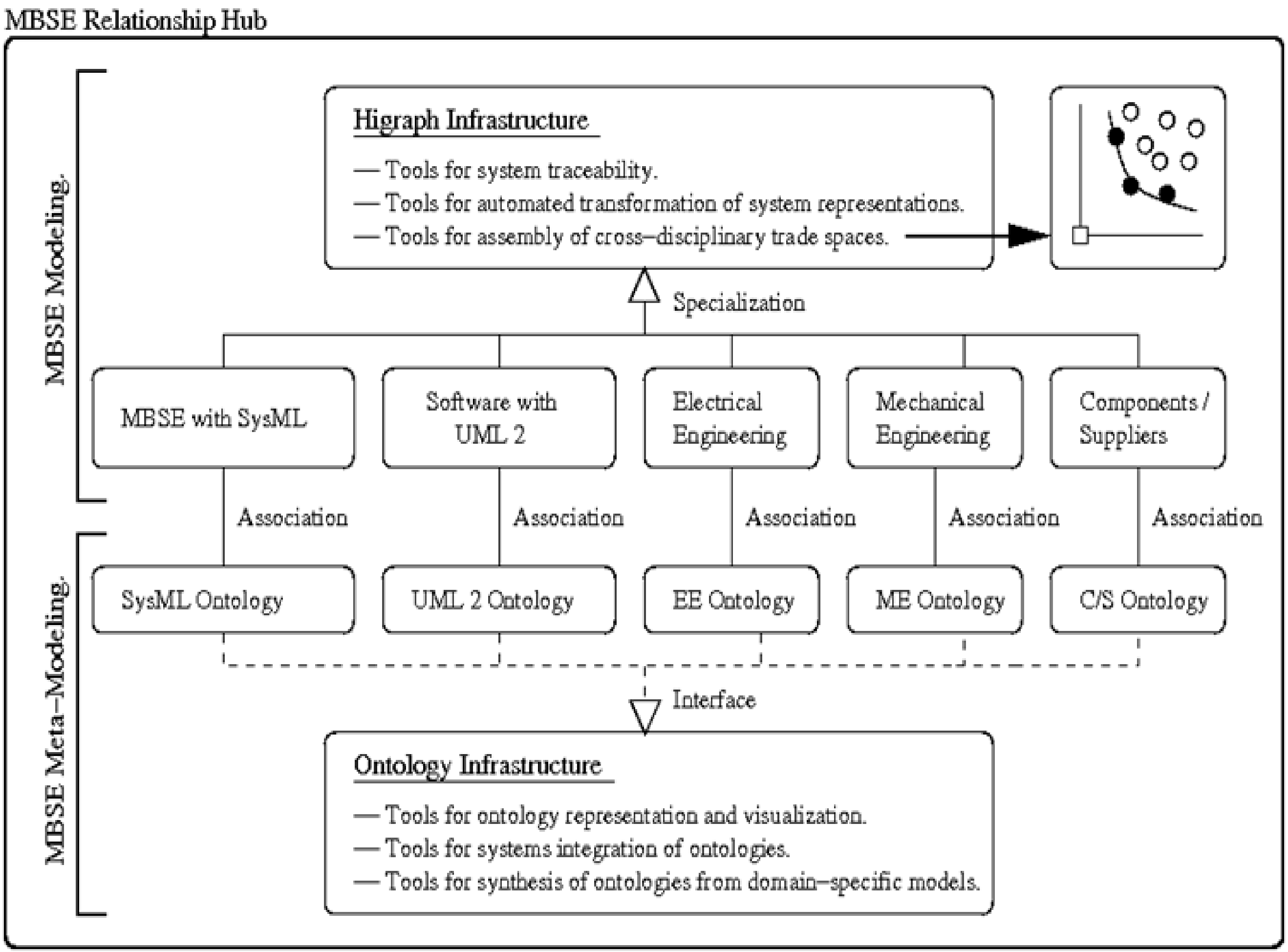
MBSE RELATIONSHIP HUB

Create a ***scalable and extensible MBSE Relationship Hub*** to:

(1) Support solutions to the MBSE challenges;

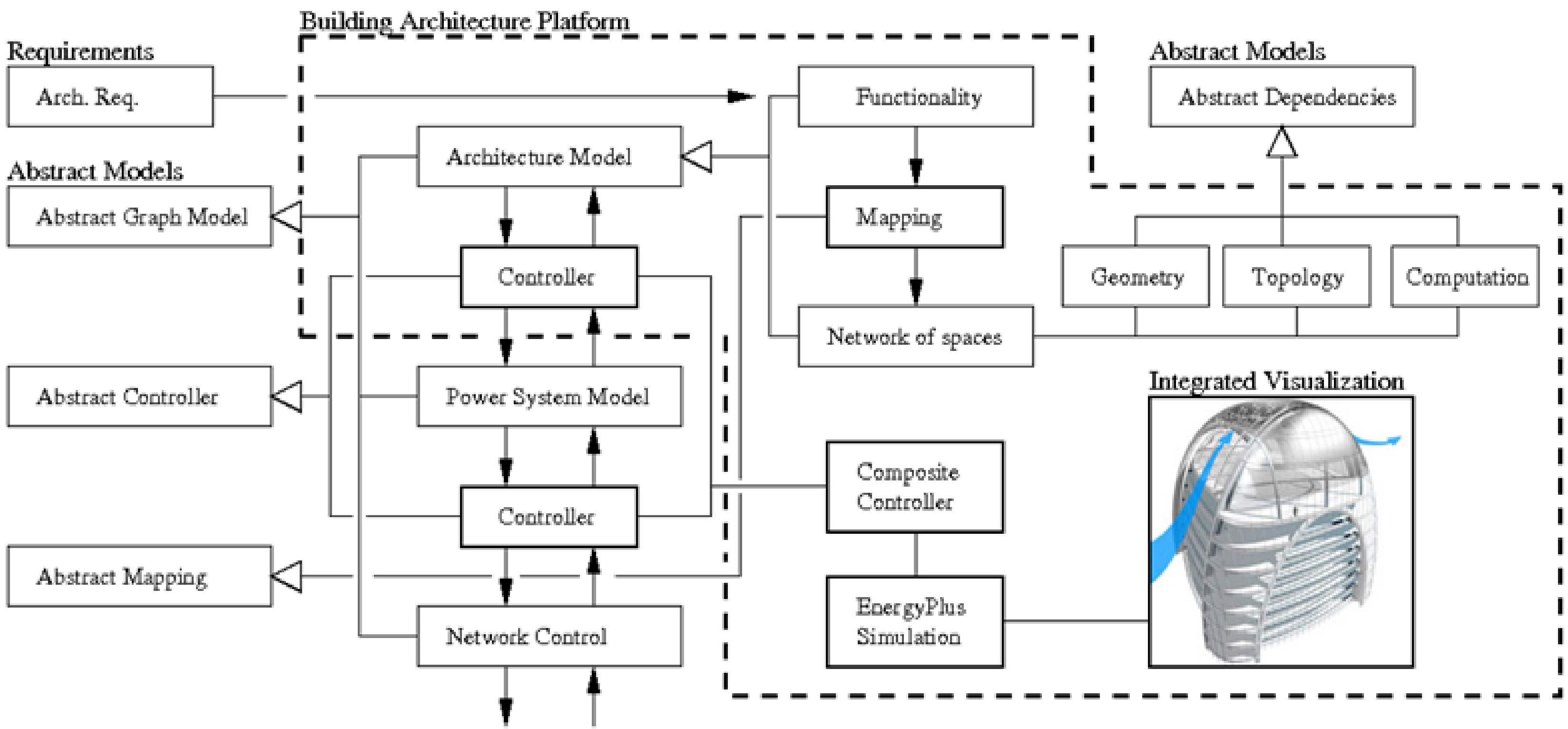
(2) Allow systems engineers to understand and appreciate the extent design strategies can be applied;

(3) Help engineers evaluate and balance competing design criteria.



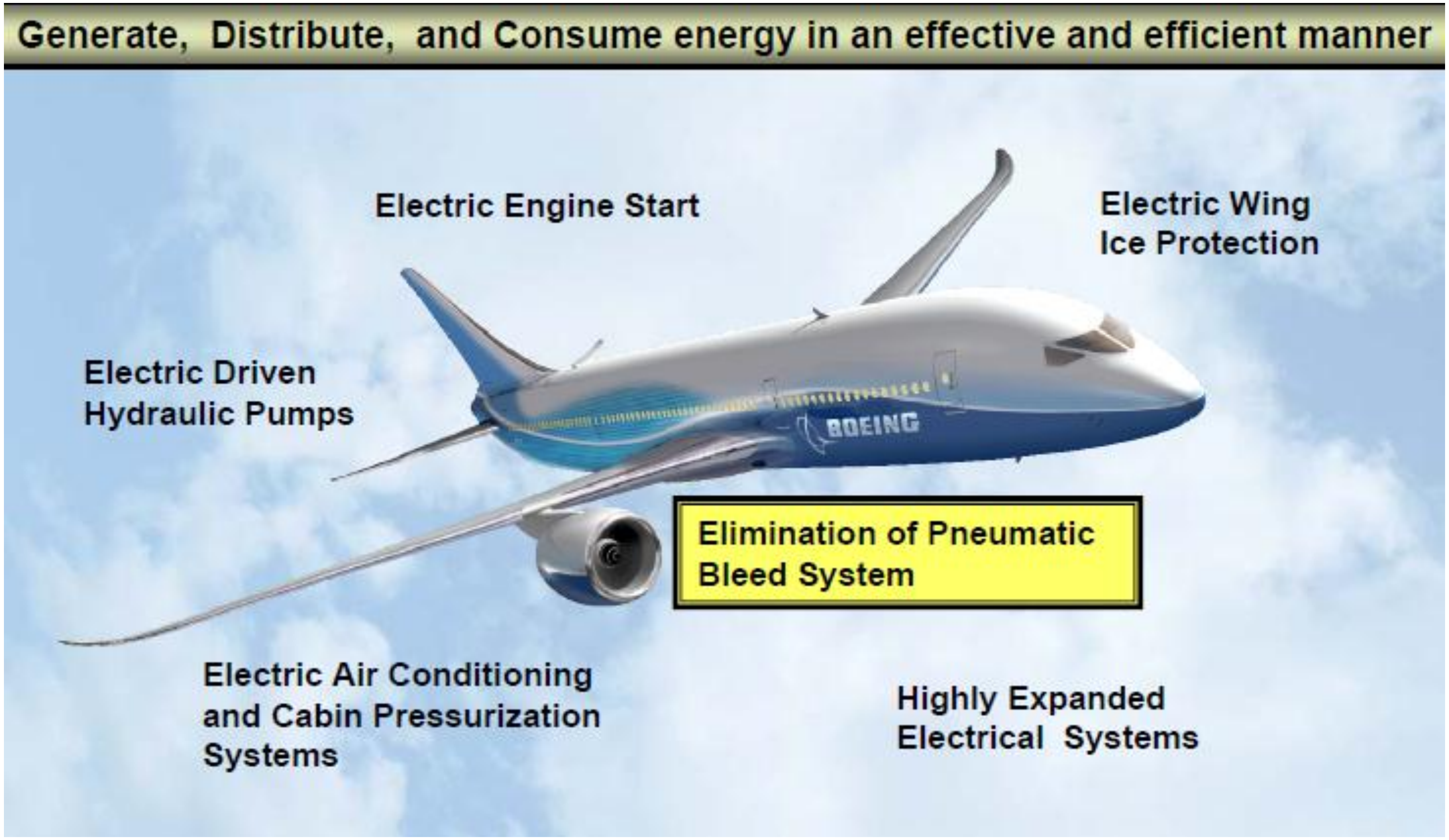
DESIGN PLATFORMS FOR SE
BUILDING-INTEGRATED ENERGY SYSTEMS

Extensible framework for assembly of (model, controller, simulation, viewpoint) process networks and communication for platform-based design of building-integrated energy systems



MORE ELECTRIC AIRCRAFT (MEA)

787 MEA ARCHITECTURE



The Institute for Systems Research

787: CLEANER, QUIETER, MORE EFFICIENT

UNIVERSITY OF MARYLAND 1856

The 787 Dreamliner delivers:

*Relative to the 767

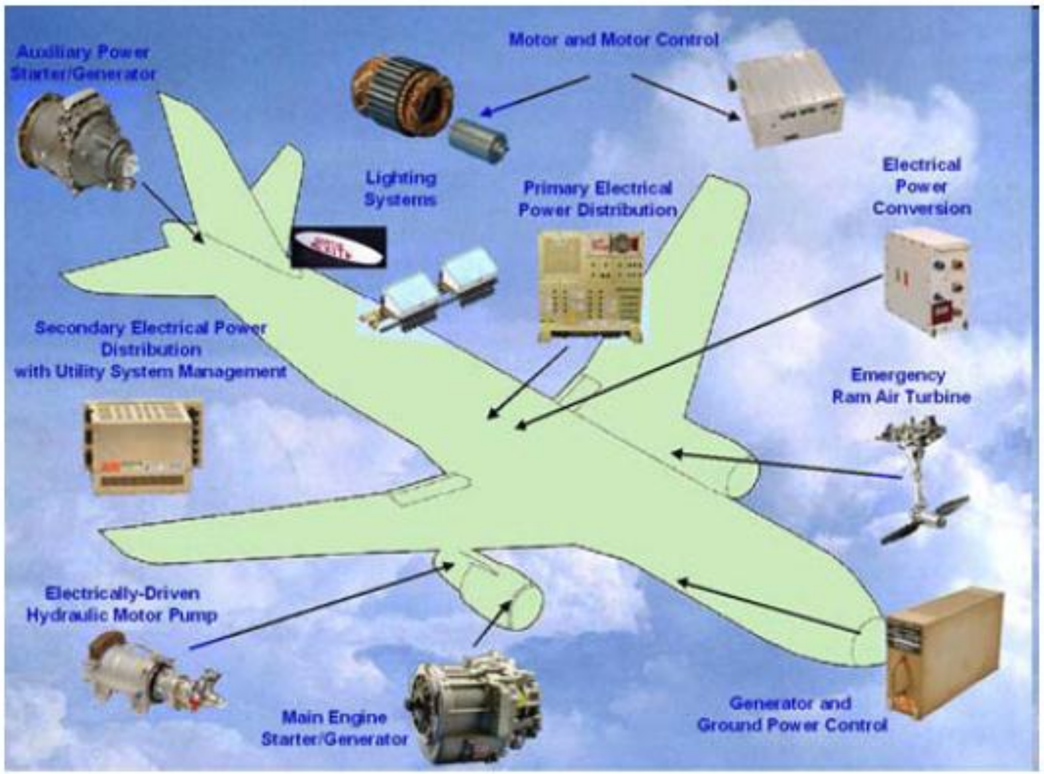
- 20%* reduction in fuel and CO_2
- 28% below 2008 industry limits for NOx
- 60%* smaller noise foot print

MuSyC Avionics Design Challenge

- Primary power distribution of an electric power system for next generation aircraft -- part of the MuSyC avionics challenge problem
- Typically consists of a combination of generators, switches, and loads
- Primary power generation elements include batteries, auxiliary power units (APU), generators connected to the air-craft engine, and a ram air turbine (RAT) used for emergency power
- Electrical power is distributed via one or more buses and connection of generators to loads is routed by way of a series of electronic control switches (contactors)
- Primary electrical loads include communications and computing systems, electrically-driven actuation systems (including electro-hydraulic systems), anti-ice and/or de-ice systems, and lighting systems.

- **Requirements categories**: ***safety***, ***performance***, ***reliability***, ….., subject to priorities, component capabilities and schedules, ….

MBSE for Fault Tolerant Vehicle Management Systems (Electrical, Hydraulic, etc.)



[Image: hamiltonsunstrand.com]

Goal: Synthesize logic to switch between generators and loads on-demand and to handle faults so as to stay within safe operating envelope

Joint with UTRC

The Institute for Systems Research

MuSyC Avionics Design Challenge

1

- **Fig. 1**: Physical architecture of the modern aircraft power system
- **Fig. 2** : Requirements analysis and allocation
- **Fig. 3&4**: System behavior using SysML and Modelica
- **Fig. 5&6**: System structure and constraints using SysML diagrams

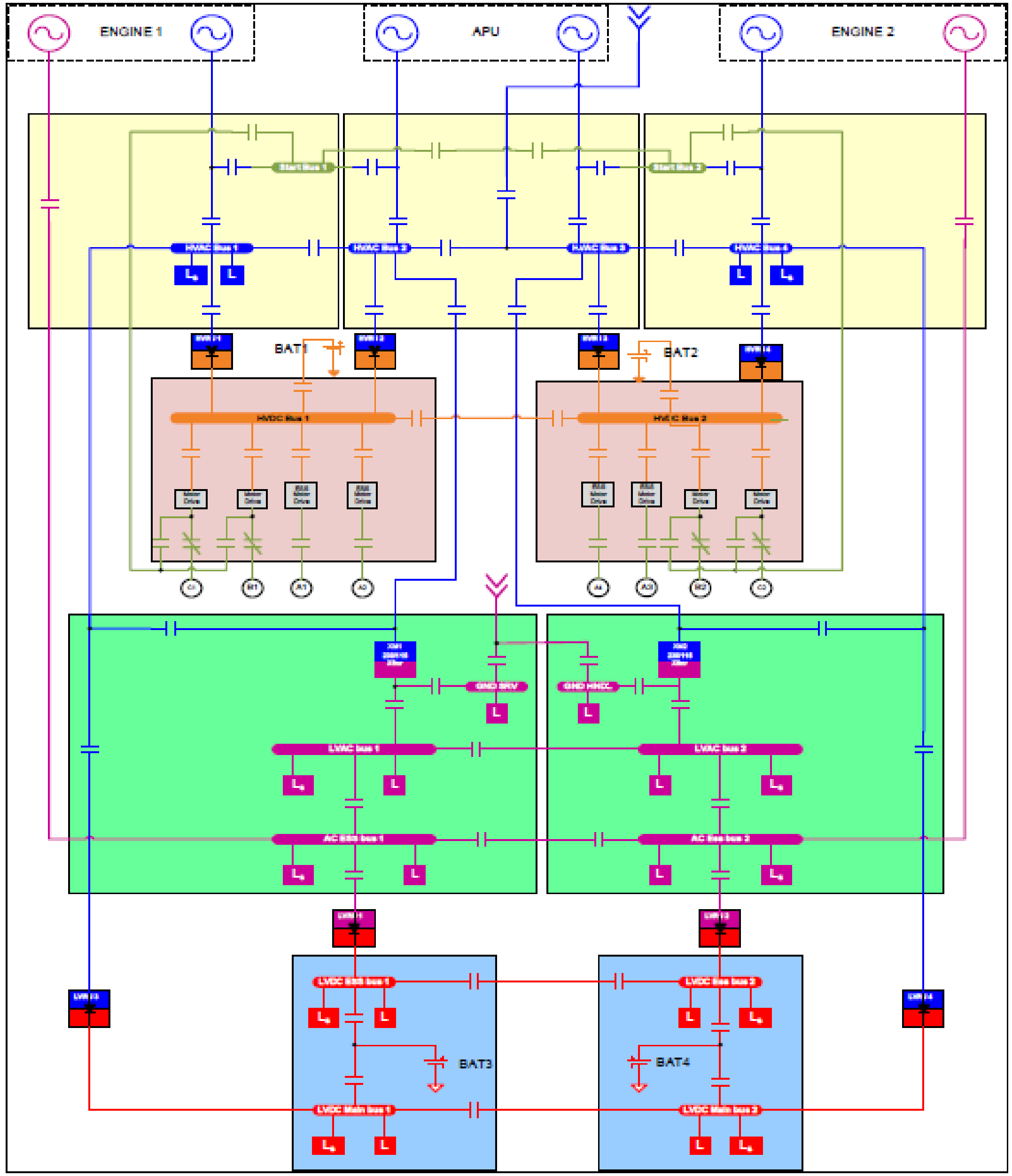
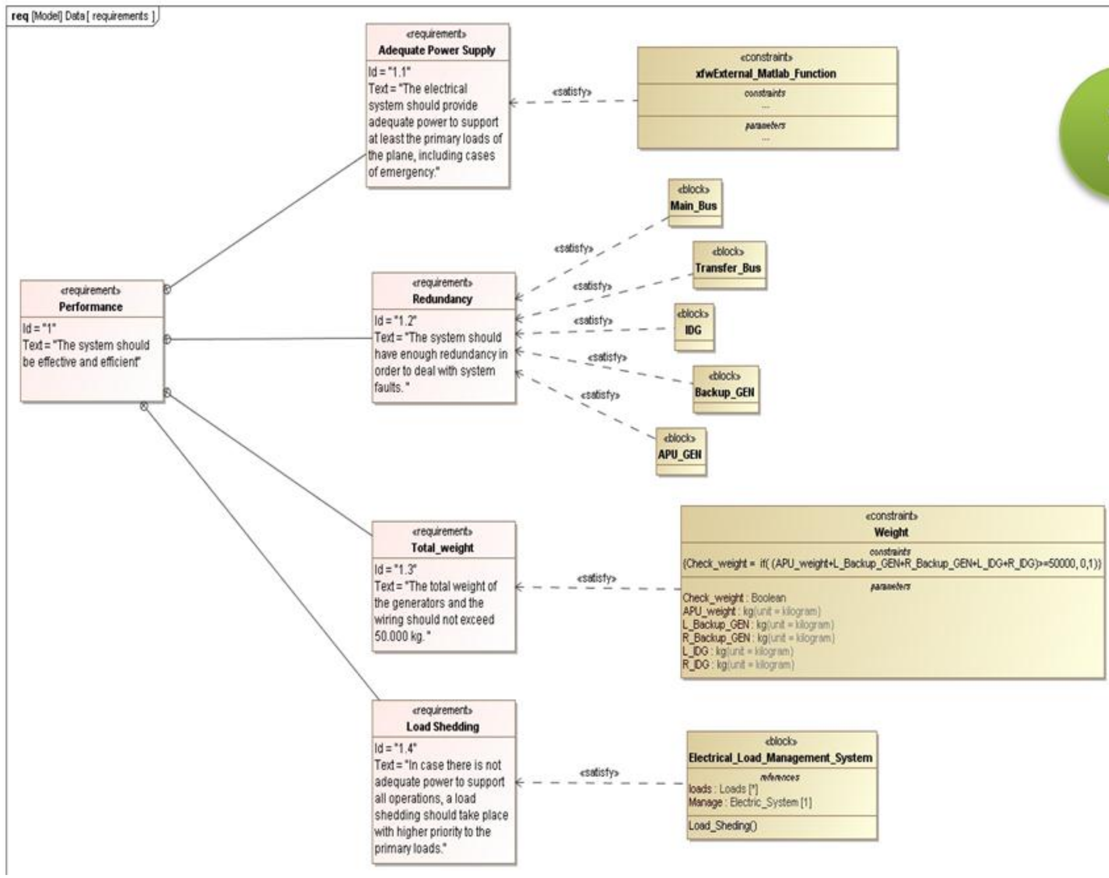


Figure 1: Single line diagram of an electric power system adapted from Honeywell Patent US 7,439,634 B2. Figure courtesy of Rich Poisson, Hamilton-Sundstrand.

MuSyC Avionics Design Challenge (cont.)

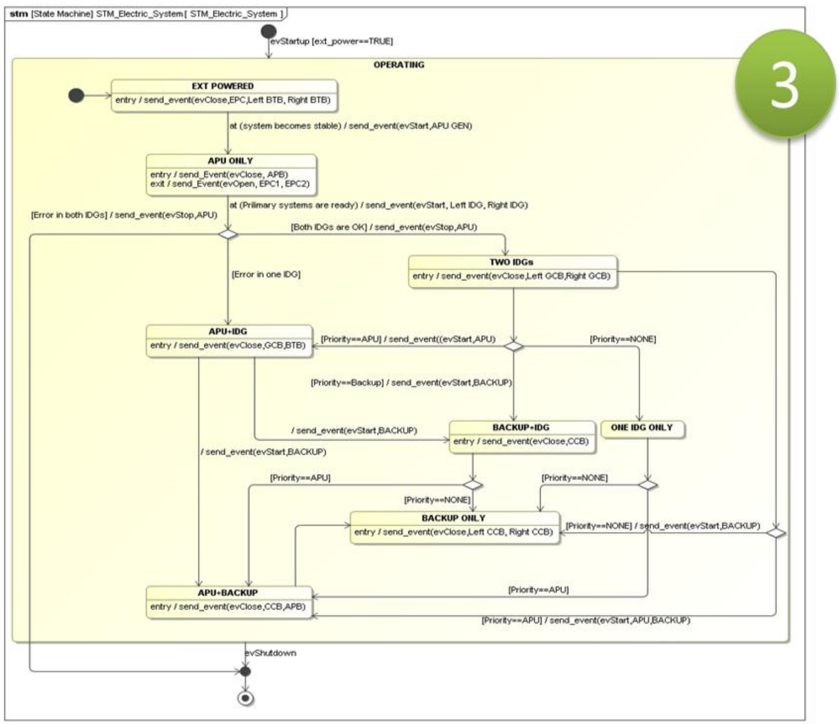
Requirements

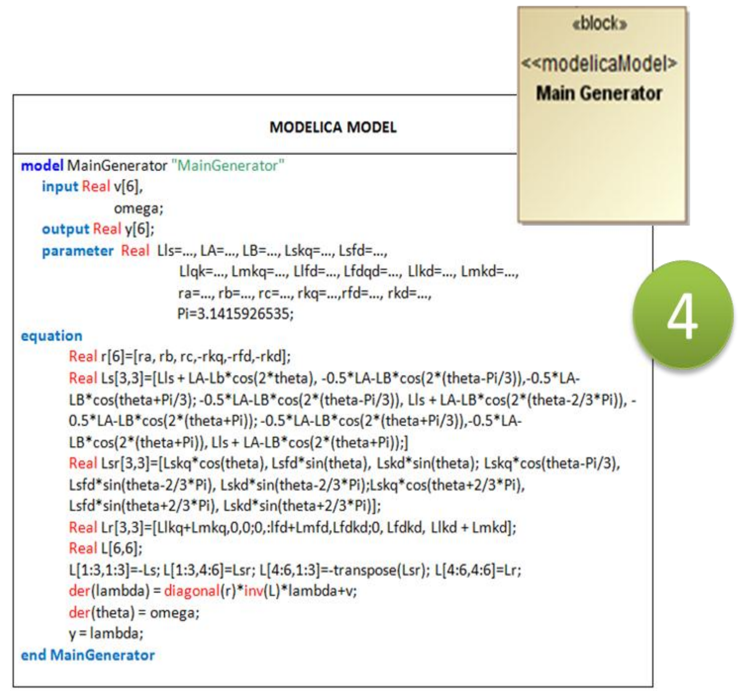


2

MuSyC Avionics Design Challenge (cont.)

System Behavior

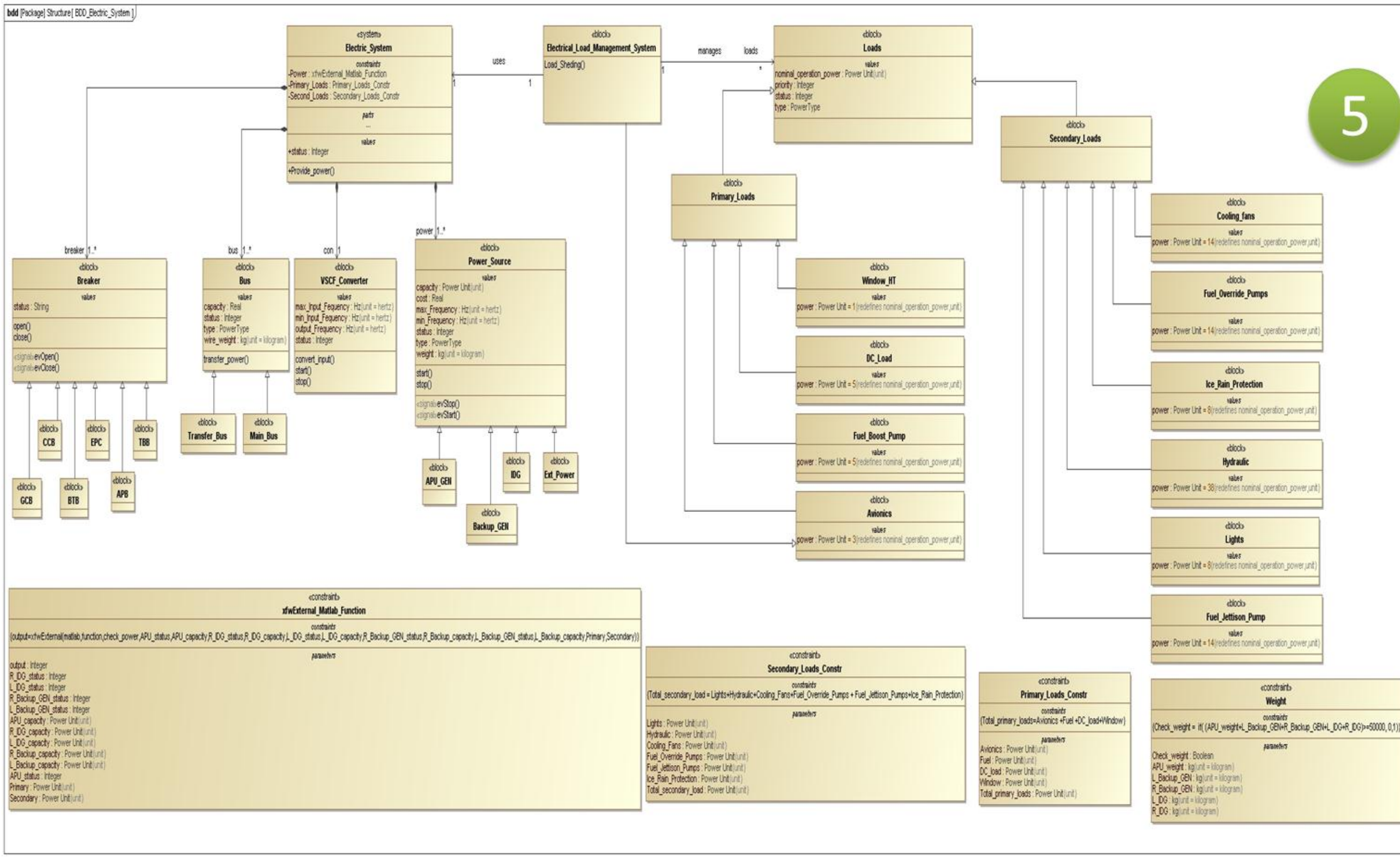




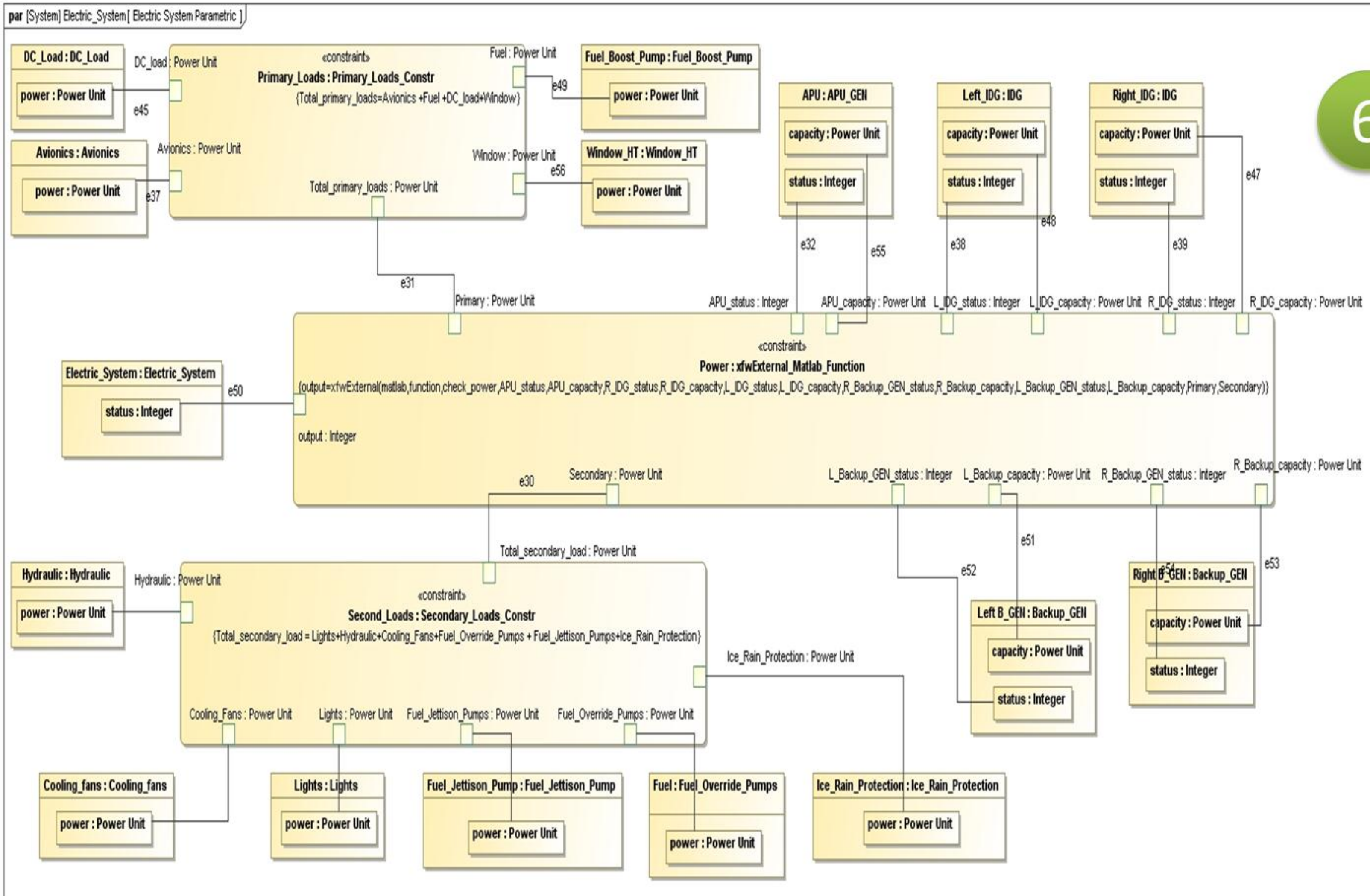
MuSyC Avionics Design Challenge (cont.)

System Structure

5

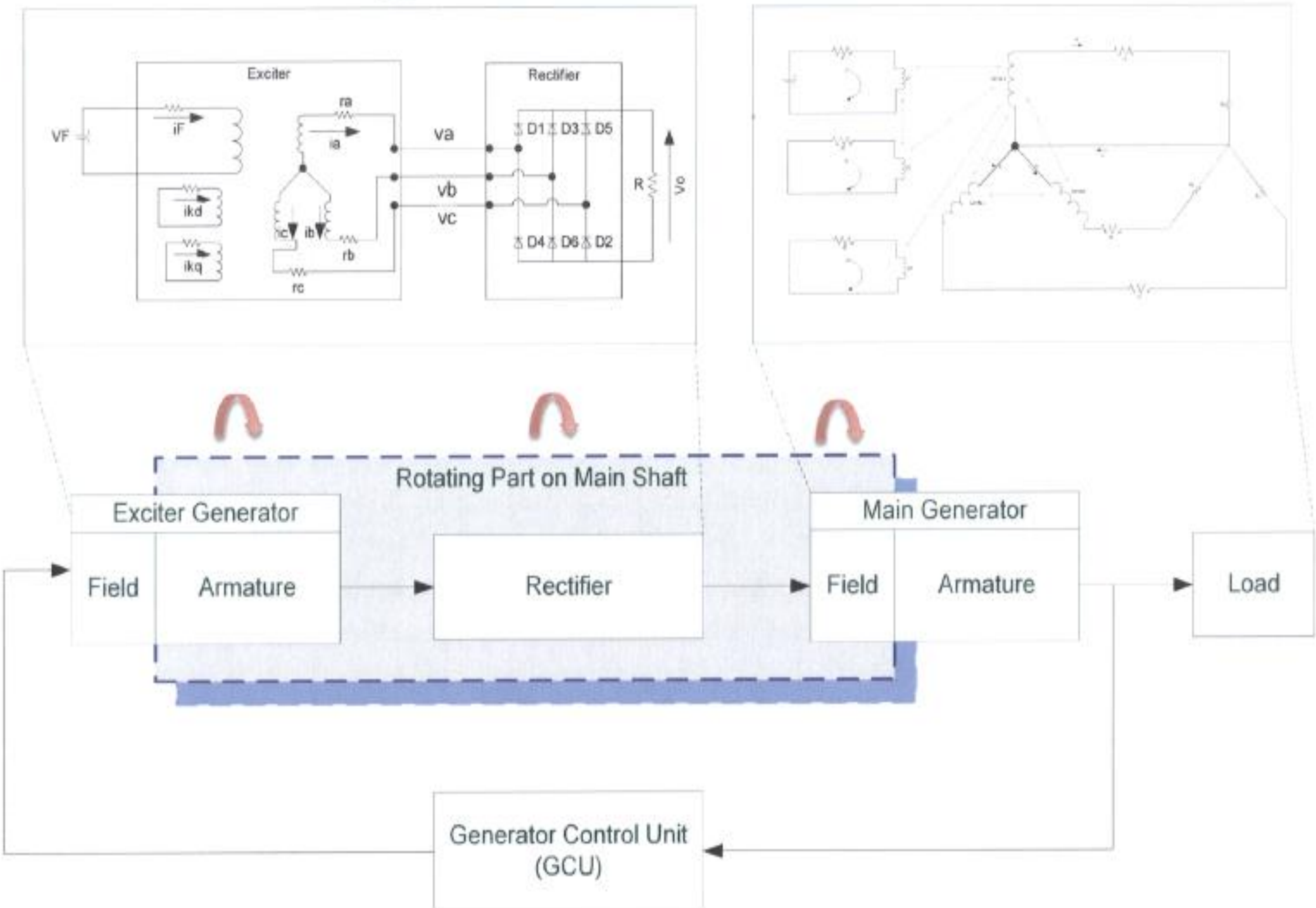


MuSyC Avionics Design Challenge (cont.)

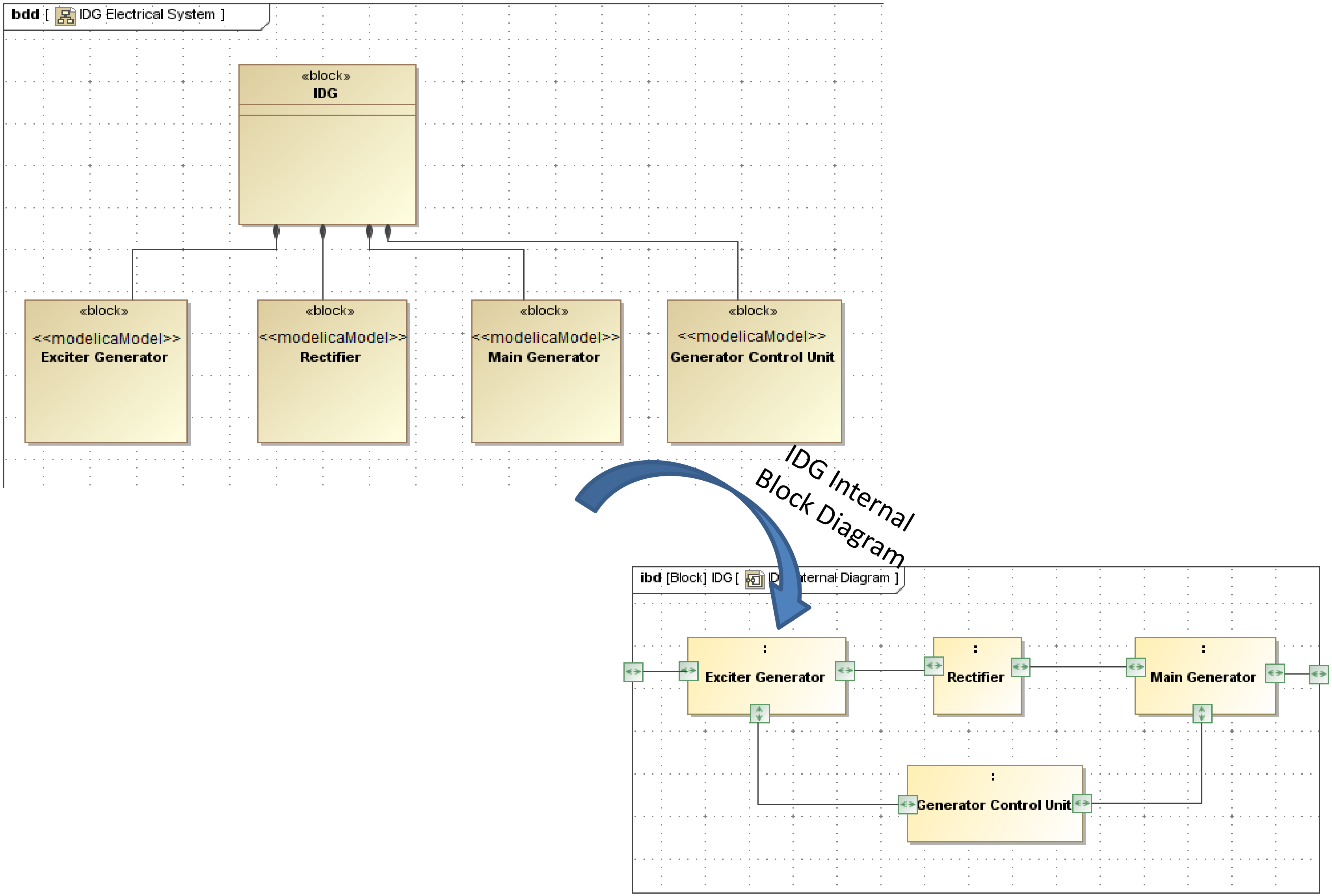


6

IDG Electrical Subsystem Block Diagram



SysML - Modelica IDG Structure Modeling



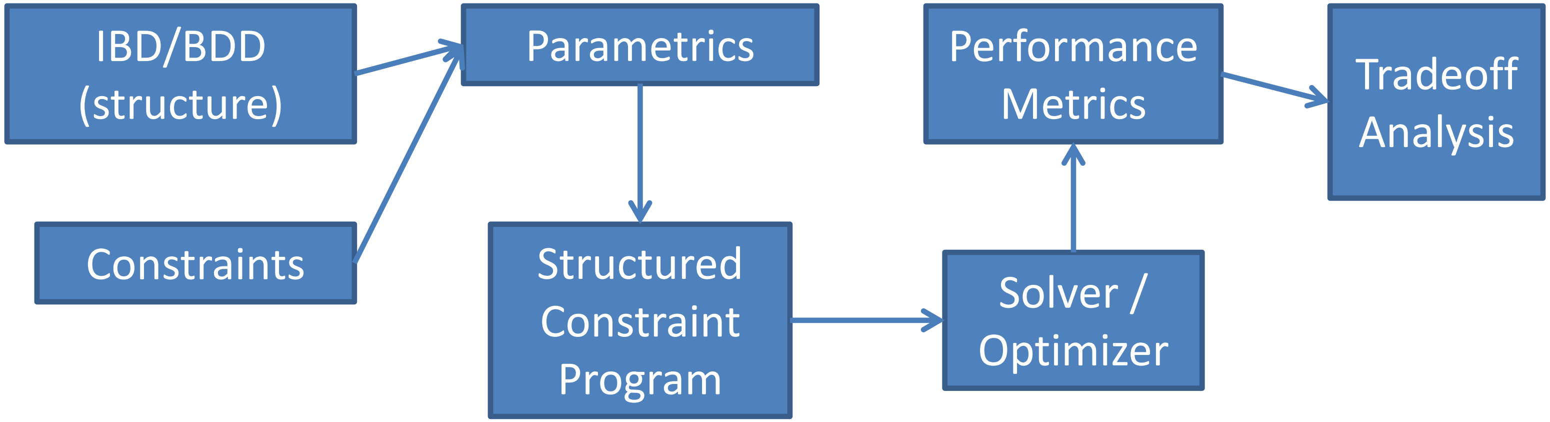
SysML - Modelica Main Generator Behavior Modeling

MODELICA MODEL

Numerical values

```
model MainGenerator "MainGenerator"
    input Real v[6],
               omega;
    output Real y[6];
    parameter  Real  Lls=..., LA=..., LB=..., Lskq=..., Lsfd=...,
                          Llqk=..., Lmkq=..., Llfd=..., Lfdqd=..., Llkd=..., Lmkd=...,
                          ra=..., rb=..., rc=..., rkq=...,rfd=..., rkd=...,
                          Pi=3.1415926535;
equation
        Real r[6]=[ra, rb, rc,-rkq,-rfd,-rkd];
        Real Ls[3,3]=[Lls + LA-Lb*cos(2*theta), -0.5*LA-LB*cos(2*(theta-Pi/3)),-0.5*LA-
        LB*cos(theta+Pi/3); -0.5*LA-LB*cos(2*(theta-Pi/3)), Lls + LA-LB*cos(2*(theta-2/3*Pi)), -
        0.5*LA-LB*cos(2*(theta+Pi)); -0.5*LA-LB*cos(2*(theta+Pi/3)),-0.5*LA-
        LB*cos(2*(theta+Pi)), Lls + LA-LB*cos(2*(theta+Pi));]
        Real Lsr[3,3]=[Lskq*cos(theta), Lsfd*sin(theta), Lskd*sin(theta); Lskq*cos(theta-Pi/3),
        Lsfd*sin(theta-2/3*Pi), Lskd*sin(theta-2/3*Pi);Lskq*cos(theta+2/3*Pi),
        Lsfd*sin(theta+2/3*Pi), Lskd*sin(theta+2/3*Pi)];
        Real Lr[3,3]=[Llkq+Lmkq,0,0;0,:lfd+Lmfd,Lfdkd;0, Lfdkd, Llkd + Lmkd];
        Real L[6,6];
        L[1:3,1:3]=-Ls; L[1:3,4:6]=Lsr; L[4:6,1:3]=-transpose(Lsr); L[4:6,4:6]=Lr;
        der(lambda) = diagonal(r)*inv(L)*lambda+v;
        der(theta) = omega;
        y = lambda;
end MainGenerator
```

Tradeoff Analysis – Design Space Exploration



- Given the requirements/specifications captured as constraints/metrics and mapped to structure/behavior in the parametric diagrams, is it possible to perform tradeoff analysis and design space exploration via multicriteria optimization-based and constraint-based reasoning methods and tools?
- Can this be done hierarchically? Respect modularity?
- How do we efficiently link the "integrated modeling hub" to tradeoff analysis tools?
- **Impact** analysis and change management?

Design Space Exploration: Integration of Logic and Optimization

- Expressed as **multi-objective optimization problem**
- **Approaches**
  - **Exact: Integer Linear Programming, Branch and Bound, Constraint Programming**

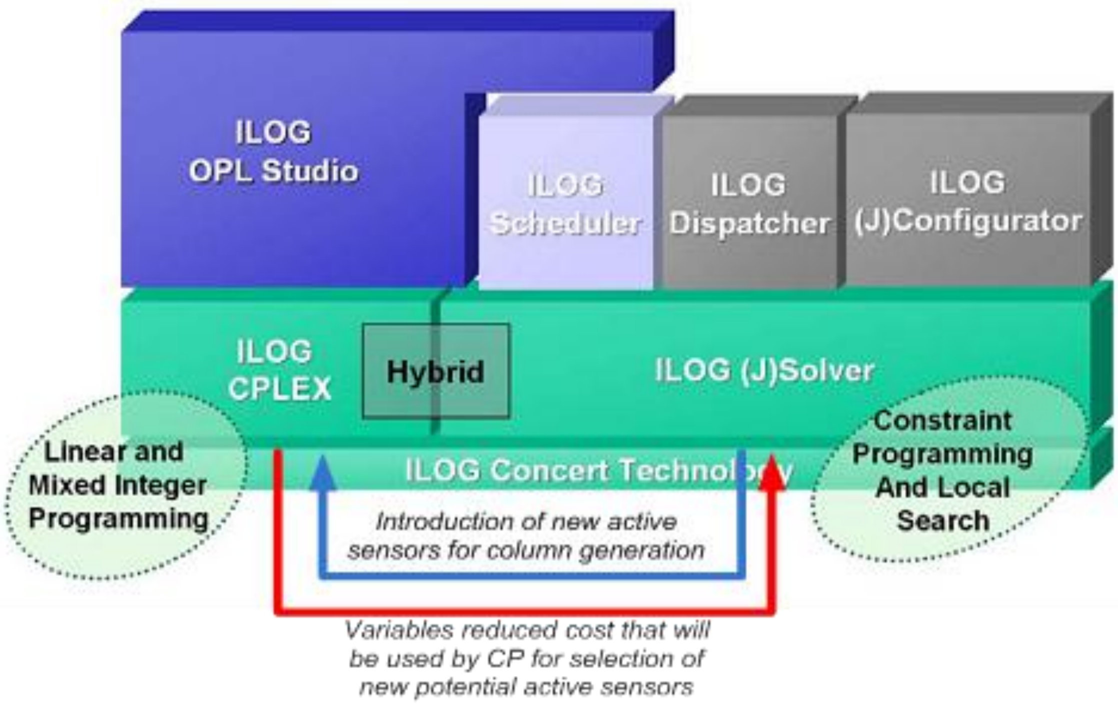
    ⇒ **Prohibitive large computation times**
  - **Heuristics: Polynomial complexity, especially crafted for the particular optimization problem**

    ⇒ **Reasonable quality solutions**
  - **Meta-heuristics: Simulated Annealing, Tabu Search, Evolutionary Algorithms**

    ⇒ **Good quality in reasonable time**

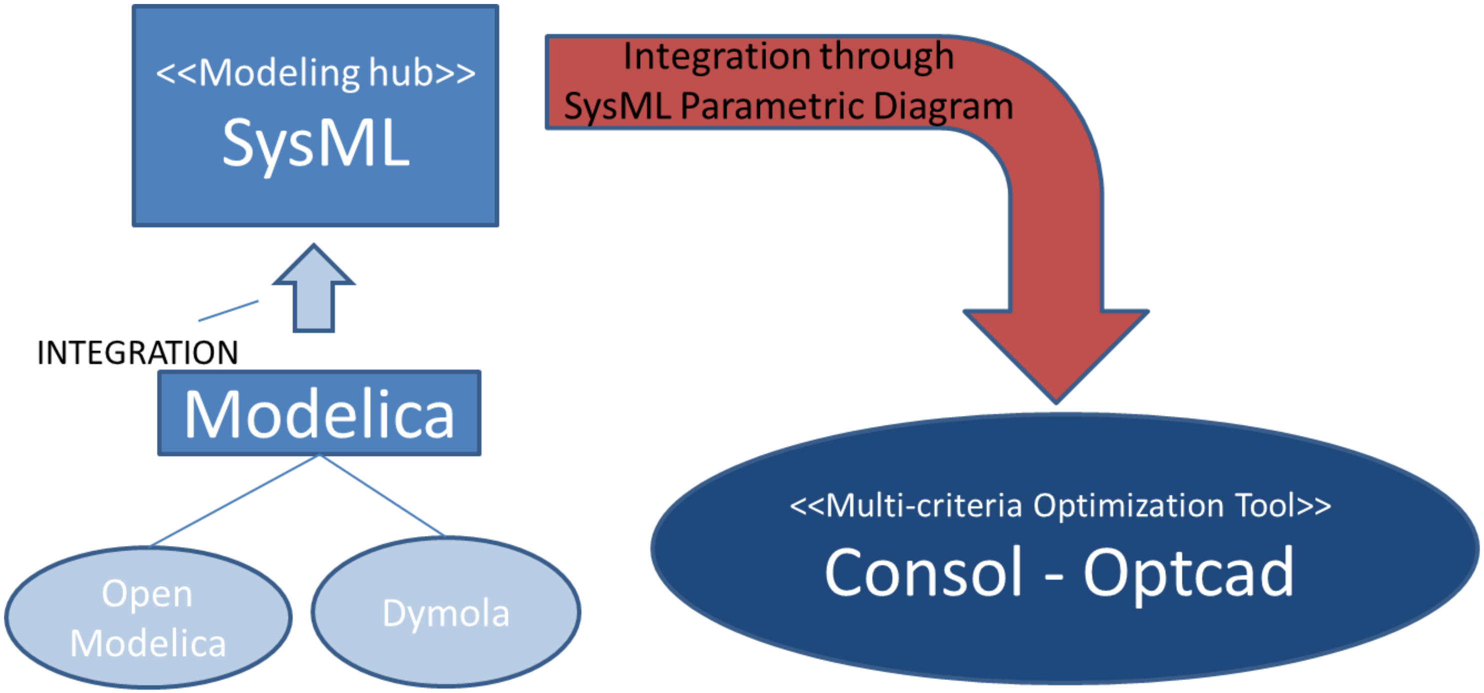
INTEGRATION OF CONSTRAINT-BASED REASONING AND OPTIMIZATION FOR NETWORKED CPS TRADEOFF ANALYSIS AND SYNTHESIS

To enable rich design space exploration across various physical domains and scales, as well as cyber domains and scales



Trade-off Analysis Integration with Modeling "Hub"

Integration of *SysML-Modelica-MATLAB "modeling hub"* with *UMD Consol-Optcad* tools for detailed trade-off analysis of complex systems with multiple objectives and for better design space exploration



Trade-off Analysis in MBSE

Goal

Build a modeling hub for trade-off analysis in MBSE environments

Our Approach:

- Consider SysML to be at the center of this hub
- Try to integrate multi-criteria optimization and constraint-based reasoning with SysML

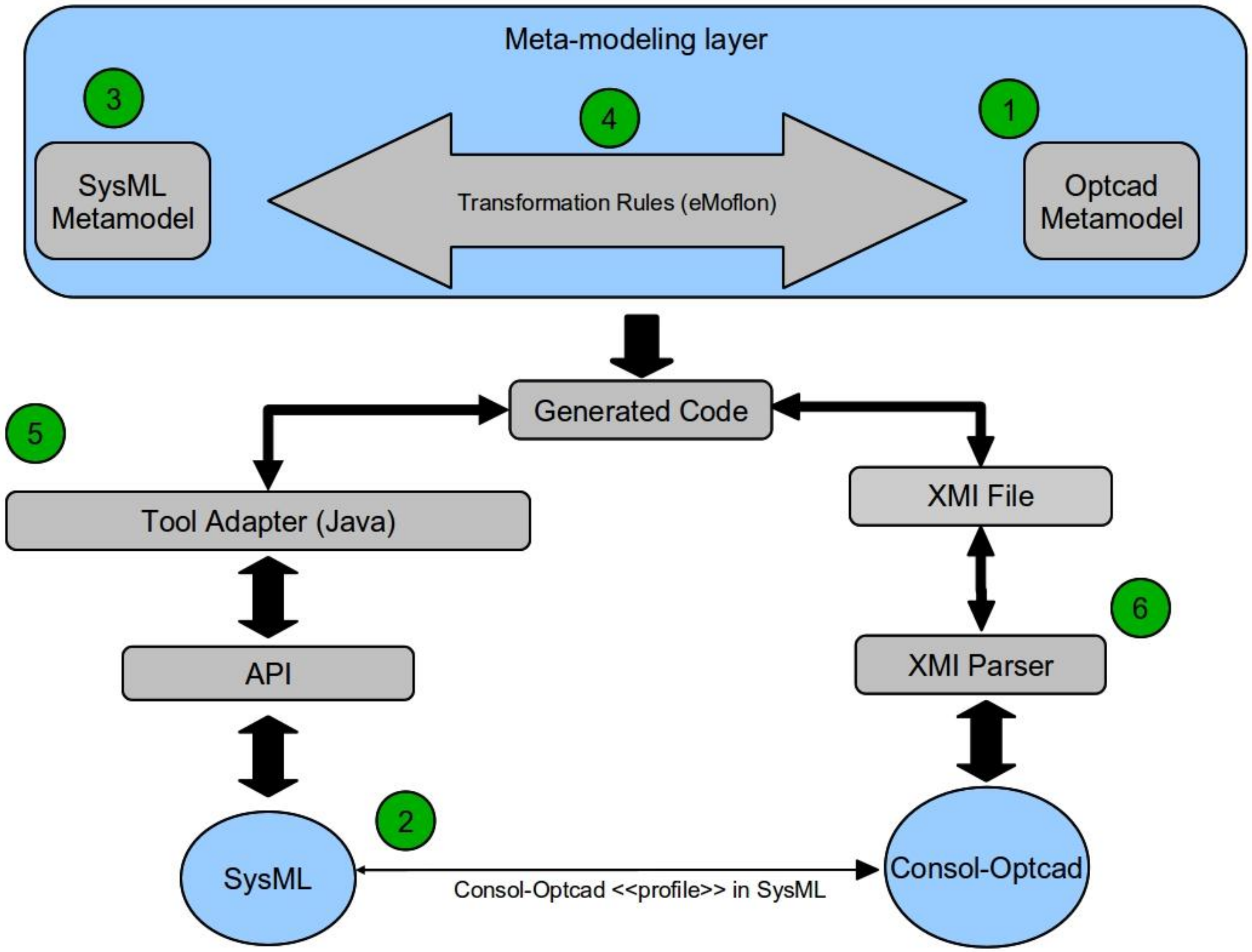
First step

- ✓ Integrate SysML with Consol-Optcad, that allows multi-criteria optimization for continuous variables
- ✓ Integration will be achieved through SysML Parametric diagram

Next steps

- ✓ Enhance the capabilities of Consol in order to handle mixed integer problems
- ✓ Integrate SysML with more trade-off tools

The big picture – Integration Steps

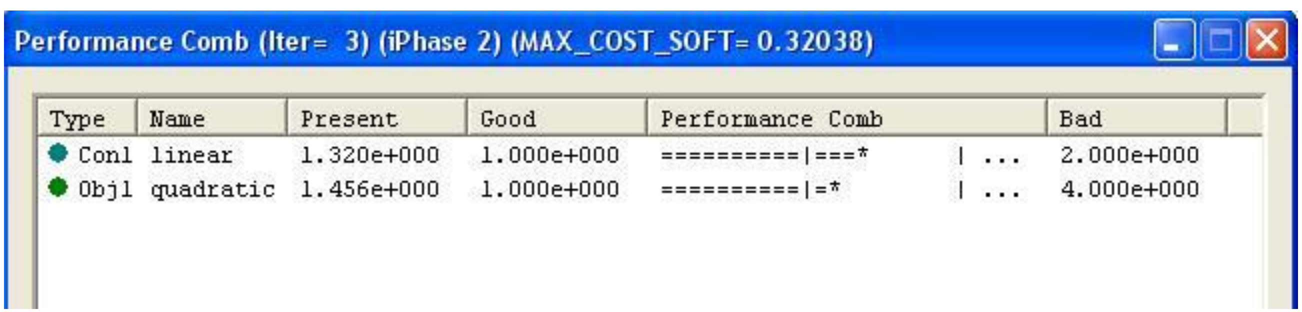


Consol-Optcad

• Trade-off tool for multi-criteria optimization
• Functional as well as non-functional objectives/constraints can be specified
• Designer initially specifies good and bad values for each objective/constraint based on experience and/or other inputs
• Each objective/constraint value is scaled based on those good/bad values, fact that effectively treats all objectives/constraints equally

<u>Major advantage</u>

➢ Gives the designer the flexibility to see results at every iteration (pcomb)
➢ Allows for a change in the good/bad values of objectives and constraints



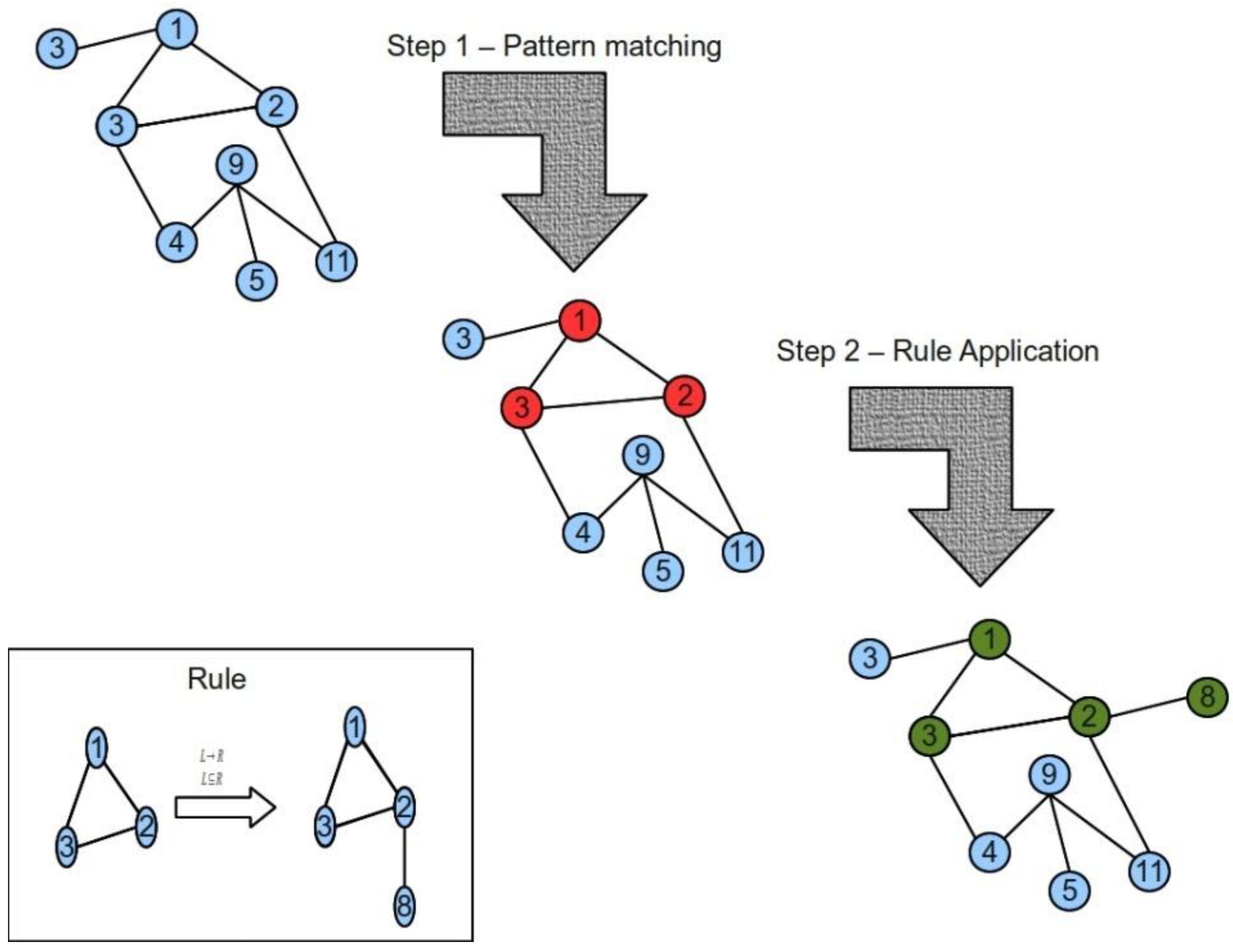
The Performance Comb
(Screenshot from Consol-Optcad)

Model Transformation

- Both meta-models are defined in Ecore format
- Transformation rules are defined within EA and are based on graph transformations
- Story Driven Modeling (SDM) is used to express the transformations
- eMoflon (TU Darmstadt) plug-in automatically generates an eclipse project
- Eclipse project hosts the implementation of the transformations in Java

Graph Transformation

Simple example of graph transformation



Tool Adapter & Parser

Tool Adapter

- ✓ It will be implemented as a SysML modeling tool (i.e. MagicDraw) plug-in
- ✓ Is used to access/change the information contained within the SysML model
- ✓ Performs the transformations by calling the generated Java methods from the previous step

Parser

- ✓ The output of the transformation will be an XMI file containing the needed Consol-Optcad constructs
- ✓The parser will translate the XMI File to a Problem Description File (input file for Consol-Optcad)

Integration of CP and MCO Tradeoff Methods and Tools with SysML− Integrated Models

The Institute for Systems Research

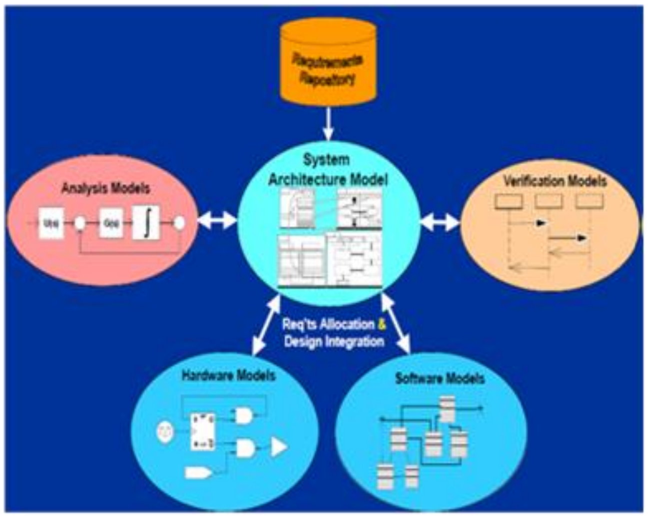
University of Maryland 1856

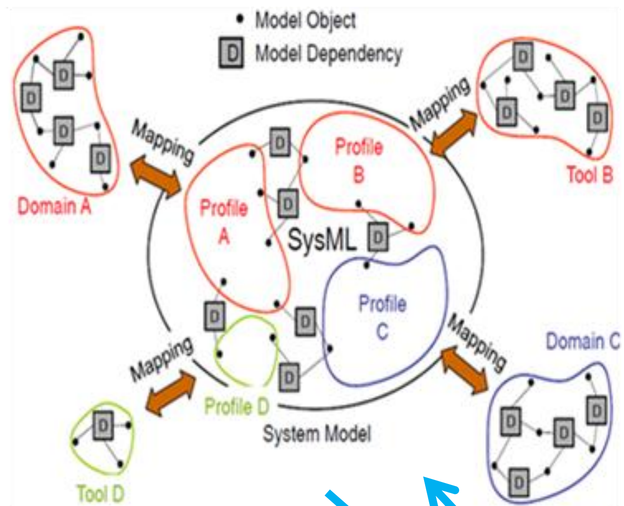
The Challenge & Need:

Develop scalable holistic methods, models and tools for enterprise level system engineering

Multi-domain Model Integration via System Architecture Model (SysML)

System Modeling Transformations





“ Master System Model”

Update System Model

Tradeoff parameters

ILOG SOLVER, CPLEX

ADD & INTEGRATE

- **Multiple domain modeling tools**
- **Tradeoff Tools (MCO & CP)**
- **Validation / Verification Tools**
- **Databases and Libraries of annotated component models from all disciplines**

DB of system components and models

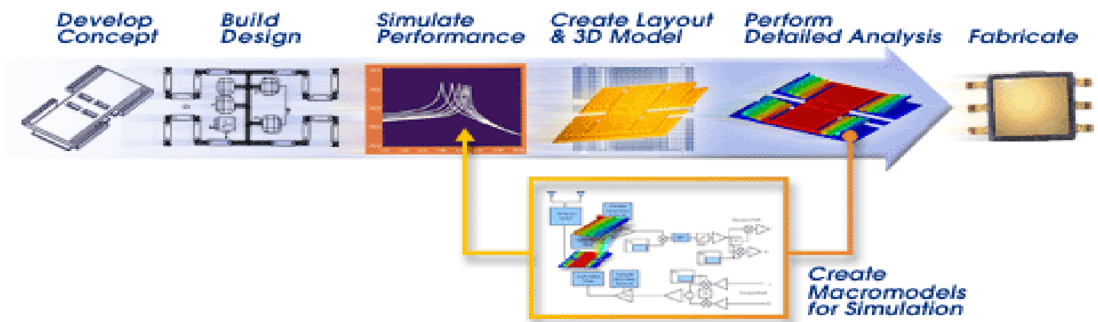
BENEFITS

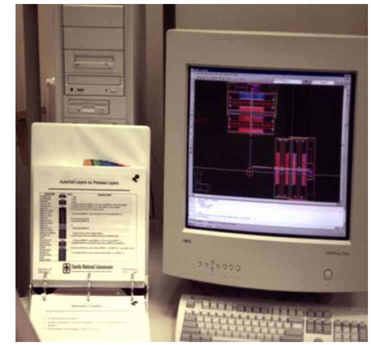
- **Broader Exploration of the design space**
- **Modularity, re-use**
- **Increased flexibility, adaptability, agility**
- **Engineering tools allowing conceptual design, leading to full product models and easy modifications**
- **Automated validation/verification**

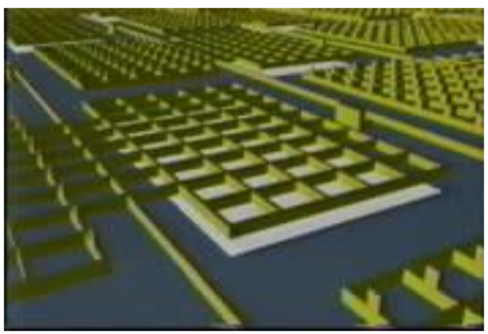
APPLICATIONS

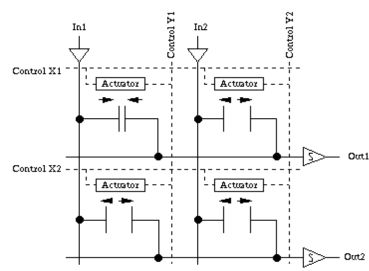
- **Aircraft and Avionics**
- **Automotive**
- **Energy Efficient Buildings**
- **Power Grid**
- **MANET and WSN**
- **Collaborating UAVs**

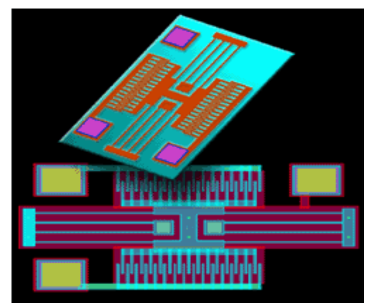
How to go From IT Abstractions to "Hardware" for MEMS and NANOS

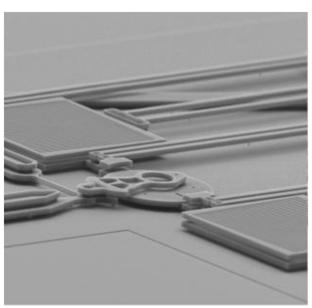


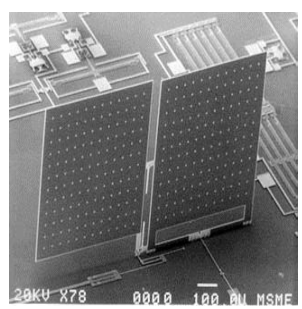


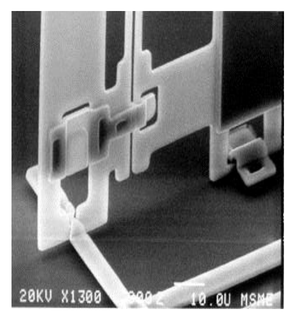


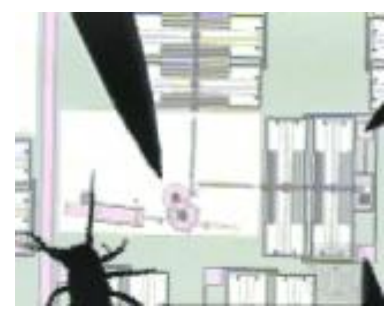












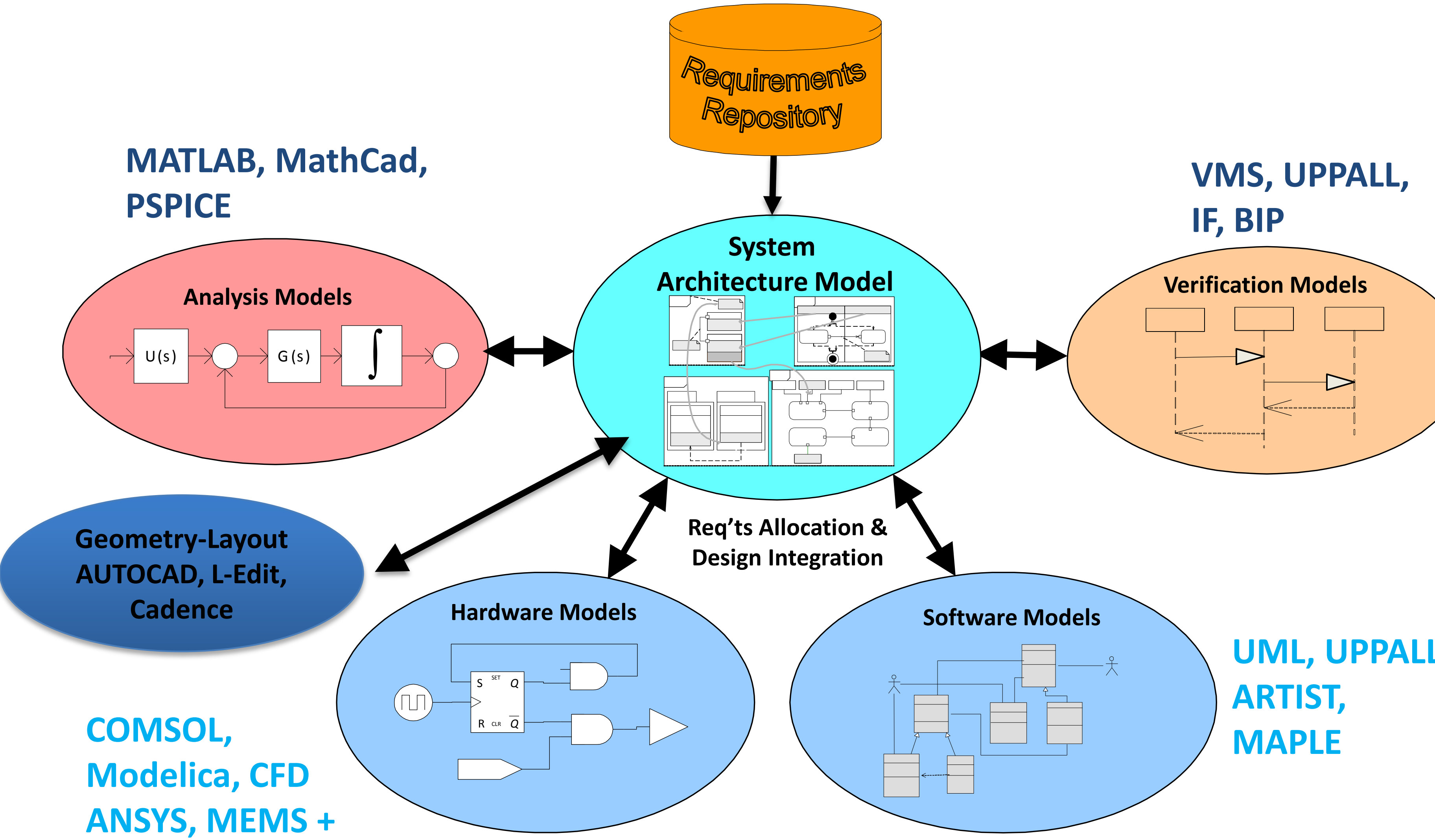
Integrated MEMS and IC Design Flow – Current Limitations

- **MEMS design currently:**
- Not well organized
- Typically requires teams of expert specialists -- mostly confined to IDMs that have their own fabs
- Traditionally separated from IC design and verification
- Little connection between the design of a MEMS device and the electronic circuitry it interacts with
- Handoff between MEMS and IC designers is *ad hoc,* manual and error-prone
- Absence of cell library of basic building blocks
- Not well suited to address cost and time-to-market demands

Integrated MEMS and IC Design Flow – Current Limitations

- **Need for a "structured" automated design flow, that links MEMS 3D design with custom IC design**
- Modeling approach defined up front repeatable, rather than made up on the fly to suit each new design
- Process variables, material properties, and geometric properties (lengths, widths, thicknesses) should be parametric to provide maximum design flexibility
- A well-characterized library of reusable MEMS building blocks (can be assembled into arbitrarily complex designs)
- Each block should have a 3D view (structure) and a behavioral model supporting all types of simulations
- Extraction and design-rule checking for MEMS devices

Using *System Architecture Model* as a MODEL Integration Framework



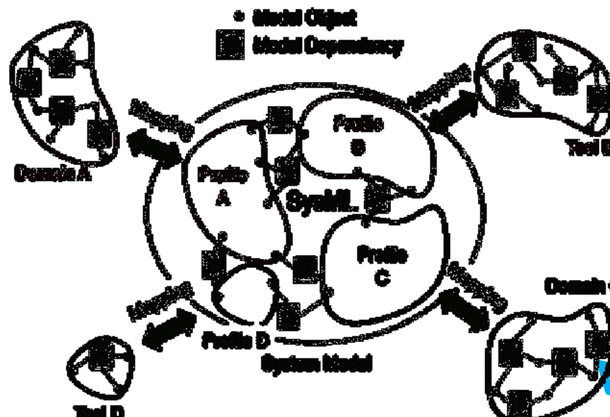
Integrated Design Environment for MEMS & NANOS

The Challenge & Need:

Develop scalable holistic methods, models and tools for MEMS & NANOS system engineering

Multi-domain Model Integration via System Architecture Model (SysML)

System Modeling Transformations



"Master System Model"

Update System Model

ILOG SOLVER, CPLEX

Tradeoff parameters

DB of system components and models

ADD & INTEGRATE

- **Multiple domain modeling tools**
- **Tradeoff Tools (MCO & CP)**
- **Validation / Verification Tools**
- **Databases and Libraries of annotated MEMS, NANO component models**

BENEFITS

- **Broader Exploration of the design space**
- **Modularity, re-use**
- **MEMS & NANO Systems Design tools allowing conceptual design, leading to full product models and easy modifications**
- **Automated validation/verification**

System Complexity Analysis and Control

Verification of Hybrid Automata via reachability analysis. Designer specifies a region of undesired behavior and method determines whether the system will exhibit it.

More recent method uses system locality to increase the efficiency of rigorous analysis via optimization, probabilistic inference or logical inference by embedding system in special structure.

Solution consists of a partially ordered set of local computations.

Sample the parameter space of the Perch Block and determine which points are feasible.

Propagate shared variables (drop PerchTime by projection).

Evaluate the Metrics Block by sampling its parameter space and taking intersections additionally with propagated data.

Propagate the shared variables (drop Weight by projection).

Evaluate the Weight Block by sampling its parameter space and taking intersections additionally with the propagated data.

Propagate the shared variables (drop FlightCurrent by projection).

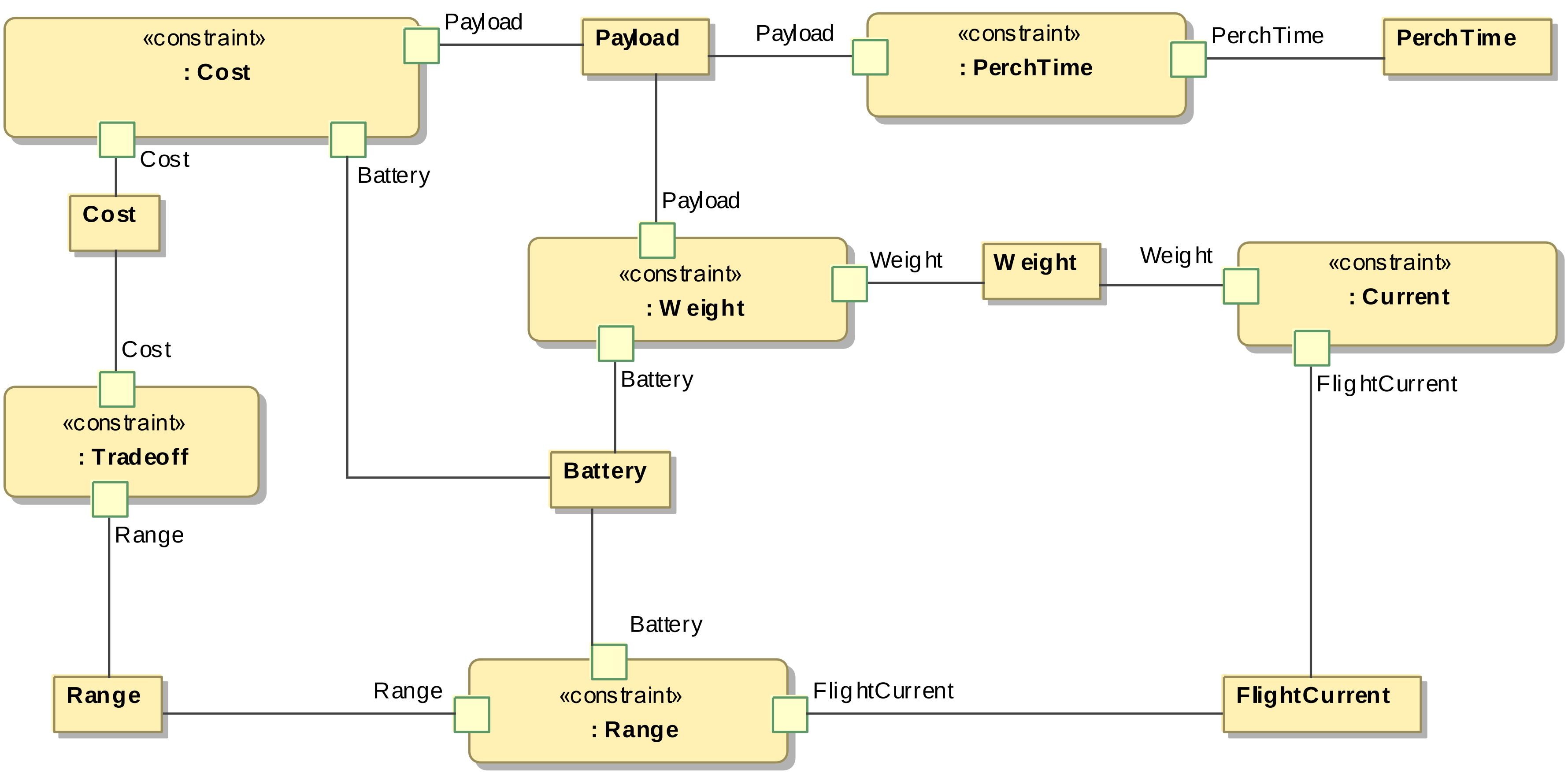
Sample the parameter space of the Range Block and determine which points are feasible.

Complexity grows linearly in the size of the system vs exponential

Whole is greater than the sum of its parts -- **Divide and Conquer**

Sneak Preview

Quadrotor Tradeoff Analysis Problem



Divide and Conquer

- Defeat in detail.
- Wedge issues.
- Divide and rule.

- Separation effective because the "*whole is greater than the sum of its parts*".
  - Difficulty of problem grows faster than the sum.
  - An enemy group of size *N* has strength ∝ N^2. *strength* ∝ *firepower* * *durability*. Both firepower and durability grow ~linearly with *N*.

- System analysis.
  - Analysis complexity grows ~exponentially with system size measured in the number of parameters.

Wedging Systems

- System represented by an undirected graph *G* = <*V*,*E*>.
  - Nodes, *V*, correspond to variables.
  - A formula $f(x_1, \ldots, x_n) = C$ induces edges $(x_i, x_j)\ \forall i \neq j \in [1, n]$.
  - Edge, $(x, y) \in E$, means that variables x, y are in mathematical relation.

- Rules of system partitioning.

1. Choose a subset of nodes that *completely separate* the graph into subgraphs.
2. Separation produces an *interface relation* that contains all the nodes in the separator.
   - By adding links, brings resulting subsystems closer to *inseparability*.

- Due to recursive partitioning this decomposition results in *trees*.

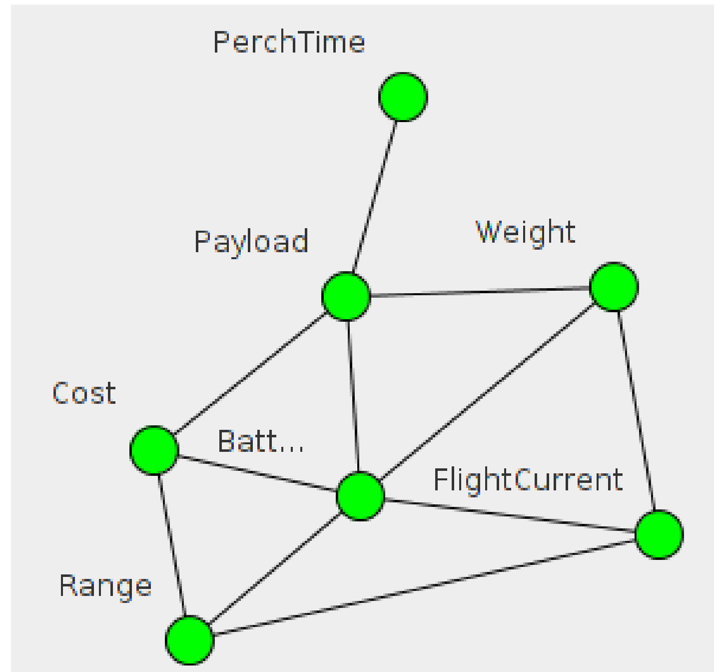
Treewidth Definition

- The "width" of a decomposition can be defined as the size of the largest component in that system.
- The treewidth is the minimum possible width over all tree decompositions-1.
- In general treewidth is NP-hard to compute.
- For many NP-complete problems on graphs, including vertex cover, independent set, dominating set, graph k-colorability, Hamiltonian circuit, network reliability, and dynamic programming , the ***complexity is exponential in treewidth and linear in problem size***.

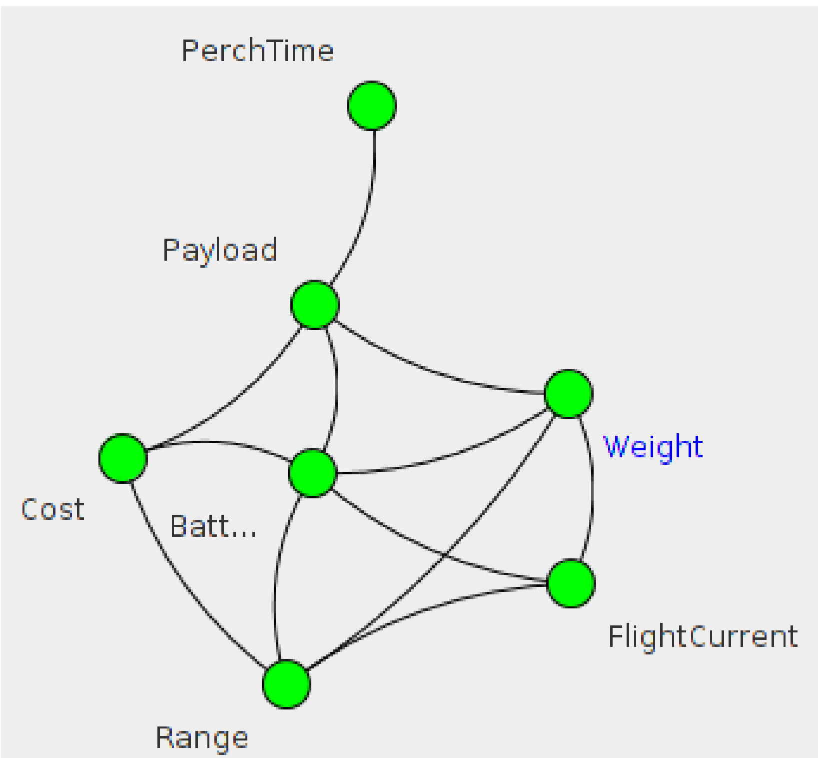
Quadrotor Analysis

NO	Func Name	
1	Cost	Cost,Battery,Payload
2	Tradeoff	Cost,Range
3	Range	Battery,Range,FlightCurrent
4	Weight	Weight,Battery,Payload
5	PerchTime	Payload,PerchTime
6	Current	FlightCurrent,Weight

Tool input from parametric diagram.

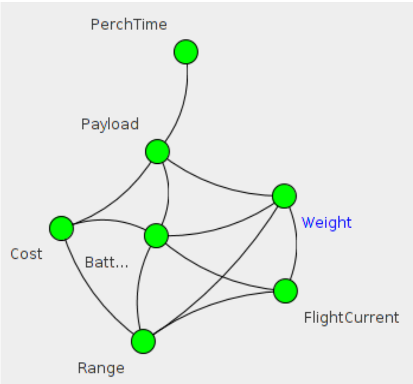


Initial graph.

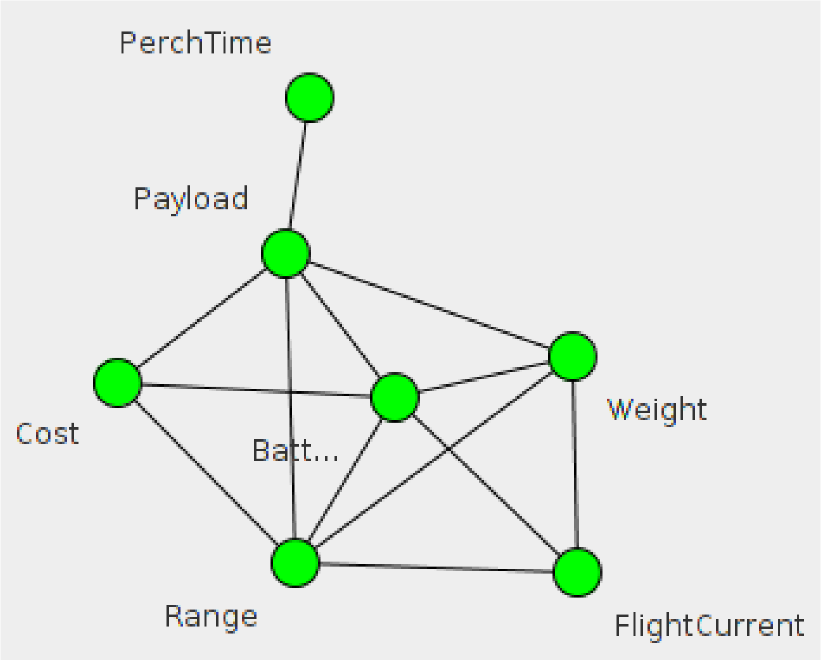


Weight to range fillin created.

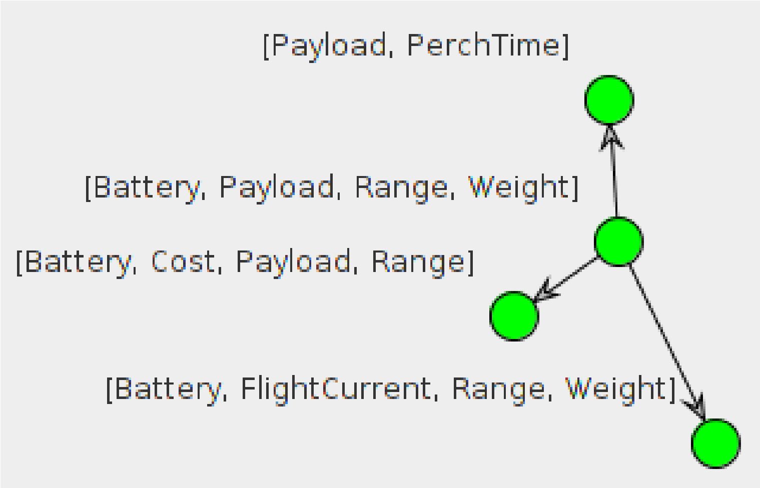
Quadrotor Analysis (cont.)



Weight to range fillin created.



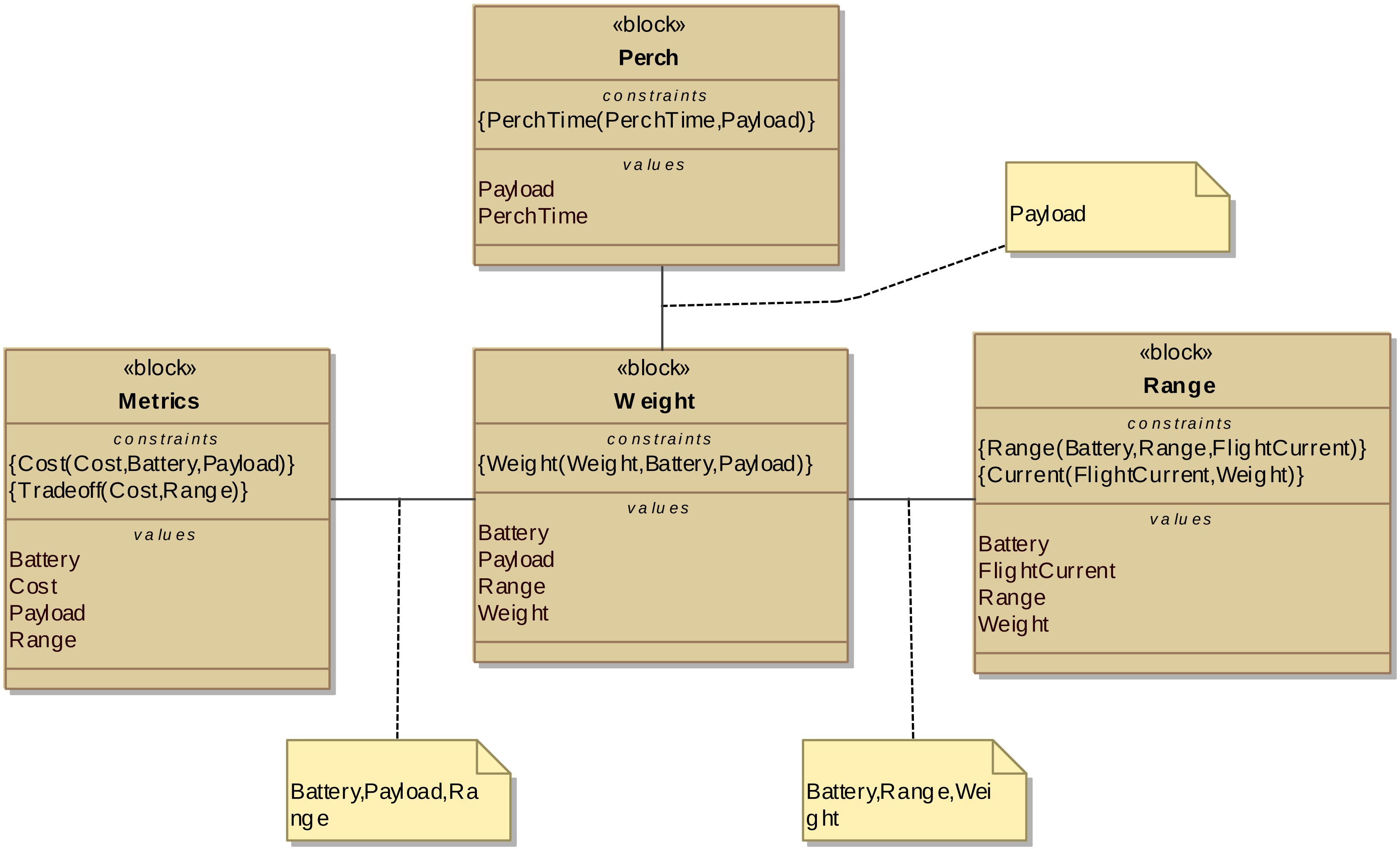
Payload to range fillin created. Graph is now chordal.



Join tree created.

The tool implemented currently uses **elimination order** rather than **separators** to perform analysis. They are mathematically equivalent. An implementation using separators is underway.

Quadrotor Factor Join Tree



Tradeoff Analysis using Summary Propagation

Solution consists of a partially ordered set of local computations.

Sample the parameter space of the Perch Block and determine which points are feasible.

Propagate shared variables (drop PerchTime by projection).

Sample the parameter space of the Range Block and determine which points are feasible.

Propagate the shared variables (drop FlightCurrent by projection).

Evaluate the Weight Block by sampling its parameter space and taking intersections additionally with the propagated data.

Propagate the shared variables (drop Weight by projection).

Evaluate the Metrics Block by sampling its parameter space and taking intersections additionally with propagated data.

- Builds tables of feasible values for each of blocks.
- Uses (weighted) *natural-semijoin* on tables to propagate information.
- Applies (aggregated) *projection* on tables to hide unnecessary information.

Summary Propagation Detail

Range: d^4 complexity of construction

Battery	FlightCurrent	Range	Weight

Weight: d^4 complexity of construction

Battery	Payload	Range	Weight

projection

T3

Battery	Range	Weight

Weighted Natural Join
Removes elements from *Weight* that do not occur in T3.

T4: Natural semijoin of *Range* and *Weight*.

Battery	Payload	Range	Weight

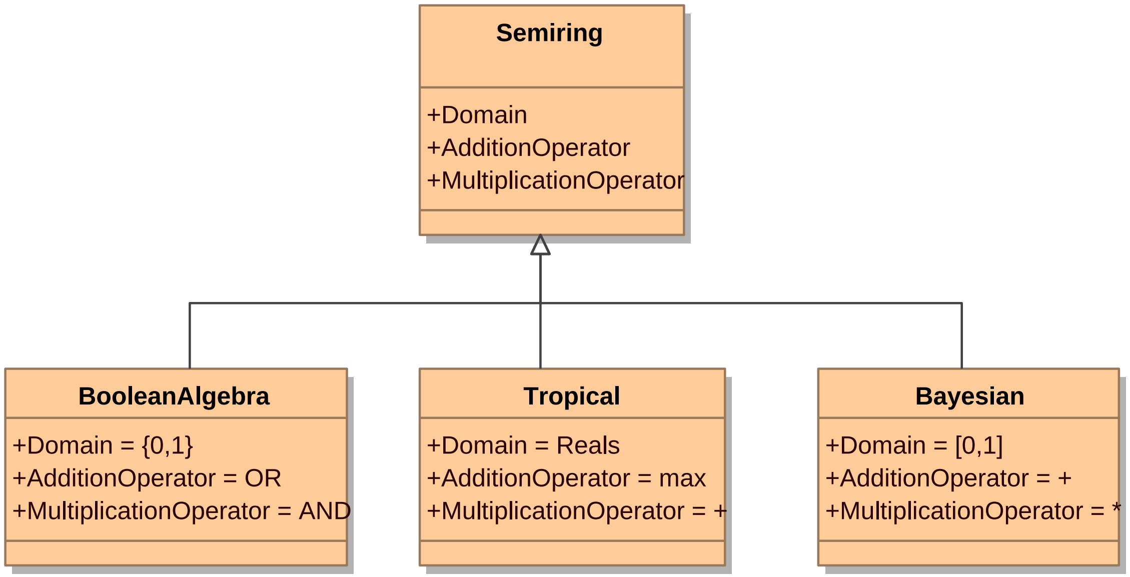
Queries

- As shown in the previous example, the query itself influences the shape of the resulting graph.
- A query that is not local can create links between non-local variables.

- The resulting graph and *analysis complexity is dependent on the query*.

Semiring Generalizations

- Inference in propositional satisfiability.
- Bayesian network inference.
- $(max, +)$ optimization; other semirings



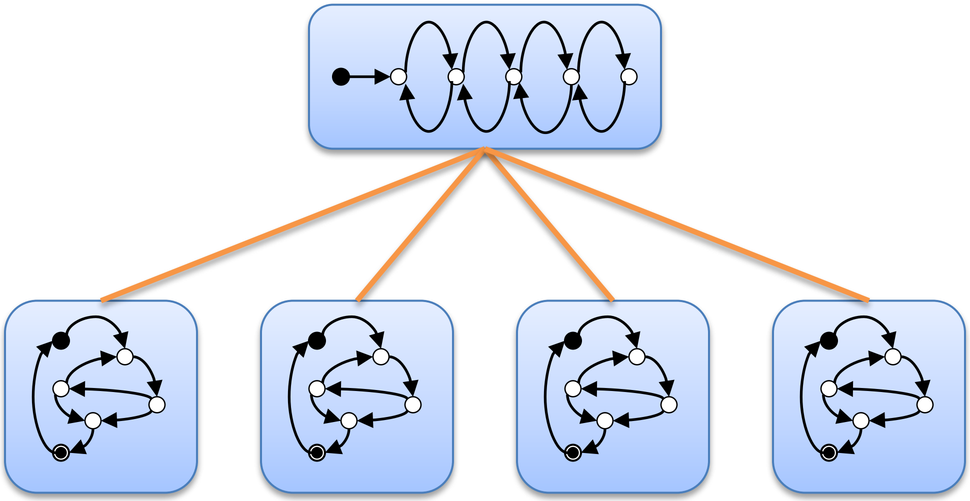
Distributed / Parallel Implementations

- Trees and a query define a partial order so parallelism exists to be exploited.

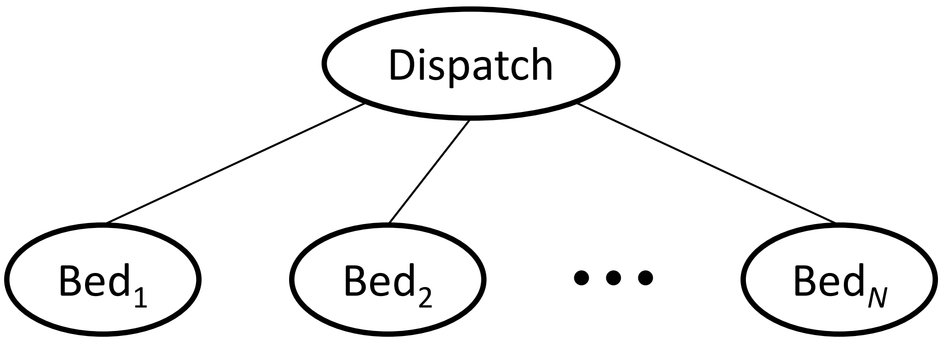
- Cliques define *local, encapsulated calculations*. These are suitable for distributed evaluation, either by computers or by teams of engineers.

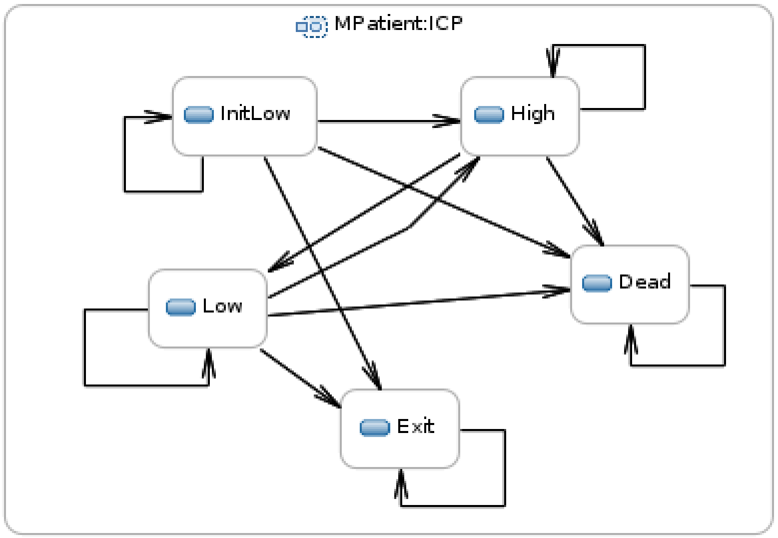
Behavioral Generalization (Ongoing Work)

- The systems examined thus far can be treated statically. What happens when the components have behavior?
- [Ferrara 2005] proves that evaluating this is EXPSPACE hard in general.



Example System: Intensive Care Unit (ICU)

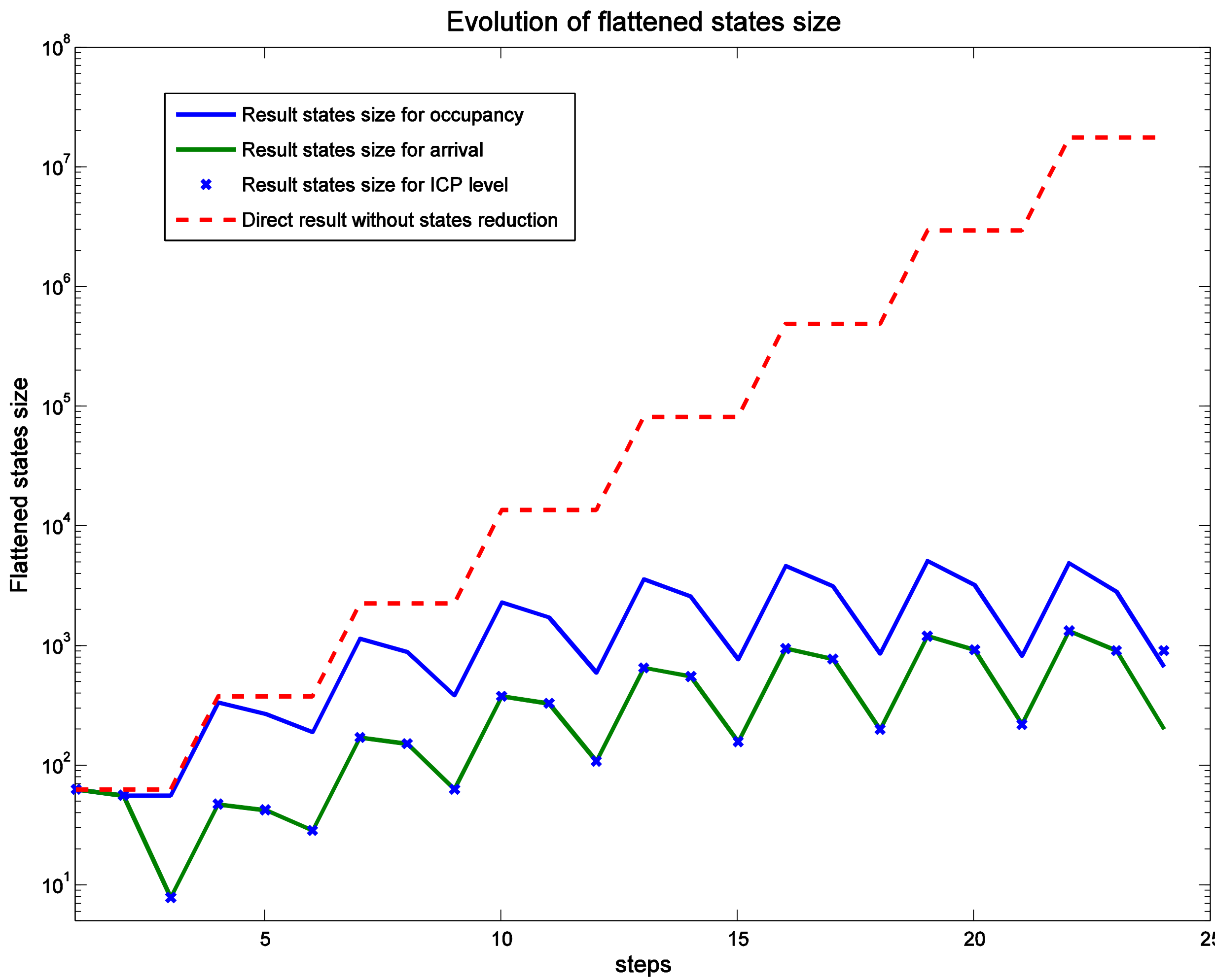




- *N*, identical, interacting behaviors at each bed.
- Interaction is via dispatch.
- Overall machine has many states due to the whole being greater than the sum of the parts.

Composition is Product of Machines
Projection is Label Erasure

- Appropriate definition of semiring operations for summation and multiplication yields significant reductions in this problem.
- Reductions are due to symmetry of the beds however.
- **What class of systems does this generalize to?**

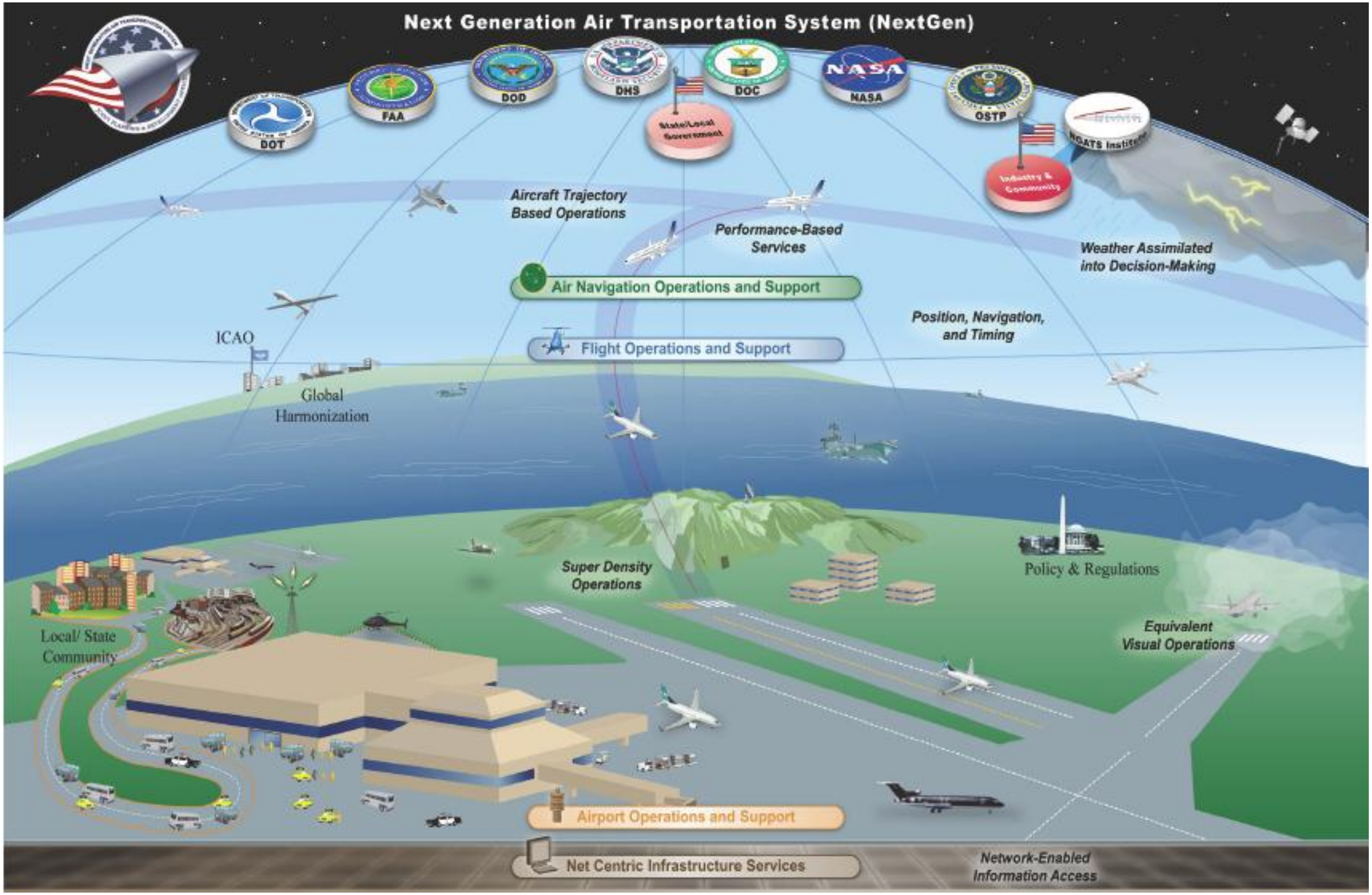


Automotive

- The new trend is
  - Break the "one-subsystem one-ECU" paradigm
  - Distribute functionalities over several nodes to optimize number and cost of ECUs
- Advantages
  - flexibility, cost reduction, redundancy (fault-tolerance)
  - more sophisticated control enabled by more powerful hardware

AUTOSAR

FAA NextGen


Next Generation Air Transportation System (NextGen)
DOT
FAA
DOD
DHS
DOC
NASA
OSTP
Aircraft Trajectory Based Operations
Performance-Based Services
Weather Assimilated into Decision-Making
Air Navigation Operations and Support
Position, Navigation, and Timing
ICAO
Flight Operations and Support
Global Harmonization
Super Density Operations
Policy & Regulations
Local/ State Community
Equivalent Visual Operations
Airport Operations and Support
Net Centric Infrastructure Services
Network-Enabled Information Access

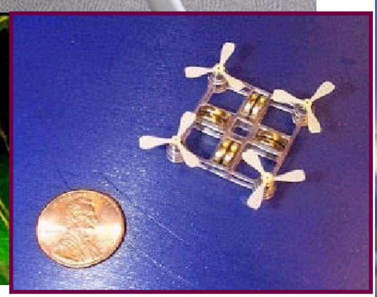
The Institute for Systems Research

MBSE for Robotic Arms and Grippers

- Transcend areas of application: from space to micro robotics
- Include material selection in design
- Include energy sources, resilience, reliability, cost
- Include validation-verification and testing
- Use integrated SysML and Modelica environment
- Link it to tradeoff tools CPLEX and ILOG Solver
- Demonstrate reuse, traceability, change impact and management

Autonomous Swarms – Networked Control

- *Component-based Architectures*
- *Communication vs Performance Tradeoffs*
- *Distributed asynchronous*
- *Fundamental limits*



SMART MANUFACTURING

- Flexible production lines
- Robotics and humans integrated
- Reduce manufacturing to compilation
- Custom materials
- Materials as a design variable
- Composite materials design

**Model-based systems engineering –
manufacture to component models**

New Home Health Platforms

- Digital home entertainment infrastructure can be used for health
- Everyday health through everyday devices
- Personalized, proactive health info/reminders/agents

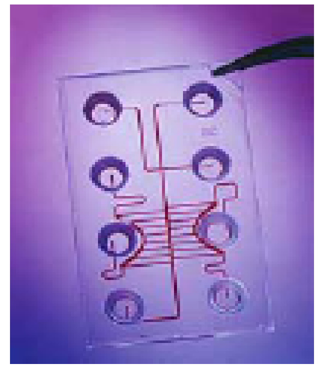
INTEL'S PROACTIVE HEALTH LAB

HYBRID LOC -- BIOCHIPS

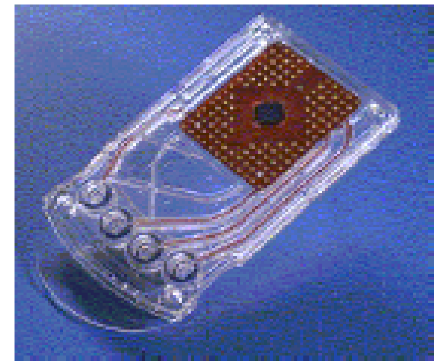
Biochips are currently emerging with different form factors and technologies for applications in research, pharma and healthcare

All biochip concepts are disposables

DNA µArray



µfluidic chip



µfluidic chip
+
DNA µArray

Applications:

- Basic research
- Pharma R&D / Drug development
- Healthcare

} "Red Biotech"

- Agriculture and environment — "Green Biotech"
- Industrial and process control — "Grey Biotech"

2

MODEL-BASED SYSTEMS ENGINEERING: *Challenges 1*

- **Domain Specific Modeling Languages** (DSML) with semantics that can be composed and manipulated
- **Composition platforms** – correct by construction systems platforms and models of computations; substantial reduction in V&V
- System and component **behavioral abstractions** that can support Incremental System Integration – while preserving testability and predictability
- **Fully integrated semantically** control, hardware, software and systems design tools and platforms
- **Much richer semantics for interfaces**, especially in the most critical physical to cyber boundary – accommodate and indeed unite the two sides of the boundary
- **Metamodels and Metamodeling Environments**, user/designer friendly

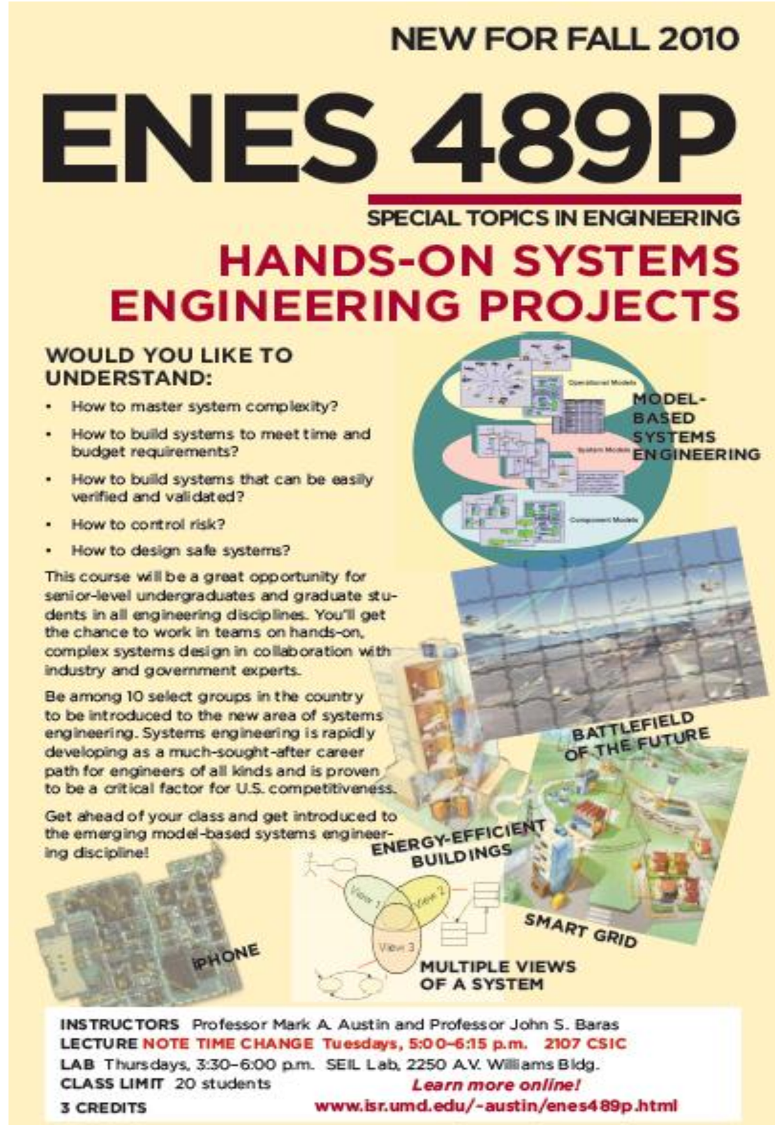
MODEL BASED SYSTEMS ENGINEERING: *Challenges 2*

- **Principles for system integration — System Science — Network Science**
- Fundamental **performance limitations** of networked systems with asynchrony, concurrency, etc.
- Fundamental implications of **physical implementation technology selection** – multi-physics
- Fundamental **performance limits of distributed hybrid asynchronous systems**, concurrency, non-collocated sensors, decision making and actuation nodes, multiple feedback loops, delay & bandwidth constraints
- Distributed control of and inference in the same – self organization – self assembly
- **Theories of compositionality**
- Much better **integration of logic and optimization** for trade-off analysis in dynamical systems

A Bold Experiment

Starting early in the education chain

Undergraduates working with industry and government mentors on SE projects



Thank you!

baras@umd.edu

301-405-6606

http://www.isr.umd.edu/~baras

Questions?