# Cyber-Physical Systems

Challenges & Opportunities

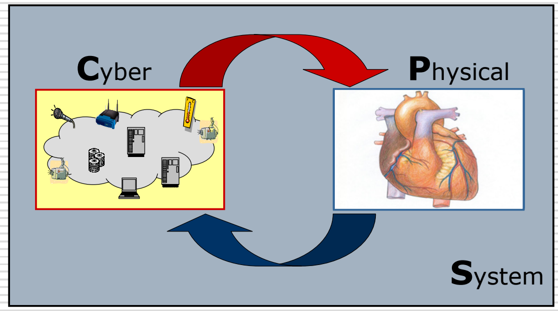
**Azer Bestavros**

Computer Science Department
Boston University

http://www.cs.bu.edu/groups/wing

Sensor Network Consortium
May 8, 2008

## Cyber-Physical Systems (CPS)






## CPS: Major Roadblock

- All our software frameworks abstract out or add uncertainty to spatio-temporal attributes:
  - Programming languages
  - Virtual memory
  - Caches
  - Dynamic dispatch
  - Speculative execution
  - Power management (voltage scaling)
  - Memory management (garbage collection)
  - Just-in-time (JIT) compilation
  - Multitasking (threads and processes)
  - Component technologies (OO design)
  - Networking (TCP)
  - …



## CPS: Certification is key

- By definition, CPS = Safety Critical
- System must be certified with respect to a variety of stringent safety constraints
  - Safety constraints
  - Real-time constraints
  - Non-interference constraints
  - Fail-safe constraints
- Not about proofs of concept
- Cost is a secondary concern!



## The Hospital Dilemma

- Paraphrasing John Rushby, SRI

  *"The patient on the operating table is the medium through which multiple life-support (respiratory and circulatory) subsystems interact. There are documented cases of deaths and severe injury due to medical device interference."*

- Who is liable? The manufacturers? The hospital? A real stumbling block for innovation!



## The Boeing Dilemma

- Paraphrasing Edward Lee, UC Berkeley:

  *"In fly by wire aircraft, it is not the software that is certified but the entire system. If a manufacturer expects to produce a plane for 50 years, it needs a 50-year stockpile of fly-by-wire components that are all made from the same mask set on the same production line. Even a slight improvement require the software to be re-certified."*

- What about outsourcing? How is Airbus doing it?

## What Could Go Wrong?

- A few potent examples
  - Interference between controllers
    - Interference between multiple life support subsystems
  - Compatibility questions
    - Will upgrading break my system? (regression is hell!)
  - Data plane interactions
    - Could I substitute a Kalman filter with another?
  - Control plane interactions
    - Do firewall security rules compose safely with my network monitoring infrastructure?



## What Could Go Wrong?

- A few potent examples
  - Interference between controllers
    - Interference between multiple life support subsystems
  - Compatibility questions
    - Will upgrading break my system? (regression is hell!)
  - Data plane interactions
    - Could I substitute a Kalman filter with another?
  - Control plane interactions
    - Do firewall security rules compose safely with my network monitoring infrastructure?

SCALE



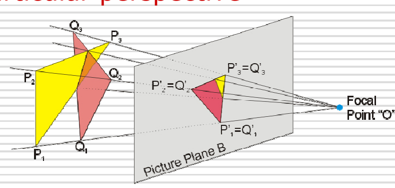
## (Scalable) Compositional Analysis

- **Composition:**
  *The system Z that results from having X interact with Y*
- **Analysis:**
  *Formally derive safety properties of a system W*
- **Analyzing a composition:**
  *Derive properties of Z by analyzing the composition of X and Y*
- **Composing the analysis:**
  *Derive properties of Z by composing the analysis of X and the analysis of Y*



## Component Property Projections

- Allows us to abstract the system for a particular perspective




- But, for CPS, no one perspective is enough!



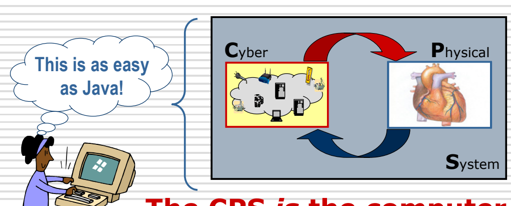
## Need to "Compose" Theories

- Different techniques are better at dealing with different types of properties
  - Thermodynamics: Heat diffusion; energy transfer, …
  - Control theory: Convergence, stability, dynamics, …
  - Network calculus: Max/min delays, b/w, loss rates, …
  - Queuing theory: Average delay, utilization,
  - Real-time theory: Schedulability/timing analysis, QoS, …
  - State-space analysis: Deadlocks, synchronization, …
  - Game theory: Price of anarchy, mistreatment, …
  - *… put your pet theory here*
- Need a seamless way to leverage all such theories and techniques



## BU Project: *sn*Bench

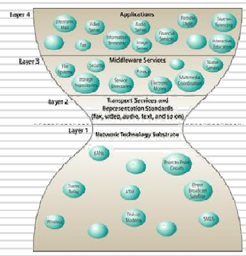
Design/implement an integrated software development and certification environment for CPS applications over a shared CP infrastructure




**The CPS *is* the computer…**

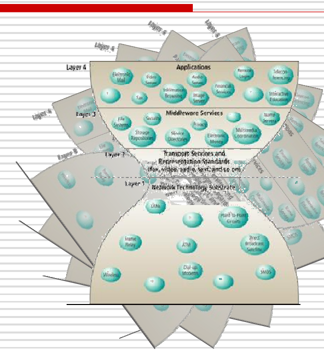
## *sn*Bench: Hourglass Model…

- If the Physical World is the Computer, then what is its ISA?
- Why an ISA?
  - Minimizes cost
  - Encourages innovation
  - Speeds up adoption
  - Scales up education






## … One Hourglass Model / Theory






## *sn*Bench: Goals

- Write once; run anywhere
- Don't program nodes; program the CPS!
  - Start with building blocks – "Gadgets"
    - Models of the physical domain objects
    - Sensors (cameras, motion sensors, biosensors, …)
    - Actuators (device controllers, net services, …)
    - Stock algs (Kalman filter, FFT, edge detect, …)
  - Glue together with high-level language
    - Conditionals, loops, triggers, functions
  - Pretend the network isn't there
    - "Single CPS System Image"
- Integrate programming and verification



## *sn*Bench: Programming Cycle

- Program
  - Program specified by gluing together building blocks using a high-level language (SNAFU)
- Compile and Certify
  - Program is compiled to produce a plan of execution expressed over a CPS domain abstraction (STEP)
- Map and Link
  - STEP plans are decomposed in smaller dispatch-able STEPs which are linked
- Load and Execute
  - STEP plans are dispatched and loaded onto the computational core of the CPS infrastructure

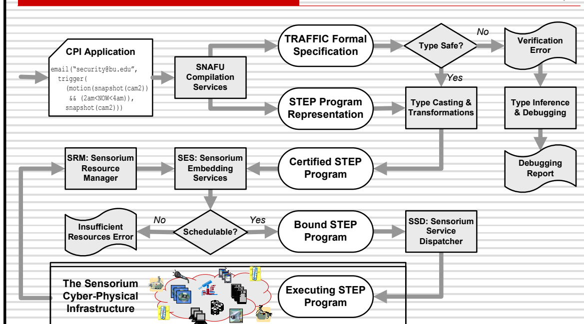


## *sn*Bench: Certification

- Annotations used to define behavioral constraints, a.k.a., "types"
- Annotations are distilled into domain-specific formal representation of the interfaces between CPS "gadgets"
- Use type-checking and type inference to mechanically verify safety properties *a priori*



## *sn*Bench: Roadmap



## *sn*Bench: Status

- Used as a platform for projects in SE and AI courses in CS since 2005!
  - Students developed opcodes, GUIs, resource managers, applications, rule-based front-end for medical devices, …
  - Multimodal sensing and actuation using motes, PTZ-cams, kismet wireless intrusion detection, Garcia robots, …
- *sn*Bench is Alive!
  - Latest code and demos from multiple case studies at http://csr.bu.edu/snbench/



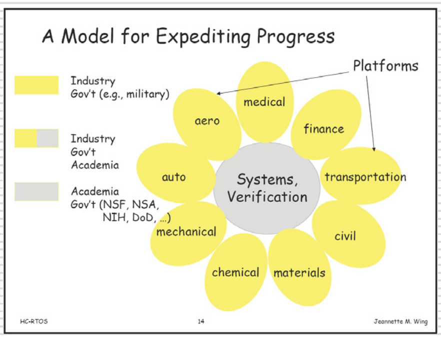
## CPS: Challenges

| | New abstraction layers for design | Semantic foundations for composing models | Composition platforms for heterogeneous systems | Predictability under limited compositionality | Foundation for system integration | Compositional certification | Agile design automation | Open Architectures | Reliable systems from unreliable components | Resiliency to cyber attacks |
|---|---|---|---|---|---|---|---|---|---|---|
| *Aerospace* | ■ | ■ | ■ | □ | ■ | ■ | ■ | ○ | ■ | ■ |
| *Automotive* | ■ | ■ | ■ | □ | ■ | ○ | ■ | □ | □ | ■ |
| *Defense* | ■ | ■ | ■ | □ | ■ | ■ | ■ | ■ | □ | ■ |
| *Energy* | ■ | □ | ■ | ■ | ■ | ○ | ○ | ■ | ■ | ■ |
| *Biomedical* | ■ | □ | ■ | ■ | ■ | ■ | □ | □ | ■ | ■ |

From executive summary of CPS Summit Report, April 2008



## CPS: Stakeholders






## CPS: A Global R&D Priority

- Federal Networking and Information Technology R&D
  *Formative planning workshops by HCSS coordinating group*
- The CSTB and the National Academies
  Sufficient Evidence? Building Certifiably Dependable Systems
- Forthcoming significant R&D funding
  *NSF started a seeding program in 2007*
- Computing Research Association
  *A focus area of the Computing Community Consortium*
- Convergence of RT, SN, Emsoft, Hybrid and Control
  *First CPS Summit with co-located events held in late April*
- Beyond academia and beyond the US
  *SCADA*, *Automotive*, *MD PnP*, *Aviation*, *EU ARTEMIS* & EPoSS



## How Could SNC Members Help?

- CPS research is domain-specific
  - We need domain experts for the modeling and control of physical (mechanical, electromagnetic, biological, chemical, biophysical, biochemical, medical, …) phenomena
  - We need test-beds and opportunities for student training
- What is *your* CPS application?



## **Cyber-Physical Systems**

Challenges & Opportunities

**Azer Bestavros**

Computer Science Department
Boston University

http://www.cs.bu.edu/groups/wing

Sensor Network Consortium
May 8, 2008