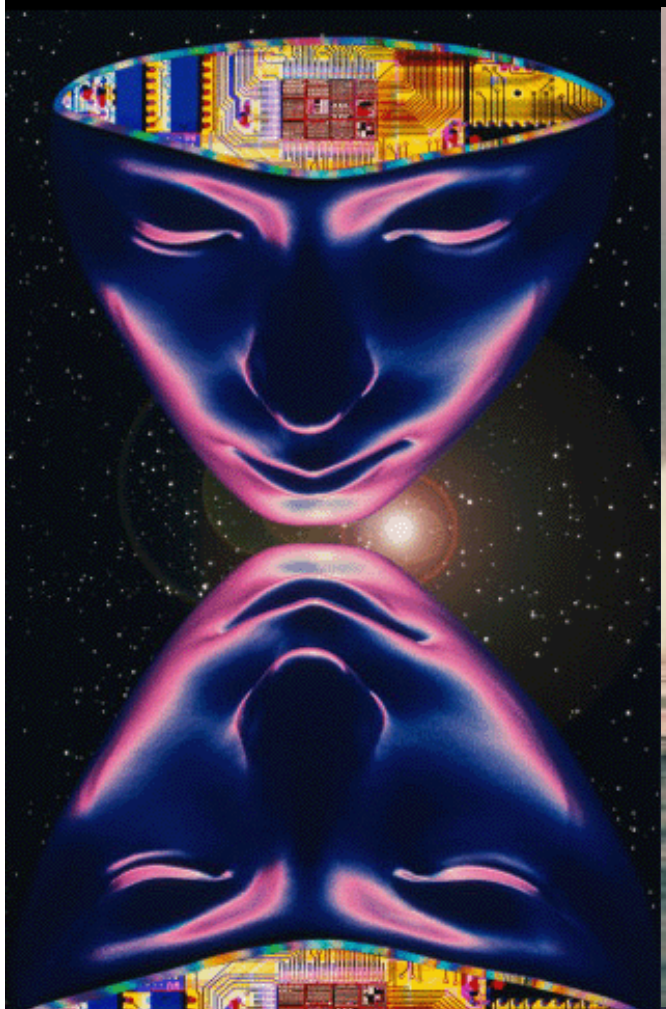


A Graduate Embedded System Education Program

Alberto Sangiovanni-Vincentelli

Department of EECS,
University of California at Berkeley

The Killer Applications for the Future?




- Energy Conservation
- Emergency Response and Homeland Defense
- Transportation Efficiency
- Monitoring Health Care
- Land and Environment
- Education


Needs for Electronics Industry of the Future

- MEMS, Analog and RF devices
- Scalable computing architectures
- Networked-oriented operating systems
- Distributed file systems
- Data management systems
- Security/privacy
- User interfaces
- Collaboration applications
- Intelligent learning systems
- Program verification
- Methodologies for HW/SW design/evaluation

The single most serious problem in the Valley today is to find PEOPLE with the right expertise!

Design “Practice”

Design Science

Innovators with solid scientific foundations

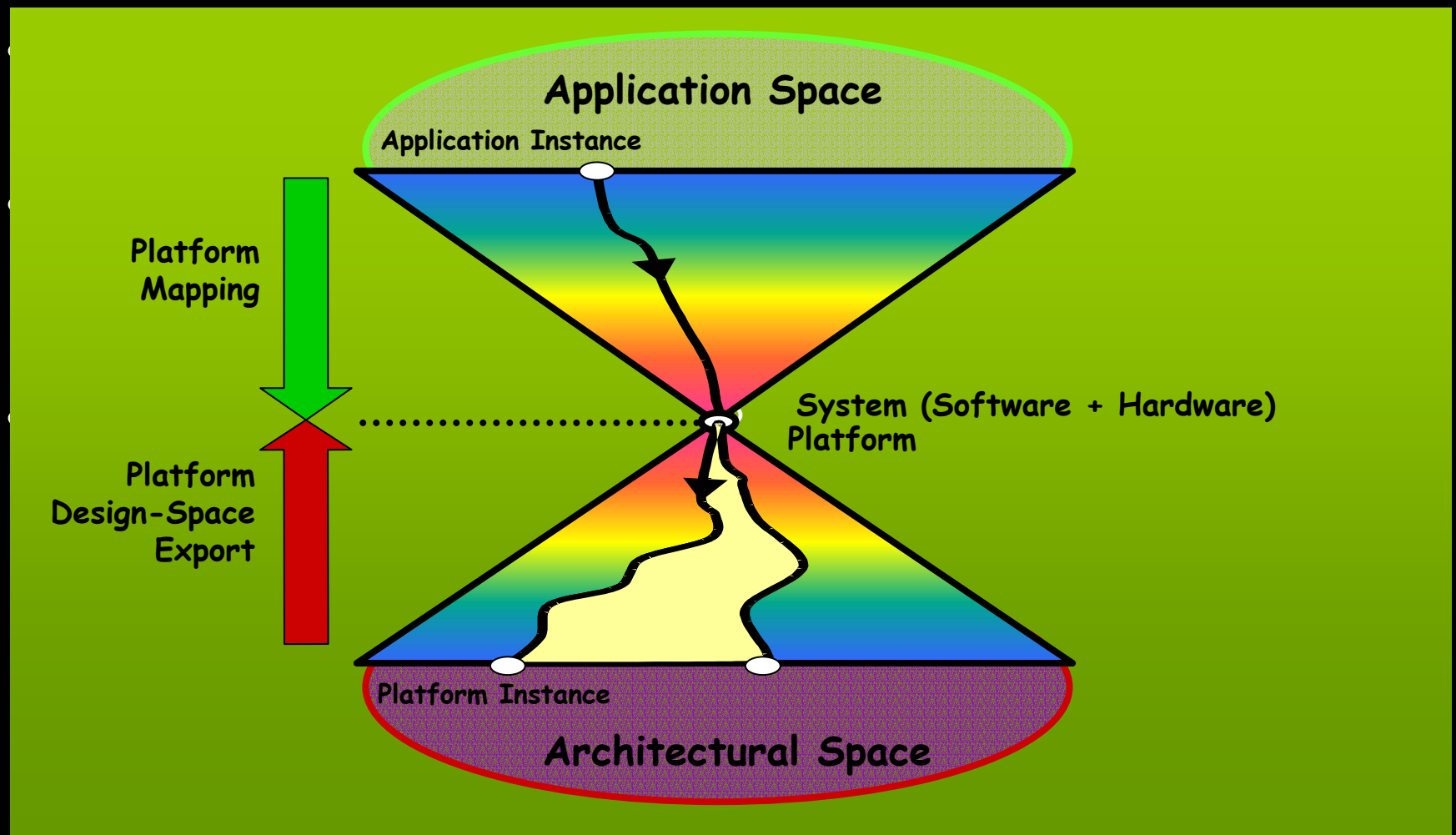
- Innovation in such a complex world must come from deep understanding of basic issues
- Do not mistake techniques for principles!
- Ad hoc engineering solutions should be avoided at all costs
- Balance of foundations and experience

Dr. Right

A set of Graduate Courses

- EE249: Design of Embedded Systems
- EE290O: Embedded Software Design
- EE290N: Concurrent Models of Computation
- CS298-4: Formal Methods for Software Reliability
- CS290A: Ubiquitous Systems
- CS290D: Oceanic Systems
- EECS290F: Dependable Computing and National Security

Electronic Design: A Vision
Platform-Based Design



Platforms: Evolution

In general, a platform is an abstraction layer that covers a *number of possible refinements (platform instances) into a lower level.* The platform representation is a library of components including interconnects from which the lower level refinement can choose (as such is a set of designs).

Platform stack {

Platform

Mapping Tools

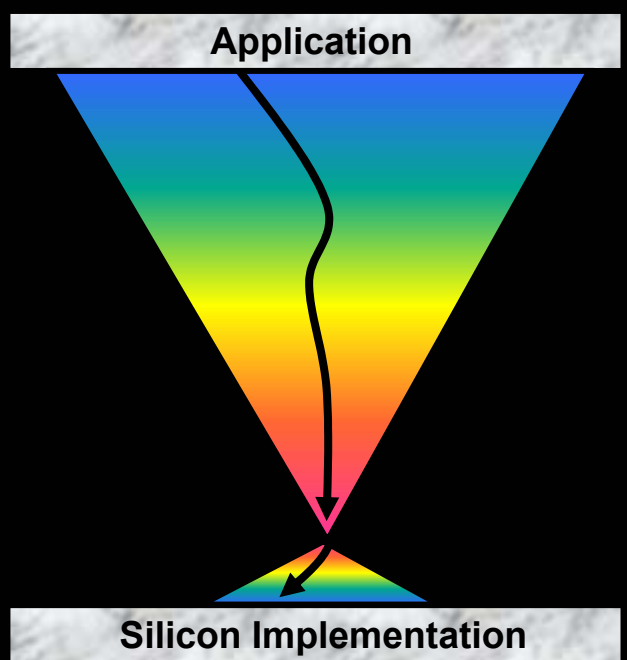
Platform

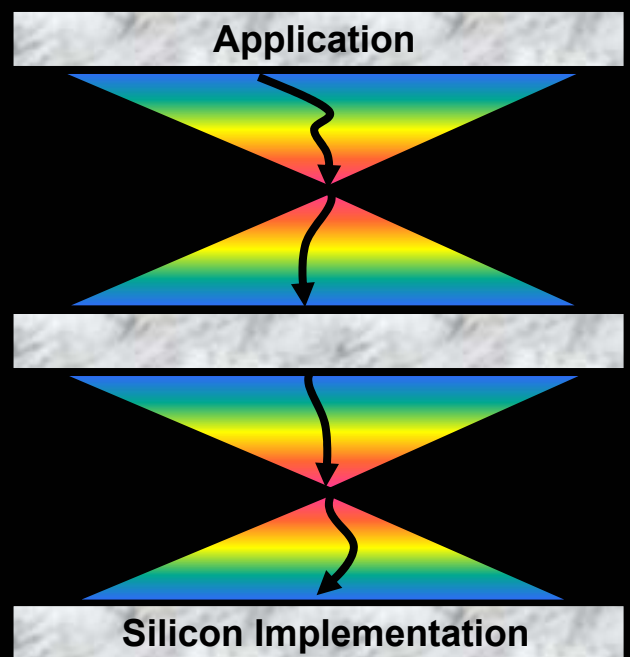
Principles of Platform methodology: Meet-in-the-Middle

- Top-Down:
  - Define a set of abstraction layers
  - From specifications at a given level, select a solution (controls, components) in terms of *components (Platforms)* of the following layer and propagate constraints
- Bottom-Up:
  - Platform components (e.g., micro-controller, RTOS, communication primitives) at a given level are abstracted to a higher level by their functionality and a set of parameters that help guiding the solution selection process. The selection process is equivalent to a covering problem if a common semantic domain is used.

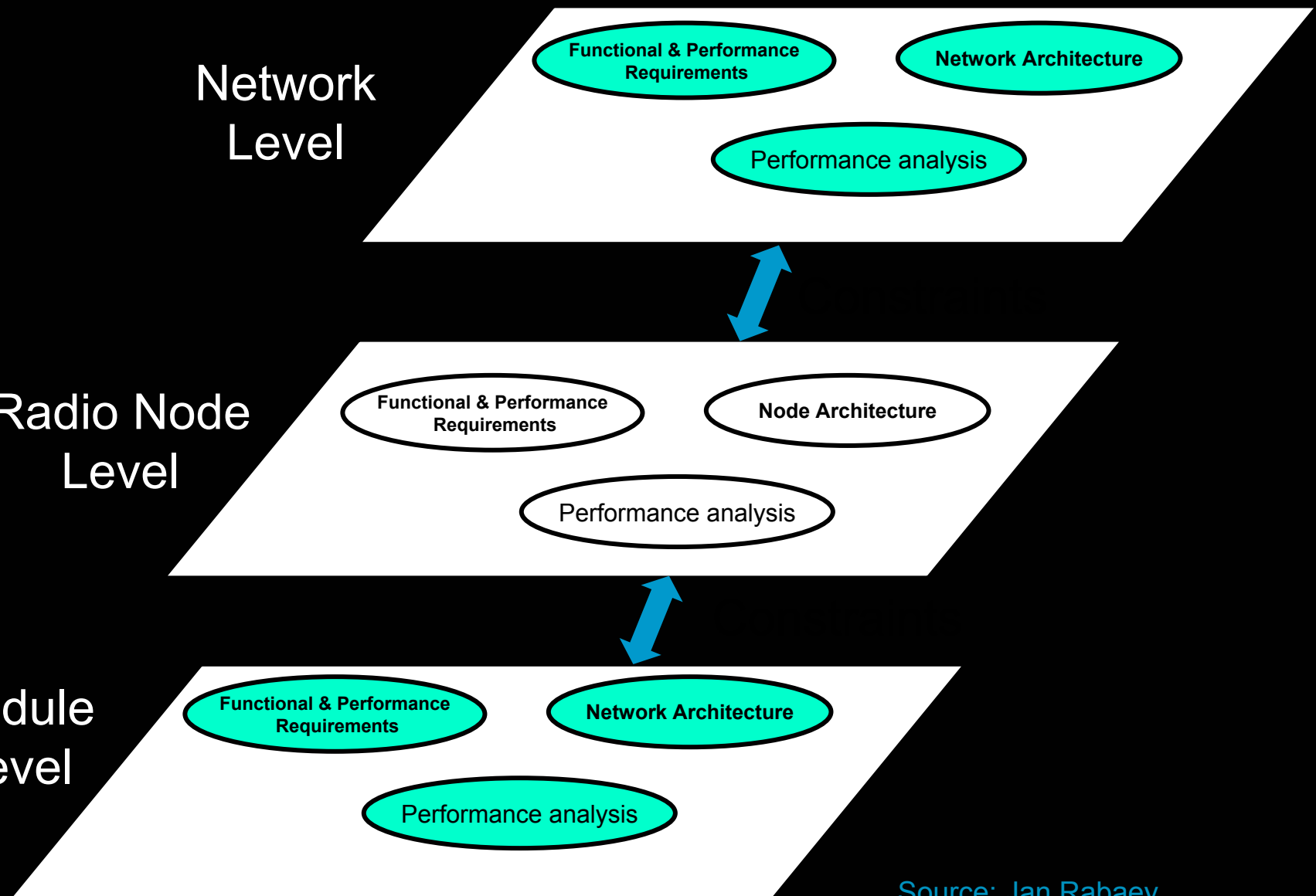
Platform-Based Implementation

- Platforms eliminate *large loop iterations* for affordable design
- Restrict design space via new forms of regularity and structure that surrender *some* design potential for lower cost and first-pass success
- **The number and location of intermediate platforms is the essence of platform-based design**



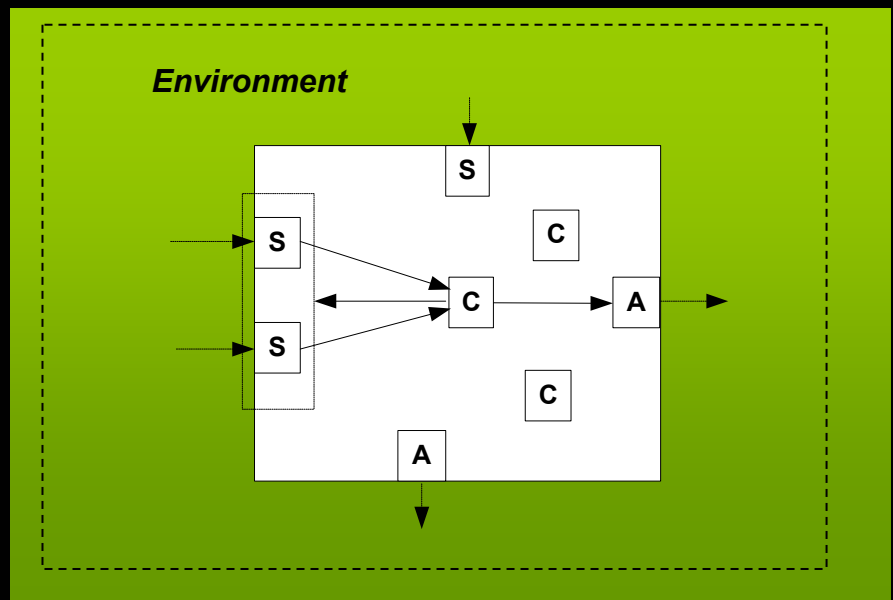


A Hierarchical Application of the Paradigm: The Fractal Nature of Design!



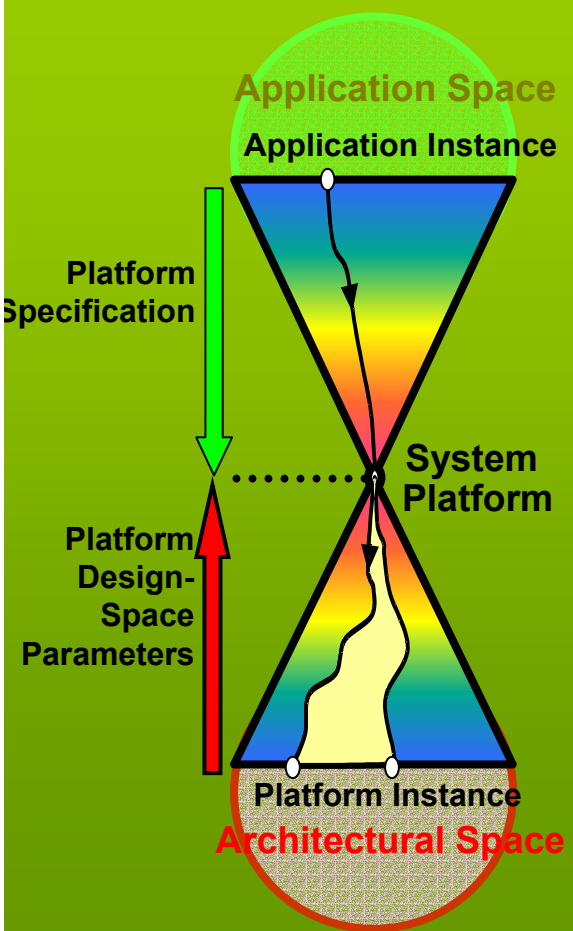
Source: Jan Rabaey

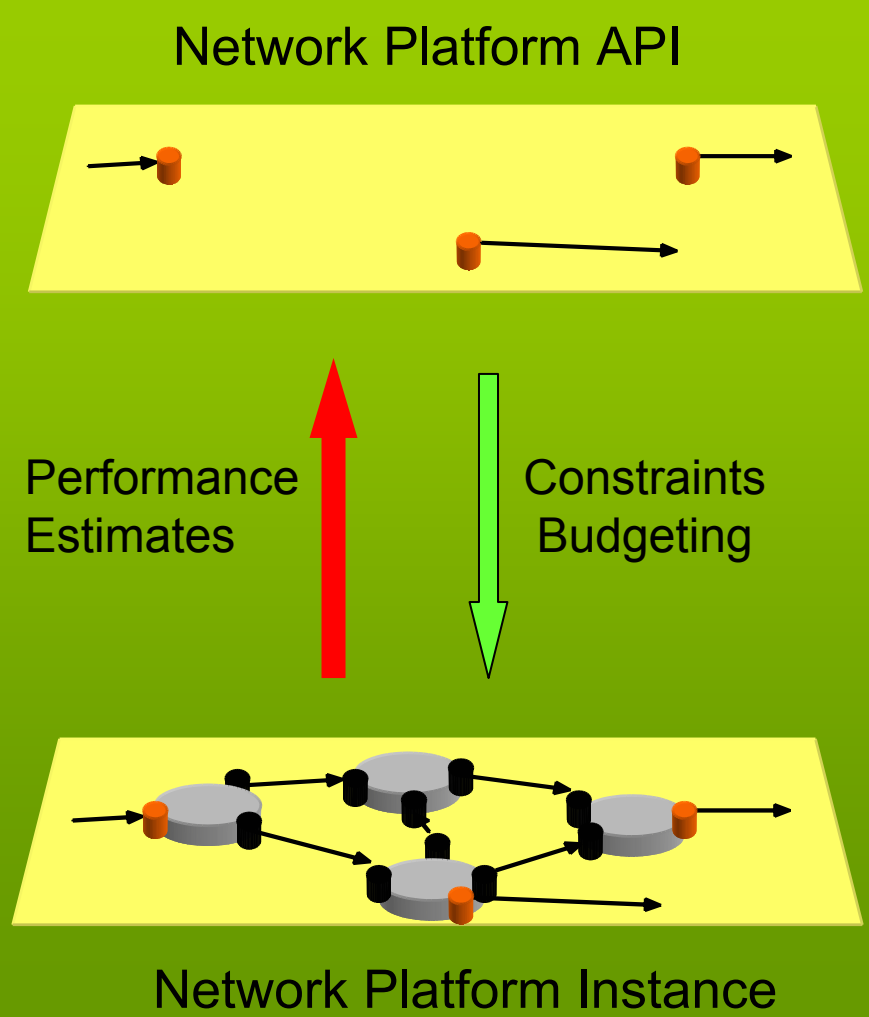
Sensor Network Applications



- ***Application*** - collection of sensors, controllers and actuators cooperating to achieve a common goal (environment control or monitoring)
- ***Sensor*** - measures the state of the environment
  - Parameters: phy. quantity, range, accuracy, ID, location...
- ***Actuator*** - sets the state of the environment
- ***Controller*** - gets the state of sensors and decides whether and how to set the state of actuators

Network Platforms





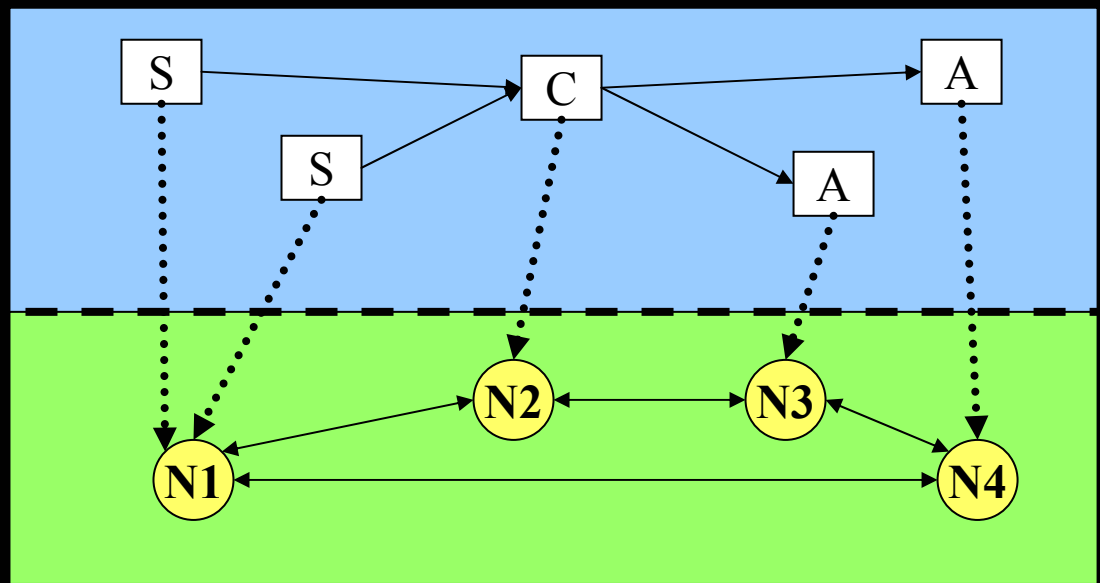
Communication Services:

- CS1:
  Lossy Broadcast
  Error rate: 33%
  Max Delay: 30 ms
- CS2: …

NP components:

- node
- link
- port
- NPI I/O port

Sensor Network Platform

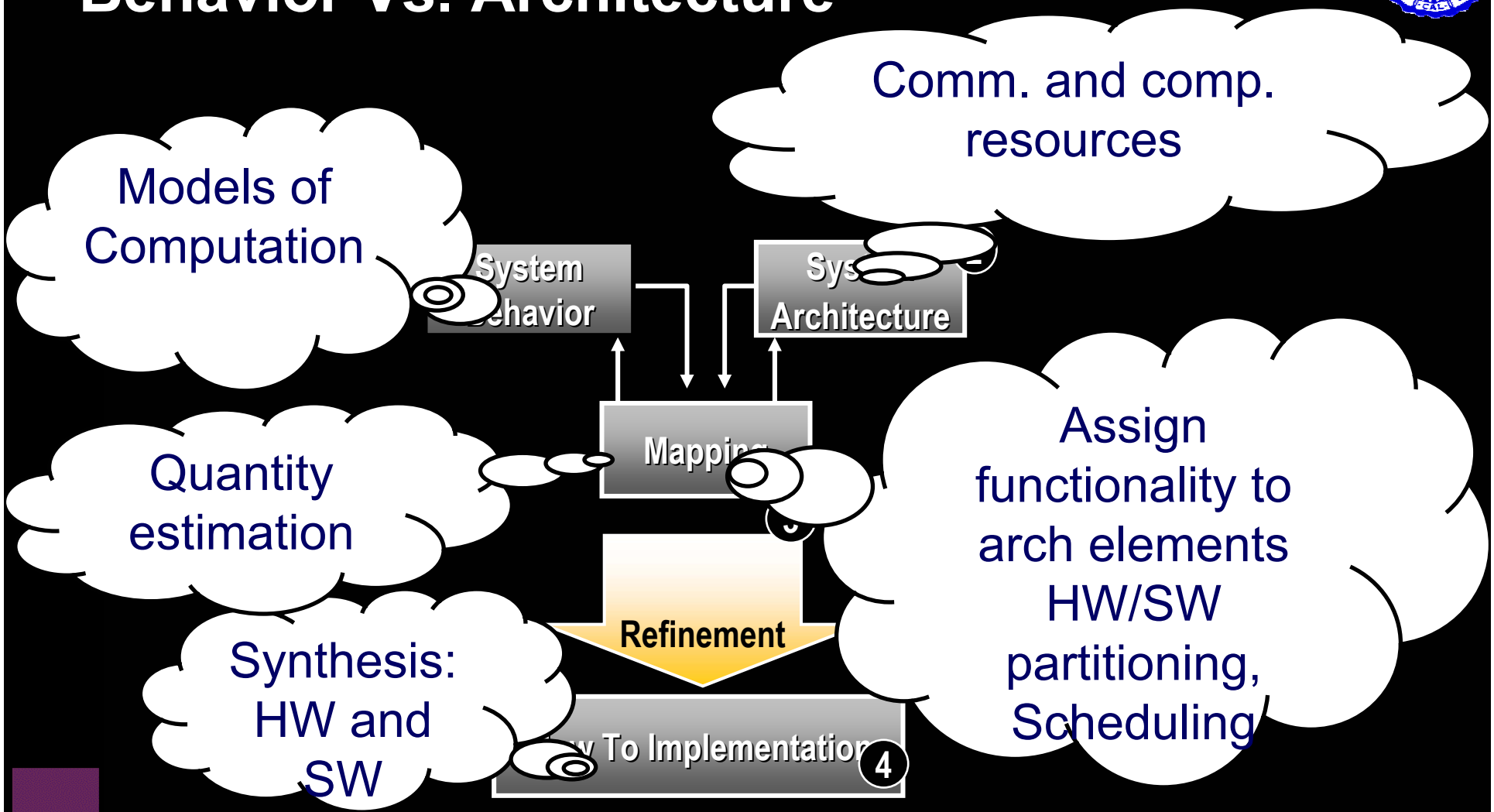


- Node
  - computation and communication platform (memory, processors…)
  - sensor/actuator devices
- Parameters:
  - memory available,
  - energy level,
  - computation/communication cost…
- Sensors, Controllers and Actuators mapped onto nodes

EE249

- A graduate 4 unit system design course:
- Emphasis on understanding of system design
  - The basic mathematical models representing system behavior independent of implementation, TSM, Abstract Algebraic Approach
  - Implementation as choice of architecture
  - Architecture as platform
  - Mapping of behavior into architecture as an exercise in design exploration
  - Software and hardware seen uniformly
  - Hands-on experience on industrial and University tools (Ptolemy, Giotto, Mescal, Matlab, Cadence VCC, WindRiver, Xilinx, Cypress, Polis, Metropolis)
  - Final Projects: Design. Methodology and Tools (most published in Conferences and Journals)

Behavior Vs. Architecture



Behavior Vs. Communication

- Clear separation between functionality and interaction model
- Maximize reuse in different environments, change only interaction model

Outline of the course

- Part 1. Introduction: Future of Information Technology, System Design, IP-based Design, System-on-Chip and Industrial Trends
- Part 2. Design Methodology: Platform-based Design
- Part 3. Functional Design: Models of Computation
- Part 4. Architecture Design: Capture and Modeling
- Part 5. Exploration and Mapping
- Part 6. Implementation Verification and Synthesis, Hardware and Software

Discussion sections

- Lab section (Th. 4-6):
  - tool presentations
- Discussion Session (Tu. 5-6)
  - students' presentation of selected papers
    - Each student is required to fill in a questionnaire in class for each discussion session
    - Each student (in groups of 2-3 people) has to make an oral presentation once during the class

Week	Lab Sections	Homeworks
1	- - -	- - -
2	Tool presentation	HW1
3	Discussion	
4	Tool presentation	HW2
5	Discussion	
6	Tool presentation	HW3
7	Discussion	
8	Tool presentation	HW4
9	Discussion	
10	Tool presentation	HW5
11	Discussion	
12	Tool presentation	HW6
13	Discussion	
14		HW7
15		

Reaching a Consensus on a Curriculum?

- Determine partition between undergraduate and graduate program
- Determine the partition between foundations and application areas
- Should a graduate program be established on the foundations of embedded system and should coordinated programs be instituted in the application domains (e.g., mechanical engineering, civil engineering)?
- How to establish a continuing education program for professionals
- What is the best mechanism to coordinate across the Ocean?